



**INFOR[⚡]ATIKA
A FELSŐOKTATÁSBAN
2014**



Debrecen, 2014. augusztus 27–29.

Informatika a felsőoktatásban 2014 konferencia

Konferencia kiadvány

Szerkesztette:

Kunkli Roland
Papp Ildikó
Rutkovszky Edéné

Debrecen, 2014. augusztus 27-29.

ISBN 978-963-473-712-4

Első kiadás, 2014. augusztus

Debreceni Egyetem, Informatikai Kar
4032 Debrecen, Egyetem tér 1.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means: electronic, electrostatic, magnetic type, mechanical, photocopying, recording or otherwise, without written permission from the copyright holders.

Ez a konferencia kiadvány különböző előadások szerzőinek kézírata alapján készült az „Informatika a felsőoktatásban 2014” konferencia szervezőbizottsága által kért formában. Néhány kézirat formája azonban nem teljesen felelt meg az egységes formai kéréseknek. A kiadvány gyors elkészítésére rendelkezésre álló rövid idő miatt ezen hibák nem mindegyike volt javítható. Ezért, az esetlegesen felfedezett hibákért a fenti indokok alapján az érintett szerzőkkel együtt a tisztelt olvasó elnézését kérjük.

A szerkesztők.

Címlap grafika: Pikó Sándor

ELŐSZÓ

Tisztelt Olvasó!

A nyolcadik "Informatika a felsőoktatásban" konferencia előadásanyaga a kiadvány jelentős terjedelme miatt elektronikus változatban kerül kiadásra. Ebben az előadások olyan gyűjteményét találja, melyeknek az előző konferenciákhoz hasonlóan számos aktualitása van. A szakterület rohamos fejlődése, követése és az oktatás tartalmi, valamint módszertani fejlesztése mellett folyamatban van a felsőoktatás átalakulása, az elmúlt években új mester szakok alapítására és indítására illetve a korábbi szakok felülvizsgálatára került sor. A kiadvány az elmúlt években e területeken végzett munkák eredményeit adja közre, illetve aktuális és új feladatokról, témákról, kutatásokról ad áttekintést.

A konferencia programbizottsága, rendezői, szervezői remélik, hogy a kiadványban megtalálható tanulmányokat haszonnal fogják olvasni. A kiadvány nem készülhetett volna el a konferenciára előadással jelentkezett szerzők munkája nélkül, ezért köszönjük a szerzők munkáját.

A konferencia megrendezéséhez és a kiadvány elkészítéséhez jelentős anyagi segítséggel járultak hozzá a rendezvény támogatói, szponzorai. A kiadvány szerkesztői ezúton is köszönik a konferencia rendezői és bizottságai nevében támogatásukat.

A kiadvány elkészítésében részt vett kollégák, Dr. Papp Ildikó és Kunkli Roland lelkiismeretes munkáját is megköszönjük. Köszönet illeti Dr. Rutkovszky Edénét, aki a konferencia titkári feladatok mellett a kiadvány előkészítésének számos feladatát vállalta.

Vaszil György
a Programbizottság elnöke

Mihálydeák Tamás
a Szervezőbizottság elnöke

A konferencia fővédnöke

Palkovics László EMMI Felsőoktatásért felelős államtitkár

A konferencia védnökei

Kósa Lajos Debrecen polgármestere
Lovász László Magyar Tudományos Akadémia elnöke
Szilvássy Zoltán DE rektora

A konferencia elnöke

Demetrovics János egyetemi tanár, akadémikus, MTA SZTAKI / ELTE

Programbizottság

Elnök: Vaszil György DE

Tiszteletbeli elnök: Pethő Attila DE

Tagjai:

Arató Péter	BME	Kovács László	ME
Bacsárdi László	NyME	Kozmann György	VE
Benczúr András	ELTE	Ködmön József	DE EFK
Borgulya István	PTE	Liptai Kálmán	EKF
Buza Antal	DF	Molnár Sándor	SzIE
Csendes Tibor	SzTE	Nagy Miklós	NIIF
Csirik János	SzTE	Pap László	BME
Csuhaj Varjú Erzsébet	ELTE IK	Racskó Péter	BCE
Dobay Péter	PTE	Raffai Mária	SzE
Fazekas Attila	DE IK	Rónyai Lajos	MTA SZTAKI
Fazekas István	DE IK	Rudas Imre	Óbudai E.
Fülöp Zoltán	SzTE	Sallai Gyula	BME
Gábor András	BCE	Selényi Endre	BME
Gilányi Attila	DE IK	Sima Dezső	Óbudai E
Gyimóthy Tibor	SzTE	Szelezsán János	GDF
Györfi László	BME	Sziray József	SzE
Hajdu András	DE IK	Szolgay Péter	PPKE
Hartung Ferenc	PE	Sztrik János	DE IK
Herdon Miklós	DE GTK	Terdik György	DE IK
Horváth Zoltán	ELTE IK	Tóth Tibor	ME
Ispány Márton	DE IK	Vajta László	BME
Jereb László	NyME	Végh János	DE IK
Kormos János	DE IK		

Szervezőbizottság

Elnök: Mihálydeák Tamás DE IK

Tiszteletbeli elnök: Herdon Miklós DE GTK

Titkár: Rutkovszky Edéné DE IK, NJSZT


Tagjai:

Kádek Tamás	DE IK	Matolcsy Zoltán	DE IK
Kása Ernőné	DE IK	Papp Ildikó	DE IK
Kunkli Roland	DE IK	Várallyai László	DE GVK

Szponzorok/Támogatók



SERVICES
HUNGARY

Member of -Systems

IT Services Hungary Kft



Neumann János Számítógép-tudományi Társaság



SAS Institute Kft



OMIKRON Informatika Kft



IBM Magyarország Kft



Schönherz Informatikai Stúdió



Jövő Internet kutatások az elmélettől az alkalmazásig projekt

TÁMOP-4.2.2.C-11/1/ KONV-2012-0001

Jövő Internet kutatások
az elmélettől az alkalmazásig

TARTALOM

PLENÁRIS ELŐADÁSOK

Pajor Ferenc: Mit vár az ipar - 5 év múlva az egyetemen végzett informatikusoktól? 11

SZEKCIÓ ELŐADÁSOK

ALGORITMUSOK

Borbola Gábor: Kombinatorikus geometriai objektumok vizsgálata 13

Borgulya István: Egy evolúciós heurisztika a 2D lap-pakolási problémára..... 23

Iványi Péter: A végeselem hálók párhuzamos konverziója..... 30

Varga Tamás, Bacszárdi László: Kvantumkeresés mérnöki megközelítésben..... 39

EGÉSZSÉGÜGY ÉS ESÉLYEGYENLŐSÉG

Balla Tibor: Mozgásérzékelős vezérlőn alapuló rehabilitációs rendszer fejlesztése..... 47

Kaczur Sándor: A PHYSIOBANK fiziológiai jelfeldolgozó szoftvereinek hatékonysága..... 54

Kovács Zsigmond, Maros-Szabó Zsuzsanna:

Informatika az egészségügyben – Személyes Kórlap (Personal Health Record) 60

E-LEARNING

Ambrusné Somogyi Kornélia: Vizsgáztatás E-learning rendszer segítségével..... 67

Hegyesi Franciska : E-learning stratégiák az Óbudai Egyetem felnőttképzésben 76

Koós Dániel, Várterész Magda: E-learning a számítástudomány oktatásában 88

Kupcsikné Fitus Ilona: Innovatív tanulásszervezés a Gábor Dénes Főiskolán 96

Kusper Gábor: Nyílt online tömegkurzusok módszertani kérdései 107

FUTURE INTERNET

Bánlaki József, Geda Gábor, Juhász Tibor, Király Roland, Kovács Emőd, Radványi Tibor:

Az EKF FutureRFID - Az RFID/NFC technológia továbbfejlesztési lehetőségei az „Internet of Things” koncepciói mentén - kutatási projektjének bemutatása 116

Gál Zoltán: Szenzor adatklaszterek wavelet elemzése 123

Kádek Tamás, Kósa Márk, Kollár Lajos: Intelligens egyetemi campus 132

Somodi Krisztián, Kádek Tamás, Kollár Lajos, Kósa Márk:

Smart Campus alkalmazás fejlesztése Android platformon 138

Vámosi Attila, Kocsis Imre, Deák Krisztián: Az SVM módozatainak alkalmazási lehetősége
műszaki optimalizációs problémák megoldásában..... 148

IKT A TUDÁS ÉS A TANULÁS VILÁGÁBAN SZIMPÓZIUM

Göncziné Kapros Katalin: Preferált algoritmus leíró eszközök 160

Kvaszingerné Prantner Csilla:

Új, ingyenesen elérhető oktatói portálrendszer a felsőoktatásban..... 171

Lengyelné Molnár Tünde:

A digitális könyvtári hálózat szolgáltatásainak fejlesztése és kutatása 179

Racskó Réka: Kompetenciák az elektronikus tanulási környezetekben a humán

teljesítménytámogató technológiai kutatások szemszögéből 192

INFORMÁCIÓS TÁRSADALOM

Bátfai Erika, Bátfai Norbert: Nem vagy Anonymus 204

Borbély Mária:

Az élethelyzetnek megfelelő aktív internet alkalmazások frissdiplomások körében 209

Csernoch Mária, Biró Piroska, Máth János, Abari Kálmán: Mit tudok Informatikából? 217

Dani Erzsébet: A kétfázisú HY-DE-modell: a hiper- és mélyfigyelem fázisváltásai a katedrától a hallgatói önfejlesztésig 231

Dobay Péter, Kruzsliz Ferenc :

Startup, vagy munkahely? IKT vállalkozások feltételrendszerének egy modellje 239

Godó Zoltán Attila, Kocsis Dénes, Kiss Gábor, Stóka György:

Algoritmusalkotó-készség újszerű vizsgálata 252

Nyéki Lajos: Szövegbányászati módszerek alkalmazása távoktatási fórum üzenetek elemzésében 259

INFORMATIKAI ALKALMAZÁSOK

Erdős Ferenc: ERP-választás KKV környezetben 269

Gludovatz Attila, Bacskárdi László:

Termelésinformatikai eszközök hazai gyárak működésében 278

Herdon Miklós, Charles Burriel, Tamás János, Várallyai László, Lengyel Péter, Pancsira János:

Kollaboratív környezet és ismeretbázis építése a mezőgazdasági erdészeti oktatására az AgroFE Leonardo nemzetközi projektben 287

Mátyás Anita, Szemán Nagy Gábor, Szeghalmy Szilvia:

TIME-LAPSE felvételek elemzése fonalgombba növekedésének vizsgálatához 295

Szilágyi Róbert, Herdon Miklós, Lengyel Péter, Rózsa Tünde:

Tudás disszemináció és innovatív információs technológiák az agrárgazdaságban 302

Zichar Marianna: Python kontra ArcGIS 311

INFRASTRUKTÚRA

Kazinczy Tamás: Storage szolgáltatás bővítése az NIIF Intézetnél 316

INTÉZMÉNYMENEDZSMENT, OKTATÁS ÉS CÉGEK

Dinnyés Álmos, Doktor Zsuzsanna:

Kollaborációs eszközök alkalmazása K+F projekt portfólió kezelésében 320

Szabó Bernadett: Sivák és brachmák a BigData korában 327

Tóth Sándor: ANSIBLE 331

IT BIZTONSÁG

Kovács Zita:

A bilineáris leképezéseken alapuló hibrid mixnet alkalmazása az e-vizsga rendszerekben..... 339

Radványi Tibor: Védekezés az RFID rendszerekben a lehallgatás ellen 347

Gyurák Gábor: Digitálisan aláírt dokumentumok hosszú távú tárolása..... 355

Pandur Béla: Radioaktív hulladékokat tároló rendszer adatainak hosszú távú megőrzése..... 365

KÖNYVTÁR ÉS INFORMATIKA

Bátfai Erika, Bátfai Norbert, Kőrösi Krisztina: OLVASÁS 2.0 – AZAZ AZ OLVASÁS

FELVILLANYOZ az e-könyvek helye az oktatásban és a könyvtárakban 374

Boda István, Tóth Erzsébet, Bényei Miklós, Csont István:

Kulturális örökség és virtuális valóság – az ókori Alexandriai Könyvtár 384

Némethi-Takács Margit: Plakátok leíró sémája 390

KUTATÁS

Bencsik Gergely, Bacszárdi László:

Statisztikai adatok közötti véletlen összefüggések tanulmányozása 403

Buza Antal, B. Kis Piroska: A mátrix faktorizáció alkalmazásának korlátai 413

Csajbók Zoltán Ernő, Mihálydeák Tamás: Membrán számítások határokkal 418

Gonda János: Néhány megjegyzés a polinomszerű Boole-függvényekhez..... 426

Dömösi Pál, Horváth Géza: Automataelméleti alapú titkosítási rendszerek 433

Kósa Balázs, Kiss Attila:

Hálózatok kutatása a Tudáskezelő rendszerek labor tantárgy keretein belül az ELTE-n..... 442

Mucsics F. László:

Kísérlet a levelezős hallgatók költségmegtakarításának növelésére Sloodle alkalmazásával 453

MÉRNÖKINFORMATIKA ÉS HÁLÓZATOK

Almási Béla, Harman András:

Eredmények a több utas kommunikációs technológiák kutatásában..... 458

Bérczes Tamás, Sztrik János, Jinting Wang, Xuelu Zhang, Fang Wang, Horváth Ádám:

Mobil hálózatok véges forrású modellezése spectrum renting és handover hívások használatával 464

Deák Krisztián, Kocsis Imre, Vámosi Attila:

Gépek és gépelemek diagnosztikája, élettartam becslése gépi tanulás segítségével 470

Háber István, Tamás László: Gépjármű elektronika programozása RTOS rendszeren a

fogyasztáscsökkentés jegyében 477

Jancskárné Anweiler Ildikó, Várady Géza, Schiffer Ádám, Sári Zoltán:

NI ELVIS & QUANSER modellek a számítógépvezérelt-irányítások oktatásában 488

Szakonyi Lajos, Sári Zoltán: Hőtechnikai folyamatszimuláció lakótérben 497

Szakonyi Lajos: Intelligens lakótér folyamatdinamikája 514

<i>Várady Géza, Sári Zoltán, Schiffer Ádám, Jancskár Ildikó:</i> Intelligens épületfelügyelet autonóm drónnal	534
<i>Varga Imre:</i> Hardverközeli programozás oktatása a DIY Calculator segítségével	540

MÓDSZERTAN

<i>Achs Ágnes:</i> Humor a programozás oktatásában.....	547
<i>Aszalós László:</i> Mesterséges intelligencia oktatása Python segítségével	558
<i>Baráth Áron, Cséri Tamás, Horváth Gábor, Pataki Norbert, Porkoláb Zoltán, Simon Melinda, Szügyi Zalán:</i> A projekteszközök oktatásának didaktikai kérdései	566
<i>Bujáki Attila, Balla Katalin:</i> Agilis szoftverfejlesztés agilis oktatása.....	572
<i>Daiki Tennó, Szlávi Péter, Zsakó László:</i> A szimuláció, mint oktatási eszköz	589
<i>Kocsis Imre, Tóth László, Vámosi Attila, Deák Krisztián:</i> A matematikai szoftverek szerepe a mérnöki problémamegoldás átalakulásában.....	600
<i>Kuki Attila:</i> OPNET® eszközök a hálózatmodellezés oktatásában.....	612
<i>Pántya Róbert:</i> Logikai programozás oktatásának szemléltetése flash animációk segítségével.....	619
<i>Papp Ildikó:</i> Térszemlélet fejlesztése a műszaki képzésben	629
<i>Sári Zoltán, Schiffer Ádám, Várady Géza, Jancskárné Anweiler Ildikó:</i> Kvadkopter drón az oktatásban és kutatásban.....	639
<i>Szabó György, Czinkóczky Anna:</i> Az informatikai kompetencia aktuális kérdései a műszaki felsőoktatásban.....	645
<i>Szendrői Etelka:</i> Adatbázisok, adatbázis alapú programozás oktatása.....	651

OKTATÁS

<i>Bakó Mária, Várallyai László:</i> Szoftverfejlesztés (programozás) az Informatikus Szakigazgatási Agrármérnök szakon	659
<i>Balázs Péter, Katona Endre, Kató Zoltán, Nagy Antal, Németh Gábor, Nyúl László, Palágyi Kálmán, Tanács Attila, Varga László Gábor:</i> Képfeldolgozás a szegedi informatikus-képzésben.....	667
<i>Biró Piroska:</i> CAEDUS oktatási keretrendszer	676
<i>Fazekas K. Anna, Fazekas Gábor, Várterész Magda:</i> A számítógéppel támogatott oktatás egy SQC alapú minőségbiztosítási modelljéről	689
<i>Friedel Attila:</i> Mikrokontroller programozó diákműhely a Gábor Dénes Főiskolán	694
<i>Gonda János, Horváth Zoltán, Pap László:</i> Csatlakozás az európai tudásközpontokhoz.....	703
<i>Gyarmati Péter:</i> Bevezetés a hálózat tudományba.....	713
<i>Kollár Lajos:</i> Programozási gyakorlófeladatok és a tömegbe kiszervezés	717
<i>Michelberger Pál:</i> Információvédelmi képzés – egy próbát megér (?).....	724
<i>Nagy Sára:</i> Ismeretalapú keretrendszerek oktatása az ELTE Informatikai Karán	730

Pozsgai Tamás, Lipovits Ágnes, Gál Balázs:

Felhő alapú szoftverek használata az informatikatanár képzésben 734

Robu Judit, Soós Anna: Magyar informatikusok képzés a Babeş-Bolyai Tudományegyetemen..... 739

Schiffer Ádám, Sári Zoltán, Sári Zoltán, Várady Géza, Jancskár Ildikó:

Grafikus programozás a mérnök informatikus képzésben 745

Szimkovics Tamás:

Középiskolai informatikai tehetséggondozás az ukrainai magyar tannyelvű iskolákban..... 752

PROGRAMOZÁS, ADATBÁZISOK, MESTERSÉGES INTELLIGENCIA

Demetrovics Zsolt, Lőrincsik Éva, Márkus Zsolt László, Szkaliczki Tibor, Szántó György:

UNI@HAND – offline mobil alkalmazás az egyetem és a hallgatók támogatására 760

Kádek Tamás, Pánovics János: A JavaScript nem Java?..... 767

Micsik András, Gárdos Judit: Tudományos repozitóriumok az MTA-ban: a KDK és a SZTAKI tanulságai..... 773

Piller Imre, Fegyverneki Sándor: Alternatív dokumentumszervezési módszerek 782

Piller Imre, Fegyverneki Sándor: Felhő alapú dokumentumkezelő rendszerek..... 787

Takács Péter, Csajbók Zoltán Ernő, Mihálydeák Tamás: Multiset számítások R-ben 793

SW FEJLESZTÉS ÉS TESZTELÉS

Boda Béla: CoSTiP Szeminárium - ISTQB Syllabus a gyakorlatban 815

Illés Zoltán, Heizlerné Bakonyi Viktória: Szoftvertechnológia - mobilfejlesztés 819

Molnár András, Horváth Ádám:

DSPNSim: Szimulációs Szoftver Determinisztikus és Sztocasztikus Petri Hálókra..... 826

Windisch Gergely, Fend Péter: Calliope - szakdolgozat nyilvántartó és követő rendszer..... 834

SZAKOK, TANANYAGFEJLESZTÉSEK

Balla Tibor, Kocsis Gergely: Elektronikus Oktatási Környezetek tananyag bemutatása 842

Herdon Miklós, Pető Károly, Várallyai László, Szilágyi Róbert:

BSc és MSc szintű agrárinformatikai képzési programok a Debreceni Egyetemen 848

Kuki Attila: Mérnök-informatikus mesterképzés a Debreceni Egyetemen..... 856

Szolgay Péter, Szederkényi Gábor, Nyékyné Gaizler Judit:

Az osztatlan képzés lehetőségei és kihívásai a mérnök-informatikus képzésben..... 863

Zsakó László: Osztatlan informatika tanárszak az ELTE-n..... 867

WEB

Fleiner Rita, Micsik András: Linked Open Data az egyetemen 872

Pataki Máté, Micsik András, Kovács László, Szabó Mihály:

KOPI-Fotó: Plágiumkeresés egy lefotózott oldal alapján 880

Takács Péter: Infografika - rétegszemléletben..... 886

Mit vár az ipar - 5 év múlva az egyetemen végzett informatikusoktól?

Pajor Ferenc

National Instruments Hungary

Ferenc.pajor@ni.com

Azt a felkérést kaptam, hogy foglaljam össze, milyen képességekkel rendelkezzen a végzős informatikus hallgató néhány év múlva. A teljes profil helyett az elvárások indoklása érdekében a működési környezetet foglaltam össze, valamint azokat a készségeket/képességeket, amelyeket fejleszteni javaslok. Mondhatom azt, hogy a végzős kollégák között mindig megtaláljuk a tehetséges és elhivatott kollégákat, akikre 13 éve építjük az IT osztályunka

Kulcsszavak: végzős hallgatók, elvárások, képességek.

1. Bevezetés

Azt, hogy az egyemről kikerülő végzősöktől mit várunk el, azt azok a körülmények szabják meg, amelyek között dolgozniuk kell, illetve ahol sikeresnek kell lenniük. A körülmények az NI gazdasági modelljéből indulnak ki. Ennek az alapvető eleme, az NI-nak legalább évi 10%-kal növekednie kell. Ez ad lehetőséget a dolgozóinak a fejlődésére szakmailag és anyagilag is. A folyamatos fejlődés tehát alapkövetelmény. Lévéen, hogy high tech cég vagyunk a kutatás – fejlesztés és innováció a fejlődésünk motorja.

Ha az NI IT-re fókuszálunk , akkor a legfontosabb tényezők:

- Az NI teljesen IT alapon működik, (IT enabled)
- Az NI IT insource-olt, vagyis a hw rendszerek üzemeltetésétől kezdve a komplex sw fejlesztésig 95%-ban magunk végzünk el IT tevékenységeket, tehát mind vertikális és horizontális tudásra szükség van
- Nagyon lényeges, az IT-nak együtt kell működnie a társosztályokkal, ha úgy tetszik az Üzlettel, hogy kiszolgálja azt.

2. A siker modellje

A siker modell, amiből kiindulunk meglehetősen egyszerű, két tényezőből tevődik össze. Ezek a motiváció és a szakmai tudás. Kiegészítői egymásnak és együtt is kell járniuk. Tulajdonképp a felvételi interjúk során is ezt, vagy az erre való törekvést, hajlamot keressünk a hozzánk jelentkezőkben a tehetség megléte mellett. (Az NI IT 100 főből áll és ebből 70-80 % a Debreceni Egyetemről került ki).Ha a sokéves tapasztalatot nézzük, a motiváció egy kicsivel fontosabb, mert az segít megszerezni a tudást. Ha meg van a motiváció, akkor a tehetség határozza azt meg, hogy valakiből jó, vagy kiemelkedő szakember lesz, adott szakmai tudás mellett.

Az egyetemről kikerült ifjú kollégáknál a motivációt kétféle módon nézzük meg, vannak-e céljai és tudja-e, hogyha munkát vállal, milyen munkát kell végeznie, milyen környezetben. A tapasztalatom az, hogy azt a környezetet, amiben tevékenykednie kell és a leendő feladatait nem túlzottan ismeri. Vagyis hacsak nem az egyetemen marad az illető célszerű lenne

megismerni azt, hogy egy a versenyszektorban működő IT hogyan működik, és milyen tevékenységeket végez egy fejlesztő, vagy egy rendszeradminisztrátor. Erre elvileg a szakmai gyakorlat lenne szánva, de ezt kevésnek érzem. Megfelelő, a gyakorlatban megszerzett tapasztalat segít abban, hogy fel lehessen készülni a leendő feladatokra és munkakörnyezetre.

A mi környezetünk fontos tényezője a céltudatosság, vagyis a reális célok kitűzése és elérése. Ezt viszonylag egyszerűen el lehet érni, gyakorlatilag mindenkinek írásos céljai vannak, amelyek mentén való haladást folyamatosan figyelemmel kísérjük.

A másik alapvető sikertényező a szakmai tudás, ami pályakezdők esetén a technológiák elsajátításának képességét jelenti, de e mellett a szükséges kreativitást és innovációs hajlamot is ide értem. Az egyetemi záróvizsgán a diplomamunkák értékelésekor annak újdonságát, hozzáadott értékét szoktam vizsgálni. Amit láttam, az a nagy különbség az innovatív és kevésbé innovatív kollégák között. Mondanom sem kell, hogy az innovatív beállítottságú kollégák sikeresebbek.

A szakmai képességeket lehet csak a számítástechnikával kapcsolatos képességekre szűkíteni, de tapasztalataink szerint az soft skilleknek is nagyon lényeges szerepük van.

A Soft skill fogalma összefüggésben van az un. EQ-val (Emotional Intelligence Quotient) és több tényező összessége, mint : személyiség jegyek, szociális érzékenység, kommunikáció, nyelvhasználat, optimizmus, másokkal való kapcsolat . Ez a mi gyakorlatunk szerint tudatosan képezhető és javítható.

Vagyis a képességek a környezet emberi oldalával való együttműködésre. AZ NI-nál sikeres IT-sok mind jó soft skillekkel rendelkeznek és a fejlesztésük része a tréningprogramoknak.

3. A fejlesztendő területek meghatározása

Készülve a cikkekre és az előadásra, vezető kollégáimmal összeszedtük azokat a képességeket és attitűdöt és tudástényezőket, amelyeket a mostani végzős diákok estében erősítenénk, fejlesztenénk

- Nyitott szellemű, az innovációkra képes, rugalmas gondolkodásmódú
- Képes legyen a folyamatos fejlődésre, keresse a lehetőségeket
- Képes csak részben definiált környezetben is dolgozni.
- Célok állítása és céltudatosság legyen jellemzője
- Önismerettel, önértékeléssel rendelkezék,
- Képes legyen az új ismeretek gyors és hatékony –önálló – elsajátítására.
- Ismerjen legalább egy olyan IT működését, amely a gyakorlatban tevékenykedik/ jóval több gyakorlati ismeret
- Alapszinten a leggyakoribb technológiák ismerete
- Legyen birtokában az alapvető soft skilleneknek. (pl. kommunikáció, üzleti angol ismeret,

4. Köszönetnyilvánítás

Rendhagyó módon nem azoknak szeretnék köszönetet mondani, akik ennek a cikknek a megírásában segítettek, hanem a Debreceni Egyetem Informatikai Karának, egyrészt az együttműködését, másrészt azért , hogy tehetséges végzősök képzésével lehetővé tették, hogy az NI IT 5 ről 100 főre növekedjék, másrészt, hogy sikeresek legyünk.

Kombinatorikus geometriai objektumok vizsgálata

Inspection of the combinatorial geometric objects

Borbola Gábor ^a

^aSzent István Egyetem, Gazdasági, Agrár- és Egészségtudományi Kar
(Debreceni Egyetem, Informatikai Tudományok Doktori Iskola)
borbola.gabor@gk.szie.hu

Absztrakt: n -oldalú goligonok nevezzük azokat a síkbeli zárt törött vonalakat, melyek oldalainak hossza egymást követő pozitív egész számok sorozata $(1, 2, \dots, n)$, csúcsai a négyzetrács rácspontjai, az egymással szomszédos oldalak pedig derékszögben csatlakoznak egymáshoz.

A goligonok fogalmának egyik lehetséges általánosításához jutunk, ha módosítjuk a dimenziók számát, és a kétdimenziós törött vonalak helyett alacsonyabb, vagy magasabb dimenziószámokban keressük az előre rögzített oldalszámú geometriai objektumokat, ha ilyenek egyáltalán léteznek.

A Debreceni Egyetem Informatikai Tudományok Doktori Iskolájában végzett kutatás során olyan számítógépes algoritmus került kidolgozásra, mely a mesterséges intelligencia területéről származó backtrack keresővel módszeresen előállítja a számegyenesen, a négyzetrácson valamint a térbeli kockarácson a goligonokhoz hasonló geometriai alakzatokat. Az algoritmus összeszámolja az adott oldalszámú, egymással nem egybevágó zárt törött vonalakat, a talált alakzatok jellemző adatait pedig egy tovább vizsgálható adatbázisban rögzíti. Az előadás célja, hogy ismertesse az egy- és háromdimenziós esetre vonatkozó eredményeket. A prezentáció célja továbbá, hogy bemutassa a keresés során alkalmazott heurisztikákat, megvizsgálva azok hatékonyságát is.

Kulcsszavak: goligon, rácssokszög, zárt törött vonal, matematikai probléma megoldása számítógéppel, backtrack, heurisztika, algoritmikus hatékonyság

Abstract: n -sided golygons are defined as plane closed polygonal chains where the lengths of sides are formed by the sequence of successive positive integers $(1, 2, \dots, n)$ and its vertexes are the lattice points of the lattice, the adjacent sides are closing right angle.

One possible way of generalizing the notion of golygons is modifying the number of dimensions and searching for geometric objects with a specific number of sides – assuming they exist at all - in higher or lower number of dimensions, instead of the two-dimensional polygonal chains.

During the researches done at the Phd School of the Information Science Department of the Debrecen University a special computational algorithm was developed which is capable of methodically generating geometrical forms similar to the golygons on the number line, on the lattice, as well as the 3-dimension cubic lattice with the help of a backtrack seeker from the field of artificial intelligence. The algorithm counts all incongruous closed polygonal chains with a specific number of sides and records the data of all found forms in a database which can be examined further on.

This lecture aims to present all the findings relating to the case of one and three dimensions, as well as the heuristics applied when searching and it also focuses on their algorithmic efficiency.

Keywords: golygon, lattice polygon, closed polygonal chain, mathematical problem solving with computer, backtrack, heuristics, algorithmic efficiency

1. Előzmények, célkitűzés

A goligonok fogalmát A. K. Dewney ismertette először a Scientific American hasábjain megjelent cikkében. [1] A velük kapcsolatos matematikai kutatás eredményeit Lee Sallows, Martin Gardner, Richard K. Guy és Donald Knuth publikálták 1991-ben. [2] A szerzők számítógépes programot dolgoztak ki a különböző oldalszámú geometriai objektumok

összeszámlálására, mely eredmény helyességét egy matematikai tétellel igazoltak. A goligonok számának sorozatát a „The On-Line Encyclopedia of Integer Sequences” rögzíti [3]. Könyv formájában Ilan Vardi [4], valamint Martin Gardner [5] szentelt különálló fejezetet a téma bemutatásának. A goligonok fogalmának egyik lehetséges általánosításával Lee Sallows foglalkozott [6], melyben olyan, a goligonokhoz hasonló síkbeli zárt törött vonalakat vizsgált, melyek szögei 90 foktól eltérők.

A goligonok fogalmának egy másik lehetséges általánosításához jutunk, ha módosítjuk a dimenziók számát, és a kétdimenziós törött vonalakat helyett alacsonyabb, vagy magasabb dimenziószámú keressük az előre rögzített oldalszámú geometriai objektumokat, ha ilyenek egyáltalán léteznek.

Jelen közlemény célja, hogy bemutassa az egy- és háromdimenziós goligonokkal kapcsolatos kutatási eredményeket.

2. Egy- és háromdimenziós goligonok fogalma, probléma megfogalmazása

n oldalú egydimenziós goligonnak nevezzük azokat az origó kezdőpontú zárt törött vonalakat, melyek egymást követő oldalainak hossza rendre $1, 2, \dots, n$ és az oldalakat alkotó törött vonal a számegyenes mentén pozitív vagy negatív irányba halad mindaddig, amíg vissza nem jut a kezdőpontba. Csúcsnak nevezzük az egydimenziós goligonokat alkotó törött vonalak végpontjait. A csúcsokat egy egész számmal adjuk meg, a számegyenesen elfoglalt helyük alapján.

Megjegyzések:

Az egydimenziós goligonok minden csúcsa a számegyenes egész osztópontjára esik.

Két egydimenziós goligont egybevágónak nevezünk, ha létezik olyan egybevágósági transzformáció, mely egyik alakzatot a másikba viszi át.

Egy goligon eltolt képét kapjuk, ha minden csúcsához hozzáadjuk ugyanazt az egész számot. Emiatt szerepel az origó kezdőpontként, mert egy egydimenziós goligonnak mindig létezik olyan eltolt képe, melynek kezdőpontja az origó.

Legyen adott az $n \in \mathbb{N}$ pozitív szám. Egydimenziós goligont kapunk, ha találunk olyan alkalmasan választott előjelsorozatot, melyre a

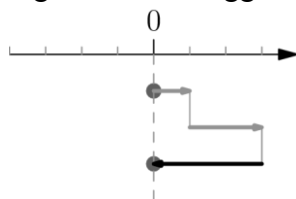
$$\pm 1 \pm 2 \pm 3 \pm \dots \pm n = 0 \quad (1)$$

egyenlet teljesül.

Ha találunk egy előjelsorozatot, melyre az (1) igaz, akkor minden előjelet az ellenkezőjére változtatva ugyancsak nyerünk egy, az origóra tükrözött megoldást.

Az egymással egybevágó egydimenziós goligonokat a későbbi vizsgálatok során nem tekintjük különbözőknek.

A legegyszerűbb egydimenziós goligon 3 oldalú, csúcsainak koordinátái: $0; 1; 3$ (1. ábra) (Az egyes törött vonal darabok a láthatóság érdekében függőlegesen szét lettek húzva.)



1. ábra. 3 oldalú egydimenziós goligon

$3n$ oldalú háromdimenziós goligonnak nevezzük azokat az origó középpontú zárt törött vonalakat, melyek oldalainak hossza a $1, 2, \dots, 3n$ természetes számok egy véges sorozata, az egymással szomszédos oldalak pedig derékszögben csatlakoznak egymáshoz. A háromdimenziós goligonok képzésének szabálya, hogy

- a $(3k-2)$ hosszúságú oldal szigorúan az x tengellyel,
- a $(3k-1)$ hosszú oldal az y tengellyel,
- míg a $3k$ hosszú oldal a z tengellyel párhuzamos,

ahol $k=1, 2, 3, \dots, n$ és $k \in \mathbb{N}$. Az oldalsorozat akkor ér véget, ha a törött vonal zárttá válik.

Háromdimenziós goligont kapunk, ha találunk egy-egy olyan alkalmasan választott előjelsorozatot, melyre a

$$\pm 1 \pm 4 \pm 7 \pm \dots \pm (3n-2) = 0 \quad (2a)$$

$$\pm 2 \pm 5 \pm 8 \pm \dots \pm (3n-1) = 0 \quad (2b)$$

$$\pm 3 \pm 6 \pm 9 \pm \dots \pm 3n = 0 \quad (2c)$$

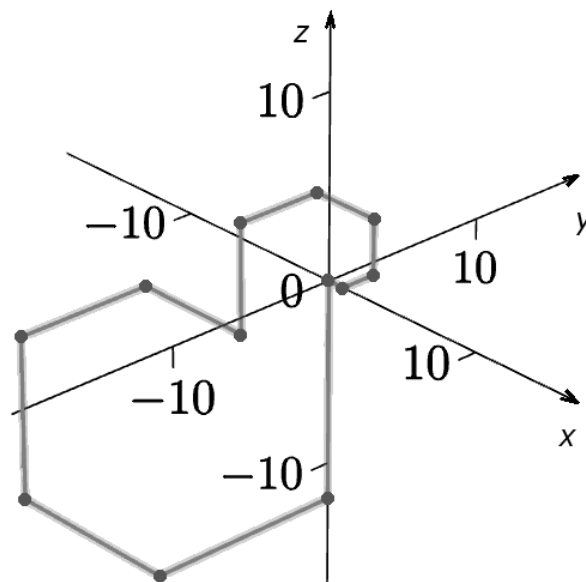
egyenletek mindegyike teljesül. A háromdimenziós goligonok vizsgálata ilyen módon három egymástól független n elemű részsorozat vizsgálatára redukálódik.

Az egybevágóság fogalma kiterjeszhető a háromdimenziós goligonokra is.

A háromdimenziós goligonok a szabályos kockarácson ábrázolhatók, minden csúcspontjuk a kockarács egész koordinátájú pontjára esik.

Az 2. ábrán az egyetlen 12 oldalú háromdimenziós goligon látható (eltekintve az egymással egybevágó esetektől). Csúcspontjainak koordinátái:

$$\begin{array}{cccccc} (0;0;0) & (1;0;0) & (1;2;0) & (1;2;3) & (-3;2;3) & (-3;-3;3) \\ (-3;-3;-3) & (-10;-3;-3) & (-10;-11;-3) & (-10;-11;-12) & (0;-11;-12) & (0;0;-12) \end{array}$$



2. ábra. A legkisebb oldalszámú háromdimenziós goligon ($n=4$, $3n=12$)

A kutatás során mindhárom vizsgált dimenziószám alatt a következő problémákra kerestük a válaszokat:

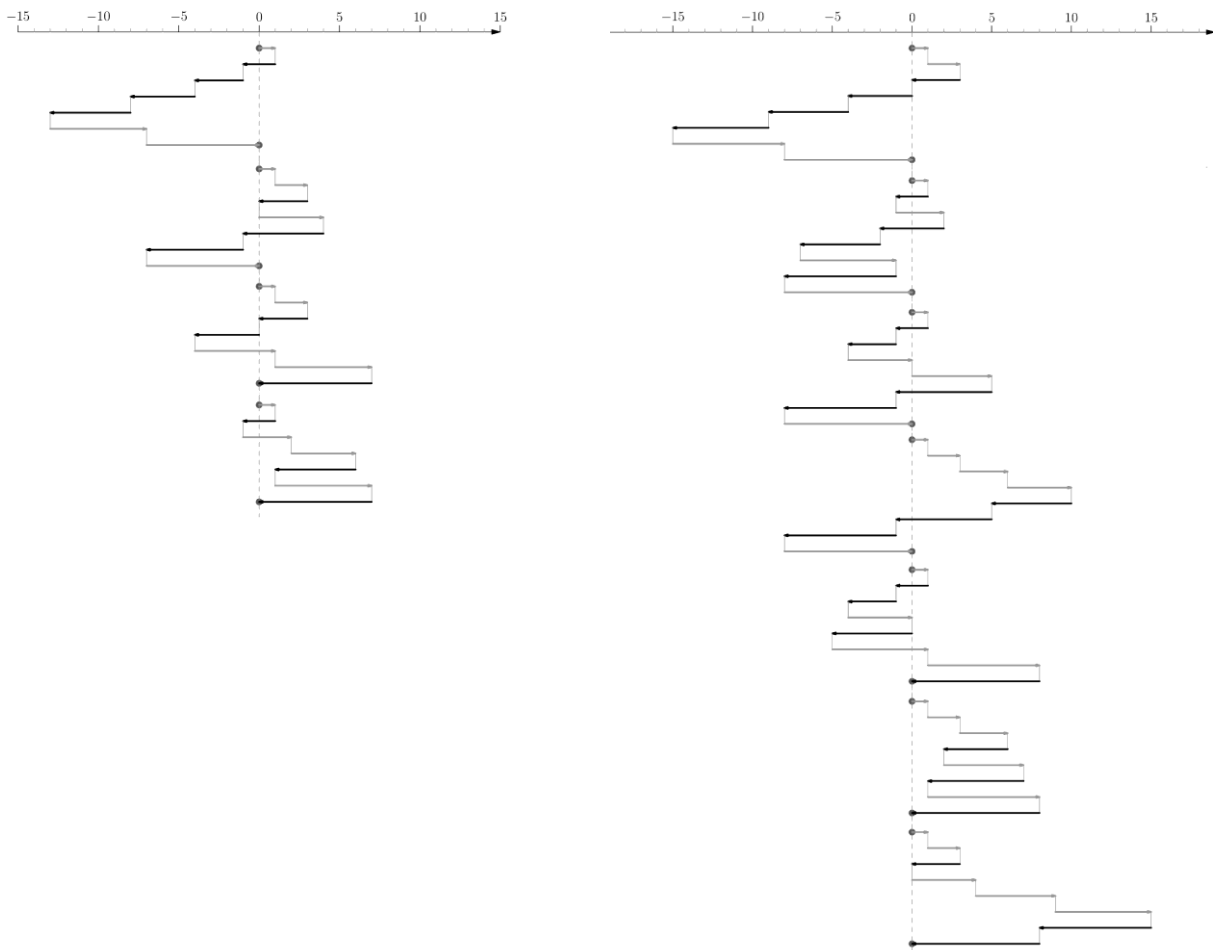
Legyen adott $n \in \mathbb{N}$ és jelölje d a vizsgált dimenziószámot ($d=1, 2, 3$). Létezik-e $d \cdot n$ oldalú d -dimenziós goligon?

Létezés esetén számoljuk össze, hány egymással nem egybevágó $d \cdot n$ oldalú goligon van!

Választ kerestünk továbbá arra is, hogy hatékonyabbá tehető-e a keresés alkalmasan választott heurisztika alkalmazásával?

3. Kereső algoritmus és alkalmazott heurisztika bemutatása

A Debreceni Egyetem Informatikai Tudományok Doktori Iskolájában végzett kutatás során olyan számítógépes algoritmus került kidolgozásra, mely kereső módszeresen előállítja az egy-, két- és háromdimenziós goligonokat. (A kétdimenziós eset vizsgálata csupán ellenőrzési célokat szolgált; a program által szolgáltatott megoldások számának meg kellett egyeznie az [1] cikk eredményeivel.) Az algoritmus összeszámolja az adott oldalszámú, egymással nem egybevágó zárt törött vonalakat, a talált alakzatok jellemző adatait pedig egy tovább vizsgálható adatbázisban rögzítette. A program előállította az egyes törött vonalak képét, a 2. ábrán az összes nem egybevágó 7 és 8 oldalú egydimenziós goligon látható.



3. ábra. Az összes nem egybevágó 7 és 8 oldalú egydimenziós goligon

Az algoritmus kiválasztásakor fontos szempontként szerepelt, hogy szisztematikusan össze akartuk számolni az összes létező, egymással nem egybevágó megoldást, erre a backtrack kereső bizonyult a legalkalmasabbnak.

Az állapotér reprezentáció során az (1) illetve (2a)-(2c) egyenletek bal oldalán szereplő előjeles részsorozatokat tekintjük állapotoknak. A kezdőállapotban a részsorozat minden eleme pozitív előjelű; a célállapotban az előjeles részsorozat összege 0. Operátorként a részsorozat elemeinek előjelcseréjét alkalmaztuk, az i -edik lépésben az i -edik sorozatelemre alkalmazzuk az operátort. Heurisztika alkalmazása nélkül visszalépést csak az után

alkalmazunk, ha már az n -edik sorozatelemet is elláttunk valamilyen előjellel és megvizsgáltuk, hogy a kapott előjeles részsorozat célállapot-e.

A kereső algoritmus futtatása előtt a kezdőállapotában célszerű megvizsgálni, hogy az adott részsorozat egyáltalán ellátható-e megfelelő előjelekkel, vagy sem. A célállapotban ugyanis az előjeles részsorozatban szereplő pozitív számok összegének egyenlőnek kell lennie a negatív számok összegével. Ezért ha a vizsgált részsorozat elemeinek összege a kezdőállapotban páratlan, akkor az elemek már nem láthatók el olyan előjelekkel, hogy az összegük 0, azaz páros legyen.

Egy nagyméretű keresőfa mérete alkalmasan választott heurisztika segítségével jelentős mértékben csökkenthető. (lásd táblázat)

A H1 heurisztika ötlete a következő: határozzuk meg az aktuális állapotban, hogy milyen messze jutottunk el a kezdőponttól. Számítsuk ki továbbá a további sorozatelemekkel megtehető távolságot. Amennyiben a hátralévő sorozatelemek összege kisebb, mint a kezdőponttól mért aktuális távolság, akkor alkalmazzuk a visszalépést.

A H2 heurisztika alkalmazásakor megfordítottuk a növekvő részsorozatokat csökkenővé. A H1 heurisztikával együtt alkalmazva a kezdeti első néhány lépést követően már jóval távolabb juthatunk a kezdőponttól. Emiatt kevesebb lépés után is kiderülhet, hogy a keresés olyan útra jutott, ahol már nem található megoldás, így hamarabb bekövetkezhet a visszalépés.

1. táblázat. Egydimenziós goligonok keresése során bejárt csomópontok számának alakulása az alkalmazott heurisztika függvényében

		n			
		20	24	28	32
Keresőfa bejárt csomópontjai	Heurisztika nélkül	1 048 575 (100,0%)	16 777 215 (100,0%)	268 435 455 (100,0%)	4 294 967 295 (100,0%)
	H1	563 025 (62,3%)	8 382 727 (50,0%)	125 984 955 (46,9%)	1 907 070 333 (44,4%)
	H1 és H2	75 603 (7,2%)	932 793 (5,6%)	11 951 627 (4,5%)	157 536 563 (3,7%)
Megoldások száma		7 636	93 846	1 199 892	15 796 439

Az 1. táblázat az n -oldalú egydimenziós goligonok keresése során előállított előjeles részsorozat számát mutatja be, négy kiválasztott n esetén. A feltüntetett százaléktételek számításakor a heurisztika nélküli vizsgálatok számát tekintettük viszonyítási alapnak. Megfigyelhető, hogy az alkalmazott heurisztikák hatékonysága az oldalszám növekedésével egyre inkább javul. Az táblázat adataiból az is megfigyelhető, hogy az H1 és H2 heurisztikák együttes alkalmazásakor közel minden tízedik részsorozat egyben megoldás is.

4. Elért eredmények

A keresőalgoritmus által összeszámolt nem egybevágó n -oldalú egydimenziós goligonok számára vonatkozó eredmények az 2. és a 3. táblázatokban tekinthetők meg. A táblázatban csak az $n=4k-1$ vagy $n=4k$ (ahol $k=1,2,3,\dots$ és $k \in \mathbb{N}$) oldalú egydimenziós goligon szerepelnek, a további n értékek esetén az algoritmus 0 értéket szolgáltatott. A megfigyeltekkel kapcsolatban az a sejtés fogalmazható meg, hogy nem léteznek olyan n -oldalú egydimenziós goligonok, ahol $n=4k-2$ és $n=4k-3$ alakú. A tükörszimmetrikus esetek kiszűrése úgy történt, hogy a sorozat első eleme csak pozitív előjelű lehetett.

Megjegyzés: Az 2. és 3. táblázatokban bemutatott sorozat értékei megegyeznek a The On-Line Encyclopedia of Integer Sequences A058377 sorszámú sorozatával, azonban az eredmény meghatározása az ott közölt rekurzív függvény számítási elvétől teljes mértékben eltér. [7]

2. táblázat. Egydimenziós goligonok száma ($1 \leq n \leq 72$),

n	1-dim. goligonok száma	n	1-dim. goligonok száma
3	1	39	1 512 776 590
4	1	40	2 915 017 360
7	4	43	20 965 992 017
8	7	44	40 536 016 030
11	35	47	294 245 741 167
12	62	48	570 497 115 729
15	361	51	4 173 319 332 859
16	657	52	8 110 661 588 734
19	4 110	55	59 723 919 552 183
20	7 636	56	116 307 527 411 482
23	49 910	59	861 331 863 890 066
24	93 846	60	1 680 341 334 827 514
27	632 602	63	12 505 857 230 438 737
28	1 199 892	64	24 435 006 625 667 338
31	8 273 610	67	182 650 875 111 521 033
32	15 796 439	68	357 366 669 614 512 168
35	110 826 888	71	2 681 644 149 792 639 400
36	212 681 976	72	5 253 165 510 907 071 170

3. táblázat Egydimenziós goligonok száma ($73 \leq n \leq 100$)

n	1-dim. goligonok száma
75	39 555 354 718 945 873 299
76	77 570 671 355 589 452 481
79	585 903 163 431 438 401 072
80	1 150 120 608 194 890 221 950
83	8 711 342 419 813 595 823 721
84	17 115 419 455 244 573 200 133
87	129 966 117 376 454 496 339 866
88	255 553 983 141 029 557 052 712
91	1 945 040 269 952 777 197 656 086
92	3 827 373 235 233 388 318 254 075
95	29 192 075 100 997 216 412 139 678
96	57 481 796 949 079 849 843 902 577
99	439 276 486 548 176 179 402 860 000
100	865 512 002 974 362 508 316 893 162

A $3n$ oldalú háromdimenziós goligonok számára nyert adatokat a 4. táblázat foglalja össze. Az x , y és z oszlopok az adott tengellyel párhuzamos oldalakból nyert nem egybevágó részsorozatok számát adja meg. Ezen részsorozatok egymástól függetlenek, így szorzatuk adja a nem egybevágó háromdimenziós goligonok számát. A táblázatban csak a négyel osztható n értékek lettek feltüntetve, a további n értékek esetén a program 0 eredményt szolgáltatott. A tapasztaltak alapján az a sejtés fogalmazható meg, hogy a $3n$ -oldalú háromdimenziós goligonok oldalainak száma 4-gyel is osztható, másként fogalmazva az oldalszám 12-vel osztható. A táblázatban szereplő egybevágó esetek kiszűrése úgy történt, hogy mindhárom részsorozat első elemének előjelét pozitív számként rögzítettük.

4. táblázat. Háromdimenziós goligonok száma ($1 \leq n \leq 100, n \bmod 4 = 0$)

n	3n (oldalszám)	x	y	z	Háromdimenziós goligonok száma
4	12	1	1	1	1
8	24	4	4	7	112
12	36	29	29	62	52142
16	48	268	266	657	46836216
20	60	2884	2853	7636	62829405072
24	72	33965	33534	93846	106888933864260
28	84	423663	418046	1199892	212513619050370216
32	96	5496421	5424779	15796439	471000355740874170001
36	108	73343319	72432717	212681976	1129863611624852407860648
40	120	999503538	987863532	2915017360	2878209713582372993899109760
44	132	13845615816	13695346589	40536016030	7686459925080442084939091532720
48	144	194329025191	192366554401	570497115729	21326554229287154922344320887430839
52	156	2757097128682	2731161838865	8110661588734	61073918157311253547244549997285072620
56	168	39473592379565	39126881342322	116307527411482	179634483057331823568360871495619452096260
60	180	569544928349817	564859813237074	1680341334827514	540587722186756080095900516384288624570711412
64	192	8273041755127160	8209100848277415	24435006625667338	1659484759962579082444879373851548755647972693200
68	204	120880418877672238	119999847338800467	357366669614512168	5183829331155423534944727327174171057116835271056528
72	216	1775430089467589975	1763203193425053304	5253165510907071170	16444740472760862806105545146544965311348798129220058000
76	228	26197602426550896488	26026561239049880266	77570671355589452481	52890282648000283713786951980141085028183629831302451909648
80	240	388170357291360375365	385761460869477069289	1150120608194890221950	172220398720685607170024046611615393060645161426431327704395750
84	252	5773105559736251126732	5738970453783289676695	17115419455244573200133	567062638691045412254072388566996197177706656690540972417172128420
88	264	86153188990180304380558	85666766760022229041147	255553983141029557052712	1886107265712055537661990858680829837996093017668919852760794843210512
92	276	1289664495349135364731343	1282697500948662403719348	3827373235233388318254075	6331429974588875703558804286473986367308529999766624970573785474674783300
96	288	19360264848978741870827609	19260008344885050637565840	57481796949079849843902577	21433747063732574075076089820077355512142782411400264859055663460801275695120
100	300	291390033985746183974819827	289941091741601494615406903	865512002974362508316893162	73123599113543882393424264808902898183238391036822425933680172234662184703089522

5. Eredmények igazolása

A fenti, keresőalgorithmus által összeszámolt megoldások számát vizsgálva a következő állításokat fogalmazhatjuk meg.

Tétel: Az n -oldalú egydimenziós goligonok száma egyenlő a

$$\prod_{k=1}^n (1+x^k) = (1+x)(1+x^2)(1+x^3)\dots(1+x^n) \quad (3)$$

szorzatpolinom felbontásában szereplő $x^{\frac{n(n+1)}{4}}$ tagjának együtthatójával.

Bizonyítás: Az n -oldalú egydimenziós goligonok számának meghatározásához rendeljük hozzá a (3) egyenlethez a

$$(x^{-1}+x^1)(x^{-2}+x^2)(x^{-3}+x^3)\dots(x^{-n}+x^n) \quad (4)$$

generátorfüggvényt. Vegyük észre, hogy a fenti szorzat felbontása után minden tag

$$x^{\pm 1}x^{\pm 2}x^{\pm 3}\dots x^{\pm n} = x^{\pm 1\pm 2\pm 3\pm \dots \pm n}$$

alakú lesz, így a x^k együtthatója adja meg a

$$\pm 1 \pm 2 \pm 3 \pm \dots \pm n = k \quad (5)$$

egyenlet megoldásainak számát. Egydimenziós goligonok esetében (1) szerint $k=0$, ami a megfelelő (4) generátorfüggvény konstans tagjának kiszámításával határozható meg.

Tehát a (3) egyenlet különböző megoldásainak számát - így egyben az n -oldalú egydimenziós goligonok számát is - a (4) kifejezés konstans tagja kiszámításával kaphatjuk meg.

A (4) szorzatból eltávolítjuk a negatív kitevőjű hatványokat. Egyetlen tényezője átírható

$$(x^{-k}+x^k) = \frac{1}{x^k}(1+x^{2k})$$

alakra, ami alapján a (4) értéke

$$\frac{1}{x^{1+2+3+\dots+n}}(1+x^2)(1+x^4)(1+x^6)\dots(1+x^{2n}) \quad (6)$$

A számtani sorozat összegképlete alapján a tört nevezőjében szereplő kitevők összege

$$1+2+3+\dots+n = \frac{n(n+1)}{2}$$

ezért az átalaktás után (6) a következővel egyenlő

$$\frac{1}{x^{\frac{n(n+1)}{2}}}(1+x^2)(1+x^4)(1+x^6)\dots(1+x^{2n}) \quad (7)$$

Az első együttható nélküli részben keressük tovább az $x^{\frac{n(n+1)}{2}}$ együtthatóját.

Mivel (7) szorzatban mindenütt x^2 hatványa szerepel, végezzük el a $x \rightarrow \sqrt{x}$ helyettesítést, de ekkor a tört nevezőjében szereplő hatványkitevőt is korrigálni kell, a tört nevezőjének 2-vel való osztásával. Így végül arra az eredményre jutottunk, hogy az egydimenziós goligonok számát a

$$(1+x)(1+x^2)(1+x^3)\dots(1+x^n) = \prod_{k=1}^n (1+x^k) \quad (8)$$

polinomszorzat felbontásában szereplő $x^{\frac{n(n+1)}{4}}$ tag együtthatója adja meg. Ezzel bizonyítottuk az állításunkat.

A bizonyított tétel alapján továbbra is nehéz kiszámítani a kérdéses együttható értékét, azonban egy matematikai programcsomag komoly segítséget nyújthat a számítás elvégzésében.

Az egydimenziós goligonok számát kiszámító *Mathematica* programkód:

```
prod[n_] := Product[x^k + 1, {k, 1, n}] // Expand;
coeff[n_] := Coefficient[prod[n], x, n (n + 1)/4];
t = Table[coeff[n]/2, {n, 1, 100}]
```

Magyarázat:

Az 1. sor *prod* függvénye meghatároz egy n -tagú szorzatot, mely tényezői $x^k + 1$ polinomokból állnak. Az *Expand* utasítás elvégzi a zárójelek felbontását.

A 2. sor *coeff* függvénye meghatározza a fenti szorzat x változó $n(n+1)/4$ -edik együtthatójának értékét.

A 3. sor rögzíti a t táblázatban az első 100 kiszámított együtthatót. A táblázat értékeit kettővel osztottuk, a tükörszimmetrikus megoldások elkerülése érdekében.

A *Mathematica* kód helyességét igazolja, hogy az általa kiszámított eredmények megegyeznek az 1. és 2. táblázat adataival.

A fenti bizonyítás módszerét alkalmazva a (2a)-(2c) egyenletekből kiindulva belátható a következő tétel.

Tétel: A $3n$ -oldalú háromdimenziós goligonok száma egyenlő a

- $\prod_{k=1}^n (1 + x^{3k-2}) = (1 + x)(1 + x^4)(1 + x^7) \dots (1 + x^{3n-2})$ polinomszorzat felbontásában szereplő $x^{\frac{n}{4}(3n-1)}$ tag,
- $\prod_{k=1}^n (1 + y^{3k-1}) = (1 + y^2)(1 + y^5)(1 + y^8) \dots (1 + y^{3n-1})$ polinomszorzat felbontásában szereplő $y^{\frac{n}{4}(3n+1)}$ tag,
- és a $\prod_{k=1}^n (1 + z^{3k}) = (1 + z^3)(1 + z^6)(1 + z^9) \dots (1 + z^{3n})$ polinomszorzat felbontásában szereplő $z^{\frac{3}{4}n(n+1)}$ tag együtthatóinak szorzatával.

A fenti tétel alapján a háromdimenziós goligonok számának meghatározásához is a *Mathematica* computer algebrai rendszer segítségét vesszük igénybe.

```
(* x irány *)
px[n_] := Product[1 + x^k, k, 1, 3 n - 2, 3] // Expand;
cpx[n_] := Coefficient[px[n], x, n/4 (3 n - 1)];
tx = Table[cpx[n]/2, n, 1, 100];
(* y irány *)
py[n_] := Product[1 + y^k, k, 2, 3 n - 1, 3] // Expand;
cpy[n_] := Coefficient[py[n], y, n/4 (3 n + 1)];
ty = Table[cpy[n]/2, n, 1, 100];
(* z irány *)
pz[n_] := Product[1 + z^k, k, 3, 3 n, 3] // Expand;
cpz[n_] := Coefficient[pz[n], z, 3/4 n (n + 1)];
tz = Table[cpz[n]/2, n, 1, 100];
szorzat = Table[tx[[m]]*ty[[m]]*tz[[m]], m, 1, 100];
t1 = Transpose[Join[Range[1, 100], 3*Range[1, 100], tx, ty, tz, szorzat]];
TableForm[t1, TableHeadings -> None, {"n", "Oldalszam", "x", "y", "z", "3D goligonok száma"}]
t2 = Table[t1[[m]], {m, 4, 100, 4}];
TableForm[t2, TableHeadings -> {None, {"n", "Oldalszam", "x", "y", "z", "3D goligonok száma"}}]
```

A tx , ty és tz táblázatok kiszámítása az egydimenziós goligonoknál megismert módon történik, a fent megfogalmazott tételben megadott együtthatók kiválasztásával. Mindhárom táblázatban a kiszámított együtthatók számát 2-vel osztjuk, az egybevágó alakzatok kiszűrése érdekében. A $t1$ táblázat tartalmazza az összes kiszámított megoldást $n \leq 100$ esetre, míg a $t2$ csak azokat a sorokat tartalmazza, ahol található megoldás, azaz ahol n 4-gyel osztható.

6. Összefoglalás

A tanulmány megismertette az olvasót az egy- és háromdimenziós goligonok fogalmával. A velük kapcsolatos felvetett problémákra nem csupán számítógépes módszert ismertetett az alkalmazott heurisztikákkal együtt, hanem a kapott eredményeket algebrai eszközökkel is igazolta. A kimondott tételek alapján pedig megadott egy-egy Mathematica programkódot, mely segítségével gyorsan kiszámíthatók az egy- és háromdimenziós goligonok száma. Ugyanakkor a vizsgálati folyamat újabb, még megoldatlan problémákat vetett fel, mely a további kutatások irányvonalát is meghatározzák. Nem került bizonyításra két sejtés, mely még igazolásra vár. Ugyanakkor felvetődhet az a kérdés is, hogy háromnál magasabb dimenziószámokban nem általánosítható-e a probléma? Vagyis tetszőleges dimenziószámokban előállíthatók-e a goligonokhoz hasonló alakzatok? És ha léteznek, akkor vajon hány nem egybevágó alakzat van adott oldalszám esetén? A Doktori Iskolában folyó kutatások ezen kérdésekre kívánnak a továbbiakban kielégítő válaszokat szolgáltatni.

7. Köszönetnyilvánítás

Köszönettel tartozom konzulensemnek, Dr. Várterész Magda tanárnőnek, a bemutatott bizonyítás matematikai helyességének ellenőrzéséért, valamint mindazért a számos kiváló szakmai tanácsért, amellyel hozzájárult a kutatás során előállított program és jelen cikk elkészítéséhez.

Irodalomjegyzék

- [1] Dewdney, A. K. :An Odd Journey Along Even Roads Leads to Home in Golygon City. *Scientific American* **263**, (July 1990), 118-121.
- [2] Sallows, L.; Gardner, M.; Guy, R. K.; and Knuth, D. :Serial Isogons of 90 Degrees, *Mathematics Magazine* **64**, (1991) 315-324. (<http://www.jstor.org/pss/2690648>)
- [3] Sloane, N. J. A. : Sequence A006718 "The On-Line Encyclopedia of Integer Sequences." (<http://oeis.org/A006718>)
- [4] Vardi, I. : Computational Recreations in Mathematica, (§5.3 "American Science.") Redwood City, CA: Addison-Wesley, (1991) 90-96.
- [5] Gardner, Martin "A Gardner's workout: training the mind and entertaining the spirit", Natick, Massachusetts, A. K. Peters, pp. 215-230, 2001
- [6] Sallows, L. C. F. :New Pathways in Serial Isogons, *The Mathematical Intelligencer* **14**, (1992), 55-67.
- [7] Sloane, N. J. A. Sequence A058377 "The On-Line Encyclopedia of Integer Sequences." (<http://oeis.org/A058377>)

Egy evolúciós heurisztika a 2D lap-pakolási problémára

An Evolutionary Heuristic for 2D Strip packing

Borgulya István

Pécsi Tudományegyetem
borgulya@ktk.pte.hu

Absztrakt: E cikkben egy memetikus algoritmus alapú hiper-heurisztikát mutatunk be a 2D lap-pakolási problémára (2DSP). Az algoritmus alaptechnikája egy memória alapú technika, a kiterjesztett virtuális vesztes (EVL) [2]. Az EVL technika az evolúciós algoritmus futása során megtanulja az egyed változóinak rossz értékeit a populáció gyenge megoldásai alapján és becslést ad a mutáció valószínűségére az egyes generációkban. Az EVL alkalmazása esetén elegendő csak mutációt alkalmazni rekombináció művelet nélkül, és egytől egy tanuló mechanizmust is kapunk a heurisztikák szelektálásához a hiper-heurisztikában. Minden egyed a hiper-heurisztikában (HH) két részből áll: a megoldásból és egy alacsony szintű heurisztika listából. Az alacsony szintű heurisztikák a hiper-heurisztikában három csoportba sorolhatók: mutáció heurisztikák, helyi kereső eljárások és elhelyező heurisztikák. Egy utódnál az alkalmazásra kerülő mutáció művelet, a helyi kereső eljárás és minden téglalaphoz egy elhelyező heurisztika kiválasztása, vagy módosítása az EVL technika segítségével történik. A hiper-heurisztika eredményeinek javítására egy párhuzamos evolúciós modellt alkalmazunk. Ez egy island modell változat [3], melynél 16 processzoron fut párhuzamosan ugyanaz a hiper-heurisztika algoritmus. A teszteredmények alapján az algoritmus jó hatásfokkal dolgozik

Kulcsszavak: island modell, hiper-heurisztika, lap-pakolás

Abstract: In this paper, we present a memetic algorithm-based parallel hyper-heuristic for two-dimensional rectangular strip-packing problems (2DSP). The basic technique of the algorithm is a memory-based evolutionary technique, the “extended virtual loser” (EVL) [2]. The EVL technique learns the bad values of the variables based on the worst solutions of the population and computes probabilities to control the mutation steps. With the help of the EVL technique, we can use a mutation-omitting recombination operator and obtain a learning mechanism for the selection of heuristics. The individual has two parts: the solution and a set of the low-level heuristics used. Among the low-level heuristics, we distinguish three groups: the placement heuristics, the local search heuristics and mutation heuristics groups. In the HH, the selection of the low-level heuristics is modified with mutations based on the EVL technique and using a local search. For better results, we also use parallel computing. This is an island model [3] and in all of the islands, we run the memetic algorithm-based hyper-heuristic. The test instances show that the proposed algorithm is efficient for the strip-packing problem.

Keywords: island model, hyper-heuristic, strip packing

1. Bevezetés

A 2DSP-ben adott téglalapok egy halmaza, amelyben n téglalap található adott w_i , h_i szélességgel és magassággal $i=1,2,\dots,n$. A cél a téglalapok elhelyezése egy W szélességű lapra úgy, hogy a téglalapok ne fedjék egymást és az elhelyezés magassága minimális legyen. A pakolási probléma specifikálható forgatással és vágási (guillotine) feltétellel. Mi a forgatás és vágási feltétel nélküli 2DSP változatot vizsgáljuk.

A 2DSP a vágási, pakolási problémák közé tartozik és számos ipari alkalmazása létezik. NP nehéz probléma és számos egzakt, heurisztikus és meta-heurisztikus módszert publikáltak megoldására. Az elmúlt évtizedben hiper-heurisztikát is alkalmaztak megoldására. A HH egy

kereső módszer, amely a heurisztikák terében dolgozik válogatva, vagy generálva jó heurisztikákat (csak korlátozottan használva fel információt a probléma megoldási teréből) és alkalmazza őket a probléma megoldására.

E cikkben egy memetik algoritmus (MA) alapú HH-t mutatunk be 2DSP megoldására. A HH alapszerkezetét egy memória alapú evolúciós technika, az EVL [2]. A memória alapú technika memorizálja az elmúlt eseményeket, pl. a sikeres lépéseit az evolúciós folyamatnak, vagy a változók rossz értékeit és befolyásolhatjuk az evolúciós algoritmus műveleteit a memória felhasználásával. Az EVL technika a változók rossz értékeit tanulja a rosszabb megoldások alapján és valószínűséget számol a mutáció szabályozására. Az EVL technika használatával a mutációt rekombináció nélkül használhatjuk és egyúttal egy tanuló algoritmust is kapunk heurisztikák szelektálására a HH-ban.

Az MA-alapú HH (MAHH) EVL alapú mutációval és helyi kereső eljárással válogat az alacsony szintű heurisztikák közt. Alacsony szintű heurisztikák a 2DSP megoldására esetünkben a mutáció változatok, helyi kereső eljárások és elhelyező heurisztikák. Az algoritmusban három elhelyező heurisztikát vegyesen alkalmazunk, melyek korábbi elhelyező heurisztikák módosított változatai.

A jobb eredmények érdekében párhuzamos számításokat is alkalmazunk. Korábbi island modellünket [3] felhasználva, minden islandon az MAHH fut. Legjobb tudomásunk szerint ez az első párhuzamos HH a 2DSP megoldására.

2. Kapcsolódó munkák

Egzakt, heurisztikus, meta-heurisztikus és HH megoldásokat egyaránt találunk 2DSP-re. Az első egzakt algoritmusok lineáris programozás alapú és fa kereső algoritmusok voltak. Több szerző a korlátozás és szétválasztás módszer változatait alkalmazza. Az egzakt módszerek csak kis méretű feladatokon alkalmazhatók gazdaságosan.

A legfontosabb heurisztika csoport a konstrukciós heurisztikák. Itt egy fontos kérdés, hogyan választjuk ki a következő téglalapot elhelyezésre. Választhatunk egy rögzített sorrendből, vagy dinamikusan választhatunk egyet. A fix, rendezett sorrendet alkalmazzák az első jól ismert heurisztikák, pl. a „bottom up, left justified” (BL) heurisztika, amely balról a legmélyebb helyre helyezi a téglalapot (részletesebben [1]-ben). A “best fit” (BF) heurisztika dinamikusan választ és a lehetséges legmélyebb helyre a legjobban illeszkedő téglalapot helyezi (részletesebben [4]-ben).

Számos meta-heurisztikát alkalmaztak 2DSP megoldására. Szimulált hűtés, tabu keresés, evolúciós algoritmus változatok és hibrid megoldások egyaránt megtalálhatók. Néhány HH-t is találunk a módszerek közt. Ezek elsősorban az elhelyező heurisztikák megválasztásában, vagy generálásában nyújtanak különböző megoldásokat.

3. A memetik algoritmus-alapú hiper-heurisztika

MAHH algoritmusunk populáció alapú és javító technikát alkalmaz. Az egyed két részből áll: a megoldás és a használt alacsony szintű heurisztikák. A megoldást a heurisztikák alkalmazásával kapjuk egy korábbi megoldásból. MAHH működésének lényege: mutációt és

helyi kereső eljárást alkalmaz az egyed heurisztika részén, majd alkalmazza a heurisztikákat új megoldás előállítására.

MAHH egy steady-state MA. Két fázisu algoritmus struktúráat alkalmaz: az első fázis a kezdő populáció minőségét javítja, míg a második egy szokásos MA.

MAHH főbb lépéseit az 1. algoritmus mutatja. Paraméterek:

$tmax$ – a populáció maximális mérete.

t – a kezdő populáció mérete.

itt – generációk száma az első fázisban.

kn - az algoritmus ellenőrzése minden kn -dik generációban történik.

$timeend$ – a maximális futásidő.

1. Algoritmus. Az MAHH algoritmus

Procedure MAHH

Kezdő populáció megadása: $\{I_1, I_2, \dots, I_t\}$, ahol $I_j = (S_j, H_j)$.

EVL memória mátrixok kezdőértéke. $it = 0$.

/* Első fázis

Do itt times

Véletlen utódgenerálás: (S, H) . Elfogadási kritérium.

$it = it + 1$. Minden kn -dik generációban memória mátrixok aktualizálása. *Filter*.

od.

/* Második fázis

Repeat

Do kn times

$it = it + 1$; Egy szülő választása truncation szelekcióval: I_j .

EVL-alapú mutációk H_j -n. Az új lista H' .

Alkalmazzuk H' heurisztikáit S_j -re. LSH.

Elfogadási kritérium.

od

If $(t < tmax)$ then

Véletlen utódgenerálás: (S, H) . $t = t + 1$. Elfogadási kritérium. **fi**

Alkalmazzuk LS_j és LSH eljárásokat az utolsó utódra. Elfogadási kritérium.

Alkalmazzuk LS_j és LSH eljárásokat a legjobb egyedre. Elfogadási kritérium.

Memória mátrixok aktualizálása. *Filter, Restart*.

until futás idő $> timeend$

end

A műveletek és algoritmus jellemzők a következők:

Egyedek. A P populáció minden egyede egy megoldás-heurisztika lista pár $I_j = (S_j, H_j)$ $j = 1, 2, \dots, |P|$, ahol S_j a megoldás, H_j az alkalmazott alacsony szintű heurisztikák listája. A H_j lista a következő: $(MS_j, LS_j, PL_{j1}, PL_{j2}, \dots, PL_{jn})$, ahol MS_j a mutáció heurisztika a megoldáson, LS_j a helyi kereső eljárás a megoldáson és a $PL_{j1}, PL_{j2}, \dots, PL_{jn}$ heurisztikák az elhelyező heurisztikák, egy elhelyező heurisztikát kapcsolva a megoldás minden pozíciójához (lásd a 4. fejezetet). A megoldás-része az egyednek az n táglalap-sorszám egy permutációja.

Kezdő populáció. A kezdő populációban az algoritmus öt olyan egyedet generál, ahol a téglalapok a megoldásban különböző kritériumok szerint vannak rendezve (csökkenő sorrendben: magasság, magasság és szélesség, kerület, terület, valamint szélesség és

magasság szerint) a H_j listát véletlen generálja. A kezdő populáció többi egyedét véletlen generálja.

Fitness függvény. A fitness függvényt a megoldáson definiáljuk: az elhelyezett téglalapok magassága a lapon.

Szelekciós művelet. MAHH truncation szelekcióval egy I_j egyedét választ. A szelekcióban a populációnak csak a legjobb tp százaléka vesz részt.

Mutáció műveletek. MAHH különböző mutációkat alkalmaz a három heurisztika csoportra H_j -ben. Minden mutáció az EVL technikát alkalmazza különböző memóriákkal. Minden csoportnál a mutáció a legnagyobb pr_j valószínűséggel választ másik heurisztikát. MAHH a mutáció műveleteket 0.5 valószínűséggel alkalmazza. (EVL működését lásd részletesebben [2]-ben).

Helyi kereső eljárás. Az elhelyező heurisztikák listájára egy összetett helyi kereső eljárást alkalmazunk (LSH). Első lépésben 10 új listát generál, ahol a háromféle elhelyező heurisztika eltérő valószínűségekkel kerül kiválasztásra. Ha egy új lista után javul az eredmény, akkor az új lista kerül az egyedbe. Az első lépést ötször ismétli. A második lépésben a lista véletlen pozícióin másik elhelyező heurisztikára cseréli a korábbi. Ha csere után javul az eredmény, akkor az új lista kerül az egyedbe.

Filter, Restart. A konvergencia gyorsítására MAHH két eljárást alkalmaz: *Filter és Restart*. *Filter* kiszűri és törli a gyengébb egyedeket, amelyek egy jobb egyed közelében találhatóak. *Restart*: ha a legjobb megoldás nem változik az utolsó gp generációban, *Restart* törli a leggyengébb megoldásokat (rp részét a populációnak).

Elfogadási kritérium. Az elfogadási kritérium a crowding technika speciális változata, amely segít elkülöníteni a helyi és globális optimumokat is. Ez a crowding technika összehasonlítja az utódot a korábbi megoldásokkal az S_j megoldáshoz való hasonlóság alapján. Az első fázisban lecserélheti a korábbi leghasonlóbb megoldást, ha jobb nála; a második fázisban a szülőket cserélheti le, ha jobb nála. Ha az egyedek száma kevesebb, mint a populáció egyedeinek száma (pl. *Restart* után), az utód vizsgálat nélkül bekerül a populációba.

Megállási feltétel. Az algoritmus befejeződik, ha a megadott futásidő letelt.

4. Alacsony szintű heurisztikák MAHH-ban

Az alacsony szintű heurisztikák közt három csoportot különböztetünk meg: az elhelyezési heurisztikák, a helyi kereső eljárások és a mutáció heurisztikák.

4.1 Elhelyezési heurisztikák

2DSP számára három elhelyező heurisztikát definiáltunk: HP1, HP2, HP3. A HP1 BL módosított változata, a második és harmadik pedig BF módosított változata. A legfontosabb eltérés, hogy minden téglalagnál egy helyi kereső eljárást alkalmaznak, amely az elhelyezést javítani próbálja 2-3 téglalapról összeállított blokkal. A harmadik heurisztika ezen kívül a lehetséges elhelyezési helyeket fontosság szempontjából pontozza és a legmagasabb pontszámú helyre keresi a következő téglalapot.

Fontos kérdés a heurisztikák hatékonysága. Összehasonlítva néhány kombinációjukat MAHH-ban megállapítottuk, hogy

- HP2 és HP3 jobb mint BF, és HP1 jobb mint BL.
- HP1 rosszabb mint HP2 és HP3.
- A kombinációk HP1-el jobbak mint nélküle és
- A három heurisztika kombinációja adja a legjobb eredményt, így a három heurisztikát együtt alkalmazzuk MAHH-ban.

4.2 Helyi kereső eljárások

Hét helyi kereső eljárás valamelyikét alkalmazza MAHH a megoldáson. Egy eljárás a megoldás permutációjának minden párját előveszi, alkalmaz egy mozgást rajtuk és kiszámolja az új fitnessz értéket. Ha javult az érték, elfogadja a mozgást a permutáción. A hét eljárás a következő:

LS1: csere mozgás, LS2: beszúrás mozgás, LS3: inverzió mozgás. LS4: LS1 után LS2, LS5: LS1 után LS3, LS6: LS2 után LS3 és LS7: LS1 után LS2 és utána LS3.

4.3 Mutáció heurisztikák

MAHH négy mutáció közül alkalmazza valamelyiket az S_j megoldáson. E mutáció heurisztikák a következők: csere EVL-alapján, beszúrás EVL-alapján, véletlen csere, véletlen beszúrás.

5. Az island modell

Island modellünk [3] master-slave struktúrát alkalmaz, amely segíti a migrációs folyamatot. A modell centralizált, melyben a slave processzorokon fut az evolúciós algoritmus és amelyek periódikusan elküldik legjobb egyedeiket a master eljárásnak. A master eljárás tárolja a küldött egyedeket és véletlen választással küld belőlük másikat a slave processzoroknak.

A mi párhuzamos számítási modellünkben (nevezzük PMAHH-nak) 16 processzoron/islandon futott az MAHH és mindegyik a megadott futási ideig dolgozhatott (a modellt részletesebben lásd [3]-ben).

Számítási eredmények

A PMAHH algoritmust C++-ban programoztuk és iMAC, Intel Core i5 2,5 GHz, 4 GB RAM számítógépen futott Mac OS X 10.9.2 operációs rendszer alatt.

Az algoritmust a publikációkban szokásos benchmark teszthalmazokon ellenőriztük. Az első csoportjuk a veszteség nélkül elhelyezhető téglalapok teszthalmazai, melyeknél ismerjük az optimális megoldást, a második pedig a veszteséggel elhelyezhető téglalapok teszthalmazai, melyek optimális megoldása nem ismert. A teszthalmazok jelöléseit az [5], [6] publikációkból vettük át.

6.1 Paraméterek meghatározása

Megfelelő paraméter értékek meghatározásához egy meta-optimalizációs módszert alkalmaztunk, amely egy helyi kereső eljárás. A meta-optimalizáló egy, vagy több teszpéldát vizsgál egyszerre és egy meta-fitnessz számol, amely a teszpéldák fitness értékeinek összege. Olyan paraméter érték kombinációkat keres, amelyek csökkentik a meta-fitnessz értékét.

A meta-optimalizáló segítségével meghatároztuk MAHH paramétereit. A legidőigényesebb helyi kereső eljárások esetén ezen kívül meghatároztuk, hogyan korlátozzuk a helyi kereső eljárás keresési tartományát, ismétléseinek számát hogy segítségükkel növekvő dimenzió függvényében egyre rövidebb futásidőket engedélyezzünk az eljárásoknak. Végül az island modell paramétereit határoztuk meg 2, 4, 8 és 16 processzort alkalmazva. A legjobb eredményeket 16 processzor esetén kaptuk, így a 16 processzoros változattal dolgoztunk minden teszpéldánál.

Futásidőnek elfogadtuk a publikációkban szereplő 60 CPU szekundum futásidőt és minden island 60 szekundumig dolgozhatott.

6.2 Eredmények összehasonlítása

Tízszer futtattunk minden teszpéldát PMAHH-val és a legjobb, valamint átlagos eredményeket gap%-kal adtuk meg. gap% az alsó korlától (vagy az optimumtól) való eltérés százaléka.

Összehasonlításra az [5], [6] cikkek alapján négy módszert választottunk. Ezek a reactive GRASP, az ISA, az SRA és az IDBS. Mindegyik módszer 60 CPU szekundum futásidőt engedélyezett és eredményeik a legjobbak közé tartoznak.

	GRASP	ISA	SRA	IDBS	PMAHH
C	0.95	0.76	0.69	0.12	0.81
N	0.95	0.41	0.23	0	0.42
Babu	0.30	0	0	0	0
NT	2.32	2.24	1.60	1.54	2.24
CX	0.88	0.88	0.52	0.43	0.66
NP	3.06	2.52	2.00	2.10	2.70
NPT	1.50	0.56	0.15	0.35	0.93
ZDF ₁₃	-	4.00	2.94	-	2.68
2sp	2.68	3.02	3.07	3.01	2.63
BWMV	1.77	1.70	1.63	2.00	1.75

1. táblázat A módszerek átlagos eredményei (gap%)

Az első és második tábla a módszerek átlagos, illetve legjobb eredményeit mutatja. Az egyes teszpélda csoportok eredményeit külön mutatják az egyes oszlopok. A legjobb eredményeket vastagabb számok mutatják minden példacsoportnál. Megállapítható, hogy PMAHH eredményei átlagosan a harmadik legjobb eredmények, de két példacsoportnál (az első 13 ZDF példánál és 2sp-nél) a legjobbak mind az átlagos, mind a legjobb eredmények alapján. Összességében megállapíthatjuk, hogy PMAHH jobb a reactive GRASP-nál és az ISA, SRA, IDBS módszerek csoportjába tartozik.

	GRASP	ISA	SRA	IDBS	PMAHH
C	0.95	0.64	0.62	0.04	0.67
N	0.95	0.20	0.13	0	0.31
Babu	0.30	0	0	0	0
NT	2.32	1.96	1.30	1.01	1.69
CX	0.88	0.67	0.45	0.40	0.50
NP	3.06	2.32	1.90	1.90	2.40
NPT	1.50	0.56	0.15	0.25	0.85
ZDF ₁₃	-	3.08	2.87	-	2.13
2sp	2.68	2.99	3.05	2.62	2.60
BWMV	1.77	1.53	1.49	1.75	1.56

2. táblázat A módszerek legjobb eredményei (gap%)

6. Összefoglalás

A cikkben egy evolúciós algoritmust mutattunk be a 2DSP megoldására. Az algoritmus egy island modell, amely memetikus algoritmus-alapú hiper-heurisztikát alkalmaz. A hiper-heurisztika memória-alapú technikával és helyi kereső eljárással módosítja az alkalmazott heurisztikák választását. Az alacsony szintű heurisztikák különböző mutáció műveletekből, helyi kereső eljárásokból és háromféle elhelyező heurisztikából állnak. Az elhelyező heurisztikák korábbi heurisztikák módosított változatai.

Algoritmusunk jó hatásfokkal oldja meg a 2DSP feladatokat. Az eredmények alapján az ISA, SRA, IDBS módszerek csoportjába tartozik.

Irodalomjegyzék

- [1] Brenda S Baker, Edward G Coffman és Ronald L Rivest: Orthogonal packing in two dimensions. *SIAM Journal on Computing*. **9** (1980) 846–55.
- [2] Istvan Borgulya: An Evolutionary Algorithm for the biobjective QAP. In: B. Reusch (ed): *Computational Intelligence, Theory and Applications „Advances in Soft Computing”* Springer series, (2006) 577-586.
- [3] Istvan Borgulya: An Island Model for the No-Wait Flow Shop Scheduling Problem. PPSN XI. Krakow, Springer, LNCS 6239, (2010) 280-289
- [4] Edmund Burke, Graham Kendall és Glenn Whitwell: A new placement heuristic for the orthogonal stock-cutting problem. *Operations Research* **52** (2004) 655–71.
- [5] Lijun Wei, Wee-Chong Oon, Wenbin Zhu és Andrew Lim: A skyline heuristic for the 2D rectangular packing and strip packing problems. *European Journal of Operational Research*. **215** (2) (2011) 337-46.
- [6] Shuangyuan Yang, Shuihua Han és Weiguo Ye: A simple randomized algorithm for two-dimensional strip packing. *Computer & Operations Research*, vol. **40**, issue 1. (2013) 1-8.

A végelem hálók párhuzamos konverziója

Parallel conversion of finite element meshes

Iványi Péter^a

^a Rendszer és Szoftvertechnológia Tanszék, Pollack Mihály Műszaki és Informatikai Kar, Pécsi Tudományegyetem

ivanyi.peter@pmmik.pte.hu

Absztrakt: A különböző mérnöki szimulációk és számítások alapjai a végelem hálók. Sajnos manapság nagyon sokféle formában tároljuk ezeket a végelem hálókat. Ez a sokszínűség ugyanakkor többféle értelemben is problémát okoz, például a szimulációk validációja különböző szoftverekben nehézségbe ütközhet, hiszen a használt szoftvereknek ismerniük kell ugyanazokat a formátumokat. Egy másik probléma a végelem háló formátumokkal, hogy hosszú távú tárolásuk és megőrzésük nem megoldott. Mind a két problémára megoldás lehet a végelem hálók konverziója. Ugyanakkor a mai végelem hálók mérete ma már gyakran meghaladja a több milliós elemszámot, így a cikk a végelem hálók párhuzamos konverziójával foglalkozik, mint egy lehetséges megoldással. A cikk szintén megemlíti a konverzió lehetséges kimeneteleit, melyek valóban megőrizhetők hosszú távon, például XML alapú tárolás.

Kulcsszavak: végelem háló, konverzió, szöveg feldolgozás, párhuzamos, XML

Abstract: The basis of the different engineering simulations and analysis are the finite element meshes. Unfortunately nowadays there are a lot of different file formats for these finite element meshes. This diversity of the formats causes problems, for example it can be difficult to validate the simulations by different software, since these softwares must recognise the same file formats. Another problem is the long-term storage of these finite element meshes. A solution for both of these problems can be the conversion of them. However the finite element meshes nowadays can contain over a couple of million elements, therefore this paper discusses the parallel conversion of these large finite element meshes. The paper will also mention what format can be the output of the conversion, that can be stored for a long-term, such as the XML format.

Keywords: finite element mesh, conversion, text processing, parallel, XML

1. Bevezetés

Három trend befolyásolta az ebben a cikkben leírt kutatást. Az első trend arra vonatkozik, hogy az elmúlt évtizedekben sok kutató intézet és cég saját végelem szimulációs szoftvert fejlesztett ki, melyek saját file formátumot használnak. Néhány példa a jelenleg is használatos végelem file formátumokra: OpenFOAM [1], CFS, FLUENT, GMSH, MEDIT, LUSAS, LS-DYNA [2], ANSYS, IDEAS, VTK [3]. Bár ezek a formátumok nagyon különbözőnek tűnnek, de három alapvető osztályba sorolhatóak [4]: elemenkénti és blokk-szintű definíció illetve az elfajuló esetek.

A második trend azt emeli ki, hogy a végelem hálók mérete folyamatosan növekedett az elmúlt évtizedekben, a néhány száz elem helyett manapság az 50 millió véges elemből álló végelem háló is nagyon gyakori. Végezetül a harmadik trend ezen végelem hálók megőrzésére és hosszú távú tárolására vonatkozik, hiszen a szimulációkhoz használt végelem hálók és adatokra szükség lehet később további elemzés, optimalizáció vagy ellenőrzés miatt. Ezt a felismerést erősíti például, hogy 2011 óta az USA-ban ha valaki támogatásért folyamodik a „National Science Foundation”-hoz, akkor egy adatkezelési tervet

(Data Management Plan) [5] is be kell nyújtania, aminek tartalmaznia kell az adatok típusát, leírását, meta adatokat, hozzáférési jogok leírását, hogy milyen szabályok alapján használhatók az adatok és tárolási terveket.

Az általános adatok hosszú távú tárolása szempontjából manapság két technikát szoktak alkalmazni [6]. Az első esetben a hardveres és szoftveres környezetet teljes mértékben szimuláljuk. Mivel ez nem mindig könnyű vagy megoldható ezért a virtualizáció [7] lépett előtérbe. Ez azt jelent, hogy például egy virtualizált számítógépre akár egy 10-15 évvel korábbi Windows 3.1-es operációs rendszer is feltelepíthető és olyan programok is, amelyek csak ezen az operációs rendszeren futnak és ez a környezet összességében így képes a régi elmentett adatokat is megjeleníteni. A másik megoldás régi adatok megjelenítésére, hogy az adatokat olyan újabb formátumba konvertáljuk, amelyek az aktuális számítógépes környezetben használhatók.

Ez a cikk ez utóbbi, konverziós technikát mutatja be végeselem hálók esetére figyelembe véve a másik kettő trendet is, így sok elemet tartalmazó végeselem hálók párhuzamos konverzióját tárgyaljuk a továbbiakban.

2. Szekvenciális végeselem háló konverzió

Ahogy előzőekben említettük a végeselem hálók három nagy csoportba sorolhatóak [4]:

- Az **elemenkénti** definíció esetén, minden egyes adatot egyesével definiálunk, amelyet valamilyen azonosító számmal vagy szöveggel jelölünk meg, hogy a későbbiekben hivatkozni lehessen rájuk. Ebben az esetben az információ széteszik a file-ban, például:

```
DEFINE POINT id1 x1 y1 z1
...
<some other definitions>
...
DEFINE POINT id2 x2 y2 z2
...
```

- A **blokk-szintű** definíció esetén, egy nagy csoportban vannak definiálva sorozatban egymás után. Ebben az esetben egy hivatkozás egy elemre lehet explicit, például egy szám vagy szöveges azonosító, de lehetőség van implicit referenciára is, amikor a blokkbeli elhelyezkedését, sorozatbeli indexét használjuk. Például:

```
COORDINATES
id1 x1 y1 z1
id2 x2 y2 z2
id3 x3 y3 z3
...
```

- A harmadik csoportba az elfajuló eseteket soroljuk. Ezek bár végeselem hálónak tűnnek, de valójában valamilyen módon sértik a végeselem háló definícióját [4].

Egy végeselem hálóban általában három információra mindenképpen szükség van: a csomópontok koordinátáira, az elemek csomópontjaira és az elemek típusára. A csomópontok koordinátái definiálják a végeselem háló geometriáját, az elemek típusa és csomópontjai definiálják a topológiát. Ebben a cikkben csak ezeknek az információknak a konverziójával foglalkozunk, bár itt kell megjegyezni, hogy ugyanilyen fontos, más adatokat is tartalmaz egy végeselem háló, például: kezdeti és perem feltételeket, illetve az eredményeket, mint feszültségek, hőmérséklet, stb. Ezek az adatok is fontosak és a végeselem háló szerves részei, de olyan sok formátumban léteznek és annyira feladat specifikusak, hogy ezen adatok konverziójával itt nem foglalkozunk. A konverzió során reguláris kifejezéseket [8] használunk, ezért csak az ASCII vagy szöveges formátumban tárolt végeselem hálók konverziójával foglalkozunk a továbbiakban.

3. Párhuzamos végeselem háló konverzió

Két stratégiát lehet alkalmazni, ha párhuzamosan szeretnénk a végeselem hálót konvertálni:

1. párhuzamos file olvasás és elemzés; és
2. párhuzamos elemzés a memóriában.

Az első esettel az a probléma, hogy ugyanazt a file-t egyszerre több processzus vagy szál is megnyitja, illetve olvassa. A párhuzamos file olvasás egy külön kutatási terület önmagában, hiszen a hatékony párhuzamos file olvasáshoz a hardvernek, az operációs rendszernek és a file rendszernek is szorosan együtt kell működnie [9].

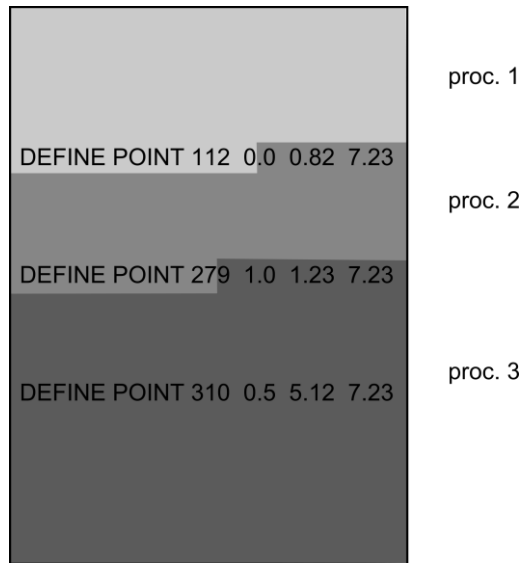
Ha a memória elég nagy, akkor a teljes végeselem hálót egyben be lehet olvasni a memóriába és ebben az esetben csak a memóriában kell dolgozni. Ebben a cikkben egy hibrid megoldást tárgyalunk, ahol a végeselem háló egy részét egy nagy bufferbe olvassuk be. Ezzel a megoldással mind a két megoldás tesztelhető, attól függően, hogy mekkora a buffer. Ugyanakkor a cikkben a továbbiakban feltételezzük, hogy mind a hardver, mind a szoftver támogatja a párhuzamos file hozzáférést és olvasást.

3.1. Elemenkénti definíciót használó végeselem hálók párhuzamos elemzése

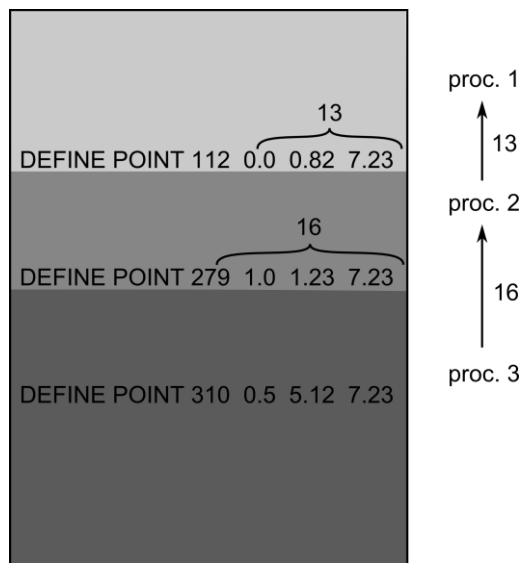
Ebben az esetben az elemek definíciója szét van szórva a végeselem hálót tároló file-ban. A file olvasás felgyorsításához a file-t feloszthatjuk régiókra, majd a file elemzését a régiókban párhuzamosan hajthatjuk végre. Ebben az esetben az a probléma, hogy két régió közötti határ kettévághat egy adat definíciót. Erre egy példát az 1. ábra mutat. Az ábrán az látható, hogy a file-t három régióra vágtuk fel és a 112. és 279. csomópontok adatait egyik processzor sem lesz képes beolvasni, hiszen az adatok egy-egy rész más-más processzoron található.

Ebben az esetben az egyik megoldás az lehet, hogy a régiók határait az adatok végén található lezáró (terminátor) karakterhez igazítjuk. Az elemenkénti definíciót használó végeselem hálóknál egy adat definíciót mindig valamilyen karakter vagy karaktersorozat zár le, például az új sor karakter. Jelen esetben az igazítás azt jelenti, hogy az első processzor kivételével minden processzor a saját régiójában a régió kezdetétől kezdve megkeresi az első lezáró karaktert. Miután megtalálták ezt a karaktert a processzorok kiszámítják a régió kezdetétől való távolságot, offsetet. Ezt az értéket minden processzor elküldi annak a processzornak, aminek egyvel kisebb azonosítója. Ilyen módon az adott processzoron a régió elejét módosítjuk, illetve az előző processzorhoz tartozó régió végét. A 2. ábra ezt a folyamatot mutatja, ahol 2. processzor 13 darab és a 3. processzor pedig 16 karaktert talál, ami az előző régióhoz tartozik. Meg kell jegyezni, hogy a karakterek száma tartalmazza a lezáró karaktert is, hiszen erre majd szükség lesz a file elemzése során. Az is fontos, hogy csak a karakterek számát küldik el a processzorok, magukat a karaktereket nem, hiszen a

szöveg beolvasása és az elemzés csak ezután következik, amikor is a konkrét karakterek beolvasása megtörténik.



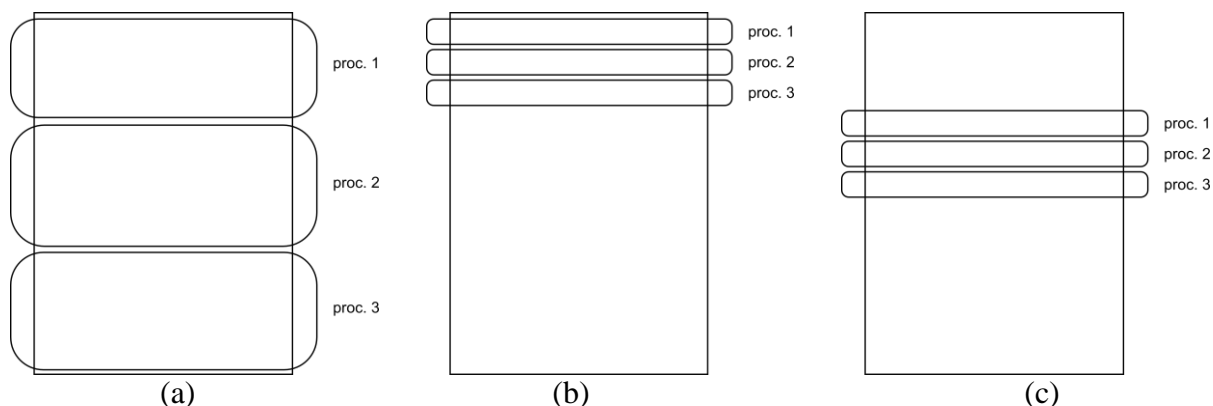
1. ábra. Elemenkénti definíciót használó végelem háló három részre darabolva. Ebben az esetben a definíciók formátuma nincs figyelembe véve.



2. ábra. Elemenkénti definíciót használó végelem háló három részre darabolva. Ebben az esetben a definíciók formátuma nincs figyelembe véve.

Egy nagyon fontos dolgot kell itt megjegyezni, hogy ebben az esetben nincs garancia a terheléelosztásra a processzorok között. Például előfordulhat, hogy minden koordináta definíció egyetlen processzor régiójába esik és így a többi processzor semmilyen hasznos munkát nem tud végezni, gyorsan végeznek a kereséssel, majd várnak, amíg csak egy processzor dolgozik. Ennek a problémának az egyik lehetséges megoldása egy „csúszó ablakos” technika, vagyis a processzorok nem a teljes file-on, hanem annak csak egy részén, szekcióján dolgoznak. A 3.a. ábra az eredeti, fenti párhuzamos implementációt mutatja, amikor a processzorok a teljes file-on dolgoznak. A 3.b. és 3.c. ábrák a csúszó ablakos technikát mutatják be. Vegyük megint azt az esetet, amikor a koordináta definíciók a file elején találhatóak. A csúszó ablakos technika alapján, amikor a processzorok a file elején

dolgoznak, akkor mindegyik elfoglalt, hiszen koordinátákat olvasnak be. Ezzel szemben, amikor a file végén dolgoznak a processzorok, akkor is egyszerre végeznek a munkával, hiszen nem találnak koordinátákat itt.



3. ábra. (a) Egy végelem hálót tartalmazó file olvasása egyszerre; (b) és (c) a csúszó ablakos technikával

3.2. Blokkenkénti definíciót használó végelem hálók párhuzamos elemzése

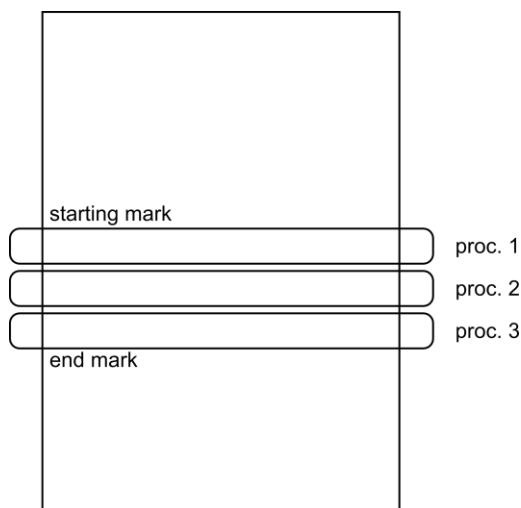
Először is megvizsgálhatunk egy végelem hálót amiben 194 720 darab csomópont és 979 860 darab tetraéder elem található. Ennek a végelem hálónak a szekvenciális konvertáló programmal történő beolvasása során lemérhetjük az egyes részek beolvasására fordított időt. Ezeket az idő értékeket tartalmazza az 1. tábla. Itt kell megjegyezni, hogy bár a táblázatban szereplő idő értékek hosszúnak tűnnek, de ezt három tényel lehet magyarázni. Először is a program csak egy „proof-of-concept” implementáció, nincs optimalizálva. Másodsor a konverziós módszer teljesen általános, bármilyen végelem hálót tartalmazó file-ra alkalmazható és szintén nem lett optimalizálva egyetlen formátumra sem. Harmadszor a konverzió során reguláris kifejezéseket használunk, melyek használata számítási szempontból drága, de teljesen általános megfogalmazást tesz lehetővé.

Geometriai blokk		Topológia blokk		Teljes konverziós idő [s]
Kezdőpont megtalálása [s]	Elemzési idő [s]	Kezdőpont megtalálása [s]	Elemzési idő [s]	
0,53	57,39	0,13	1156,39	1263,79

1.táblázat. Egy millió elemet és néhány százezer csomópontot tartalmazó végelem háló szekvenciális konverziója

A táblázat azt mutatja, hogy a geometriai és topológiai blokk beolvasása és elemzése sokkal tovább tart, mint a blokk elejének megtalálása. Ettől függetlenül a blokk elejének és végének megkeresését is párhuzamosan végezzük. Ebben az esetben is régiókra osztjuk fel a file-t, de ebben az esetben a régiók között átfedés van. Az átfedés mértéke hosszabb kell legyen mint, a blokk kezdetét meghatározó karaktersorozat. Mivel reguláris kifejezéseket használunk, ezért ennek a karaktersorozatnak a hossza nehezen meghatározható. Tapasztalatok alapján az átfedés mértéke 30 karakter hosszúságot nem szokta meghaladni, így a biztonság miatt az átfedés hossza 128 karakterben lett meghatározva. Miután a blokk kezdetét sikerült meghatározni, a megtaláló processzus értesíti a többi processzust is.

A geometriai és topológiai blokk párhuzamos beolvasásához és elemzéséhez a blokk végét is meg kell keresni, hasonló módon, ahogy ez az előzőekben ismertetésre került a kezdőpont megtalálására. Azzal a különbséggel, hogy ebben az esetben nem a file elejétől, hanem a blokk elejétől kell végezni a keresést. Egy blokk beolvasása és elemzése ezután ugyanúgy történik, mint ahogy ezt a 3.1. bekezdésben tárgyaltuk, csak itt a blokk kezdőpontja és végpontja közötti tartományban. Illetve ebben az esetben már nincs szükség a csúszó ablakos technikára, hiszen a blokkban minden processzus adatokat tud feldolgozni. Ez látható a 4. ábrán.



4. ábra. Blokk adatok párhuzamos feldolgozása

4. A párhuzamos végeselem háló olvasás és elemzés korlátai

A fent ismertetett módszernek természetesen vannak korlátai. Először is, amikor elemenkénti definíciót alkalmazó végeselem hálót párhuzamosan olvasunk be illetve dolgozunk fel, akkor fontos, hogy az elemek azonosító száma vagy szövege explicit módon legyen definiálva. Az explicit definíció az jelenti, hogy a file-ban egyértelműen szerepeljen, és egyedi legyen az egész file tekintetében.

A másik fontos feltétel a párhuzamos file olvasás és elemzés esetén, hogy minden adat definíció végén legyen lezáró karaktersorozat. Általában ez az új sor karakter, de nem minden esetben. Ennek fontossága a fenti leírások alapján is látható, mivel például erre szükség volt a régió határok igazításánál is.

5. A konverzió lehetséges kimeneti formátumai

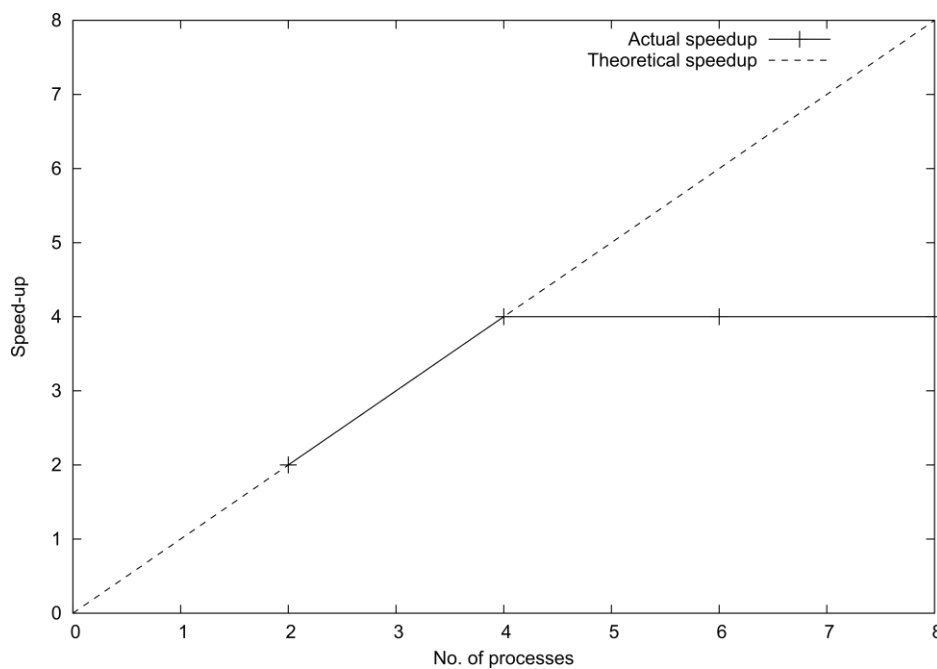
A hosszú távú tárolás szempontjából fontos, hogy mi legyen a konverzió kimenete, ami valóban megőrizhető. Jelenleg erre nincs végső konszenzus az irodalomban, de az egyik legerősebb trend az XML formátumra épülő file-ok használata [6]. Ennek az az oka, hogy az XML alapú file-okat „önleíróként” is szoktak nevezni. Az önleíró tulajdonság abból ered, hogy az XML file-ok nem csak az adatokat tartalmazzák, hanem azok jelölését (markup) is. Az XML jelölésnek több előnye is van: a szintakszisa nagyon pontosan definiált; egyszerű de akár kiterjeszhető további tulajdonságokkal; ember is képes olvasni és értelmezni; számítógép is képes hatékonyan feldolgozni; illetve lehet olyan jelölést alkalmazni, ami csak

a tartalomra vonatkozik és nem a megjelenésre. Ugyanakkor az egyik legnagyobb problémát az jelenti, hogy a jelölésként használt részek és a konkrét adatok közötti arány igen magas lehet, például:

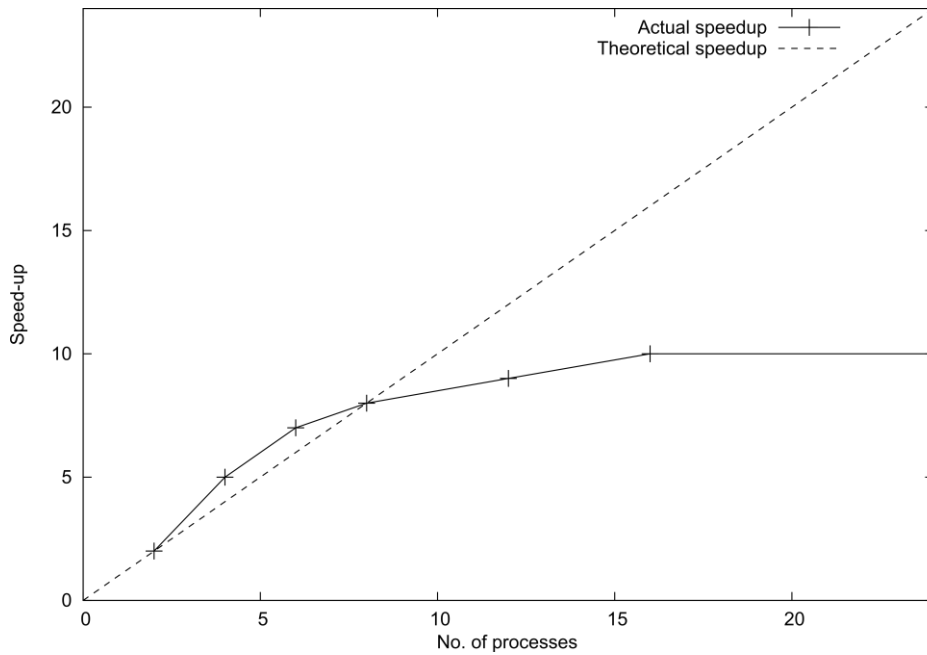
```
<vertex><id>112</id><coords>0.0 1.0 0.0</coords></vertex>
```

6. Mintapéldák

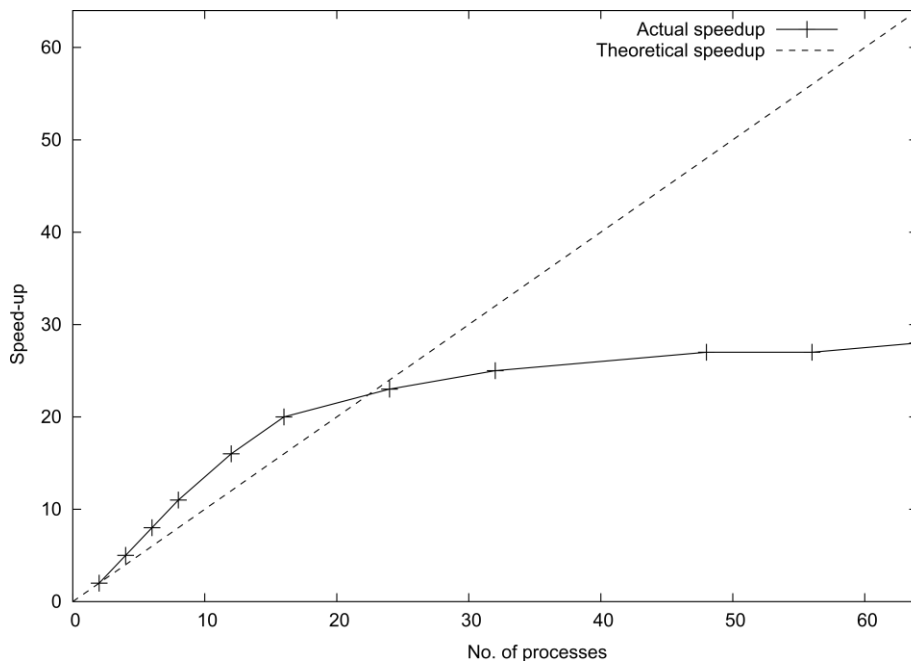
A párhuzamos végelem háló olvasás és elemzés tesztelésére egy Intel Pentium G620 dual core processzorral rendelkező számítógépet használtunk. A processzor sebessége 2.6 GHz. A számítógép 4 GB RAM-al rendelkezik és 64 bites Microsoft Windows 7 operációs rendszert használ. A program az MPICH [10] programozási környezettel lett implementálva és az 'mt' kommunikációs csatornával, ami több szálal futtatást jelent. A teszt végelem hálóknak 100 000, 200 000 és 500 000 csomópont található és a file-ban egyenletesen vannak szétosztva. Ezek az esetek fogják reprezentálni a legjobb esetet. A gyorsulás (speedup) mértékét a 100 000 csomópont esetén a 5. ábra mutatja, míg 200 000 csomópont esetén a 6. ábra és 500 000 csomópont esetén a 7. ábra mutatja a gyorsulás mértékét. A mérések során csak a végelem hálót tartalmazó file olvasását és elemzését mértük. Ez egy kicsit félrevezető lehet, hiszen az adatok összefésülése is időt igényel.



5. ábra. Gyorsulás (speed-up) 100 000 csomópontot tartalmazó végelem háló párhuzamos beolvasása esetén



6. ábra. Gyorsulás (speed-up) 200 000 csomópontot tartalmazó végeelem háló párhuzamos beolvasása esetén



7. ábra. Gyorsulás (speed-up) 500 000 csomópontot tartalmazó végeelem háló párhuzamos beolvasása esetén

Az ábrákból az látható, hogy bár a teszt számítógépnek csak két core-ja van az algoritmusok nagyon jó gyorsulást produkálnak kettőnél több processzus esetén is. Az algoritmusok gyorsulása ráadásul szuper lineáris 200 000 és 500 000 csomópont esetén. 500 000 csomópontot tartalmazó végeelem háló beolvasása szekvenciálisan 760,5 szekundumig tart, míg 64 processzussal a párhuzamos beolvasás 26,7 szekundum.

7. Összefoglalás

A cikk párhuzamos stratégiákat és algoritmusokat mutatott be végeelem háló párhuzamos olvasására és elemzésére. A cikkben néhány futtatási eredmény is bemutatásra került,

amelyek nagyon biztatóak, hiszen bizonyos processzus számig az algoritmusok szuper lineáris gyorsulást mutatnak. Ez azt jelenti, hogy ez a konverziós technika szerepet játszhat akár a végelelem hálók hosszú távú megőrzésében, bár még további kísérletekre van szükség. Ezen felül szükséges egy kimeneti formátum kidolgozása is, vélhetően XML formátumban.

8. Köszönetnyilvánítás

A kutatást támogatta a "Szuperszámítógép, a nemzeti virtuális laboratórium" c. TÁMOP-4.2.2.C-11/1/KONV-2012-0010 sz. pályázat.

Irodalomjegyzék

[1] www.openfoam.com

[2] LS-DYNA, Keyword users manual, Livermore Software Technology Corporation, April 2003.

[3] L. B. Schroeder Will, Martin Ken, The visualization toolkit: an object oriented approach to 3D graphics. Kitware Inv, 2006.

[4] P. Ivanyi, "Finite element mesh conversion based on regular expressions," *Advances in Engineering Software*, **51** (2012), 20–39.

[5] http://www.nsf.gov/pubs/policydocs/pappguide/nsf11001/gpg_2.jsp#dmp

[6] U. M. Borghoff, P. Rödig, J. Scheffczyk, and L. Schmitz, Long-Term Preservation of Digital Documents: Principles and Practices. Springer, 2010.

[7] M. Portnoy, Virtualization Essentials. Sybex, 2012.

[8] J. Goyvaerts and S. Levithan, Regular Expressions Cookbook. O'Reilly Media, 2012.

[9] J. Dongarra, I. Foster, G. Fox, W. Gropp, K. Kennedy, L. Torczon, and A. White, Eds., Sourcebook of Parallel Computing. Morgan Kaufmann Publishers, 2003.

[10] <http://www.mpich.org>

Kvantumkeresés mérnöki megközelítésben

Quantum search in engineer approach

Varga Tamás^a, Dr. Bacsárdi László^b

^aNymE SKK Informatikai és Gazdasági Intézet
varga@inf.nyme.hu

^bNymE SKK Informatikai és Gazdasági Intézet
bacsardi@inf.nyme.hu

Absztrakt: A technika fejlődésével a rendelkezésünkre álló adatmennyiség rohamosan növekszik. Jó példa erre a különböző szenzorok által folyamatosan szolgáltatott adatok tömege. Ezek azonban még csak nyers adatok, információtartalom nélkül. A hatékony és gyors információnyerés egyik lehetséges megoldása a kvantuminformaticai algoritmusok alkalmazása.

A kvantuminformatica a kvantummechanikán alapuló informatikát jelenti. Ennek alapkövei a kvantumbit és a kvantumregiszter. A kvantum párhuzamosság az egyik fő erőssége a kvantum-alapú algoritmusoknak, mellyel számos számításiigényes problémát meg lehet oldani, például a rendezetlen adatbázisban való hatékony keresést. A Grover-algoritmussal, melyet a 90-es évek közepén publikáltak, hatékonyabban tudunk keresni rendezetlen adatbázisokban, mint klasszikus módon. Ugyanakkor a kvantumszámítógépekre tervezett kvantumalgoritmusok szimulálása klasszikus számítógépeken kihívást jelentő probléma.

Munkánk során különböző kvantum algoritmusok szimulálására fejlesztettünk egy Java alapú rendszert, mely alkalmas például a Grover-algoritmus szimulálására klasszikus környezetben. A szimuláció során megfigyelhetőek az egyes szimulációs lépések, ezáltal az egyes algoritmusok hatékonysága elemezhető.

Kulcsszavak: Grover-algoritmus, kvantumalgoritmus, kvantuminformatica

Abstract: The available amount of data is increasing rapidly with the evolution of technology. For example, the mass of data is continuously supplied by various sensors. However, these are only raw data without information content. One possible solution to the efficient and rapid use of information retrieval the quantum algorithms.

The quantum-based computing is based on the laws of quantum mechanics. We can define quantum bit (qubit) and quantum register (or quregisters) as basic brick stones. From the viewpoint of quantum algorithms, the not quantum-based solutions are named as classical. The quantum parallelism is one of the major powers behind the quantum algorithms. It allows solving many complex problems in an efficient way, e.g., finding prime factors of large numbers or searching in unsorted database. The Grover's algorithm was published in middle of '90s, and it can search in an unsorted database much more efficiently than classical computers. To run a quantum algorithm, we need a quantum computer. Due to the lack of a quantum computer, the simulation of quantum algorithms in classical computer is a challenging problem.

We have developed a Java-based system to simulate different quantum algorithms which are based on elementary quantum gates. Our program is suitable to simulate Grover's algorithm in classical computer as well. We can observe the simulation steps and analyse the effectiveness of a given algorithm.

Keywords: Grover's algorithm, quantum algorithm; search in unsorted database

1. Bevezetés

A rendezetlen adatbázisokban való hatékony keresés a mai kor embere számára kiemelkedően fontos. Klasszikus módon, különösen nagy adathalmazok esetén, a probléma rendkívül idő- és számításigényes feladat. A klasszikus informatika fejlődésének van egy korlátja, melyet Gordon E. Moore, az Intel cég egyik alapítója mondott ki 1965-ben. Eszerint az integrált áramkörök összetettsége körülbelül 18 hónaponként megduplázódik [1], viszont ez a tendencia nem folytatódhat a végtelenségig. Jelenleg az Intel cég 22 nanométeres csíkszélességen készült processzorokat gyárt [2] és még idén a 14 nanométeres variánsok érkezése is várható, míg 2016-ra 10 nanométeresre szeretnék zsugorítani a gyártási technológiát. Bár a chip gyártási technológiája folyamatosan fejlődik, de az előző generációkhoz képest nem tapasztalható olyan kiugró teljesítménynövekedés, mint az elmúlt évtizedben. Az egyre komplexebb és folyamatosan dráguló chipgyártás következtében 2014-2015-ben elérhetjük azt, hogy az egy tranzisztorra levetített normalizált költségek nem csökkennek a kisebb csíkszélességre való átállás hatására, ami gyakorlatilag életképtelenné teszi a Moore-törvényt. Mostanára elérkezett az ipar ahhoz a ponthoz, amelynél a gyártási technológiák összetettsége minden lépéssel exponenciálisan nő. Eddig minden egyes generációval sikerült javítani a processzorok sebességén, fogyasztásán és árán, hamarosan azonban a gyártóknak dönteniük kell, hogy a három dolog közül melyik kettőt részesítik majd előnyben. Számos olyan probléma létezik, melyre az említett számítási kapacitás megsokszorozásával választ kaphatnánk, például a matematikában a prím tényezőkre bontás, vagy nagy adathalmazokban való gyors keresés, azonban a fennálló fizikai korlátok miatt ez a jelenlegi eszközeinkkel nem megvalósítható. A probléma megoldását jelentené a kvantumszámítógépek alkalmazása, melyeken kvantumalgoritmusokat futtatnánk. A kutatás célkitűzése, hogy a kvantum keresés során az elméletben meghatározott számítási lépésszámot a gyakorlatban is igazoljuk, valamint több kvantumalgoritmus modellezésére is alkalmas szimulátort fejlesszünk.

2. A kutatás elméleti háttere

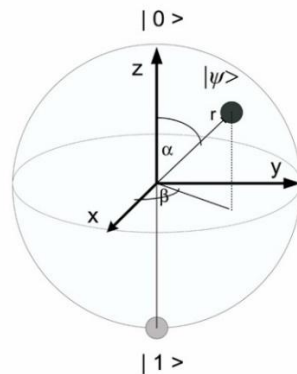
A kvantuminformatika ma elméleti tudományterületnek számít, hisz a tényleges megvalósítások elmaradnak az elméleti kutatások mögött. A kvantum világban például elméletben képesek vagyunk teleportációra [3], de komoly kutatások folynak a műholdas, azaz űrbéli kvantumkommunikáció megvalósításáról, fejlesztéséről is [4].

A kvantummechanikában a következő posztulátumokból indulunk ki [5]:

1. Bármely zárt fizikai rendszer aktuális állapota felírható egy \vec{v} állapotvektorral, melynek komplex koefficiensei és egységossza van egy V Hilbert térben (egy komplex lineáris vektortérben (állapottér), egy belső szorzattal kiegészítve).
2. Egy zárt fizikai rendszer időbeli evolúciója unitér transzformációkkal írható le, amik kizárólag az evolúció kezdési és befejezési időpontjától függenek.

3. Bármilyen kvantum mérés leírható mérési operátorok egy halmazával $\{M_m\}$, ahol m a mérés lehetséges elemeit jelöli. A mérést a mérési valószínűséggel és a mérés utáni állapottal definiálhatjuk. A mérés egy kvantum rendszerben nem megfordítható folyamat, vagyis irreverzibilis, így ez az operátor nem tekinthető unitérnek.
4. Egy összetett W fizikai rendszer állapottere meghatározható az őt alkotó különálló rendszerek tenzorszorzataként.

A klasszikus számítógépek bitekkel dolgoznak, melyek állapota bármely időben csak 0 vagy 1 lehet, míg egy kvantumbit (angol szakszóval qubit) egyazon időben tartalmaz olyan komponenseket, melyek nullának, illetve egynek felelnek meg [6]. Más megfogalmazásban a kvantumbit egyesnek és nullának bármely kvantum szuperpozíciójában lehet. Fizikusok a kvantumbitét például egy elemi részecske spinjével szemléltetik, ahol például a „fel” spin értéknek az 1-es, a „le” spin értéknek a 0 felel meg. A Dirac által bevezetett jelölésrendszerben a 0-nak $|0\rangle$ (ejtsd ket nulla), az 1-nek $|1\rangle$ felel meg. Egy általános kvantumbit az alábbi alakban írható fel: $|\varphi\rangle = a|0\rangle + b|1\rangle$, vagyis egy qubit ezen két bázisállapot szuperpozíciójában lehet, ahol az első posztulátum alapján $|a|^2 + |b|^2 = 1$ ($a, b \in \mathbb{C}$), továbbá $|0\rangle$ és $|1\rangle$ egymásra merőleges bázisállapotok. Egy tetszőleges kvantumbit Bloch-gömbön való szemléltetését láthatjuk az 1. ábrán.



1. ábra Kvantumbit szemléltetése a Bloch-gömbön

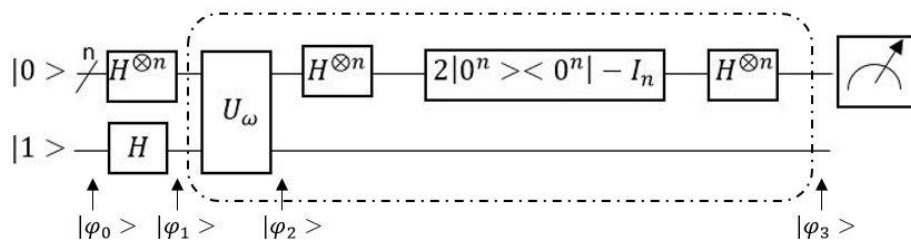
Egy kvantumregiszter a klasszikus világban ismert regiszterrel azonos szerepet tölt be. Míg klasszikus esetben biteket tárol egy regiszter, eddig a kvantum világban kvantumbiteket. A fő különbség az adott méretű regiszterekben tárolható adatmennyiségben van. A klasszikus n bites regiszterben 2^n érték közül tárolhatunk egyet, viszont egy n kvantumbites kvantumregiszterben a 2^n lehetséges érték mindegyike lehet egyszerre. Az elektronikában, digitális technikában használt logikai kapukhoz hasonlóan a kvantuminformatikában is beszélhetünk hasonló szerepet betöltő egységekről [7]. A kvantumregisztereken, kvantumbiteken végezhetünk valamilyen műveletet alkalmazásukkal, kombinálásukkal. Logikai kapuk egymás utáni alkalmazásával kvantum áramköröket építhetünk. Klasszikus esetben, ha egy állapotot vektornak tekintünk, akkor egy ezen végzett transzformáció leírható mátrix alakban is. Nincs ez másként a kvantum kapuk esetében sem. Jelentős különbség azonban, hogy míg klasszikus esetben a transzformációt egy sztochasztikus mátrix biztosítja (melyben egy adott i állapotból valamely másik állapotba való átmenet aggregált valószínűsége 1 kell, hogy legyen), addig a kvantum világban ennek unitérnek kell lennie (unitér mátrix esetében az amplitúdók négyzetösszegének kell 1 értéket adniuk).

3. Keresés rendezetlen adatbázisban: a Grover algoritmus

A Grover-algoritmus egy olyan kvantum-algoritmus, mely egy N elemű rendezetlen adatbázisban (halmazban) $O(\sqrt{N})$ idő alatt $O(\log_n)$ tárhely felhasználásával tud keresni, jóval hatékonyabban, mint a klasszikus számítógépeken futó, bináris elven működő algoritmusok, melyek idő költsége $O(N)$. Egy ilyen keresési példa, amikor a nevek sorrendjében rendezett telefonkönyvben kell egy adott telefonszámhoz tartozó nevet megkeresni.

Az alap Grover algoritmus működéséhez szükséges két kvantumregiszter. A felső n darab qubitet tartalmaz, mely $N=2^n$ méretű adathalmazt jelent. Ezt a regisztert inicializáljuk $|0\rangle$ -val, a másikat pedig $|1\rangle$ -gyel. Az első n -bités Hadamard kapu az állapotok szuperpozícióját állítja elő.

Az algoritmus blokkvázlata látható a 2. ábrán.



2. ábra a Grover algoritmus blokkvázlata

Jelöljük a 2. ábra felső n bites regiszterének kezdő állapotát $|\gamma_0\rangle$ -val, az alsó regiszterét pedig $|\psi_0\rangle$ -val. A $|\gamma_1\rangle$ állapotban a rendszer az alábbi kifejezéssel írható le:

$$|\gamma_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$$

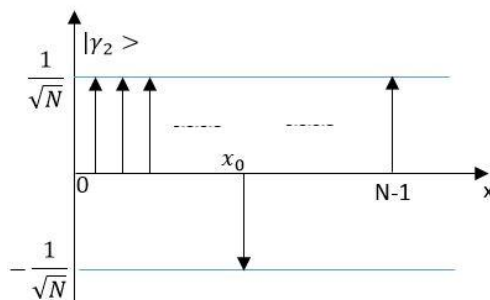
Vagyis az elemek kezdeti valószínűségi amplitúdói azonosak. Az inicializálás utáni első lépés, hogy a keresett elem helyét valahogy megtaláljuk, erre ad megoldást az Orákulumnak nevezett speciális kapu, melynek működése:

$$O: |x\rangle|y\rangle \rightarrow (-1)^{f(x)} |x\rangle|y\rangle, \text{ ahol } f(x) = \begin{cases} 1, & \text{ha } x = x_0 \\ 0 & \text{különben} \end{cases}$$

Vagyis másképp felírva

$$O: |x\rangle|y\rangle \rightarrow |x\rangle|y \otimes f(x)\rangle$$

Az Orákulum utáni állapotot vizuálisan szemlélteti a 3. ábra.



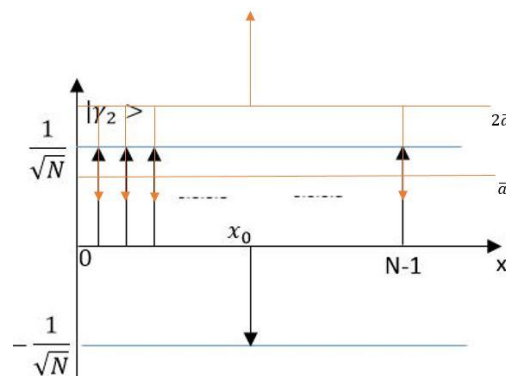
3. ábra A felső kvantumregiszter állapota az Orákulum alkalmazása után

Az ábrán látható, hogy a különbség még csak a kvantumos tartományban létezik, a keresett elem valószínűségi amplitúdója negatív. Ezt már kvantuminformatikailag meg tudjuk

különböztetni, de ha most hajtánánk végre egy mérést, akkor a valószínűségi amplitúdó négyzetét kapnánk vissza, amely mind a keresett, mind a nem keresett elemek esetében ugyanaz. Ezért a következő teendőnk az, hogy a keresett elemek amplitúdóját erősítsük, egészen addig, amíg már klasszikusan is mérhető különbség lesz a keresett és nem keresett elemek között. A Grover algoritmus során az amplitúdó erősítését az átlagra való tükrözéssel érjük el. Az átlagot az alábbi képlet szerint számítjuk:

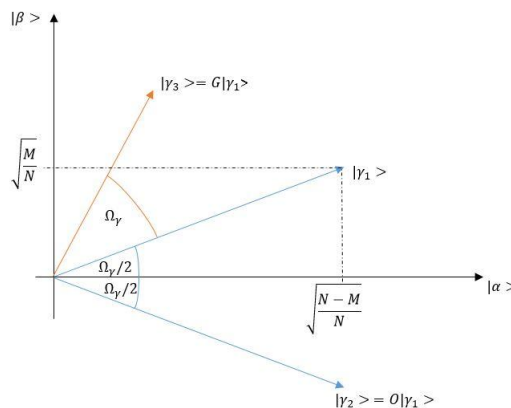
$$\bar{a} = \frac{1}{N} \sum_{x=0}^{N-1} \gamma_{2,x}$$

Az átlagra való tükrözéssel elérjük azt, hogy az addig negatív elemek amplitúdója átlag feletti lesz, míg a többi elemé csökkenni fog. Ezt szemlélteti a 4. ábra.



4. ábra Amplitúdó erősítés az átlagra való tükrözéssel

Vezessünk be két halmazt, legyen $S = \{f_x = 1\}$, $\bar{S} = \{f_x = 0\}$. A megoldásra nem vezető eredmények szuperpozícióját jelölje $|\alpha\rangle$, a megoldásra vezetőket pedig $|\beta\rangle$. A Grover operátor geometriai interpretációját szemlélteti az 5. ábra.



5. ábra A Grover-operátor geometriai interpretációja

Látható a kezdeti állapot, mikor a vektorunk és a $|\alpha\rangle$ által bezárt szög $\Omega_g/2$, és a vektorunk az alábbi képlet alapján számítható:

$$|\gamma_1\rangle = \sqrt{\frac{N-M}{N}} |\alpha\rangle + \sqrt{\frac{M}{N}} |\beta\rangle$$

Az orákulum negálását követő állapotot a $|\gamma_2\rangle$, majd az átlagra való tükrözést követően a $|\gamma_3\rangle$ mutatja, hogy a transzformált vektorunk az eredeti állapothoz képest Ω_g szöget zár be, vagyis

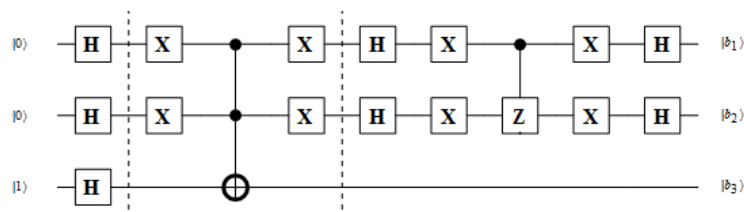
$|\beta\rangle$ irányába mozdult el. Célunk, hogy a vektorunkat $|\beta\rangle$ -ba forgassuk. A Grover operátort tehát egymás után többször kell alkalmazni, hogy a keresett elem nagy (közel 1) valószínűséggel kerüljön visszaadásra. Azt, hogy hányszor kell a Grover operátort alkalmazni, azt az alábbi képlet adja meg.

$$L_{opt_0} = \left[\frac{\pi}{4} \sqrt{\frac{M}{N}} - 1 \right] \cong \frac{\pi}{4} \sqrt{\frac{M}{N}},$$

melyben a számláló fejezi ki a szöveget $|\gamma_1\rangle$ és $|\beta\rangle$ között, a nevező pedig a forgatás léptékét, azaz L_{opt_0} meghatározása $O\sqrt{\frac{N}{M}}$ költségű.

4. Kvantum keresés szimulátor

Java alapú szoftvert fejlesztettünk, amivel elemezhetőek a különböző kvantumalgoritmusok. Az algoritmusoknak megfelelő áramkörök kvantumkapukból épülnek fel. Az alkalmazásunk képes a Grover algoritmus szimulálására klasszikus környezetben, miközben az egyes lépések megfigyelhetőek és elemezhetőek. A kvantumalgoritmusok szimulációinak egyik legfőbb nehézsége a kvantumbitek számával exponenciálisan arányos memóriaszükséglet. A Grover algoritmusnak megfelelő áramkört szemlélteti a 6. ábra.



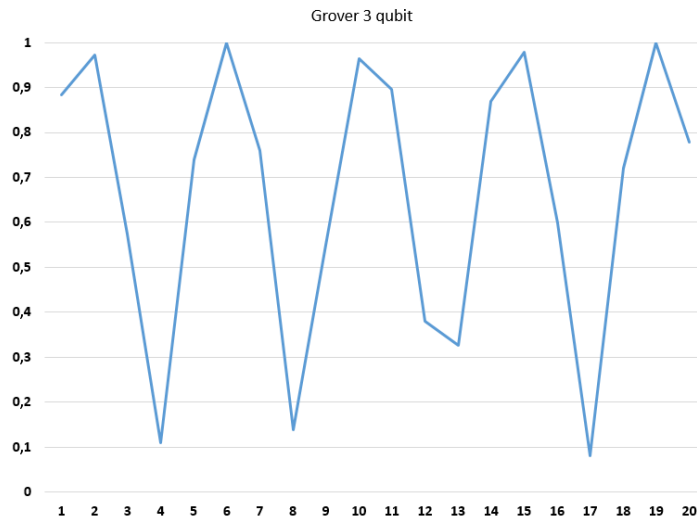
6. ábra a Grover algoritmust implementáló áramkör vázlatja

Inicializáláskor meg kell adni a regiszter méretét és kezdő állapotát, majd ezt követően különböző kvantumkapukat adhatunk az áramkörhöz. Az X, Z és H jelek a Pauli-X, Pauli-Z és Hadamard kapukat szimbolizálják, a fekete pontok pedig a vezérelt CNOT és Z kapukat. A szimuláció lépésenként és egy lépésben is futtatható, az eredmények pedig egy log fájlból visszaolvashatóak.

A szimuláció során azt vizsgáltuk, hogy a szimuláció során kapott optimális lépésszámok megegyeznek-e az elméleti részben szereplő $O\sqrt{\frac{N}{M}}$ értékkel. A vizsgálatot több adatbázis méretre is lefuttattuk, az eredményekről a következő fejezetben olvashatnak.

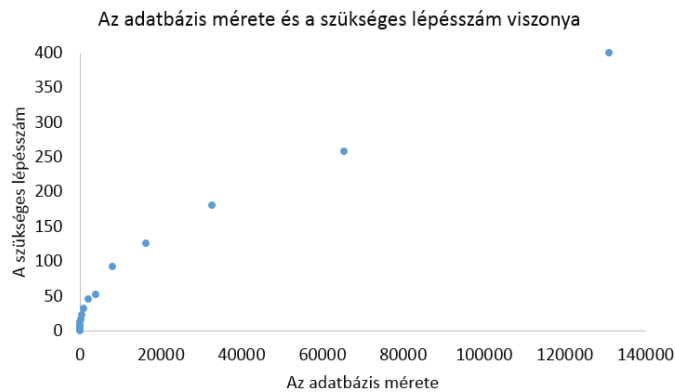
5. Eredmények, értékelés

Kezdetben kisméretű adatbázissal dolgoztunk, mindösszesen 3 kvantumbites regisztert vizsgáltunk. Képlet alapján az optimális iterációs szám 2, viszont a szimulációk során kapott grafikon alapján elmondhatjuk, hogy a Grover operátor további alkalmazásával jobban meg tudjuk közelíteni az 1 mérési valószínűséget. A 3 kvantumbites regiszterhez tartozó grafikon szemlélteti a 7. ábra.



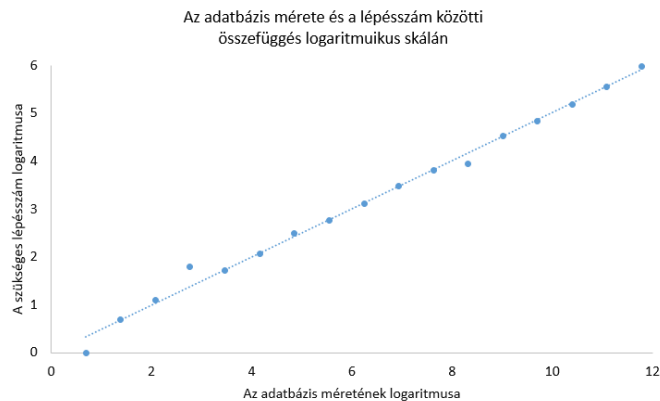
7. ábra Az egyes lépésszámokhoz tartozó valószínűségi értékek

Látható, hogy esetünkben 6 lépés után az eredeti 0,97229 helyett 0,99993-ra növelhető a keresett érték mérési valószínűsége. Megjegyezzük, hogy az iterációk növelésével elérhető pontosságnövekedés mérnöki oldalról nem minden esetben szükséges. Ezen vizsgálatokat további regiszterméretekre is elvégeztük, az eredményeket a 8. ábra mutatja.



8. ábra Az adatbázis mérete és a szükséges lépésszám viszonya

A fenti ábrán látható, hogy a görbe gyökfüggvény jelleget mutat. Ezt igazolandó az értékeket logaritmikus skálán is ábráztuk, melyet a 9. ábra szemléltet.



9. ábra Az adatbázis mérete és a szükséges lépésszám viszonya logaritmikus skálán

A pontokra illesztett egyenes meredeksége nem pontosan 0,5, melynek oka lehet, hogy az egyes lépéshosszok nem pontosan egyeznek meg a vizsgált regiszterméret gyökével.

A szimuláció futtatása során nyert adatok alapján azt a következtetést tudjuk levonni, hogy kvantum alapú keresés során várt, az adatbázis méretének gyökével arányos lépésszámot a szimulációval is igazoltuk. A szoftver alkalmas a kvantum alapú keresés egyes lépésinek bemutatására, a Grover-algoritmus demonstrálására. Az algoritmus hatékonyságának demonstrálásán túl vizsgálni fogjuk, hogy milyen változásokat jelent a Grover-algoritmus működésében, ha az algoritmust felépítő egyszerű kvantumkapuk hibamentességét megkérdőjelezzük. Kutatásunkban elemezzük majd, hogyan működik a kvantum alapú keresés rendezetlen adatbázisokban, ha a különböző kapuk (bizonyos valószínűséggel) hibás műveletet hajtanak végre.

Irodalomjegyzék

- [1] G. E. Moore: Cramming more components onto integrated circuits. *Electronics*, **38 (8)** (1965), 114–117.
- [2] Intel 22nm Technology <http://www.intel.com/content/www/us/en/silicon-innovations/intel-22nm-technology.html> (Utolsó megtekintés dátuma: 2014. június 16.)
- [3] C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, W. K. Wootters: Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels. *Physical Review Letters*, **70 (13)** (1993), 1895–1899
- [4] L. Bacsárdi: On the way to Quantum Based Satellite Communication. *IEEE Communications Magazine*, **51 (8)** (2013), 50–55
- [5] S. Imre és F. Balazs: Quantum Computing and Communications: An Engineering Approach. *Wiley, Chicester* (2005), 12–15
- [6] Bacsárdi L. Galambos M., Imre S.: Kvantumalapú algoritmusok. Informatikai Algoritmusok, III. kötet. *Mondat Kiadó, Vác* (2013), 1785–1827
- [7] M. A. Nielsen és I. L. Chuang: Quantum Computation and Quantum Information. *Cambridge University Press, Cambridge* (2010), 17–24

Mozgásérzékelős vezérlőn alapuló rehabilitációs rendszer fejlesztése

Developing motion sensing input device based computer assisted rehabilitation system

Balla Tibor^a

^aDebreceni Egyetem, Informatikai Kar, Informatiótechnológia Tanszék

balla.tibor@inf.unideb.hu

Absztrakt: A cikkben bemutatott rendszer olyan számítógéppel támogatott rehabilitációs játékok és szoftverek fejlesztését segíti, melyek mozgásérzékelős vezérlőt használnak. Egyszerre próbál megfelelni a telemedicina, telerehabilitáció és a komoly játékok követelményeinek. A fejlesztett keretrendszer a járóbeteg terápia, rehabilitációs tevékenységhez nyújt támogatást, és ezentúl használható idősek és kis gyermekek mozgásfejlesztésében. Segítséget nyújthat az egyes területek mozgásán túl a rehabilitáció állapotának követésében, segítségével folyamatosan figyelhető a fejlődés, és feltárhatóvá válhatnak az esetlegesen elhanyagolt területek is. A rendszer erőssége, hogy a egy teljes kollaborációs teret biztosít, mely segítségével a felek térben és időben függetlenül képesek kommunikálni. A rehabilitációt végző személy képes ellenőrizni, hogy a rehabilitált személy milyen gyakran, milyen intenzitással és milyen pontossággal végzi a kiadott feladatokat.

Kulcsszavak: telerehabilitáció, kollaboráció, mozgásérzékelős vezérlő, komoly játék

Abstract: During my research I develop a videogame based physical rehabilitation system and framework which allows the monitoring of the progress of patients from anywhere and anytime, without regard to time and space. The software use the motion capture technology of the Microsoft Kinect. The major goal of this telerehabilitation system is to let patients perform complex rehabilitation programs using entertaining therapies in their own homes.

Keywords: telerehabilitation, collaboration, motion sensing input device, serious games

1. Bevezetés

A játék motiváló erejét nem kell bemutatni. Már igen korán felfedezték, hogy bizonyos területeken a játék relevánsan növelheti a készségek és képességek fejlesztését. Már Plátón törvényeiben is olvashatunk arról, hogy a fiatal fiúknak hogyan oktatták a számtant játékosan almákkal, és arról, hogy valódi szerszámok kicsinyített másait adták azoknak a tehetséges gyermekeknek a kezébe, akiből építéssz akartak nevelni. [1] Arisztotelész úgy vélte, hogy a játéknak céllal kell történni, és a gyermekeknek a játszás során kell elsajátítani azokat a képességeket, melyeket felnőtként is hasznosítani tudnak. Ezek a gondolatok köszönnek vissza Karl Gross filozófia professzor játékról szóló műveiben, melyben igyekszik befoltozni azt a rést, melyet a játék ütött a darwini evolúciós elméleten. Hogyan illik bele a játék - legyen az emberi, vagy állati - egy olyan elméletbe, mely szerint az élőlények tevékenységei elsősorban a faj- és a létfenntartására irányulnak? A válasz: a készségek és képességek fejlesztése. Gross szerint a fejlettebb fajok játékaiban azok a tevékenységek jelennek meg, melyek a felnőtt állat viselkedésének: a pázásnak, a vadászatnak és a harcnak a részei. Az emberi játék, akárcsak az állatkölykök játéka, olyan viselkedési mintákat tartalmaz, melyek a felnőtt korban egységes láncolatot alkotnak. Mivel a készségek és képességek elsajátítása

játékokon keresztül evolúciós sajátosság, nem meglepő, hogy a rehabilitációban és rehabilitációban is fontos szereplők a játékok. [2]

2. Rehabilitáció és fejlesztés

A fejlesztett keretrendszer a járóbeteg terápiás, rehabilitációs tevékenységhez nyújthat támogatást.

A rehabilitáció az egészségügyi törvény szövege szerint egy komplex fogalom, mely egyaránt irányul a sérült egyén szellemi és testi épségében beállt változás helyreállítására, és az egyén szociális beilleszkedésének segítésére.

„A rehabilitáció olyan szervezett segítség, amelyre egészségükben, testi és/vagy szellemi épségükben tartósan vagy véglegesen károsodott emberek rászorulnak a társadalomba, a közösségbe történő visszailleszkedésük érdekében. Az orvosi rehabilitáció célja, hogy az egészségi állapotukban károsodottakat és a fogyatékosokat - az egészségtudomány eszközeivel - meglévő képességeik (ki) fejlesztésével, illetve pótlásával segítsék abban, hogy önállóságukat minél teljesebb mértékben visszanyerjék, és képessé váljanak a családba, munkahelyre, más közösségbe való beilleszkedésre.” [3]

A rehabilitáción túl az alkalmazás használható idős és gyermekek mozgásfejlesztésében. A gyermekfejlesztés az egyik olyan terület, ahol nagyon elfogadott a játékok használata. Tipikusan olyan problémákra lehet használni, melyek organikus okokra nem vezethetők vissza:

- nagy- és finommozgások zavara
- testséma zavara
- testtudat, a vizuális-poszturális testséma zavara
- a testhatárok érzékelésének zavara
- térészlelés zavara
- térirányok zavara
- kognitív térkép (tájékozódás) zavara
- pozíció meghatározás zavara

3. Komoly játékok

A 20. század a számítógépes játékok megjelenésével újraformálta a játék fogalmát. Az emberi faj történelmében soha sem volt ekkora szerepe a játéknak. [4] Megjelentek a komoly játékok, melyekben a játék elsődleges szórakoztató funkciója mellett megjelenik egy másodlagos, komolyabb cél. A komolyabb cél eléréséhez a játék maga a motiváció és a jutalom is egyben. A komoly játékok az első pillanattól kezdve jelen vannak az egészségügyben, és ma már senki sem vitatja a hatékonyságukat. A korai alkalmazások, és kísérletek mindegyikében az adott betegségekhez és kísérletekhez fejlesztett vezérlőt és virtuális környezetet használtak. Egy korábbi kutatásában Erb-féle bénulásban szenvedők rehabilitációjához egy erre a célra fejlesztett speciális vezérlő segítségével kontrollált játékot használtak eredményesen. [5] A hagyományos, fájdalmas, ismétlődő mozdulatokra támaszkodó fizioterápiával szemben a résztvevő mozdulatai spontánok és önkéntesek voltak. Más kutatások a felsőtest motorikus

funkcióinak javítását célozták meg agyvérzéses betegeknél. [6] [7] Ezekben a kísérletekben is releváns javulást értek el a kontrollcsoporthoz képest.

Az utóbbi években megjelenő mozgásérzékelős játékvezérlők – mint a Nintendo Wii, vagy a Microsoft Kinect – egészen új dimenziókat nyitottak ezen a területen. A valós térben megtett mozdulatok hatása a virtuális térben teljesebbé válik. A virtuális térben a „fizika törvényeit” könnyedén a „sérült” egyén képességeihez igazíthatjuk, így növelhető a sikerélmény és vele együtt a motiváció is. A játékos a játékba merülve észre sem veszi, hogy milyen sok időt tölt mozgással, és végzi el azon mozdulatokat akár több százszor, melyek ismétlése egy unalmas terápián néhány tízszer is megterhelő lett volna. Ezen eszközök megjelenése előtt is léteztek olyan megoldások, melyek hasonló elvek mentén igyekeztek segíteni a rehabilitációt. Az ilyen rendszerek több százezer dollárba kerülnek, a használatukhoz komoly szakértői gárda, és drága eszközök sokasága szükséges. [8]

A Microsoft és a Nintendo új vezérlői lehetőséget biztosítanak olyan rendszerek kialakítására, melyek olcsón, szinte otthoni körülmények között is lehetővé teszik az egyszerű rehabilitációs feladatok elvégzését. Ehhez sokszor nincs is szükség speciális szoftverre, elegendő az ezekkel a vezérlőkkel használható játékokkal játszani. [9] Ezzel a megközelítéssel magát a mozgást motiválja az eszköz, amely sok esetben elegendő is, hiszen nem mindig egy konkrét testrész megmozgatása a cél, ilyen többek között az idősek rehabilitációja. [10] Más alkalmazások pedig egyszerűbb mozdulatok ismétlését követelik meg a játékosoktól (izomsorvadásos, agyvérzéses betegek rehabilitációja), az alkalmazás ellenőrzi a mozdulat helyességét, és jutalmazza a játékost, ha az kellően pontos volt. [11]

4. A keretrendszer



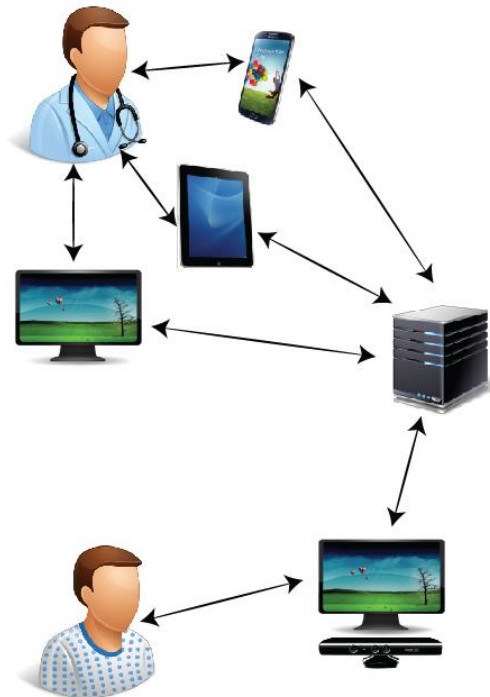
1. ábra <reMove> rendszer

Az fejlesztés során elsődleges cél volt egy olyan keretrendszer megalkotása, mely egyszerre próbál megfelelni a telerehabilitáció és a komoly játékok követelményeinek. Egyszerre támaszkodik a játékok motiváló erejére, és biztosít egy olyan felületet, ahol az elvégzett feladatokról készített részletes statisztikák és riportok segítségével nyomon követhetővé válik az alkalmazást használó személy állapota, annak fejlődése. A rendszer eszközt biztosít ahhoz, hogy egy ezzel a keretrendszerrel fejlesztett Microsoft Kinect mozgásérzékelős vezérlőt

használó játék vagy alkalmazás a működése közben előálló adatokat egyszerűen vizsgálja és azok eredményeit látványos vizualizációs eszközökkel tegye elérhetővé a terápiát vezető személy számára.

A keretrendszer két jól elhatárolható részből áll. A kliensoldali alrendszer feladata, hogy a felhasználóktól származó valós mozdulatokat összehasonlítsa az elvárt mozdulatokkal, vizsgálja azok pontosságát, és sikerességét.

A szerveroldalon futó alkalmazás biztosítja a kliensekkel való kommunikációt, valamint ennek a feladata a felhasználói adatok menedzsmentje, rehabilitációs utak tárolása valamint a teljes szinkron és aszinkron kollaborációs tér biztosítása.



2. ábra A rendszer működésének folyamata

Az alkalmazással két szerepkörben lehet interakcióba lépni. A gyakorlatokat definiáló, koordinátor és gyakorlatokat elvégző résztvevő. (Az alkalmazás széleskörű felhasználhatósága miatt nem célszerű a beteg, sérült, vagy az orvos, fejlesztőpedagógus, gyógytornász elnevezések használata.)

A koordinátor hozzáfér a résztvevők adataihoz, beteg és rehabilitációs utakhoz, a statisztikákhoz. Ezek ismeretében a résztvevők számára feladatokat, és gyakorlatokat definiálhat. A résztvevő hozzáfér a saját naptárához, ahol megtekintheti, hogy mikor milyen feladatokat kell elvégeznie. Az adott időpontban elindítja a megfelelő alkalmazást. Az alkalmazás a szerveren hozzáfér az elvégzendő feladatokhoz, mozdulatokhoz és ismétlés számokhoz, működés közben folyamatosan monitorozza a résztvevő eredményeit.

5. Mozdulatok reprezentációja

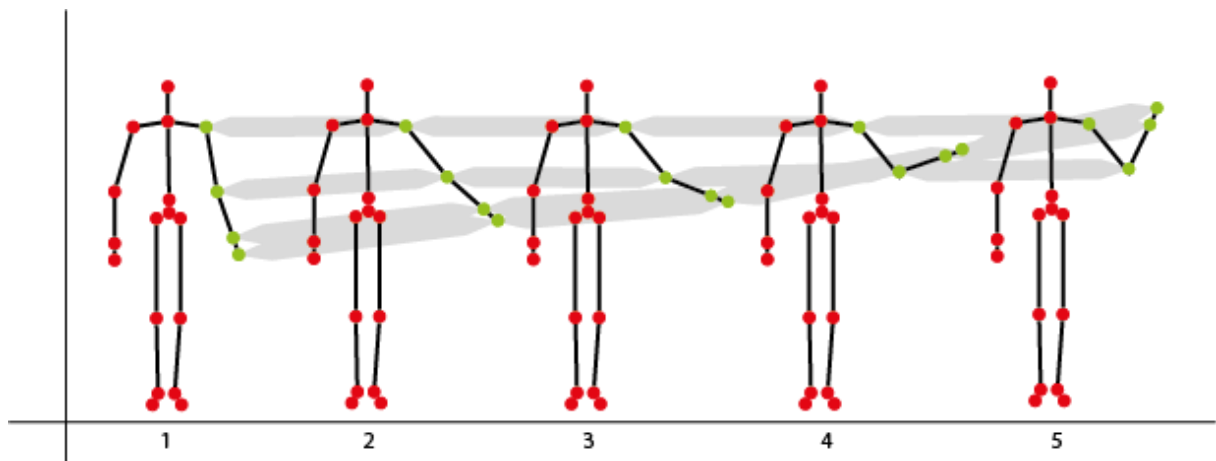
Az alkalmazás szempontjából kritikus kérdés a mozdulatok hatékony reprezentációja. Egy teljes mozdulat – biomechanikai megközelítésben – olyan összetett mozgásminta, melyet egy vagy több elemi mozgásminta alkot. Az elemi mozgásminta nem más, mint egy ízület adott irányban végzett mozgása. Az elemi mozgásmintában az egyes izmok térbeli és időbeli

aktiválása genetikus, tehát ha pontosan ismerjük a mozgást, akkor elég pontosan megadható az aktiválás időbeli és térbeli sorrendje. Az összetett mozgásmintában résztvevő elemi mozgásminták aktiválása már tanult, és meglehetősen egyedi. A mozdulatok „izomaktivitás” alapú reprezentációja meglehetősen nehézkes lenne, hiszen a Microsoft Kinect vezérlő az emberi csontváz mindösszesen 20 főbb ízületének digitális megfelelőjét tartalmazza. Egy-egy mozdulatsor kivitelezésében résztvevő izmok terhelése nagyban függ olyan ízületek állapotától, melyről a Microsoft Kinect nem szolgáltat információt. Vegyük például azt az esetet, amikor a kezünket a testünk mellé szorítjuk és behajlítjuk: ha a tenyerünk felfelé néz, akkor szinte csak a bicepsz terhelődik, de ha a tenyerünk lefele néz, akkor az alkar is komoly terhelést kap.

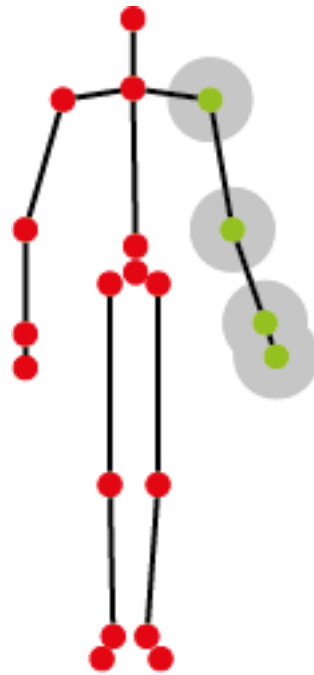
A Microsoft Kinect képességeit figyelembe véve sokkal célravezetőbb olyan reprezentációt választani ahol egy mozdulat a teljes test, vagy a test ízületeinek egy részhalmazának pillanatnyi állapotaiból összeállított sorozat, azaz minden mozdulat tetszőleges számú egymást követő póz rendezett sorrendje. Egy mozdulat esetében beszélhetünk vertikális szemcsézettségről és horizontális szemcsézettségről.

A mozdulat horizontális szemcsézettsége azt határozza meg, hogy egy mozdulat hány pózból, hány köztes állapotból tevődik össze. Egy mozdulat horizontális szemcsézettsége, lehet egészen kicsi, tartalmazhat egyetlen pózt, vagy csak egy kezdő és egy vég állapotot, de lehet egészen összetett, mely filmszerűen tartalmaz számos köztes állapotot. A kidolgozott koncepció szerint egy mozdulat elvégzése akkor sikeres, ha a mozdulat végrehajtása során minden póz felvétele sikeres volt. Egy mozdulat vertikális szemcsézettsége adja meg, hogy a lehetséges ízületekből hányról tartalmaz információt az adott mozdulat. A vertikális szemcsézettség bevezetésére azért volt szükség, mert egy-egy mozdulat esetében gyakran nem releváns minden ízület állapota. Ha egy gyakorlat felsőtest megmozgatását célozza meg, akkor a mozdulatok sikeressége szempontjából nem számít, hogy a lábak milyen pozícióban vannak.

A mozdulatok reprezentációja XML formátumban történik.



3. ábra A mozdulatok reprezentációja



4. ábra "Hibahatárok" a figyelt ízületek körül

6. Összegzés

Napjainkban az egészségügyi informatika a legtöbb országban a legnagyobb ütemben növekvő ágazat. Ezen törekvésekkel összhangban a telerehabilitáció is egyre nagyobb hangsúlyt kapott az elmúlt években, a fejlesztendő rendszer idomulva a nemzetközi elvárásokhoz egy olyan telerehabilitációs kollaborációs tér kialakítását célozza meg, melyben a rehabilitációt végző személyek és a rehabilitált személyek térben és időben függetlenül képesek a feladataik elvégzésre. A koordinátor képes ellenőrizni, hogy a résztvevő milyen gyakran, milyen intenzitással és milyen pontossággal végzi a kiadott feladatokat. A rendszer segítségével figyelhető az egyes területek terhelése, és annak relatív fáradása. A bármikor definiálható újabb és újabb feladatok, melyeket a gyakorlatokat végző személy a következő használatkor meg is kap, leegyszerűsítve ezzel egy igen magas erőforrás-igényű egészségügyi ellátást.

Természetesen az alkalmazás nem képes a rehabilitációs és fejlesztési folyamatok teljes kiváltására, hiszen egy ilyen rendszer csak bizonyos rendellenességek, betegségek és sérülések esetén használható. Tipikusan olyan feladatok esetében, ahol egyébként is csak felügyelőként, irányítóként van jelen a rehabilitációt végző személy. Az olyan gyakorlatok esetén nem használható, ahol a rehabilitációt végző személy aktív részvétele szükséges, ilyen például egy járás képtelen ember mozgatása.

7. Köszönetnyilvánítás

Az alkalmazás fejlesztése az **ÉAOP-1.1.3-12-2012-0052** kódszámú pályázat keretei között az Európai Unió támogatásával, az Európai Regionális Fejlesztési Alap társfinanszírozásával valósult meg.



ÚJ SZÉCHENYI TERV

Nemzeti Fejlesztési Ügynökség
www.ujsechenyiterv.gov.hu
06 40 638 638



A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

Irodalomjegyzék

- [1] Plato – *The Laws*
- [2] Millar, S. *The psychology of play*, Penguin Books LTD, Harmondsworth, Middlesex, England, 1968.
- [3] 1997. évi CLIV. törvény az egészségügyről 100. § (1) alapján
- [4] Ahn, L. V. *Human Computation*, Google Talks, 2006.
- [5] A. N. Krichevets†, E. B. Sirotkina, I. V. Yevsevicheva and L. M. Zeldin, *Computer games as a means of movement rehabilitation*, Disability and Rehabilitation, 1995, Vol. 17, No. 2, 100-105
- [6] Broeren J, Claesson L, Goude D, Rydmark M, and Sunnerhagen KS. *Virtual rehabilitation in an activity centre for community-dwelling persons with stroke. The possibilities of 3-dimensional computer games*. 2008, Cerebrovascular Diseases (Basel, Switzerland) Vol. 26, 289-296.
- [7] Subramanian S, Knaut LA, Beaudoin C, McFadyen BJ, Feldman AG, and Levin MF. *"Virtual reality environments for post-stroke arm rehabilitation."* Journal Of Neuroengineering And Rehabilitation Vol. 4 20-20
- [8] CAREN (Computer Assisted Rehabilitation Environment), www.motekmedical.com/products/caren/
- [9] *Mozgásérzékelős komoly játék a magyar idősök szellemi és fizikai rehabilitációjában*, „Az öregedő társadalom kihívásai” kutatás beszámolója, 2012.
- [10] *Stroke Patients at Royal Berkshire Benefit from Playing Kinect*, <http://www.xbox.com/en-US/Kinect/Kinect-Effect>
- [11] Yao-Jen Chang, Shu-Fang Chen, Jun-Da Huang, *A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities*, 2011, Research in Developmental Disabilities, Elsevier, 2566-2570

A PhysioBank fiziológiai jelfeldolgozó szoftvereinek hatékonysága

The efficiency of physiological signal processing software for PhysioBank

Kaczur Sándor

Gábor Dénes Főiskola Informatikai Intézet

kaczur@gdf.hu

Absztrakt: EKG analízis során az adatokat elemezzük, csoportosítjuk, jellemzőiket elkülönítjük, értelmezzük, következtetéseket vonunk le belőlük, diagnosztizálunk, ezzel is támogatva az orvos-szakértői tevékenységet. Ezt több, ingyenes, nyílt forráskódú szoftvergyűjtemény is támogatja. Az egyik ilyen ismert keretrendszer a PhysioBank, amely a <http://www.physionet.org> webhelyen található. A szerző kutatási témájához tartozik olyan algoritmusok, szoftverek, szoftvermodulok hatékonyságának elemzése, amelyek az EKG analízis során – előfeldolgozást követően, különböző tanuló és statisztikai módszerekkel – az EKG hullám olyan jellemzőinek kinyerését végzik, amelyek automatikus diagnosztizálást tesznek lehetővé. Ebben a cikkben a szerző a Physiologic signal processing kategóriába tartozó néhány szoftver hatékonyságát vizsgálja.

Kulcsszavak: algoritmusok hatékonysága, EKG analízis, PhysioBank

Abstract: In the analyzing of ECG analysis we have to diagnose, create groups, collect their characteristics, make an interpretation, thus supporting the activities of the expert medical data. This is supported by several free, open-source software collections as well. One of these is known as the PhysioBank framework, which is located in the <http://www.physionet.org> site. The author belongs to a research topic of algorithms, software, analysis of the effectiveness of software modules that are in the ECG analysis – after pre-processing and statistical learning different techniques – the ECG wave is carried out by extracting features that allow automatic diagnosis. In this article, the author examines the category of physiological signal processing for some of the software effectiveness.

Keywords: effectiveness of algorithms, ECG analysis, PhysioBank

1. Bevezetés

Kutatási témám az automatikus EKG analízis és annak orvos-szakértői értelmezését végző szakértői rendszerének fejlesztéséhez kapcsolódik. Az elektrokardiogram a szív elektromos aktivitását figyeli, veti papírra, tárolja elektronikus formában.

Az EKG készülék működési elve: a szív elektromos folyamatait (változásait) érzékel(het)jük a test felszínén, hiszen az emberi test egy jól vezető elektrolit oldat. Az EKG-görbe ezen apró elektromos változásokat ábrázolja felerősítve, az idő függvényében. A modern EKG-k 12 ponton mérnek, ezek közül 6 a mellkason található. A különféle pontok összehasonlításával (az egymáshoz viszonyított feszültségértékekből) nemcsak az egyes zavarok állapíthatók meg, hanem azok pontos elhelyezkedése is értékelhető.

Egy szabályos EKG-felvételen öt csúcspot lehet megkülönböztetni, ezek P, Q, R, S, T betűkkel vannak jelölve. Az egyes csúcsok megfelelnek bizonyos eseményeknek a szívben [1].

Az összehúzóadások számát az adott idő alatt bekövetkező R csúcsok száma adja (legtöbbször percenként 60 és 100 közötti). Az EKG-görbéken jellemezhető például az egyes hullámok (P, Q) hossza, előjele, szimmetriája, ezek távolsága (P-R, Q-T, R-R, S-T), a QRS-komplexum.

Az EKG elemzésével könnyen megállapítható, ha bizonyos részek túl szaporán húzódnak össze (tachycardia), esetleg túl ritkán jönnek ingerületbe (bradycardia), vagy kaotikusan viselkednek (fibrilláció). Abból, hogy mely hullámokkal van gond, megállapítható, hogy az ingerületképző és -vezető rendszernek melyik része működik rosszul. A különböző elvezetések (összesen 12) egybevetésével pontosan meghatározható a probléma helye, ami például az infarktusnál lehet nagyon fontos.

Az EKG analízis során a fenti adatokat elemezzük, csoportosítjuk, jellemzőiket elkülönítjük, értelmezzük, következtetéseket vonunk le belőlük, diagnosztizálunk, ezzel is támogatva az orvos-szakértői tevékenységet. Ennek során – előfeldolgozást követően, különböző tanuló és statisztikai módszerekkel – az EKG hullám olyan jellemzőinek kinyerése történik, amelyek automatikus diagnosztizálást tesznek lehetővé.

Olyan statisztikai módszereket alkalmazó algoritmusok megismerése, hatékonyságának elemzése, készítése a cél, amelyek alkalmazhatóak az EKG hullámok vizsgálatára [2].

2. A fiziológiai jelfeldolgozás alapjai

A biológiai/fiziológiai mérések célja objektív információszerzés, megismerés olyan módon, amely nem igényel – műtéti jellegű – beavatkozást. Módszerei közé tartozik az érzékelés, valamint az összehasonlítás referenciaadatokkal, alapegységekkel. Ennek eszköze valamilyen mérőeszköz, mérő-, illetve diagnosztizáló műszer. A mérés eredménye valamilyen mérőszám és mértékegység párral jól kifejezhető, ami alapján a diagnózis felállítható, elkészíthető a kezelési terv. Létezik analóg mérés és digitális mérés.

A biológiai/fiziológiai jelek mérése mindig élő emberen történik, a jelek/paraméterek mérés közben is folyamatosan változhatnak. A mérés – általában az orvosi szituációhoz kötődő körülmények miatt – nem jól reprodukálható. A mérés során több, egymástól független vagy egymással kapcsolatot mutató/feltételező paraméterek sokasága kerül feldolgozásra egyidejűleg. Néhány mérési módszer: közvetlen hatás az érzékelőkre, külső hatás modulációja, indikátoros módszer, külső ingerkeltés válasza, mintavételezés. Ezek a fenti sorrendben például: EKG, MRI, PET, EEG, vérvétel. A mérés pontossága esetén a következő alapfogalmakkal kell tisztában lennünk: valódi érték, mért érték, abszolút hiba, relatív hiba, műszer pontossága, műszer megbízhatósága, mérési módszer helyessége, a mérőeszköz jelzéseinek korlátai. Illik érteni a biológiai/fiziológiai jelek mérésének hatásvázlatát, az ezt alkotó komponensek mélyebb összefüggéseit is [3].

A PhysioBank keretrendszer elemzendő minta adatai/adatbázisai tartalmaznak többféle helyes, majdnem helyes, illetve hibás mérésből származó adatsort is, hogy minél teljesebb körűen tudják támogatni a feldolgozó algoritmusok analízisét, magát a jelfeldolgozást, az idősor-analízist, és a megerősítéses tanulás különböző módszereit alkalmazó lehetőségeket.

3. Az algoritmusok lehetséges összehasonlítási szempontjai

Algoritmusok hatékonyságának elemzéséhez, összehasonlításukhoz a cél(ok) egyértelmű meghatározása után követelményrendszer állítható fel. A fejlesztőeszköztől, programozási nyelvtől, implementációtól függő szempontokra nem térek ki, valamint a grafikus felhasználói felületen történő tevékenységek sem kerülnek említésre. Az alábbi szempontok adódnak: végrehajtási idő, helyfoglalás, bonyolultság [4, 5, 6].

A végrehajtási idő mérhető a végrehajtott utasítások számával, függ a hardvertől. Nem feltétlenül függ a bemenő adatok számától. Lehet minimális, átlagos és maximális.

A helyfoglalás a változók, adatszerkezetek tárigényével mérhető, függ az adat- és programábrázolástól, így nyelv- és implementációfüggő is. Kifejezhető a programkód méretével, illetve a memóriában, háttértáron elfoglalt helyigénnyel is. A helyfoglalás csökkentése általában növeli a végrehajtási időt.

A bonyolultság lehet globális és lokális. Globális, ha az algoritmus egészének megértése szükséges a csökkentéséhez. Lokális, ha kódoptimalizálással (legalább részben) megoldható. Szintén függ az adat- és programábrázolástól, így nyelv- és implementációfüggő. Objektív módon kevésbé definiálható. Csoportosítható logikai és szerkezeti bonyolultságként. Matematikai alapjait [7, 8] tartalmazza.

4. A PhysioBank fiziológiai jelfeldolgozó szoftverei

A PhysioBank keretrendszer QRS komplexumhoz kapcsolódó szoftvermoduljai közül korábban elemeztem az adatsorok állományformátumait és az adatmegjelenítő szoftvereinek hatékonyságát [9]. Most a PhysioBank keretrendszer fiziológiai jelfeldolgozó szoftverei kerülnek sorra [10], amelyek több szoftvercsomagba integrálva összesen 10 db különálló szoftverből/alkalmazásból állnak.

Az sqrs/sqrs125 szoftver egy meredekségen alapuló egycsatornás QRS detektor, azaz a QRS komplexum jellemzőinek meghatározására alkalmas. Felnőtt ember EKG-jához konfigurált. Több, ciklikusan kifejlesztett, jobb változat is rendelkezésre áll belőle. C nyelvű implementációja több fordítási direktíva alkalmazásával is optimalizált. A WFDB szoftvercsomag szolgáltatásait igénybe veszi. Tartalmaz mutatókezelést, dinamikus memóriefoglalást, és ehhez kötődő hibakezelést is. Az algoritmus néhány egyszerű metódusból áll, amelyek egymástól független iterációkat, azon belől pedig szelekciókat tartalmaz.

A wqrs szoftver egy hosszúsági transzformáción alapuló egycsatornás QRS detektor.

A wabp szoftver egy artériás vérnyomás (ABP) hullámformáján alapuló impulzus detektor. Felnőtt ember ABP-jéhez optimalizált.

Az apdet szoftver egy Hilbert transzformáción alapuló alvási apnoé detektor. Több lépésből áll, például: kiugró adatokat szűr, lineárisan skáláz, amplitúdót és frekvenciát elkülönít, medián alapú kiválogatást végez, amplitúdót normalizál, felező amplitúdót meghatározó szórást és küszöböt is figyelembe véve. A tesztadatok két csoportján elkülönítve 82,1 illetve 84,5%-ban detektál helyesen.

Az ecgpuwave kétkomponensű szoftver tartalmaz egy QRS detektort és egy EKG limit lokátort. A QRS komplexum P és ST-T hullámformáinak keresi az elejét, a csúcsát és a végét.

A pr-comp szoftver a hullámformák főkomponenseinek észlelésére használható, minimális mértékben befolyásolja a zaj. 4 modulból áll, korábban részletesen elemzésre került [11].

A CO estimators algoritmusai artériás vérnyomás hullámformáit elemezve adnak becslést perctérfogatra, ami a szív teljesítményét fejezi ki az időegység alatt átpumpált vér mennyiségével. A gyűjtemény 5 egymáshoz kapcsolódó modulból áll, csupán az analizáló modult vizsgáltam.

Az EDR szoftver egy vagy több EKG hullám együttes vizsgálatával is képes kinyerni a légzési szignál jellemzőit. Működése azon alapul, hogy az EKG elektródák pozíciója változik a szívhez képest be- és kilégzés során.

A puka szoftver detektálja az elnyúló be- és kilégzés, köhögés, illetve EKG jellemzőit. Különálló, önállóan használható szoftvercsomagként is elérhető. Igen népszerű, különösen a szakterülettel még csak most ismerkedők számára. Funkcionálisan jól használható grafikus

felhasználói felülettel rendelkezik. A gyűjtemény több egymáshoz kapcsolódó modulból áll, csupán a respiration modult vizsgáltam.

A TWAnalyser szoftver az EKG T hullámjellemezőit képes észlelni.

5. A PhysioBank fiziológiai jelfeldolgozó szoftvereinek hatékonysága

Aszimptotikus elemzést és esetvizsgálatot is végeztem [8], következésképpen az átlagos eseteket vizsgáltam. Ügyeltem a legmagasabb fokú tagok együtthatójának pontos kifejezésére. Csak N függ a feldolgozott adatmennyiségtől, k elenyészően kicsi. Az elemi lépések eltérő tényleges végrehajtási idejét nem tekintettem lényegesnek. A több modulból álló szoftverek esetén csupán az analízáló, jelfeldolgozó modul hatékonyságát határoztam meg. Vizsgáltam a strukturális és elvi bonyolultságot is, amelyeket négyféle módon fejeztem ki: hány különböző programozási tétel fedezhető fel a forráskódban, vannak-e külső metódushívások, belső metódushívások, beágyazott metódusok, milyen mélységben vannak egymásba ágyazva a ciklusok, strukturális van objektumorientált a szoftver felépítése.

A fenti szoftverek elemzésének összegzését az 1. ábra tartalmazza. Az apdet szoftver forráskódja nem publikus, így nem szerepel az ábrán.

	Idő	Hely	Bonyolultság
sqrs/sqrs125	$4N+k$	$2N+k$	3 db programozási tétel metódusok
wqrs	N^3+6N+k	$4N+k$	4 db programozási tétel metódusok beágyazott ciklusok (3 szint)
wabp	N^3+7N+k	$6N+k$	4 db programozási tétel metódusok beágyazott ciklusok (3 szint)
ecgpuwave	N^3+N^2+3N+k	$N^2+14N+k$	5 db programozási tétel külső metódusok beágyazott ciklusok (3 szint)
pr-comp	$4N^3+10N^2+8N+k$	$7N^2+8N+k$	9 db programozási tétel beágyazott metódusok beágyazott ciklusok (3 szint)
CO estimators (analyze)	$3N+k$	$4N+k$	3 db programozási tétel metódusok
EDR	N^2+7N	$4N+k$	4 db programozási tétel metódusok beágyazott ciklusok (2 szint)
puka (respiration)	$4N+k$	$2N^2+3N+k$	3 db programozási tétel metódusok ciklusok, objektumok
TWAnalyser	N^4+8N^2+6N+k	$6N^2+16N+k$	10 db programozási tétel külső metódusok beágyazott metódusok beágyazott ciklusok (4 szint)

1. ábra. A PhysioBank fiziológiai jelfeldolgozó szoftvereinek hatékonysága

Az elemzés során természetesen nem számítanak az alábbiak: telepítési idő, konfigurálási idő, mintaadatok előkészítése és konvertálása, az eredmények elemzése, az eredmények grafikus megjelenítő képessége, az archiválás képessége szabványos formátumokban, a többféle szoftververzió, az eltérő platformokból illetve programozási nyelvekből adódó jelentős különbségek, a szükséges memory dump eszközök és debuggerek közötti eltérések. Szintén nem lényeges, hogy elérhető-e a szoftverekhez dokumentáció, az analizáló, jelfeldolgozó rész elméleti háttérét tartalmazó publikáció vagy csupán különféle programozási nyelveken (C++, Java, Fortran, Basic) áll rendelkezésre forráskód.

Néhány elemzett szoftver épít más szoftvercsomagokra, teljes utasításkészlet átvételével, adatdefiníciós rész meghivatkozásával, konvertáló és I/O műveletek importálásával. Ezek a keresztivatkozások lényegesen csökkentik a strukturális redundanciát, de lényegesen növelik a tervezéshez és megértéshez szükséges absztrakciós szintet és egy-egy algoritmus bonyolultságát.

6. Javaslatok az elemzett szoftverek hatékonyságának növelésére

Az elemzett szoftverek mintegy 20 éves fejlődési szakaszon vannak túl. Általában izolált, szegregált, egyedi fejlesztések eredményei, egymással nem teljesen kompatibilisek és néhol gyökeresen más céllal készültek.

Összesen nyolcféle input értelmezésére és hatféle output előállítására képesek. Ha a mintaadatok esetleg más forrásból származnak, akkor szabadon adott a lehetőség a kompatibilissé konvertálásra. Ki lehetne dolgozni egy lényegesen kevesebb elemszámból álló formátumhalmazt, amelyeket az egészségügyi szakterületben elérhető szabványos mérőeszközök mindegyike tudna támogatni, célszerűen ez XML vagy JSON formátum lenne. Ahol a mérési adatok valós típusú adatformátumban tároltak az egésze skálázott, arányított, normalizált helyett, ott a konvertáláskor a számábrázolási pontatlanság és a műveletvégzés adattípusból következő hibái is jelentősek tudnak lenni. Ezt is célszerű lenne elkerülni.

A végrehajtási idő csökkentésére ismert módszerek [4] közül szinte bármelyik bevethető. Az elemzett szoftverek esetén nagyon könnyen megvalósítható a ciklusok végrehajtási számának és idejének csökkentése. Előbbire javaslatom: jelsorozat részekre osztása, jelsorozat elemeinek csoportos feldolgozása, ciklustranzformálás. Utóbbira javaslatom: elágazás tranzformálása, kivételes esete kiküszöbölése, adatok előfeldolgozása. Nehezebben megy, de koncepcionális átgondolást követően alkalmazható a jelsorozatok párhuzamos feldolgozása iteratív módon. Nem javaslom a jelsorozatok rekurzív feldolgozását, mert teljes újratervezést igényelne.

A helyfoglalás csökkentésére ismert módszerek [4] közül is számos bevethető. Az elemzett szoftverek esetén nagyon könnyen alkalmazható az adatok mennyiségének csökkentése és a programkód méretének csökkentése. Előbbire javaslom speciális szerkezetű sorozatok definiálását, adatterületek megosztását, valamint adatelemek – ésszerű – kódolását a belső ábrázolás során. Utóbbira javaslom a ciklusok összevonását és az azonos/hasonló funkciók közös eljárásba foglalását.

A bonyolultság csökkentésére is adódik néhány lehetőség az ismert módszerek [4] közül. Az elemzett szoftverek esetén redukálható az algoritmus bonyolultsága és az adatszerkezet bonyolultsága. Mindkettőre javaslom az adatabsztrakció különböző lehetőségeit.

Az elemzett szoftverek között mindössze egyetlen egyről állítható, hogy teljes objektumorientált szemléletmóddal tervezték és implementálták. A többi strukturált szemléletmóddal megvalósított program forráskódjaiban található sok-sok párhuzamosan deklarált adatszerkezet, direkt memóriakezelés, mutatók alkalmazása, egyedi trükkök a számábrázolásra, fordítási direktívák, szintaktikus „cukorkák”, implementáció és verziófüggő

elemek is. Mindezek sokkal egyszerűbben és átlátható módon is megvalósíthatók lennének újratervezett, következetes objektumorientált szemléletmód alkalmazásával.

Az elemzett szoftverek többségéről állítható, hogy lényegesen egyszerűbb lenne – természetesen az alkalmazott feldolgozási módszerek teljes körű ismeretében – újra implementálni, mivel az előző javaslatokat figyelembe véve és alkalmazva alig kimutatható javulást érhetünk csak el a hatékonyság egyes területein.

Irodalomjegyzék

- [1] EKG-felvétel, http://www.egeszsegugy.info/friss_hirek/tobbfunkcios-internet-alapu-telemedicinas-kardiologiai-merorendszer-, 2014.06.30.
- [2] Kaczur S.: Részletes kutatási terv, ELTE Informatikai Doktori Iskola, 2011
- [3] Biofizikai és orvostechnika alapjai – Biológiai jelek mérése, http://sotepedia.hu/_media/etk/targyak/a03_meres.pdf, 2014.06.30.
- [4] Zsakó L.: Módszeres programozás: Hatékonyság, Mikrológia 6., ELTE, 1999
- [5] Iványi A.: Informatikai algoritmusok I., II. ELTE, 2004., 2005., ISBN 9634637752
- [6] Kaczur S.: A hozzárendelési feladat két megoldási módszerének összehasonlító elemzése, Informatika, A GDF Közleményei, Budapest, XI. évfolyam 2. szám, 2009, ISSN 1419-2527, p. 24-27
- [7] Szlávi P., Zsakó L., Temesvári T.: Módszeres programozás: A programkészítés technológiája, Mikrológia 21., ELTE, 2000
- [8] Mágoriné H. Á.: Algoritmusok és adatszerkezetek, JGYF Kiadó, Szeged, 2000, ISBN 963 9167 36 3
- [9] Kaczur, S.: Néhány nyílt forráskódú EKG analízáló algoritmus hatékonysága, A Tudomány Hete a Dunaújvárosi Főiskolán, Dunaújváros, Dunaújvárosi Főiskola, 2011. november 7-11.
- [10] PhysioToolkit, <http://physionet.org/physiotools/software-index.shtml>, 2014.06.30.
- [11] S. Kaczur: The efficiency of data visualization software for Physiobank, Informatika, A GDF Közleményei, Budapest, XV. évfolyam 1. szám, 2013, ISSN 1419-2527, p. 17-19

Egészségünkkel kapcsolatos információk a saját kezünkben

Health information in our own hands

Dr. Maros-Szabó Zsuzsanna^a, Kovács Zsigmond^b

^aLarkbio Kft.

zsuzsa.szabo@larkbio.com

^bLarkbio Kft.

ziggy.kovacs@larkbio.com

Absztrakt: Az egészségügy egyike azon kevés területek egyikének, ahol az informatika fejlődése által nyújtott hatalmas lehetőségek még csak nagyon kis mértékben kerültek kihasználásra. Egy informatikával jól támogatott egészségügynek pedig nem csak az egyének egészsége szempontjából lenne pozitív hatása, hanem az egyre nehezebben elviselhető makroszintű egészségügyi költségek menedzseléséhez is elengedhetlenül szükséges. Cikkünkben az egészségügy és informatika kapcsolódásának egy fontos elemét vesszük vizsgálat alá. Az Elektronikus Kórlap (vagy E-Kórlap – angolul: Electronic Health Records) egy meghatározott egyén vagy populáció egészségügyi adatainak gyűjtésére, tárolására, nyomkövetésére és megosztására szolgáló informatikai alkalmazás. A Személyes Kórlap (angolul: Personal Health Records) koncepciója nagyon hasonlít az E-Kórlap-hoz, a lényegi különbség abban van, hogy míg az E-Kórlap-ot általában valamely egészségügyi intézmény kezeli, addig a Személyes Kórlap esetében az adatokért maga az érintett személy a felelős. A Személyes Kórlapokkal szemben megfogalmazott legfontosabb követelmények a következők. A rendszer legyen: (1) átfogó; (2) interaktív; (3) az egyén által kontrollált és (4) biztonságos. Bár a Személyes Kórlap ötlete nem új, széles körű elterjedéséről még a legfejlettebb országokban sem beszélhetünk. Nem született még meg az az üzleti modell, amit sikeresnek nevezhetünk. Alapvetően három megközelítés lehetséges: (1) biztosító által üzemeltetett rendszer; (2) egészségügyi intézmény által üzemeltetett rendszer; (3) független rendszer. Bár nagyon sokan láttak fantáziát a Személyes Kórlap elterjedésében, a mai napig ezek a remények nem váltak valóra. Még mindig sokan hisznek a Személyes Kórlap ígéretében, az nyilvánvaló, hogy az igazi áttöréshez egy felhasználóbarátabb és sokkal szélesebb körben elfogadott rendszer megjelenésére van szükség. Az Európai Unió 7. keretprogramján belüli támogatással megvalósuló MyHealthAvatar projekt egy olyan újfajta Személyes Kórlap modellt kíván létrehozni, amely a remények szerint lehetővé teszi a koncepció széles körű elterjedését. A korábban leírtakon túl a projekt az alábbi újdonságokkal kívánja felruházni az avatárt: (1) A klinikai információkon túl az avatár életmóddal és környezettel kapcsolatos információkat is tartalmaz. (2) Egyedi alkalmazások automatizálják az adatok feltöltését. (3) Segédprogramok támogatják a tárolt információk elemzését és vizuális megjelenítését.

Kulcsszavak: Elektronikus Kórlap, Személyes Kórlap, aktív beteg

Abstract: Healthcare is one of those few areas where the advancements provided by Information Technology have been used scarcely. However, healthcare supported by IT would not only benefit the health of the individual, but it is also indispensable in managing the increasing healthcare expenses at a macro level. In this article we study one of the elements of the connection between the IT and healthcare. The Electronic Health Records is an IT application for collecting, storing, following-up and sharing healthcare data of an individual or a certain population. The concept of Personal Health Records is very similar to the Electronic Health Records; the significant difference is that the Electronic Health Records are managed by a healthcare institute, while the individual is in charge of the Personal Health Records. The four main requirements of the Personal Health Records: (1) comprehensive, (2) interactive, (3) patient-controlled and (4) secure. Although the idea of PHR is not new, it is not widely used in the most developed countries either. There is no business model we may call successful. Basically, there are three approaches: (1) the system is provided by insurance companies, (2) the system is provided by healthcare institutes and (3) the system is provided independently. Despite the

expectations were high for the Personal Health Records, they still have not taken effect. Although there are hopes for the Personal Health Records, it is obvious that to reach a breakthrough the system has to be widely accepted and made user friendly. The MyHealthAvatar, being completed with grants by the European Union 7th Framework Program, is establishing a PHR model that makes possible for the concept to spread widely. Besides the features mentioned earlier, the novelties of the project are: (1) Beyond clinical information, the avatar contains information about the lifestyle and environment; (2) Unique applications automate uploading data. (3) Utilities support the analysis and visualization of the stored information.

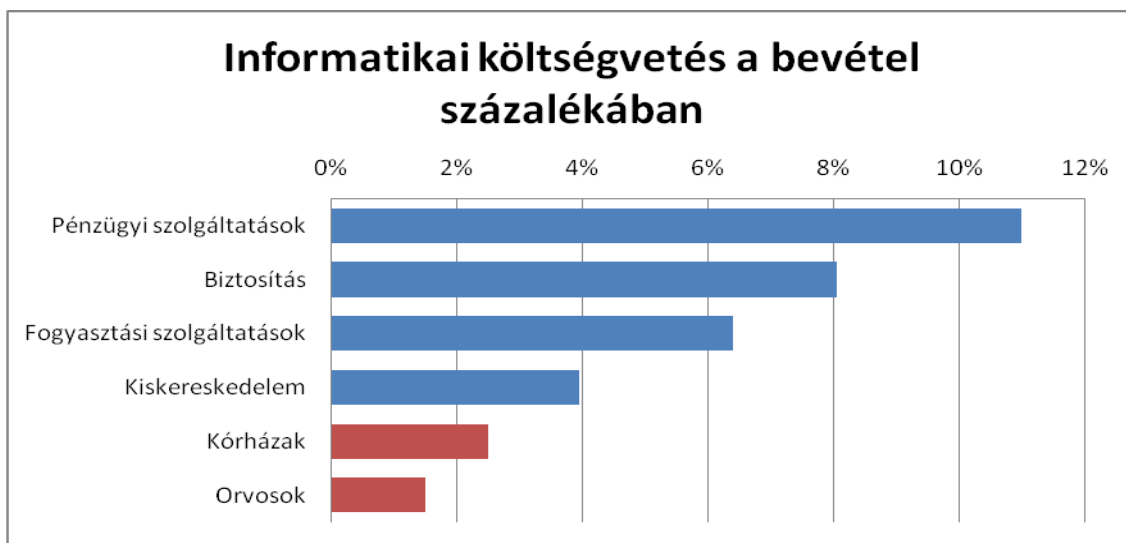
Keywords: Electronic Health Records (EHR), Personal Health Records (PHR), patient empowerment

1. Bevezetés

Képzeln el egy alkalmazást a mobiltelefonján, amely tartalmazza az összes Önnel kapcsolatos egészségügyi információt és képes azt megosztani más szoftverekkel. Egy nemrég elvégzett kisebb kórházi műtét után először látogatja meg háziorvosát. A látogatás pontos időpontját egy internetes felületen egyeztetni utazás közben. Az időpont egyeztetése közben az alkalmazás nem csak a kórházi zárójelentést küldi el automatikusan háziorvosának, hanem a gyógytornász által javasolt – a műtét során érintett izmok rehabilitálására szolgáló – gyakorlatok leírását is. A háziorvos a látogatás során felír egy fájdalomcsillapítót, az elektronikus receptet azonnal elküldi az Ön által megnevezett patikába, a gyógyszer szedésével kapcsolatos tudnivalók pedig bekerülnek a mobiltelefonján futó alkalmazásba. Másnap reggel emlékeztető üzenet figyelmezteti a gyógyszer bevételére és egy internetes portálon elolvashatja mások valós tapasztalatait a gyógyszer mellékhatásairól. Fikció? Ma talán még igen.

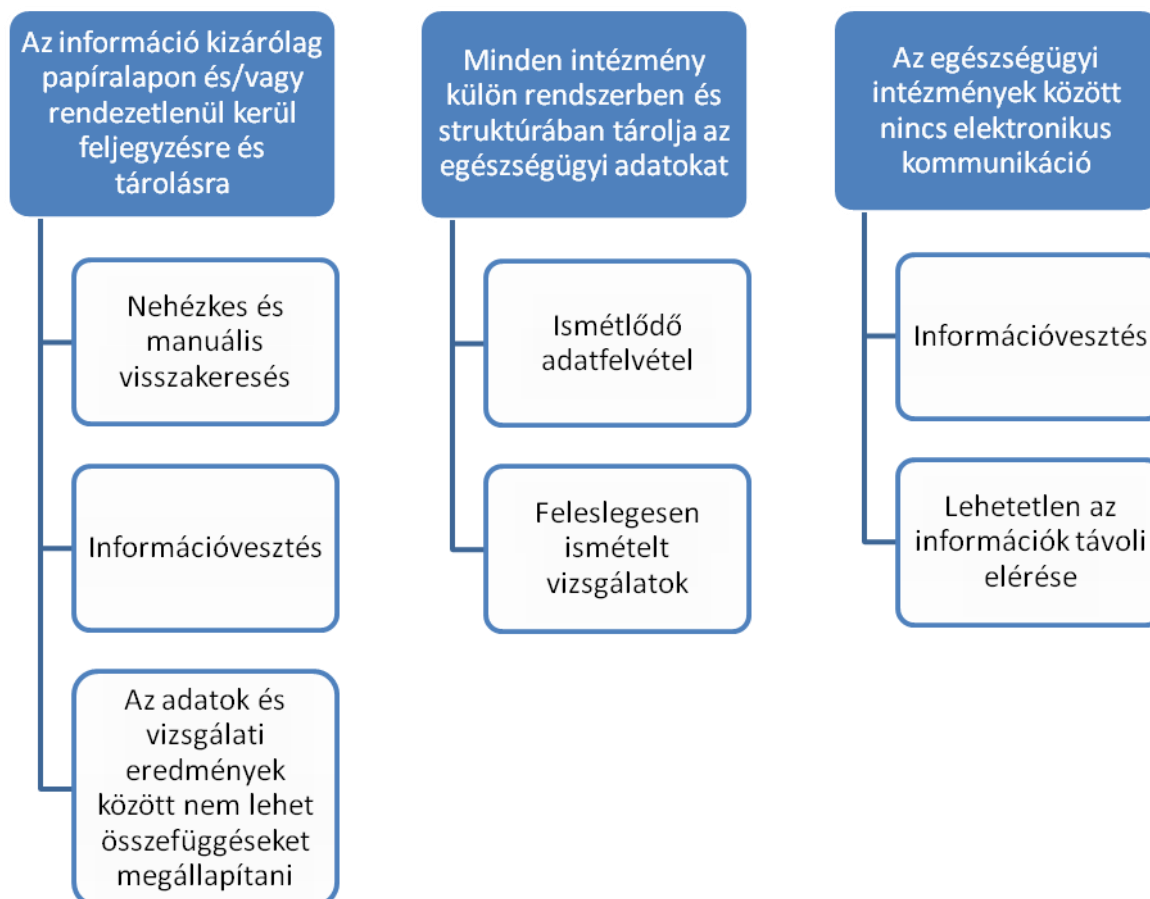
2. Az informatika szerepe az egészségügyben

Az információs és kommunikációs forradalom átalakította mindennapjainkat. Másképp dolgozunk, intézzük ügyeinket és tartjuk a kapcsolatot ismerőseinkkel, mint akár 5-10 évvel ezelőtt. Van azonban egy nagyon fontos terület, ahol a lehetőségekhez képest igen komoly az elmaradás az informatika kínálta lehetőségek és a mindennapi valóság között. Ez a terület az egészségügy. Az elmaradást jól szemléltetik az alábbi adatok.



1. ábra. Informatikai költségvetés a bevétel százalékában néhány jelentősebb iparágban (Forrás: Lewin Group, Forrester Research) [1]

Cikkünk nem vizsgálja azokat az okokat, amelyek ezek az elmaradáshoz vezetnek, sokkal inkább a hatásokra szeretnénk koncentrálni, azokat is az egyének/betegek szempontjából tekintve. Az alábbi ábra szemlélteti az alapproblémákat és azok legfontosabb következményeit.



2. ábra. Az informatika hiánya az egészségügyben

Tulajdonképpen miért vesszük mindezeket természetesnek az információs társadalomban? A komplex egészségügyi rendszerek létrehozásának és működtetésének valójában már évek óta nincs technológiai akadálya [2]. Igazából már most is léteznek azok az eszközök, amelyekkel sokszorosára lehetne növelni az egészségügy hatékonyságát.

3. Elektronikus Kórlap és Személyes Kórlap

Az Elektronikus Kórlap (vagy E-Kórlap – angolul: Electronic Health Records) egy meghatározott egyén vagy populáció egészségügyi adatainak gyűjtésére, tárolására, nyomonkövetésére és megosztására szolgáló informatikai alkalmazás [3]. Az E-Kórlap kialakításától és a felhasználás módjától függően alapvetően az alábbi információkat tartalmazhatja:

- azonosító adatok;
- szedett gyógyszerek;
- korábbi betegségek/műtétek/kezelések;
- kezelési feljegyzések és utasítások;
- kivizsgálások eredményei;
- oltások jegyzéke.

A benne található adatok elméletileg (bizonyos szabályokat betartva) megosztásra kerülhetnek az egészségügyben érintett szervezetek és/vagy személyek között – orvosok, nővérek, gondozók, betegek, rendelők, kórházak, biztosítók stb.

A Személyes Kórlap (angolul: Personal Health Records) koncepciója nagyon hasonlít az E-Kórlap-hoz, a lényegi különbség abban van, hogy míg az E-Kórlap-ot általában valamely egészségügyi intézmény kezeli, addig a Személyes Kórlap esetében az adatokért maga az érintett személy a felelős. A Személyes Kórlapban található adatok segítségével jobban figyelemmel lehet követni az adott személy egészségi állapotát és hatékonyabban lehet koordinálni a kezeléseket. A Személyes Kórlapokkal szemben megfogalmazott legfontosabb követelményeket a 2. ábra mutatja be.



3. ábra. A Személyes Kórlapokkal szemben megfogalmazott legfontosabb követelmények [4]

4. Személyes Kórlapok a gyakorlatban

Bár a Személyes Kórlap ötlete nem új, széles körű elterjedéséről még a legfejlettebb országokban sem beszélhetünk. Nem született még meg az az üzleti modell, amit sikeresnek nevezhetünk. Alapvetően három megközelítés lehetséges:

1. Biztosító által üzemeltetett rendszer. Erre elsősorban az Egyesült Államokban vannak példák, ahol az egészségügyi magánbiztosítók szerepe jóval nagyobb az Európában (különösen pedig Magyarországon) megszokottnál. Ezek a rendszerek általában csak azokat az adatokat tartalmazzák, amelyek a biztosító rendelkezésére állnak, némelyikük lehetővé teszi az egyéneknek a saját maguk által generált vagy összegyűjtött adatok felvitelét. Ebből következően az átfogóság követelményének szinte egyáltalán nem, az interaktivitás követelményének is csak kis részben felelnek meg.
2. Egészségügyi intézmény által üzemeltetett rendszer. Ez lehet egy kórház, magánrendelő vagy más egészségügyi szolgáltatást végző szervezet. A betegek számára általában elérhetőek a következő típusú adatok: gyógyszerek, védőoltások, vizsgálatok eredményei. Közülük több lehetőséget biztosít a beteg-orvos kommunikációra a rendszeren belül. Ebben a kategóriában is meglehetősen korlátozottan érvényesül az átfogóság elve, de az egyéni kontroll, az interaktivitás és sok esetben a biztonság igénye sem kerül kielégítésre az elvárható szinten.
3. Független rendszer. Ezeknél az egyén kezében van a kontroll, hátránya viszont, hogy sok esetben az információkat is az egyénnek kell megtalálni, feltölteni és menedzselni. Ezzel a megközelítéssel sajnos az a legnagyobb probléma, hogy az emberek többsége ezekkel a feltételekkel egyszerűen nem használja a rendszert [5]. Egy krónikus beteg könnyebben rávehető, hogy rendszeresen gyűjtse és figyelemmel kísérje az egészségével kapcsolatos információkat, erre egy alapvetően egészséges ember általában nem veszi a fáradságot.

Bár nagyon sokan láttak fantáziát a Személyes Kórlap elterjedésében, a mai napig ezek a remények nem váltak valóra. Az egyik leglátványosabb kudarcot a Google Health produkálta, amely nagy csinnadrattával elindított szolgáltatást alig néhány évvel később csöndben bezárta. Hiába állt a rendszer mögött a világ egyik legnagyobb informatikai óriása, a független rendszereket sújtó (korábban említett) akadályokat képtelen volt legyőzni [6]. Jelenleg a legelterjedtebb független rendszer a Microsoft Health Vault szolgáltatása. A szakértők szerint itt sem beszélhetünk tömeges aktív használatról, inkább csak úgy tűnik, hogy a Microsoft némileg türelmesebb. Bár még mindig sokan hisznek a Személyes Kórlap ígéréiben, az nyilvánvaló, hogy az igazi áttöréshez egy felhasználóbarátabb és sokkal szélesebb körben elfogadott rendszer megjelenésére van szükség.

5. MyHealthAvatar

Az Európai Unió 7. keretprogramján belüli támogatással megvalósuló MyHealthAvatar projekt egy olyan újfajta Személyes Kórlap modellt kíván létrehozni, amely a remények szerint lehetővé teszi a koncepció széles körű elterjedését. A modell középpontjában az ún. avatar áll, egy mindenkit egész életén át végigkísérő dinamikus adatbázis, amely alkalmas az

egészségügyi adatok tárolására és kinyerésére. A korábban leírtakon túl a projekt az alábbi újdonságokkal kívánja felruházni az avatárt:

- Az alapvetően a betegségekhez köthető klinikai információkon túl az avatár tartalmazni fogja az életmóddal (pl. étkezési és sportolási szokások) és környezettel (pl. lakó- és munkahelyi környezet) kapcsolatos információkat is, teljesebb képet adva az egyének egészségére hatással levő tényezőkről.
- Olyan szoftverek kerülnek alkalmazásra, amelyek megkönnyítik – vagy akár automatizálják – a szükséges adatok feltöltését az avatárba. Az avatárt képessé lehet tenni például a közösségi oldalakhoz történő intelligens kapcsolódásra – az ott fellelhető információk strukturált beépítése érdekében.
- Segédprogramok fogják támogatni az avatárban tárolt információk elemzését és vizuális megjelenítését is. Ez nemcsak az egyén, de az orvos számára is átláthatóbbá teszi az adattömeget és lehetővé teszi a gyorsabb és pontosabb döntések meghozatalát.

A Személyes Kórlapok és így az avatár esetében is kulcsfontosságú tényező az egyének bevonása az egészségügyi folyamatokba és döntésekbe. Hagyományosan a betegek az egészségügy passzív résztvevői voltak, akik kezelést kaptak és utasításokat kaptak. Az utóbbi években ez a magatartás (különösen a fiatalabb generációk körében) átalakulóban van. Az egyének jobban informáltak és ezzel együtt egyre aktívabb és tudatosabb szerepet játszanak saját egészségükkel kapcsolatosan [7]. Ennek a változásnak már az előidézésében is kiemelt szerepe van az informatika fejlődésének, az internet által nyújtott kommunikációs lehetőségek robbanásszerű változásának. Magyarország legnagyobb egészségügyi portáljának több mint 100 ezer egyéni látogatója van naponta, a felhasználók ráadásul nem csak elolvassák az őket érdeklő anyagokat, de rendszeresen kérdeznek az orvosoktól és egymással is megosztják kérdéseiket/tapasztalataikat. A Személyes Kórlap ebben az értelemben csak egy természetes kiterjesztése az egyén aktív részvételének, a lehetőségek számának és minőségének növekedése – ha a betegek élnek vele – már rövidebb távon is az életminőség valódi javulásához vezethet.

Az avatár legfontosabb tulajdonsága a sokrétűsége. Arra az igényre kíván elsősorban reagálni, amelyben az eddigi kezdeményezések kudarcot vallottak: a szereplőkön és rendszereken átívelő átfogó megközelítés igényére. Előtérbe helyezi a különböző szereplők érdekeltiségének megteremtését, anélkül ugyanis elképzelhetetlen a tömeges elterjedés.

Nagyon fontos ugyanakkor, hogy megteremtsük ennek a sokrétűségnek az alapjait. Ennek érdekében a MyHealthAvatar projekt során minden részletre kiterjedően kerül megtervezésre és kialakításra a rendszer. Ez a következő nagy tevékenységeket foglalja magában:

1. Felhasználói igények felmérése
2. Architektúra megtervezése és kialakítása
3. Szemantikus interoperabilitás biztosítása
4. Modellkészítés
5. Részletes esettanulmányok megalkotása
6. Jogi keretek meghatározása (az adatbiztonság és a titkosság biztosítása)

A három éves projekt jelenleg féldőnél tart és bár a kezdeti eredmények biztatóak, a végső eredmény jövőbeni elfogadottsága a feladat komplexitása miatt nehezen megjósolható. A projekt eredményeit a projekt hivatalos honlapján (<http://www.myhealthavatar.eu/>) és

Facebook oldalán (<https://www.facebook.com/feng.myhealthavatar>) is figyelemmel kísérhetik az érdeklődők.

6. Köszönetnyilvánítás

Cikkünk megírásában és szerkesztésében sok segítséget kaptunk kollegáinktól, azaz a Larkbio csapatától. A cikk nem jöhetett volna létre a MyHealthAvatar konzorcium tagjai (http://www.myhealthavatar.eu/?page_id=27) által az elmúlt másfél évben végzett munka nélkül.

Irodalomjegyzék

- [1] <http://www.nyhealthcarecommission.org/docs/hit.pdf>
- [2] Kathrin M Cresswell, David W Bates, Aziz Sheikh: Ten key considerations for the successful implementation and adoption of large-scale health information technology, *Journal of the American Medical Informatics Association* (2013)
- [3] <http://www.healthit.gov/providers-professionals/faqs/what-electronic-health-record-ehr>
- [4] http://assets1.csc.com/health_services/downloads/CSC_A_True_Personal_Health_Record.pdf
- [5] <http://www.informationweek.com/healthcare/patient-tools/why-personal-health-records-have-flopped/d/d-id/1102247?>
- [6] http://www.pulseitmagazine.com.au/index.php?option=com_content&view=article&id=954:feature-why-did-google-health-fail&catid=16:australian-ehealth&Itemid=328
- [7] Robert M. Anderson, Martha M. Funnell: Patient empowerment: Myths and misconceptions, *Patient Education and Counseling*, Volume 79, Issue 3, Pages 275-384 (2010)

Vizsgáztatás E-learning rendszer segítségével

Interrogation with help the E-learning system

Ambrusné Dr. Somogyi Kornélia

Óbudai Egyetem, RKK, Médiatechnológiai és Könnyűipari Intézet

a_somogyi.kornelia@rkk.uni-obuda.hu

Absztrakt: Egyetemünkön már több mint 5 éve használjuk a Moodle e-learning rendszert. Kezdetben csak a tananyagok elektronikus közzétételére, s a házi feladatok beadására alkalmaztuk az e-learninget. Karunkon már akkor is sok oktató használta a rendszert. 2013 februárjában az egyetemen bevezetésre került a rendszer általános, kötelező jellegű használata. Ez azt jelenti, hogy nem nekünk kell kezdeményezni a kurzusok létrehozását, hanem a félév elején, a Neptun rendszerben kiírt kurzusok alapján a Moodle-ben megkapjuk mindazokat a kurzusokat, ahol oktatunk. A hallgatók beíratása is automatikusan történik. Úgy gondoltuk, mivel feladatbeadásra már régóta használtuk a rendszert, ideje továbblépnünk, s a rendszer más lehetőségeit is kihasználni. Mivel a mérnökképzésben az alapozó jellegű Informatika I tárgy csak elméleti kiosztású és vizsgával zárul, úgy gondoltuk, érdemes lenne a rendszer által nyújtott lehetőségeket kihasználni a vizsgáknál is. Ezért az elmúlt évben összeállítottunk egy több mint 100 kérdésből álló kérdésbankot, s az egyik szakon a vizsgát számítógép mellett, tesztvizsgaként tartottuk meg. Ebben a félévben – levelező tagozatos hallgatókkal – tovább folytattuk a számítógépes vizsgáztatást, s jövőre minden szakunkon bevezetjük. A különböző jellegű kérdések összeállításáról, a kérdések értékeléséről, két félév tapasztalatairól számolok be előadásomban.

Kulcsszavak: vizsgáztatás, Moodle e-learning rendszer, teszt

Abstract: We have been using the Moodle e-learning system at our university for more than five years. At first we used the e-learning system only for publishing electronic curricula and for uploading homework. Even at the start a lot of teachers used the system at our faculty. In February 2013 the university introduced the universal and obligatory use of this system. It means that not the teachers are the ones who have to generate the courses manually anymore but at the start of semester – according to the available courses in the Neptun system – teachers receive in the Moodle system all their courses individually tailored. The registration of the students happens automatically too. As we used the system for uploading of exercises for a long time, we felt encouraged to go on and take advantage of other possibilities of system. At the engineering education the Information Technology I course is a theoretical course with exam. We thought, it is worth exploring other possibilities of Moodle system at the exams too and make use of it. Last year we created a so called “Question bank” with more than 100 questions and started a pilot with the full-time students of the Environmental Engineering Faculty who took their exam on the computer. In this semester we expanded the online exam system and involved correspondence students of the same specialization. Next year we plan to rollout the new system at all specializations. In my presentation, I will walk you through the methodology of the test questions, the online evaluation system and will share with you our overall experience of two semesters.

Keywords: interrogation, Moodle e-learning system, test

1. Bevezetés

Egyetemünkön, illetve jogelődjén a Budapesti Műszaki Főiskolán 2008 óta használjuk a Moodle e-learning rendszert. A rendszer bevezetéséről az előző Informatika a felsőoktatásban konferencián már beszámoltunk [1]. Kezdetben elsősorban a tananyagok feltöltésére, illetve az informatikai tárgyakon belül a házi feladatok feltöltésére használtuk a rendszert. Karunkon, a Rejtő Sándor Könnyűipari és Környezetmérnöki Karon - az egyetem néhány más karával ellentétben, nem voltak saját e-learninges rendszerek, így mi rögtön a központi rendszert

kezdtek el használni. Egyetemünkön 2013 januárjában megváltozott a Moodle rendszer használata [3]. Létrejött az Elektronikus Tananyagok Irodája, s a Moodle rendszer használatát úgymond kötelezővé tették minden oktató és hallgató számára. Ettől az időponttól kezdve nem nekünk, oktatóknak kell kérni a kurzusok létrehozását, hanem a kurzusok a Neptun rendszer alapján központilag kerülnek létrehozásra, ha egy hallgató felveszi a Neptunban a tárgyat, bekerül az elearning kurzusba is. Minden oktátónak használnia kell a rendszert, s az elektronikus tananyagokat kötelezően a Moodle rendszerbe kell feltölteni. Ahhoz, hogy az átállás problémamentes legyen, minden karon kinevezésre került ún. Moodle koordinátor, akinek feladata az oktatók munkájának segítése. Tanfolyamokat tartottunk a rendszer használatáról, ahol megmutattuk a leggyakrabban használt funkciókat. Azt mondhatjuk, hogy karunkon most már minden oktató használja a rendszert.

2. Informatika oktatás és a Moodle rendszer

Az Óbudai Egyetem mérnökképzésében karunkon 3 szakot oktatunk, könnyűipari, környezetmérnöki és ipari termék és formatervező mérnöki szakot, továbbá hozzánk tartozik a Keleti Károly Gazdasági Kar műszaki menedzser szakos hallgatóinak oktatása. A 2008-ban bevezetett tanterv szerint a tárgy 3 részre bontható. Az első félévben az Informatika I tárgy keretein belül heti 2 óra előadás vizsgával, a második félévben az Informatika II tárgy szintén heti 2 óra előadás és vizsgával zárul. Továbbá van a második félévben egy Informatika labor tárgy, ez is heti 2 óra, de évközi jeggyel zárul. A Moodle rendszert mind a 3 tárgynál használjuk.

Az elméleti tárgyaknál egyrészt az előadások anyagához kapcsolódó diákat töltjük fel, másrészt a hallgatóktól itt kérjük a beadandó házi feladatot. A labor tárgynál az órai munkához szolgáló alapfájlokat innen tölthetik le a hallgatók, az óra végén feltölthetik a munkájukat, így otthonról is hozzáférhetnek külön küldözgetés, másolás nélkül. Nemcsak a beadandó házi feladatot kell feltölteniük a hallgatóknak a Moodle rendszerbe, hanem a labor zárthelyik elkészítése is a rendszer segítségével történik. Itt vannak az alapfájlok, s ide kell megadott formában feltölteniük az elkészített fájlokat is. Az értékelés eredménye is ide kerül fel.

Az Informatika tárgyak oktatásáról, az eredményekről korábbi konferenciákon már beszámoltam [2]. Nem tartjuk szerencsésnek a tárgy ilyen jellegű kiosztását, de igyekszünk ezen belül a lehető legjobbat kihozni. Mit oktassunk az elméletben, ami utána számon is kérhető? Hogyan történjen a számonkérés? Karunkon kötelezővé tették az előadások látogatását. Hogyan ellenőrizzük ezt, ellenőrizzük-e egyáltalán?

Az Informatika I tárgy a környezetmérnök levelező szakot kivéve az oktatás első félévében van. Az oktatás témájánál egyrészt az általános műszaki műveltséghez tartozó témákat igyekszünk előtérbe helyezni, másrészt összefoglalni a korábban tanultakat:

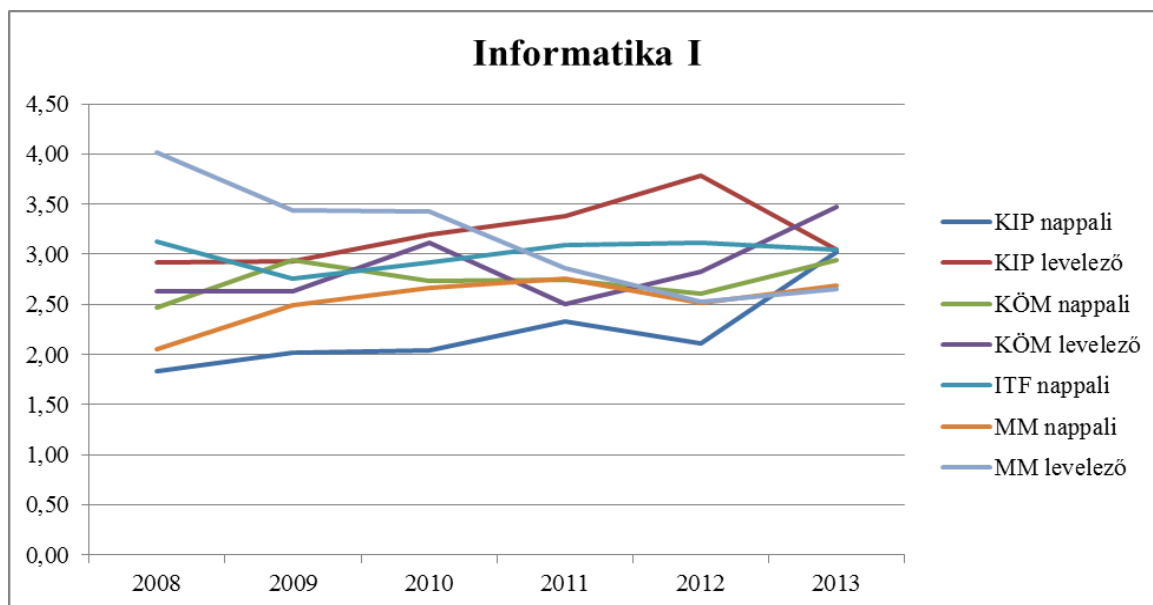
- alapvető hardver és szoftver ismeretek,
- rövid történeti áttekintés
- számítógépek adatábrázolása,
- matematikai logika alapjai,

- hálózati alapismeretek
- szövegszerkesztési, táblázatkezelési és prezentációkészítési alapok.

Természetesen itt nemcsak a Microsoft programcsomag kerül előtérbe, hanem a nyílt forráskódú szoftverek, a felhő alapú dokumentumkezelés, valamint a Prezi is szóba kerül.

Ekkor még nem célszerű túl szabadon engedni a hallgatókat, ezért a jelenlét ellenőrzésére azt a módszert vezettük be, hogy előre be nem jelentett 4 alkalommal úgynevezett katalógus zárthelyit írtatunk – 5-5 tesztkérdéssel. Az aláírás feltétele 50 %, aki egy tesztet sem írt meg, annak a hiányzása biztosan túllépi a megengedett 30 %-ot, így azok a hallgatók letiltásra kerülnek. A tesztek vagy kinyomtatva kapták meg a hallgatók, vagy kivetítettük őket, s csak az eredményt kellett megadniuk. A tesztkérdések többfélék voltak: igen-nem válaszok, egy vagy több jó válasz lehetséges megadásával. Ha több jó válasz is volt, csak azt a megoldást fogadtuk el, ha minden jó megoldást megadott a diák. 50 % alatti eredmény esetén aláírás pótló vizsgán lehetett az aláírást megszerezni. A nappali tagozatos környezetmérnök hallgatóknál ebben a tanévben 69 % szerezte meg az aláírást elsőre, az aláírás pótló vizsgával együtt 78 % jöhetett vizsgázni. A teszten kívül az aláírás feltétele volt még az elfogadott beadandó feladat – egy tetszőleges programmal elkészített, a volt középiskoláját bemutató prezentáció. Levelező tagozaton csak a házi feladatot kértük a hallgatóktól. ECDL vizsgával rendelkező hallgatóink az első félévben az aláírást megkapják a bemutatott ECDL bizonyítványra, nekik csak vizsgázniuk kell.

A félévet írásbeli vizsga zárta – itt általában 20 kérdés volt, számolásos, egy-két szavas választos és néhány kifejtős kérdés. Az elmúlt évek Informatika I vizsgaeredményeit szemlélteti az 1. ábra – 1-2 kiugró eredménytől eltekintve 2,5 és 3,5 közötti eredményekkel. A vizsgaeredmények az őszi féléveket jelölik a környezetmérnök levelezőket kivéve, nekik egy félévvel később van a tárgy.



1. ábra. Informatika I tárgy vizsgaeredményei

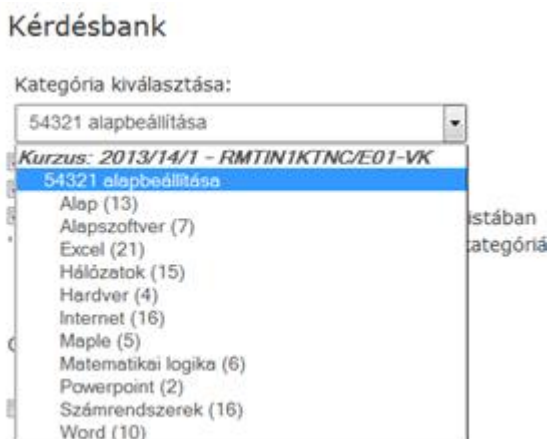
3. Vizsgáztatás a Moodle rendszerben

Már régóta gondolkodtunk azon, hogyan lehetne az írásbeli vizsgát kiváltani egy olyan vizsgával, amely:

- megkönnyíti a nagy tömegű dolgozatok értékelését,
- a mai diákok számára szimpatikusabb, mint az írásbeli vizsga,
- lehetővé teszi a tananyag átfogó számonkérését,
- kiküszöböli azt, hogy a diákok írását sokszor elég nehéz elolvasni,
- reális eredményt ad a diákok tudásáról – nincs szignifikáns eltérés az előző évek eredményeitől.

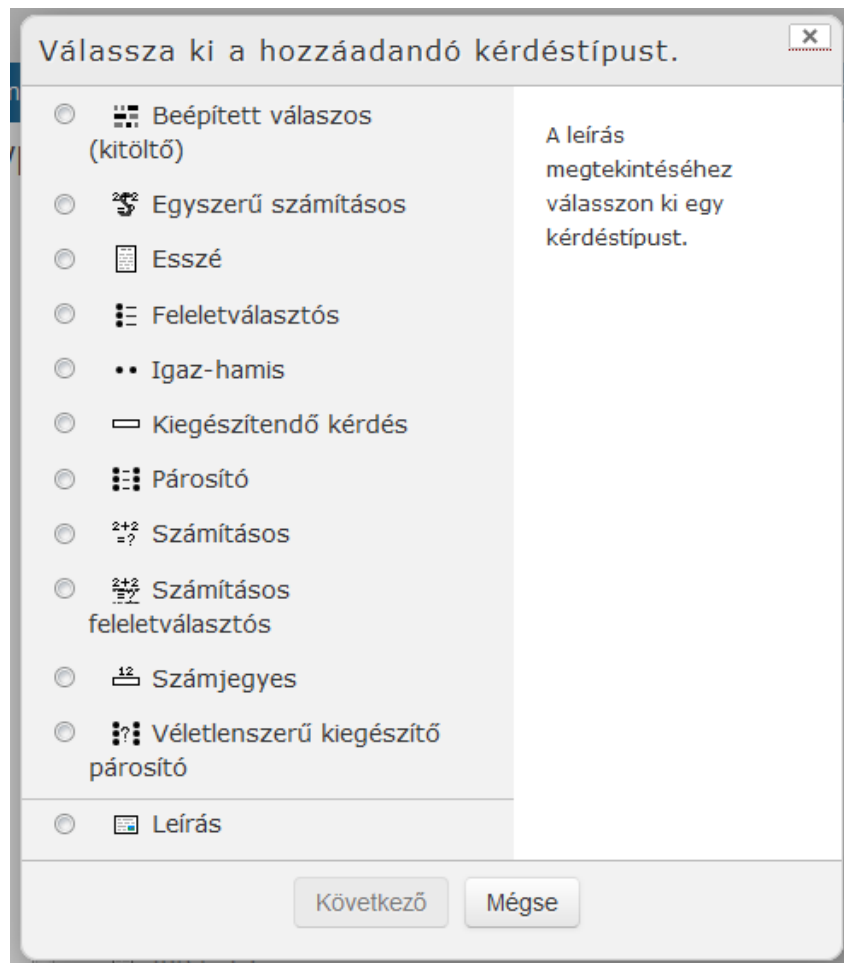
Megvizsgálva a Moodle rendszer tesztkészítési lehetőségeit, úgy gondoltuk érdemes kipróbálni diákjainknál. A rendszer bevezetése előtt osztályzási rendszerünk olyan volt, hogy 50 % kellett az elégséges eléréshez, s 12 %-onként emelkedtek a jegyek. Úgy gondoltuk, ezt meg kellene tartani a továbbiakban is. A feleletválasztós teszteknel azonban a véletlennek is van szerepe. Ha csak olyan tesztkérdéseket használunk, ahol egy helyes választ kell megadni, akkor a válaszok számától függően például 4 megadott válasz esetén a helyes találati arány 100 %, ha tudja a hallgató a választ, 25 % akkor, ha nem tudja a választ. Ezért, hasonlóan több más tesztet használó versenyhez (matematikai Zrínyi és Kenguru versenyek), úgy gondoltuk, hogy nemcsak a jó választ kell pontoznunk, hanem a hibás válasz bejelölését mínusz ponttal kell értékelnünk.

A Moodle rendszerben először is létre kellett hoznunk egy kérdésbankot. Ehhez a tananyag alapján kategóriákat állapítottunk meg, a kategóriák a 2. ábrán láthatók:



2. ábra. Első félévi kérdések kategóriái

Az egyes kategóriákon belül különböző típusú kérdések hozhatók létre. A létrehozható kérdéstípusokat a 3. ábrán tekinthetjük meg:



3. ábra. Kérdéstípusok

Mi egyelőre háromféle kérdéstípust használtunk, feleletválasztós, igaz-hamis és esszé típusú kérdéseket. A kérdések helyes megválaszolása 2 pontot ért. Az esszékérdéseket a tanárnak kell értékelnie, ez hasonló ahhoz, mint amit eddig csináltak, arányos pontozással. Az igaz-hamis kérdésnél nincs lehetőség mínusz pont adására, itt vagy megkapta a hallgató a 2 pontot, vagy 0 pontot ér el.

A feleletválasztós kérdések esetén további két lehetőség közül választhattunk. Amennyiben 1 helyes válasz van, jó válasz esetén 2 pontot kap a hallgató, rossz válasz esetén -1 pontot. Több lehetséges jó válasz esetén más a helyzet. Míg a papíron kitöltött tesztek esetében csak a teljesen jó választ fogadtuk el, itt arányosan megkapják a jó válaszáért a pontot – ha 2 helyes válasz van, 1-1 pontot érnek, stb... A hibás válaszok esetén itt is arányosan pontlevonás történik, de ebben az esetben nincs negatív pont, a rossz válasz esetén is csak 0 pontot kap a hallgató – a rendszer valamiért így van beállítva, ez nem tőlünk függ.

Egyelőre összeállítottunk egy 115 kérdésből álló kérdésbankot. A kérdésbankból készült egy 20 kérdésből álló Próbateszt – amely minden témából tartalmazott kérdéseket, ahhoz hasonlóan, mint amit a vizsgán is szándékoztunk adni. Ezt kitettük a kurzushoz, lehetőséget adva a hallgatóknak, hogy gyakorolják a teszt kitöltését, megbarátkozzanak vele. Továbbá felkértem a kollégáimat is a teszt kipróbálására – egyrészt a kérdésbank átnézésére, másrészt a

próbateszt értékelésére is. Kaptam javaslatokat a kérdések bővítésére, néha pontosítására, s így lett készen a kérdésbank. Ahhoz, hogy a teljes tananyagot áttekintsük a vizsga során, én 20 kérdésben gondolkoztam. Mivel a Moodle rendszer is az interneten keresztül érhető el, így a vizsga során az internet letiltása nem lehetséges. Fontos kérdés volt, mennyi időt adjunk a vizsgára – legyen elég idejük a kérdések átgondolt megválaszolására, de ne az internetről keresgéljék a válaszokat. Mivel kolléganőimtől azt a választ kaptam, hogy 15 perc alatt úgy töltötték ki a 20 kérdésből álló tesztet, hogy volt idejük a válaszoknak utána olvasni a Wikipédián, ezért - figyelembe véve a diákok lassabb munkáját is, a 20 kérdés kitöltésére 15 perces határidőt állítottam be. Minden alkalommal 1 esszékérdés volt, általában 2-3 igaz-hamis kérdés, a többi pedig feleletválasztós.

Néhány kérdés látható az alábbi (4-7. ábra) ábrákon:

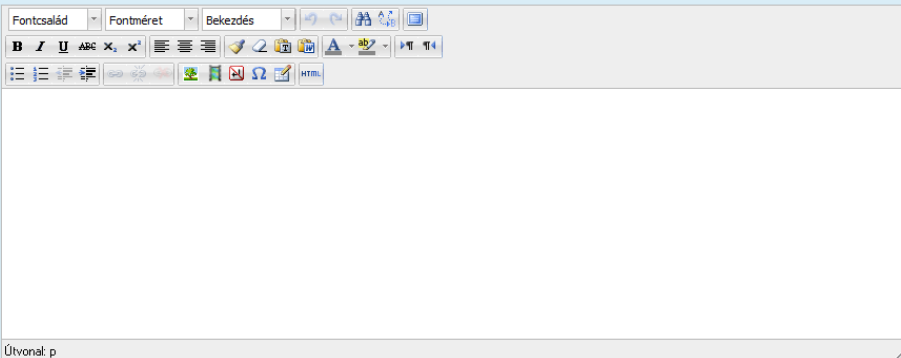
Válassza ki az alábbi felsorolásból azt az eszközt, ami nem illik a többi közé!

Válasszon ki egyet:

- a. Mikrofon
- b. Vonalkód olvasó
- c. Rajzdigitalizáló
- d. Hangszóró

4. ábra. Feleletválasztós teszt 1 jó válasszal

Mit tud az IP címről?



Útvonal: p

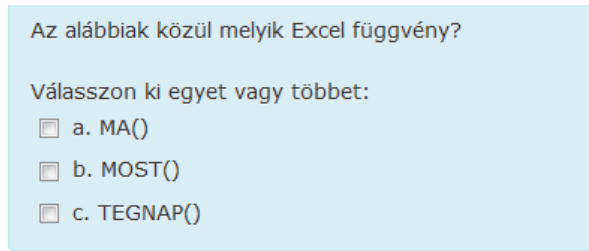
5. ábra. Esszékérdés

A Wordben egy bekezdés Enter-től Enter-ig tart.

Válasszon ki egyet:

- Igaz
- Hamis

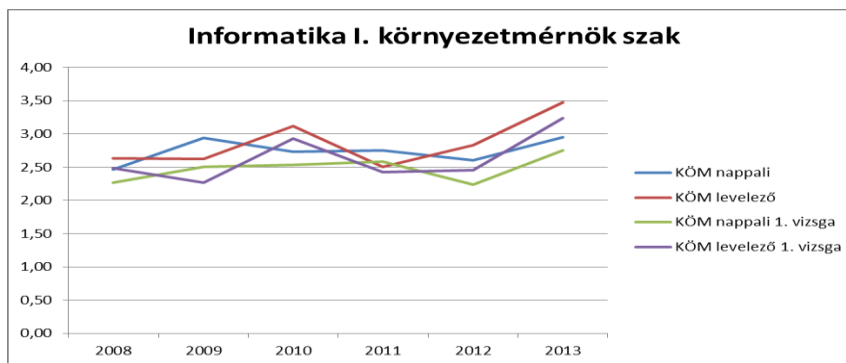
6. ábra. Igaz-hamis kérdés



7. ábra. Feleletválasztós teszt több helyes válasszal

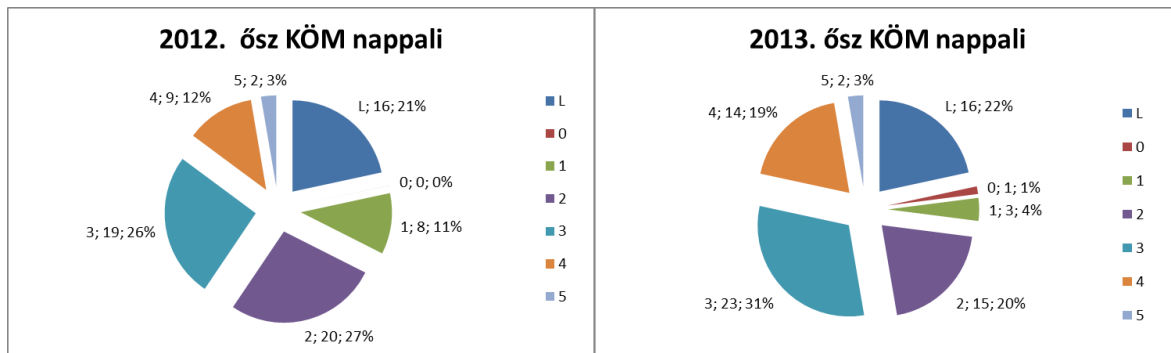
4. Vizsgaeredmények

Eddig két félévben teszteltük a vizsgát a környezetmérnök szakos hallgatóknál. A 2013/14. tanév első félévében a környezetmérnök szakos nappali tagozatos hallgatók vizsgáztak ezzel a módszerrel, a második félévben pedig a levelező tagozatos hallgatókat és néhány nappalis vizsgakurzusos hallgatót vizsgáztattunk így módon. Megfigyelve az eredményeiket, azt láthatjuk, hogy nem különböznek lényegesen a korábbi évek eredményeitől, talán egy kicsit jobbák csak, főként a levelező tagozaton. A 8. ábrán láthatjuk a környezetmérnök szakos hallgatók eredményeit, nemcsak a végeredményt, hanem az első vizsgák utáni eredményt is.

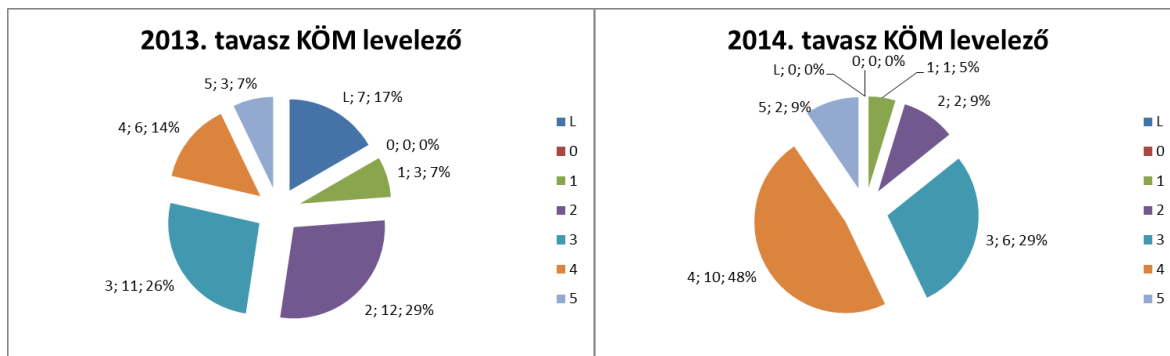


8. ábra Környezetmérnök hallgatók eredményei

Az eredmények megoszlását a 9-12. ábra mutatja. Megfigyelhetjük, hogy az elégtelenek száma csökkent, némileg több közepes és a jó eredmény.



9-10. ábra Nappalis eredmények megoszlása



11-12. ábra Levelezős eredmények megoszlása

A L jelöli a letiltást, 0 azt, ha nem vizsgázott a hallgató egyáltalán. A különböző típusú kérdéseket vizsgálva azt láthattuk, hogy a kifejtős kérdésekre teljes választ ritkán adtak a hallgatók. A feleletválasztós teszteknel az Excel, illetve a Maple kérdések esetén volt a legrosszabb a válasz hasonlóan ahhoz, amikor papíron vizsgáztak diákjaink.

Összességében elmondhatjuk, hogy a Moodle rendszerben történő vizsgáztatás beváltotta a hozzáfűzött reményeket. A továbbiakban szándékaink szerint teljeskörűen, a többi általunk oktatott szakon is bevezetjük ezt a vizsgáztatási formát az elmélet számonkérésére. Ehhez természetesen a kérdéskönyt bővítenünk kell. Legalább 3-400 kérdésre lesz szükségünk ahhoz, hogy ne ismétlődjenek rendszeresen a kérdések. A rendszerben ugyan beállítható, hogy a kérdéseket összekeverje és a válaszok sorrendjét is keverten adja. Így az nem probléma, hogy a számítógépeknél egymás mellett ülő diákok ugyanazt a tesztet töltik ki, de azért a jelenleginél több kérdésre van szükségünk. A diákok is jónak tartották a vizsgát, úgy vettük észre, szimpatikusabb nekik, mint a régi módszerrel történő vizsgáztatás.

5. Köszönetnyilvánítás

Köszönetemet fejezem ki egyetemünk Elektronikus Tananyagok Irodája igazgatójának, Hegyesi Franciskának a tesztek elkészítéséhez nyújtott kezdeti segítségért, továbbá kolléganőimnek, Gyöngyné Maros Juditnak és Kormány Eszternek a tesztkérdések ellenőrzéséért, a kérdések összeállításához nyújtott hasznos javaslatokért. Munkájukra tovább is számítok a kérdéskönyv bővítésében.

Irodalomjegyzék

- [1] Ambrusné Dr. Somogyi Kornélia, Hegyesi Franciska: E-learning rendszer használata az Óbudai Egyetemen, Informatika a felsőoktatásban 2011 Konferencia, Debrecen, 2011. augusztus 24-26. DVD kiadvány pp. 502-510. – ISBN 978-963-473-361-1.
- [2] Ambrusné Dr. Somogyi Kornélia: Informatikai műveltség - informatika oktatás a mérnökképzésben, MAFIOK 2013 Közlemények: Matematikát, fizikát és informatikát oktatók XXXVII. országos konferenciája. Miskolc, 2013.08.26-.08.28. Miskolc: Miskolci Egyetem, 2013. pp. 5-14.(ISBN:978-963-358-035-6; 978-963-358-036-3).

- [3] Hegyesi Franciska: Moodle in Higher Education and at Óbuda University, IEEE 8th International Symposium on Applied Computational Intelligence and Informatics (SACI), Temesvár, 2013. május 23-25. pp. 67 - 71

E-learning Sstratégiák az Óbudai Egyetem felnőttképzésben

E-learning Strategies for Óbuda University in adult education

Hegyesi Franciska^a

^aÓbudai Egyetem/Kandó Kálmán Villamosmérnöki Kar
hegyesi.franciska@kvk.uni-obuda.hu

Absztrakt: Egyre több felnőtt ül vissza az iskolapadba, egyre többen választják az egyetemek képzéseit. Egyesek azért, mert új szakmát szeretnének tanulni, mások azért mert szükségük van a diplomára karrierjük szempontjából.

Napjainkban nagyobb szerepet kapnak a virtuális képzések, egyre több külföldi elit egyetem hirdet online kurzusokat, 3-18 hónap időintervallumban. Egyes kurzusok ingyenesek, de a kreditekért fizetni kell, ezek között vannak, amelyek pár ezer dollárba kerülnek, mások több tízezerbe. Ez a tendencia a világban, MOOC fogalomként terjedt el. A "The New York Times" 2012-es évet a MOOC évének nevezte. A MOOC kifejezést Dave Cormier és Bryan Alexander alkotta 2008-ban, válaszul egy "Connectivism and Connective Knowledge" (CCK08) kurzusra. A kurzuson 25 fizetős, valamint 2200 független hallgató vett részt. Minden kurzus tartalma elérhető volt valamilyen keretrendszeren keresztül és az online hallgatók különféle együttműködési eszközöket használhattak, például bogbejegyzéseket.

Az Óbudai Egyetem a következő években erőteljesebben kívánja megvalósítani a 2013-ban elkezdett programját, amelyben a hallgatók a felvett kurzusaikat segítő tananyagokhoz, minél nagyobb mértékben tudjanak hozzáférni.

Ezen törekvések közé tartozik az a kezdeményezés is, hogy a 2013/2014-es tanév II. félévében, 5 szabadon választható tárgyból indítottunk e-learning kurzust, ami azt jelenti, hogy nem volt személyes kontaktus az oktató és hallgatók között. Az előadásban bemutatjuk, hogyan vélekednek hallgatóink és oktatóink az Óbudai Egyetem e-learning stratégiáról, az egyetem hozzáállásáról a felnőtt tanulókhöz, milyen tapasztalatokkal zártuk az 5 elektronikus kurzust, milyenek az elvárásaik azoknak a hallgatóknak, akik a levelező, távoktatás vagy esti képzési formát választották.

Kulcsszavak: E-learning, Moodle, MOOC

Abstract: More and more adults return to school, they chose courses from a variety of universities. Some chose further training because they want to change their jobs, others because they need a degree to progress in their carriers. Worldwide this tendency was speed as MOOC concept. "The New York Times" named 2012 the year of MOCC. The MOOC expression was created by Dave Cormier and Bryan Alexander in 2008 as an answer to the "Connectivism and Connectie Knowledge" (CCK08) course. The course had 25 paying and 2200 independent students. Every course was accessible in some form of online system, which had communication sharing opportunities, like blogs. The University of Óbuda would like to use more vigorously the program started in 2013, in which the students could access as much as possible learning materials for their chosen courses. In the second half of 2013-2014 academic year, the university made accessible 5 free of choice e-learning courses, in which students did not have personal contact with a teacher. This presentation summarizes the opinions of our students about these 5 online courses, the teaching staff, the University's support of adult education and the expectations of our students how study either on a distance learning or correspondence courses.

Keywords: E-learning, Moodle, MOOC

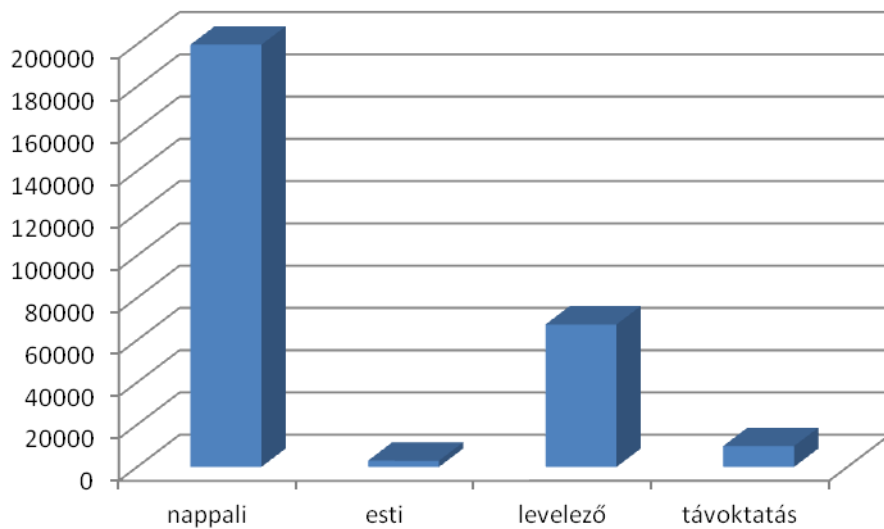
1. Bevezetés

Napjainkban egyre több felnőtt ül vissza az iskolapadba, mivel a tudásalapú gazdaságban és társadalomban a tanulás a versenyképességet és az életminőséget meghatározó egyik legfontosabb tényező. A tanulás létfontosságú mind az egyén, mind a társadalom és a gazdaság szempontjából. Ez óhatatlanul maga után vonja a tanulás időbeni kiterjedését.

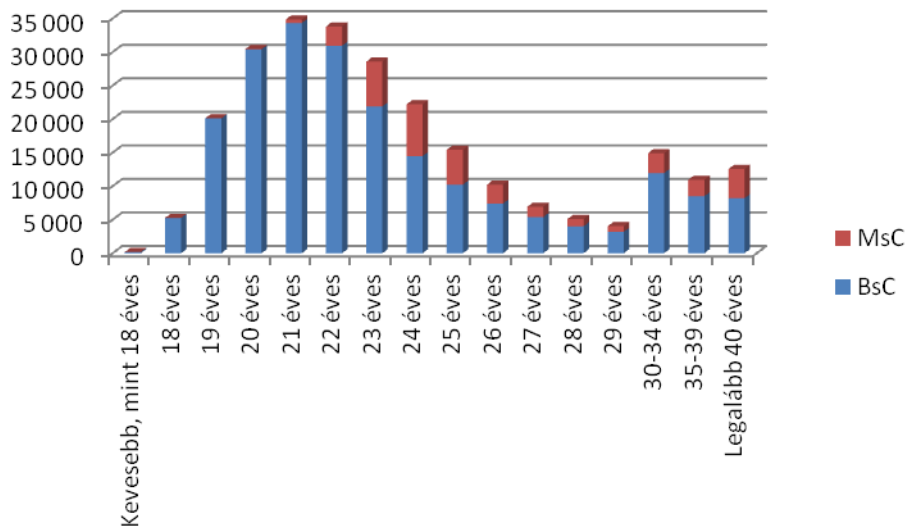
Az, hogy ki számít felnőttnek számtalan pszichológiai meghatározásban olvashatjuk, de az, hogy felnőttnek tekinthető e az egyetemi hallgató, abban már nem olyan nagy az egyetértés. A felsőoktatásban részt vevő fiatal felnőttek tipikusan a 18–25 év közötti életkorcsoportba tartoznak. Naptári és szociológiai életkoruk alapján, tehát biológiai és jogi szempontból felnőttek, jogi értelemben nagykorúak.

Az egyik szemlélet "A felnőttképzéshez szűkebb értelemben a felnőttek felsőoktatásban folyó esti- és levelezőoktatása, távoktatása, és mindezekben a formákban posztgraduális képzése tartozik, tágabb értelemben viszont ide sorolható a nappali oktatásban a fiatalok között tanuló felnőttek képzése is, aminek a világ számos felsőoktatási intézményében hagyományai vannak"

Magyarországon a felsőoktatási képzés az alábbiak szerint alakul:



1. ábra. Nemzeti Erőforrás Minisztérium, 2012/2013. tanév, kézirat saját szerkesztés



2. ábra. Nemzeti Erőforrás Minisztérium, 2012/2013. tanév, kézirat saját szerkesztés

E két diagram alapján kijelenthetjük, hogy a felsőoktatásban a hallgatók 20% biztosan felnőttnek tekinthető (47 559 hallgató 26 évesnél idősebb a 216 454-ből).

A diagramból azt is láthatjuk, hogy a 30 és 40 év közötti felnőttek többnyire visszaülnek az iskolapadba. Ennek okai szerteágazóak lehetnek (38 418, 30 és 40 év közötti hallgató).

Abból a kutatásból, mely azt vizsgálta, hogy a hallgatók milyenek ítélik meg az oktatók módszertani felkészültségét és mit várnak el az Egyetemről kiderül, hogy többségük felnőttnek tekinti magát és többségük úgy ítéli meg, hogy felnőttnek tekintik őket az oktatási intézményben is. A 848 kitöltőből, akik nappali képzésben vesznek részt, 80% nyilatkozta, hogy felnőttnek tekinti magát.

Mindezek függvényében levonhatjuk azt a következtetést, hogy a felsőoktatás képzéseiben többnyire felnőttek vesznek részt, de e kutatásnak nem e folyamatok feltárása volt a célja, így ezeket az összefüggéseket nem vizsgáljuk mélyrehatóan.

A felsőoktatás számára az e-generáció kezelése, az oktatás hatékonyságának szem előtt tartása alapvető érdeke. A digitális nemzedék és a felnőttek számára az infokommunikációs eszközök bevonásával számukra megnyílhatnak a terek, átjárhatóvá válhatnak az oktatási intézmény által biztosított keretek, a tudás megszerzése nem idő és helyhez kötött többé.

A magyar felsőoktatásban az a különleges helyzet áll fent, hogy az oktatók kinevezéséhez nem szükséges a pedagógiai (módszertani) végzettség, pedig ez elengedhetetlen lenne az oktatás hatékonysága érdekében.

Magyarországon az e-learning programok, nem terjednek túl gyorsan, ennek egyik fő oka, hogy kevés a speciális szakmai felkészültségű szakember. A felsőoktatásban oktatók még mindig jobban érzik magukat egy frontális (face to face) oktatási folyamatban, egy tanteremben, ahol kihasználhatják oktatói fölényüket nem hagyván helyet az interakciókra.

Ezeknek a kurzusoknak, lényeges képessége, hogy költséghatékonyak, személyre szabottak, interaktívak tudnak lenni, előnyöket jelentenek az oktatók és hallgatók számára is.

2. E-learning

Sok definícióval találkozhatunk az e-learningről, melyek nem tükrözik saját elképzelésünket. A tartalomfejlesztők másképp beszélnek/értelmezik az e-learninget, mint a tananyagokat közlétezők, ha ez nem egy és ugyanaz az egyén.

Az LMS (Learning Management System) forgalmazók/működtetők különbözően befolyásolják a leendő vásárlókat, hogy mit gondoljanak az e-learningről, mint a tartalomfejlesztők. Mindenkinek ki kell alakítania a saját definícióját, mely visszatükrözi, hogy mire van szüksége egy sikeres e-learning stratégia megvalósításához.

Az én definícióm az e-learning-re:

"Az e-learning a tudás és készségek folyamatos asszimilációja, szinkron és aszinkron tanulási esemény, melyet az Internet technológiák segítségével kézbesít, támogat vagy szabályoz."

A kihívás az e-learning-be ugyanaz, mint minden más oktatási programban az, hogy olyan módon kell felépíteni a tananyagot, hogy az összeegyeztethető legyen az emberi tanulási eljárásokkal. Ahhoz, hogy hatásos tudjon lenni az oktatási stratégiáknak támogatniuk kell ezeket az eljárásokat, vagyis nekik elő kell segíteniük a pszichológiai eseményeket, amik nélkülözhetetlenek a tanuláshoz.

1970-ben Knowels meglepő pontossággal vetítette előre az olyat, mint "e-learning":

"A végéhez közeledünk annak a korszaknak melyben alapvető meggyőződésünk, hogy a tiszteletre méltó tanulás csak osztályteremben, kampuszokban zajlik. A felnőttek kezdik követelni, hogy az ő tanulmányaiknak ideje, helye és üteme nekik kényelmes keretek között történjen. Valójában biztosnak érzem, hogy a legtöbb oktatási szolgáltatás a század végéig (ha nem évtized) elektronikus úton fog történni... A nagy kihívásunk most, hogy megtaláljuk a

lehetőséget, hogy fenntartsuk az emberi érdeklődést, hogy megtanuljuk új módon használni a médiát"

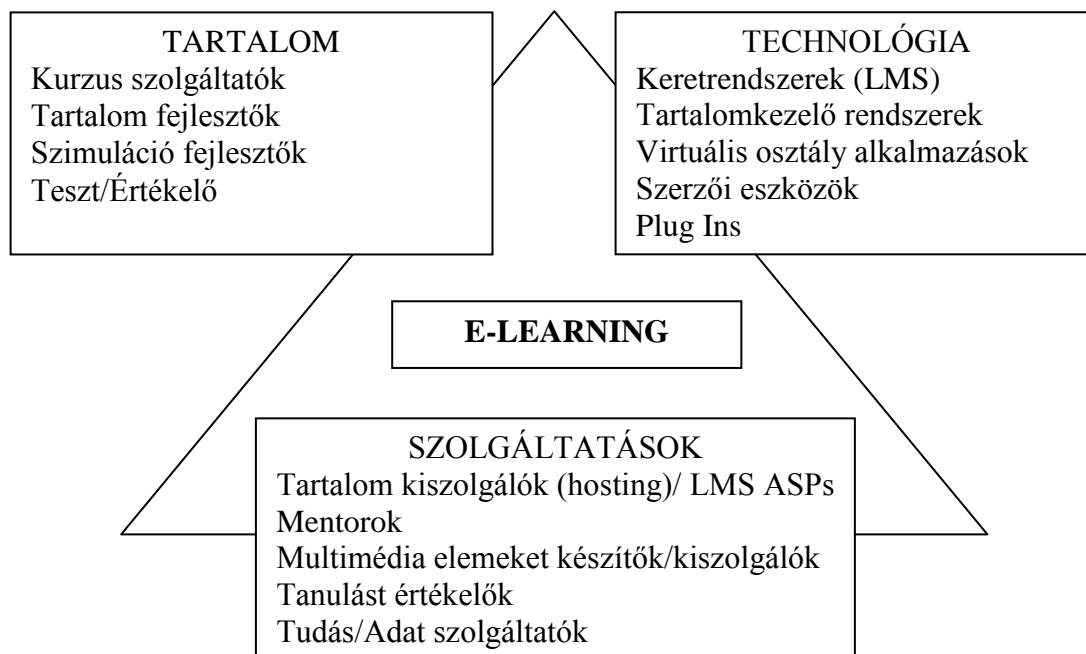
Egy jól strukturált tananyag tartalmaz egy tartalmat, mely lényeges a tanulási cél érdekében, felhasználja a multimédiás elemeket, mint szavak, képek melyek segítségével átadja a tartalmat, mintapéldákat, gyakorlatokat és visszacsatolásokat alkalmaz, ezzel is támogatván a tanulási folyamatot. Lehet oktató által vezetett szinkron (synchronous e-learning) tanulási folyamat vagy egyéni tanulás (asynchronous e-learning).

Mindez segít a tanulóknak, hogy új ismereteket és készségeket szerezzenek, melyek kötődnek az egyéni tanuláshoz, mely javítja a szervezési teljesítményt.

A technológia lehetővé teszi, mint folyamat, mint keret, hogy az e-learning-nek hatalma legyen átalakítani a munkavállalók tanulási szokásait egy új gazdaságban, ahol a tudásnak ára van és a változás állandó.

Ma az e-learning technológiákat azonosítják az Internet technológiákkal, a kurzusok házigazdája egy Web szerver és különböző csatornák által elérhető, akár egy honlapról vagy LMS rendszer keretében.

Az e-learning három fő szegmense, a tartalom, a technológia és a szolgáltatások. [2]



3. ábra. Az e-learning három fő szegmense

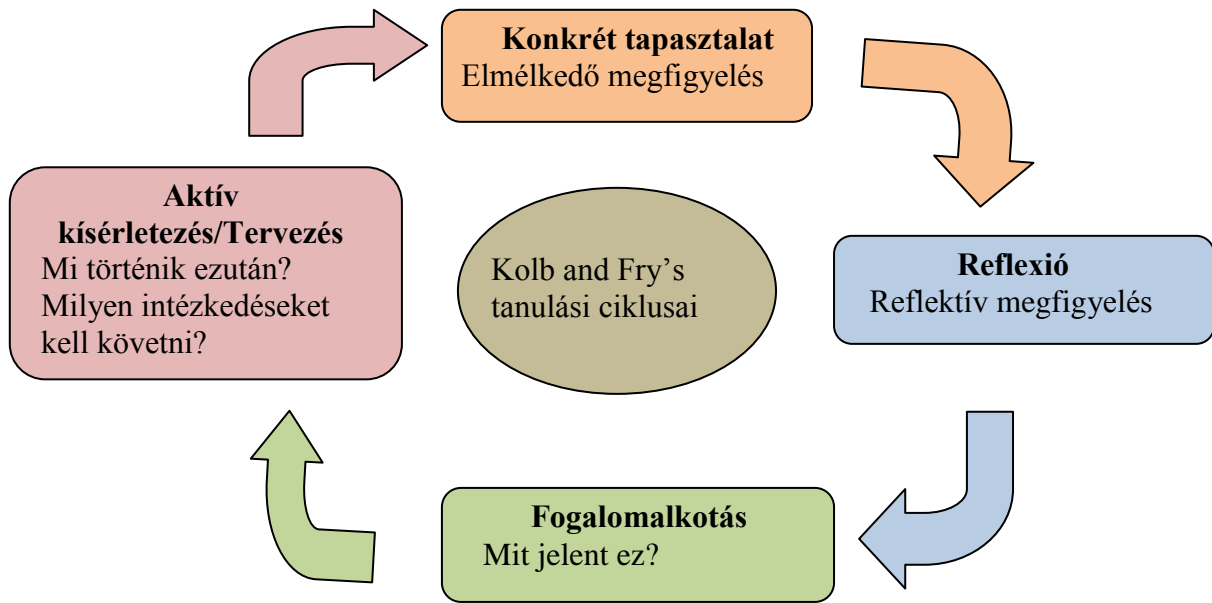
Egy e-learning kurzus keretében a hallgatók azt a rendszert fogják használni, amit felépítünk nekik, de a használaton túl el kell fogadniuk, hogy ez az új módszer, amely képes létrehozni egy alapvető változást a tanulásban.

Továbbra is egy korszakalkotó hivatkozási pont, hogy 1975-ben az amerikai pszichológus David Kolb és társa Roger Fry megalkotott egy négy szakaszból álló tanulási modellt, mely képviseli a tapasztalati tanulás ciklusát.

Kolb érdekelt volt azon folyamatok feltárásában, hogy értelmet adjon a tapasztalatoknak és a különböző tanulási stílusoknak.

Kolb és Fry azt állítják, hogy bármelyik tanulási fázis egy konkrét élménnyel kezdődik és hitték, hogy ez egy végtelen folyamat.

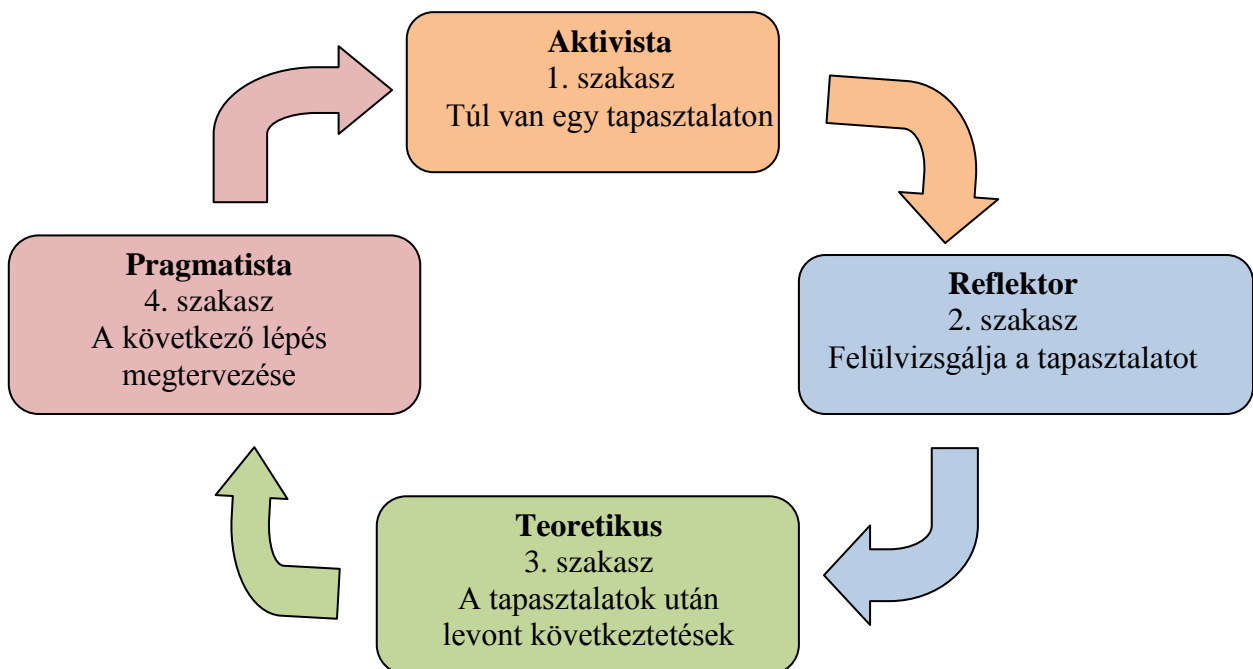
Kolb és Fry felismeri, hogy minden egyes tanulási stílusnak megvannak a maga erősségei és gyengéi, és ezért nem tökéletes tanuló az az ember, aki egy kizárólagos és egyetlen stílus közé van bezárva.



4. ábra. Kolb and Fry's tanulási ciklusai

Két vezető brit pszichológus Dr Peter Honey és kollégája, Prof. Alan Mumford megfigyelték, hogy óráikon különböző emberek különböző magatartást tanúsítottak, voltak zárkózottak és nyitottak, vidámak, komolyak, szószólók és csendesek, egyesek kockáztattak a spontán ötleteikkel, másoknak szükségük volt az alapos felkészülésre.

Honey és Mumford felfedezték, hogy ezek a viselkedési formák közvetlen hatással vannak a hallgatók tanulási folyamataira, feljegyezték ezeket az észrevételeiket és 1982-ben kiegészítették a Kolb és Fry tanulási ciklusát és elnevezték tanulási folyamatnak.



5. ábra. Tanulási folyamat

A tanulók személyiségében rejlő gyengeségek és erősségek néha tükrözik a kedvelt tanulási stílusokat.

Tanulási stratégiát úgy kell értelmezni, mint egy komplex eljárások, ahol a módszerek és az eszközök egyesülnek. [3]

Annak ellenére, hogy előnyben részesítenek egy stílust, egyik szakaszban, egy másik szakaszban viszont minden tanulónak adoptálnia kell más szerepeket is, aktivistának, teoretikusnak és pragmatistának kell lennie.

A szerepek nem követik egymást feltétlenül, például egy tanuló lehet együttműködő és reflektáló, amikor áttekinti a tanulási tapasztalatait társaival.

Az e-learning tanulási folyamatban minden tanuló minden szerepet magáévá tesz, másokkal ellentétben jellemzően több sikert tapasztalhat meg, ez a továbbiakban is pozitívan befolyásolja a tanulási szokásait.

3. E-learning stratégiák

Az e-learning stratégiának rugalmasnak kell lennie, gyorsan kell reagálni a változásokra, a követelményekre, folyamatosan fejleszteni kell az e-learning szolgáltatásokat és technológiát. Ez azt jelenti, hogy a stratégia nem épülhet egyetlen termék vagy megoldás köré. Ezt jelen körülmények között elég nehéz végrehajtani, az egyetemeken nincsenek olyan gazdasági helyzetben, hogy mindezt finanszírozzák tudnák.

Amikor egy felsőoktatási intézmény úgy dönt, hogy felépít egy e-learning stratégiát vagy bármilyen tevékenységet, mely e-learninggel kapcsolatos el kell készítenie egy felmérést/elemezést (mintegy SWOT elemzés).

A célok megfogalmazása az egyik legfontosabb elem egy stratégia elkészítésénél. Ahol nincsenek célok és tervek, ott nem várhatunk el reális mérvadó növekedést és változást, ez különösen igaz akkor, amikor a hagyományos tanítási módszereink mellé illesztünk egy keretrendszert.

Az Óbudai Egyetem jogelődjénél, a Budapesti Műszaki Főiskolán a MOODLE rendszert 2006-ban vezették be, kezdetben csak egy próbálkozás volt, különösebb stratégia nem fűződött a bevezetéséhez. Egy célunk volt, hogy az oktatók felismerjék, hogy egy ilyen elemnek is van létjogosultsága egy műszaki képzésben is.

Egy ilyen rendszer nagyban megkönnyítené a hallgatók életét, ha az egyes kurzusaik tananyagait egységesen, egy helyen érnek el helytől és időtől függetlenül.

Felmértük erőseinket, gyenge pontjainkat, mérlegeltük a veszélyeket és a rendszerben lévő lehetőségeket.

Erősségeink:

- A digitális kompetenciák nem hiányoznak a leendő felhasználóknál.
- A technológia nem jelent akadályt.
- A tanulók tudni fogják, hogy az e-learning javítani fogja a személyes teljesítményüket, a tananyagok bárhol és bármikor elérhetőek számukra.

Gyengeségeink:

- Egyre több felhasználó lesz a rendszerben, akik értékelni fogják az előnyöket és egyre magasabb elvárásaik lesznek, amit nem biztos, hogy tudunk teljesíteni.
- A tartalmak nem biztos, hogy egyformán használhatóak minden tanuló számára.
- Az oktatók ellenállást fognak tanúsítani, mert egy plusz feladatként élik meg a rendszer bevezetését.
- Mivel a felsőoktatásban nem elvárás a pedagógiai tudás az oktatók részéről nem mindig születnek minőségi tartalmak.

- Az anyagi háttér hiánya.

Lehetőségeink:

- E-learning technológia lehetővé teszi számunkra, hogy létrejöjjön egy globális tanulási közösség, ahol a legjobb gyakorlatokat meg lehet osztani.
- Az e-learning lehetővé teszi a tanulók számára, hogy ha hosszabb ideig távol vannak az iskolától, akkor sem maradnak le.
- A rugalmasság: mindenki a saját ütemében stílusában sajátíthatja el a tananyagot.

Veszélyek:

- Ha a vezetők/oktatók nem támogatják az e-learning technológia által létre jött tanulást, mert úgy vélik, hogy nem valósulhat meg valódi tudás átadása.
- A tanulók úgy értelmezik az e-learning bevezetését, mint szigorúan csak költségkímélő megoldás.
- Az e-learning vita tárgyává válhat, mivel különböző csoportok különböző e-learning technológiákat/megoldásokat támogatnak.

Az egyetem továbblépett és az OE RT (Rektori Tanács) döntésének alapján 2013. januárjától megkezdődött a Moodle rendszerben meghirdetett kurzusok folyamatos feltöltése elektronikus tananyagokkal. Valamennyi Neptunban meghirdetett kurzus átkerült a Moodle-ba és minden oktatónak ide kell feltöltenie a tananyag elsajátítását segítő elektronikus anyagait (jegyzetek, kérdéssorok, PPT-s anyagok, videók, stb.) A kurzusokhoz hozzáférésre, az arra Neptunban feliratkozott hallgatók és az oktató kapott jogosultságot, illetve kérésre az oktató bármelyik hallgatót, oktatót felveheti a kurzusára.

„The New York Times” 2012-es évet a MOOC évének nevezte, Massachusetts Institute of Technology (MIT) közzé teszi az OpenCourseWare programban tananyag részletét. A cél az, hogy a következő 10 évben az egyetemi anyagok szinte minden kurzusban szabadon választhatóak legyen az Interneten. Az MIT kurzusai csak egy a sok jelből, hogy az e-learning hosszú távú sikere elkerülhetetlen.

A legismertebb 3 MOOC gyűjtőhely a Coursera, az Udacity és az edX. Ezek közül a Coursera a legnagyobb számú és legszélesebb körű kurzusokat kínálja, a szociálpszichológiától az adatelemzésen át a hálózatok felépítéséig.

2014 első negyedévében az edX kínálatában 176 kurzus található és több mint 2,1 millió felhasználó, az Udacity 1,6 millió felhasználóval büszkélkedhet, 26 ingyenes kurzussal és 12 kreditet érő kurzussal és a legnagyobb óriás a Coursera mely 7,1 millió felhasználóval büszkélkedhet, 108 oktatási intézmény keretében 641 kurzust kínál.

Az Egyetemen úgy döntött, hogy 2013/2014- es tanév II. félévében 5 e-learning alapú kurzust indít, hogy ezek segítségével valós tapasztalatokat szerezzünk az e-learning oktatás felől.

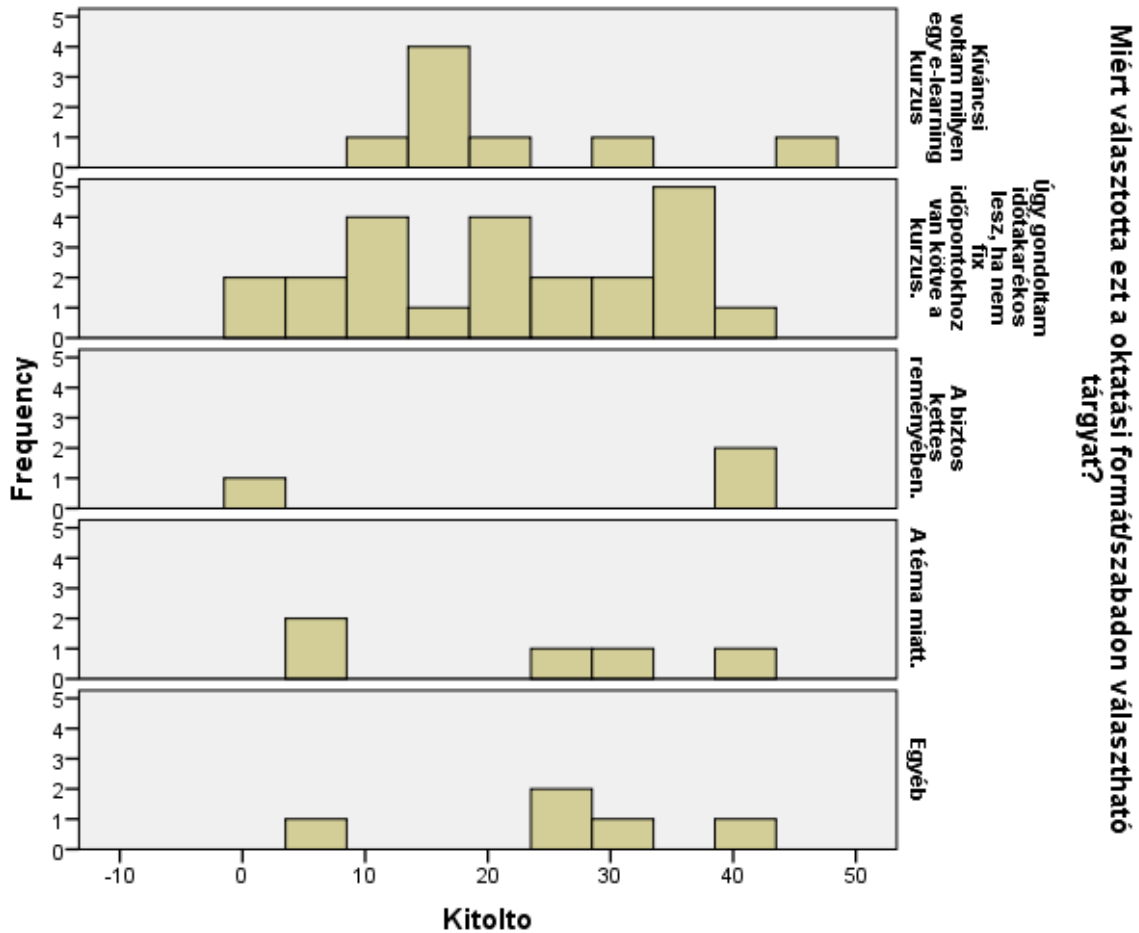
Az öt kurzus, különböző témakörben indult 3 Kar és 2 Központ keretében;

- A pedagógia alapjai (Trefort Ágoston Mérnökpedagógiai Központ), 53 hallgatóval
- ASP.NET MVC web application development using the Orchard content management framework (Neumann János Informatikai Kar), 18 hallgatóval
- Bevezetés az ergonómiába (Bánki Donát Gépész és Biztonságtechnikai Mérnöki Kar), 41 hallgatóval
- Egyéni és csoportos problémamegoldó technikák (Kandó Kálmán Villamosmérnöki Kar), 16 hallgatóval
- Összetett hálózatok vizsgálata (Alba Regia Egyetemi Központ), 12 hallgatóval

A kurzusokon 122 hallgató vett részt, a kurzusok szabadon választható tárgyak voltak, kreditpontokért.

4. Elemzés és eredmények

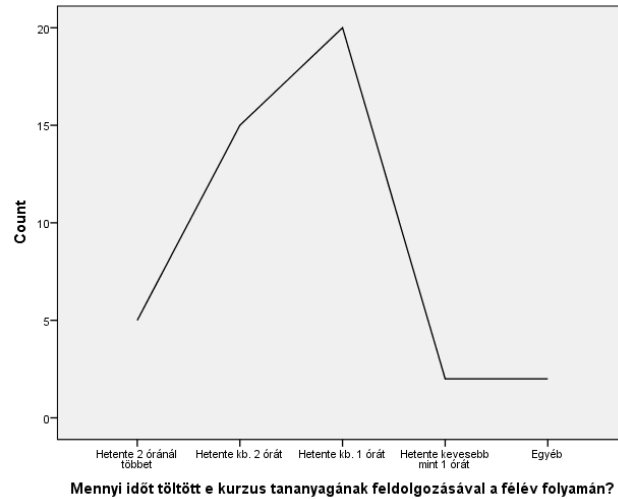
A kérdőívet 44 hallgató töltötte ki, 39 férfi, 5 nő és 22,6 volt az átlag életkor.



5. ábra. Miért választotta ezt az oktatási formát?

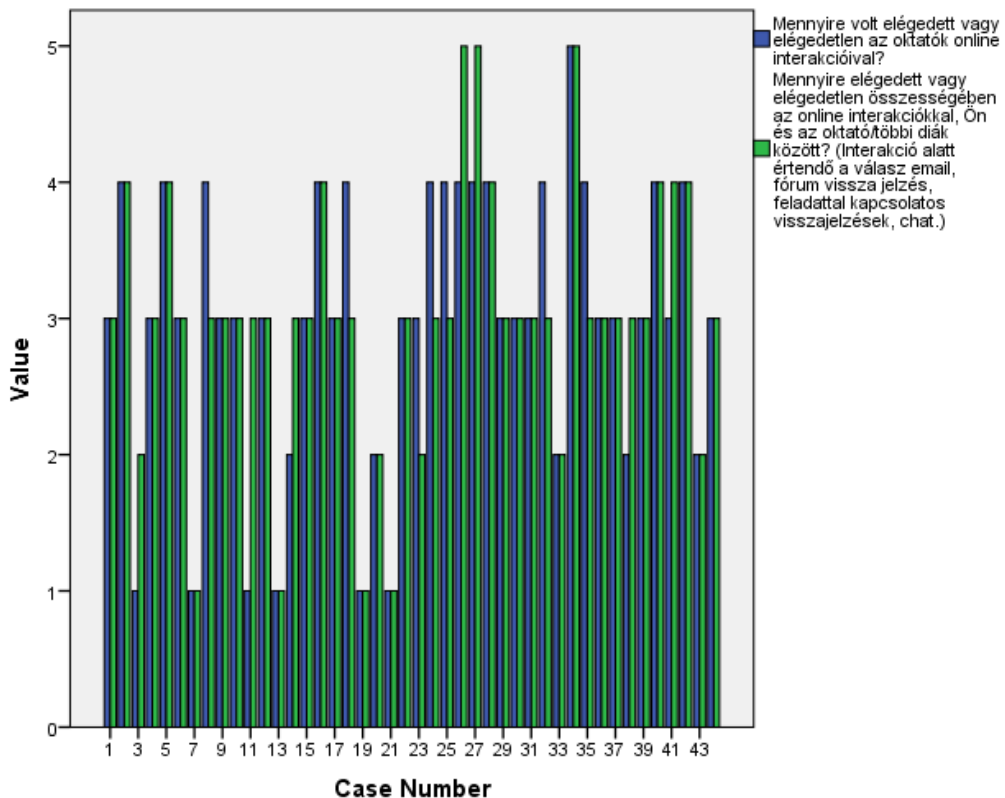
Egyik fő szempont a kurzus kiválasztásánál az időtakarékoság, hogy a kurzus ne legyen helyhez, de főleg pontos időhöz kötve ez a szempont nagyon fontos a levelező hallgatók esetében is. Korai lenne azt a következtetést levonni, hogy a "mindegy csak ne kelljen bejárni", de azért az egy fontos szempont, hogy tárgyaik időpontjait össze tudják egyeztetni más kötelezettségeikkel.

A tanulók többsége átlagban hetente kb. egy két órát töltött a tananyaggal, ami nem rossz arány, sok tárgynál csak az előadásra vagy gyakorlatra járnak be és az első megmérettetés előtt megpróbálják megtanulni a tananyagot.



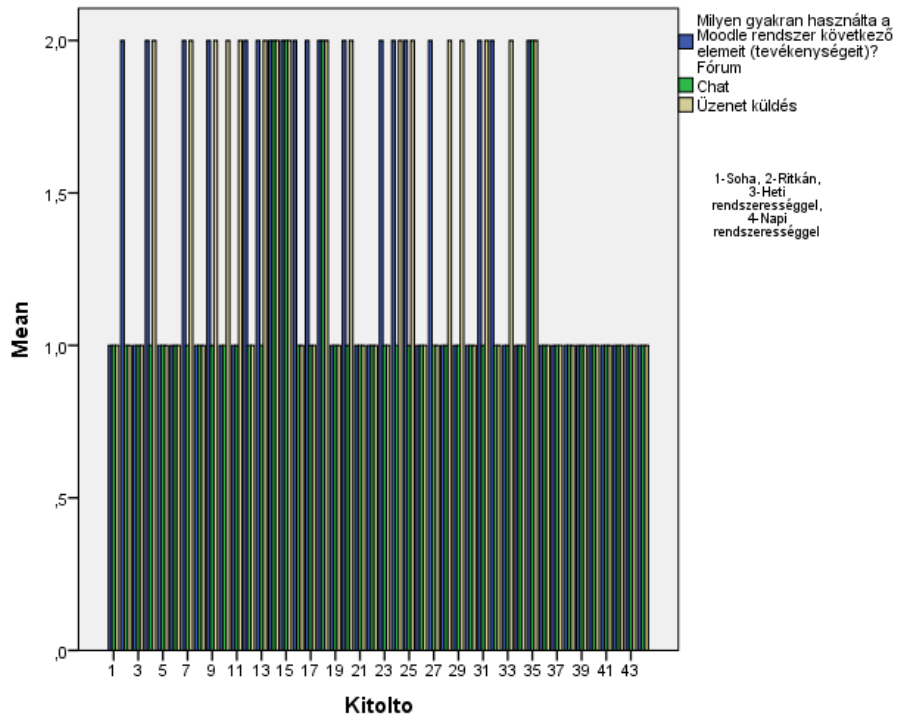
6. ábra. Mennyi időt töltött a kurzus tananyagának feldolgozásával a félév folyamán?

A tanulók összességében meg voltak elégedve a kurzuson belüli interakciókkal/kommunikációval ebben az esetben az 1 jelölte "Nagyon elégedetlen voltam" kijelentést, 2 "Elégedetlen voltam", 3 "Elégedett voltam", 4 "Elégedett voltam", az 5 pedig az "Egyéb" kategóriát ez a kódolás érvényes az egész kérdőívre.



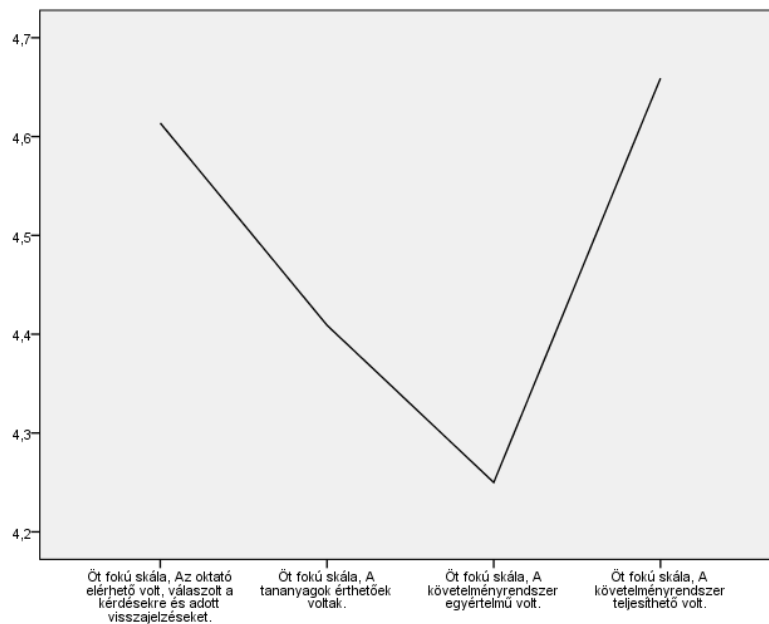
7. ábra. Kurzus interakciók

Mindezeket annak függvényében nyilatkozták, hogy a Moodle-n belül a Fórum, Chat és még az Üzenetküldés funkciót is ritkán használták. Ez az eredmény azt feltételezi, hogy a Moodle keretében a kurzusban nem alakult ki egy közösség és kapcsolatok.



8. ábra. Moodle interakció elemek

A tanulók egyértelműen meg voltak elégedve a kurzusokkal, mint azoknak tartalmával, az oktatók segítségével.



9. ábra. Elégedettségi eredmények

5. Konklúzió

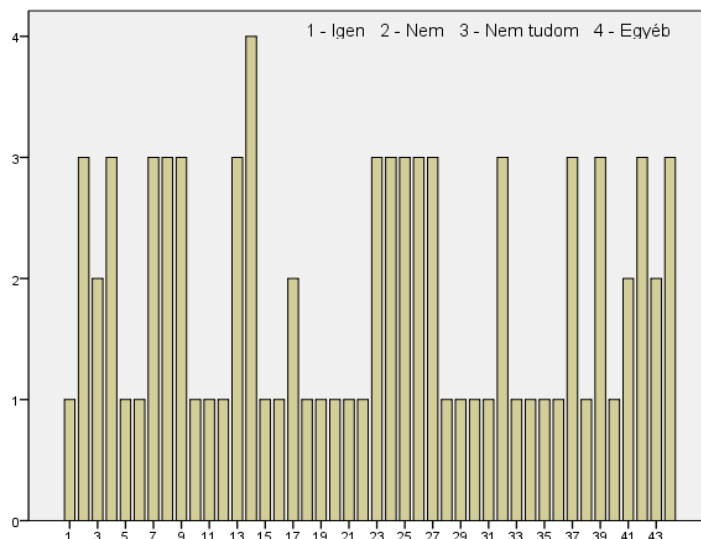
Természetesen a fentiek felvetnek kérdéseket is. Pótolhatja-e ez a hagyományos egyetemek kurzusait, oda ezentúl még inkább csak a papírért fogunk járni? Miként tudják az egyetemek a fizetős kurzusaikból finanszírozni ezeket az ingyeneseket? Miként lehet megtartani az amúgy is nehezen betartható etikai sztenderdeket, pl. hogy a kiadott feladatok megoldásánál ne másolják a diákok egymás anyagait?

A hivatalos és gazdasági kérdéseken kívül azonban pedagógiai, pszichológiai kérdések is felvethetők a MOOC-okkal kapcsolatban. Elsőként felmerül a kérdés, hogy nem túl információ-központú-e ez a fajta oktatási forma. Lehet-e készségeket fejleszteni, a gyakorlatban is hasznosítható tudást adni virtuálisan. Ezzel összefüggésben az is kérdéses, hogy a tanulás személyes részének kikapcsolása milyen előnyökkel és hátrányokkal jár.

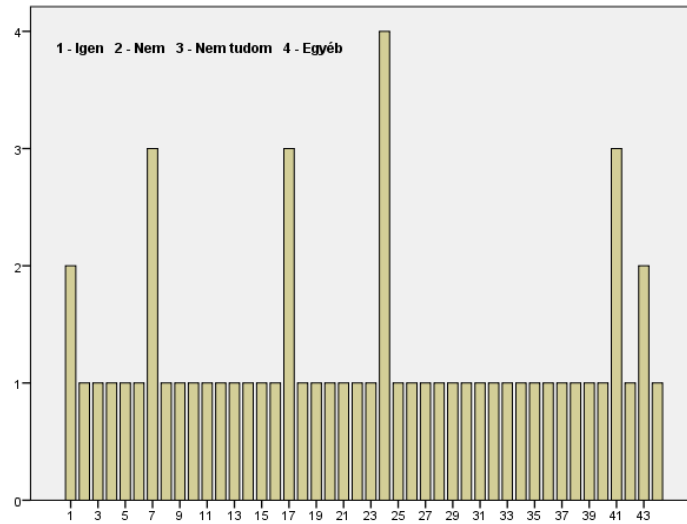
Az előnyök elvitathatatlanok: mindenki a saját tempójának, érdeklődésének és képességeinek megfelelően haladhat a kurzusokkal. Miközben tanul a földrajzi korlátok is eltűnnek: kapcsolatot is építhet hasonló érdeklődésű emberekkel a világ minden tájáról. Ugyanakkor hiányzik a tanulás személyes része, hiányzik az offline kooperáció, és a közös élmény.

Mindezek miatt talán nem várhatjuk, hogy az egyetemek falai a közeljövőben fognak leomlani, de a MOOC-ok érdekes kiegészítői lehetnek a formális keretek között zajló felsőoktatásnak.

A tanulók többsége pozitív élményekkel végezte a kurzusokat és, hogy minden nehézségünk ellenére valóban sikeren kell elkönyvelnünk ezt a próbálkozásunkat és tovább kell haladni az úton, melyen elindultunk mi sem bizonyítja jobban, mint az alábbi két eredmény.



10. ábra. Tervez-e, hogy részt venni más e-Learning kurzuson a következő évben?



11. ábra. Ajánlani fogja ezt a kurzust másoknak is?

Irodalomjegyzék

- [1] Benedek András, Csoma Gyula, Harangi László, Felnőttoktatási és -képzési lexikon (2002), 164 old.
- [2] Don Morrison ” E-learning Strategies: how to get implementation and delivery right first time” 2003 p.9
- [3] Péter Tóth (2012): Learning Strategies and Styles in Vocational Education, Acta Polytechnica Hungarica Volume 9 Issue Number 3 2012

E-learning a számítástudomány oktatásában

E-learning in the education of Computer Science

Koós Dániel^a, Várterész Magda^b

^aDebreceni Egyetem, Informatikai Tudományok Doktori Iskola
koos.daniel@inf.unideb.hu

^bDebreceni Egyetem, Informatikai Kar, Számítógéptudományi Tanszék
varteresz.magda@inf.unideb.hu

Absztrakt: Minden informatikus alap- és mesterszak képzési és kimenetei követelménye tartalmaz előírást számítástudományi (logika, formális nyelvek és automaták, mesterséges intelligencia, algoritmusok tervezése és elemzése, számításelmélet) ismeretek elsajátítására a szakképzettség szempontjából meghatározó ismeretkörként. Ugyanakkor a számítástudományi kurzusok tananyagának megértése és megtanulása gyakran okoz nehézséget a hallgatóknak. Ezekben a kurzusokban jelenleg még általában hagyományos frontális előadásokat és papír/ceruza megközelítésű gyakorlatokat tartunk. Alapvetően nem változtat a helyzeten az sem, hogy az órákon előre elkészített, szemléltető anyagokat vetítünk, és a tananyagot elektronikus formában elérhetővé tesszük. Pedig a kurzusokon elhangzó fogalmak, tételek, technikák, algoritmusok megértése és elsajátítása ügyes, dinamikus, interaktív segédeszközök segítségével lényegesen könnyebb lenne. Előadásunkban összegyűjtöttük tehát a számítástudomány oktatásában dokumentáltan pozitív eredménnyel alkalmazott szemléltető eszközöket, áttekintettük előnyeiket és számba vettük korlátaikat, továbbá megvizsgáltuk, hogyan lehetne ezeket tanóráinkon használhatóvá tenni. Sajnos, kevés olyan eszközzel találkoztunk, amely számunkra minden tekintetben megfelelő, testre szabható és paramétereztető (dinamikus) lett volna, és még kevesebb segítette a már megértett fogalmak és algoritmusok alkalmazásának begyakorlását. Így indokoltnak tartunk egy olyan online gyakorló rendszert megtervezni és elkészíteni, ami a számítástudományban felmerülő fogalmak elsajátítását nem csak ábrák, animációk és szöveges leírások formájában segíti, hanem interaktív és dinamikus: az egyes hallgatók számára különböző feladatokat generál és azok azonnali kiértékelésének felhasználásával rugalmasan alkalmazkodik a hallgató tudásához, személyre szabottan irányítva a gyakorlást.

Kulcsszavak: dinamikus e-learning, online gyakorló rendszer, SCORM szabvány, számítástudomány

Abstract: In all university-level educational program for IT specialists, the requirement of outputs prescribes acquisition of knowledge in computer science (logic, formal languages and automata, artificial intelligence, algorithms and his analysis, computation theory). However understanding and learning the study materials of computer science courses often cause difficulties for the students. Within these courses, we're still usually take traditional frontal lectures and paper/pencil approach exercises. Projected presentations are often used to get (and keep) students' attention. Course materials are also provided in an online accessible form, but it doesn't make learning easier for students. But understanding and mastering all these definitions, theorems, techniques, algorithms, which are being heard during the courses, can be easily done by far, using handy, dynamic, interactive tools. In this paper we present some of these interactive visual illustrative tools with documented positive results in education using in conjunction with other traditional teaching tools. Unfortunately only a few of these tools would be suitable for us, because most of them are difficult to customize and only a few of them help in practicing the definitions and algorithms presented in our courses. So that is why we think designing and constructing an online training system would be reasonable, which can help students mastering the definitions, techniques and algorithms of Computer Science using interactive and dynamic tools along with traditional teaching resources including pictures, animations, books and textual descriptions. The system could automatically generate different exercises for each student, and could adapt to the students knowledge with the ability of immediate exercise generation and evaluation, hereby personally directing the practice.

Keywords: dynamic e-learning, online exercise system, SCORM standard, computer science

1. Jelenlegi helyzet a DE IK-n a számítástudomány oktatásában

A 15/2006-os OM rendeletben az informatika képzési terület minden szakján és szintjén szerepelnek számítástudományi ismeretek a törzsanyag ismeretkörei között. Ezeket az ismereteket a Debreceni Egyetem Informatikai Karán *Az informatika logikai alapjai, Adatszerkezetek és algoritmusok, Automaták és formális nyelvek, A mesterséges intelligencia alapjai, Algoritmusok tervezése és elemzése, Számításelmélet és Számítástudomány* kurzusok keretein belül tanítjuk. A kurzusokon elsajátított ismeretek a képzés során nélkülözhetetlen alapot szolgáltatnak a szakmai törzsanyag információtechnológiai moduljai számára, továbbá szükségesek ahhoz, hogy a végzett informatikus az ICT szektorban betöltött munkakörben a megkövetelt készségekkel, képességekkel és tudással rendelkezzen. Ugyanakkor tapasztalataink azt mutatják, hogy a számítástudományi kurzusokon tárgyalt fogalmak, technikák és algoritmusok megértése, a különböző fogalmak közötti összefüggések felismerése gyakran nehézséget okoz a hallgatóinknak.

A probléma okai közül az egyik az lehet, hogy ezeken a kurzusokon már elsősorban – az egyetemi oktatásban hagyományosan szokásos – szimbolikus reprezentációt használunk, pedig a hallgatók fogalmi fejlődése ezen a tudásterületen még nem járta végig a *J. Bruner* amerikai pszichológus által leírt utat a cselekvéstől a vizualitáson át az elvont fogalmakig. Hozzájárulhat a nehézségekhez az is, hogy a mai egyetemi oktatási gyakorlat fő paradigmája az elméleti tárgyak esetében a prezentációs módszer (frontális tanári szóbeli előadás és kérdés, fő célja a tananyag megjelenítése). Alapelve így foglалható össze: a tanár tud valamit, amit a hallgatók nem tudnak – el kell mondani nekik. A prezentáció

- az átadandó ismeretekre koncentrálnak, nem foglalkoznak az ismeretszerzés folyamatával. A pedagógiai irodalomból jól ismert felfedezéses tanítási stratégia során viszont az oktató kérdéseket tesz fel, problémákat fogalmaz meg, rávéve a hallgatót az önálló, aktív munkára. Időigényes, de gondolkodásra ösztönöz.
- során nyilván nem világos az sem, hogy az egyes hallgatókban milyen ismeretszerzési folyamat zajlik le, alig van visszajelzés az előrehaladásáról. A közvetlen oktatási cél pontos leírásával és a cél elérését lehetővé tevő technológia kidolgozásával operáló adaptációs stratégia segíthet a problémán.
- közben a hallgatók nem tudnak egymástól tanulni, nem tanulják meg az együttműködést sem. A kooperációs paradigma alkalmazását javasolja a módszertan.
- gyakran szétválasztja az elméletet az alkalmazástól. A gyakorlatban összetartozó tudáselemek tárgyalására különböző kurzusokon kerül sor, így a hallgató fejében is különböző, nehezen összekapcsolódó sémákban reprezentálódnak. Segíthet a projekt módszer.

Karunkon ezek az alternatív tanítási stratégiák nyomokban persze jelen vannak ma is. Külön-külön pedig egyik sem lesz feltétlenül hatékonyabb, mint a prezentációs, főleg az időigényes alkalmazásuk miatt.

Tanszékünkön nemrég, szinte egy időben, több kollégában is megfogalmazódott az igény: használjunk a számítástudományi tantárgyak oktatásának megtámogatásához interaktív, algoritmus vizualizációs eszközöket is tartalmazó e-learning tananyagokat, amelyek segítségével a hallgatók tanórán és tanórán kívül is (akár egyéni ütemben is) tanulhatnak. Elvárásunk az volt, hogy az e-learning tananyagok vegyék figyelembe a nálunk tanított tananyagot, és lehetőleg a fenti gondokat is (legalább részben) orvosolják.

2. A témában elérhető e-learning rendszerek

Az információs technológia rohamos fejlődésével párhuzamosan az infokommunikációs eszközök az oktatásban is egyre nagyobb szerephez jutnak. A felsőoktatásban részt vevő és ezután belépő hallgatók számára ma már természetes, hogy a jegyzeteket és tankönyveket elektronikus formában letölthetik az internetről, a kurzusokkal kapcsolatos információkról, beadandó feladatokról valamely oktatásmenedzsment szoftver használatával kapnak értesítést, és e-learning tananyagokhoz is hozzáférhetnek.

Az e-learning tananyag a hallgató számára számítógépes hálózaton elérhető, tér- és időkorlátoktól függetlenül használható, pontosan megfogalmazott oktatási cél elérése érdekében létrehozott interaktív oktatási anyag, amely hatékony tanítási-tanulási módszerek segítségével megszervezi és ellenőrzi az ismeretszerzési folyamatot, lehetővé teszi az oktató-hallgató (esetleg hallgató-hallgató) kommunikációt. Animációk, videók, interaktív szimulációk, továbbá az önálló, otthoni gyakorlást és teljesítménymérést segítő, személyre szóló feladatok generálására szolgáló gyakorló rendszerek használatával lehetőség nyílik a hallgatók egyéni preferenciáit figyelembe vevő, személyre szóló oktatási környezet létrehozására. Általánosan elfogadott nézet, hogy a hagyományos prezentációs stratégiával történő oktatás és az e-learning eszközök jól végiggondolt ötvöztetésével a tanulás hatékonyságát gyorsan és eredményesen lehet növelni [1]. A kevert oktatás (*blended learning*) a programozásoktatás során már Karunkon is igazolta eredményességét (hiszen a gyakorlatokon elhangzó új ismeretek elsajátításához szükséges az otthoni gyakorlás, amelyet személyre szabott gyakorlatokkal és a (*ProgCont* [2] rendszeren keresztül) beküldött feladatok automatikus ellenőrzésével segítünk). Az alapozó elméleti ismeretek elsajátítására szolgáló számítástudományi kurzusainkon viszont még nem használunk ilyen eszközöket. Körbenéztünk, hogy megismerjük a témában elérhető különféle e-learning tananyagokat és az alkalmazásuk tapasztalatait: miként fejlődtek az évek során, hogyan alkalmazhatóak a számítástudományi képzésben és miként segíthetik a hallgatókat a tananyag gyors és alapos elsajátításában.

Algoritmusanímáció, programvizualizáció

Ronald Baecker úttörőnek számító *Sorting out sorting* (1981) című, különböző rendezőalgoritmusok működését bemutató, audiokommentárral ellátott kisfilmje [3] tekinthető az első olyan algoritmusvizualizációs eszköznek, amely hatékonyan segítette a programozásoktatást. Az oktatóvideó megjelenése után az algoritmusanímáció az algoritmusok tanításának hatékony eszközévé vált. Egy rövid, 30 perc hosszúságú videóval a készítők ugyanis egy harminc oldalnál is hosszabb írásos tananyagot tudtak könnyen érthető formában a hallgatóság elé tárni. A tapasztalatok, a hallgatói visszajelzések nagyon biztatóak voltak. Ilyen formában prezentálva a hallgatóság nem csak az algoritmusok működésének lényegét értette meg, hanem az egyes algoritmusok közötti hatékonyságbeli különbség is hamar szemmel láthatóvá vált – amit sem a szóbeli magyarázat, sem a statikus ábrák sorozata önmagában nem tett volna lehetővé.

Az oktatóvideó sikere számos oktató érdeklődését felkeltette, így 1981 és 90 között számos, az informatikai eszközök fejlődésével párhuzamosan egyre fejlettebb animáció-készítő eszköz született meg [4, 5]. Az első rendszerek, mint például a *BALSA* (1985), *Zeus* (1992), *TANGO* és *XTANGO* (1989), *SAMBA*, és *POLKA* (1990) az implementációs technológiájuk bonyolultsága miatt nem terjedtek el széles körben. Az első generációs

platformfüggő rendszereket az internet és a Java elterjedésével 1990 és 2000 között azonban olyan platformfüggetlen, böngészőből futtatható rendszerek váltották, mint például a *JSamba* (1990), a *JHAVE* (2000), az *ANIMAL* (2002), a *JAWAA* (2002) vagy a *Jeliot 3* (2004), amelyeket már sokkal szélesebb körben használhattak a fejlesztők különféle animációk készítésére. A második generációs szoftverek közös jellemzői, előnyei és korlátai:

- Egyszerű elemekből felépülő szkriptnyelven programozhatóak, ezáltal bárki által (akár a hallgatók által is) készíthető programvizualizáció, amely az adott eszköz által használt nyelv ismeretében könnyen automatizálható.
- Lehetővé teszik több vizuális elem egyidejű megjelenítését, így lehetővé válik több algoritmus működésének egy képernyőn történő összehasonlítása vagy akár párhuzamos algoritmusok vizualizációja is. Egy algoritmus szöveges leírása, pszeudokódja és a mögöttes adatszerkezetek párhuzamos megjelenítésével az egyes elemek közötti összefüggések vizsgálatára is módot kapunk – ma ez az egyik leggyakoribb reprezentációs forma.
- Számptalan adatszerkezet megjelenítését támogatják, így a legkülönbözőbb kurzusok (például: mesterséges intelligencia, adatszerkezetek és algoritmusok, automaták és formális nyelvek, kriptográfia) algoritmusai szemléltethetőek általuk.
- Kevés interakciót engednek meg, ezáltal kevésbé segítik az önálló tanulást. Az animáció alapvető paramétereinek (sebesség és lépésköz) beállítására a legtöbb rendszer lehetőséget biztosít, azonban az algoritmusok további paramétereinek módosítása nem vagy csak nagyon korlátozottan lehetséges. (Az *ANIMAL* szoftver esetén például megadható az algoritmus néhány bemeneti paramétere, viszont más változtatásra nincs lehetőségünk.)

A fesorolt algoritmusvizualizációs keretrendszerek közül az *ANIMAL* és a *JAWAA* szoftverek kiemelkedő népszerűsége tette szert, ezen alkalmazások által készített animációk széles gyűjteményéből válogatva számos olyanra bukkanhatunk, amellyel a különféle rendező, kereső és gráfalgoritmusok szemléltetése könnyen lehetséges, míg a *Jeliot 3* [6] szoftver az algoritmusok tervezését és elemzését könnyíti meg, teszi kézzel foghatóbbá. Utóbbi szoftverben egy Java kód futtatására és különböző futási paraméterek (adatszerkezetek, változók, hívási lánc) elemzésére nyílik lehetőség, amely komoly segítség a hallgatók számára az egyes algoritmusok működésének megértésében. Nagy hátránya, hogy a Java nyelv újabb elemeit (például a generikus típust és a kollekciókat) nem támogatja, így a programozás kurzusokon ismertett, Java 6 vagy újabb nyelven íródott kódok csak alapos refaktorálás után használhatóak.

Az eddig ismertett szoftverek túlnyomó része Java alapokra épült, amit a modern webes technológiák (HTML5, CSS3, JavaScript, jQuery) mára háttérbe szorítottak, hiszen a nagy népszerűségnek örvendő okoseszközök (tabletek, phabletek, okosmobilok, melyek a hallgatók körében nagyon népszerűek) nem, vagy csak nagyon körülményesen képesek a Java appletek futtatására. Az infokommunikációs készülékek teljesítménye azonban ma már lehetővé teszi az asztali alkalmazások teljesítményével egyenrangú, modern, webes technológiákra épülő webalkalmazások fejlesztését, amelyek bárhol, bármikor, bármilyen internetcsatlakozással rendelkező eszközről elérhetőek, legyen az egy Androidot futtató okostelefon, az Apple egyik táblagépe vagy egy Windows-t futtató notebook. A *Sorting Algorithm Animations* [7] oldalon például *Baecker* videójához hasonló módon, csak éppen böngészőből elérhető vizualizáció formában követhetjük nyomon az egyes algoritmusok működését. Hasonló célokat szolgál az *Algo-rhythmics* projekt [8, 9] is, ahol nem csak webes animáción, hanem az algoritmusok működését követő, magyar néptánclepeket használó mozgássorokat bemutató videókkal is szemléltetik az egyes algoritmusokat. Ugyanazon probléma különböző reprezentációira láthatunk itt példát, amelyek mindegyike hatékonyan segíti az algoritmusok működésének megértését. Általánosan elfogadott nézet, hogy nem az animáció formája, hanem az animációs

lépések időzítése, valamint az illusztrációt kiegészítő szóbeli magyarázat együtt segíti elő a jobb megértést [10, 11]. A problémák széles körére megoldást nyújtó, univerzális eszközt ugyan csak elvétve találunk (ezek közül is kiemelkedik a *Data Structure Visualization* [12] webes animációkészítő rendszer), azonban a különféle webes technológiákkal és keretrendszerekkel nagyon hamar elkészíthetjük saját algoritmusvizualizációnkat, sőt, akár ennél még fejlettebb, az önálló tanulást jobban segítő alkalmazások fejlesztésére is lehetőségünk van.

Gyakorló rendszerek

Az e-learning filozófia egyik kulcseleme a személyre szóló, önálló tanulás. Interaktív, automatikus visszajelzésre alkalmas, különböző feladatok generálására és automatikus ellenőrzésére képes rendszerek létrehozása az egyszerű programvizualizációs szemléltető eszközöknél lényegesen összetettebb feladat, hiszen az egyes feladatokat témaspecifikusan kell megtervezni. Ezért látunk egyre kevesebb univerzális eszközt, míg egy-egy konkrét problémátípus megértését elősegítő alkalmazásokat könnyen találhatunk.

Az *Algo-rythmics* weboldalán például a néptáncos videókon és az online algoritmusanimációkon túlmenően az oldalra látogató hallgatók lehetőséget kapnak az önellenőrzésre: a kiválasztott rendezőalgoritmust egy véletlenül generált tömb elemein kézzel végrehajtva gyakorolhatják az algoritmust. Hibás lépés esetén a rendszer felugró ablakban értesítést küld, és amennyiben elakadnak az algoritmus végrehajtásában, akkor egy kattintással hozzáférnek a továbbhaladást elősegítő, rövid, szöveges üzenetekhez.

A szakirodalomban gyakran esik szó a *PILOT* [13], a *TRAKLA2* és az *AnimalSense* [14] szoftverekről is. Amíg a *PILOT* és *TRAKLA2* szoftver az *Algo-rythmics* alkalmazásához hasonlóan az algoritmusok működésének manuális végrehajtását várja el a hallgatóktól, addig az *AnimalSense* ellenőrző kérdések segítségével segíti a hatékony, önálló tanulást. Akármelyik módszert is alkalmazzák, ezeknek az eszközöknek az adatszerkezetek és algoritmusok és a mesterséges intelligencia alapjainak oktatásánál hasznát vehetnék. Utóbbi tárgy oktatásában segíthet még az *AIspace* honlapon [15] található néhány eszköz [16] is, amely az alapvető gráfalgoritmusok, döntési fák, neurális hálózatok, kényszerkielégítési problémák és a bizonyításelméletben felmerülő problémák értelmezését könnyíthetik meg, míg a formális nyelvek és automaták kurzusokat a *JFLAP* [17, 18] szoftverhez hasonló alkalmazásokkal segíthetjük.

Sajnos az ismertett szoftverekben szereplő algoritmusok pszeudokódja és főleg a használt jelölésrendszer többnyire lényeges eltérést mutat a mi kurzusainkon alkalmazottaktól, így ezeknek az eszközöknek éles környezetben való alkalmazását alapos megfontolás és a szükséges módosítások elvégzése kell megelőzze. Amennyiben az animációkhoz tartozó forráskód is a birtokunkban van, akkor elég egyszerűen módosíthatjuk a kész animációkat, tisztán webes eszközökkel készített alkalmazások esetén pedig még ennél is könnyebb dolgunk lehet. Az interaktivitásban viszont a legtöbb eszköznek van hova fejlődnie, hiszen a személyre szóló, önálló tanulás támogatásához véletlen problémagenerálásra és automatikus feladatellenőrzésre van szükség. Erre azonban a bemutatott rendszerek nem, vagy csak igen korlátozottan képesek. Ugyanakkor ma már minden technológiai eszköz a rendelkezésünkre áll, amellyel ez a funkció hatékonyan implementálható.

3. Az informatika logikai alapjainak oktatását támogató rendszer fejlesztése

A logika oktatásában az e-learning eszközök használata még nem terjedt el, így a korábban bemutatott algoritmusvizualizációkkal és gyakorló rendszerekkel ellentétben ez a terület nem bővelkedik a hatékony oktatást ehhez hasonló módon támogató eszközökkel. A tematikánk anyagában mindössze az ítéletlogika néhány alapfogalmának (igazságtábla, jól formált formula, teljesen zárójelezett formula) gyakoroltatására szolgáló online eszköz és az elsőrendű logika néhány alapfogalmának megértését szolgáló gyakorló rendszer (*Tarski's World* [19], *FOLST* [20]) áll rendelkezésre. Ehhez az első félévben kötelező kurzushoz ítéljük szükségesnek elsőként a fogalmak pontos megértését, a termek és formulák ekvivalens átírását, szintaktikai és szemantikai vizsgálatát irányítva és ellenőrizve támogató interaktív saját gyakorló rendszer létrehozását.

A logikai gyakorló rendszerrel szemben a következők az elvárások:

- Legyen megadható a logikai nyelv ábécéje (logikai és nemlogikai jelek), a nyelvtani szabályok (valamint a logikai jelek precedenciája) – ezzel alkalmazkodik a rendszer a különböző kurzusokon használt különféle jelölésrendszerhez. (A gyakorló rendszer interaktív testreszabhatósága.)
- Legyen képes termek és formulák automatikus generálására (a különféle gyakorló feladatokhoz), ugyanakkor ezek kézi bevitelét is biztosítsa ellenőrzéssel (az otthoni önellenőrzéshez). Hibás formula megadása esetén jelezze (vizuálisan) a hiba helyét és (szöveges formában, a kapcsolódó definíciókra és tételekre való hivatkozás megjelölésével) az okát. (Visszacatolás.)
- A szintaxissal kapcsolatos következő fogalmakat gyakorló feladatok segítségével tisztázza: term (közvetlen) résztermje, formula (közvetlen) részformulája, összetettség, fő logikai összekötőjel, logikai jel hatásköre, változóelőfordulás és státusza (szabad/kötött), változóiban tisztaság. A kötött változót szabályosan átnevező, továbbá a formulát változóiban tiszta alakra hozó átalakításokat megvilágítsa és gyakoroltassa. Lehesse gyakorolni annak a kérdésnek az eldöntését, hogy két formula kongruens-e egymással. (Dinamikus e-learning megjelenése.)
- A szemantikával kapcsolatban első lépésben elégedjünk meg az igazságtábla-módszerrel, konjunktív, illetve diszjunktív normálformára hozással, az elsőrendű logikában pedig a prenexizálással, azaz ezeknek fogalmaknak és logikai feladatoknak a megértésével és gyakoroltatásával. (Itt is a dinamikus e-learning megjelenése.)
- Az oktató (bizonyos szabadságfokkal) válogathassa ki és rendezhesse sorba az egyes témákat. (A gyakorló rendszer interaktív testreszabhatósága.)
- A hallgató által megadott válaszokat a rendszer automatikusan értékelje ki, és az eredményről szöveges valamint grafikus formában a hallgatónak is adjon visszajelzést. (Visszacatolás.)
- A gyakorló feladatok a kurzusok tematikáját követve egymásra épüljenek és fokozatosan nehezedjenek. A rendszer legyen bizonyos mértékben adaptív: a hallgató és a rendszer közötti kommunikáció során a rendszer alkalmazkodjon a hallgató egyéni preferenciáihoz, és a tanulási folyamatban az előrehaladása üteméhez. Ez az igazodás a tanulás idejére, a résztémák kiválasztására és prezentálásának sorrendjére és feldolgozásának módszerére is vonatkozhat. (Adaptivitás, visszacsatolás)
- Tegye lehetővé az oktatóval és a kurzus hallgatóival való kapcsolattartást (nem csak email, hanem a népszerű közösségi oldalak integrálása révén).

- A rendszer bármely infokommunikációs eszközzel legyen kompatibilis (ezt modern webes technológiákra építkezve érhetjük el), ne csak nagy képernyőkön, de a mobilok kijelzőjén is legyen optimálisan használható (ezt rezponzív webdesign alkalmazásával valósítjuk meg).

A hallgatók egyéni fejlődését és motiválását elősegítendő a számítógépes játékokból merített ötleteket is használunk, hiszen a digitális generáció tagjai az ilyen formában eléjük tárt információkra nagyon fogékonyak [11, 21]. A hallgatók a megoldott feladatokkal pontokat gyűjtenek, amelyek a tanulmányokban az előrehaladáshoz lesznek szükségesek. Helytelen feladatmegoldás esetén ugyanakkor pontot veszítenek, amivel azt kockáztatják, hogy egy-egy tananyagrészt gyakorlását előlről kell kezdeniük. Egyes pontrendszer kialakításával biztosítható, hogy egy újabb tananyagrészt csak az előfeltételek alapos ismerete után sajátítható el.

A rendszer elkészítése során a legnagyobb gondossággal kell eljárni a különböző feladatok generálásakor és a válaszok feldolgozásakor. Mivel itt a papír alapú, előre elkészített tesztekkel ellentétben véletlenül generált tesztek alkalmazunk, így azok megtervezése sokkal alaposabb munkát igényel, mintha egy tanulásmenedzsment szoftverben (a SCORM szabványnak megfelelően) állítanánk össze a gyakorlásra vagy számonkéréshez használt tesztsort.

Az eredményeket folyamatosan naplózzuk, amelyet nem csak a feladatok generálására és statisztikai célokra használunk fel, hanem a hallgatók egyéni és csoportos teljesítményének mérését is ezzel oldjuk meg. Így a hallgatók között csapatverseny is kialakítható, valamint az egyéni teljesítmény is értékelhetővé válik. Az elért eredményekről a hallgató a közösségi oldalakon keresztül értesítést küldhet a társainak, a közösséggel való kapcsolata a rendszer bizonyos szakaszaiban aktív lehet, ami a netgeneráció tagjai számára fontos.

Irodalomjegyzék

- [1] Komenczi Bertalan (2004): Didaktika elektromagva? Az e-learning virtuális valóságai. *Új Pedagógiai Szemle*, **54**. 11. sz. 31–49.
- [2] Kádek Tamás, Kósa Márk, Pánovics János (2012): A ProgCont szoftverrel támogatott programozó versenyek tapasztalatai. In: *New Technologies in Science, Research and Education*, XXV. DIDMATTECH Konferencia, 2012, Révkomárom, Szlovákia, 152-157.
- [3] Baecker, R. (1998): Sorting Out Sorting: A Case Study of Software Visualization for Teaching Computer Science. In: John Stasko et. al. (szerk.): *Software Visualization: Programming as a Multimedia Experience*. MIT Press, Cambridge, 396–381.
- [4] Shaffer, C. A., Cooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S., Edwards, S. H. (2010): Algorithm visualization: The state of the field. *ACM Trans. Comput. Educ.* **10**, 3, Article 9 (August 2010), 22 pages. DOI=10.1145/1821996.1821997. URL: <http://doi.acm.org/10.1145/1821996.1821997>.
- [5] AlgoViz.org (2014). *The Algorithm Visualization Portal, Annotated Bibliography*. URL: <http://algoviz.org/biblio>
- [6] Bednarik, R., Moreno, A., Myller, N. (2005): Jeliot 3, an Extensible Tool for Program Visualization (Demo). In: *Proceedings of the Koli Calling 2005: 5th Annual Finnish / Baltic Sea Conference on Computer Science Education*. November 17-20, 2005.

- [7] Martin, D. R. (2007): *Sorting Algorithm Animations*. URL <http://www.sorting-algorithms.com/>
- [8] *Algo-rythmics - Technologically and artistically enhanced inter-cultural computer science education* (2014). URL: <http://algo-rythmics.ms.sapientia.ro/>
- [9] Kátai Zoltán, Tóth László: Technologically and artistically enhanced multi-sensory computer programming education. In: *Teaching and teacher education* **26** (2010), 244-251, ISSN: 0742-051X. DOI=:10.1016/j.tate.2009.04.012.
- [10] Mayer, R. E. (2001): A Cognitive Theory of Multimedia Learning. In: Richard E. Mayer (szerk.): *Multimedia Learning*. Cambridge University Press, Cambridge, 41–62.
- [11] Kulcsár Zsolt (2008): *Az integratív e-learning felé*. Kulcsár Zsolt, Budapest. URL: <http://mek.oszk.hu/06600/06695/06695.pdf>.
- [12] Galles, D. (2011): *Data Structure Visualizations*. URL: <http://www.cs.usfca.edu/~galles/visualization/>
- [13] Bridgeman, S., Goodrich, M. T., Kobourov, S. G., Tamassia, R. (2000): *PILOT: An interactive tool for learning and grading*. In: *SIGCSE 2000*, 139–143.
- [14] Rößling, G., Mihaylov, M., Saltmarsh, J. (2011): AnimalSense: Combining automated exercise evaluations with algorithm animations. In: *Proceedings of the 16th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, ITiCSE 2011*, Darmstadt, Germany, June 27-29, 2011. DOI: 10.1145/1999747.1999831.
- [15] *AIspace: Tools for Learning Artificial Intelligence* (2010). URL: <http://aispace.org/>
- [16] Knoll, B., Kisyński, J., Conati, C., Mackworth, A., Poole, D. (2008): *AIspace: Interactive Tools for Learning Artificial Intelligence*. In: *AAAI AI Education Colloquium*. Technical Report WS-0802, Menlo.
- [17] Rodger, S. H. (2011): *JFLAP* URL: <http://www.jflap.org/>
- [18] Rodger, S. H., Finley, T. W. (2006): *JFLAP: An Interactive Formal Languages and Automata Package*. Jones & Bartlett Publishers, Sudbury, MA. ISBN: 0763738344.
- [19] Barker-Plummer, D., Barwise, J., Etchemendy, J., Liu, A. (2008): *Tarski's World*. URL: <http://ggweb.gradgrinder.net/tarskisworld/>
- [20] Mauco, M. V., Maggiori, E., Gervasoni, L., Ferrante, E., L., Felice (2012): *FOLST: A Didactic Tool to Support First Order Logic Semantics Learning*.
- [21] *ClassCraft – Make learning an adventure* (2014). URL: <http://www.classcraft.com/gamification/>

Innovatív tanulásszervezés a Gábor Dénes Főiskolán

Innovative organized learning at Dennis Gabor College

Kupcsikné Fitus Iлона

Gábor Dénes Főiskola

kupcsik@gdf.hu

Absztrakt: Az ILIAS tananyagfejlesztő és szolgáltató keretrendszer által támogatott tanulási környezet a kontakt tevékenységhez illeszkedően számos lehetőséget kínál újszerű tanulásszervezési megoldások alkalmazására. Az alapképzésben három tárgyból (Problémamegoldás és Algoritmusok, Adatbázisok, Valószínűségszámítás és Matematikai Statisztika) tartok online vizsgafelkészítő konzultációkat, melyek hatékonysága mérhető. Ezen konzultációk felépítését és levezetését odáig fejlesztettem az évek során, hogy általuk egyértelműen lerövidíthetővé válik az önálló tanulásra fordított idő. Tapasztalataim alapján az is megállapítható, hogy amíg a távoktatásban tanulóknál eredményes az online csoporttevékenységek kommunikációs sajátosságainak kiaknázása, addig a nappali képzésben az általános alulmotiváltság miatt nem. Előadásomban kitérek a speciális konzultációk módszertani vonásaira és a motivációs hatások különböző fogadókészségére a kétféle képzési formában.

Kulcsszavak: tanulásszervezés, online-vizsgafelkészítő, ILIAS

Abstract: The study environment supported by the ILIAS learning management system complements in-person education by providing many opportunities to apply novel learning management solutions. I provide online preparation for exams in three BSc courses (Problem Solving and Algorithms, Databases, Probability Theory and Mathematical Statistics) with measurable effectiveness. Over the years, I developed the structure and management of these consultations to the point where the reduction of the time needed for independent learning is clearly seen. In my experience, distance learning students react positively to the unique communication properties of online group activity but the full-time students benefit less because of the general lack of motivation. In my presentation I will compare the didactic elements of the special consultations and the different effectiveness of motivational influences for the types of courses.

Keywords: learning management, online exam preparation, ILIAS

1. Bevezetés

Minden évben nő a betöltetlen informatikai álláshelyek száma az Informatikai Vállalkozások Szövetsége szerint. Ez az információ összecseng azzal, amit a GDF legutóbbi diplomaátadó ünnepségén mondott végzőseinknek Laufer Tamás, az IVSZ elnöke, a diplomaosztó díszvendége. A GDF megalakulásától kezdve tömegek számára kívánta elérhetővé tenni a felsőoktatást, tehát 22 év óta a hozzánk beiratkozottaknak lehetőséget adunk a tanulásra. De motiváció hiányában vagy az önálló tanulásra való alkalmatlanság miatt csak az ötödük szerezte meg eddig a diplomát.

A PISA2012-jelentés szerint a 15 évesek között tovább nőtt a funkcionális analfabéták száma, és ezen felül az élethosszig tartó tanulásra alkalmatlanok száma. Külön aggodalomra ad okot, hogy a magyar gyerekek több mint egyharmada olyan rossz teljesítményt ért el e téren, amely őket – ha így marad – gyakorlatilag kizárja a munkaerőpiacról, ugyanis nem rendelkeznek azokkal a minimális képességekkel sem, amelyek alkalmassá tennék őket bármilyen, a

betanított segéd munkától bonyolultabb feladat elvégzésére – olvasható több cikkben. Hogy mennyire nem tudják adaptálni a diákok a középiskolában tanultakat, az a felsőoktatási bemeneteli eredményeken évek óta látszik. Sajnálatos eredmény, hogy a tanárképzésbe sem sikerült tehetséges fiatalokat vonzani – mondta Jánosi Imre (ELTE TTK) a középiskolai oktatás eredményeiről egy felsőoktatási intézmény felmérései alapján.

Ennek okait én is felismerni vélem, szívesen vitatkozom azokról, de nekünk, a felsőoktatásban oktatóknak mégiscsak a felvettek kései felzárkóztatását kell megoldanunk hosszú évek óta. További nehézséget jelent, hogy az informatikai szakokra jelentkezőknek a számítógépes szakmákról nem reálisak az elképzeléseik. Megjegyzem, az 5. PISA-felmérés kiegészítő számítógépes mérést is végzett a digitális képességszint kiderítésére. Ráadásul informatikusképző helyként a gyors ismeretelavulás drámai következményeivel kell szembenéznünk és arra kell törekednünk, hogy a mostani tanuló adaptálódni tudjon a ma még nem is létező igényekhez. A most megtanítandó tartalom tehát másodlagos a tanulás képességéhez képest, előtérbe kerül a tanulásra, ismeretbefogadásra való képzés [3].

Magyarország első magánfőiskolájaként úttörő szerepet játszottunk a távoktatás módszertanának kifejlesztésében, ezzel együtt sokat teszünk az élethosszig tartó tanulásra való felkészítésért. Mi nem hagyhatunk fel a tanulásszervezés folyamatos megújításával.

2. Alapértelmezett tanulásszervezés

A kollégákkal el kellett fogadnunk, hogy az iskolánkba zömmel alacsony pontszámmal kerülnek be a hallgatók, de vannak közöttük igazán elszántak, akik igénylik a törődést a tanulás teljes folyamatában, és elvezethetők egy kiváló szakdolgozat megvédéséig. Bármely hallgatónkból válhat jó gyakorlati szakember, hisz minden segítséget megkapnak hozzá; nem kell tankönyvet beszerezniük, külső linkek alapján felkészülniük, mert komplex tananyagokkal látjuk el őket.

A távoktatásra alkalmas, didaktikai fogásokkal dúsított *e-tananyagaimat az elmélet – önellenőrzés – példatár – feladattár – fogalomtár struktúrában* készítettem mindig. 10 éve az ILIAS tananyagfejlesztő és szolgáltató keretrendszerben tartom azokat, ápolom a tantárgyaimat és menedzselem a kurzusaimat. Multimédiás segédletem a maga idejében úttörőnek számított; egy 1999-ben készített *SQL-oktatólemez*. A hőskorból megemlítem a 19 órás *oktatófilmet Adatbáziskezelésből*, ami a kiválóan felszerelt stúdióinkban készült. A mai online törekvések az oktatás evolúciójában (Khan Academy és MOOC-ok megjelenése) is azt bizonyítják, hogy nem jártam rossz úton a módszertanommal. Megjegyzem, a jövőkutatók újra az érthető oktatófilmek és a kiscsoportos tutorálás együttesét tartják a leghatékonyabb tanításnak.

Szakmai törekvéseimre visszatekintve azok valóban haladónak számítottak, amennyiben a közoktatásbeli gondok begyűrésére előre felkészültem. Kifejlesztettem egy logikai képességeket javító, a megfigyelt adatok rendszerezését körüljáró és a hatékony adatfeldolgozási szemléletet erősítő tananyagot, melyből Selmeci István kollégával 2007-ben *Problémamegoldás címmel* könyvet is írtunk. Az algoritmusépítés képessége ugyanis belépőként szolgál a későbbi szemeszterek szaktárgyainak elsajátításához. Valljuk, hogy a fogalmak szerepe minden diszciplínában elengedhetetlen és a Pólya-féle problémamegoldó gondolkodás nem csak a matematikai vagy programozási feladatok megoldására használható. Már akkor fontosnak tartottam a függvényfogalom több oldalról való megközelítését felhasználni valamint a véletlen jelenség kimeneteleinek és elemi eseményeinek megkülönböztetését hangsúlyozni, mert úgy tapasztaltam, hogy a hallgatók egyiket sem értik attól függetlenül, hogy hány éve érettségiztek. Ide vonatkozó megjegyzésem: az 5. PISA-felmérés kiemelten vizsgálta az adat és bizonytalanság tartalmi kategóriát 2012-ben.

2004 óta minden tárgyból a tutorálásra is kiválóan alkalmas *ILIAS rendszerben* egyengetem a hallgatóim felkészülését. Ez nagyon fontos egyrészt a munka és család mellett tanuló, távoktatásban résztvevő hallgatóknak, akik legalább motiváltak, de alacsony óraszámban kapják a kontakt foglalkozásokat, másrészt a nappali képzésben résztvevő hallgatóknak, mert nagy részük nem tanult meg tanulni 12 év alatt a közoktatásban. Az egyre több rossz szövegértésű hallgató szinte behozhatatlan hátránnyal indul neki a tanulmányoknak, mert a szövegértés képessége köztudottan függ a családi háttértől. (megjegyzés: A PISA-jelentés szerint Magyarországon az egyik legszorosabb a kapcsolat a háttérváltozók és a matematikateljesítmény között).

Az oktatáselmélet Comenius óta keresi a megnyugtató választ arra a kérdésre, hogy *mit és hogyan kell tanulni és tanítani*. Az akkreditált szakok tanterveinek tartalma és felépítése adott, a tananyagtartalmak optimalizálása nem lehetetlen az általunk szükségesnek vélt korszerűsítések mellett, de bizonyos alaptárgyakból, amiket az informatikai fejlődés nem is érint, zsúfolt hagyományos tematikát kell követnünk, és legfeljebb egy célravezetőbb sorrendiség, ami kivitelezhető. Tehát nagyon nehéz a tanulási kedvet megteremteni olyan tanulóknál, akik nagy része alkalmatlan a komplex feladatokból íratott és ritkán történő számonkérésre, mert megszokta az órai munkákra vagy házi feladatokra összegyűjtött kisjegyek befolyását. Sok hallgatónk bizik a számonkérés előtti utolsó pillanatban történő felkészülésben, aminek idejét rosszul méri fel, mert például informatikából bizonyára komolytalan volt a középiskolai megerősítés. Az olyan idősebbek, akik még poroszabb iskolába járhattak, már sokat felejtettek. Ezért a legváltozatosabb technikákkal és módszerekkel igyekszem leadni az anyagot, aktivizálni a hallgatókat a gyakorlatokon és nem csüggedni a rossz dolgozatok magas aránya miatt, hanem felismerni a bennük lévő helytelen okoskodást, mert akkor tehetek ellene. Örökös céloom elfogadtatni a kurzusaim hallgatóival, hogy a hordozható tudás, nem pedig a gyorsan amortizálódó eszközök ismerete a kulcs.

Mivel magam is azon nem pedagógiai végzettségű tanárok egyike vagyok, aki hajlamos a neveléskutatást áltudománynak tekinteni, mindig fenntartásokkal szemlélem a motiváltság szintjének fokozásáért vett lendületet a *neveléskutatók* részéről. Több esetben láthatjuk, hogy a modellalkotás próbája csak részletekre és nem a teljes tematikát felölelően történik, vagyis az eredményesség kimutatása csak szűk korlátok között érvényes. „A korszerű IKT eszközök megfelelő alkalmazása pozitívan hat a tanulók motivációs szintjére” című hipotézist rengeteg neveléstudományi kutatás támasztja alá valamekkora megbízhatósággal. Ugyanakkor keveset foglalkoznak azzal, hogy az új szükséges kompetenciák kifejlesztésének ígéretével az új környezet is kitermeli a mechanizmusok útján történő viselkedést, vagyis nő a valódi problémamegoldásra való alkalmatlanság.

25 éve még mást és máshogy kellett tanítani; manapság előfordul, hogy a tanári kompetenciák közül már a technológiai nagyobb súllyal szerepel, mint a tartalmi, és a pedagógiait erősen befolyásolja a technológiai. Érdeklődéssel olvastam arról, hogy a változó tanulói kompetenciák közé időszerűvé vált hozzávenni egy ún. computational thinking kompetenciát, és nem csak az informatika oktatás esetében. Ez olyan problémamegoldó gondolkodás a laikusok számára írt meghatározás szerint, ami számítógépes eszközökkel „megfogható”. Bevallom, számomra akkor is zavaró az elnevezés, amikor a komputációs szint mellett a fogalomba beleértik az algoritmikus szintet, hiszen az implementációs szint elhagyásával lehet csak igazán elrettentő algoritmusok szerint „dolgoztatni” egy számítógépet, legalábbis az informatika oktatás területén maradva. Fejlesztése kapcsán sajnos nem Balázs Béla nevét -, aki a matematikai programozott rendszer megalkotója, - hanem a külföldi példákat említik. A népszerű tanár „bemelegy” a tanulók vagy hallgatók kedvenc terébe és elhiszi, hogy ha a tanulók vagy hallgatók szeretnek tanulni, akkor az majd úgy marad akkor is, ha egy teljes kurzus anyagát kell velük elsajátíttatni. Ki kell mondani sajnos, hogy nem. Ráadásul a teljes tananyag korszerűsítése e-tanulási környezetre rengeteg fejlesztési órát és a szerzón kívül

kivitelezőt is igényel. Gyakran ezzel együtt a számonkérés mégsem lesz gépesíthető. Helyette a tanár gyakran az ún. játékosítás elemeivel próbálkozik (laposabb ellenőrzés, díjazások). Teszi azt vagy azért, mert muszáj élvezhetővé tenni az órát, vagy azért mert kényelemből elfogad és használ valamilyen pályázati nyertes alkalmazást. A legutóbbi PISA-sokk után a pedagógusoknak bizonyára újra kihirdették, hogy a tanárnak elsősorban szakembernek kell lennie, aki korunknak megfelelően és megfontoltan használja a kibővített tanulási környezetet, tehát digitális eszközöket használ, digitális tevékenységeket végez, de *értékkeremtés* céljából [6]. De ahogy józan becslésem szerint eddig is kiderültek az óriási különbségek a különböző középiskolákban lefektetett érték fogalmáról, továbbra is nőni fog a felzárkóztatásra szoruló gólyák száma, még informatikából is.

Véleményem szerint a felsőoktatásban is csak biztos tárgyi tudás mellett érdemes a hagyományostól eltérő tanulási terekből választani, ami ráadásul haszontalan a tanári visszacsatolás vagy a mediátori szerepvállalás nélkül. Például a hálózatszemléletű tanulás elsősorban az infokommunikációs technológia adta lehetőségeket figyelembe vevő tanulási módszer. Kovács Ilma - számunkra ismét megnyugtató módon - csak a gyakorlati alkalmazását javasolja. Szerinte *a mediatisztált hálózati tanulás* egy új ismeretszerzési mód. Az elektronikus tanulási környezetben dominál a kommunikáció központú szemlélet. A kommunikációs rendszer központjában a tanuló egyén helyezkedik el, de a kommunikáció új dimenziót kap a tanuló társakkal végzett munka során. *A legfontosabb rendszerszervező a kijelölt téma.* A tanár a mediátor. Továbbá az online oldal nyilvánosságra bátorít [2]. Ki kellett próbálnom...

3. Online vizsgafelkészítő konzultáció

Mivel hallgatóink nehezen birkóznak meg a tárgyaim vizsgakövetelményeivel (a matematikai diszciplínák közül a Valószínűségszámítás szinte mumusa a hallgatóknak, az Algoritmusok és az Adatbáziskezelés meggyőző gyakorlati alkalmazása pedig elengedhetetlen az informatikai szakokon), *minden vizsga előtt felkészítő konzultációt* tartottam, ahogy több kolléga is megteszi a GDF-n. *2011 óta az ILIAS-ban tartom, online.* Igazság szerint ráuntam a lefutott vizsgasoraim kidolgozására, amivel egy gyűjteményt bővítettem időszakosan, mert csak azokat lobogtatva közlekedtek a vizsgázóim, mintha versmondásból kellene jegyet szerezni. A szemeszterenként kb. 750 tantárgyfelvevőm tanulásának irányítása a sikeres vizsga reményében egyre reménytelenebb feladatnak tűnt és új tanítási-tanulási teret követelt. Általában a hallgatóim nagyobbik fele nappalis, akik 2-3 zárthelyit is írnak, és mindenki elővizsgázhat egyszer nálam, mire elkezdődik a 4-hetes vizsgaidőszak. Heti 1 vizsgaalimat biztosítok 100 főre mind a 3 tárgyamból. Nagyon sok időt töltök dolgozatok javításával, mert egyik tárgy sem számonkérhető tesztjelleggel. Tehát van mintám és lehetőségem megtapasztalni az új és divatos szokásokat a vizsgázók megoldásainak menetében. E téren különösen az egyetemista korú nappalis hallgatók kifogyhatatlanok a „rövid szemle után megszerzem és megosztom” Z generációs felhasználói szokás szerint.

A tantárgyi fórumokban hirdetem meg *az online felkészítőket* a számonkérés előtti kb. 4. nap estéjére, amiről automatikus mélből értesülhetnek az érintettek. A lefutott *vizsgasorok* most már 7 szemeszter óta *az online felkészítő fóruma* mellett gyűlnek; minimum 30 vizsgasor már fent van az egyes tárgyaimból, melyek legfrissebbikét a meghirdetéskor teszem fel, mint a soron következő *megvitatandó témát.* Az *összes téma minden hozzászólása* természetesen visszamenőleg is mind elérhető; szemeszterenként archív mappában.

Az eddig lezajlott 86 online konzultációm látogatottsága magas volt, és azok egyértelműen hasznosak a vizsgafelkészítés szempontjából. Sajnos, még mindig sok a passzív hallgató, akiket egyre nehezebb lesz eljuttatni a záróvizsgáig, kizárólag a saját negatív hozzáállásuk

miatt. Ebben az általam választott tanulási környezetben is jól látható, hogy ha hagynám, akkor a hallgatók a rögzült, de korokra jellemzően más-más szokások szerint *viselkednének*:

- A *kreatív „digitális bennszülött”* (az Apple Education portálja nevezte így el a web 2.0-t használó generációt) rákeres egy részletre és azonnal bemásolja a találatot, ami egy hasonló feladat megoldása. Nem az a baj, hogy erre figyelmeztetni kell, hanem az, hogy részpontokat remél hasonló esetben a vizsgán. A hallgatók zöme nem érti, hogy szigorú logikán alapuló megoldás menetéből miért nem elég, ha nála csak néhány paraméter vagy konstans nem jó, miért kevés a javító tanárnak a végeredmény vagy az, hogy elég hosszú a kidolgozás.
- A *lustábbak* -, akik megvárják a kreatívok első eseményeit – nem mind elemző típusok. A megvitatott feladatra sorozatban beírják a saját adataikkal ugyanazt a megoldást. Én tisztáztam, hogy időrendben követem a hozzászólásokat, és mindig az elsőként bejövő hibákat javítom ki részletesen. Sokan nem figyelnek a többiekre, így nem okulnak azok hibájából.
- Algoritmuspépítési készségük a *tapasztaltabbaknak*, leginkább a munkahellyel rendelkező távosoknak van. Ott arra kell figyelni, hogy sértődés ne essék, amikor ragaszkodnak a berögzült receptjeikhez. Büszke vagyok rá, hogy megnyílnak, megértenek és nem félnek digitális lábnyomot hagyni az én foglalkozásaimon, pedig többen a kollégáikkal vagy a főnökükkel vannak nálam.
- Vannak az igazi *harcosok*, akiken látszik, hogy a konzultációra felkészültek. Megszokták, hogy egyesével illik csatolni a kidolgozásaikat, mert akkor gyorsan tudok reagálni, és senki sem kezd másik feladatmegoldásba. (Érdekes módon adatbázist ők sem szívesen hoznak létre és töltenek fel tesztadatokkal, hogy legyen miben futtatni a lekérdezéseket; egymásra várnak vagy rám. Már következetesen kivárom; általában ugyanis párhuzamos foglalkozást tartok másik tárgyból.)
- Gyakran tapasztalom, hogy az *intuitívabb résztvevők* konkrét feladat megvitatása esetén is csak a szemrevételezhető szabályokat képesek felfedezni vagy inkább vélt szabályokat alkotnak szívesen a logikus gondolkodás helyett. De ott maradnak akkor is, ha meg kell őket állítani a sokadik következtetéseikben.
- Az *önálló tanulásra képes felnőtt* kiváló társaság. Ha nem is készült, mert épp hogy beesett a foglalkozás másik felére, képes az irányításom szerint lépésenként haladni és a többi hozzászólást is figyelni. Adaptív tanulási stílusa van. Nem ugrik előre, mert tudja, hogy a menetredek értelme van.
- Az *analitikus* gyakran nem szól hozzá a témához. Akad, aki másnap beírja, hol tévedtünk, vagy csak személyes méiben érdeklődik, ha még kérdése van. Többen kész megoldás-gyűjteményt készítenek a vizsga napjára.
- Van még az *önző*, aki azonnal másik feladatba kezd, ha elveszíti a fonalat, és követeli is, hogy neki reagáljak. Lepereg róla, amikor visszakérdezek: minek írunk addig lekérdezéseket, amíg nem tisztáztuk a kapcsolatokat; milyen valószínűségi változók szerepelnek a feladatban vagy melyik próbát kívánja használni, mielőtt a számításokat elvégezné; melyik adatból hány érték szerepel az a feldolgozandó bizonylaton vagy milyen adattípusú a függvényérték, amit felhasznál.

Az én további *szokásom*, hogy egy nehézkesen végigvezetett megoldásra további példát is megfogalmazok, de ha 3,5 óránál is jobban elhúzódik az ilyen csoportfoglalkozás, akkor másnap még visszatérek és reagálok. Ez leginkább Valószínűségyszámításból történik meg, ahol bizony a feladatokat beírása sem megy annyira gyorsan a sok képlet miatt. Adatbáziskezelésből a foglalkozás után a máshogy összeállított SQL-parancsaikat kívánják velem ellenőriztetni.

A leggyakrabban leírt szabadkozások: nem úgy értettem... nem olvastam el a feladat szövegét rendesen... nem úgy szokott lenni...

A legértelmetlenebb érvek, amikor a kardinális hibák kiűzése miatt pontlevonással „fenyegetőzöm” például egy nagyon rossz SQL-megoldás miatt: hisz működik... csak az összekapcsolás maradt le...

Korholásból csak ennyit engedek meg magamnak a fórumban: nehogy nálam bukjanak meg programozás alapjaiból meg elemi matematikából! Persze néha meg bátorítani kell őket azzal, hogy ezen a helyen ingyen van a hibázás. Egyébként egymás segítése meglepően óvatos hangnemben történik.

Kifejezetten *sikernek* könyvelhetem, amikor egy kiváló volt hallgató betér a foglalkozásra és segít. Nem beszélve arról, amikor adott esetben engem nem fél valaki kijavítani.

Az elképzelésem bevált: a később betérők is végignézhetik a csoportfoglalkozások produktumait; látják, mire nem kapnának pontot a dolgozatban, szóval tanulhatnak egymás hibáiból. Az online konzultációt nem látogatók - hasonló feladatsorban bízva a következő vizsgán - elkezdik beszerezni a jó hozzászólásokból összeállított végső dokumentumot. Baj akkor van, amikor annak használatára inkább az IKT eszközeivel készülnek fel a tartalom érdemi áttekintése helyett.

Évekkel ezelőtt hoztam *egy szigorú szabályt*: nem csatoljuk az összesített megoldásokat a fórum végén. Indokaim: a fórum célja a közös munka, a bátor kérdésfeltevés és próbálkozás, azaz a hibás elképzelések felvetése és kijavítása egyben. A hibás megoldások javításáról szóló hozzászólások elolvasása rengeteget segít egy később megtekintőnek. Ha ezek megszűnnének, és csak a végső kidolgozások kerülnének fel, akkor semmivel sem különbözne a régi típusú kidolgozott vizsgasorok gyűjteményétől. A legfontosabb érvem pedig az, hogy a foglalkozás alatt én is figyelem az ő hozzáállásukat a feladathoz, mert a hibáik gyökerének felismerése után tudok hatékonyan segíteni vagy elmagyarázni valamit újabb szemszögből.

Előfordult, hogy egy egyetemista korú nappalis hallgatóm az *egyik online szabadegyetem* videóit ajánlotta, pedig az e-tananyagomban is vannak videó-magyarzatok a megfelelő helyen és szöveggörnyezetben. Elgondolkodtam, miben állhat a Coursera, a Udacity vagy az edX sikere: a jól áttekinthető keretrendszer és a csupa videó-előadás mellette a bizonyítványok/érmek megszerzésének lehetőségében. Annak ellenére népszerűek az oldalak, hogy tananyag és önellenőrzés nincs és a követelmények tisztázatlanok. Az aktívan (tanár nélkül) fórumozók aránya kicsi (3%) és a végzettké is csak 8-10%. A nagy lemorzsolódásnak oka lehet, hogy a bemutatók és a számonkérés szintje gyakran távol esik egymástól.

Főiskolánkon a *Tantárgyi Útmutató* kiemelt dokumentum és van *e-tananyag* (amiben lehet keresni) az *önellenőrzés lehetőségével*. Az én tárgyaimból a tananyag része egy részletes megoldásokat tartalmazó *példatár* és a zárthelyi dolgozatokra illeszkedő *feladattár* is. A *fórumban* pedig véges időben szoktam reagálni a hozzám forduló szakmai vagy technikai kérdéseire és egyéb felvetéseire. A keretrendszerben futtatható *tesztek* csak az alkalmazások oldaláról kiemelt fogalmak megértését segítik; a számonkérés ugyanis nem tesztjellegű.

De mindez a nagyobb létszámú és alulmotivált nappalis hallgatók meg a munkahelyükről egyre nehezebben szabaduló, éppen ezért az önálló tanulásra egyre kevesebbet fordító és a hétvégi konzultációkat is esetenként kihagyó távos hallgatók számára *nem bizonyult elégnek*; még több tantárgy-újrafelvételes lett 2 tárgyból, ráadásul sokuknak ez már megfontolásra adott okot a főiskola folytatását illetően.

Az egyes online konzultációimon (egy alkalom egy téma) átlagosan 69 hozzászólás születik, aminek a felét mediátorszerepemnél fogva általában én írom. Az aktívak a bent lévők felénél is kevesebben vannak, de a helyben megfigyelők meg a később betérő olvasgatók miatt a megtekintések száma eléri a több százat. Nem csak a későbbi megoldások vagy javítások lereagálása miatt térek vissza még én is, hanem megfigyelem: még mit értenek félre a feladatmegoldóim. Az est hőse címet is megállapítom, ha valaki(k) nagyon megérdemli(k); de

feltételhez kötöm a díjazást: a plusz pontot csak az elégtelennél jobb dolgozat pontjaihoz adom hozzá.

Az *online konzultációk látogatása* nem mindenre jellemző egyik tárgyból sem. Általában alkalmanként 8-10 fő az *aktív résztvevő* és 4-szer annyi a *passzív résztvevő*, ami az utolsó vizsgaidőszakban 19%-a volt a tantárgyfelvevőknek. A *nem mutatkozók* száma (egyszer sem vizsgáztak és nem is felmentettek) 31%-ot tett ki, így a *kurzus szempontjából megfigyelőnek nevezett* csoport 50%. A *konzultáció szempontjából* ez utóbbi halmazba esnek a későbbi *megtekintők*, de bizonyára annál szűkebb halmazt alkotnak. Hinni szeretném, hogy az egyébként magas számú látogatások mögött sok megfigyelő is van még akkor is, ha a volt résztvevők velem együtt bizonyára többször visszatérnek utólagos megtekintésre. Kérdőíves felmérésre nincs szükségem, célom tisztán a vizsga sikerének növelése.

Azt tudtuk, hogy a *kompetenciák fejlesztése* a tanulási formák és módszerek sokféleségét követelik meg. A pusztán kognitív ismeretek mérése helyett a kezdetektől fogva a *kompetenciákat mértem* mindhárom tárgyból. Vizsgasoraim tartalma:

- *Adatbáziskezelésből* mindig új konkrét adatszerkezetre vonatkozó fejlesztői megvalósítások,
- *Problémamegoldásból* összetett logikai feladvány, megfigyelt értékek rendszerezése és konkrét adatfeldolgozós algoritmusok készítése,
- *Valószínűségszámításból* konkrét, paraméter nélküli szöveges feladatok ill. alkalmazott statisztikai feladatok képletgyűjtemény használatával.

Azon a gyakorlaton nem változtatok, hogy a számonkérés legfeljebb 20%-ban lehessen független a dolgozatírástól (szorgalmi feladatra vagy órai munkára).

A modern tanuláspszichológia szerint az ismeretek akkor maradnak tartósak, ha funkcionálisan beágyazódnak a szakmai viselkedésbe, lényeges és hiteles összefüggésekbe kapcsolódnak a képességekkel és az értékrenddel. Ezt nevezzük *kompetenciaelvű tanulásnak*. Lényegében a „tudni, hogy mit” központúságtól a „tudni, hogy hogyan” tanulási cél felé való elmozdulás jellemzi. A tudás szükséges, de nem elégséges feltétele a kompetenciáknak [3].

Hallgatói visszajelzések szerint pontosan ez a legnehezebb a dolgozataimban -, ami egyúttal a leghasznosabb az álláskeresőkre való felkészülés miatt -, hogy nem az ismeretekből kell számot adniuk, hanem az ismeretek alkalmazásából. Olyannyira tartom a dolgozataim ezen jellegét, hogy a 60 perces vizsgán engedem használni az e-tananyagot és segédleteket. A vizsgasorok éppen ezért mindig mások, amit kisebb változtatásokkal is elérhetek, viszont pontosan az ilyen „hasonló” vizsgasorok sikerülnek a legrosszabbul azok miatt, akik nem a problémamegoldási képességeiket használják, hanem egy hasonló tartalom másolásával élnek. Világos, hogy az ember a napi rutin miatt már a dolgai 90%-át mechanizmusok szerint végzi, de egy vizsgázó nem tehet úgy.

Mivel az *utolsó vizsgaidőszak eredményei* sokkal elkeserítőbbek voltak az előző évekenél a vizsgaidőszak közepén mindhárom tantárgyamból *új ötletet vettem be*: vizsgafeladatok hangalámondásos/vetített magyarázatainak készítését.

4. „Nekem való magyarázat”

Ahogy hatalmas információmennyiséget sem vagyunk képesek áttekinteni, véleményem szerint a hallgató sem képes hagyományos eszközökkel jól kezelni a hatalmas tananyagot. Újabb segédletekkel ugyan elláthatom, de a számonkérés magasságában segítenem kell őt a *megfelelő információ kiválasztásában*. Ez nem azt jelenti, mintha a tananyagom valamelyike felesleges fogalmakat vagy levezetéseket tartalmazna, hanem azt, hogy a vizsgák előtt a hallgató számára *fókuszba kell hozni a kiemelt témák mindegyikét* vizsgafeladatként

szerepeltetve, több lehetséges kérdéssel. Amiatt, hogy a saját tempójukban haladhassanak benne, a legalkalmasabb a részletes megoldás felvétele hangalámondással.

Fentebb említettem, hogy az online konzultációk fórumaiból gyűjtöm össze a hallgatói félreértéseket, a nehézségeket és a hitetlenkedéseket. Meggyőződésem, hogy a legnagyobb hiányosságok bármelyike visszavezethető hallgatóink esetében a középiskolai számonkérések alkalmatlanságára, miután maga az értékelő megfelelőnek tartotta a szövegértés nélkülözését és a matematikai alapfogalmak motorikus használatát. Tehát a dolgozatokban tapasztalt hibákra és a vizsgafelkészítő konzultációkon felismert tévedésekre alapozva készítettem el a 2 utolsó vizsga előtt a *nekik való magyarázatokat*, amiket a vizsgasorok mellett tartok az online konzultációk terében.

Adatbázisok: hangalámondásos felvételeket készítettem mind az adatbázis értelmezésére, mind az alapismereti kérdések megválaszolására, és csatoltam a konkrét adatbázist is a tesztadatokkal meg a lekérdezések megvalósításával. A nyilvántartás leírását félmondatonként kiderítettem a megadott adatbázisban. A kapcsolatokat a létrehozott kapcsolatok fölött magyaráztam, a beépíthető megszorításokat teszteltem, az eseményeket illusztráló mintasorokat bemutatásuk előtt számba vettem, a tervezési feladatot vagy a kérdéses függőséget nem az elméleti fogalmakkal indokoltam, hanem azok célja szerint magyaráztam. A lekérdező parancsok összeállítását megterveztem és lépésenként futtatva illusztráltam: mi a főtábla, kell-e további összekapcsolás, mire szűrünk, a feltétel összetett-e; a csoportosítást 2 lépésben, az újabb szűrést vagy a több lépésben megoldható lekérdezést illusztráltam, mintha a végrehajtás interpreter módban történne meg. *Erőltettem, hogy lássák:* az állítmány elárulja, melyik a tények táblája, amin a feldolgozás ciklusa halad, a kérdőszavak pedig pontosan jelzik, miket kell megjelenítenünk. A lekérdezések módosított verzióban is elkészültek, mert gyakran tapasztalom, hogy a vizsgázó kérdés nélkül is akar törzsadatokkal válaszolni.

Problémamegoldás és Algoritmusok: A logikai feltételek kiértékelése előtt tisztáztam a feladatban előforduló változók adattípusát és az előforduló függvények visszaadott értékét. A magyarázataimban részletes volt a beágyazott és az összesítő függvények kiértékelése; a rendezési lehetőség ellenőrzése és a null-érték kezelése nem maradt el; a származtatott logikai művelet alkalmazása célirányos volt, a formális felírást indokoltam. A véletlen jelenségben a megfigyelés tárgyát pontosan azonosítottam, adattípusát és a lehetséges értékeit megállapítottam, majd minden értékre több elemi eseményt is felsoroltam. A hagyományos algoritmus megírása a programnyelv-független pseudokódban történik, és mindig a feladatban jól meghatározott bizonylatok kupacára. Ez a legtrikábban megoldott és jól megoldott feladat a vizsgázók körében. A kupac előkészítettségére és a pozicionálásra figyeltem, a klasszikus adatfeldolgozás sémáját követtem; a tesztelést rájuk bízta. Sűrűn helyeztem el kommenteket és a nagyvonalútól a részletesig írtam meg a pseudokódokat; ráadásul több változatban, mert a hallgatók gyakran választanak kevésbé optimális algoritmust a kupac feldolgozására. A szabványos lekérdező mondat megfogalmazása most 3 adattáblás szerkezetre vonatkozott, amely a tananyagban ismertett szerkezetek egyike lehet az ilyen vizsgán. Ezzel a feladattípussal az Adatbázisok kurzust készítem elő. A magyarázatok mellé konkrét adatbázist is csatoltam a mintasorok gyártásának megkönnyítésére. Aki megpróbálkozik a meglévő SQL-tudása birtokában válaszolni, szintén tesztelheti, mennyire stabil a tudása.

Valószínűségszámítás és matematikai statisztika: a vizsgázónak a legnagyobb gondot mindig az okozza, hogy a szöveges feladatban megállapítsa a valószínűségi változó vagy változók eloszlását. A kis lépésekből felépített vetíthető magyarázatban mindig visszakérdeztem a megfigyelés tárgyára, hogy ne keverjük össze a Poisson és exponenciális, vagy exponenciális és binomiális vagy 2 független változó létezése miatt a feladat állításaiban és kérdéseiben szereplő változókat. A feltételes valószínűségek ábrázolására fa-szerkezetű gráfot ajánlottam, amiből a szorzási tétel, a teljes valószínűség tétele és a Bayes-tétel alkalmazása könnyen leolvasható. Sajnálatos tény, hogy egy feltételes valószínűség kiszámítására vonatkozó

mondatban sokan nem tudják megállapítani, mi az állítmány, vagyis melyik eseménynek melyik másik esemény feltétele mellett kell venni a valószínűségét. A legegyszerűbb hipergeometrikus valószínűség kiszámításánál nagyon fontos a megfigyelés szempontjából alkalmas teljes rendszer megállapítása. A normális eloszlás standardizálása és a táblázatbeli értékek kikeresése a levétített lépések után csak azoknak lehet gond, akik még a sűrűségfüggvényt is összekeverik az eloszlásfüggvénnyel. Eseményalgebrai illusztrációkat ajánlottam a 2 dimenziós diszkrét eloszlás megadásához. Minden vizsgasor tartalmaz intervallumbecslést vagy hipotézisvizsgálatot, esetleg lineáris regressziót a matematikai statisztika részről. Az első 2 feladattípus kis vagy nagyminta esetén ugyanolyan kezdő számításokat igényel, de a további lépések sincsenek távol egymástól. A kötelező lépéseket összevettem. A lineáris regresszió mindennél jobban megy a hallgatóknak, még centrálózással is.

A visszajelzések nagyon jók, a fórumban néhányan közzétették a *véleményüket* a nekik való magyarázatokról:

- nagyon hasznos, mert részletes
- jók, mert rövidek és érthetőek
- saját tempójukban tudtak haladni
- nem kell szelektálni a helytelen válaszokat, mint az online konzultáció végén tették
- több érdekes változat készült az algoritmusokra, lekérdezésekre
- sokat segít a valószínűség kiszámításának menetében

Néhányan hozzátették, hogy csak abból készültek; mindkét hibámat észrevették és figyelmeztettek rá; a felvételek minőségét nem kifogásolták.

A szóbeli vélemények alapján is elmondható, hogy a feladatok értelmezésében és az algoritmikus gondolkodásban előbbre viszi a vizsgára készült a neki való magyarázat. Ez viszont felvetheti azt a kérdést, hogy akkor miért nem áll csak ilyenekből az egész tananyag. Hát azért, mert akkor az lenne visszataszítóan hosszú. Így a tanár a hangulatától függően helyezi el a kiemelt témákat valamilyen szövegkörnyezetbe, amire majd ráhangolódik a vizsgázó. Kérdőíves felmérést nem végeztem, mert a *vizsgaeredmények szignifikáns javulásban bíztam*. Annak ellenére, hogy bizonyos hallgatók klasszisokkal jobb jegyet szereztek a következő alkalommal, a 2 utolsó vizsga eredményei egyik tárgyból sem hozták a remélt eltérést. Persze számításba kellene vennem, hogy a vizsgaidőszak végén nagy létszámban jelennek meg a szerencsét próbálók meg a kapkodók, akik csak nem akartak lemaradni az utolsó lehetőségről, de nem készültek vagy sok elmaradásuk lévén alig készültek rá. Nélkülük biztatóbbak lennének az eredmények.

Az átlagok tekintetében *néhány megállapítást* mégis tehetek:

- Gazdasági meg mérnökinformatikusok tömege mellett ebben a félévben kevés műszaki menedzser meg gazdálkodás és menedzsment szakos is tanulta a tárgyaimat a Problémamegoldás kivételével; vizsgázási szokásaik azonosnak tekinthetők az informatikus szakokéval.
- A Valószínűségszámítást (VA) vették fel újra a legtöbben, de összesük 39%-a egyszer sem vizsgázott. Mind a nappali (n) mind a távos (t) képzésen ebből lett a legtöbb elégtelenre lezárt hallgató (50%).
- Problémamegoldásból (PR) lett a legkevesebb bukás, Adatbázisokból (AB) már csak 35%, de ebből van a legtávolabb egymástól a nappali és távos képzési forma.
- Mindhárom tárgyból a távosok teljesítettek jobban, de a végső elégtelenek aránya szerint a következő lett a rangsor: ABt, PRt, PRn, ABn, VAAt, VAn.
- Nem meglepő módon a megajánlott jegyek illetve a távos elővizsgajegyek átlaga a legjobb minden tárgyból és képzési formában az összes alkalom (további 4 vizsga) vonatkozásában.

- A vizsgaidőszak 2. felében még nem volt érezhető a magyarázó segédletek hatása. Ha egy pillanatra megengedem magamnak, hogy csak a sikeres vizsgajegyeket tekintsem, akkor azt látom, hogy a távosok 1 egész osztállyal jobb átlagot értek el az utolsó vizsgán 2 tárgyból, csak Adatbázisokból nem; ennek oka lehet a kevés óraszámú gyakorlat. A nappalisoknál viszont csak Adatbázisokból volt alig érezhető javulás a 4 vizsgából álló sorozatban.

Mindezeket összevetve kijelentem, hogy a részletes magyarázatok felvételeinek készítését *folytatni kívánom*. El szeretném érni, hogy Problémamegoldásból ne törődjenek bele oly sokan az elégséges eredménybe, amit elsőre sikerült szerezniük, amikor maguk is jobbra számítottak, mert abból a tárgyból tudnak a legtöbbet profitálni. Érdemi visszacsatolásképpen továbbra is fel fogom dolgozni a hallgatóim szokásait, hibáit és konvencióit, amiket a megoldásaikban tapasztaltam. A tanulni vágyók egyszerűen megérdemlik, a többiek pedig még rászokhatnak a használatukra, mert a közeljövőben fel fogom ajánlani a magyarázó felvételek közös munkában történő előállítását is.

5. Összegzés

„Az oktatási rendszereken belül a hangsúlynak el kell tolódnia. A jövőben nem annyira kiképzett embereket kell az oktatásnak produkálnia, mint inkább képezhető embereket, akik képesek tanulni, és hatékonyan alkalmazkodni egész életükben a folytonosan változó környezethez. De ha az oktatási rendszer maga nem alkalmazkodik a környezeti körülményekhez, hogyan várható, hogy alkalmazkodásra képes embereket produkáljon?” - írja Philip H. Coombs máig jelentős és híres könyvében [4].

A Gábor Dénes Főiskola alkalmazkodik a munkaerőpiac elvárásaihoz, korszerű módot és környezetet biztosít a tanulásra és a csoportmunkára. Előadásomban bemutattam, hogy az e-tananyagok mellett eddig 7 szemeszteren keresztül adtam újabb lehetőségeket a tanulási kultúránk bővítésére, ami által alkalmassá válunk az élethosszig tartó tanulásra.

Sok végzett hallgatónk dolgozik vezetőként. Véleményük egybecseng a bevezetőben említett kijelentéssel: nincs elég megfelelő kompetenciákkal rendelkező informatikus. Egy munkavállaló számára megfelelő kompetenciákkal rendelkezni annyit jelent, hogy képes a különböző cégek olykor gyorsan változó követelményeinek megfelelni. A munkaadók belefáradtak az álláskeresők interjúztatásába és az állásbörzéről hozott szakmai életrajzok nézegetésébe. A reményteljes alanyok eszközhasználati képességeinek mindig utánanéznek, mert a bemutatott tanúsítványokban nem bíznak. Szerintem pont elég információval szolgálhat adott esetben bármely hallgatónk digitális lábnyoma például az online konzultációim fórumában.

6. Köszönetnyilvánítás

Köszönöm Littvay László kollégámnak az önszerveződő tanulási tér biztosítását az ILIAS-ban és György Lászlónak a további hallgatói adatokat a Neptunból. Karácsony Péternek, az Adatbázis szerverek vezetőtanárának a kezdetektől fogva hálás vagyok a laborok megtartásában és a vizsgáztatásban nyújtott folyamatos segítségért. Az online konzultációkat látogató hallgatóimnak pedig megköszönöm az együtt töltött izgalmas estéket, amivel a segítségüket célzó segédletek készítésére inspiráltak.

Irodalomjegyzék

- [1] PISA2012 Összefoglaló jelentés, Oktatási Hivatal, Budapest (2013)
http://www.oktatas.hu/pub_bin/dload/kozoktatas/nemzetkozi_meresek/pisa/pisa2012_ossz_efoglalo_jelentes.pdf
- [2] Kovács Ilma: Esettanulmány a tanulás új módjáról 2011-ben,
<http://www.oktatas-informatika.hu/2011/12/kovacs-ilma-esettanulmany-a-tanulas-uj-modjarol-2011-ben/>
- [3] Benedek András: Szakképzés-pedagógia, Typotex Kiadó (2006) 65-75.
- [4] Philip H. Coombs: Az oktatás világválsága Rendszerelemzés, Tankönyvkiadó(1971)
- [5] Bessenyei István: Tanulás és tanítás az információs társadalomban,
http://www.ittk.hu/netis/doc/ISCB_hun/12_Bessenyei_eOktatas.pdf
- [6] Ollé János, Lévai Dóra, Domonkos Katalin, Szabó Orsi, Papp-Danka Adrienn, Czirfusz Dóra, Habók Lilla, Tóth Renáta, Takács Anita, Dobó István: [Digitális állampolgárság az információs társadalomban](#). Budapest, Eötvös Kiadó (2014)

Nyílt online tömegképzés módszertani kérdései

Didactic Issues of Massive Open Online Courses

Kusper Gábor

Eszterházy Károly Főiskola, Matematikai és Informatikai Intézet

gkusper@aries.ektf.hu

Absztrakt: A világon egyre inkább elterjednek a MOOC (Massive Open Online Course) megoldások. A MOOC kifejezés általunk használt fordítása: Nyílt online tömegkurzus. Ezek a kurzusok bárki számára elérhetők. A hallgatók megtekinthetik a felvett előadásokat, megkapják a segédanyagokat, sőt még beadandó feladatokat is, amiket kiértékelve kapnak vissza. Majdnem mindig elérhető egy fórum is az órához, ahol a tapasztalatokat lehet kicserélni, kérdéseket feltenni. Korlátozott módon előre magadott időben az előadást tartó személlyel beszélgetni is lehet. Ezek a kurzusok úgynevezett freemium (ingyenes – prémium) üzleti modellel működnek, azaz maga a kurzus teljesen ingyenes, de a prémium szolgáltatásokért fizetni kell. Az Európai Unió belül az Open Education Europa nevű portál fogja össze a nyílt online tömegkurzusokat. Ezen egyelőre csak a Miskolci Egyetem néhány kurzusa érhető el vendégként, regisztrálni nem sikerült a kurzusokra. Ebben a cikkben áttekintjük a távoktatás keretein túlmutató, nyílt online tömegkurzusokkal kapcsolatos módszertani kérdéseket. Ilyen kérdés például, hogy hány perces legyen egy videó? A nemzetközi irodalom szerint egy átlagos hallgató egy 10-15 perces videó első 4 percét tekinti meg, ezért elfogadott a 6-7 perces videók használata. Nagyon fontos a hallgatók motiválása, hiszen a nagy tömegeből csak kevés igazán motivált hallgató lesz. A videók elkészítésénél tanácsos a klasszikus motiváció – ismeretátadás – példa hármasra építeni. A legegyszerűbb a nehéz példa – könnyű példa motivációt használni, ami szerint először mutatunk egy példát, ami a hallgatóság számára nehéz, majd megmutatjuk, hogy általában az ilyen típusú problémákat hogyan kell megoldani, végül megoldjuk a konkrét példát, ami immáron egyszerű. A videók mellett mindig biztosítani kell a szkriptet is, hiszen a videón beszélő tanárt néha nehéz érteni. A tananyagnak hangsúlyoznia kell, hogy ez csak egy segédeszköz, pl. a programozás videó megtekintése után senki nem lesz programozó, csak ha sok programot verejtékes munkával elkészít, azaz gyakorol. Ezért a videók mellett kellene példatárak is, illetve nagyon részletesen kidolgozott példák, amik alapján a hallgatók gyakorolni tudnak. Ebben a cikkben ilyen és ehhez hasonló módszertani kérdéseket tárgyalunk.

Kulcsszavak: MOOC, nyílt online tömegkurzus, módszertan

Abstract: One can realize that there are more and more MOOC (Massive Open Online Course) solutions. These courses are open for everyone. The students can watch videos of the course, they gain access to the teaching material, they get tests, and feedback on these tests. For each course there is a discussion forum, where students can post questions and ideas about the course. On a restricted base students can also chat with the lecturer. The business model of MOOCs are usually the so called freemium (free – premium) model, where the course is free of charge but the premium services cost some money. Inside the European Union the Open Education Europa portal collects the available MOOCs. From Hungary there is only one from University of Miskolc, but it seems that the registration site of this MOOC is broken. In this article we study does didactic questions which are beyond e-learning. Such a question is: How long should be a video? From the literature it seems that an average student watches only the first 4 minutes of a 10-15 minutes long video, hence, the best practice is to use 6-7 minutes videos. It is recommended to use the following sequence in the videos: motivation, knowledge transfer, example. Here the best practice is to use the so called “difficult task, easy task” motivation technique, where we present first a task, which is difficult for the students, then we show how to solve in general such problems. finally we solve the very same task, which is now an easy one. One has to give a script also to each video, because sometimes it is not easy to understand the speech of teacher on the video. Each teaching material has to stress that it is only a tool. After watching a video on how to write programs, no one become a programmer, only after if he or she writes lot of programs, which is a hard work, i.e., if he or she practices a lot. This means that beyond videos one has to also provide a collection of tasks, so the students can practice as self-study. In this article we study such didactic questions.

Keywords: MOOC, Massive Open Online Course, didactics

1. Bevezetés

A világon egyre inkább elterjednek a MOOC (Massive Open Online Course) megoldások. A MOOC kifejezés általunk használt fordítása: Nyílt online tömegkurzus.

A nyílt online tömegkurzusnak az alábbi jellemzői vannak a szakirodalom szerint [2, 4]:

- Tömeges (angolul: Massive): Egy képzésre korlátlan számú résztvevő jelentkezhet.
- Nyílt (angolul: Open): Két értelemben nyílt, egyrészt nagyon kevés formális követelményt kell teljesítenie a résztvevőknek, másrészt a kurzusok vagy ingyenesek, vagy alacsony díj ellenében hallgathatók.
- Online: A tananyagok az Interneten keresztül érhetőek el. Minden interakcióra, így a vizsgára is, online módon kerül sor.
- Kurzus alapú (angolul: course): fontos, hogy a tananyagot leíró meta adatok könnyen elérhetőek és mindenki által érthetőek legyenek. Így egy kurzus nem csak a tananyagból áll, hanem van előre ismert adatokból is, úgymint a kurzus célja, fontos határidők, az interakciók, pl. a vizsga, módja. Ezek alapján tud választani a hallgató a sok kurzus közül.

A nyílt online tömegkurzusoknak (MOOC) további két altípusát (cMOOC, és xMOOC) különbözteti meg a szakirodalom:

- Kapcsolat alapú nyílt online tömegkurzus (angolul: connectivist MOOC, vagy röviden: cMOOC): Ebben a képzésben a tanár-diák, és a diák-diák kapcsolat nagyobb szerepet kap, pl. nem teszt kérdésekre, hanem esszé kérdésekre épül a vizsga, a kapcsolódó fórumokon a diákok megvitathatják a kérdéseiket, felhívhatják a tanár figyelmét a tananyagban lévő esetleges hibákra. Korlátozott módon előre magadott időben az előadást tartó személlyel beszélgetni is lehet [3].
- Nyílt online tömegkurzus, mint egy normál kurzus kiegészítése (angolul: MOOC as eXtension of a normal course, vagy röviden: xMOOC): Ebben a képzésben a hangsúly a tömeges képzést segítő módszereken van, pl. a vizsgákat egy program értékeli automatikusa, a beküldött anyagokat automatikus plágiumszűrő ellenőrzi. Általában az ilyen kurzusokat lehet normál hallgatóként is hallgatni magas költségek mellett [1].

Összefoglalva, a nyílt online tömegkurzusok jellemzői: A kurzusok bárki számára elérhetőek. A hallgatók megtekinthetik a felvett előadásokat, megkapják a segédanyagokat, sőt még beadandó feladatokat is, amiket kiértékelve kapnak vissza. Majdnem mindig elérhető egy fórum is a kurzusokhoz, ahol a tapasztalatokat lehet kicserélni, kérdéseket feltenni. Korlátozott módon előre magadott időben az előadást tartó személlyel beszélgetni is lehet.

Ezek a kurzusok általában úgynevezett freemium (ingyenes – prémium) üzleti modellel működnek, azaz maga a kurzus teljesen ingyenes, de a prémium szolgáltatásokért fizetni kell. Az Európai Unió belül az Open Education Europa nevű portál fogja össze a nyílt online tömegkurzusokat. Ezen egyelőre csak a Miskolci Egyetem néhány kurzusa érhető el vendégként, de a szolgáltatás kipróbálása során nem sikerült ezekre regisztrálni.

Ebben a cikkben áttekintjük a távoktatás keretein túlmutató, nyílt online tömegkurzusokkal kapcsolatos módszertani kérdéseket. Ilyen kérdés például, hogy hány perces legyen egy videó? Egy kurzus hány leckéből álljon? Hogyan építsük fel a gyakorlati példasorozatokat?

A kérdések megvizsgálása után egy konkrét nyílt online tömegkurzus kialakítására teszünk javaslatot a programozó képzés területén. Feltárjuk azokat a problémákat is, amik szükségessé teszik egy ilyen képzés indítását.

Feltárjuk az úgynevezett pozitív visszacsatolást nyílt online tömegképzés esetén, miszerint: minél többen veszik fel a kurzusokat, annál igényesebbeknek kell lenniük a kurzusoknak, hiszen ellenkező esetben a képző intézménynek ez nem reklám, hanem ellenreklám lenne. És innentől megjelenik a pozitív visszacsatolás, hiszen minél igényesebbek a kurzusok, annál többen veszik fel a kurzusokat.

Ezzel indokoljuk azt, hogy a tervezett képzésünket egyre többen fogják felvenni és egyre többen elvégezni.

2. Módszertani kérdések

A nemzetközi irodalom szerint egy átlagos hallgató egy 10-15 perces videó első 4 percét tekinti meg, ezért elfogadott a 6-7 perces videók használata. Nagyon fontos a hallgatók motiválása, hiszen a nagy tömegeből csak kevés igazán motivált hallgató lesz. A videók elkészítésénél tanácsos a klasszikus motiváció – ismeretátadás – példa hármásra építeni. A legegyszerűbb a nehéz példa – könnyű példa motivációt használni, ami szerint először mutatunk egy példát, ami a hallgatóság számára nehéz, majd megmutatjuk, hogy általában az ilyen típusú problémákat hogyan kell megoldani, végül megoldjuk a konkrét példát, ami immáron egyszerű. A videók mellett mindig biztosítani kell a szkriptet is, hiszen a videón beszélő tanárt néha nehéz érteni. A tananyagoknak hangsúlyoznia kell, hogy ez csak egy segédeszköz, pl. a programozás videó megtekintése után senki nem lesz programozó, csak ha sok programot verejtékes munkával elkészít, azaz gyakorol. Ezért a videók mellett kellenek példatárak is, illetve nagyon részletesen kidolgozott példák, amik alapján a hallgatók gyakorolni tudnak.

Az eddigi tapasztalatok azt mutatják, hogy xMOOC rendszerek esetén nagyon sok ember vesz fel egy-egy tárgyat, de csak nagyon kevesen végzik azt el, körülbelül a jelentkezők 2-3%-a. Ezzel szemben a cMOOC rendszerben futó képzéseket kevesebben veszik fel, de a teljesítők aránya sokkal nagyobb, 10-12%.

Ez azzal magyarázható, hogy az xMOOC rendszerek normál előadásokat ültetnek át az online térbe, nem használják ki az online tér előnyeit, csak a tömeges képzés technikai problémáit oldják meg.

Ezzel szemben a cMOOC rendszerek segítik a kooperatív tanulást, például azzal, hogy lehetőséget adnak a hallgatóknak helyi tanuló csoportok kialakítására, ahol együtt gyakorolhatnak, lásd például az Udacity rendszert. Ezek mellett fórumok, blogok, wiki oldalak segítik az információ cserét. Sok cMOOC megoldás lehetővé teszi, hogy a hallgatók is megoszthassák az általuk fejlesztett összefoglalókat, példákat, lásd például a NovoEd rendszert. Ezen túl cMOOC rendszereknél hús vér ember javítja az esszéket, így a visszajelzés sokkal jobban segíti a hallgatók fejlődését.

Néhány cMOOC már a MOOC 2.0 felé mutat, ahol közösségi tanulásról beszélünk, ahol a hallgatók nem magukban dolgozzák fel a tanultakat, hanem közösségben. Ilyen megoldás, amikor a hallgatók maguk közül mentorokat választanak, egymás tesztjeit javítják, a házi feladatokat nem egyéneken, hanem csoportok kapják, közösségi eseményeket szerveznek a kurzusok köré.

Érdeemes a kifejlesztett tananyagokat nyílttá tenni. Ez ilyen anyagokat nevezik OER-nek (Open Educational Resources). De vajon miért? Azért mert így a tananyag tovább tud élni akkor is, ha az eredeti alkotó / alkotó csapat már nem akar az anyag fejlesztésével foglalkozni. Különösen fontos ez a folytonosan fejlődő informatika területén.

3. Programozói tömegképzés tervezete

A tömegképzés célja, hogy mindenki, aki szeretne programozó lenni, megtanulhassa a programozást bárhol, bármikor a saját időbeosztása szerint. Mindezt magyar nyelvű portálon, magyar nyelvű tananyagból, magyar nyelvű programozási környezetben.

Az eddigi módszertani megfontolásoknak megfelelően minden kurzushoz tartozna 10 lecke, minden leckéhez 1 videó, 1 szöveges rész, sok-sok feladat, illetve 1 ellenőrző feladatsor. A videók 5-6 percesek. Minden kurzus vizsgafeladattal zárul.

4. Alapelvek

A kurzusok elkészítésénél a következő alapelveket vesszük figyelembe:

- Programozni tanulni csak úgy lehet, ha az ember sok-sok programot ír, és eközben okul (csúnya szóval: szív)! Okul, mert a programok, amiket írunk, általában először rosszak, ki kell javítani a hibákat és ez az a nehéz feladat (a szívás) az, miközben elsajátítjuk a programozást. Tehát programozni tanulni csak szívással lehet!
- Nem lehet megtanulni a programozást, ha csak működő programokat másolunk le. Jobb, ha rossz a kód és megmutatjuk, hogyan kell kijavítani.
- Minden kurzus elem annyira legyen egyszerű, amennyire csak lehet, és annyira legyen csak bonyolult, amennyire muszáj. Angolul: As Simple As Possible, As Complicated As Necessary!
- Programozni mindenkit meg lehet tanítani, akinek van algoritmikus gondolkozása, és aki tud sonkás rántottát készíteni, annak van.
- A problémamegoldásnál érdemes a konkrét felől haladni az általános felé, tehát megoldjuk a konkrét feladatot papíron ceruzával, az általános megoldás pedig megadjuk programmal.
- A jó programozónak van rendszerszemlélete, hogy meglássa, minden lehetséges esetet lefedette-e.
- Fokozatosság elve: a könnyű példától haladunk a nehezebbek felé.
- Egy videó három részből áll: motiváció, ismeret átadás, ismeret alkalmazása.

5. A programozói szemlélet kialakítása

Ahhoz, hogy valakiből jó programozó legyen, el kell sajátítani azokat a közös alapokat, amiket minden programozó sajátjának vall, függetlenül attól, hogy milyen nyelven programoz. Ez a programozók közös tudása. A közös tudás elsajátítása hosszú folyamat. Ennek első lépése a programozó szemlélet, illetve ennek kialakításához szükséges ismertanyag, amely a következőkből áll:

- Az első program, amit meg kell tudnom írni: az üres program! Csak a második a Helló Világ!
- A Helló Világ program nem a képernyőre ír, hanem a Standard Outputra!
- A program utasításokból és kifejezésekből áll.
- A kifejezésnek van típusa és értéke.
- A program utasítások sorozata, néhány utasítás tartalmaz kifejezéseket is.
- 3 fajta utasítás létezik: szekvencia, szelekció, iteráció.
- Minden utasításnak van szintaxisa: hogyan néz ki egy szabályos, szintaktikailag helyes utasítás.
- És minden utasításnak van szemantikája: mit csinál az adott utasítás.

6. Kurzus sorozatok és nehézségi szintek

Minden kurzussorozatnak van egy célja. A cél mindig az, hogy a kurzusok elvégzése után a hallgató képes legyen gyakorlati programozói feladatokat megoldani.

A gyakorlati feladatok nehézségi szintje, illetve ezek jelölése csillagokkal:

- tanuló I. *
- tanuló II. **
- segéd ***
- junior programozó ****
- senior programozó *****

7. Programozási nyelv

A tömegképzés programozási nyelve a Java, hiszen ilyen szakemberekre van a legnagyobb kereslet a piacon. A Java nyelvhez is van több fajta integrált fejlesztési környezet (angolul: IDE), mint pl. a NetBeans, Eclipse, Ezek közül a legelterjedtebb az Eclipse, ráadásul ehhez található magyar nyelvű felület, így ideális megoldás.

Platform

Jelen pillanatban a 4 legnagyobb MOOC rendszer a következő:

- Coursera
- edX
- UDACITY
- Khan Academy

Ezek közül az edX nyilvánossá tette rendszerének forráskódját, ráadásul olyan nagy nevek támogatják többek közt ezt a projektet, mint az MIT és a Google.

Jó választásnak tűnik még az OpenMOOC rendszer is, de ezt sokkal kevesebben fejlesztik.

Az edX platform Python nyelven készül, alap építőköve az XBlock, ami egy leckét ír le. A videókat a YouTube-ról lehet egyszerűen áthúzni. Jól dokumentált. Felhasználói licenye lehetővé teszi a szabad felhasználást, de előírja, hogy minden továbbfejlesztését szabad szoftverként elérhetővé kell tenni.

A licenz nem korlátozza saját márkanev használatát, illetve lehetővé teszi, hogy mindenki saját maga működtesse saját tömegképzését. Támogatja a felhő alapú megoldásokat, de egy sima Linux szerverre is feltelepíthető.

Az OpenMOOC és a Open edX első generációs MOOC rendszer. A második generációs MOOC rendszerek a közösségi tanulásra helyezik a hangsúlyt. Ezek közül meg kell említeni a NovoED és a OpenupEd rendszereket.

8. A probléma

A hazai GDP 12 százalékát előállító cégeket tömörítő Informatikai Vállalkozások Szövetsége szerint, ha be tudnák tölteni azt a tízezer munkahelyet, amelyet az ágazatban 2013-ban regisztráltak, az 120 milliárd forinttal növelné a bruttó hazai terméket. Tizenötezer új informatikai munkahely létrehozása pedig már egyszázalékos GDP-növekedést eredményezne.

9. A jelenlegi helyzet

A felsőoktatásban 3 informatikai BSc van, ezek népszerűsége: / www.felvi.hu /

- Mérnökinformatikus BSc, 5. legnépszerűbb szak, 2014-ben 4.939 fő jelentkezett a szakra,
- Gazdaságinformatikus BSc, 10. legnépszerűbb szak, 2014-ben 3.522 fő jelentkezővel,
- Programtervező informatikus BSc, 15. legnépszerűbb szak, 2013-ban 2.555 fő jelentkezővel.

Erre a 3 szakra 2013-ban ennyi hallgatót vettek fel: / www.felvi.hu /

- Mérnökinformatikus BSc, felvettek száma 2013-ban 1.968 fő,
- Gazdaságinformatikus BSc, felvettek száma 2013-ban 1.263 fő,
- Programtervező informatikus BSc, felvettek száma 2013-ban 1.089 fő,
- Összesen: 4.320 fő.

Végzetek száma 2011-ben: / www.felvi.hu /

- Mérnökinformatikus BSc, 732 fő,
- Gazdaságinformatikus BSc, 197 fő,
- Programtervező informatikus BSc, 331 fő,
- Összesen: 1.260 fő.

A nem pontosan összevethető számok alapján körülbelül a felvettek 25 – 35% része végez, ami megegyezik a saját tapasztalataimmal is. Tehát jelenleg a felsőoktatás kb. 1.500 fő programozót bocsát a munkaerőpiacra évente.

10.A terv

Magyar nyelvű ingyenes nyílt online programozó tömegképzés indítása, amely legalább 500 fő programozót bocsát a munkaerőpiacra évente.

A nyílt online tömegképzés (angolul: massive open online course, MOOC) azt jelenti, hogy a kurzusokat bárki felveheti az interneten, a kurzushoz kapcsolódó anyagokat általában ingyen letöltheti, vizsgázhat és a képzés végén oklevelet kap. Ezek a fajta képzések egyre népszerűbbek, sok nagy egyetem elérhetővé tette legnépszerűbb kurzusait ilyen módon. Ez jó reklám ezen egyetemeknek, hiszen tényleg tömegek veszik fel ezeket az órákat. E kurzusok általában ingyenesek, csak a prémium szolgáltatásokért kell fizetni.

A nemzetközi tapasztalatok szerint egy nyílt online tömegképzésre jelentkezők 2% jut el a képzés végére és szerez oklevelet.

Figyelembe véve ezt a számot, az 500 fős kibocsátási célt 25.000 fő jelentkező tömeg mellett lehet elérni.

Ez csak úgy elérhető, ha ezzel nem a fiatalokat, hiszen ők egyébként is egyetemre / főiskolára vágnak, hanem azokat a felnőtt tömegeket célozzuk meg, akik egy jobb állás reményében szeretnének megtanulni programozni.

A 25.000 fős jelentkező létszám eléréséhez magas színvonalú oktatóanyagok szükségesek, illetve komoly infrastruktúra, hogy a jelentkezők biztosítva érezzék magukat, hogy ezen a helyen meg lehet tanulni programozni.

11.Az indokoltság

A munkaerőpiacon sok honfitársunkból jó programozó válhat, hiszen ehhez csak algoritmikus gondolkodás, rendszerszemlélet és egy év tapasztalat kell valamilyen programozási nyelvvél. Ugyanakkor ezek az emberek dolgoznak valahol, gyermekeket nevelnek, szórakozni járnak,

nincs kedvük / energiájuk / idejük egy magasabb életszínvonallal kecsegtető képzettség megszerzésére a felsőoktatásban.

A mai felsőoktatás nekik levelező, illetve elvétve távoktatási lehetőségeket nyújt. Tapasztalatból tudom, hogy a levelező oktatás majd minden péntek és szombat feláldozását megköveteli. A távoktatási megoldásokkal nincs tapasztalatom, de úgy tűnik, hogy nem túl népszerűek.

Egy nyílt online tömegképzés esetén működik a pozitív visszacsatolás: minél többen veszik fel a kurzusokat, annál igényesebbeknek kell lenniük a kurzusoknak, hiszen ellenkező esetben a képző intézménynek ez nem reklám, hanem ellenreklám lenne. És innentől öngerjesztő, hiszen minél igényesebbek a kurzusok, annál többen veszik fel a kurzusokat.

12. Várható előnyök

A következő előnyöket látom:

- Sokan elkezdhetik a programozó képzést, akik egyébként nem, vagy csak nagy áldozatok árán tudnának erre a pályára lépni.
- A frissen végzett programozók jobban fognak keresni, nagyobb vásárlóerővel bírnak.
- Ezeket a programozókat olyan cégek szívják fel, akik kis befektetéssel nagy értéket tudnak előállítani.
- A nagy és jó programozó-utánpótlás hatására külföldi programozó cégek települnek meg hazánkban, programfejlesztő vagy kutató részlegeket hoznak létre, ezzel nem csak programozó állásokat létrehozva.
- A jó példát más szakok is követni fogják és valószínűsíthető más, elsősorban mérnöki nyílt online tömegképzések indítása.

13. Várható eredmények

A hagyományos felsőoktatás továbbra is 25-35%-os hatásfokkal évente körülbelül 1.500 fő programozót fog kibocsátani.

A tervezett nyílt online programozó tömegkurzus kezdetben évi 2%-10% hatásfokkal fog programozókat kibocsátani. Az első évben talán csak pár tized, de az évek haladtával, ahogy nő a képzés híre és így a jelentkezőszám, illetve ahogy javul a hatékonyság, évente néhány száz, középtávon néhány ezer fős programozó kibocsátással számolhatunk.

Ha úgy számolunk, hogy a jelentkezők száma 1.000 fővel indul, minden évben 40%-kal nő, egészen 25.000 főig, illetve a hatékonyság 2%-ról 10%-ra nő évi 20%-os javulással (mindkét növekményt a pozitív visszacsatolás okozza), akkor a következő kibocsátási számokat prognosztizálhatjuk:



1. ábra. A tervezett képzésben kibocsátott programozók száma, kezdetben 1.000 jelentkezővel, 2%-os hatékonysággal, jelentkezők száma évente 40%-kal nő, kibocsátás hatékonysága évente 20%-kal nő

Az ábra tanulsága szerint 6 év alatt érhető el, hogy 500 fő programozót bocsásson ki ez az újfajta képzési rendszer. Illetve kicsit több, mint 8 év múlva már a jelenlegi felsőoktatással azonos számú programozó kibocsátására képes. Ugyanakkor egy szint után a kibocsátások száma nem tud nőni.

Egy sokkal pesszimistább kalkuláció látható a következő ábrán. Itt 200 fős induló jelentkezővel, de 10%-os induló hatékonysággal számoltunk. Azért számolhatunk itt a 2%-nál sokkal jobb, 10%-os induló hatékonyságot, mert kisebb jelentkező létszám esetén várhatóan jobban tudjuk a cMOOC-ra jellemző intenzív, határfok növelő, kommunikációt megvalósítani, mint az előző ábra 1.000 fős kezdeti jelentkezése esetén.

Itt a javulás csak 30%, illetve 10%. Jó látható, hogy ebben az esetben 9 év kell az 500 fős kibocsátás eléréséhez:



2. ábra. A tervezett képzésben kibocsátott programozók száma, kezdetben 200 jelentkezővel, 10%-os hatékonysággal, jelentkezők száma évente 30%-kal nő, kibocsátás hatékonysága évente 10%-kal nő

Irodalomjegyzék

- [1] S. Cooper, M. Sahami: Reflections on Stanford's MOOCs. *Communications of the ACM* **56(2)** (2013), 28–30.
- [2] D. Clow: MOOCs and the funnel of participation. *Proc 3rd international conference on learning analytics and knowledge* (2013), 185–189.
- [3] R. Kop, A. Hill: Connectivism: learning theory of the future or vestige of the past? *IRRODL* **9(3)** (2008).
- [4] M.Y. Vardi: Will MOOCs destroy academia? *Communications of the ACM* **55(11):5** (2012).

Az EKF FutureRFID - Az RFID/NFC technológia továbbfejlesztési lehetőségei az „Internet of Things” koncepciói mentén - kutatási projektjének bemutatása

Development possibilities in the RFID / NFC

**Bánlaki József, Geda Gábor, Juhász Tibor, Király Roland, Kovács Emőd,
Radványi Tibor**

Eszterházy Károly Főiskola

banlaki.jozsef@aries.ektf.hu, gedag@ektf.hu, juhaszti@ektf.hu,
kiralay.roland@ektf.hu, emod@ektf.hu, dream@aries.ektf.hu

Absztrakt: Kutatásaink során az RFID (Radio Frequency IDentification) [9], és NFC (Near Field Communication) [10][11] alapú technológiákkal, továbbfejlesztésükkel és új felhasználási területekkel foglalkozunk. Az RFID és az NFC szinte észrevétlenül válik mindennapjaink részévé. Amellett, hogy elterjedése új területeket nyit meg a számunkra, az RFID chippek bekerülnek a szórakoztató elektronikai, valamint a biztonságtechnikai eszközeinkbe is, így válnak az életünk részévé. Az említett okok miatt Időszerűnek érezzük a hardver és szoftver eszközök továbbfejlesztését, védelmét és azt, hogy időben fel tudjunk készülni és reagálni a kihívásokra. Mindezek mellett kísérletet teszünk arra is, hogy az RFID alapú technológiát minél szélesebb körben megismertessük a nagyközönséggel és a társadalmi elfogadottságát növeljük a mihamarabbi elterjedés érdekében. Meggyőződésünk, hogy kutatási eredményei alapján az „Internet Of Things” magyarországi kompetenciaközpontjává válhatunk és bekapcsolódhatunk a Horizon2020 program ide vonatkozó kutatási feladataiba. Másrészt alkalmazásorientált kutatás-fejlesztéssel a technológia hazai elterjesztésében katalizátor szerepet szeretnénk játszani. Ebben a publikációban a kutatási tevékenységünket, a terveinket és az előttünk álló lehetőségeket mutatjuk be a kutatócsoportjaink eredményein keresztül.

Kulcsszavak: RFID, NFC, lokalizáció, IOT, Future Internet, transzponder, reader, interrogátor, hibrid technológia, szenzor hálózat, user interface, RFID tag, RFID címke.

Abstract: Our research deals with RFID (Radio Frequency IDentification) [9], and NFC (Near Field Communication) [10][11] based technology, and the development and further application fields thereof. Using RFID and NFC become almost invisibly an everyday part of our lives. Besides its spreading fields of applicability, RFID chips are integrated into daily used consumer electronics, and security technology devices. Because of the mentioned reasons, we think it is timely to consider the advancement of hardware and software components, their protection and to prepare to be able to handle new challenges. Besides all these, we try to introduce this technology to the general public, to raise its social acceptance for its continuous spreading. We believe that based on our research results we can become an “Internet of Things” competence center in Hungary, and are able to join the relevant research tasks of the Horizon2020 program. On the other hand, we would like to play a catalyst role in the national spreading of this technology with application-oriented research and development. In this paper, through the results of our research teams we will present our research activity, our plans, and the possibilities ahead of us.

Keywords: RFID, NFC, localization, IOT, Future Internet, transponder, reader, interrogator, hibrid technologie, sensor networks, user interface, RFID tag, RFID.

1. Bevezetés

Az Eszterházy Károly Főiskola (EKF) a 2011-ben meghirdetett TÁMOP 4.2.2.C-11/1/KONV pályázati felhívásra a Bay Zoltán Alkalmazott Kutatási Közhasznú Nonprofit Kft.-vel alkotott konzorcium vezetőjeként „FutureRFID - Az RFID/NFC technológia továbbfejlesztési lehetőségei az Internet of Things koncepciói mentén” címmel projektjavaslatot nyújtott be. A támogató a projektjavaslatot nyertessé nyilvánította, melynek eredményeként a 24 hónap futamidejű kutatási projekt végrehajtása 2013. január 1-én megkezdődött.



1. ábra IoT/RFID/NFC Kompetencia Központ kialakítása

Az 1. ábrán látható a projekt során az Eszterházy Károly Főiskolán kialakítandó IoT/RFID/NFC Kompetencia Központ felépítését. Az alábbi táblázatban részletezzük az a Központ egyes tevékenységeit:

Kutatási kapacitás fejlesztése	Kutatási tevékenység	Hálózatépítés	Oktatás
<ul style="list-style-type: none"> Műszaki infrastruktúra kialakítása Humán erőforrás 	<ul style="list-style-type: none"> (R1) Kritikák (R2) Digitális tulajdonjog 	<ul style="list-style-type: none"> Vállalkozásokkal való kapcsolattartás Hazai és nemzetközi kutatási hálózatokkal 	<ul style="list-style-type: none"> Elméleti képzés tantervének kidolgozása

kiépítése	<ul style="list-style-type: none"> • (R3) Megbízhatóság • (R4) Egységesítés • (R5) Lokalizáció • (R6) Szenzorok • (R7) Hibrid technológiák • (R8) ROI • (R9) Alkalmazási területek 	<ul style="list-style-type: none"> való együttműködés • Közös kutatások • Közös pályázatok • Work-shopok, szakmai utak 	<ul style="list-style-type: none"> • Gyakorlati képzés tantervének kidolgozása • Elméleti és gyakorlati képzés indítása
-----------	---	--	---

A projekt célkitűzési között szerepel Internet of Thing (IoT) kutató központ létrehozása, mely elősegítheti az EKF közép- és hosszú távú stratégiai céljainak érvényesülését.

Céljaink szerint az IoT területén végzett kutatások révén a kutatóintézet egyrészt országos kompetenciaközponttá válhat, másrészt érdemben bekapcsolódhat a területen folytatott nemzetközi kutatási tevékenységbe. A kitűzött célokat megalapozza, hogy – a várható gazdasági és társadalmi előnyöket felismerve – az Európai Unióban kiemelt kutatási témaként tekintenek az Internet of Things területére és azon belül az RFID technológiára. Ezek további elterjedéséhez ugyanis számos jelenlegi akadályt kell leküzdeni, ezért – hasonlóan a világ más régióihoz – az EU-ban is jelentős kutatási forrásokat allokálnak a 2014-től kezdődő FP8-as programozási időszakban.

A kutatóintézettel szembeni gazdasági elvárás, hogy közép- és hosszútávon önfenntartó legyen, valamint legyen képes hozzájárulni az EKF intézményi működéséhez. Ezt a követelményt a piaci szereplőkkel kiépített projektportfólió bevételei biztosítják.

A továbbiakban néhány kutatást kívánunk bemutatni.

2. RFID alapú lokalizáció

Az RFID alapú lokalizációt célzó kutatásaink egyik központi eleme az, hogy milyen infrastruktúra és matematikai apparátus segítségével lehet meghatározni az RFID technológiával működő transzponderek helyét, valamint hogyan lehet azok mozgását követni. Az ilyen irányú kutatások segítenek abban, hogy a beltéri lokalizációt olyan hardverek segítségével tudjuk megvalósítani, amelyek jelenleg is rendelkezésre állnak, vagy kis átalakítással alkalmazhatóak a feladat elvégzésére. A fő probléma a beltéri lokalizációval az, hogy a GPS alapú rendszerek az árnyékolás miatt nem találják meg a megfelelő számú műhold kapcsolatot, vagy egyáltalán nem is látnak ki az adott (vizsgált) területről. Ezen okból javasoljuk a GPS alapú navigációs rendszerek kiváltására az RFID alapú eszközök bevezetését, viszont nem az ismert, igen költséges módszerek segítségével, amelyek kizárólag az olvasókkal ellátott kapukon való áthaladást képesek használni a lokalizáció megvalósítására.

A szabványos RFID hardver némi átalakítással, és az egyes protokollok felül bírálata mellett kiválóan alkalmazható ilyen célokra. Véleményünk szerint az antennák és protokollok intelligensebbé tételével, valamint egy merőben új middleware réteg bevezetésével az RFID technológia alkalmassá tehető objektumok pontosabb lokalizálására. Ezzel kapcsolatban számos kísérletet és kutatást végeztünk el, valamint kifejlesztettünk az elméleteink

igazolására egy prototípust. A célunk ezzel az, hogy az EPCGlobal RM (Reader Management) standardjának olyan irányú kiterjesztésére tehessünk javaslatot (ezt a koncepciót szabadalom irányába is szeretnénk továbbvinni), amelyben a lokációs képesség az olvasók szintjéről az olvasási pontokra, vagyis az antennák és a transzponderek szintjére helyezi át a lokalizációs információt. A szoftverekben történő átalakításról később még szót ejtünk. Most vizsgáljuk meg az antenna karakterisztikák dinamikus változtatásában rejlő lehetőségeinket. Mivel a célunk az, hogy az észlelt transzpondert ne csak felismerjük, hanem annak a helyzetét is meg tudjuk határozni, szükségünk van egy speciális tulajdonságra az antennák hardverébe integrálva. Ezt a tulajdonságot „Bólogatásnak” neveztük el [2], és a lényege, hogy az adott antenna több kis patch antennából áll össze, amely patch antennákat az olvasó szoftvere páronként be, vagy ki tud kapcsolni. Minden egyes ilyen átkapcsoláskor az antenna karakterisztikája megváltozik, és a vizsgált terület egy jól határolt térrészét képes mérni. Ezeket a térrészeket ezután azonosítóval kell ellátni, ami alapján az ott elhelyezkedő transzponderek helyzete könnyedén meghatározható, mivel minden méréshez rendelkezésre áll a térrész azonosítója. Ez az azonosító tárolható az vizsgált transzponder, vagy címke azonosítójával együtt. Az így kapott információt ezután fel lehet használni navigációs célokra, vagy transzponderrel ellátott tárgyak útvonalának a követésére. A technológia nagyon hasznos, de használhatósága függ a vizsgált terület felosztásának részletességétől, valamint az alkalmazott antenna rendszer pontosságától. A mérések során is számos problémába ütközhetünk. Az első ilyen, mikor az antenna által észlelt transzponder két egymás melletti térrészben egyszerre vannak jelen (a transzponder átlóg egy másik területre is). Ebben az esetben egy másik megoldást kell alkalmaznunk, amely szintén a kutatásaink részét képezi [2][3][5]. Ezzel a modellel, és az implementált prototípus segítségével egy adott transzponderről az előzőleg végzett mérések és a területről szerzett információink alapján egy speciális matematikai apparátus használata mellett [2][3] el tudjuk dönteni, hogy az a terület szeparált térrészei közül a legnagyobb valószínűséggel melyikben foglal helyet. Ez kimondottan hasznos, mikor a vizsgált transzponder átmenetileg nem látható az olvasók antennái számára, de tudjuk róla, hogy a vizsgált területen belül helyezkedik el. Ez a modell felhasználja a terület RSSI lefedettség térképét, amelyet a rendszer kalibrálása során gyűjtünk, és a transzponderről korábban mért lokalizációs információt. Ezekkel az adatokkal képes becsülni a transzponder lehetséges helyzetét. Ez a fajta lokalizációs alkalmazás önmagába nem elegendő a hely meghatározáshoz az RSSI alapú mérések bizonytalansága és megbízhatatlansága miatt, de kiváló arra, hogy a kiegészítse más (pl.: a korábban említett lokalizációs middleware) lokalizációs framework-ök funkcionalitását. A lokalizáció további javítására nem csak az olvasók és az antennák funkcionalitását egészítettük ki. A további javítás érdekében bevezettünk egy új, speciális transzpondert is, amely rendelkezik felhasználói interface-el [1] és amely ki és bekapcsolhatóvá teszi az adott transzponder észlelhetőségét. Ez amellet hogy rendelkezik a lokalizációt segítő extra információval, segít a technológia társadalmi elfogadottságának a növelésében, mivel a használójának lehetőséget kínál az azonosítóval ellátott eszközeinek az elrejtésére, vagy átmeneti üzemen kívül helyezésükre. Véleményünk szerint az emberek számára hasznos alkalmazások készítése és terjesztése mellett ez a másik lehetséges út a társadalmi elfogadottság növelésére és az ismert „Big Brother” effektus megszüntetésére, amelyet a mobil telefonok bevezetésénél már megismertünk.

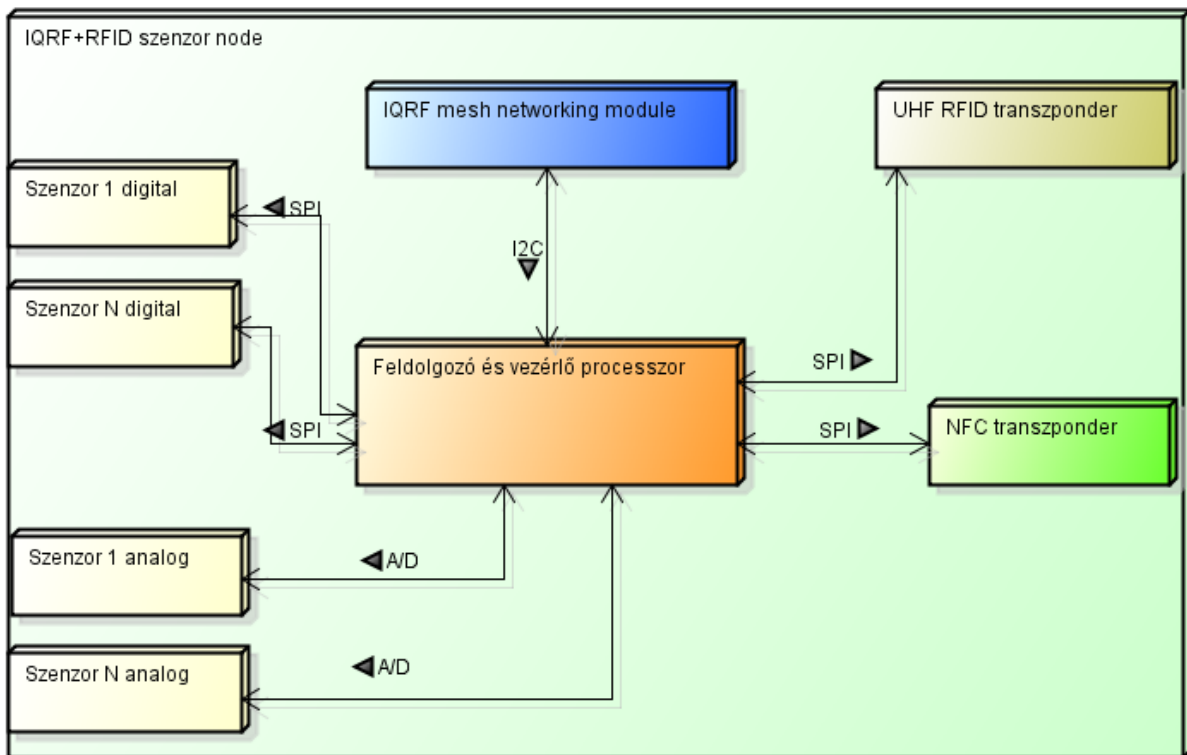
A fejezetben bemutatott technológiák részét képezik az „Internet of Things” és a „Future Internet” koncepcióknak. A kifejlesztésüket számos kísérlet és kutatási tevékenység előzte meg. A matematikai modellt, valamint az új működési elveket publikáltuk tudományos szakfolyóiratokban, és azok tökéletesítésén jelenleg is dolgozunk.

3. Szenzorhálózatok és modellezésük

A lokalizáció javítása mellett céljaink között van a különböző szenzor hálózatok alkalmazása és azok kiterjesztése RFID és NFC technológiák hozzáadásával [4]. A szenzorhálózat létrehozása során arra a következtetésre jutottunk (a szakirodalom- és műszaki kutatásból szerzett ismereteink alapján), hogy a szenzorhálózat kommunikációs vázát a MICRORISC s.r.o. által gyártott IQRF mesh hálózati eszközökre [8] alapozzuk. Használatuk mellett a fejlesztési idő valamint a költségek csökkentése miatt döntöttünk, mivel ezekkel az eszközökkel viszonylag alacsony áron egy jól skálázható RF hálózatot tudunk kialakítani. Mindezek mellett a rendszer lehetőséget biztosít saját fejlesztésű eszközeinkbe történő integrálásra is.

Az IQRF eszközökkel 868 és 916 MHz ISM sávban működő RF mesh hálózatot hozhatunk létre, melyekben hálózatonként 65000 eszköz működhet. Az IQRF rendszerben a hálózati protokollok, a saját tervezésű vezérlőkkel csak az IQRF modul illetve az ezekből összeálló hálózat konfigurálását kell végezni.

A szenzor node-ok (csomóponti eszközök) rendelkeznek UHF RFID és NFC interfésszel, (A kutatásaink során AMS SL900A és AMS AS3953 integrált áramkörök alkalmaztunk). Az így létrehozott RFID és NFC interfészek célja, hogy egy saját tervezésű protokollal lehetséges legyen pl. egy node leolvasásával hálózat többi node-jának szenzoradatait kiolvasni. A hálózat passzív interfészeket használ, tehát tápellátás nélkül is működhet, és segítségével lehetőségünk van a node-okat UHF RFID olvasóval megtalálni nagyobb távolságból, ha a node-ban található akkumulátor lemerült. A node-on található szenzorok digitális interfésszel rendelkeznek, és akár közvetlenül az IQRF modulra is csatlakoztathatóak.



2. ábra Szenzorhálózat

A bemutatott hálózat segítségével a következő környezeti tulajdonságokat szeretnénk mérni: hőmérséklet, páratartalom, légnyomás, fényerő (látható tartományban), UVA és UVB sugárzás, CO és CO₂ gáz koncentráció és talaj nedvességtartalom mérés, amely attribútumok

mind nagyon fontosak például élő környezetek, botanikus kertek, vagy egyéb haszonnövény kultúrák életben tartásához, vagy a környezeti hatások optimálissá tételéhez.

4. Hibrid technológiák alkalmazása

Az RFID/NFC technológia és más technológiák együttes alkalmazása számos felhasználási lehetőséget nyithat meg. Ezért került kutatásaink középpontjába annak vizsgálata is, hogy a mobiltelefonokban elérhető képességek kombinálásával milyen további hozzáadott funkcionalitást biztosíthatunk a felhasználó számára [5][6].

Vizsgáljuk a mozgásérzékelő és egyéb szenzorok, az NFC és a mobil telefon, mint számítógép adta képességek együttes felhasználásának lehetőségeit a beltéri pozicionálás érdekében neurális hálózat támogatásával.

A mobil készülékben elérhetőek a magnetométertől és a mozgásérzékelő szenzoroktól származó információk, amelyekből a mozgás irányát, sebességét, lassulást, gyorsulást kapjuk meg. Emellett a GSM, WIFI, Bluetooth hálózat jeleinek erősségéből az adótól való távolság közelítő számítása (RSSI – Received Signal Strength Indication) is lehetséges, ami szintén segíthet a felhasználó pozíciójának pontosításában.

A szenzorok által szolgáltatott adatok pontatlanok és zajosak, ugyanakkor a különböző szenzorok adatainak összevetése révén a hiba csökkentése lehetséges. Ehhez több módon számítjuk a pozíciókat, ami általában csökkenti a hibát.

Emellett a pozíció további pontosítása érdekében neurális hálózat algoritmust kísérünk meg alkalmazni.

A kutatás során android mobil telefonra telepíthető alkalmazást készítettünk, amely a mért adatokat képes egy adatbázisba feltölteni az algoritmusok tesztelhetősége érdekében. Adatbázisban tároljuk a feltöltő eszköz típusát, a feltöltés dátumát, valamint a feltöltő által beírt megjegyzést, amely alapján az adatok feldolgozásakor ismerhetjük az elvégzett teszt körülményeit. Természetesen az adatbázisba bekerülnek a szenzorokról (lineáris gyorsulásmérő, giroszkóp, gravitáció szenzor, magnetométer, WIFI RSSI) érkező adatok.

A feltöltött adatokon hibajavító algoritmusokkal dolgozunk, valamint neurális hálózatot tanítunk, amelytől a hiba csökkentését várjuk.

A feldolgozásokat Python nyelven végezzük, amely kényelmesen használható listák, tömbök, mátrixok feldolgozására.

A neurális hálózat algoritmusok közül a többrétegű perceptronon működő hiba-visszaterjesztés algoritmust választottuk. Egy Python [7] könyvtárat használtunk programozási célokra, valamint saját fejlesztésű neurális hálózat algoritmusokat alkalmaztunk.

5. Köszönetnyilvánítás

A kutatás az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával a TÁMOP-4.2.2.C-11/1/KONV-2012-0014 FutureRFID - Az RFID/NFC technológia továbbfejlesztési lehetőségei az "Internet of Things" koncepció mentén című projekt keretei között valósult meg.

Irodalomjegyzék

- [1] József Bánlaki, Miklós Hoffman, Tibor Juhász: Add a User Interface to UHF RFID Tags, March 23 (2014). <http://www.rfidjournal.com/articles/view?11593>
- [2] Zoltán Ruzsa, Zsolt Parisek, Roland Király, Tamás Szakács, Henrik Hajagos: Build a mathematic based RFID localization framework, *The 9th International Conference on Applied Informatics* to be held in Eger, Hungary January 29-February 1 (2014).
- [3] Szakács Tamás, Roland Király, Ruzsa Zoltán, Parisek Zsolt, Hajagos Henrik: Matematikai alapú lokalizációs keretrendszer, *NETWORKSHOP'14* Pécs, április 23-25 (2014).
- [4] Sándor Király, Tibor Radványi, Csaba Biró, Péter Szigetváry, Péter Takács: Programming an RFID reader for getting data from different semi-passive sensor tags, *The 9th International Conference on Applied Informatics* to be held in Eger, Hungary January 29-February 1 (2014).
- [5] Tibor Tajti, Gábor Geda, András Perjési, Péter Magyar, Tamás Balla, Attila Szívós, Tamás Gregus: Indoor localization using NFC and mobile sensor data corrected using neural net, *The 9th International Conference on Applied Informatics* to be held in Eger, Hungary January 29-February 1 (2014).
- [6] Péter Magyar, Gábor Geda, Tibor Tajti, András Perjési, Tamás Gregus: Business Oriented Creature Identification, *The 9th International Conference on Applied Informatics* to be held in Eger, Hungary January 29-February 1 (2014).
- [7] Python Programming Language: <https://www.python.org/>
- [8] MICRORISC s.r.o. <http://www.microrisc.cz/new/webcz/>
- [9] Anokwa, Y., et al.: A User Interaction Model for NFC Enabled Applications. in *IEEE International Conference on Pervasive Computing and Communications Workshops 2007*. New York ,USA.
- [10] ISO/IEC 18092(ECMA-340): Information technology – Telecommunications and information exchange between systems - Near Field Communication - Interface and Protocol (NFCIP-1). First Edition, 2004-04-01.
- [11] José Bravo, Ramón Hervás , Gabriel Chavira From Implicit to Touching Interaction: RFID and NFC Approaches ,*Sixth International Conference on the Management of Mobile Business* (ICMB 2008)0-7695-2803-1/07 © 2008 IEEE

Szenzor adatklaszterek wavelet elemzése

Wavelet analysis of the sensor data clusters

Gál Zoltán

Debreceni Egyetem

ZGal@unideb.hu

Absztrakt: A Jövő Internet technológiák és szolgáltatások egyik kiemelten kezelt területe a Tárgyak Internete (IoT – Internet of Things). A „smart” jelzővel ellátott szolgáltatások a városok, az energiahasználat, az egészségügy, az ipar és a közlekedés területén már most erőteljesen éreztetik gazdasági, társadalmi és technológiai hatásukat. A szenzorok segítségével mintavételezett mennyiségek szignifikáns módon megnövelik a feldolgozásra kerülő adatok volumenét. A „Big Data” jelenség itt is szükségessé teszi annak átgondolását, hogy mennyi és milyen adatokat kell és érdemes mintavételezni a különböző folyamatok optimális módon történő intelligens vezérléséhez. Az Internet csomópontok számának több mint egy nagyságrenddel való növekedése az évtized végéig komoly erőfeszítéseket tesz szükségessé az alacsony fogyasztású, biztonságos, vezeték nélküli csatornákon működő kommunikációs technológiák tekintetében. A szenzorok saját energiaforrásának terhelés csökkentése érdekében a mintavételezések gyakoriságának és típusának meghatározása egyike azon kihívásoknak, amikre már a közeljövőben választ kell adni ahhoz, hogy a különböző folyamatok intelligens vezérlése költséghatékonyan történhessen. A cikkben a nagyszámú szenzortól származó adathalmazok információ tartalmának kinyerésére vonatkozó megoldásokat vizsgáljuk meg. Klaszter analízissel végezzük el a különböző változók, illetve szenzorok csoportosítását, majd ezekből a reprezentáns entitást a DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algoritmussal azonosítjuk be. Az így kiválasztott változókat, mint idősorokat elemezzük wavelet módszerrel annak érdekében, hogy a szabályozott folyamatban a váratlan események kialakulását előzetesen detektáljuk. Különböző rendszerekből, mint forrásokból (szuperszámítógép, illetve országos hálózat) származó adathalmazokon végezzük el az elemzéseket és ezek összehasonlításának eredményeit mutatjuk be.

Kulcsszavak: Tárgyak Internete, wavelet, idősor, Internet, szenzor, aktuátor

Abstract: One of the hot topics of the Future Internet technologies is the Internet of Things (IoT). Services marked with „smart” attribute like smart cities, smart energy usage, smart health, smart industry, smart transportation have huge effects on the economy, society and technology. Variables sampled with sensors increases significantly the volume of processed data. The Big Data symptom makes necessary to re-evaluate the maximum number of variables sampled for optimal controlling of different processes. The increase of the number of Internet nodes by one order of magnitude until the end of current decade induces serious effort to develop low consumption, secure wireless communication technologies. Need to determine the sampling type and sampling frequency to relieve the usage of finite energy source of the sensors battery. This is one of the most important challenges required to be answered for cost efficiency of the intelligent processes control. In this paper we analyse different information extraction methods from data sampled from high number of sensors. We use data cluster analysis to group variables and determine the representing entity with DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm. Variables determined are analysed with wavelet transformation to detect in time the surprise event formation of the controlled process. Analysis results and comparison of data sets sampled from different systems as sources (i.e. supercomputer, wide-area network) are presented.

Keywords: Internet of Things, wavelet, time series, Internet, sensor, actuator

1. Bevezetés

Az ITU (International Telecommunication Union), illetve az EU által működtetett IERC (Internet of Things European Union Research Cluster) szerint a Tárgyak Internete (IoT – Internet of Things) egy olyan dinamikus globális infrastruktúra, amely szabványokra és együttműködő kommunikációs protokollokra épülő önkonfiguráló képességeivel intelligens interfészeket használ a fizikai, valamint virtuális, egyedi azonosítóval rendelkező tárgyak címezéséhez. A tárgyak fizikai attribútumokkal, illetve virtuális jellemzőkkel rendelkeznek és tökéletesen kapcsolódnak az információs hálózathoz. A nagy IKT (Info-Kommunikációs Technológia) szolgáltató cégek (pl. Google, Apple, Cisco stb.) az utóbbi években komoly üzleti döntéseket hoztak az IoT területén, ami egyben generálja és indokolja a témakör üzleti húzóerejét. A távközlési vállalatok szerint a gép-gép (M2M – Machine-to-Machine), illetve IoT rendszerek az üzlet központi tényezőjévé válnak azért, hogy a hálózatba kapcsolt tárgy objektumok száma lényeges és gyorsuló növekedést mutat.

Az IoT fejlődése más olyan területekkel való viszonyát és kapcsolatát is hangsúlyosabbá teszi, mint a felhőszámítás (Cloud computing), Jövő Internet (Future Internet), Big Data, robotika és szemantikus technológiák. Ezek nem önálló területekként léteznek, hanem egyre inkább kiegészítik egymást. Komoly kihívást jelent a jelenlegi IoT számára a kapcsolt eszközöknek, objektumoknak, okos környezeteknek, szolgáltatásoknak és személyeknek dinamikus konfigurált platformok webjeként való kiterjesztése. Az ipari elemzők (pl. Acatech, Cisco, Ericsson, IDC, Forbes) az IoT fejlődését okos környezetekbe és okos platformokba integráltan képzelik el, ahol a „smart web of everything” a társadalmi és gazdasági változásokat okozó, következő időszak meghatározó koncepcióját jelenti. A következő néhány évben az innovatív ökoszisztémák lényegesen meg fogják haladni az okos telefonok, tablet gépek és más elektronikai eszközök mai piaci potenciálját. Az IoT számára megoldásra váró feladatok a következők: i) A hátralévő technológiai korlátok kiküszöbölése, különösen a biztonsági szemszögből: a nagyszámú IoT eszköz ellenőrzése automatikusan és egységesen kell, hogy megtörténjen. ii) Az IoT integrációs potenciál feltárása a felhőszámítással és a Big Data megközelítéssel. iii) Felhasználók részéről való befogadó készség, életképesség validálása: a ma még nem létező alkalmazások kutatása (pl. autók közötti közvetlen kommunikáció). iv) Szenzor és platform felületek innovációja: a Jövő Internet pilot tevékenységek támogatják az egyéni innovációt és az okos objektumokkal való kommunikáció könnyítését. v) Athidaló használat kihívásainak demonstrációja: alap technológiák többféle környezetben való alkalmazhatósága, mint amilyen az okos otthonok, okos gyártási folyamatok, dedikált okos városrészek stb.

A fenti problémák közül a szenzorokból kinyert hatalmas adathalmaz kezelhetőségének és információ tartalmának kinyerésével foglalkozunk a jelen cikkben. A második fejezetben azokat a megoldásokat ismertetjük, amelyeket az adatklaszter analízis alkalmaz változókból alkotott csoportok létrehozása céljából. A harmadik fejezet a wavelet transzformáció tulajdonságait mutatja be annak érdekében, hogy szokatlan események beazonosítását elvégezhessük. A negyedik fejezetben az ismertetett módszereket különböző produktív rendszerekből kinyert adathalmazokra alkalmazzuk és megállapításokat teszünk a megfigyelt folyamatok jellemzőire vonatkozóan. Ezt az összefoglalás követi, ahol a témával kapcsolatosan további elemzési lehetőségeket említünk meg.

2. Klaszteranalízis és adatklaszterezési módszerek

A klaszteranalízis objektumok halmazából hasonló elemeket csoportosít klaszterekbe [1] [2]. A csoportosítás eredményeként a hasonló objektumok azonos csoportba kerülnek, a kialakított csoportok pedig eltérőek lesznek. A csoportok kialakításának módja nem csak egyetlen algoritmus alapján lehetséges. A módszerek széles köre ismeretes, ami az objektumok eltérő csoportosítási megoldásához vezet. A klaszter kialakításához elterjedten alkalmazzák az objektum elemek közötti távolság fogalmát, az adat terek sűrű területét, az intervallumokat vagy a statisztikai eloszlásokat is. Ezáltal a klaszterezés felfogható egy többcélú optimalizációs feladatnak is. Az alkalmazott klaszterezési algoritmus és a paramétereik meghatározása nagymértékben függ az elemzett adathalmaz tulajdonságaitól. A paraméterek közé tartozik az alkalmazott távolság függvény, a sűrűsödési küszöbérték, valamint a becsült klaszterek darabszáma. A klaszterek kialakításának mechanizmusa egy iterációs folyamat, ami által egy interaktív, többobjektumos optimalizációs feladat megoldására történik próbálkozás. Ez a tevékenység az eredményen túlmenően magába foglalja úgy a próbálkozást, mint az esetleges sikertelenséget is.

A különböző klaszter modellek között a legfontosabbakat említjük meg: i) Távolság kapcsolódási modellek, amelyek közül a legismertebb a hierarchikus klaszterezés; ii) Középpont modellek, pl. k-átlag algoritmus, ami minden klasztert egy átlag vektorral reprezentál; iii) Eloszlás alapú modellek, ahol olyan statisztikai eloszlás modelleket alkalmaznak, mint a többváltozós normális eloszlás, illetve a várhatóérték maximalizálás algoritmus; iv) Sűrűség alapú modellek, amelyek sűrű tartományok beazonosítását végzik (pl. DBSCAN vagy annak továbbfejlesztett változata, az OPTICS). Az hard (erős) klaszterezés minden egyes objektumot csoportba sorol vagy sem, míg a soft (enyhe) klaszterezésnél mindegyik objektum bizonyos klaszterekbe jól meghatározott mértékig tartozik. A szigorú partícionálási klaszterezés mindegyik objektumot pontosan egyetlen csoportba sorolja be. Ennek egy továbbfejlesztett változata beazonosítja a nem csoportosítható elemeket is és kívülállóként (outlier) kezeli. A hierarchikus klaszterezésnél az objektum a csoporthoz és annak szülő klaszteréhez is tartozik.

A klaszterezési algoritmusok száz feletti száma miatt csak a legelterjedtebbeket említjük. Adott csoportosítási feladat elvégzéséhez a leginkább találó algoritmus kiválasztása próbálgatások útján történik. Hierarchikus klaszterezésnél az alapfeltételezés szerint a közelebb lévő objektumok azonos, míg a távoliak különböző csoportba tartoznak. Itt definiálni kell a távolság függvényt, ami a kapcsolódás mértékét képes mérni. A hierarchikus klaszterezés lehet tömörülő vagy megoszló. Előbbinél egy elemmel kezdődik a klaszter és további elemek kerülnek bele, míg utóbbinál kezdetben minden elem ugyanabba a klaszterbe tartozik, majd partíciók kialakítására kerül sor. A középpont csoportosításnál a klasztert egy középponti vektor képviseli, amely nem feltétlenül tartozik a halmazhoz. A k-átlag algoritmus a k elemű halmaz középpontját határozza meg. Az új elem ahhoz a klaszterhez fog tartozni, amelynek középpontja a legkisebb távolságra helyezkedik el az objektumtól. Az eloszlás alapú klaszterezésnél a leginkább hasonló eloszlású objektumok kerülnek azonos csoportba. Itt sajnos bekövetkezhet hibás illeszkedés is, ami az algoritmus komplexitásának növelését teszi szükségessé. Ez természetesen számolási többlet kapacitást igényel, azaz egy költségesebb megoldás. A sűrűség alapú modelleknél a legsűrűbb fizikai tartományok képezik a klasztereket.

2.1. DBSCAN algoritmus

A sűrűség alapú modellek legismertebb módszere a DBSCAN (Density-based spatial clustering of applications with noise) [3]. Egy A pont közvetlenül elérhető B pontból, ha az

attól mért távolsága kisebb, mint EPS és ha A pont körül elegendő számú, ehhez a klaszterhez tartozó elem létezik. B pontot A ponttól sűrű-elérhetőnek nevezzük, ha létezik A_1, \dots, A_n pontsorozat ($A_1=A$ és $A_n=B$), ahol A_{i+1} közvetlenül sűrűség-elérhető A_i – ből. A fenti definíció nem szimmetrikus. Aszimmetria miatt a sűrű-kapcsolat fogalom bevetésére van szükség. A és B pontok sűrű-kapcsolatban állnak, ha létezik egy olyan köztes C pont, amelytől A és B sűrű-elérhető. A sűrű-kapcsolat szimmetrikus viszony. A DBSCAN algoritmussal meghatározott klaszter két tulajdonságot teljesít: 1) a klaszter elemek kölcsönösen sűrű-elérhetőek; 2) ha egy elem sűrű-elérhető a klaszter bármelyik elemétől, akkor része a klaszternek. Az algoritmus nem tesz különbséget a határ-, illetve zajobjektumok között [4]. A klaszter csak kölcsönösen sűrű-kapcsolatú objektumokat tartalmaz. Ez a megfogalmazás a sűrű-kapcsolatú elemek statisztikai értelmezéséhez közelebb áll. A létrehozott klaszter ezáltal több zajjelemtet fog tartalmazni.

Az algoritmus két kezdőparaméterrel rendelkezik: 1) szomszédsági távolság, EPS; 2) sűrű régió kialakulásához szükséges elemek minimális száma, minPts. Az algoritmus tetszőleges, meg nem látogatott elemmel kezdődik. Ennek az elemnek az EPS környezetét vizsgáljuk, és ha találunk elegendő elemet, akkor belőle klasztert hozunk létre. Ellenkező esetben az elem zajkategóriába kerül. A későbbiekben a feltételek teljesülése esetén ez a pont bekerülhet más klaszterbe. Ha egy pont egy klaszter sűrű részéhez tartozik, akkor annak EPS-környezete is része ennek a klaszternek. Így az EPS környezetben lévő összes elem bekerül a klaszterbe, ha ezek ugyanakkor sűrűk. A folyamat addig ismétlődik, amíg a sűrű-kapcsolatú klaszter teljes mértékben előáll. Ezután egy másik nem-látogatott elem vizsgálatára kerül sor, aminek eredményeként újabb klaszter vagy zaj beazonosítása történik meg. Indexelt struktúra esetén az algoritmus komplexitása $\sim O(n \log n)$, egyébként $\sim O(n^2)$. Az igényelt memória kapacitás ugyancsak $\sim O(n^2)$, de skaláris elemeknél ez $\sim O(n)$ értékre csökken. Az algoritmus előnye, hogy nem kell megadni a keresett klaszterek számát, tetszőlegesen szétterült klasztereket is be tud azonosítani. Ugyancsak be tudja azonosítani a kívülálló elemeket a zaj kategóriával. A módszernek csak két paramétere van és nem érzékeny az elemek sorrendjére.

Az algoritmus hátrányaként azonban meg kell említeni, hogy nem teljesen determinisztikus, ugyanis a határvonalon lévő elemek bármely klaszterhez tartozhatnak. A klaszterezés minősége függ a távolság metrika típusától. Itt leggyakrabban az Euklideszi távolságot alkalmazzák. Sokdimenziós adat esetén nehéz becsülni az EPS paramétert, mivel sok adatkomponens befolyásolja a távolságfüggvény eredményét. A nagy eltérésű elemeket tartalmazó halmaz esetén az adatok klaszterezése nem hatékony, mivel a (MinPts, EPS) paraméterpár megválasztása az összes klaszter számára nem egyformán optimális. Tapasztalat alapján a $\text{minPts} \geq D + 1$, ahol D az adatok dimenziója. Az EPS meghatározásához a $k = \text{minPts}$ görbe inflexiós pontját használjuk fel. Általánosan érvényes, hogy kis EPS esetén sok elem zaj kategóriába kerül, míg nagy EPS mellett a szignifikáns klaszterek összeolvadnak.

3. Wavelet transzformáció és tulajdonságai

Egy $f(t)$ jel

$$\{T_f(a,b) = \langle Y, \Psi_{a,b} \rangle, a \in \mathbb{R}^+, b \in \mathbb{R}\} \quad (1)$$

típusú felbontását CWT (Continuous Wavelet Transform), folytonos wavelet transzformációnak nevezzük, ahol a $C_{a,b}$ együtthatókat az elemző függvény és az $f(t)$, $t \in \mathbb{R}$, jelfüggvény belső szorzatából állítjuk elő:

$$C_{a,b}(f(t), \Psi_{a,b}(t)) = \int f(t) \Psi_{a,b}(t) dt \quad (2)$$

A $\Psi_{a,b}$ függvényeket egy Ψ_0 referencia, u.n. anya waveletből állítják elő az alábbi módon:

$$\{\Psi_{a,b}(t) = 1/\sqrt{a} \Psi_0((t-b)/a)\} \quad (3)$$

Bármely kétváltozós $\Psi_{a,b}(t)$ függvény előállítható Ψ_0 anya waveletből egy D_a , dilatációs (skála) és egy τ_b , időeltolás operátor segítségével:

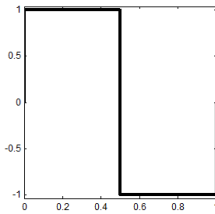
$$(\tau_b) \equiv \Psi_0(t-b) \quad (4)$$

$$(D_a) \equiv 1/\sqrt{a} \Psi_0(t/a) \quad (5)$$

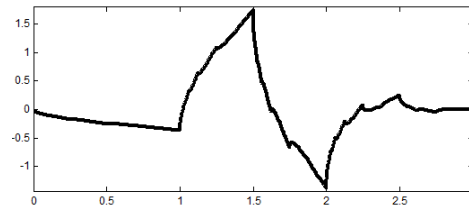
A Ψ_0 anya wavelet (pl. DaubechiesN, CoifletN, HaarN, SymmletN stb.) tulajdonsága, hogy négyzetesen integrálható és átlaga nulla, azaz (ld. 1-3 ábrák):

$$\int (\Psi_0(t))^2 dt < \infty, \quad (6)$$

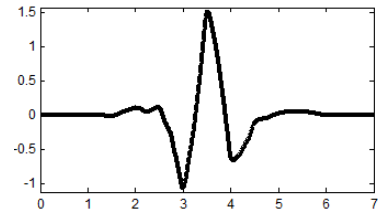
$$\int \Psi_0(t) dt = 0. \quad (7)$$



1. ábra. Haar wavelet



2. ábra. Db2 wavelet



3. ábra. Sym4 wavelet

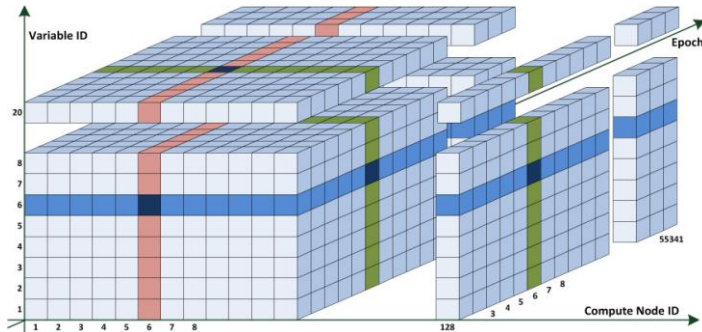
A folytonos wavelet transzformáció képes detektálni a hirtelen változásokat, mivel ilyen időpillanatban a $C_{a,b}$ együtthatók nagy abszolút értékűek. Fontos, hogy a $C_{a,b}$ wavelet együtthatók az elemzett függvény szakadási pontjait az alacsony skálákon képesek érzékelni. Ugyanakkor, az elemzett függvény szakadási pontjai a $C_{a,b}$ együtthatóknak csak egy szűk körét befolyásolják [5].

4. Produkciós rendszerekből nyert szenzoradatok elemzése

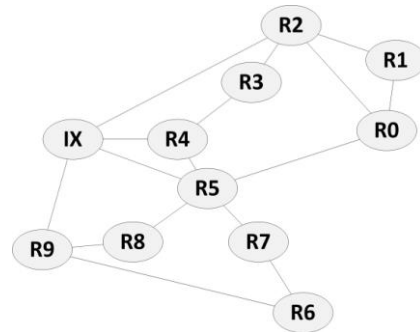
Szenzorokból kinyert két fajta adathalmaz elemzését végeztük el. A két megfigyelt rendszer funkciója lényegesen eltér egymástól. Hasonlóság a kettő között az, hogy mindkettő sok csomópontból álló hálózati rendszer, amelyeknek fizikai, illetve logikai változóit, mint állapot jellemzőt lehetséges mérni.

Első esetben egy $s = 128$ számoló csomóponttal rendelkező, 18 TFLOP/s számolási teljesítményű MPP (Massive Parallel Processing) architektúrájú szuperszámítógép (HPC – High Performance Computer), azaz szorosan csatolt sokprocesszoros rendszer. Ennek csomópontonkénti $m = 20$ különböző állapot-adatsorait elemeztük [6] [7]. A mintavételezések száma $N = 55.341$, amit $T = 10$ s periodikus (epoch) időközönként, az erre a célra dedikált hardver alrendszerrel nyertünk ki. Minden egyes periódusidőben $s \cdot m = 2560$ változóérték begyűjtése történt. Ezt N -szer ismételtük összesen, így a mérés 2013. szeptemberben közel egy hétig tartott (ld. 5. ábra). Minden egyes CN (CN – Compute Node) számoló csomópont

két darab hatmagos processzorral, illetve magonként 4 GB RAM memória DIMM sorral rendelkezik, ezáltal a HPC rendszer operatív memóriájának kapacitása 6 TB. A változók közül az első tizenegy virtuális szenzoroktól, a maradék kilenc pedig fizikai szenzoroktól származott.



5. ábra. HPC állapot-adatok



6. WAN topológia

Az ütemező Parallel.Q várakozási sorához tartozó 120 darab számoló csomópontjára alkalmaztuk a DBSCAN adatklaszterezési algoritmust. A HPC gép maradék nyolc számoló csomópontja a Serial.Q (7 db), illetve Test.Q (1 db) várakozási sorok job-jait futtatta. A változók így kapott adatklaszterei az alábbiak:

1. Táblázat: HPC állapot-változók klasztereződése

Klaszter	Változó neve
C1	1:Load_1min, 2:Load_5min, 3:Proc_run, 4:Proc_total, 10:CPU_user
C2	5:Pkt_in, 6:Pkt_out, 8:Bytes_out
C3	9:Memory_free
C4	7:Bytes_in, 11:CPU_system
C5	12:System_temp
C6	13:CPU1_temp, 14:CPU2_Temp, 17:P1_DIMM3A_temp
C7	15:P1_DIMM1A_temp, 16:P1_DIMM2A_temp, 17:P1_DIMM3A_temp
C8	18:P2_DIMM1A_temp, 19:P2_DIMM2A_temp, 20:P2_DIMM3A_temp

Megfigyelhető, hogy a logikai szenzoroktól származó adatsorok egymás között alakítanak ki adatklasztert. Ugyanez a csoportosulási hajlam igaz a fizikai szenzoroktól származó adatsorokra is. A processzorok terhelését a felhasználói módban futó programok okozták (C1). Meg kell azonban említeni, hogy a CN számoló csomópontok HPC hálózati rendszeren belüli forgalma két klaszterben is megjelenik (C2, C4). A fel nem használt memória kapacitás önálló erőforrásként működik (C3). A számoló csomópont alaplapjának hőmérséklete, valamint a processzorok, illetve DIMM RAM sorok hőmérséklete külön klasztereződik, ami a processzorok egyedi, eltérő mértékű terhelését igazolja (C5, C6-C8).

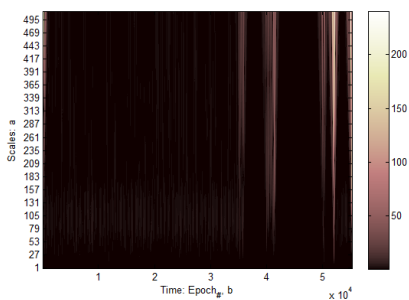
A második esetben a magyar WAN (Wide Area Network) országos akadémiai Internet hálózat $s = 10$ darab vidéki, nagyteljesítményű útválasztó (router) csomópontját, mint lazáncsatolt sokprocesszoros rendszert vizsgáltuk. Az Internet kijáratot az IX (Internet eXchange) csomópont biztosítja. Az állapot-adatsorok csomópontjai egyenként $m = 45$ darab változó lekérdezését teszik lehetővé SNMP (Simple Network Management Protocol) alapú belsőmenedzsment felületen (ld. 6. ábra). Minden egyes útválasztó megfelelő controllerrel és több 10 GE interfésszel és rendelkeznek. Interfész szinten képesek mérni irányonként a hálózati forgalmat (egyes/többes/üzenetszórás-szám, bájtszám, keretszám), valamint eszköz és interfész szinten az energiafogyasztást és a hőmérsékletet és is. A hálózaton mozgatott

állapot-adatok mennyisége az interfészek sebességéhez képest négy nagyságrenddel kisebb, vagyis az elemzett rendszert gyakorlatilag nem befolyásolja a mérési folyamat. Az epoch időtartamok $T = 30$ s és a lekérdezések száma $N = 23.664$, ami 2014. áprilisban több mint nyolc napig tartott. Itt is alkalmaztuk a DBSCAN algoritmust az állapotváltozók klaszterezése érdekében és az alábbi csoportosítást kaptuk:

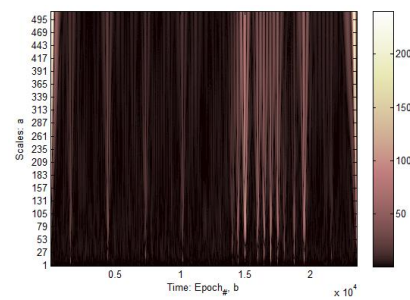
2. Táblázat: WAN router állapot-változók klasztereződése

Klaszter	Változó neve
C1	1:CPU1_Load1min, 2:CPU2_Load1min, 3:CPU3_Load1min, 4:CPU4_Load1min, 5:System_temp
C2	6:IF1_InUcastPkts, 7:IF1_InMulticastPkts, 8:IF1_InBroadcastPkts, 9:IF1_InOctets, 10:IF1_OutUcastPkts, 11:IF1_OutMulticastPkts, 12:IF1_OutBroadcastPkts, 13:IF1_OutOctets, 14:IF1_Power, 15:IF1_Temp
C3	16:IF2_InUcastPkts, 17:IF2_InMulticastPkts, 18:IF2_InBroadcastPkts, 19:IF2_InOctets, 20:IF2_OutUcastPkts, 21:IF2_OutMulticastPkts, 22:IF2_OutBroadcastPkts, 23:IF2_OutOctets, 24:IF2_Power, 25:IF2_Temp
C4	26:IF3_InUcastPkts, 27:IF3_InMulticastPkts, 28:IF3_InBroadcastPkts, 29:IF3_InOctets, 30:IF3_OutUcastPkts, 31:IF3_OutMulticastPkts, 32:IF3_OutBroadcastPkts, 33:IF3_OutOctets, 34:IF3_Power, 35:IF3_Temp
C5	36:IF4_InUcastPkts, 37:IF4_InMulticastPkts, 38:IF4_InBroadcastPkts, 39:IF4_InOctets, 40:IF4_OutUcastPkts, 41:IF4_OutMulticastPkts, 42:IF4_OutBroadcastPkts, 43:IF4_OutOctets, 44:IF4_Power, 45:IF4_Temp

A router állapot-változók klasztereződése az interfészek, illetve a globális jellemzők alapján történik. Ez várható is volt, hiszen a WAN útválasztón áthaladó adatfoglalom az adott interfészen belül az összes változót befolyásolja. Az alkalmazások interaktivitása miatt adott interfészen kifelé, illetve befelé haladó forgalmak között korreláció létezik, ezért a két irány szerinti változók azonos adatklaszterbe kerültek. A router berendezés önmaga is egy többprocesszoros kontrollerrel rendelkezik. Ezek SMP (Symmetric Multi Processing), azaz szimmetrikus terhelés-megosztási mechanizmus alapján futtatják a routing processzeket, így a vizsgált útválasztókban mind a négy CPU (Central Processing Unit, processzor) közel azonos viselkedésű.



7. ábra. CN₁ node Load1min meglepetésszerű állapotváltozásai



8. ábra. R0 router CPU₁_Load1min meglepetésszerű állapotváltozásai

Mindkét mérési környezetben egy-egy klaszter reprezentánsra alkalmazott CWT, folytonos wavelet módszer azt mutatja, hogy a végrehajtó csomópontok (CN, illetve router) igénybevétele borsztös jellegű, ami meglepetésszerű terheléseket okoz a kiszolgáló rendszer működésénél (ld. 7. ábra és 8. ábra). Ezt a CWT ábrák világos függőleges sávjai azonosítják. A vízszintes tengelyen ezáltal megtalálhatók azok az epoch idők, ahol az adott rendszer működésénél lényeges változás következett be. Ilyen időszakokban a tanulmányozott rendszer a normálisnál alacsonyabb hatásfokkal működik.

5. Összefoglalás

Mindkét tanulmányozott rendszer sokkal több állapot-változóval rendelkezik, mint amennyiből az adott rendszer érdemi működésére vonatkozóan információt lehetne kinyerni ezen az elemzési szinten. A HPC változók esetén adatklaszterezéssel 60%-os adatmennyiség csökkentést lehet elérni. Ez a módszer WAN hálózatnál 89%-os adatmennyiség csökkenést képes okozni. Mindkét rendszernél a „Big Data” probléma szignifikáns mértékű kezelhetőségét bizonyítja a DBSCAN algoritmus alkalmazása. A CWT, folytonos wavelet transzformáció jól azonosítja a meglepetésszerű eseményeket. A WAN router rendszer esetén gyakrabban bekövetkeznek ilyen események, ugyanakkor a mérési időtartam első felében periodikus terhelésnövekedés is megfigyelhető. Ezt az Interneten továbbított periodikus archiválási adatfolyamok okozhatják.

A kutatott témakörhöz kapcsolódóan további kérdésekként fogalmazódnak meg a hatékonysági lehetőségek vizsgálata, valamint a DBSCAN módszernél alkalmazott távolság metrika módosításának hatása az adatklaszterek kialakulására vonatkozóan. A wavelet transzformációval beazonosított meglepetés szerű események előtti epoch-ok időben további elemzések szükségesek, ami a váratlan esemény kialakulásának magyarázatát teszi lehetővé. Ezáltal időben detektálhatóvá válik a különböző rendszerek normálistól eltérő üzemmódjának kialakulási tendenciája, így a szükséges vezérlési ellenutasítások kiadására is mód nyílik.

6. Köszönetnyilvánítás

A publikáció elkészítését részben a TÁMOP-4.2.2.C-11/1/KONV-2012-0001számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

Jelen kutatást részben a Szuperszámítógép, a nemzeti virtuális laboratórium című, TÁMOP-4.2.2.C-11/1/KONV-2012-0010 azonosítószámú projekt támogatta az Európai Unió és az Európai Szociális Alap társfinanszírozása mellett.

Irodalomjegyzék

- [1] Bailey, Ken: Numerical Taxonomy and Cluster Analysis. Typologies and Taxonomies. p. 34. ISBN 9780803952591, (1993).
- [2] Estivill-Castro, Vladimir: Why so many clustering algorithms — A Position Paper. ACM SIGKDD Explorations Newsletter 4 (1): 65–75. doi:10.1145/568574.568575, (2002).
- [3] Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; Xu, Xiaowei: A density-based algorithm for discovering clusters in large spatial databases with noise. In Simoudis, Evangelos; Han, Jiawei; Fayyad, Usama M. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226–231. ISBN 1-57735-004-9. CiteSeerX: 10.1.1.71.1980, (1996).
- [4] Campello, R. J. G. B.; Moulavi, D.; Sander, J.: Density-Based Clustering Based on Hierarchical Density Estimates. Proceedings of the 17th Pacific-Asia Conference on Knowledge Discovery in Databases, PAKDD 2013. Lecture Notes in Computer Science 7819. p. 160. doi:10.1007/978-3-642-37456-2_14. ISBN 978-3-642-37455-5, (2013).

- [5] Zoltan Gal, Gyorgy Terdik): Statistical Analysis of Next Generation Network Traffics Based on Wavelets and Transformation ON/(ON+OFF), Applied Computation Intelligence in Engineering and Information Technology, Springer-Verlag Berlin Heidelberg, DOI 10.1007/978-3-642-28305-5, pp. 107-122, (2012).
- [6] Zoltan Gal, Tibor Tajti: Complex Event Processing in Supercomputer Environment: Sensor and Neural Network Based Analysis, Proceedings of IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom 2013), Budapest, Hungary, December 2-5, 2013, ISBN: 978-1-4799-1544-6, pp 735-740, (2013).
- [7] Gyorgy Terdik, Zoltan Gal: Advances and practice in Internet of Things: A case study, Proceedings of IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom 2013), Budapest, Hungary, December 2-5, 2013, ISBN: 978-1-4799-1544-6, pp 435-440, (2013).

Intelligens egyetemi campus

Intelligent Campus

Kádek Tamás, Kollár Lajos, Kósa Márk

Debreceni Egyetem, Informatikai Kar

kadek.tamas@inf.unideb.hu, kosa.mark@inf.unideb.hu, kollarl@inf.unideb.hu

Absztrakt: Az intelligens közösség, a „smart community” többet jelent annál, mint hogy a közösség működésének információi és a közösség számára hasznos szolgáltatások elérhetőek online (web 2.0) formában. Szükség van az ezek használatából keletkező információk gyűjtésére és elemzésére annak érdekében, hogy képet kapva a közösség működéséről, mindezt újabb szolgáltatások formájában visszaforgathassuk a közösség életébe. Nekünk, egyetemi oktatóknak kézenfekvő, hogy intelligens közösségként tekintünk a karunkon tanulóokra és dolgozókra, az egyetemi campuson tevékenykedőkre.

Cikkünkben bemutatjuk a smartcampus.hu alkalmazást, melyet abból a célból fejlesztünk, hogy egyetlen helyen gyűjtsük össze a karunkat érintő fontos információkat, ezeket számos formában elérhetővé téve a közösség számára online formában, az egyszerű weboldaltól egészen a webszolgáltatások gyűjteményéig. Néhány egyszerű alkalmazáson keresztül példát mutatunk rá, hogy a smartcampus.hu alkalmazás használatából származó információ hogyan forgatható vissza a közösség életébe értéknövelt, intelligens, „smart” szolgáltatásként.

Kulcsszavak: intelligens közösség, intelligens kampusz, intelligens szolgáltatások

Abstract: Intelligent, or smart, community means more than only making information about the operation of the community available in an online form (via web 2.0) along with several useful services for the community. There is a need to collect and analyze information generated from the use of such services in order to get a picture of the community's operation. These pieces of information can serve as a feedback based on which further services can be offered. We consider students and employees of our faculty as a smart community.

Intelligent, or smart, community means more than only making information about the operation of the community available in an online form (via web 2.0) along with several useful services for the community. There is a need to collect and analyze information generated from the use of such services in order to get a picture of the community's operation. These pieces of information can serve as a feedback based on which further services can be offered. We consider students and employees of our faculty as a smart community.

Keywords: intelligent community, intelligent campus, intelligent services

1. Intelligens közösségek

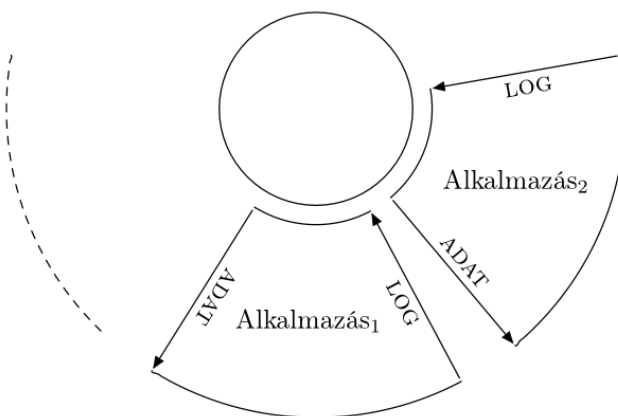
Az intelligens közösség fogalmának körbejárásakor jelen cikkünkben a közösségben működő szoftverekkel kapcsolatos elvárásokra koncentrálnak. Ebben a megközelítésben az intelligens közösség olyan közösség, amelyet intelligens alkalmazások szolgálnak ki. Igazodva az „Apps for Smart Cities Manifesto” intelligens városokról alkotott definíciójához [1, 2], az alábbiakban felsoroljuk egy intelligens közösségi alkalmazással szemben támasztott elvárásokat:

- érzékelési képesség: az alkalmazás szenzorok segítségével érzékeli a környezetét,

- kapcsolódási képesség: hálózati eszközök segítségével az észlelt információkat juttassa el a webre,
- hozzáférhetőség: a megosztott információkat tegye közzé a felhasználóknak,
- elérhetőség: a felhasználó férhessen hozzá az információhoz a weben, vagy még inkább mobil eszközökön keresztül bármikor és bárhol,
- közösségi szemlélet: a felhasználó publikálhassa az információt a közösségi hálójában,
- megoszthatóság: ne csak az adatok, hanem maguk az objektumok is legyenek hozzáférhetőek és megcímezhetőek,
- láthatóság: a fizikai környezet átalakításával váljanak láthatóvá a rejtett információk.

Egy-egy alkalmazás, mely részben vagy egészben megfelel a fent megfogalmazott kritériumoknak, önmagában még nem alkot intelligens szolgáltatást. Ugyanakkor az intelligens szolgáltatás alapvető feltétele, hogy információval rendelkezünk a közösségről. Elsőként tehát a rendelkezésre álló információkat be kell gyűjteni és közzé kell tenni. A begyűjtött információ a jelenlegi technikai fejlettség mellett igen nagy tömegű és heterogén. Egy közepes teljesítményű okostelefon (és ezzel együtt a rajta lévő szoftver) is képes GPS-alapú helymeghatározásra, mobil és wifi hálózati eszközök érzékelésére, fénymérésre, képrögzítésre és még sorolhatnánk. Persze a szenzorok nem minden esetben a végfelhasználók eszközeiben találhatóak, gondoljunk csak a szabadparkolóhely-számláló berendezésre, digitális hőmérőre vagy akár az elektronikus beléptető rendszerre. Ezek is mind-mind olyan információk, amelyek felhasználásával szolgáltatások készíthetők.

Sok esetben a problémát épp az okozza, hogy túl sok információ áll rendelkezésre, ezeket (átmenetileg) az adatokat begyűjtő eszközön tárolni kell. A cél persze az, hogy minél előbb megosztásra kerüljenek az adatok. A megosztás színtere pedig a web. Itt nem térnénk ki arra, hogy a begyűjtött adatokat aggregálni és szűrni célszerű, sőt olykor az anonimitás biztosítása érdekében kötelező. Legyen most elég annyi, hogy az intelligens közösségi alkalmazások számára gyűjtött adatok tárolásáról és kezeléséről lehetőleg központosított, vagy még inkább egységesített formában gondoskodni kell, mert ezek az adatok képezik az intelligens szolgáltatások alapját.



1. ábra

Az intelligens közösség nem statikus, hanem állandóan változik, folytonos mozgásban van. Időről időre változik a begyűjtött információk jellege és mennyisége. Az intelligens alkalmazások fejlődése pedig egy végtelen folyamat. Új alkalmazások jöhetnek létre, melyek puszta léte (a felhasználásuk mikéntjéből adódó információ) is bővíti mindazt, amit érzékelni lehet. A közösség működéséről így ez által is új adatok kerülhetnek a birtokunkba. Az új adatokból ismét megtudhatunk valamit a közösség működéséről, amelyet aztán akár új alkalmazások formájában forgatva vissza a közösség életébe, alakíthatjuk tovább annak működését.

Az intelligens szolgáltatás életciklusa

Megközelítésünkben az intelligens közösség alappillére, hogy a közösség szolgáltatásainak online elérhetőnek kell lenniük. Először gyűjtjük tehát össze online elérhető és kereshető formában azokat az információkat, amelyekre a közösség tagjainak szüksége lehet. Ez egyszerűen adatszolgáltatás kialakítását jelenti. Ilyen lehet például hírek gyűjteménye, egy menetrend vagy akár az órarendi adatok összessége. Második lépésben figyeljük meg az elérhetővé tett információ felhasználását. Nézzük meg például, hogy mely hírekre voltak a közösség tagjai leginkább kíváncsiak, milyen rovatokra van a közösségnek szüksége. Majd a megfigyelésből levont következtetések felhasználásával fejlesszük tovább a szolgáltatást, szervezzük át az információt. Maradva az iménti példánál, egy nagyon egyszerű szolgáltatás, amelyet a hírolvasottsági statisztikát felhasználva készíthetünk el, a népszerű rovatok kigyűjtése (esetleg személyre szabva minden egyes olvasónak). Nem akarjuk persze azt a látszatot kelteni, hogy mintha így máris intelligens alkalmazást készítettünk volna. Ez önmagában még korántsem elég. Amikor intelligens szolgáltatásról beszélünk, nem egyetlen alkalmazást kell magunk elé képzelnünk, hanem együttműködésre képes, egymással információkat megosztó alkalmazások állandóan fejlődő és bővülő körét.

Az intelligens platform

Ha az intelligens szolgáltatást egyetlen mondatban egymással együttműködni képes, egymással adatokat megosztó, és ezen adatokat felhasználó alkalmazások gyűjteményeként írtuk le, akkor az intelligens platform az a közös nyelv vagy interfész, amelynek segítségével az emített együttműködés lezajlik. Ugyanakkor itt sem csupán arról van szó, hogy az adatok begyűjtését, tárolását és megosztását biztosítani kell. Önmagában nem könnyű feladat ez sem, hiszen esetenként igen nagy tömegű adat beáramlásáról és aggregálásáról lehet szó, ráadásul sok esetben az anonimitás biztosítása legalább ilyen fontos. A platform azzal, hogy lehetővé teszi az adatok koncentrált elérését, lehetőséget biztosít azok elemzésére, különböző analitikák kidolgozására, melyek a közösségben megjelenő új szolgáltatások alapját képezik. Minthogy kritérium az online működés, a platformnak webes elérést kell biztosítania. Elegendő most csak két tipikus példát említeni: a távoli eljárás hívatást lehetővé tevő SOAP alapú webszolgáltatásokat és az eseményszemléletű XMPP protokollt mint a platform szinte kötelező elemét.

Az analitikák feladata a begyűjtött információ csoportosítása és új következtetések levonása a közösség egészének vagy tagjainak a viselkedéséről. Szűkebb értelemben ettől lesz intelligens a szolgáltatás, vagy más szavakkal, az analitikára építve lehet intelligens szolgáltatást készíteni.

2. Intelligens campus

Intelligens campus alatt olyan szoftverek gyűjteményét, együttesét értjük, amelyek kielégítik az intelligens közösségi alkalmazásokkal szemben támasztott követelményeket, szolgáltatásaik az egyetemi élethez szorosan kötődnek, felhasználóik nagy többsége pedig az egyetemi hallgatók, oktatók és egyéb egyetemi alkalmazottak közül kerül ki.

Egy egyetemi közösség élete sok tekintetben eleve online szerveződik. Gondoljunk csak az elektronikus tanulmányi rendszerre, a teremfoglalásokra, és a különböző szervezeti egységek információs weblapjaira. Ugyanakkor első ránézésre is egyértelmű, hogy az online elérés még nem elegendő. Az eleve online formában adott szolgáltatások többsége ugyanis közvetlenül a végfelhasználónak szól, nem könnyítve meg ezzel az elektronikus feldolgozást. Hiányzik a közös platform, már csak amiatt is, hogy a meglévő információk valamilyen szoftverrel egyszerűen elérhetők legyenek. Az első lépés ebből adódóan az intelligens platform megalapozása. Kezdetben ez csupán annyit jelent, hogy a rendelkezésre álló információkat egyetlen közös, szabványos protokollokon keresztül hozzáférhető felületre kell közzétenni. Ebből a célból készült a Debreceni Egyetemen a smartcampus.hu alkalmazás, mely a kar szolgáltatásait kívánja közös felületen megjeleníteni, nem titkoltan azzal a céllal, hogy helyet adva újabb – rá épülő – alkalmazásoknak, szolgáltatson információt a későbbi analitikák számára.

Intelligens naptár

Az intelligens kari naptár egyszerű példája jól mutatja az intelligens platform létrehozásának első lépéseit. Az alkalmazás segítségével kezdetben egyetlen felületre gyűjtöttük össze a kar életében szerepet játszó fontos eseményeket és határidőket, melyek ugyan már korábban is rendelkezésre álltak, de több különböző adatforrásból voltak elérhetők. A naptáralkalmazáshoz mint platformhoz (webszolgáltatás-gyűjteményhez) kapcsolódó kiegészítő alkalmazások segítségével gyűjtött adatok szolgálnak arra, hogy a közösség működéséről szerezzünk információt. Például, ha egy esemény – mondjuk a Tanulmányi Osztály fogadóórája – kiugróan nagy érdeklődésre tart számot, akkor az valószínűleg túlterhelt, ami fontos és értékes visszajelzés az esemény szervezőjének, és idejében biztosít lehetőséget a beavatkozásra.

A ProgCont és az intelligens campus

A ProgCont rendszert [3, 4] alapvetően programozóversenyek szervezésére és lebonyolítására terveztük 2011-ben. Magját egy elektronikus feladatgyűjtemény, a hozzáférést lehetővé tevő és a beküldéseket regisztráló webalkalmazás, valamint a megoldásokat kiértékelő segédprogramok alkotják. Önmagában ez a rendszer sem nevezhető intelligensnek, még akkor sem, ha a feladatok nagy részét a nemzetközi programozóversenyek feladatgyűjteményeiből ültettük át hallgatóink számára, gyakorlási célokkal. A rendszer azonban a használata során számtalan olyan adatot gyűjtött össze, amelyek értelmes csoportosításával plusz információt tudunk szolgáltatni a felhasználóknak.

Egy ilyen, szinte magától értetődő adat a feladatok megoldásának a sikeressége, azaz az, hogy az egyes feladatokat hány próbálkozásból hány felhasználónak sikerült megoldania. A saját próbálkozásainak a számát mindenki ismeri, a plusz információt ebben az esetben az jelenti, ha a nagy tömeg összesített adatait is a felhasználó rendelkezésére bocsátjuk.

Egy másik hasznos információ lehet a feladatok sikeres megoldásainak a minőségi értékelése. Ez alatt azt kell érteni, hogy a beküldött és jónak ítélt programkódokat valamilyen további szempontrendszer alapján minősítjük, rangsoroljuk. Ez lehet valamilyen szoftvermetrika éppúgy, mint a program által a futás során felhasznált memória mérete, vagy éppen a program futási ideje. A rangsorokban elfoglalt helyezések további munkára sarkallhatják a dobogós helyekről lemaradó hallgatókat.

A ProgCont persze más módon is segítheti a hallgatók gyakorlatozását. Tematikus feladatgyűjteményként is funkcionál, azaz lehetőség van a hasonló típusú, témakörű problémák szerinti válogatásra. Kellő számú feladat és beküldött megoldás esetén a rendszer a bejelentkezett felhasználó tudásszintjének megfelelő megoldandó feladatot is ajánlhat. Ami a feladatok nehézségét illeti, kétféleképpen érdemes értékelni: egyfelől a sikeresség és sikertelenség lehet az objektív mérce, másfelől viszont a felhasználó is véleményezheti a feladatot, hogy mennyire érezte azt könnyűnek vagy nehéznek. Mindkét értékelés segítséget nyújthat a közösség többi tagjának a további gyakorláshoz.

3. Összegzés

Cikkünkben bemutattuk az intelligens közösségi alkalmazásokkal szemben támasztott követelményeket. Igyekeztünk aláhúzni, hogy egy intelligens szolgáltatás fejlesztése egy soha véget nem érő folyamat, hiszen a szolgáltatás megjelenése maga is kihat a közösségre, és akár újabb alkalmazások alapja lehet. A két röviden bemutatott alkalmazás azt példázza, hogy miként gondolható tovább egy-egy egyszerű szolgáltatás annak reményében, hogy később a közösséget gazdagító, valóban intelligens alkalmazássá válhasson. Bár még messze állunk attól, hogy valóban intelligens campusról beszélhessünk, a fejlesztés irányát már kijelöltük.

4. Köszönetnyilvánítás

A előadást a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú, a „Jövő Internet kutatások az elmélettől az alkalmazásig” című projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

TÁMOP-4.2.2.C-11/1/KONV-2012-0001

Jövő Internet kutatások az elmélettől az alkalmazásig

Nemzeti Fejlesztési Ügynökség
www.ujszchenyiterv.gov.hu
06 40 638 638



A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] The Apps for Smart Cities Manifesto, <http://www.appsforsmartcities.com/?q=manifesto>, letöltve 2014. 07. 22-én.
- [2] Mikel Emaldi, Oscar Peña, Jon Lázaro, Diego López-de-Ipiña, Sacha Vanhecke, Erik Mannens: To trust, or not to trust: Highlighting the need for data provenance in mobile apps for smart cities, *39th International Conference on Very Large Data Bases*, 2013, Riva del Garda, Trento, Olaszország.

- [3] Kádek Tamás, Kósa Márk, Pánovics János: Programozó versenyek támogatása webes alkalmazással, *XXI. Számítástechnika és Oktatás Konferencia*, 2011, Kolozsvár, Románia, 184–187.
- [4] Kádek Tamás, Kósa Márk, Pánovics János: A ProgCont szoftverrel támogatott programozó versenyek tapasztalatai, *New Technologies in Science, Research and Education, XXV. DIDMATTECH Konferencia*, 2012, Révkomárom, Szlovákia, 152–157.

Smart Campus alkalmazás fejlesztése Android platformon

Development of a Smart Campus application on Android platform

SOMODI Krisztián, KÁDEK Tamás, KOLLÁR Lajos, KÓSA Márk

Debreceni Egyetem, Informatikai Kar
somodikrisztian1@gmail.com, {kadek.tamas,kollarl,kosa.mark}@inf.unideb.hu

Absztrakt: Cikkünkben a Debreceni Egyetem Informatikai Karán futó FIRST projekt keretében létrehozott androidos Smart Campus alkalmazás tervezése és megvalósítása során felmerült kérdéseket járjuk körül. Bemutatjuk a Smart Campus projekt felépítését, az architektúrát és a felhasznált technológiákat, és ismertetjük, hogy milyen indokok alapján döntöttünk natív alkalmazás kifejlesztése mellett. A kifejlesztett alkalmazás fő funkciói között az intelligens időbeosztás használati esetéből származó számos elem megvalósítása mellett, lehetőség van az előadónak, gyakorlatvezetőnek történő azonnali visszajelzésre, illetve helyzetérzékeny tartalom megjelenítésre. A cikket az alkalmazás továbbfejlesztési lehetőségeinek tárgyalásával zárjuk.

Kulcsszavak: Smart Campus, Android, intelligens időbeosztás

Abstract: In this paper several issues that have emerged during the design and implementation of an android application for a Smart Campus, within the frame of the Future Internet Research, Services and Technology (FIRST) project running at the University of Debrecen, are discussed. The Smart Campus concept, its basic architecture and the applied technologies are also introduced while the reasons of developing a native application are accounted. The developed application, besides implementing several items based on the use cases of intelligent schedule, provides functionalities to give rapid feedback to the lecturer or displaying location-aware content to users. The paper concludes with possible directions of further development.

Keywords: Smart Campus, Android, intelligent schedule

1. Bevezetés

A Debreceni Egyetemen futó Jövő Internete (FIRST) projekt céljai között egyebek mellett olyan információs és kommunikációs technológiák kifejlesztése, és azokra épülő alkalmazások megvalósítása szerepelt, amelyek így együttesen segítik a felhasználó munkáját avagy szórakozását, könnyítik életét. A projekt keretei között célunk és feladatunk olyan alkalmazások tervezése és fejlesztése volt, amelyek egy egyetemi campus intelligenssé válásának lépéseiként tekinthetők.

Ennek a folyamatnak a részeként elsőként számos használati eset kidolgozását végeztük, amelyek segítségével véleményünk szerint „felokosítható” egy campus. Ezekhez kapcsolódóan hoztunk később létre szolgáltatásokat (webszolgáltatások formájában), amelyekre (egyelőre) alapvetően kétféle felületet valósítottunk meg: egy webalkalmazást és egy natív androidos alkalmazást. Ebben a cikkben ez utóbbi áll a középpontban: ismertetjük, milyen kihívásokkal szembesültünk, és azokat milyen módon megoldva készült el végül az az alkalmazás, amelyet remélhetőleg szívesen használnak majd a campus androidos telefons és/vagy tablettel használó polgárai.

A cikk további részének felépítése a következő: a 2. fejezetben a Smart Campus projekt és azon belül az intelligens ügyintézés használatieset-együttes rövid bemutatása történik. A 3. fejezet a Smart Campus alkalmazások számára kialakított alaparchitektúrát ismerteti. A 4. fejezet tartalmazza a cikk legfontosabb részét, a kifejlesztett Android alkalmazás képernyőképekkel illusztrált, az elérhető funkciók szemszögéből történő bemutatását. A cikket az Összefoglalás zárja.

2. Smart Campus és intelligens ügyintézés

Az utóbbi néhány évben egyre inkább előtérbe kerültek az informatikában az intelligens („smart”) eszközök és szolgáltatások, amelyek a megszokottól valamilyen módon többet nyújtanak. Egy elektronikus eszközt akkor nevezünk intelligensnek, ha bizonyos mértékig interaktív és önálló módon képes működni. Gyakorta elvárjuk az ilyen eszközöktől, hogy rendelkezzenek olyan számítási kapacitással, amely lehetőséget biztosít mesterséges intelligencia alkalmazására. Egy közösséget akkor tekinthetünk ebben az értelemben intelligensnek, ha az információs és kommunikációs technológiák (information and communications technologies, ICT) segítségével intelligens megoldásokat keres a közösség egésze számára [4]. Szorosan ide tartozik a különféle infrastruktúrák és rendszerek integrált kezelésének és optimalizált vezérlésének a kérdésköre, hiszen ezek azok a területek, ahol talán a legnagyobb sikerrel kecsegtethet az ICT megoldások alkalmazása. A szolgáltatások tekintetében azokat tekinthetjük intelligensnek, amelyek online módon elérhetőek és a felhasználás során szerzett tapasztalatokat képesek saját működésükbe beépíteni.

Smart (intelligens) Campus alatt egy olyan, egy felsőoktatási intézményt felölelő közösséget értünk, amelyben a hallgatók és oktatók (a campus többi szereplőjével együtt) a számukra fontos információkhoz online hozzáférnek, továbbá ügyeiket elektronikus úton intézhetik. Az intelligens campus monitorozza az online tevékenységet, abból a célból, hogy felhasználja a felgyülemlett információt (alkalmazásnaplókat, érzékelőnaplókat stb.) saját működésének hatékonyabbá tételére.

Intelligens ügyintézés

Az „Intelligens ügyintézés” projekt a Smart Campus használati eseteinek egyikeként abból a célból jött létre, hogy egy prototípus segítségével bemutassa a Smart Campus szolgáltatásait, illetve az azok megvalósításához szükséges hátteret. A projekt megvalósítása két lépésben történt. Első lépésként egy egyszerű információszolgáltatás megvalósítása volt a cél, melynek keretében olyan hasznos szolgáltatások kialakítása állt a középpontban, amelyek használata közben mintegy melléktermékként információkat kapunk a felhasználoktól a második lépésben beindítandó intelligens szolgáltatásokhoz.

A kezdeti szolgáltatás kialakítása során relációs adatbázisba gyűjtöttük össze a szűkebb értelemben vett campuson – az DE Informatikai Karán – egy-egy adott félévben az oktatáshoz kapcsolódó eseményeket, melyek esetünkben legalább az alábbiakat jelentették:

- az Informatikai Kar honlapján közzétett adott félévre vonatkozó időbeosztás,
- az Informatikai Kar honlapján közzétett oktatói fogadóórák,
- egy-egy félév vizsgaidőszakra vonatkozó vizsgaidőpontok és -helyszínek,
- egy-egy félév szorgalmi időszakra vonatkozó órarendi információk a nappali és levelező tagozatos hallgatók számára.

Az összegyűjtött adatok elérésére a hálózati azonosítóval rendelkező dolgozók és hallgatók számára weben, illetve Android alkalmazás formájában biztosít lehetőséget.

Az adatok összegyűjtése az adatforrások heterogenitása miatt nem egyszerű feladat. Mivel e cikk fő célja nem ezen adatintegrációnak, hanem az összegyűjtött adatokra épített Android alkalmazásnak a bemutatása, ezért itt ezzel nem foglalkozunk. Az adatok gyűjtésének és integrálásának kérdései iránt érdeklődő olvasók [5]-ben az ilyen jellegű adatok integrációjának számos nehézségével és lehetséges megoldási módjával megismerkedhetnek.

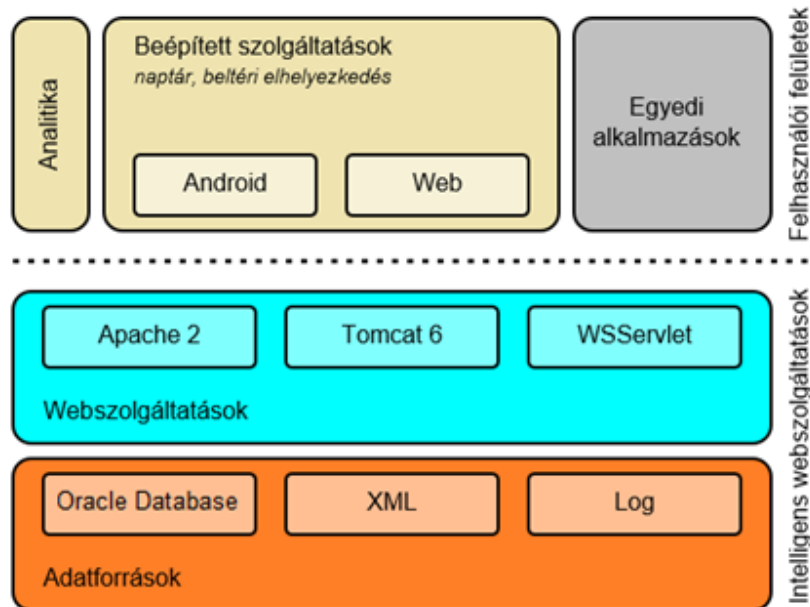
A szakterület feltérképezésekor az alábbi fogalmakból készítettünk fogalomszótárt:

- *Felhasználó* az Informatikai Kar valamely hálózati azonosítóval rendelkező dolgozója vagy hallgatója.
- *Eseménynek* tekintünk adott időpontban bekövetkező határidőt vagy meghatározott időponttól meghatározott időpontig tartó tevékenységet.
- *Esemény helyszíne* a campus valamely egyértelműen meghatározható területe, például egy épület valamely helyisége lehet. Nem minden esemény köthető helyszínhez (jellemzően a határidők például nem).
- Az esemény *szervezője* az Informatikai Kar valamely hálózati azonosítóval rendelkező dolgozója (egyben felhasználó), aki az eseményért felelős vagy az esemény időpontjában annak helyszínén az eseménnyel kapcsolatos ügyintézkést végez. Nem minden esemény esetében határozható meg az esemény szervezője, továbbá adott esemény több személyhez is tartozhat egyidejűleg.
- Az esemény *szerkesztője* az Informatikai Kar valamely hálózati azonosítóval rendelkező dolgozója (egyben felhasználó), akinek lehetősége van az esemény adatainak módosítására.
- *Kategória* alatt az események csoportosításának hierarchikus rendszerét értjük.
- *Feliratkozott* az olyan felhasználó, aki adott eseményt érdekesnek vagy érdeklődésére számot tartónak jelöl. Tetszőleges eseményre fel lehet iratkozni.
- *Felhasználói naptár* azon események összessége, amelyre az adott felhasználó feliratkozott.
- Az *üzenet* adott esemény változásával kapcsolatos megjegyzés vagy értesítés, melyet a változtatás időpontjában az eseményre feliratkozottak számára fogalmaz meg az esemény szerkesztője.
- *Felhasználói csoport* az Informatikai Kar valamely hálózati azonosítóval rendelkező dolgozóinak (egyben felhasználóknak) a halmaza.

3. Architektúra

A megvalósított szolgáltatások három jól elkülöníthető logikai rétegbe szervezhetők. Az alapokat az adatbázisok adják, ahol hagyományos, relációs adatbázist választottunk a naptár alkalmazás működésének támogatására (Oracle Database), statikus XML állományban tároljuk a beltéri pozíció információknak termekre, helyiségekre való leképezését, míg az alkalmazásnaplót egyszerű szöveges állományban gyűjtjük. A naplóállományok feldolgozására természetesen a későbbiekben szükség lesz. Az adatbázisok elérését webszolgáltatások segítségével biztosítjuk, ahol az alkalmazáslogika szintén ebben a rétegben kerül elsősorban implementálásra. Java alapú technológiát választottunk, Apache webszerver mögé helyezett Tomcat webalkalmazás-szerveren kaptak helyet a webszolgáltatások (1. ábra). Az alapfunkciók elérésére létrejött egy olyan webalkalmazás (portál), amelynek segítségével interneten keresztül, böngészőprogrammal használhatják az alkalmazásokat az első két felhasználói csoport tagjai (bejelentkezés nélkül és bejelentkezéssel), valamint elkészítettünk okostelefonra egy Android alkalmazást, amely ugyanezt biztosítja. Az előbbi azért fontos, mert nem rendelkezik mindenki okostelefonnal, az utóbbi viszont azért kiemelten fontos, mert

csak innen várhatunk beltéri pozíció adatokat. Az architektúra kidolgozásánál fontos szempont volt, hogy a nyilvánosan is elérhető webszolgáltatásokon keresztül teljes funkcionalitást nyújtsunk. Így bárkinek lehetősége van kiegészíteni az általunk kínált megoldásokat, saját felhasználói felülettel helyettesíteni azt, vagy egyes részeit felhasználni nagyobb alkalmazásban. A webszolgáltatások nyújtotta alapra építkezni lehet és kell.



1. ábra: Az alkalmazott 3 rétegű kliens-szerver architektúra

4. Az Android alkalmazás

Rendszerünkben az információhoz való hozzáférés mindenki számára bejelentkezés (regisztráció és felhasználói név) nélkül biztosított. A szolgáltatás két felületről is hozzáférhető, webböngészőből és okostelefonról, Android alkalmazás formájában. Az utóbbi eset a számunkra azért különösen érdekes, mert naplózni tudjuk a kar épületén belül működő eduroam hálózaton keresztül történő hozzáféréseket, amely a későbbi analitika bemenetül szolgáló beltéri pozíció adatok forrása. Természetesen felhasználhatók a házon belüli, asztali számítógépen, vezetékes hálózaton történő elérések naplói is. Ezek az adatokból – megfelelő analitikát feltételezve – következtethetünk például a szabadon hozzáférhető géptermekek kihasználtságára vagy felhasználhatjuk őket a hozzáférési pontok felszerelési helyének optimalizálásához.

Hallgatóink és oktatóink egyetemünkön hálózati azonosítóval rendelkeznek, amelyet számos informatikai rendszer eléréséhez használhatnak (WIFI hálózat, tanulmányi rendszer). A felhasználói azonosítóval rendelkezők lehetőséget kapnak arra, hogy a számukra fontos eseményeket megjelöljék, így saját naptárat alakítsanak ki. A bejelentkezésért cserébe figyelmeztetést kaphatnak a közelgő fontos eseményekről, illetve értesítést kaphatnak az ezekben bekövetkezett változásokról. Az így kapott adatokból következtethetünk az események várható látogatottságára, amely egyaránt hasznos információ lehet az esemény szervezőjének vagy például a hosszú sorbanállást elkerülni vágyó felhasználónak. Ez tipikus példája annak, hogy a közösség egyéni szereplőitől származó adatok hogyan hatnak vissza magára a közösségre, és hogyan képeznek újabb szolgáltatási réteget.

Szem előtt tartottuk az androidos világ igényeit a fejlesztés során, ezért az alkalmazás mindenhol forgatható és lefedtük szinte az összes előforduló képernyőméretet, így az applikáció végig megtartja szép küllemét. Odafigyeltünk továbbá, hogy a régebbi Android verziókkal is kompatibilis legyen az applikáció és végül egészen a 2.2.x platform verzióig, más néven a Froyo kiadásig sikerült megoldani a támogatást, ami megfelel a piac elvárásainak, továbbá angol és magyar nyelven is használható, tekintettel a külföldi hallgatókra. A nézetek görgethetőek, de igyekeztünk úgy kialakítani a felhasználói interfészt, hogy minden ráférjen egy képernyőre.

A megvalósított alkalmazás főbb funkciói az alábbiak:

1. Be- és kijelentkezés.
2. Események listázása különböző paraméterek szerint, értékelése, le- és feljelentkezés, jelentkezett felhasználók mutatása.
3. Feljelentkezett történések exportálása a beépített naptár alkalmazásba.
4. WIFI alapú helymeghatározás, amelynek segítségével környezetérzékeny módon megadhatjuk a felhasználóknak, hogy milyen események zajlanak éppen a közelükben.
5. Offline módban a lokális adatbázisból elérhetőek a feljelentkezett események.

Felhasznált eszközök

Ebben a szakaszban nagyon röviden bemutatjuk, hogy mely, az androidos alkalmazások fejlesztése során felmerülő főbb eszközöket alkalmaztuk a megvalósítás során.

Ilyenek például a *tevékenységek* (activityk) [3 (2.2.1., 29. o.)], amelyek a legtöbbször használt alkalmazáskomponensek. Saját, önálló felülettel rendelkeznek, de tipikus, hogy egy alkalmazáson belül több tevékenység is található, amelyek különböző, de egymástól jól elhatárolt célokat szolgálnak, és együtt adják az alkalmazás teljes funkcionalitását.

A rendszer egy alkalmazáson belül a tevékenységet egy úgynevezett *tevékenységveremben* (LIFO adatszerkezetben) [3 (2.5., 56. o.)] tárolja. Minden esetben az előtérben levő tevékenység van a verem tetején. Ha a felhasználó vagy az alkalmazás átvált egy másik tevékenységre, akkor eggyel lejjebb kerül a veremben, és a következő tevékenység lesz legfelül. Vissza gomb esetén a rendszer legfelülről veszi ki a megjelenítendő tevékenységet.

A *töredékek* (fragmentek) [3 (12.1., 332. o.)] a tevékenységhez vannak rendelve, és tevékenységenként önálló veremmel is rendelkeznek. Így oldható meg például, hogy a Vissza gomb hatására ne a tevékenységből lépünk vissza, hanem csak a tevékenységen belül az előző töredék kerüljön ismét előtérbe, ha már létezett. Ha már nincs több töredék, akkor a megszokott módon az előző tevékenység kerül előtérbe. Töredékek felhasználásával megoldható az is, ha külön szeretnénk támogatni a fekvő és az álló nézetet úgy, hogy a fekvő esetében a két töredék egymás mellett jelenjen meg, álló nézetben viszont csak az egyik.

A *műveletsáv* (action bar) [3 (12.2.1., 347. o.)] tulajdonképpen egy dedikált felület az alkalmazáson belül, amelyen az alkalmazás logója, a főbb parancsok és a globális navigációs eszközök láthatók. A műveletsáv célja az, hogy a gyakran használt funkciókat egyszerűen elérhetővé tegye a felhasználók számára, és ne kelljen őket eldugott menükben, illetve különféle nézeteken keresgélni.

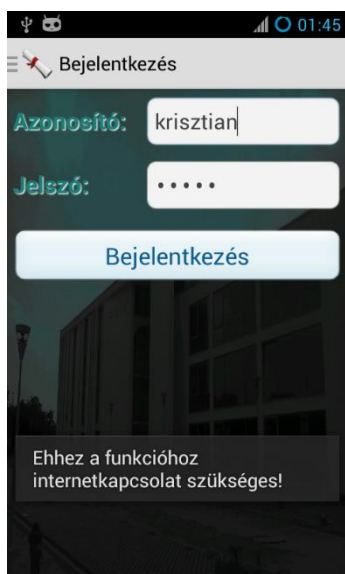
Az *adásvevő* komponensnek (broadcast receiver) [3 (2.2.4., 32. o.)] az a feladata, hogy a különféle események hatására aktiválódjon, és valamilyen feladatot végrehajtson. Az Android rendszer számos eseményt jelez, amelyekre feliratkozhatunk a saját alkalmazásból.

Felhasználóbarát kialakítás

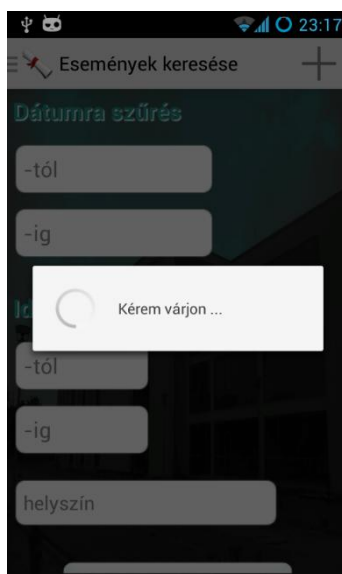
Az alkalmazás kialakítása során mindvégig fontosnak tartottuk, hogy kényelmesen használható legyen, hiszen ellenkező esetben nemigen várhatnánk el a felhasználóktól, hogy ne hagyják el az alkalmazást. Egyebek mellett Fontos, hogy a felhasználót sokféle hibaüzenet tájékoztassa arról, hogy milyen probléma történt (3. ábra), hogy pontosan megérthesse, hogy mi történt.

A 4. ábrán a várakozó kisablak látható, amíg a háttérszálon fut a kommunikáció a webszerverrel (ameddig a várakozás tart a készülék nem forgatható). Ez azért hasznos, mert így a hosszabb ideig zajló tevékenységek alatt a felhasználót nem hagyjuk kétségek között a tekintetben, hogy vajon történik-e valami az alkalmazásban.

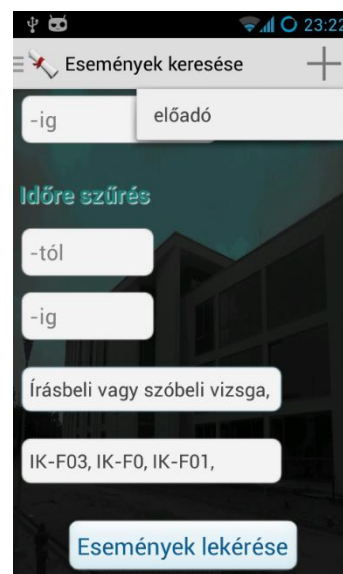
Az 5. ábrán a dinamikus, kényelmesen használható szűrőfelület látható, amely újabb és újabb szűrőfeltételek hozzáadásakor rendre bővül.



3. ábra: Informatív hibaüzenetek



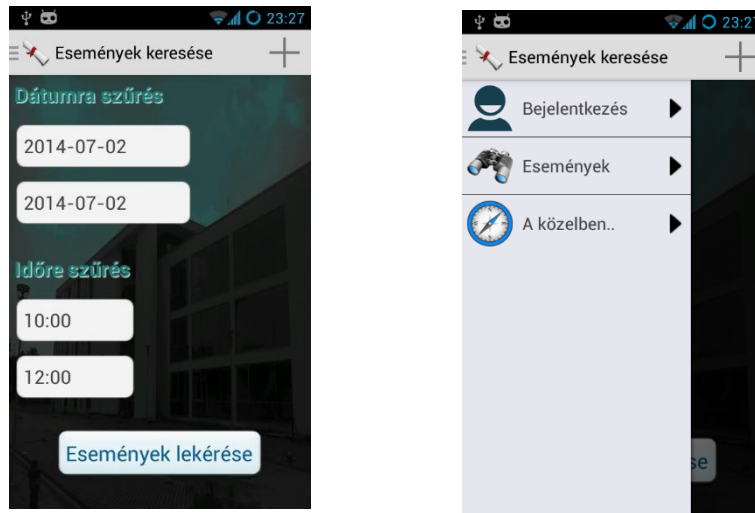
4. ábra: Várakozás



5. ábra: Dinamikus szűrőfelület



Navigáció

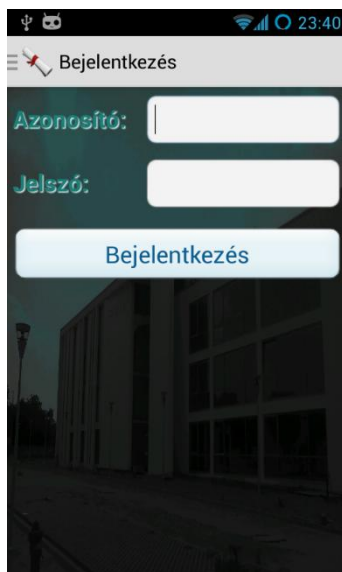
A navigáció az úgynevezett navigációs fiókkal (navigation drawer) történik (6. ábra), amelyet a Google három vagy annál több felső szintű nézetek közötti navigáláshoz javasol. Jelen esetben a nézetek a következők: bejelentkezés/feljelentkezések, események keresése, közelben zajló történések. Ezeket egy-egy töredék jeleníti meg, melyek egy fő tevékenységhez vannak kötve. A bal felső alkalmazásikonra kattintva, illetve bal oldalról középre tartó ujjmozdulattal jelenik meg, míg az ezzel ellentétes irányú mozdulat bezárja.



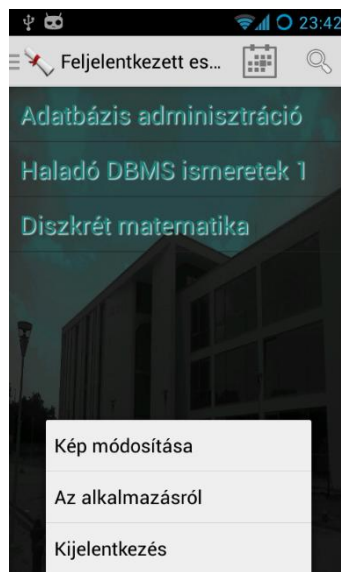
6. ábra. Navigációs fiók

Be- és kijelentkezés

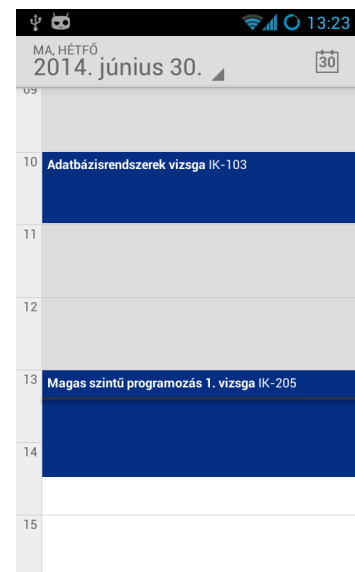
Bizonyos funkciók elérése csak autentikált felhasználók számára lehetséges, de az események szűrése és listázása ennek hiányában is használható. Bejelentkezés után (7. ábra) a menü címe átvált egy üdvözlő szövegre, ahol láthatjuk profilképünket is. A fiók legfelső listaelemének kiválasztása után most már nem a bejelentkező felületet, hanem azokat az eseményeket láthatjuk, amelyekre feljelentkeztünk. Az ehhez a nézethez (8. ábra) tartozó menüben lehetséges a kijelentkezés, a profilkép megváltoztatása, illetve egy rövid leírás is megtalálható az alkalmazásról. A műveletsávon található  gombbal a beépített naptár alkalmazással megtekinthetők azon események, amelyekre feljelentkeztünk (9. ábra), a  nagyítóval pedig cím szerinti szűrés végezhető. Utóbbi több nézetnél is előfordul az alkalmazásban, ugyanilyen funkcióval.



7. ábra: Bejelentkezés



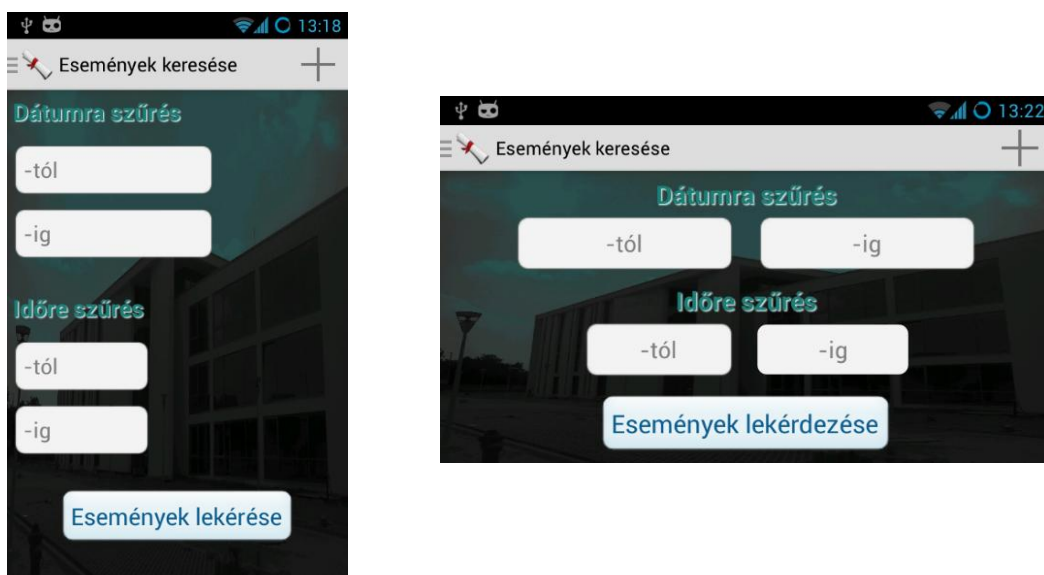
8. ábra: Feljelentkezett események



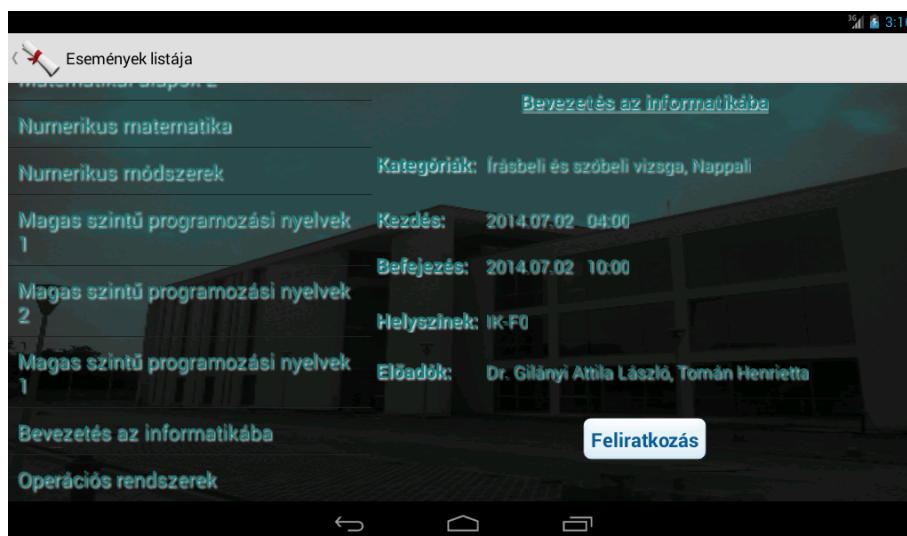
9. ábra: Naptár

Események keresése

A fő szűrési szempont a dátum és idő, de ezen túl a + gombra kattintva dinamikusan hozzáadódnak a nézethez további mezők melyekben beállíthatóak a helyszín, előadó, kategória kívánt értékei. A lehetséges értékek kiválasztását az alkalmazás automatikus szövegkiegészítéssel segíti. Fektetett kijelzőnél optimálisabb elrendezésű nézet jelenik meg (10. ábra). Az események lekérdezése gombra kattintva megjelenik a szűrésnek megfelelő események listája. Ennek a nézetnek érdekessége, hogy nagy képernyőméretnél fektetett állapotban nem egy, hanem két töredék töltődik be egymás mellé, így kényelmesebb kezelést tesz lehetővé (11. ábra). Egy címre kattintva megjelennek a részletek, ahol megtörténhet a feljelentkezés, illetve meg lehet tekinteni az esemény résztvevőit 👁 és értékelést ☆ is lehet adni rá. A gombok csak akkor láthatóak, ha lehetséges az értékelés, vagy ha vannak feliratkozott emberek, és szerkesztői vagyunk az adott eseménynek. A menüben a „Kimentés naptárba” opcióval exportálhatóak az esemény adatai a natív naptár alkalmazásba.



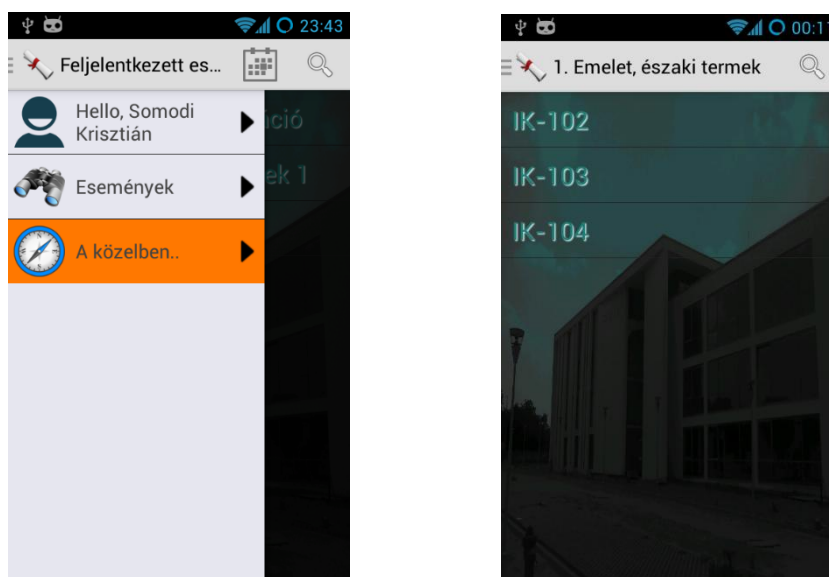
10. ábra: Kijelző orientáció változása esetén is marad a felhasználóbarát elrendezés



11. ábra: Nagy kijelzőnél két töredék látszik egyszerre

Közelben

A Közelben funkció segítségével azon tantermek kilistázását végezhetjük, amelyek a felhasználó közelében találhatóak. A webszolgáltatásnak elküldésre kerül egy olyan SOAP kérés, mely tartalmazza annak a WIFI hozzáférési pontnak az azonosítóját, amelyhez éppen kapcsolódik a készülék. Válaszban visszaküldi a közeli termeket, melyekből történik egy lekérés, hogy mik zajlanak az adott időpillanatban. Az adatok megjelenítése itt is úgy történik, hogy először a termék neveit látjuk, majd kiválasztás után megjelennek az épp akkor zajló esemény részletei (12. ábra). A helyiség, ahol tartózkodunk (pl: 1. Emelet, északi termék) lesz a nézet címe a műveletsávban. Az adásvevő komponens észleli az Android rendszer által szétküldött üzenetet, amikor az egyik hálózati csomóponttól a másikra átkapcsolódott a készülék. Ilyenkor frissül a lista és az aktuális információkat mutatja. Így sétálgatás közben mindig tudjuk, mi zajlik a közelben.



12. ábra: WIFI alapú helymeghatározás

5. Összefoglalás

Cikkünkben vázoltuk a „smart” fogalom egy lehetséges értelmezését, a Smart Campus projekt célját, fogalmait és architektúráját. Részletesen bemutattuk a hozzá készült Android alkalmazást, mellyel okoskészülékről kényelmesen elérhető a szolgáltatás mindenki számára.

Az alkalmazás továbbfejlesztési lehetőségei között elsőként arról kell szót ejtenünk, hogy az architektúrának köszönhetően az alkalmazás funkciói anélkül továbbfejleszthetők (például hatékonyabbá tehetők), hogy magához az Android alkalmazáshoz hozzá kellene nyúlni. Mivel a funkcionalitás a webszolgáltatásokba van bezárva, ezért egy hatékonyabb (például pontosabb beltéri navigációt megvalósító) szolgáltatásra cserélve a meglévőt, az androidos alkalmazás kódjának változtatása nélkül tehetjük azt jobbá.

A továbbfejlesztés másik iránya újabb szolgáltatások kifejlesztése, és azokhoz felhasználói felület megvalósítása. Az androidos alkalmazás szempontjából elsősorban (de nem kizárólagosan!) olyan jellegű fejlesztések megvalósítását tervezzük, amelyek a mobiltelefonok különféle szenzoradatainak felhasználásával képesek szolgáltatásokat megvalósítani, hiszen ez az, amelynek segítségével igazából kihasználhatóak a natív alkalmazások képességei. Ilyen alkalmazás lehet például többek között a GPS-adatok alapján

a fizikailag közel lévő barátokat illetve tennivalókat listázni, amellyel optimalizálható az időbeosztás.

6. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Kádek Tamás és Kósa Márk: *Intelligens campus Debrecenben*, 23. Számítástechnika és Oktatás Konferencia (SzámOkt 2013), Nagyszében, 210–213, 2013.
- [2] Adamkó Attila, Kádek Tamás, Kollár Lajos, Kósa Márk, Szathmáry László, Vágner Anikó: *Intelligens ügyintézés, relációs modell*.
- [3] Ekler Péter, Fehér Marcell, Forstner Bertalan, Kelényi Imre: *Android-alapú szoftverfejlesztés*, ISBN 978-963-9863-27-9, 2012.
- [4] Toshiba: *What is a Smart Community?* Online elérhetőség: <http://www.toshiba-smartcommunity.com/EN/smart-community> (Hozzáférés időpontja: 2014. július 4.).
- [5] Attila Adamkó and Lajos Kollár: *Extensible Data Management Architecture for Smart Campus Applications: A Crowdsourcing based Solution*, Proceedings of the 10th International Conference on Web Information Systems and Technologies (WEBIST 2014), Barcelona, 226–232, 2014.

Az SVM módozatainak alkalmazási lehetősége műszaki optimalizációs problémák megoldásában

Possibility of using SVM variations in solving of engineering optimization problems

Vámosi Attila^a, Kocsis Imre^b, Deák Krisztián^c

^aDebreceni Egyetem Műszaki Kar
vamosi.attila@eng.unideb.hu

^bDebreceni Egyetem Műszaki Kar
kocsisi@eng.unideb.hu

^cDebreceni Egyetem Műszaki Kar
deak.krisztian@eng.unideb.hu

Absztrakt: Az SVM (*Support Vector Machine*) módszert eredetileg osztályozásra fejlesztették ki, de egy ε érzéketlenségű veszteségfüggvény alkalmazásával regressziós feladatok megoldására is alkalmas. Használata széles körben elterjedt, osztályozásra vagy regresszióra visszavezethető műszaki problémák megoldására is használják. Az eredeti alapfeladatban az osztályozó optimális hipersík meghatározása kvadratikusan programozási feladat megoldásával történik. A módszer kifejlesztésekor a számítástechnika fejlettségét tekintve a feladat végrehajtása időigényes és nagy memóriakapacitás igényű volt, ezért különböző módozatokat fejlesztettek ki a kvadratikusan programozás hatékonyabb megoldására. Ilyenek voltak például az ún. szeletelési (*chunking*) eljárások és a dekompozíciós módszerek, különös tekintettel a legelterjedtebben használt SMO (*Sequential Minimal Optimization*) eljárás. A módozatok másik típusa magát a kvadratikusan programozást helyettesíti valamilyen más optimalizálási megoldással, például az SSVM (*Smooth Support Vector Machine*) egy egyszerű gradiens módszerekkel megoldható optimalizálási feladatot használ, vagy a legkisebb négyzetek elvén alapuló LS-SVM (*Least Squares Support Vector Machine*), melynek megoldása egy lineáris egyenletrendszer eredményeként áll elő. Ebben a munkában műszaki optimalizációs feladatokat oldunk meg ezen módozatok saját fejlesztésű implementációinak alkalmazásával, végül a kapott eredmények illetve tapasztalatok alapján összehasonlítjuk és vizsgáljuk a különböző feladatok különböző módozatokkal történő megoldásának lehetőségeit és korlátait.

Kulcsszavak: LS-SVM, optimalizálás, osztályozás, SMO, SVM

Abstract: The SVM (*Support Vector Machine*) method was originally developed for classification, but we can solve regression tasks by using an ε insensitive loss function. It has been widely used to solve engineering problem, which can be traced back to classification or regression. In the original base classifier we have to solve a quadratic programming problem to determine the optimal hyperplane. The implementation of this procedure was time-consuming, therefore various ways have been developed to effectively solve the quadratic programming. These included the so-called chunking procedures and decomposition methods, in particular, the most widely used SMO (*Sequential Minimal Optimization*) method. Another type of methods replace the quadratic programming some other optimization solution, e.g. SSVM (*Smooth Support Vector Machine*) uses a simple possible gradient methods to solve the optimization task, LS-SVM (*Least Squares Support Vector Machine*), which is based on the principle of least squares, where the solution is a result of a system of linear equations. In this work, we solve different engineering problems with use own developed implementations of the different methods. Finally, we compare and examine the possibilities and limitations of these methods.

Keywords: classification, LS-SVM, optimization, SMO, SVM

1. Bevezetés

A gépi tanulás módszereinek alkalmazása széles körben elterjedt a különböző tudományterületeken, így a műszaki tudományok terén, műszaki problémák megoldásában is egyre sűrűbben alkalmazzák. Azok a feladatok, melyek visszavezethetők osztályozási alapfeladatra, megoldhatók a gépi tanulás egyik speciális módszerével az SVM (*Support Vector Machine*) módszerrel [1]. Ennek a módszernek az alkalmazási lehetőségét tovább szélesíti az, hogy a függvényközelítés (regresszió) egy ε érzéketlenségű veszteségfüggvény alkalmazásával felírható egy olyan osztályozási feladatként, ahol a helyes osztályozás hibája nulla, míg a margón belülre kerülő, illetve a rosszul osztályozott pontokat gyengítő változók (ξ , *slack variables*) bevezetésével büntetjük [2].

Az SVM alapelve, hogy a különböző osztályokba tartozó pontokat egy hipersíkkal választja el egymástól és a pontokat aszerint osztályozza, hogy azok a hipersík melyik oldalára esnek. Egy ponthalmazt többféle síkkal is szeparálhatunk. Az SVM módszer megoldásakor azt az optimális hipersíkot keressük, melynek a legnagyobb a margója, mert a széles margóval rendelkező döntési rendszerek jobban általánosítanak, ezáltal pontosabban osztályozzák azokat a pontokat, melyekkel a tanulási folyamat során még nem találkoztak. A margó a hipersík és a hipersíkhöz legközelebb eső pont közötti távolság. Ezt kell maximálnunk, így a hipersíkkeresési feladat egy konvex optimalizálási probléma, melyet kvadratikus programozással oldhatunk meg.

2. Az SVM módozatai

Az elmélet megalkotásakor a számítógépek kapacitása sebesség és memóriakapacitás tekintetében sem voltak olyan fejlettek, mint napjainkban, így egy ilyen feladat megoldása nagyszámú tanulólalmaz esetén rendkívül sok időt és memóriát igényelt. Ennek áthidalására a kvadratikus programozás egyszerűsítésére illetve megkerülésére különböző módszereket fejlesztettek ki. A kvadratikus programozás memóriai igényének csökkentését segíti az úgynevezett szeletelési (*chunking*) eljárás [3], mely az eredeti feladatot kisebb méretű részfeladatok megoldásával helyettesíti. Ennek továbbfejlesztése *Osuna algoritmus* [4], mely azonos méretű részfeladatokra bontja az eredeti problémát. A legelterjedtebb módszer, mely lényegében már kiváltja a kvadratikus programozást az SMO (*Sequential Minimal Optimization*) eljárás [5], mely minden lépésben pontosan két változó analitikus módszerekkel történő optimalizálását hajtja végre.

A módozatok másik irányvonala kvadratikus programozás helyettesítése más optimalizálási módszer alkalmazásával. Az SSVM (*Smooth Support Vector Machine*) [6], az eredeti feladatot egy olyan optimalizálási feladatként írja fel, mely megoldható egy egyszerű gradiens módszerrel. A legelterjedtebb ilyen módosítás a négyzetes hibára optimalizált LS-SVM (*Least Squares Support Vector Machine*) [7], mely a feltételeket egyenlőségek formájában írja fel, így az eredmény egy lineáris egyenletrendszer megoldásával állítható elő.

2.1. Az eredeti SVM osztályozó

Lineáris osztályozás esetén a keresett hipersík egy lineáris függvény, melynek egyenlete:

$$\mathbf{w}^T \mathbf{x} + b = 0,$$

ahol:

- \mathbf{w}^T – a súlyvektor transzponáltja;
 \mathbf{x} – egy jellemzővektor;
 b – az eltolás érték.

A margó nagysága az alábbi módon számítható:

$$M = (\mathbf{x}_{+1} - \mathbf{x}_{-1}) \cdot \mathbf{n} = (\mathbf{x}_{+1} - \mathbf{x}_{-1}) \cdot \frac{\mathbf{w}}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|}$$

Ez a margó akkor lesz maximális, ha a súlyvektor normája minimális, tehát az optimális súlyvektor meghatározásához az alábbi feladatot kell megoldanunk:

$$\min \frac{1}{2} \mathbf{w}^T \mathbf{w}$$

az alábbi feltétel teljesülése mellett:

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \quad (i = 1, \dots, n),$$

ahol:

- y_i – a tanulóponatok osztálycímkéje;
 n – a tanulóponatok száma.

Ennek a feltételes szélsőérték-keresési feladatnak a megoldását Lagrange egyenlet segítségével adhatjuk meg. A megfelelő deriváltak eredményét behelyettesítve a Lagrange egyenletbe kapjuk a duális alakot:

$$\max \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

az alábbi feltételek mellett:

$$\sum_{i=1}^n \alpha_i y_i = 0 \text{ és } \alpha_i \geq 0 \quad (i = 1, \dots, n),$$

ahol α_i -k a Lagrange multiplikátorok. A feladatot kvadratikus programozással megoldva kapjuk meg az optimális Lagrange multiplikátorokat, melyek nagy része általában nullával egyenlő. Emiatt a megoldás kialakításában a tanulóponatoknak csak egy része fog részt venni, azok a pontok, melyekhez nem nulla értékű Lagrange együtthatók tartoznak. Ezeket a pontokat nevezzük tartóvektoroknak (*support vectors*). A tartóvektorok felhasználásával az optimális súlyvektor a

$$\mathbf{w}^* = \sum_{i \in SV} \alpha_i y_i \mathbf{x}_i,$$

ahol SV a tartóvektorok indexeit tartalmazó halmaz; az optimális eltolás érték pedig bármelyik tartóvektor felhasználásával a

$$b^* = y - \mathbf{w}^{*T} \mathbf{x}$$

összefüggésekkel számítható. Az optimális eredményeket felhasználva a kétosztályos lineáris osztályozó gép válasza egy tetszőleges \mathbf{z} pontra az alábbi alakban írható fel:

$$y(\mathbf{z}) = \text{sgn} \left(\sum_{i \in SV} \mathbf{w}^{*T} \mathbf{z} + b^* \right)$$

Ha a tanulóponatok lineárisan nem szeparálhatók hiba nélkül, akkor az osztályozást úgy lehet végrehajtani, hogy az osztályozás margóját ún. lágy margóként kezeljük, azaz megengedjük, hogy tanulóponatok a margón belülre, illetve rossz oldalra is kerülhessenek. Ebben az esetben ξ_i gyengítő változók (*slack variables*) bevezetésével az

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i \text{ és } \xi_i \geq 0 \quad (i = 1, \dots, n)$$

feltételek teljesülését követeljük meg. Az optimális hipersík meghatározását ekkor úgy kell elvégezni, hogy a hibásan osztályozott pontok száma minimális legyen, ezért a primál alak

$$\min \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n \xi_i$$

alakra változik, ahol C a két tag közötti kompromisszumot beállító tényező (büntető paraméter). A duál alak ebben az esetben is megegyezik a korábban megadottal, csupán az együttthatókra vonatkozó feltétel második része kiegészül egy C felső korláttal. A lágy margó az osztályozó választát sem befolyásolja, az eredmény megegyezik a lineáris osztályozó válaszával.

A megoldandó példáinkban a feladat a bemeneti térben még lágy margó alkalmazásával sem oldható meg lineárisan. A lehetséges megoldáshoz keresnünk kell egy olyan többnyire magasabb dimenziójú teret, melyben a pontok lineárisan szeparálhatóak. Ehhez egy $\mathbf{x} \rightarrow \varphi(\mathbf{x})$ nem lineáris transzformációval a pontokat ebbe a térbe (tulajdonságok tere) kell transzformálni, és a szeparáló sík keresését ebben a térben kell elvégezni. Így az osztályozó válasza a tulajdonságok terében az alábbi alakban írható fel:

$$y(\mathbf{z}) = \text{sgn} \left(\sum_{i \in SV} \alpha_i y_i \varphi(\mathbf{x}_i)^T \varphi(\mathbf{z}) + b^* \right)$$

A megfelelő transzformációs függvény meghatározása azonban nehéz feladat, sőt bizonyos esetekben szinte lehetetlen. Itt alkalmazható az ún. kernel trükk (*kernel trick*) [8], azaz ezt a belső szorzatot helyettesíthetjük egy olyan megfelelően választott magfüggvénnyel (kernel függvénnyel), mely a kernel térben generálja ezt a belső szorzatot. Így az osztályozó kerneltérbeli válasza:

$$y(\mathbf{z}) = \text{sgn} \left(\sum_{i \in SV} \alpha_i y_i K(\mathbf{x}_i, \mathbf{z}) + b^* \right)$$

Egy függvényt akkor tekintünk kernel függvénynek, ha kielégíti a Mercer tétel feltételeit. Kernel függvényekből számtalan létezik, sőt különböző műveletekkel az alapfüggvényekből további érvényes kernel függvények származtathatók. Példáinkban a leggyakrabban használt Gauss kernelt a radiális bázisfüggvényt (RBF, *Radial Basis Function*) fogjuk használni, melynek alakja:

$$K(\mathbf{x}_i, \mathbf{x}) = \exp \left(-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2 \right)$$

2.2. Az SMO módszer

Az SMO módszer minden lépésben két heurisztikusan kiválasztott változót (α_i és α_j) optimalizál analitikus módszerrel [5]. A változóra vonatkozó korlátozó feltételek alapján megadható az a tartomány, melyben az optimum keresendő. Ennek alsó (L) és felső (H) értéke ha $y_i \neq y_j$, akkor

$$L = \max(0, \alpha_j - \alpha_i) \text{ és } H = \min(C, C + \alpha_j - \alpha_i)$$

ha $y_i = y_j$, akkor

$$L = \max(0, \alpha_i + \alpha_j - C) \text{ és } H = \min(C, \alpha_i + \alpha_j)$$

Ezek alapján meghatározható a második változó új értéke az alábbi képlettel:

$$\alpha_j = \alpha_j^{old} - \frac{y_i (E_i - E_j)}{\eta},$$

ahol

$$E_k = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_k) + b - y_k \text{ és } \eta = 2K(\mathbf{x}_i, \mathbf{x}_j) - K(\mathbf{x}_i, \mathbf{x}_i) - K(\mathbf{x}_j, \mathbf{x}_j)$$

A korlátozó feltételek figyelembe vételével a tényleges érték:

$$\alpha_j^{new} = \begin{cases} H & \text{ha } \alpha_j \geq H \\ \alpha_j & \text{ha } L < \alpha_j < H \\ L & \text{ha } \alpha_j \leq L \end{cases}$$

Az első változó új értéke pedig:

$$\alpha_i^{new} = \alpha_i^{old} + y_i y_j (\alpha_j^{old} - \alpha_j^{new})$$

A két változóhoz tartozó eltolás érték (b) pedig

$$b^{new} = \begin{cases} b_1 & \text{ha } 0 < \alpha_i < C \\ b_2 & \text{ha } 0 < \alpha_j < C \\ (b_1 + b_2) / 2 & \text{egyébként} \end{cases}$$

ahol

$$b_1 = E_i + y_i (\alpha_i^{new} - \alpha_i^{old}) K(\mathbf{x}_i, \mathbf{x}_i) + y_j (\alpha_j^{new} - \alpha_j^{old}) K(\mathbf{x}_i, \mathbf{x}_j) + b^{old}$$

$$b_2 = E_j + y_i (\alpha_i^{new} - \alpha_i^{old}) K(\mathbf{x}_i, \mathbf{x}_j) + y_j (\alpha_j^{new} - \alpha_j^{old}) K(\mathbf{x}_j, \mathbf{x}_j) + b^{old}$$

A fenti eljárást a módszer addig ismétli, míg az így előállt halmaz által szolgáltatott eredmény és az elvárt eredmény közötti különbség egy megadott küszöbérték alá csökken vagy a különbségváltozás már nem számottevő. Ebből következik, hogy a módszer nem feltétlenül a pontos értékeket határozza meg, hanem csak egy közelítést, mely az optimalizációs feladatok esetén olyan minimális hibát eredményez, ami már elfogadható.

2.3. Az LS-SVM osztályozó módszer

Az LS-SVM optimalizálandó feladata a lágy margójú osztályozás alakjához hasonlít azzal a különbséggel, hogy az ε veszteségfüggvényt egy négyzetes veszteségfüggvényre cseréli [7]. A primál feladat alakja tehát:

$$\min \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \frac{1}{2} \sum_{i=1}^n e_i^2$$

az

$$y_i (\mathbf{w}^T \varphi(\mathbf{x}_i) + b) = 1 - e_i \quad (i = 1, \dots, n)$$

feltétel mellett. A megfelelő deriváltak Lagrange egyenletbe történő behelyettesítése után kapjuk meg a kvadratikus programozást kiváltó lineáris egyenletrendszert az alábbi alakban:

$$\begin{bmatrix} \mathbf{I} & 0 & 0 & -\mathbf{Z}^T \\ 0 & 0 & 0 & -\mathbf{Y}^T \\ 0 & 0 & C\mathbf{I} & -\mathbf{I} \\ \mathbf{Z} & \mathbf{Y} & \mathbf{I} & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ b \\ \mathbf{e} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \mathbf{1} \end{bmatrix},$$

melyből \mathbf{w} és \mathbf{e} kiküszöbölésével az alábbi egyszerűbb alak jön létre:

$$\begin{bmatrix} 0 & \mathbf{Y}^T \\ \mathbf{Y} & \mathbf{\Omega} + C^{-1}\mathbf{I} \end{bmatrix} \cdot \begin{bmatrix} b \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix},$$

ahol

$$\Omega_{ij} = y_i y_j \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j) = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$\mathbf{Y} = [y_1, \dots, y_n]$$

$$\mathbf{1} = [1, \dots, 1]$$

$$\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_n]$$

$\mathbf{I} = n$ elemű egységmátrix

Az LS-SVM osztályozó válasza megegyezik a kvadratikus osztályozó válaszával. Fontos különbség viszont, hogy az eredeti SVM ún. ritkasági tulajdonságának köszönhetően a Lagrange multiplikátorok nagy része nulla, így a válasz megadásában csak a tanulóponatok egy része vesz részt, addig az LS-SVM esetén az egyenlőség miatt nincsenek nullaértékű együtthatók, így a válasz megadásában az összes tanulóponat részt vesz. Nagyszámú tanulóponat halmaz esetén ez a válaszadás idejét nagymértékben megnövelheti.

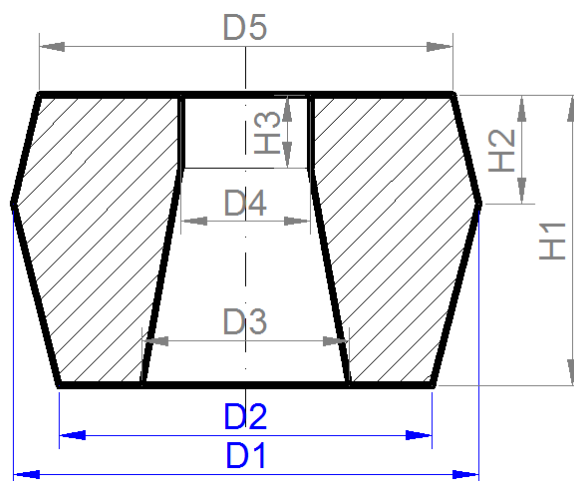
3. Osztályozásra visszavezetett műszaki optimalizációs feladatok

A különböző módozatok működésének és eredményének összehasonlításához osztályozásra visszavezetett alakoptimalizációs feladatokat fogunk megoldani. Adott egy gép, melybe egy gumialkatrészt kell beépíteni. A beépítendő gumialkatrésznek egy elvárt alakú rugókarakterisztikát kell teljesítenie. A rugókarakterisztika függ a gumialkatrész geometriai alakjától és méreteitől. A különböző méretű gumialkatrészek rugókarakterisztikája eltérő. A geometriai méretek egy adott tartományból, meghatározott léptékekkel választhatók. Keressük azt a halmazt, melyből a kiválasztott paraméterekkel rendelkező gumialkatrészek rugókarakterisztikája egy adott tőrészel közelíti az elvárt karakterisztikát.

Az elvárt és a tényleges rugókarakterisztika közötti különbség lényegében a nyomóerő által az alkatrészen végzett munka különbsége. Ez a különbség számítható végeelem módszerrel [9], de az összes variációra kiszámítani rendkívül időigényes feladat lenne. SVM módszer alkalmazásával néhány tanulóponat alapján gyorsan, jó közelítéssel elvégezhető az osztályozás.

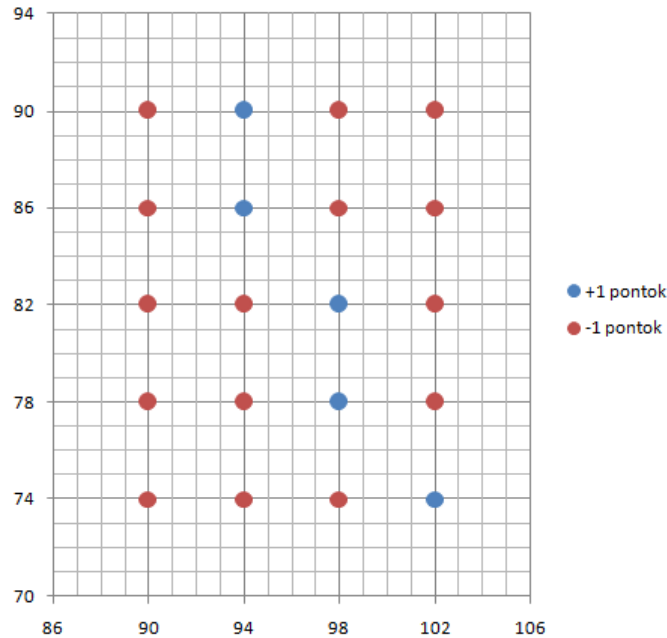
3.1. Két bemeneti paraméteres alakoptimalizálás

Első példánkban a gumialkatrész alakjának két paramétere ($D1$ és $D2$) módosítható (lásd 1. ábra). [10]



1. ábra. A gumialkatrész geometriája

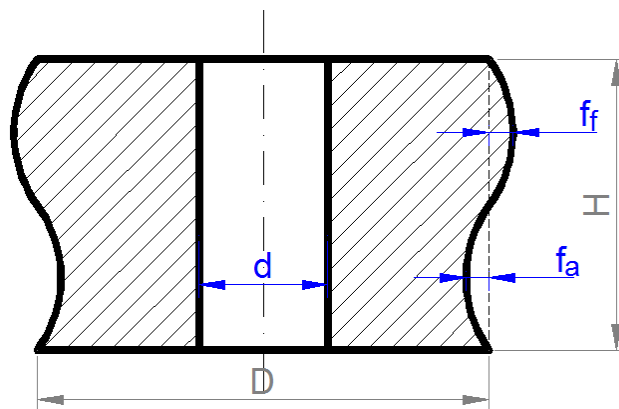
$D1 \in \{90, \dots, 102\}$ és $D2 \in \{74, \dots, 90\}$ mm közötti tartományból 1 mm pontossággal választható, ez összesen 221 darab különböző gumialkatrészt jelent. Ebből kiválasztunk 20 darab tanulópontot, kiszámítjuk végeelem módszerrel a munkakülönbségeket és azok alapján osztályozzuk őket úgy, hogy +1 címkét kapnak a megfelelő és -1 címkét a nem megfelelő alkatrészek (lásd 2. ábra).



2. ábra. A tanulópontok

3.2. Három bemeneti paraméteres alakoptimalizálás

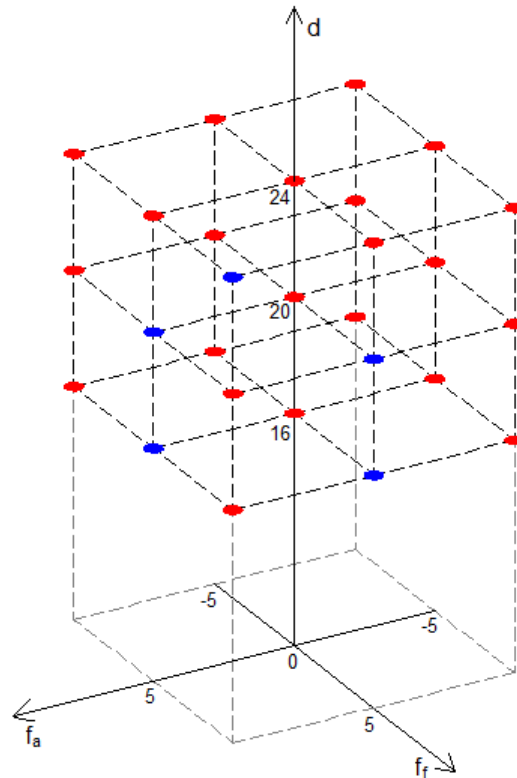
Második példánkban a gumialkatrész alakjának három paramétere (d , f_a és f_f) módosítható (lásd 3. ábra) [11].



3. ábra. A gumialkatrész geometriája

$d \in \{16, 18, 20, 22, 24\}$, f_a és $f_f \in \{-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5\}$ mm közötti tartományból választható, ez összesen 605 darab különböző gumialkatrészt jelent. Ebből kiválasztunk 27 darab tanulópontot, kiszámítjuk végeelem módszerrel a munkakülönbségeket és az előző

példához hasonlóan osztályozzuk őket úgy, hogy +1 címkét kapnak a megfelelő és -1 címkét a nem megfelelő alkatrészek (lásd 4. ábra).



4. ábra. A tanulópontok

4. A feladatok megoldása a különböző módszerek alkalmazásával

A feladatokat mindhárom módszerrel megoldottuk. A módszereket az azonos feltételek biztosítása érdekében a National Instruments által fejlesztett LabVIEW környezetben implementáltuk. Az eredeti SVM módszer kvadratikus programozási algoritmusát criss-cross módszerrel, az LS-SVM megoldását Gauss eliminációval, az SMO módszert pedig a Platt által megadott pszeudokód [5] alapján programoztuk le. A megoldások ugyanazon a gépen lettek futtatva, melynek főbb adatai:

CPU: 2,20 GHz Intel Pentium B960 Dual Core,

RAM: 4,0 GB DDR3,

Op.rendszer: Microsoft Windows7 64bit

Környezet: National Instruments LabVIEW 2013

Mindhárom módszer esetében a tanulópontok és tesztpontok felhasználásával *grid-search* és *cross-validation* [8] eljárások alkalmazásával határoztuk meg az optimális hiperparamétereket, ezek közül a legkisebb hibát eredményező megoldást közöljük.

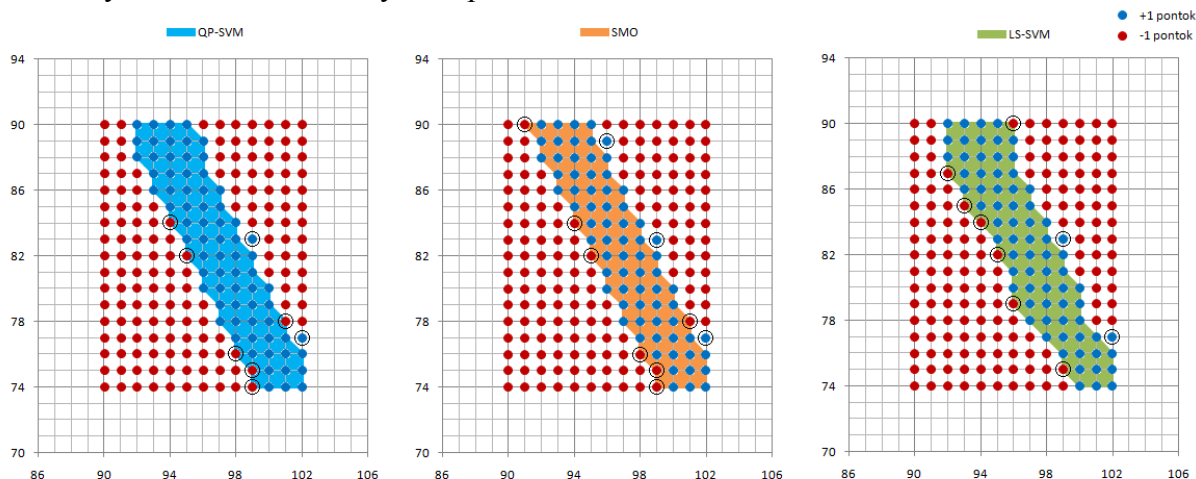
4.1. A két bemeneti paraméteres feladat megoldásai

A feladatot mindhárom módszerrel sikerült megoldani. Az optimális hiperparaméterek ismeretében minden egyes módszer esetén osztályoztuk az összes alkatrészt és a kapott eredményeket összevetettük az alkatrészek tényleges osztálycímkéivel. A módszerek összehasonlításának eredményét az 1. táblázatban foglaltuk össze.

	QP-SVM	SMO	LS-SVM
Hyperparaméterek	$C=100; \gamma=0,01$	$C=1000; \gamma=0,1$	$C=100; \gamma=0,01$
Tartóvektorok száma	10 db	10 db	20 db
Hibás osztályozás	8 hiba	10 hiba	9 hiba

1.táblázat A kétparaméteres feladat különböző módszerekkel történő megoldásainak összehasonlítása

A hibás osztályozás mindhárom módszer esetén 5 % alatti értékre csökkent. Hasonlóság a módszerek között, hogy szélesebb sávot adtak vissza, azaz a hibák többnyire abból adódtak, hogy rossz alkatrészeket soroltak be a jó kategóriába. Legkevesebb hibával a kvadratikus programozás oldotta meg a feladatot, legtöbb hibát az SMO módszer adta. A három módszer eredménye és a hibásan osztályozott pontok az 5. ábrán láthatók.



5. ábra. A teljes halmaz osztályozási eredményei

Az optimális hiperparaméterek között is látható némi különbség, valamint a tartóvektorok száma is eltérő. Az LS-SVM már említett egyenlőségi feltétele miatt az összes tanulópont tartóvektor, míg a QP és az SMO azonosan 10-10 tartóvektort határozott meg, melyek indexei is teljesen egyezők.

4.2. A három bemeneti paraméteres feladat megoldásai

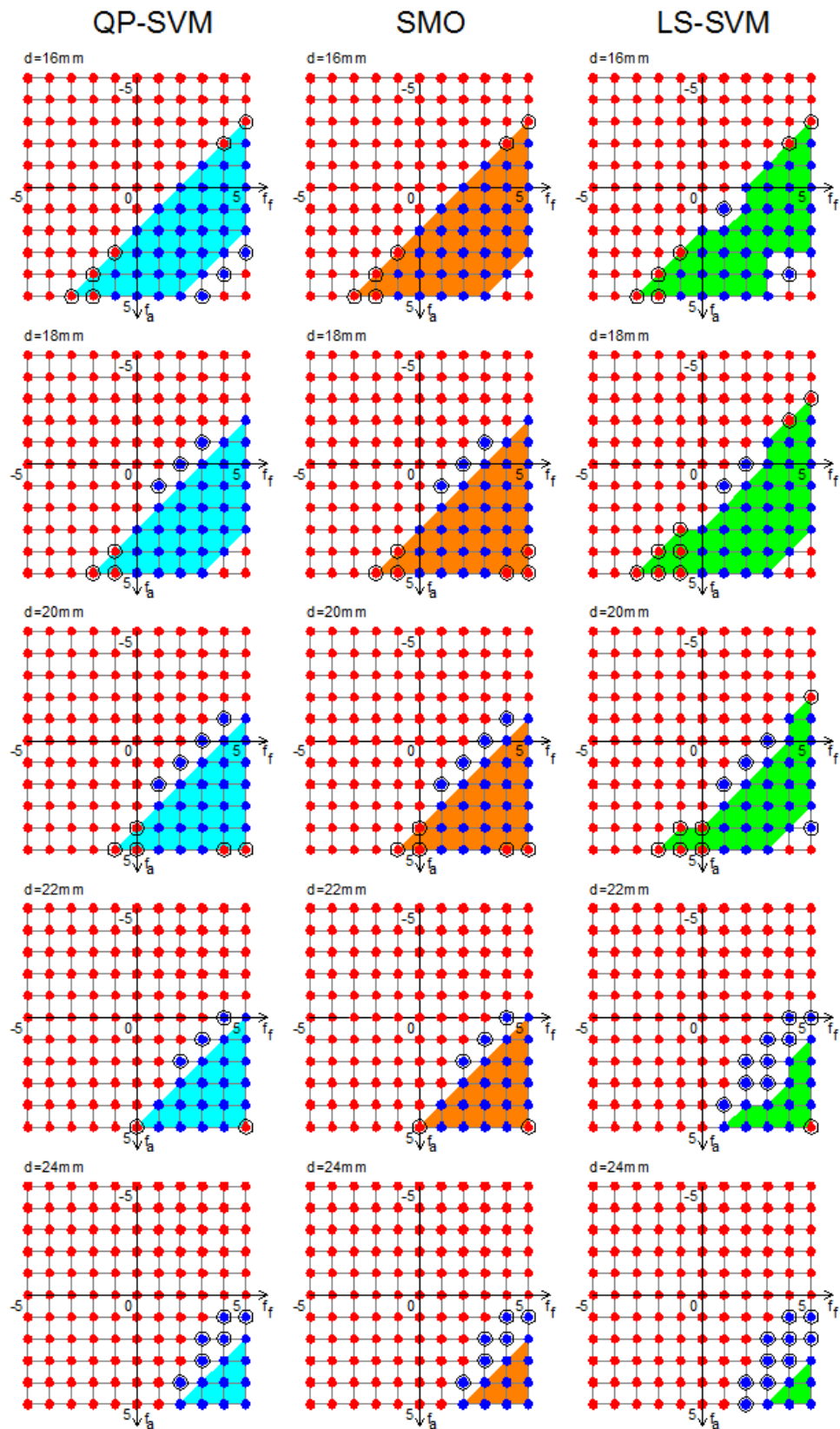
Ezt a feladatot is sikerült mindhárom módszerrel megoldani. Az előbbi feladathoz hasonlóan a módszerek összehasonlításának eredményét a 2. táblázatban foglaltuk össze.

	QP-SVM	SMO	LS-SVM
Hyperparaméterek	$C=10000; \gamma=0,001$	$C=10000; \gamma=0,01$	$C=100; \gamma=0,01$
Tartóvektorok száma	11 db	12 db	27 db
Hibás osztályozás	35 hiba	35 hiba	48 hiba

2.táblázat A háromparaméteres feladat különböző módszerekkel történő megoldásainak összehasonlítása

A hibás osztályozás mindhárom módszer esetén 10 % alatti értékre csökkent. Legtöbb hibát ebben a feladatban az LS-SVM vétett, míg a QP és az SMO ennél lényegesen kevesebb

hibával oldotta meg a feladatot. A három módszer eredménye és a hibásan osztályozott pontok a 6. ábrán láthatók.



6. ábra. A teljes halmaz osztályozási eredményei

Az optimális hiperparaméterek között most is látható némi különbség, és a tartóvektorok száma is eltérő. Az LS-SVM esetében természetesen az összes tanulópontra tartóvektor, míg a QP és az SMO azonosan 11-12 tartóvektort határozott meg, melyek indexei is azonosak, a 9. tanulópontra kivételével, melyet a QP nem használt fel.

A feladatok megoldása során a futási időt mértük, de ilyen kis számú tanulópontra esetében a különbség olyan minimális, hogy ebből következtetést levonni nem lehetséges. Futási időt a tanulópontraok száma egyedül az LS-SVM esetén határozza meg egyértelműen, mert az elimináció annál tovább tart, minél több tanulópontraal számolunk. A QP és az SMO heurisztikus kiválasztása miatt nincs egyértelmű összefüggés a tanulópontraok száma és a futási idő között. Nagyobb tanulópontraaszám esetén is tarthat rövid ideig a futtatás, míg kevesebb tanulópontra is okozhat hosszabb futási időt. Az idő tekintetében azt lehet kijelenteni, hogy az osztályozás sebessége nagy számú tanulóhalmaz esetén az LS-SVM módszerrel biztosan hosszabb, hiszen több tartóvektorral kell számolnia, mint a ritkább eredményt adó QP vagy SMO eljárásnak.

5. Összefoglalás

A bemutatásra került három módszer mindegyike alkalmas osztályozásra visszavezethető műszaki problémák megoldására. Az optimális hiperparaméterek gondos meghatározásával a megoldás pontossága növelhető. Kis számú tanulópontra esetében a pontosság gyengébb, de a futási idő és a számítógép kapacitásigénye lényegesen kisebb. Nagy számú tanulópontra halmaz esetén nagy pontosság is elérhető, de ez a futási idő és a kapacitásigény növekedésével jár. Valós idejű osztályozási feladatra az LS-SVM csak kis számú tanulópontra esetében alkalmas, mert nagy számú tanulópontra esetén mivel minden tanulópontra tartóvektor lesz, így a számításiidő megnőhet és előfordulhat, hogy a következő mintavétel idejére még nem készül el az előző minta osztályozása.

Mivel a három módszer eredménye, pontossága, hibája közel azonos lett, így a fenti észrevételek figyelembe vételével műszaki osztályozási problémák megoldására bármelyik alkalmazását javasoljuk.

Irodalomjegyzék

- [1] Corina Cortes és Vladimir Vapnik: Support-Vector Networks, *Machine Learning* **20** (1995), 273-297.
- [2] Vladimir Vapnik et al.: Support Vector Regression Machines, *Neural Information Processing Systems* **9** (1997), 155–161.
- [3] B. E. Boser et al.: A training algorithm for optimal margin classifiers, *Fifth Annual Workshop on Computational Learning Theory* (1992)
- [4] E. Osuna és F. Girosi: Reducing run-time complexity in SVMs, *Advances in Kernel Methods – Support Vector Learning* (1999), 271–284.

- [5] John C. Platt: Fast Training of Support Vector Machines using Sequential Minimal Optimization, *Advances in Kernel Methods – Support Vector Learning* (1999), 185–208.
- [6] Yuh-Jye Lee és O.L. Mangasarian: SSVN: A Smooth Support Vector Machine for Classification, *Computational Optimization and Applications* **20** (2001), 5–22.
- [7] Johan Suykens et al.: Least Squares Support Vector Machines, *World Scientific* (2002)
- [8] Simon Haykin: Neural Networks and Learning Machines, *Pearson Edition Ltd.* (2009), 268–296.
- [9] Mankovits Tamás és Szabó Tamás: Finite Element Analysis of Rubber Bumper Used in Air-springs, *Procedia Engineering* **48** (2012), 388–395.
- [10] Mankovits Tamás et al.: Optimization of the Shape of Axi-Symmetric Rubber Bumpers, *Strojniski vestnik – Journal of Mechanical Engineering* **60** (2014), 61–71.
- [11] Mankovits Tamás et al.: Shape Design of Rubber Part Using FEM, *International Review of Applied Sciences and Engineering* **4** (2012), 85–94.

Preferált algoritmus leíró eszközök

Preferred algorithm descriptor devices

Göncziné Kapros Katalin

Eszterházy Károly Főiskola

kaprosk@ektf.hu

Absztrakt: Az algoritmusok jelentősége tanulmányainkban teljesebbé válik. A tanulás, valamint a tanítás folyamatának megvan a sajátos lépéssorozata, melyet lépésről lépésre követve eredményes tanuláshoz, tanításhoz jutunk. A számítógépes programok készítésének folyamatában is erre támaszkodunk, a megfelelő algoritmus előállítását tekinthetjük elsődlegesnek, és így a tanításban is ez kapja a legnagyobb hangsúlyt. Mindezek mellett számos tudományterület épül rá, és különösen jelentős az algoritmusok szerepe a matematika, az informatika és a könyvtári informatika oktatásában (pl. programozás, webtechnológia, script nyelvek stb.). Kutatásom fókuszában a könyvtári informatika fejlesztési terület áll.

Előadásomban az algoritmusok oktatásának alakulását mutatom be, az ebben a témában készített tanulmányom alapján. Megvizsgáltam az elmúlt 20 év oktatási stratégiáit, fejlődését, a koncepciók változásait. Kutatásom során három területre fordítottam kiemelt figyelmet, melyek a következők: az általános, a közép valamint a felsőoktatás. A felsőoktatás területén belül a könyvtári informatika fejlesztési területre fektettem a hangsúlyt. Vizsgálatomat mindhárom területen arra fókuszáltam, hogy melyik algoritmus leíró eszközöket részesítik előnyben, melyek azok az alapnyelvek és algorithmusok, amelyeket felhasználnak és alapjául szolgálnak a későbbi ismereteknek. Valamint, hogyan vezetik rá a tanulókat, hallgatókat az algoritmikus gondolkodás elsajátítására.

Kulcsszavak: algoritmus, oktatás, informatika

Abstract: The significance of algorithms is revealed even in our daily lives. Each task to be done also has its own algorithm. In the process of creating computer programs we rely on that as well; the design of the appropriate algorithm is considered to be primary, therefore it is most highlighted in education. In addition, algorithms serve as a basis for numerous disciplinary areas and they play a highly important role in teaching mathematics, information technology and library informatics (for instance programming, web technology, script languages, et cetera)

I have investigated its teaching strategies, development and changing concepts of the past 20 years. In my research I have paid special attention to the following three fields: primary, secondary and tertiary education. In the field of tertiary education my emphasis has been laid on library informatics development. In all three fields I have focused my investigation on the preferred algorithm description tools, along with the application of those basic computer languages and basic algorithms that serve as a basis for further studies. I have also examined how students and undergraduates are led to algorithmic thinking.

Keywords: algorithm, education, IT

1. Bevezetés

Kutatásomban megvizsgáltam az elmúlt 20 év oktatási stratégiáit, fejlődését, a koncepciók változásait, az oktatásban használt elektronikus és hagyományos, tananyagok alapján. 54 tankönyvet vizsgáltam meg. Kutatásom során három területre fordítottam kiemelt figyelmet, melyek a következők: az általános, a közép valamint a felsőoktatás. A felsőoktatás területén belül, a könyvtári informatika fejlesztési területre fektettem a hangsúlyt. Vizsgálatomat mindhárom területen arra fókuszáltam, hogy melyik algoritmus leíró eszközöket részesítik előnyben, melyek azok az alapnyelvek és alapalgoritmusok, amelyeket felhasználnak és alapjául szolgálnak a későbbi ismereteknek. Valamint, hogyan vezetik rá a tanulókat, hallgatókat az algoritmikus gondolkodás elsajátítására. További célom az volt, hogy felkutassam mennyiben más az e-learninges környezetben használt módszerek, milyen új lehetőségeket biztosít a könnyebb megértés érdekében. Az algoritmusok leírásban rejt-e magában új lehetőségeket? Mennyiben más e-learninges környezetben az elsajátítandó tananyag? Valamint az egyes algoritmusokat milyen módszerekkel írja le, a hagyományos eszközök mellett, annak segítségével jobban alkalmazhatóak-e azok a leíró eszközök, amelyeket a hagyományos könyvekben legtöbbször csak említés szintjén alkalmaznak.

2. Leíró módszerek

Az algoritmusok leírására számos módszer áll rendelkezésünkre. A hazai szakirodalmakban négy algoritmus leíró módszerrel találkozhatunk. Ezek a következők:

- **szöveges:** A szöveges leíró módszer a legkevésbé pontos, és a legterjedősebb, de mindenki számára érthető természetes nyelvű leírás. Nyelvezetében az emberi beszédhez közel áll. Egymást követő mondatokkal, hétköznapi nyelven, mondatszerűen írható le. Előnye, hogy közelebb áll a gondolkodásunkhoz. Természetes nyelvű leírásként is definiálható.
- **metanyelv:** Az adott nyelv (pl. angol, magyar) szavainak, kifejezéseinek segítségével, illetve szimbólumok (pl. \rightarrow , \leftarrow stb.) használatával írja le az algoritmus vezérlési szerkezeit, végrehajtandó műveleteit. Az elemi tevékenységeket és feltételeket megállapodás szerint alkalmazzuk. Jól definiált, kizárólagos szabvány nem létezik rá. A hazai és nemzetközi szakirodalom gyakorlatában kialakult egyfajta szintaxis kisebb-nagyobb eltéréseket mutat a különböző szerzők, kiadók műveiben. Nevét onnan kapta, hogy szerkezetében hasonlít a programozási nyelvekre, mégsem követel meg aprólékos precizitást, így kényelmesen fogalmazhatjuk meg mondanivalónkat. Valahol a természetes emberi nyelv, és a programnyelvek közt helyezkedik el. Megnevezése, többek között a következők: pszeudokód, pszeudo-nyelv.
- **grafikus metanyelv:** A folyamatok megtervezését, leírását segítő grafikus eszközök a vizuális megjelenítést biztosítják. Ebbe a csoportjába tartozik a vizuális megjelenítő nyelvek: ezek a folyamatábra, srtuktogram, és a Jackson módszer. Könnyen megtanulható, jól

használható nyelv. A folyamatok megtervezését, ill. a folyamatok leírását többféle grafikus eszköz segíti. A folyamatábrával leírt szerkezet az egyik legegyszerűbb, szimbólumokat használó tervezési módszer. A struktogram-ban történő tervezés, hasonlóan a folyamatábrához, itt is fentről lefelé történik, az egész feladatnak egy téglalapot feleltetve meg. A részfeladatoknak megfelelően a téglalapot részekre osztjuk.

- **programnyelv:** a programok is tekinthetők egy algoritmus leírásának (Pl. PASCAL)

Jackson módszer

Ezzel a módszerrel sajnos csak nagyon kevés szakirodalom foglalkozik. Szerkezetében egy hierarchia-ábrát hoz létre. A probléma, illetve a megoldás szerkezetének ábrázolására alkalmas. Alapfilozófiája, hogy egy probléma logikai szerkezete egyértelműen előállítható az általa kezelt vagy előállított adatok szerkezete alapján. A Jackson módszer, gondolkodási egysége a struktúra, amely egy olyan szerkezeti egység, amely megadja, hogy egy magasabb szintű dolog hogyan bomlik tovább komponensekre. A struktúra, ebben a módszerben is lehet elemi vagy összetett. Az elemi struktúrák azok, amelyeket nem bontunk tovább. Az összetett struktúra lehet: szekvencia, szelekció vagy iteráció. Az összetett struktúrák komponenseit Jackson alapján „rész”-nek nevezzük és egy összetett struktúrán belül a részek között egyféle (homogén) kapcsolat lehet, amit a struktúra típusa határoz meg.

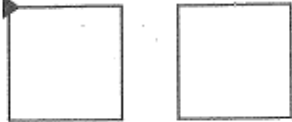
3. Általános iskolai algoritmusok oktatása

Az algoritmusok oktatása általános iskolában az 5-6. osztálytól elkezdődik. Oktatásában két irányelv figyelhető meg. Az algoritmusokat megalapozó ismeretek átadása, és a konkrét kezdeti lépések.

A LOGO nyelv használatával egyszerű, de ugyanakkor nagyon látványos „program”-ok írhatóak. A rögtön megjelenő produktum, ösztönző hatása pozitívan befolyásolja a további munkát. Jó alapot biztosít, játékosan vezeti rá a tanulókat az utasítások szekvenciálisan történő megadására, rávezeti a ciklusok alkalmazására, valamint a teknőc taníthatósága révén az eljárások definiálására. Majd ezt követi az algoritmusok tényleges bevezetése.

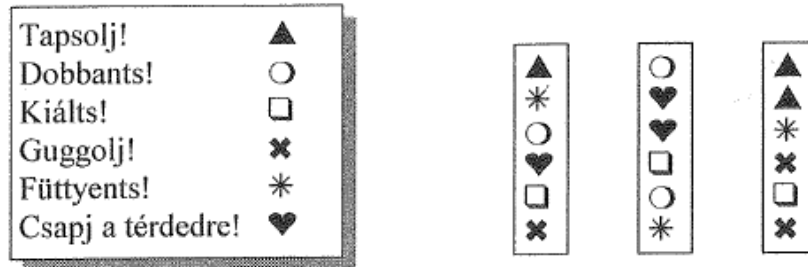
Egyes könyvekben, mint ahogyan az 1. ábrán is látható Mészáros Tamásné: Informatika számítástechnika I. című könyvében, ezen elveket LOGO használata nélkül, egy képzeletbeli „Robot” irányításával sajátítják el. A parancsok megegyeznek a LOGO nyelv parancsaival.

Tanulj	négyzet n ismételd 4 előre n jobbra 90
Kész	
program	rajzolj négyzet 2 ne rajzolj előre 4 rajzolj négyzet 2 ne rajzolj



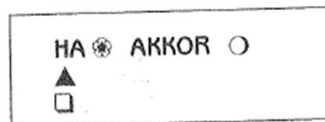
1. ábra Robot tanítása

Más irodalmakban cselekvésre készítő jelek bevezetését alkalmazzák, amely hidat képez a tényleges cselekvés és a szekvenciálisan leírt utasítások között. Így, egyfajta játékkal vezetve rá a gyermekeket az algoritmus 3 alapvető szerkezeti egységére: a szekvenciára, a szelekcióra és az iterációra. (2. 3. ábra)



2. ábra Szekvencia

Ha a tábla sarkába virágot rajzolunk, akkor a mellette levő részletet is végrehajtjuk. Ha nincs a tábla sarkában a virág, akkor a hozzá tartozó részt meg sem nézzük!



Ha van virág, akkor dobbantunk, utána tapsolunk, majd kiáltunk. Ha nincs virág, akkor csak tapsolunk, majd kiáltunk.

3. ábra Szelekció

A háromszög a „Tapsolj!”, a kör a „Dobbants!”, a négyzet a „Kiálts!”...stb. Ezen utasítások meghatározásával, egyértelmű hozzárendelést képez a cselekményhez. Felhívja a figyelmet, hogy az utasításokat csak fentről lefelé lehet végrehajtani az előírt lépéssorozatnak megfelelően, és valahányszor hajtódik is végre, mindenki ugyan ezt, ugyan úgy fogja végrehajtani, mint maga a számítógép. A játék továbbfejlesztésével megismerteti az elágazást és a ciklust.

A másik irányelv a konkrét kezdeti lépések. Megfigyelhető, hogy bevezetésként mindenhol a szöveges algoritmus leíró nyelvet használják - konkrétan a természetes nyelvű leírást -, valamint a metanyelveket (pszeudo-kód). Ezek a feladatok általában az életből, a mindennapokból vett cselekvéseket algoritmizálják, vagy a gyermekekhez oly közel álló mesékből vett részletet. (4. ábra) A grafikus metanyelvek közül a folyamatábrával és a struktogrammal ismerkedhetnek meg a tanulók. (5. ábra) Az algoritmusok bevezetésére véleményem szerint a hasonló meserészletek leírása a leghatékonyabb módszer - legyen szó bármelyik korosztályról -, mert elúzi a tárgyal szemben támasztott kétségeket, félelmeket valamint felkelti az érdeklődést.



6. ábra Animáció

Az általános iskola 7-8. osztályában az eddigi ismeretekre alapozva elhagyják a természetes nyelvű leírást, és a pszeudo-nyelvet, folyamatábrát valamint struktogramot alkalmazzák algoritmus leíró eszközként. Az eddigi ismeretek felszínre hozása érdekében általában a mindennapi életünkből vett példák segítségével elevenítik fel (teakészítés, telefonálás, fogmosás, borsófejtés...stb.).

A következő alapalgoritmusok oktatása figyelhető meg: összegzés, szorzás; megszámlálás; eldöntés; kiválogatás; maximum, minimum kiválasztás; csere; rendezés: (egyszerűsített) minimum-, és maximum kiválasztásos. Az érdeklődés fenntartására optimális lehetőséget nyújt a játékok algoritmusának megírása. Pl. számlétra; számkitaláló; kő, papír, olló. Az algoritmusokra épülő nyelvek a LOGO, és a Quick BASIC.

A 7. osztályos e-learninges tananyagokban teljesen más szemléletmód figyelhető meg. Az alap algoritmusok hagyományos leíró eszközök segítségével történő bevezetése szemlélet helyett a LOGO nyelv haladó szintre történik az emelése. Az Imagine LOGO programozásban elsajátítják a rajzlisták használatát, események, objektumok kezelését, megismerkednek az osztályokkal és az örökléssel. Megtanulják a párhuzamos folyamatok kezelését. Ezt a FreePascal programnyelv bevezetése követi, amelyben a alapalgoritmusokat párhuzamosan a pszeudo kód segítségével tanulják.

4. Közép és felsőoktatási algoritmusok oktatása

A 9-10. osztályban az algoritmusok bevezetésére, hasonlóképpen az előző évfolyamhoz, a pszeudo-kódot és a folyamatábrát alkalmazzák, új leíró eszközként megjelenik a PASCAL programnyelv. A struktogram ezen évfolyamon is általában csak bemutatásra kerül, feladatokban ritkán, csak az egyszerűbbeknél használják. A feladatok összetettsége, bonyolultsága követi a korosztálynak megfelelő elvárásokat.

Az előző évekhez hasonlóan itt is a konkrét példán, főleg matematikai problémák meghatározására koncentrálnak, valamint szerkezetének meghatározásaként történik az ismeretátadás. Alpalgoritmusokat pszeudo-nyelven, de a legtöbb esetben az adott programnyelven vizsgálják. Az eddigi ismereteket felhasználva optimalizálják az algoritmusokat, hatékonyabbá téve azt. Az alapalgoritmusok közül új ismeretként szolgál a

tömbkezelő algoritmusok (tömbfeltöltés), a hatékonyabb rendezés, valamint az egyszerű keresők.

Az emelt szintű algoritmusokhoz az oktatás folyamán csak kevés olyan osztály van aki eljut, ezt a főiskolai képzésben ezt figyelembe kell venni.

Emelt szintű algoritmusok a következők: a buborékos-, és beillesztéses rendezés, a keresések (lineáris és bináris keresés), az összefuttatás és a backtrack.

Az algoritmusok számítógépes megvalósítása Pascal és Visual Basic Script nyelveken történik.

E-learninges környezetben ebben a korosztályban az oktatás során nem lelhető fel kimagasló különbség. A 9-12. osztályos tanulók algoritmus oktatása meghatározó a későbbi tanulmányaik során. A legfontosabb, nem az, hogy minél több ismeretet elsajátítsanak, hanem az, hogy megértsék amit tanulnak, hogy mindez nem új, csupán más szemlélet mód. Fontos, hogy demonstrálva legyenek az algoritmusok, hogy el tudja képzelni a diák pl. a bináris rendezést. Mert ha az elvét megérti, akkor az algoritmust vagy a programot is könnyebben fel tudja építeni köré. Az e-learninges környezet lehetőségei szinte kimeríthetetlenek.

A felsőoktatásban az eddig megismert algoritmus leíró nyelveken kívül megismerkedhetünk az un. Jackson módszerrel. Az általam megvizsgált könyvek mindegyikében bemutatásra kerül az algoritmus leíró nyelvek közül a pseudo-kód, folyamatábra és a struktogram. Az előző tanulmányokra alapozva, ezek leírása célirányos, lényegre törő.

Az alapalgoritmusok pseudo-kód segítségével minden esetben megtalálhatóak, legtöbb esetben folyamatábrával és Pascal programnyelven is. A középiskolában vett alapalgoritmusok a következőkkel egészülnek ki. Shell beszűrő rendezés, Fibonacci-sorozat, euklidészi-algoritmus, prímszámkereső algoritmusok. Az e-learninges felület számos animációval teszi látványosabbá, szemléletesebbé az elsajátítandó tananyagot. Jelentős különbség nem fedezhető fel a tananyagok között.

A felsőoktatásban valósul meg a haladó szintű algoritmusok oktatása. Kutatásomban az alapalgoritmusokat helyeztem a középpontba, további kutatásomban tervezem kutatásom ilyen irányú kiterjesztését. Az alapalgoritmusokra épülő nyelvek: Turbo Pascal, Turbo Basic, Quick Basic, JavaScript, C#, C++....

5. Összegzés

Az általános iskolában alkalmazott játékos bevezetés leköti a diákok figyelmét, ösztönzi a minél több ismeret megszerzésére. A mindennapi cselekvések folyamatának és a meserészletek algoritmizálása észrevétlenül rávezet az algoritmusok alapelveire. Tapasztalataim szerint a struktogram használata nehézkes, bonyolultabb algoritmusok leírása nehezen kivitelezhető, és egy idő után a helyhiánnyal is meg kell küzdenünk. A középiskolai szinten történő algoritmusok oktatásában véleményem szerint nagyobb hangsúlyt kellene kapniuk az alapalgoritmusoknak. Fontosnak tartom az alapelvek rögzülését, inkább több gyakorlással és ha nem is az összes alapalgoritmus megtanításával, de amit megtanulnak

azokat valóban értsék és tudják alkalmazni, ne csak „bemagolják”. Nagyban megnehezíti a főiskolai tanulmányokat, ha az alapok nem megfelelőek. A Jackson módszer hierarchia szerkezete igen látványos jól használható, tapasztalatom szerint mégis nehezen küzdenek meg a hallgatók az elsajátításával.

Az algoritmus-orientált képzés jellegzetes, eddig nem vagy nem tudatosan használt képességeket, rálátást igényel. Ennek elsajátítása mérföldkő az oktatási folyamatban. Ezért nagyon fontos, hogy az algoritmusok oktatása, az oktatás hagyományos és új módszerei reflektorfénybe kerüljenek. Eddig végzett kutatásaim is megerősítettek abban, hogy az algoritmusok tanítása és tanulása számos problémát vet fel. Szükséges új eszközök, módszerek felkutatása, fejlesztése.

6. Köszönetnyilvánítás

Ezúton is szeretném megköszönni Dr. habil. Boda Istvánnak és Lengyelné dr. Molnár Tündének segítőkészségüket, tanácsaikat, építő jellegű kritikáikat, szakmai odaadásukat és bizalmukat. Segítségük nélkül nem tudnám elérni kitűzött céljaimat.

Irodalomjegyzék

- [1] Angster Erzsébet: Programozás Tankönyv, *4KÖR Bt.* (1999)
- [2] Angster Erzsébet: Programozás tankönyv I., Strukturált tervezés Turbo Pascal, *Akadémiai Nyomda* (1999)
- [3] Aszalós János, Erki Irén: Bevezetés a strukturált programozásba. *KSH Nemzetközi Számítástechnikai Oktató és Tájékoztató Központ* (1980.)
- [4] Bárdos Attila, Lócs Gyula, Rácz Lajos: A programozás alapjai, *LISI Oktatóközpont*
- [5] Csantó Béla, Ludányiné Prém Judit: Informatika 7. *Pedellus Novitas Kiadó* (2003.)
- [6] Csantó Béla, Ludányiné Prém Judit: Informatika 8. *Pedellus Novitas Kiadó* (2003.)
- [7] Csőke Lajos, Garamhegyi Gábor: A számítógép-programozás logikai alapjai *Nemzeti Tankönyvkiadó Rt.* (1997.)
- [8] D.E. Knuth: A számítógép programozás művészete 1. Alapvető algoritmusok, *Műszaki Könyvkiadó* (1994.)
- [9] Devecz Ferencz, Juhász Tibor, Makány György, Végh András: Informatika 10. *Nemzeti Tankönyvkiadó* (2008.)
- [10] Dr. Koncz József: Technika – Informatika *Agria Kiadó Kft* (1990.)
- [11] Dr. Perge Imre: Bevezetés az informatikába. *EMTEX-JATEX* (1993.)

- [12] Énekes Ferenc, Hanák D. Péter, Horváth Attila, Horváth Katalin, Hubert Tibor, Keczer Zoltán, Koltai Márta, Kőhegyi János, Németh Piroska, Sárkány Ernő, Zsakó László, Gulyásné Lőrincz Ilona, Környei László, Lörentei Tamás, Orbán Anna, Pethő József, Szlávi Péter, Varga Balázs: Számítástechnika középfokon, *OMIKK* (1987.)
- [13] Fazekas Ildikó, Simon Gyula: Számítástechnika 10. osztály számára, *Pedellus Novitas Kiadó* (2000)
- [14] Fazekas Sándor, Ziliziné Bertalan Gabriella: Számítástechnika 5-6 osztály számára, *PEDELLUS Novitas Kiadó 2.* (2000.)
- [15] Gordon Erzsébet, Körtvélyesi Gézáné, Sós István, Székely Zoltán: PASCAL programozási nyelv, *Számítástechnika-alkalmazási Vállalat, 6.* (1988.)
- [16] Hetényi Pálné Dr. (szerk.). Számítástechnika középfokon, *OMIKK* (1994.)
- [17] Horváth Tibor, Kiss Marietta: Számítástechnika iskolásoknak II. *Nemzeti Tankönyvkiadó* (1996.)
- [18] Járdán Tamás, Pomaházi Sándor: Adatszerkezetek és Algoritmusok, *EKF Líceum Kiadó* (1998.)
- [19] Jávor András, Benkő Tiborné: Diszkrét rendszerek szimulációja. *Műszaki Könyvkiadó* (1979.)
- [20] Jozef Hvorecky, Jozef Kelemen: Ötlettől az algoritmusig. *Tankönyvkiadó* (1987.)
- [21] Kiss Albert, Ludányiné Prém Judit: Informatika tankönyv a 9-10. osztály számára. *Pedellus Novitas Kft, 2.* (2004.)
- [22] Koncz József: A PASCAL programozási nyelv elmélete és gyakorlata. *EKTF Líceum Kiadó* (1999.)
- [23] Körösné Miklósi Márta, Mészáros Tamásné: Adatok és algoritmusok I., *Nemzeti Tankönyvkiadó* (2000.)
- [24] L. Artiaga, L. D. Davis: Algoritmusok és FORTRAN programjaik. *Műszaki Kiadó* (1997.)
- [25] Lipschutz S. Adatszerkezetek: *McGraw-Hill Inc.-Panem Kft.* (1993.)
- [26] Lovász László, Gács Péter: Algoritmusok. *Műszaki Kiadó* (1978.)
- [27] Mágoriné Huhn Ágnes: Algoritmusok és adatszerkezetek. *JGYF Kiadó* (2000.)
- [28] Mészáros Tamásné: Informatika számítástechnika I. *Poli-Kvitt Kiadó* (1992.)
- [29] Mészáros Tamásné: Informatika 6., *Nemzeti Tankönyvkiadó* (2002.)
- [30] Molnár Csaba, Sági Gábor: VII. Programozás. *BBS-E Bt.* (2000.)
- [31] Molnár Csaba, Sági Gábor: Programozás Turbo Pascal nyelven. *BBS-E Bt.* (2001.)

- [32] Nagy Eleménné, Nagy Elemér: Rendszertervezés (főiskolai jegyzet). *SzTE SzÉF* (2005.)
- [33] Niklaus Wirth: Algoritmusok + Adatstruktúrák = Programok. *Műszaki Könyvkiadó* (1982.)
- [34] Pitrik József: Informatika, könyvtárhasználat, számítástechnika. *Apáczai Kiadó* (1998.)
- [35] Rónyai Lajos, Ivanyos Gábor, Szabó Réka: Algoritmusok, *Tipotex* (1998.)
- [36] Roszak T: Az információ kultusza. *Európa Könyvkiadó* (1990.)
- [37] Schneider Ferenc: Basic Algoritmusok Basic nyelven minden gépen, *APC-Stúdió BT.* (1993.)
- [38] Simon Gyula: Számítástechnika középiskolásoknak. *Pedellus Bt.* (1995.)
- [39] T. H. Cormen, C. E. Leiserson, Ronald L. Rivest: Algoritmusok, *Műszaki Könyvkiadó* (2003.)
- [40] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest: Algoritmusok. *Műszaki Kiadó, 1.* (1997.)
- [41] Tóth Tamás: Informatika 8. *Nemzeti Tankönyvkiadó* (2003.)
- [42] Tóth Tamás: Informatika 7. *Nemzeti Tankönyvkiadó* (2002.)
- [43] Tömösközi Péter: Algoritmizálás alapjai. *Médiainformatikai kiadványok* (2011.)
- [44] Turcsányiné Szabó Márta: A LOGO programozási nyelv. *Műszaki Könyvkiadó* (1986.)

Elektronikus irodalom:

- [45] Aszalós László, Herendi Tamás: Algoritmusok, elektronikus kiadvány (http://www.tankonyvtar.hu/hu/tartalom/tamop425/0046_algoritmusok/ch03s02.html)
Megtekintve: 2012. június)
- [46] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 4. évfolyam, Infotechnológia/Algoritmusok megismerése, elemzése, egyszerű algoritmusok megalkotása, algofejtörők
- [47] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 4. évfolyam, Algoritmizálás, adatmodellezés
- [48] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 5. évfolyam, Infotechnológia (problémamegoldás informatikai eszközökkel és módszerekkel)
- [49] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 6. évfolyam, Algoritmizálás, adatmodellezés
- [50] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 7. évfolyam, Algoritmizálás, adatmodellezés

- [51] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 8. évfolyam, Algoritmizálás, adatmodellezés
- [52] tudasbazis.sulinet.hu (Megtekintve: 2012. május) Informatika 9-12. évfolyam, Algoritmizálás, adatmodellezés, programozói ismeretek

Új, ingyenesen elérhető oktatói portálrendszer a felsőoktatásban

New freely available educational portalsystem in higher education

Kvaszingerné Prantner Csilla

Eszterházy Károly Főiskola/Médiainformatika Intézet, Humáninformatika Tanszék

csilla@ektf.hu

Absztrakt: 2013-ban megtervezésre került egy új oktatói portálrendszer az Eszterházy Károly Főiskolán a TÁMOP 4.2.2. C pályázat keretén belül. Az eszközzel felsőfokú intézményekben oktató tanárok számára kívánunk kész, alaposan dokumentált telepítési csomagot adni, olyat, amellyel bármilyen szakos oktató könnyedén létre tudja hozni, továbbá karbantartani és működtetni saját oktatói portálját különösebb informatikai segítség nélkül. A projekt keretén belül rövid kurzusokat is tervezünk, amelyek során bemutatjuk a portálsablon használatát a felsőoktatásban lévő tanárok számára, s javaslatokat teszünk annak minél hatékonyabb kihasználási módjára.

A portálrendszer koncepciója a szoftvertervezésnél használt Felhasználó-központú tervezés folyamataira épült. Így a projekt keretén belül felmérésre kerültek az oktatók s a hallgatók igényei egyaránt, az oktatást-tanulást segítő portálokkal kapcsolatosan. Többéves rendszerfejlesztői és oktatói tapasztalatokon, továbbá a felmérés eredményein alapulva megtervezésre került a portál, amelynek erőssége abban rejlik, hogy azt a minimum eszközt tartalmazza, amely a felmérések szerint mindkét csoport tagjai számára fontosak vagy bármely oldal felől külön igényként merült fel. Igyekeztünk tehát csak a legszükségesebb elemeket beépíteni, annak érdekében, hogy a rendszer átlátható legyen és működtetése se legyen nehézkes vagy bonyolult.

A megtervezett rendszer Drupal 7 tartalomkezelővel készült, mely számos tesztelési folyamaton ment keresztül. A technikai teszteken túl felhasználói teszteket is végeztünk, továbbá szemmozgáskövető eszköz használatával webergonómiai teszteket is folytattunk. A weboldal felépítésének teljes folyamatát s a rendszer koncepcióját, struktúráját szeretném ismertetni előadásomban.

Kulcsszavak: oktatói portál, felhasználó-központú tervezés, szemmozgáskövető, webergonómiai teszt, Drupal 7

Abstract: In the Károly Eszterházy College a new education portal system was designed, within the scope of tender TÁMOP 4.2.2. C. With this tool we aim to provide a ready-to-use, thoroughly documented installation package for teachers of higher education. Using this tool any teacher can easily create, maintain and operate his own education portal, without any special IT assistance. We plan short courses within the project, where we introduce the usage of the portal template to the teachers of higher education, and we make recommendations for its as effective use as possible.

The conception of the portal template is built on the processes of user friendly planning used for software planning. Thus the requirements of teachers and students were assessed within the project, and the requests of students related to portals supporting teaching and learning. The portal was designed based on many years of system development and education experiences and the the results of assessment. Its strength comes from the usage of a minimal set of tools, which, according to the assessment, are important for members of both groups or was a separate demand of any groups. Thus we made efforts to include only the most necessary elements in order to make the system transparent and avoid its operation being difficult or complicated.

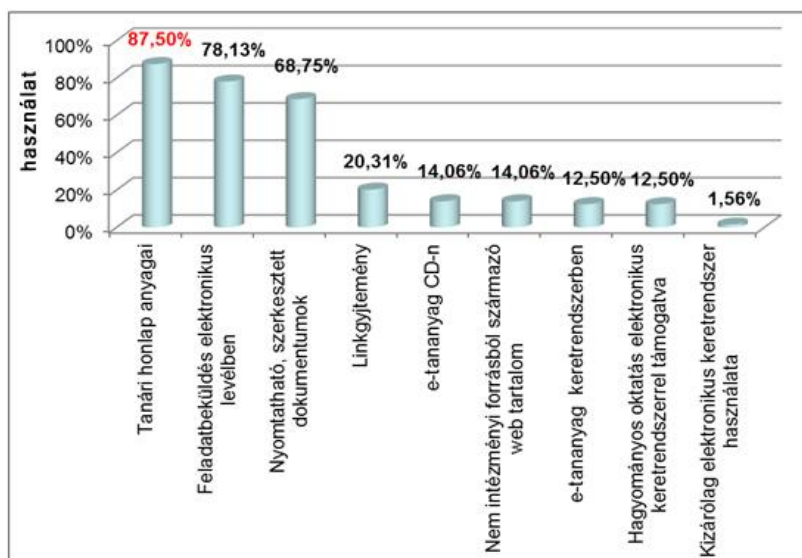
The planned system was made with Drupal 7 CMS-Contact Management System, which underwent several testing phases. Beside technical tests we made user tests as well, and we also made web ergonomics tests by eyeball movement tracking. I would like to introduce the full website building process and the concept and structure of the system in my presentation.

Keywords: educational portal, User-Centered Design, eyetracker, web ergonomics test, Drupal 7

1. Bevezetés

A jól kialakított, jól strukturált és hasznos tartalommal bíró tanári portálok léte elengedhetetlen a „digitális kor” tanulói számára. A diákoknak megbízható és biztos kiindulópont egy oktató igényes kivitelezésű website-ja, ezeken az oldalakon ugyanis a tanár feltétlenül megbízható információkat, tananyagokat és segédleteket oszt meg a diákokkal. Egy ismert oktató szakmai oldala vagy linkgyűjteménye jóval hitelesebb forrás, mint bármely kereső által kidobott találati lista, vagy linkgyűjtemény.

Az Eszterházy Károly Főiskolán az utóbbi években készült egy felmérés a hallgatók körében, melynek eredménye az volt, hogy a tanulást támogató elektronikus megoldások közül a diákok az oktatók saját weboldala (portálja) nyújtotta segítséget veszik 87,5%-kal a legtöbben igénybe (Lengyel, 2009).



1. ábra. A hallgatók által, tanuláshoz segítségükre választott, elektronikus eszközök igénybevétele

2. A fogalmi keret és az eljárás folyamata

A létrejött oktatói portálkonceptió a Felhasználó-központú tervezés módszerén (Corry–Frick–Hansen, 1997) alapszik. Fontos cél volt az, hogy a tervezett rendszer valóban olyan összetevőket és szolgáltatásokat tartalmazzon, amelyekre az oktatóknak és a hallgatóknak egyaránt – saját véleményük szerint és a rendszer tervezői szerint is – nagy szükségük van.

Ahhoz, hogy valóban jól használható eszköz valósuljon meg fontos az, hogy a felület kezelése, tartalmának frissítése, a megosztandó tananyagsegédletek feltöltése és a diákok felé történő gyors információkövetítés egyaránt gördülékenyen és hatékonyan megvalósítható legyen vele. Mindezek eléréséhez technikai és didaktikai szempontból is jól átgondolt rendszert kellett létrehozni. A projekt eredményeként így nem csak egy keretrendszer jött létre, hanem egy olyan koncepció is kidolgozásra került, amely az oktatás-tanulás folyamatot és az oktató-diák közti kommunikációt valóban hatékonyan támogatja.

Igyekeztünk csak a lehető legegyszerűbb és legszükségesebb elemeket beépíteni a rendszerbe, annak érdekében, hogy annak működtetése és átláthatósága ne legyen bonyolult. Ezzel igyekszünk megkímélni az oktatókat a felület kezelése közben adódó felesleges munkafolyamatoktól továbbá mentesíteni a diákokat, s magát a rendszert is olyan funkcióktól, amelyek általában nem is kerülnek kihasználásra a legtöbb oktatást támogató rendszerben vagy LMS (Learning Management System) rendszerekben. A létrehozott portálkoncepció ereje tehát az egyszerűségében, könnyű kezelésében és a felmért igényekhez való igazodásban rejlik; továbbá ki szeretném emelni azt, hogy a tervezés és kivitelezés során nagy figyelmet szenteltünk a webergonómiai, a használhatósági és az akadálymentes-webbel kapcsolatos elvekre.

2.1. Felhasználó-központú tervezés

Annak érdekében, hogy az elkészítendő portál mind az oktatók, mind a diákok számára valóban hasznos és könnyen használható legyen, alkalmaztuk a Felhasználó-központú tervezés módszerét, az egész folyamat ezen a modellen alapul, a modell egyes lépéseinek a projektben konkrét lépések feleltethetők meg.



2. ábra. A Felhasználó-központú tervezés modellje magyárosítva [3]

2.2. Az eljárás folyamata

1. Terv:

Pontosan megfogalmaztuk a fejlesztendő oktatói portállal kapcsolatos céljainkat, megneveztük és körvonalaztuk a célcsoportokat. Szakmai igényességgel vettük számba, s neveztük meg a weboldalon elhelyezni kívánt lehetséges tartalomelemeket és funkciókat. A fenti folyamatok eredményeképpen született két kérdőív, amelyek segítségével az oktatói portálok két célcsoportjának (oktatók és diákok) igényei a tervezett website-tal kapcsolatosan könnyen felmérhetővé váltak.

2. Kutatás, felmérés:

Az egyik kérdőívvel az oktatók, a másikkal a diákok igényeit mértük fel a tervezett weboldal tartalomelemeivel és funkcióival kapcsolatosan. Jogosultsági kérdéseket is felvetettünk a weboldalon lévő egyes tartalmi elemek illetve funkciók elérhetőségével kapcsolatban: melyeket lehessen bármely látogató számára, s melyeknek adott kurzusra regisztráltak számára elérhetőnek lenniük.

3. Rendszertervezés:

- a. A rendszer technikai tervezése a W3C Organization legújabb szabványain alapul (HTML5, CSS3, WCAG 2.0 A).
- b. A tartalmi elemek, a funkciók kiválasztása és azok megtervezése a következő pilléreken nyugszik:
 - Kérdőívek eredményei.
 - Általános webergonómiai szabályok és használhatósági tényezők figyelembe-vétele.
 - Az oktatásban szereplő webes applikáció-tervezés minőségi faktorainak figyelembe vétele az EEQM (E-Learning Evaluation Quality Model) alapján (Marvromoustakos–Papanikolau, 2009).
 - Rendszertervezési és -fejlesztési szakmai tapasztalatok.
 - Oktatói tapasztalatok és többféle e-learning rendszer több mint egy évtizedes múltra visszatekintő használatának gyakorlata.

4. Megvalósítás:

A kapott eredmények figyelembevételével a kialakított koncepció megvalósítása Drupal 7 tartalomkezelő rendszerrel.

5. Tesztelés:

- a. A rendszer technikai tesztelése.
- b. A felhasználók kérdőíves kikérdezése a kész rendszer működéséről, használatáról.
- c. Felhasználói tesztek manuális elvégzése mind a hallgatók mind az oktatók körében.
- d. A portál ergonómiai tesztelése a következő eszközökkel: TestCompleted szoftver, Ogama szoftver szemmozgáskövető hardver.
- e. Az egyes funkciók tesztelése EEQM modell szerinti értékelése – minőségbiztosítási szempontok.

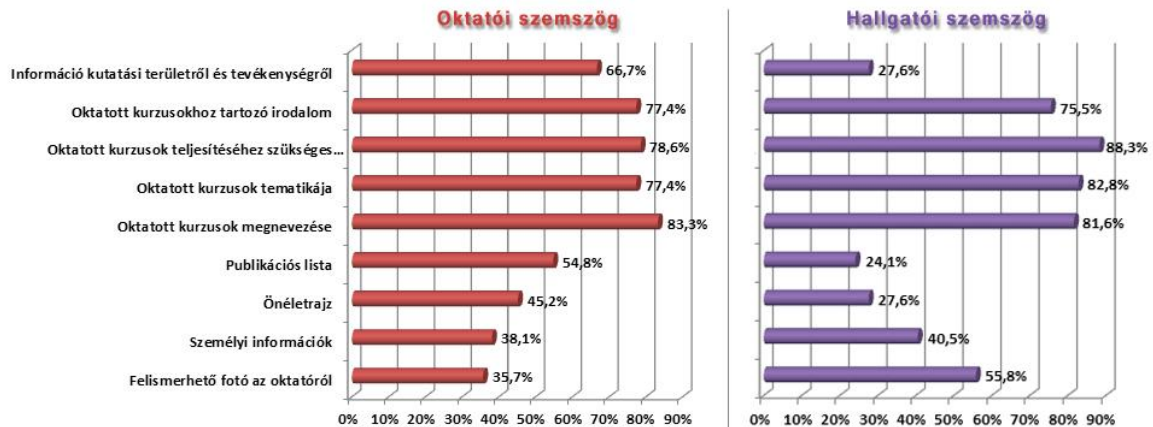
6. A hatodik lépésben történik rekurzívan a hibák javítása:

A teszt során feltárt hibák javítása. A hibás összetevőket vagy funkciókat külön-külön visszavezetjük az első lépésre. Azaz egy adott blokkban vagy egy adott mikro-alkalmazásban előkerült probléma kijavítása érdekében szükséges az adott blokkot vagy alkalmazást újbóli tervezés alá vetni, s újból végigvezetni a Felhasználó-központú tervezés minden egyes lépésén. Ezzel a módszerrel rekurzív módon javíthatóak ki a hibák, azaz az egyes folyamatokon addig haladunk ismételten végig, míg az újbóli tesztekkel jó eredményeket nem kapunk. Amennyiben minden részlet problémamentessé válik, azzal a rendszer egésze az elvárásoknak megfelelő lesz.

3. A felmérések eredményeinek ismertetése

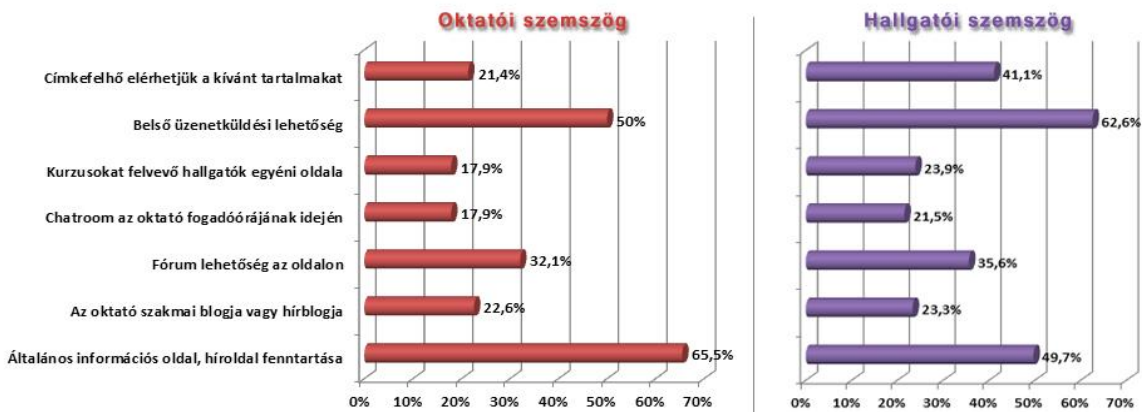
Az oktatók és a diákok tartalmi elemekre és funkciókra való igényeinek összesített és százalékosan kimutatott eredményei jelennek meg az alábbi grafikonokon.

3.1. A tartalommal kapcsolatos igények százalékos aránya



3. ábra. Az egyes tartalmi egységekre való igények eredményei oktatói és hallgatói szemszögből

3.2. A funkciókkal kapcsolatos igények százalékos aránya



4. ábra. Az egyes funkciókra való igények eredményei oktatói és hallgatói szemszögből

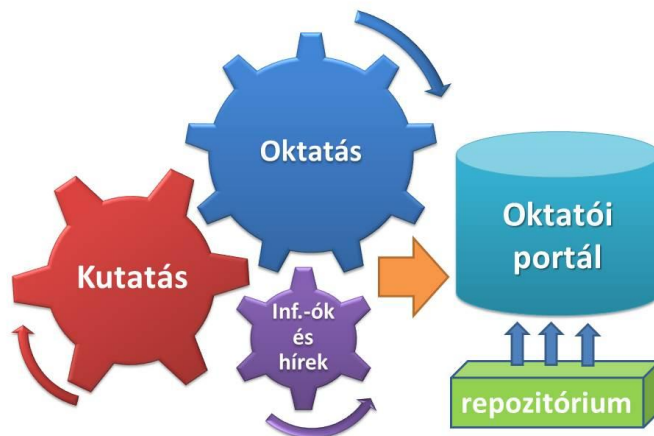
4. A portál felépítésének koncepciójával kapcsolatos döntések

A portálnak három fő, egymástól jól elkülöníthető tartalmi egységből kell felépülnie:

- - információk az oktatóról és elérhetőségéről;
- - oktatással és kurzusokkal kapcsolatos blokk és
- - a tudományos munkával és a kutatási területekkel kapcsolatos információk.

Mindezek mellett folyamatosan frissülő hírek és dinamikus funkciók érhetőek el kurzusonként és az oldal egészét illetően.

A rendszerben a fenti egységeket egy-egy menüpont előredefiniálásával (Bemutakozás, Kurzusok és Tudományos munkák) és a kezdőlap kialakításával teremtjük meg. Az említett menüpontokon felül további, új menüpontok és almenüpontok is létrehozhatóak a rendszerben, így minden oktató a saját oktatására és tudományos munkájára szabhatja az oldalát.



5. ábra. Az új oktatói portálkonceptió tartalmi egységei

4.1. Az oktatóról szóló információk és a hírek terén született döntések

1. Az oktatóról készült fotó, róla szóló általános információk (egység, beosztás, pár soros bemutatkozás vagy jelmondat és elérhetőség) a főoldalon bárki számára látható legyen.
2. Kurzusonként érhetik el a híreket/blogbejegyzéseket illetve naptárbejegyzéseket és hozzászólásokat az adott kurzusra regisztrált diákok. A nagyközönség számára ezek a hírek alapértelmezetten nem láthatóak. Amennyiben ezeket a nem regisztrált látogatók számára is elérhetővé szeretné tenni az oktató, azt jelezheti a rendszer felé, így az kikerül a főoldalra is.
3. A kurzusokban küldött híreket, naptárbejegyzéseket vagy hozzászólásokat a regisztrált diákok e-mailüzenetben is megkapják, amennyiben az oktató ezt az alapértelmezést ki nem kapcsolja a rendszerben.

4.2. Az oktatási területen született döntések

1. Az oktatási tartalmak egy központi repozitóriummal vannak támogatva, amelyhez csak az oktató férhet hozzá. Az oktatók ebből válogatva töltik fel segédanyagait az egyes kurzusok alá.
2. Az oktatási rész a kurzusok köré szervezeten jelenik meg. Minden kurzusnál következő elemek feltöltését biztosítja elsődlegesen a rendszer, s ezek szerepeltetését erősen ajánlja:
 - a kurzus neve;
 - a kurzus tematikája;

- a kurzus teljesítéséhez szükséges követelmények leírása;
- kötelező és ajánlott irodalom.

Ezek a nem regisztrált diákok számára is megtekinthetők, hiszen innen tájékozódhatnak az egyes tantárgyak tartalmáról és követelményeiről, ez segíthet eldönteni, hogy az egyes tárgyakat felvegyék vagy sem:

3. A fentiekén kívül a következő elemek feltöltését biztosítja a rendszer, ezeket a tartalmakat csak a kurzusra regisztrált hallgatók érhetik el:

- oktatási segédletek:
 - tananyag elektronikus formában,
 - órai prezentációk,
 - feladatok (órai feladat, házi feladat),
 - próbatesztek, próba zárthelyi dolgozatok,
 - beadandók pontos leírása, minta beadandók;
- kurzusonkénti hírek/blogbejegyzések (e-mail üzenetek formájában is);
- kurzusnapár;
- kurzusfórum;
- a kurzus címkefelhője.

4. A hír- és a naptárbejegyzések létrehozásakor választható, hogy azok mely kurzus(ok) alatt jelenjenek meg. Minden rendszerben lévő, aktív kurzus a neve előtt egy checkboxal együtt jelenik meg, amelyek segítségével megadható, hogy mely kurzus alatt jelenjen meg az éppen felvitel alatt álló hír- vagy naptárbejegyzés. Továbbá választható az is, hogy a feltöltendő hír- vagy naptárbejegyzés a főoldal hírei és a főoldal naptáreseményei között megjelenjen-e – ezáltal bármely, nem regisztrált felhasználó számára is publikussá váljon.

5. A diákok mindig kurzusra regisztrálnak. A regisztráció alkalmával egy tantárgylistán jelezhetik, hogy mely kurzusok tartalmához szeretnének hozzáférni az adott félévben (ez a választás a későbbiekben módosítható általuk vagy az oktató által). A kurzusregisztráció egy félévre szól, a félév végén a rendszer automatikusan küld egy figyelmeztető e-mailt a diákoknak, hogy a kurzusregisztrációjuk megszűnik, ám ezt meg tudják hosszabbítani újabb félévre egy, a levélben szereplő link segítségével. Azon hallgatók, akik nem hosszabbítják meg egyetlen kurzusra sem a regisztrációjukat és nem is jelentkeznek fel új kurzusra az adott félévben, törölődnek a rendszerből. 5 év elteltével minden regisztrált hallgató automatikusan törölődik a rendszerből, ezzel megelőzve a felhalmozódó felhasználókat az adatbázisban.

4.3. A tudományos munka és kutatási területtel kapcsolatos döntések

A tudományos munkával és kutatással kapcsolatos blokkban az alábbiakat javasolt elhelyezni (az oktató dönti el, kik láthatják a következő tartalmakat):

- CV;
- publikációk;
- szakmai és általános közéleti tevékenység;
- kutatási területek és érdeklődés;
- kutatási projektekből való részvétel;
- hazai és nemzetközi tudományos együttműködések
- aktuálisan futó tudományos projektek rövid leírása/bemutatása;

- jelentősebb szakmai cikkek gyűjteményei (ajánlások);
- jelentősebb tudományos és/vagy kutatói ösztöndíjak, tanulmányutak;
- ajánlott szakdolgozati illetve TDK-s témakörök felsorolása;
- TDK-n szereplő hallgatók és dolgozataik, ill. esetleges OTDK-helyezéseik;
- egyéb hallgatói tehetség gondozás során elért eredmények.

Az oktatók tudományos életüket és aktivitásukat illetően igazán széles skálán mozognak, s arról is megoszlanak a vélemények, hogy mely információkat jelenítsék meg weboldalukon. A fenti okokból kifolyólag nem szeretnénk erős korlátokat adni ezen blokk felépítését illetően. A webportál tulajdonosok a weboldaluk ezen részét tehát nagy szabadsággal, saját tevékenységüknek és „szájízüknek” megfelelően építhetik ki. Ezért szabadon hozhatnak létre a meglévő menüpontokon felül újabb pontokat, s ezeket alpontokra is tagolhatják.

4.4. A technikai megvalósítás területén született döntések

A megtervezett rendszer Drupal 7 tartalomkezelővel készül. Kétféle alkalmazási formája kerül kidolgozásra. Az egyik egy jól dokumentált, úgynevezett Installation Profile (telepítőcsomag), amely a rendszer működéséhez szükséges összes modult magában foglalja, s amely könnyedén telepíthető egy webszerverre. Ezzel a megoldással a rendszer frissítését magának az oktatóknak kell felvállalniuk. A másik megoldás a MultiSite rendszer megvalósítása, amelynek lényege, hogy az oktatói portálrendszer egy központi webszerveren foglal helyet, s arra oktatók regisztrálhatnak. A regisztráció eredményeképpen az oktatók saját rendszerüket tudják konfigurálni s tartalommal feltölteni, ez esetben telepítésre nincsen szükség. A portál regisztrációhoz kötött üzembe-helyezésének és a tárhely biztosításának felvállalása mellett a MultiSite lehetővé teszi azt, hogy a rendszer frissítését is központilag tudjuk ellátni, azaz a rendszer és a szerver biztonsága érdekében nem kell frissítési műveletekkel foglalkozni az oktatóknak, hiszen azt informatikus szakemberek végzik el központilag. A felépült oldalak URL címe a következő: <http://www.tanitlap.ektf.hu>, ezt követően a „/” jel után lévő karaktorsor azonosítja be majd az egyes usereket.

5. Köszönetnyilvánítás

Köszönet az Eszterházy Károly Főiskola Médiainformatika Intézet dolgozóinak: Nagy Dénesnek, Szabó Bálintnak, Tóthné Parázsó Lenkének.

Irodalomjegyzék

- [1] Corry, M. D.; Frick T. W. és Hansen L.: User-Centered design and usability testing of a web site: An illustrative case study, *Educational Technology Research and Development*, Vol 45. Issue 4 (1997), 65–76.
- [2] Marvromoustakos S.; Papanikolau K.: A Quality Evaluation Model for Web 2.0 e-Learning Systems, In: *Experiences and Advances in Software Quality*, Vol X. Issue 5 (2009), 64–74.
- [3] The User-Centered Design: http://www.sapdesignguild.org/resources/ucd_process.asp, (Megtekintés: 2014.07.01.).
- [4] Tünde Lengyel Molnár: Hallgatói kérdőív elemzése, *Elektronikus tanulási környezetek kialakítása I.*. Eger: Líceum Kiadó (2009), 52–63.

A digitális könyvtári hálózat szolgáltatásainak fejlesztése és kutatása

Development and research of digital library network services

Lengyelné dr. Molnár Tünde

Eszterházy Károly Főiskola, Humáninformatika Tanszék

mtunde@ektf.hu

Absztrakt: A növekvő mennyiségű digitális adathalmazok elérhetővé tétele magyar nyelvterületen nem rendelkezik elegendő automatikus tartalomfeltáró eszközzel, kutatásunk során javítani kívánjuk ezt a helyzetet. Az IKT a tudás és tanulás világában – humán teljesítménytechnológiai kutatások és képzésfejlesztés című pályázat része a digitális könyvtári hálózatok szolgáltatásfejlesztési lehetőségeit szélesíteni. Célunk az automatikus tartalomelemzés területén előbbre lépni és az integrált könyvtári rendszer alatt működő referátumkészítést megvalósítani egy automatikusan kivonatoló program előállításával.

A folyamathoz elengedhetetlen a kulcsszó-meghatározó eljárások bemutatása, továbbá a kulcsszavak és az emberek által a szöveg leglényegesebbnek tartott mondatai közti kapcsolat feltárása. A kulcsszavak kivonatolásának elmélete és gyakorlata statisztikai és nyelvészeti eljárásokon nyugszik, melyek közül a legfontosabb módszereket ismertettem előadásom során. Ezen túlmenően bemutatom a humán kivonatolási technika feltárására irányuló felmérések eredményeit, valamint a szignifikáns mondatkiválasztásban rejlő szabályszerűségekre vonatkozó eredményeket, és ezek beépítési lehetőségeit a kivonatoló programba.

Kulcsszavak: kulcsszó meghatározási technikák, kulcsszó, automatikus kivonatolás,

Abstract: The research effort introduced in this presentation aims at improving the limited availability of automatic content exploration devices facilitating the interpretation of continuously increasing digital data via using the Hungarian language. The expansion of the service provision capability of digital library networks is a crucial component of the ICT in the world of knowledge and learning—human performance technology research and training development project. Our principal goal is to make further advances in the field of automatic content analysis and abstract production within the framework of an integrated library system. In order to achieve this objective we will develop an automated extract preparation program.

Said process requires familiarity with key-term recognition procedures along with the exploration of the connection between the crucial words and the manually identified essential sentences of a given text. During my presentation I will introduce the most important statistical and linguistic procedures determining the theory and practice of extracting key concepts. Furthermore, in addition to sharing the results of a survey focusing on non-automated or manual extract preparation technology my lecture will focus on the rules behind significant sentence selection along with respective incorporation options for the latter into the extract preparation program.

Keywords: keyword techniques, keyword, automatically extracting

1. Bevezetés

A digitális könyvtári hálózat szolgáltatásainak fejlesztését segítő kutatások a TÁMOP-4.2.2.C-11/1/KONV-2012-0008 „IKT a tudás és tanulás világában – humán teljesítménytechnológiai (Human Performance Technology) kutatások és képzésfejlesztés” pályázat keretében kerültek megvalósításra. A kutatás egyik fontos eleme a szemantikus web adta lehetőségek kihasználásával gazdagítani az információ-visszakereső eszközök táráát és előre lépni az automatikus tartalomelemzése területén. A növekvő mennyiségű digitális adathalmazok elérhetővé tétele magyar nyelvterületen nem rendelkezik elegendő automatikus tartalomfeltáró eszközzel, kutatásunk során javítani kívánjuk ezt a helyzetet. Jelen tanulmányban a kulcsszó-kutatás fogalmi rendszerének ismertetésén túl a kulcsszó meghatározó eljárások összegyűjtésén van a hangsúly, bemutatva a legjelentősebb kulcsszókeresési technikákat és modelleket.

2. Kulcsszó-meghatározás jelentősége

A szöveg-visszakeresési eljárások alkalmazása a tartalom reprezentációjának megalkotására egyértelmű aktualitással bír. A tartalomelemzési kutatások-, tudományos vizsgálatok tradicionálisnak tekinthetők, azonban a kulcsszó-keresési eljárások, főként az informatika fejlődésének, valamint az információtudomány fejlődésének hatására a 20. század végén, de különösen századunkban új lendületet kapott. A fejlődés mozgatórugója alapvetően a gazdaság, a kereslet és kínálat piacának örökös körforgása, de nem hagyható figyelmen kívül, hogy a hagyományos dokumentumok elektronikussá konvertálása és az elektronikus környezetben keletkező források számának ugrásszerű növekedése is indikálta az információ-visszakeresési eljárások hatékonyságának fejlesztését. Az elektronikus források hozzáférhetősége az internet széleskörű elterjedésével egyre nagyobb teret kap; mivel a problémát nem az adatok, hanem az információ, a tudás elérése jelenti. Az információmenedzsment eszközrendszerén keresztül tehát a kulcsszavak kutatása kitör a szűkebb tudományos diskurzusból és a modern információgazdaság részévé válik. Így a szakmai szempontok gyakran keverednek az üzleti megfontolással, amely minden esetben arra koncentrál, hogy a feldolgozás technikáinak alkalmazását és eredményét rentábilissá tegye. A sokféle megközelítés a kulcsszavak kutatását multidiszciplináris kutatási területté tette, integrálva számos tudományág terminológiáját, különösen az információtudományét, a matematikáét, a nyelvészetét, a statisztikáét, a számítástechnikáét; illetve egyre inkább a közgazdaságtanét. A fentiek alapján körvonalazódó fogalmi rendszert szorosabb és lazább kapcsolatok jellemzik, az egyes határterületek érintkezésének intenzitása meglehetősen változatos képet mutat. Mielőtt tehát a kulcsszavak keresésének, meghatározásának és használatának technikáit, valamint modelljeit bemutatnánk, elengedhetetlen, hogy a terminológiai problémákat feloldva átfogó képet adjunk a vizsgált témáról.

3. A kulcsszó-kutatás fogalmi rendszere

A kulcsszó

A mélyebb elemzés előtt feltétlenül szükséges meghatároznunk a kulcsszó fogalmát, amelyet több aspektusból is jellemezhetünk. Az információtudomány hagyományos

terminológiáját vázolja UNGVÁRY Rudolf és VAJDA Erik a kulcsszót az ismerv számítástechnikai gyakorlatban használt szinonimájaként azonosítja, amelyet főleg a rendszerelméletben és a Web dokumentumaiban alkalmaznak.ⁱⁱ Ezek a tartalmat reprezentáló ismérvek származhatnak a dokumentum szövegéből vagy külső forrásbólⁱⁱⁱ – a kulcsszavak az előbbi kritériumnak felelnek meg. A nyelvészetben ezt a fogalmat nem definiálják, annak ellenére sem, hogy gyakran használják. Statisztikai nézőpontból jellemzően kulcsszónak nevezünk azokat a szöveg tartalmára utaló szavakat vagy kifejezéseket, amelyek a szövegben legalább olyan gyakorisággal tűnnek fel, mint amekkora a felhasználók által meghatározott legkisebb gyakoriság.^{iv} A következő meghatározás átvezet az internetes keresés témakörébe, ahol a hatékony kulcsszavaknak megkülönböztetett szerep jut, ugyanis: „*a kulcsszavak azok a szavak vagy kifejezések, amelyeket a keresőgépek az kívánt információ utáni internetes keresések bázisaként használnak.*”^v

A fentieket összegezve elmondható, hogy a kulcsszavak szerepe az információ feltárásban és visszakeresésében egyaránt tetten érhető; amikor tehát a kapcsolódó tudományterületek és fogalmak rendszerét tárgyaljuk, figyelembe kell vennünk a kulcsszavak kettős természetét.

Adatbányászat, szövegbányászat, információkivonatolás

Tekintettel arra, hogy a kulcsszavak a szövegből kerülnek kivonatolásra, érdemes összefoglalnunk az adatbányászat, szövegbányászat és az információkivonatolás közötti összefüggéseket és különbségeket. Az adatbányászat egy számítógépes rendszer által támogatott hatékony keresési szisztéma, amely a nagy adatállományokban található rejtett, de potenciálisan hasznos információkra irányul. Segítségével azonosíthatók a dokumentumban található minták és szerkezetek, valamint az információk kiértékelésének akadályai kiküszöbölhetők. Ennek érdekében figyelembe veszi a dokumentum különböző helyein található információk közötti kapcsolatokat és hasonlóságokat, támogatja a hipotézismentes keresést; tehát a strukturált adatok feltáró jellegű (exploratív) analízisét kínálja.^{vi} A klasszikus szövegbányászatban a szöveg a szavak egy halmazára (és saját dokumentumvektorára) korlátozódik, az információkivonatolás esetében viszont nyelvészeti eljárásokat kell bevonnunk a feldolgozás folyamatába. Jó példa erre a szófajok felismerésén alapuló, ún. „*Part of Speech*” taggelés és a szintaktikai elemzés, amely a mondat szerkezetek azonosítására szolgál.^{vii} Az adat- és szövegbányászat rendszerét az alábbi táblázat szemlélteti:

	Strukturált adatok	Strukturálatlan adatok
Hozzáférés (megszerzés)	Adatok lekérdezése <i>Adatbázisrendszerek</i>	Tartalom lekérdezése <i>Keresőgépek,</i> <i>dokumentummenedzsmet-rendszer</i>
Hozzáférés (értelmezés)	Adatok keresése <i>Adatbányászati eszközök</i>	Tartalom keresése <i>Szövegbányászati eszközök</i>

1. ábra: Az adat- és szövegbányászat rendszere^{viii}

Kulcsszavak kivonatolása

A kulcsszavak kivonatolása (függetlenül a technológiai környezettől és az automatizáltság fokától) javarészt statisztikai és nyelvészeti eljárásokon nyugszik. A statisztikai módszerek jellemzően a dokumentum strukturált reprezentációjára irányulnak,^{ix} illetve a szavak előfordulási gyakorisága és a hozzájuk rendelt pozitív vagy negatív súlyozás alapján állapítják meg a szignifikáns kifejezéseket.^x Az olyan statisztikai adatelemző módszerek, amelyek lehetővé teszik a szavak „*átlagos viselkedésének*” meghatározását, majd a hasonló tulajdonságokkal rendelkező szavakat csoportosítják, igen hatékonyak bizonyultak a feltáró munkában. A *természetes nyelvi feldolgozás (Natural Language Processing, NLG)* kifejezés az értelmezési lehetőségek széles skáláján helyezhető el; a kapcsolódó tanulmányok témakörei a beszédfelismeréstől az automatikus szövegenerálásig terjednek. Ezek közül néhány a hasznos információ kivonatolásával foglalkozik. A kutatások egy részében létező nyelvi adatokat használnak, vagy nyelvi információt kreálnak belőle.^{xi} A legeredményesebb és legtöbb szempontú feldolgozás érdekében azonban gyakran kombinálják a kétféle megközelítést. A kulcsszavak kivonatolása az indexeléstől a szöveges összefoglaló generálásáig számos területen felhasználható;^{xii} a kiválasztott modell alapján a következő típusok kivonatolása válik lehetségessé:

- entitások,
- attribútumok,
- tények,
- események.

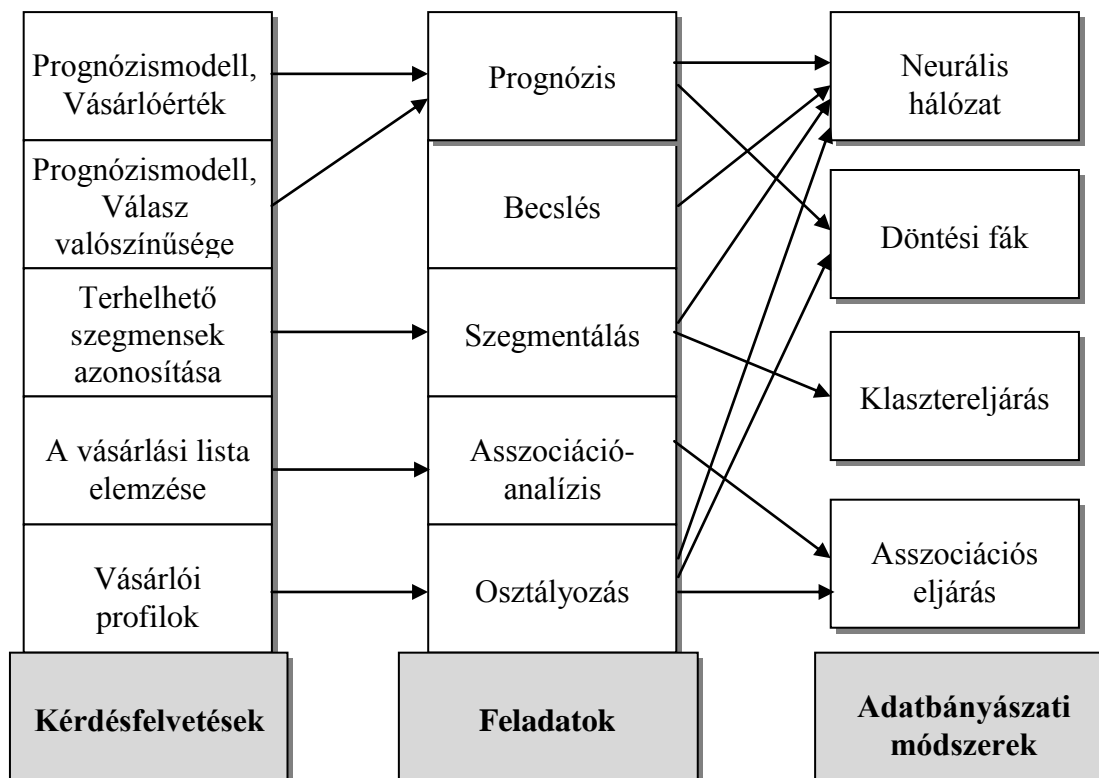
Ezek az objektumok többnyire rendelkeznek olyan névvel is, amelyet kivonatolni lehet, ezért a folyamatot *entitás-kivonatolásnak (Named Entity Extraction)* nevezik.^{xiii} Léteznek ugyanakkor a dokumentumnak, mint a kivonatolás forrásának olyan attribútumai, amelyek segítségével a dokumentum részletesen jellemezhető, így azokat figyelembe kell vennünk a megfelelő módszer kiválasztásánál, mégpedig:

- a dokumentumot hordozó médium,
- a dokumentum fizikai elérési helye,
- a dokumentum formátuma,
- a dokumentum nyelve,
- karakterkészlet, karakterkódolás, kódkészlet;
- statisztikai jellemzők,
- a dokumentum mérete,
- a dokumentum információtartalma,
- a dokumentum témája,
- a dokumentum stílusa, motívumai.^{xiv}

A kulcsszavak kiválasztásával kapcsolatos piaci megközelítés – főként az internetes források tekintetében – megkülönböztet úgynevezett *márkanév (branded)* és *nem márkánév (nonbranded)* kulcsszavakat. Előbbi csoportba sorolható a hirdető neve és webcíme ezek helytelenül írt változataival, utóbbihoz pedig azok a generikus kifejezések sorozhatók, amelyek a hirdető termékeit és szolgáltatásait írják le.

A kulcsszavak és az internet

Az internet rohamos fejlődés és a benne rejlő lehetőségek kiaknázásának szándéka a hatékony kulcsszavak megválasztásnak új dimenzióit nyitotta meg, ezért ebben a pontban kitérünk annak a webes környezetnek a jellemzésére, amelyben a kulcsszavas keresés megvalósul. A kulcsszavak jelentőségének felértékelődését ebben a milióben a *keresőmotorok* (*search engine*) által preferált kulcsszavas keresés elterjedése indukálta. Az egyes keresési típusok illeszkednek a korábban már tárgyalt adatbányászat céljaihoz és módszereihez, amelyet az alábbi ábra alapján tekinthetünk át.



2. ábra: Az adatbányászat kérdésfelvetései, feladatai és módszerei^{xv}

A célokat szolgáló keresési lehetőségek között a legfontosabbak a teljes szöveges, a kulcsszavas és a hipertext alapú keresés, illetve az adatbázisokban használt lekérdező nyelvek használata.^{xvi} Témánk szempontjából lényegesnek a teljes szöveges és a kulcsszavas keresés közötti különbség megvilágítása mutatkozik. Míg a teljes szöveges keresés a szavak szintjén valósul meg (lényegében a betűsorok puszta összevetésével), a kulcsszókeresés esetében a többlet abban áll, hogy a folyamat során nem szóalak, hanem *lemma* (itt: *szótári alak*) alapú keresést végzünk és figyelembe vesszük a szinonimákat.^{xvii} A kulcsszavak helyes megválasztása lehetőséget nyújt a kitértetett keresési mód támogatására strukturált és részben strukturált adatok esetében egyaránt, így:

- relációs adatok,
- XML-adatok,

- gráfokba rendezett adatok,
- adatfolyamok,
- munkafolyamatok között.^{xviii}

A keresés kontextusa: a szemantikus Web

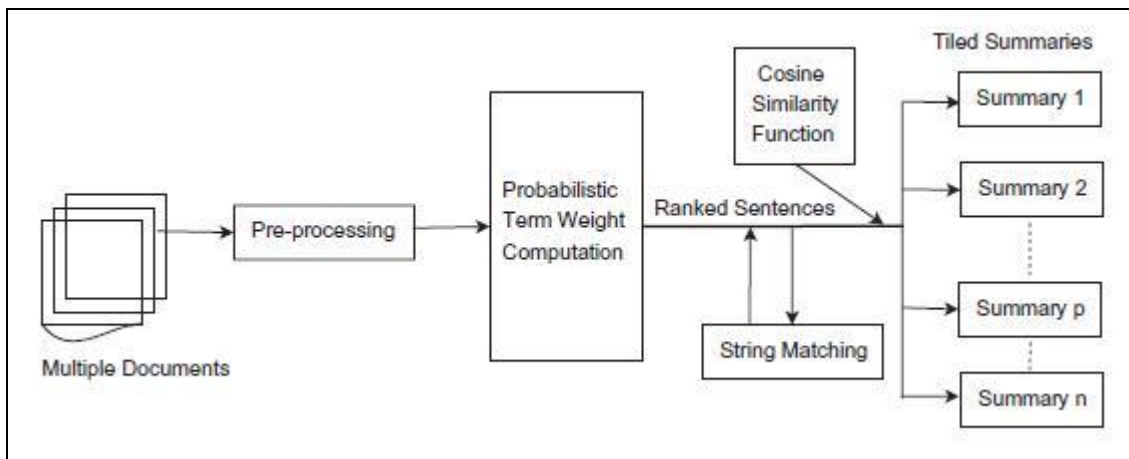
A felhasználói igényektől függően változatos technikákat használhatunk az információ visszakeresésére a szemantikus Weben, amelyek a mögöttes információs térrel kapcsolatos különböző nézőpontokon alapulnak. A *szemantikus Web* növelheti a hagyományos WWW-kulcsszavas keresés eredményességét, ugyanis lehetőségeire támaszkodva a keresőmotor bővítheti vagy szűkítheti a keresőkérdést azon fogalmak alapján, amelyek kifejezetten a felhasználó által keresett terminushoz kapcsolódnak. A szemantikus Webet a keresés kontextusaként használva meghatározhatjuk az adott kifejezés helyes értékelését és értelmezését. A kulcsszavak segítségével lehetőség nyílik a források közötti szemantikus asszociációk különböző típusainak definiálására is.^{xix} A következő fejezetben áttekintünk néhány olyan módszert, amelyek előremutatónak bizonyultak a kulcsszavak keresésében és helyes megválasztásában.

4. Kulcsszókeresési technikák és modellek

A K-keverék (K-mixture) modell

A kifejezések osztályozására remekül alkalmazható KATZ *K-keverék modellje*, amely egy módosított kifejezés-súlyozás alapján rangsorolja a mondatokat és a magasan rangsoroltakat választja ki a végső összefoglalóhoz. Tekintsük át a modellben rejlő lehetőségeket SARAVANAN- RAMAN-RAVINDRAN alapján!^{xx}

Ez a technika az ismétlődő mondatokat eltávolítja és egy csempézett összegzést készít. A modell elkerüli a terjengősséget és összefüggően olvasható, sőt, böngészhető összegzést készít, amit esemény-specifikus csempézett összegzésnek nevezünk. A probléma ezzel a gyakorlattal, hogy nehéz megfelelően, automatikusan előállítani kifejezéseket, és a rosszul beágyazott kifejezés teljesen leronthatja az összegzés minőségét. A fentebb említett problémákat elkerülendő, egy statisztikai megközelítést alkalmazunk, amely előrejelzi az összegzésbe illő mondatokat a betáplált dokumentumokból. A statisztikai rendszerek tapasztalati alapú, „*tanítható*” rendszerek, amelyek minimálisra csökkentik az emberi erőfeszítések szükségességét. Az összegzésre kiválasztott kifejezések eloszlása szorosan kapcsolódik a szöveg jelentéséhez. Ezért olyan *valószínűségelméleti modellt* készíthetünk, amely egyben egy módosított kifejezés-súlyozó séma is, és növeli az összegzés hatékonyságát. Az *vektortér modellel* elő-feldolgozott kifejezéseket a *kifejezés-eloszlási modellel* (*K-keverék modell*) dolgozzuk fel, amely azonosítja a rejtett szabályszerűségeket a kifejezések között, majd végül sorba állítja a mondatokat. A rendszer blokkdiagramja az a következő ábrán látható.



3. ábra: A rendszer blokkdiagramja^{xxi}

Elemi:

- dokumentumok (*Multiple Documents*),
- előfeldolgozás (*Pre-processing*),
- a kifejezések súlyozásának valószínűségi alapú kiszámítása (*Probabilistic Term Weight Computation*),
- sorba állított mondatok (*Ranked Sentences*),
- sorba illesztés (*String Matching*),
- koszinusz hasonlósági függvény (*Cosine Similarity Function*),
- csempézett összefoglalók (*Tiled Summaries*),
- összefoglaló 1-n (*Summary 1-n*).

A modellt kétféleképpen értékelték. Elsőként a modell által generált összegzést összehasonlították tíz, emberek által készített összegzéssel. Ezután az eredményeiket két, az interneten található automatikus összegzővel is összevetették, amelyek közül az egyik a szöveg nyelvi elemzésére épült, míg a másik statisztikai alapú megközelítésre. Az értékelést több különböző kivonati szinten is elvégezték (15%, 20%, 25%, 30%, 40%, 50% és 60%).

Valószínűségelméleti modellek, egyedi súlyozási technikák

A valószínűségi modellek (*probabilistic models*) relatív gyakorisági és feltételes valószínűségi értékek figyelembevételével végzik a dokumentumok felett végrehajtott információ-visszakeresési műveleteket. A valószínűségi modellek kidolgozásának indíttatása az a megfontolás volt, hogy sok esetben maguk az emberi szakértők sem tudják teljesen egyértelműen osztályokba sorolni a dokumentumokat. A különböző szakértők által megadott kulcsszavak gyakran ugyanannak a dokumentumnak az esetében is mások (bizonyára jelentős átfedések mellett). Az egyedi szavak súlyozása egy tanulási cikluson keresztül folyamatosan változtatható. Ez az úgynevezett relevancia-visszacsatolás módszere. Ekkor a felhasználtól kapott visszacsatolások alapján a szavak súlyozása fokozatosan javítható.^{xxii}

Az egyedi szavak súlyozása abból a megfontolásból ered, hogy az egyes lekérdező szavak esetében igyekszünk meghatározni, hogy melyiknek mekkora a relatív fontossága a

lekérdezésben. Ez a művelet a szövegbányászati előfeldolgozások közé sorolható, mert eredményét a későbbi műveletek felhasználják az információ-visszakeresés során. Ezen súlyok alkotják a *szó-dokumentum mátrix* (*term-document matrix, TD-mátrix*) elemeit is.^{xxiii}

A TF-IDF

A *TF-IDF* (*term frequency – inverse document frequency, vagyis: kifejezés-gyakoriság – inverz dokumentumgyakoriság*) egy ad- hoc súlyozási rendszer, mivel nem közvetlenül egy matematikai modelltől lett levezetve. Hasznos lehet azonban olyan esetekben, amikor vektorok hasonlóságának megközelítő mérése szükséges.

A kifejezés-eloszlási modellekkel valószínűségi alapú kifejezés-súlyozási sémát vezethetünk le a vektortér modellhez. Következő modellünk két matematikai modellen, a *Poisson- és a negatív binomiális eloszlásokon* alapul. JOHNSON és KOTZ következtetése szerint a negatív binomiális eloszlás gyakran helyettesítheti a Poisson-t, amikor kétséges, hogy a Poisson szigorú követelményei – elsősorban a függetlenség – teljesülnek-e. Saját tapasztalataink is megerősítették az említett eredményt, miszerint a negatív binomiális eloszlás jól illeszkedik a szavak eloszlására a dokumentumokban. Mivel a negatív binomiális eloszlással meglehetősen nehéz dolgozni, annak egy egyszerűsített változatát, az úgynevezett K-keverék modellt felhasználását mutatjuk be. Az eredmények alapján a K-keverék jobb közelítést ad, mint a Poisson-modell.

Miután a súlyokat a K-keverék képlettel megkaptuk, a leginkább releváns mondatok a következő folyamattal választhatók ki a dokumentumokból:

1. A kifejezések normalizálása a képlet *et* paramétere alapján.
2. A mondat súlyozásának kiszámítása a kifejezések valószínűségi értékeinek összeadásával.
3. A mondatok rangsorba állítása súlyozásuk alapján.
4. A mondatok csökkenő sorrendbe állítása.

A folyamat során minden mondatához egy értéket rendelünk, ami az adott mondat összefoglalásban való szereplésének alkalmasságát jelzi. A módszerhez egy *sorba illesztési* (*String Matching*) algoritmus is tartozik, melynek lépései a következők:

1. Input adatok: a kifejezés-eloszlási modellel sorba állított mondatok.
2. Használjuk az előfeldolgozási szakaszban megállapított szótöveket!
3. Ha a két összehasonlított mondatban szereplő szótövek min. 75%-ka megegyezik, töröljük az egyik mondatot, és a legmagasabban rangsorolt következőt léptessük a helyébe!
4. Folytassuk 2. és 3. lépést, amíg az összes mondatot fel nem dolgozzuk!
5. Az eredményeket közvetítsük a szöveg-csempézési folyamat felé!^{xxiv}

Random kivetítési mátrix használata (a WEBSOM- és a liGHtSOM-modell)

A módszer eredményesen használható nagy terjedelmű szövegek archiválására, illetve a szövegegységek kulcsszavainak kivonatolására; a fontosabb jellemzők és lépések a fejezet címében jelölt kétféle modell összevetésével kielégítően feltárhatók.

Az terjedelmes szövegek archiválására használt WEBSOM-módszertan a szövegegységek címkéinek/kulcsszavainak kivonatolása esetén nagyon lassú. Ennek az oka, hogy a modell az egységgel kapcsolatos összes dokumentum minden egyes szavához relatív gyakoriságot számol, majd ezeket összehasonlítja a térképen lévő többi egység minden szavának relatív frekvenciájával. Mivel a térképek 100.000 egységnél is többet, az archívum pedig akár 7 millió dokumentumot is tartalmazhat, a WEBSOM-modell nem praktikus. A liGHtSOM-modell azonban a *térkép* („*trained map*”) súlyvektorainál a súlyok eloszlásán, valamint a beviteli adatok tömörítésére használt random kivetítési mátrix egyszerű módosításán alapul.

Kulcsszó/jelentős címke: a WEBSOM-ban w kifejezés kulcsszó egy C -dokumentumcsoportban, ha: 1) w prominens C -ben más C -ben található kifejezésekhez képest és 2) w prominens C -ben w -nek a többi előfordulásához képest a teljes gyűjteményben. Azok a kifejezések, amelyek magas súlyértékkel rendelkeznek, fontosabbak, mint amelyek alacsonyabbal. Más szóval ezek potenciális kulcsszavai az adott térképegységhez tartozó dokumentumoknak. Azonban, mivel random kivetítési mátrixot használtunk, minden súlykomponenshez számos kifejezés tartozik. Nincs tehát olyan egyértelmű módszer, amivel megállapítható, hogy mely kulcsszavak felelősek az adott egység magas súlyozásáért. Viszont a random kivetítési módszer alapján visszakövethetjük, hogy melyek azok a változó szókombinációk, amelyek a tömörített beviteli vektor minden rétegéhez hozzájárulnak. A kombinációk alapján levezethetjük a valóban fontos kulcsszavak csoportját. A liGHtSOM-módszer folyamata:

- Minden dimenzióra kiszámoljuk a *súlyok átlagát* (μ) és *sztenderd deviációját* (σ). Az átlagos $\mu + z\sigma$ sztenderd deviációt meghaladó súlyok jelentősek az adott dimenzióban.
- Minden esetben, ha egy bizonyos dimenzió jelentősen magas, valószínűsíthető, hogy csak a kifejezések egyike járult hozzá jelentősen a magas súlyhoz, míg a többi csak *járáulékos* („*piggy-back*”) kifejezés.
- Mivel a random kivetítési módszer véletlenszerűen helyezi el a kulcsszavakat az r számú dimenzióban, az igazán jelentős kulcsszavak konzisztensen hozzájárulnak az r dimenzió magas súlyaihoz. Ha minden kifejezésre megszámláljuk, hogy az r dimenzióból mennyiben magas a súlyérték, az igazán jelentős kulcsszavaknál az eredmény közel lesz r -hez.
- Kiválasztjuk azokat a kifejezéseket, amelyek több mint r^0 dimenzióban magas súlyozásúak voltak (a kutatás szerint $r^0=0.6r$).
- Végül súlyaik összege szerinti sorrendbe állítjuk a kiválasztott kifejezéseket.^{xxv}

A vektortér modell

A vektortér modell a szövegbányászat klasszikus reprezentációs eszköze. A *korpusz* (vagy dokumentumállomány) minden dokumentuma egy pontnak felel meg a sokdimenziós térben, maga a korpusz tehát pontok sokaságaként jelenik meg. Ebben az esetben egy dokumentum az ortogonális egyedi szavak lineáris kombinációjának tekinthető. A modell egyik központi eleme a *szó-dokumentum* (*Term-Document, TD*) mátrix, amely a vektortér

leírásának és reprezentációjának eszköze; egy-egy dokumentumpont koordinátáját a benne szereplő egyedi szavak határozzák meg. Így a TD-mátrix sorai *egyediszó-vektorok*, míg oszlopai az ún. *dokumentumvektorok*. Megkülönböztethetjük a TD-mátrix három jellemző típusát is:

1. *Bináris TD-mátrix* esetében a TD-mátrix celláiban 0 vagy 1 szerepel. Ha a $TD_{i,j}$ cella értéke 1, akkor az *i*-edik szó a *j*-edik dokumentumban benne van, egyébként 0 (ha nincs benne abban a dokumentumban).
2. A *szógyakorisági TD-mátrixnál* egy cellában azon érték szerepel ($TD_{i,j}$), amely megadja, hogy az *i*-edik egyedi szó a *j*-edik dokumentumban hányszor fordul elő.
3. A *súlyozott TD-mátrix* használatakor a fenti gyakorisági értékeket még egy súlyozáson vezetik keresztül. Ennek köszönhetően a kevésbé releváns (pl. stopszó közeli) szavak kisebb súllyal kerülnek bele a mátrixba, ezáltal hatásuk a vektortérben kevésbé lesz érzékelhető.

A klasszikus vektortér modell hátránya, hogy alkalmazásával elveszhetnek a szavak egymáshoz képesti pozíciójából származó információk, ugyanis a TD-mátrix csak az egyedi szavak gyakoriságát tárolja, helyzetüket nem. Így nehézkessé válhat a formailag azonos alakú, de a kontextustól függően más jelentésű szavak kezelése.^{xxvi}

5. A kulcsszavak és a kivonat kapcsolata

Ahogy az előzőekben bemutatásra került a kulcsszavak kivonatolásának elmélete és gyakorlata statisztikai és nyelvészeti eljárásokon nyugszik. A következőkben nézzük meg a humán kivonatolási technika feltárására során nyert tapasztalatokat, melynek fontos eleme volt a kitöltők kulcsszó megjelölési illetve a kiválasztott releváns mondatok közti kapcsolat feltárása. A vizsgálat végső célja a szignifikáns mondatkiválasztásban rejlő szabályszerűségek beazonosítása, és ezek beépítési lehetőségeit a kivonatoló programba.

A vizsgálat során a kitöltőket az alábbi két cikk elolvasására, és kulcsszavainak megjelölésére kértem:

- Forgó Sándor: Az új média és az elektronikus tanulás^{xxvii}
- Komenczi Bertalan: A digitális pedagógus – elméleti megközelítések, fogalom meghatározások^{xxviii}

A kulcsszó megjelölésen kívül a kitöltőknek el kellett készíteni a cikkek kivonatát is megjelölve a 10 leggyakoribb mondatot a cikkben. Elemzésem során megvizsgáltam, hogy a megjelölt kulcsszavak a kivonatba választott mondatoknak részét képezik-e. Ennek eredményét meglepő volt, ugyanis, Forgó Sándor cikke esetén míg a kulcsszavak megjelölésében 10-ből 7 kulcsszót azonosat jelöltek meg a mintacsoportok, addig a 10 leggyakoribb mondatban, mely a kivonathoz megjelölésre került, csak a kitöltők 5%-nál található meg valamely kulcsszó a referátumnak megjelölt mondatok között.

Komenczi Bertalan cikke esetén is hasonlóan alacsony az egyezés szintje, ugyan itt a 14 kulcsszó megjelenik a kivonatba megjelölt mondatok között, de elenyésző módon, mivel csak a kulcsszavak 12%-a található meg a kivonatba megjelölt leggyakoribb 10 mondat szavai közt.

6. Összefoglalva

A kulcsszavak és releváns mondatok megjelölése közti összefüggés sokkal kisebb egyezést mutat, mint azt a vizsgálat elején feltételeztük. Összefoglalva elmondható, hogy a humán gondolkodás vizsgálata során, míg a kulcsszó kiválasztásokban szignifikáns egyezés mutatható ki különböző mintacsoportok között, addig a kitöltők által a cikkek kivonatának elkészítéséhez kiválasztott mondatok, nem azok, melyek tartalmazzak az ugyanazon személyek által kiválasztott kulcsszavakat.

Irodalomjegyzék

- i Saravanan, M. – Raman, S. – Ravindran, B.: A probabilistic approach to multi-document summarization for generating a tiled summary. In: International Journal of Computational Intelligence & Applications 2(2006), pp. 231-243., p. 231.
- ii Ungváry Rudolf – Vajda Erik: Az információkeresés szavai [elektronikus dokumentum]. 2006.
In: URL http://tmt.omikk.bme.hu/show_news.html?id=3451&issue_id=446 (letöltés: 2014. 06. 25.)
- iii Csík Tibor – Varga Katalin: A tudás és az információfeldolgozás [elektronikus dokumentum]. In: URL http://tmt.omikk.bme.hu/show_news.html?id=4007&issue_id=464 (letöltés: 2014. 06. 25.)
- iv Wordsimth tools. Definition von Gewichtung [elektronikus dokumentum]. In: URL http://www.lexically.net/downloads/version4/html/deutsch/index.html?keyness_definition.htm (letöltés: 2014. 06. 10.)
Ford: Mizera Tamás
- v Faulhaber, Patricia: Working words. SEO style for search-engine optimization. In: Public Relations Tactics 2(2009), p. 19.
- vi Möhrle, Martin G. – Walter, Lothar (Hrsg.): Patentierung von Geschäftsprozessen. Monitoring, Strategien, Schutz. Berlin-Heidelberg, 2009. XV, p. 92.
- vii Pribe, Torsten – Kolter, Jan – Kiss, Christine: Semiautomatische Annotation von Textdokumenten mit semantischen Metadaten. In: Wirtschaftsinformatik 2005. Economy, egovernment, esociety. Heidelberg, 2005. pp. 1309-1328., p. 1319.
- viii Möhrle, Martin G. – Walter, Lothar (Hrsg.): Patentierung von Geschäftsprozessen. Monitoring, Strategien, Schutz. Berlin-Heidelberg, 2009. XV, p. 92.
- ix Pribe, Torsten – Kolter, Jan – Kiss, Christine: Semiautomatische Annotation von Textdokumenten mit semantischen Metadaten. In: Wirtschaftsinformatik 2005. Economy, egovernment, esociety. Heidelberg, 2005. pp. 1309-1328., p. 1319.

- x Wordsimth tools. Definition von Gewichtung [elektronikus dokumentum]. In: URL http://www.lexically.net/downloads/version4/html/deutsch/index.html?keyness_definition.htm (letöltés: 2014. 06. 10.)
Ford: Mizera Tamás
- xi Rossignol, Mathias – S'ebillot, Pascale: Combining statistical data analysis techniques to extract topical keyword classes from corpora. In: Intelligent Data Analysis 9 (2005), pp. 105-127., p. 105.
- xii Nohr, Holger: Grundlagen der automatischen Indexierung. Ein Lehrbuch. Berlin, 2003. p. 127.
Ford: Mizera Tamás
- xiii Pribe, Torsten – Kolter, Jan – Kiss, Christine: Semiautomatische Annotation von Textdokumenten mit semantischen Metadaten. In: Wirtschaftsinformatik 2005. Economy, egovernment, esociety. Heidelberg, 2005. pp. 1309-1328., p. 1319.
Ford: Mizera Tamás
- xiv Vázsonyi Miklós: Szöveges dokumentumok attribútumai [elektronikus dokumentum]. Copyright 2006. In: URL <http://www.vazsonyi.hu/szovegbanyaszat/4.html> (letöltés: 2011. 10. 30.)
- xv Abraham, Kiryo: Business Intelligence. Aufgaben, Prozess und Architektur, Verlag: GRIN, 2008 p. 39.
- xvi Ui. p. 42-43.
- xvii Gaiser, Brigit – Hampel, Thorsten – Panke, Stefanie (Hrsg.): Good tags, bad tags. Social tagging in der Wissensorganisation. Münster, 2008., p. 221.
- xviii Chen, Yi – Wang, Wei – Liu, Ziyang – Lin, Xuemin: Keyword Search on Structured and Semi-structured Data. In: 28th ACM SIGMOD International Conference on Management of Data, 2009., p. 1005.
- xix Mukherjea, Sougata: Information retrieval and knowledge discovery utilising a biomedical Semantic Web. In: Brief Bioinform 3(2005), p. 259. Lásd még jelen értekezés 2.4.1 Szemantikus asszociációk meghatározása című pontját.
- xx Saravanan, M. – Raman, S. – Ravindran, B.: A probabilistic approach to multi-document summarization for generating a tiled summary. In: International Journal of Computational Intelligence & Applications 2(2006), pp. 231-233.
- xxi Forrás: ui., p. 233.
- xxii Vázsonyi Miklós: Valószínűségi modellek [elektronikus dokumentum]. Copyright 2006. In: URL <http://www.vazsonyi.hu/szovegbanyaszat/15.html> (letöltés: 2011. 11. 05.)
- xxiii Vázsonyi Miklós: Egyedi szó súlyozási technikák [elektronikus dokumentum]. [Copyright 2006]. In: URL <http://www.vazsonyi.hu/szovegbanyaszat/12.html> (letöltés: 2011. 11. 02.)

- xxiv Saravanan, M. – Raman, S. – Ravindran, B.: A probabilistic approach to multi-document summarization for generating a tiled summary. In: International Journal of Computational Intelligence & Applications 2(2006), pp. 234-235.
- xxv Azcarraga, Arnulfo P. – Yap Jr., Teddy – Chua, Tat Seng: Comparing Keyword Extraction Techniques for Websom Text Archives. In: International Journal on Artificial Intelligence Tools 2(2002), pp. 219-221.
- xxvi Vázsonyi Miklós: A vektortér modell [elektronikus dokumentum]. Copyright 2006. In: URL <http://www.vazsonyi.hu/szovegbanyaszat/13.html> (letöltés: 2011. 11. 01.)
- xxvii Forgó Sándor: Az új média és az elektronikus tanulás. – In: Új pedagógiai szemle, 2009. (59. évf.) 8-9. sz. 91-96. old.
- xxviii Komenczi Bertalan: A digitális pedagógus - elméleti megközelítések, fogalommeghatározások, In: Lévai Dóra, Tóth-Mózer Szilvia, Szekszárdi Júlia (szerk.) Digitalis_de_generacio 2.0. Budapest: Underground Kiadó és Terjesztő KFT, 2013. pp. 193-202

Kompetenciák az elektronikus tanulási környezetekben a humán teljesítménytámogató technológiai kutatások szemszögéből

Competences in Electronic Learning environments with regards to studies on Human Performance Technology

Racsko Réka

“Eszterházy Károly Főiskola. Médiainformatika Intézet. Humáninformatika Tanszék

racsko@ekt.f.hu

Absztrakt: A humán teljesítménytámogató technológia (HPT) a minőségbiztosítás egy lényeges eleme. A HPT célja a mérhető és költséghatékony teljesítmény mérése. Alapja az a feltételezés, hogy az emberi teljesítményt számos alapelv határozza meg, mint a pszichológia, a rendszerelmélet, a mérnöki és üzleti menedzsment (Chyung, 2008). Módszertani eszközei a döntéshozók tájékoztatása számára az empirikus módszerek, például a megfigyelések és kísérletek. Az oktatás területén az egyik legadekvátabb terület erre a pedagógusi munka minőségi értékelése, (Mourshed–Chijioke–Barber, 2012) és a támogató rendszerek (Simmons–Hawkins, 2009; Hargraves 2012; Leask, 2013;) szerepe. Vizsgálatom során kitérek a teljesítményfejlesztés nemzetközi szervezete (International Society for Performance Improvement (ISPI) által kodifikált szabványokra, amelyek a minőségi teljesítménytámogatás a gyakorlatban való alkalmazhatóságát vizsgálják. Ennek során olyan fókuszpontokat határoznak meg, amelyek a későbbi pedagógiai kultúra fejlesztésében és a kompetenciafejlesztésben is fontos szerepet töltenek be. A humántőke kapcsán ismertetném a digitális kompetencia keretrendszerét (DigComp 2013 alapján). Ezen komponensek köré szerveződik az ISTE (International Society for Technology in Education), illetve a hazai kutatók (Lévai–Ollé és mtsai., 2013) által is definiált digitális állampolgárság (digital citizenship) (Mossberger– Tolbert–McNeal 2008; Ohler, 2010; Ribble 2009, 2011) kompetenciarendszere. Ezen túlmenően bemutatom a digitális állampolgárokkal szemben támasztott hazai (Nemzeti Infokommunikációs Stratégia 2014-2020; NAT) és Európai Unió követelményeket (EU 2020) és említést teszek a globális állampolgárság (Zajda et. al, 2009) (global citizenship) fogalmáról is.

Kulcsszavak: humán teljesítménytámogató technológia, digitális állampolgár, kompetencia

Abstract: Human Performance Supporting Technology (HPT) is an important part of quality control. HPT was first applied in the education sector to measure measurable and cost effective performance. It is based on the presumption that human performance depends on many principles, including psychology, system theory, as well as engineering and business management. Its methodology consists of empirical methods – like surveillances and tests – for informing decision makers. In the sector of education one of the most adequate area is the quality measurement of teachers' work, (Mourshed–Chijioke–Barber, 2012) and the role of supporting systems (Simmons–Hawkins, 2009; Hargraves 2012; Leask, 2013;). During my examination I cover standard codified by International Society for Performance Improvement (ISPI), which deal with the practical applicability in quality performance support. During these several focal points were defined which play an important role in the latter development of teaching culture and development of competences. With regards to human capital I introduce the frame system of digital competence (based on DigComp 2013). ISTE (International Society for Technology in Education) is built around these components, and the digital citizenship defined by Hungarian researchers (Lévai–Ollé and colleagues, 2013) as well as the system of competences (Mossberger– Tolbert–McNeal 2008; Ohler, 2010; Ribble 2009, 2011). Furthermore I introduce requirements for digital citizens (National Information and Communication Strategy 2014-2020; NAT) and the requirements of EU (EU 2020), and I mention the concept of global citizenship (Zajda et. al, 2009).

Keywords: human performance supporting technology, performance supporting technology, digital citizenship, competence

1. Bevezetés

Napjaink oktatásában egyre erősödő szerepe van a meritokráciának, annak a társadalmi modell, amelyben az egyén pozícióját a teljesítményétől teszik függővé. Tanulmányomban a teljesítmény témakörét járom körbe a humán teljesítménytámogató technológia (HPT) rendszerében. Munkám során ismertetem HPT fogalmi kereteit, illetve nemzetközi sztenderdjét, majd az oktatás oldaláról mutatom be a humán teljesítménytámogatás megjelenését a nemzetközi és a hazai oktatáspolitikai trendekben. Végül a tanulók oldaláról is bemutatom azokat az elvárásokat és elvárt kompetenciákat, amelyekkel a digitális korban rendelkeznie kell. Munkám célja a humán teljesítménytámogató technológia sokszempontú bemutatása, és a humántőke szerepének több oldalról történő megközelítése.

2. A Humán Teljesítménytámogató Technológia (HPT) fogalmi keretei

A humán teljesítménytámogató technológia fogalma az oktatástechnológia területén bontakozott ki az '50-es-'60-as években. A '70-es években a gyakorlati alkalmazás révén terjedt el széles körben¹. Később a humán teljesítménytámogatás és az oktatórendszerek tervezésének (Instructional Systems Design -ISD) területe kettévált.

Több fogalmat is találunk a szakirodalomban a humán teljesítménytámogató technológia (továbbiakban HPT) definiálására, ezek közül jelen tanulmányban kettőt ismertetnék. Az első, Pershing² definíciója, amely alapján a HPT, az egyének és a szervezet teljesítmények javítása érdekében tett lépések összessége. A másik fogalom, a legnagyobb nemzetközi szervezet, az International Society for Performance Improvement (ISPI) által definiált fogalom, mely szerint a HPT alatt egy olyan komplex megközelítést értünk, amely segít fejleszteni a hatékonyságot, a termelékenységet, és bizonyos kompetenciákat, speciális módszerek és eljárások révén. Ezen túlmenően olyan probléma megoldási stratégiákat kínál, amelyek növelhetik az egyének teljesítményét. Konkrétan, olyan komplex folyamatot értünk alatta, melynek elemei a kiválasztás, az elemzés, a tervezés/fejlesztés, a végrehajtás és az értékelés, annak céljából, hogy az alkalmazott programok minél költséghatékonyabb módon befolyásolják az emberi viselkedést a teljesítmény növelése érdekében.

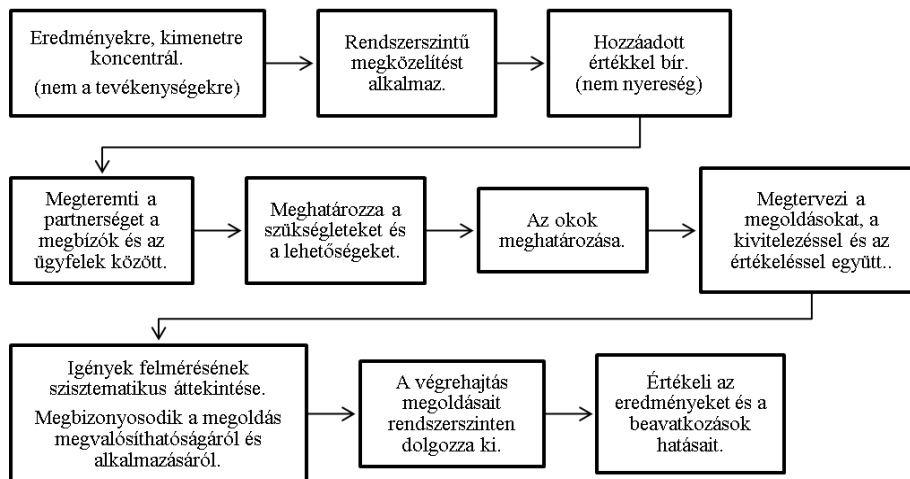
A rendszer három alapvető folyamat kombinációját foglalja magában: a teljesítményelemzést, ennek okainak feltárását, valamint a beavatkozás folyamatának kiválasztását, az egyén, a csoport és a szervezet szintjén.³ Több helyen Human Performance Improvement, azaz humán teljesítmény fejlesztés néven ismertes, és számos más teljesítménytámogató rendszerhez hasonlatos, azonban ezeknél komplexebb. Ennek fő fókusza a teljesítmény javítása a társadalom, a szervezet és az egyén szintjén.

A HPT alapja az a feltételezés, miszerint az emberi teljesítmény több oldalról is mérhető. A mérés empirikus módszerekkel, megfigyelések és kísérletek által történik, melynek célja a döntéshozók tájékoztatása. Ezen mérések eredménye a célzott és eredményorientált, költséghatékony változások bevezetése a teljesítményjavítás érdekében reaktív és proaktív módon egyaránt.⁴ A módszer három fő komponensből épül fel, az ember, a teljesítmény és a technológia egységéből.

Ember (Human)	szervezetet alkotó egyénekből és csoportjaikból áll
Teljesítmény (Performance)	tevékenységek és mérhető eredmények
Technológia (Technology)	a gyakorlati problémák megoldására alkalmazott rendszeres és szisztematikus megoldások

1. ábra A Humán Teljesítménytámogató Technológia három összetevője

A HPT több tudományterületet használ módszereiben, például a viselkedépszichológiát, a pedagógiai/oktatási rendszertervezést, a szervezetfejlesztést, valamint az emberi erőforrás-menedzsmentet. A módszer lényege, hogy a jelenlegi teljesítmény elemzése során beazonosítják azokat a teljesítményben jelentkező szakadékokat, hiányosságokat, amelyeket később a változásmenedzsment és a teljesítménynövelés egyéb módszerei során fejleszt, majd ennek eredményét értékeli. A folyamat a teljesítményfejlesztési stratégiára épül. Általános félreértés, hogy a technológia alatt az infokommunikációs eszközöket értik. Mint a fenti (1.) ábra is mutatja, a technológia ebben az értelemben tudományos ismeretek, tudáselemek gyakorlati alkalmazását jelentik például az iparban. A tudományos ismertek elsősorban műszaki irányultságú, a mérnöki és alkalmazott tudományok területéről származik. A humán teljesítménytámogató technológia rendszere tíz alapelveen keresztül mutatja be működésének legfontosabb elemeit.



2. ábra A HPT tíz alapelve (ISPI, 2012) (saját fordítás)⁵

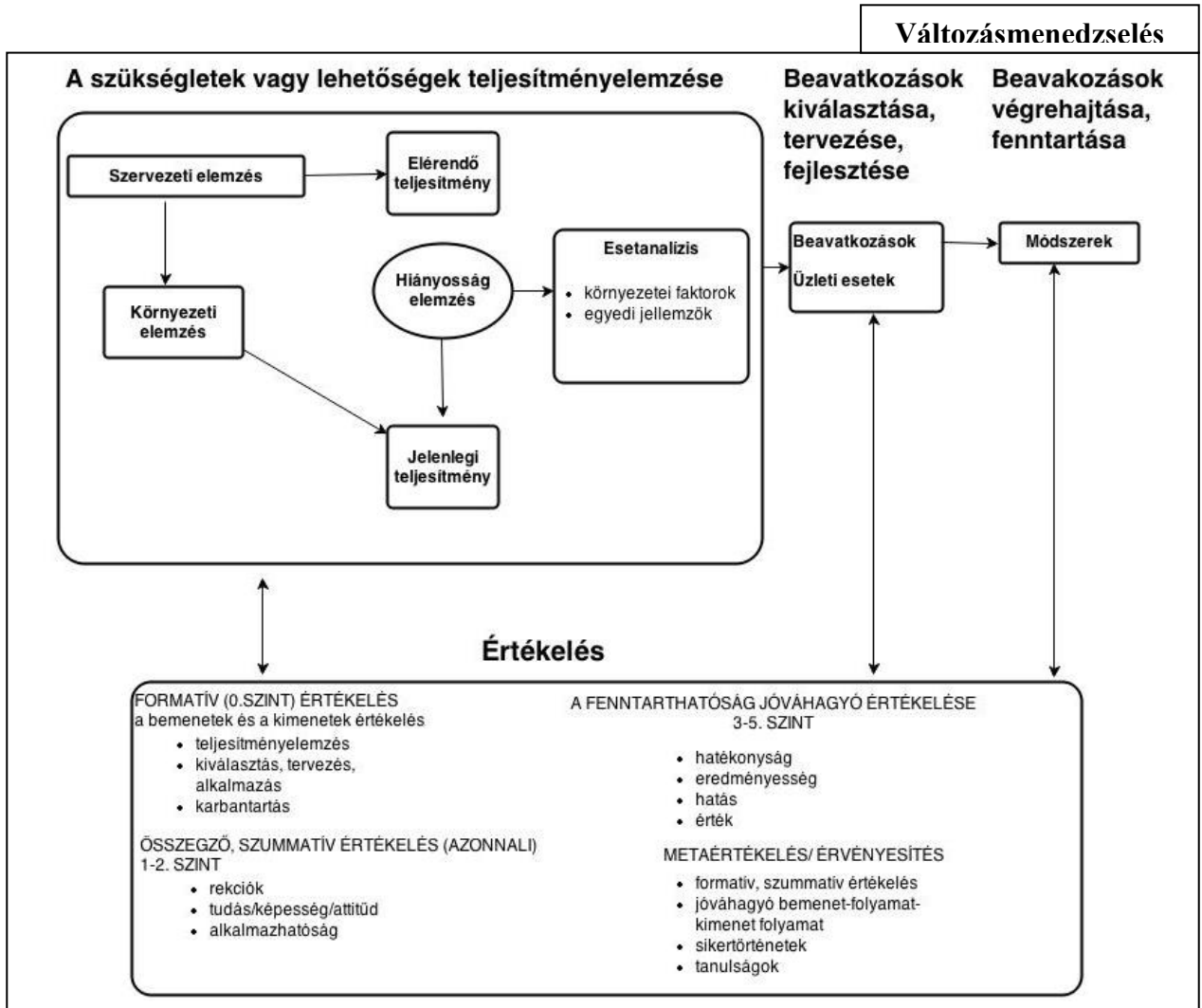
A fogalommal az 1990-es évektől foglalkozik behatóbban az akadémiai szféra, és számos intézmény képzési programjába is bekerült, diplomás és posztgraduális képzések formájában. A '90-es évek végén azonban⁶ egyre nagyobb szakadék keletkezett a humán teljesítménytámogató technológia gyakorlati alkalmazása és az akadémiai/kutatói szféra munkái között, ennek orvoslására számos, ezzel kapcsolatos kutatás vette kezdetét. Jól példázza a fogalmak tisztázatlanságát, hogy a teljesítményfejlesztés nemzetközi szervezete csak többszöri névváltoztatás után, 1995-ben vált véglegessé, International Society for Performance Improvement (ISPI), azaz a Teljesítménytámogatás Nemzetközi Szervezete néven,⁷ amely a mai napig támogatja tevékenységével a fejlesztést.

A következő fejezetben áttekintjük a humán teljesítménytámogató technológia modelljeit, sztenderdjeit, amelyez az ISPI szervezet deklarált a 2012-es évben.

3. A teljesítményfejlesztés nemzetközi sztenderjei (International Society for Performance Improvement (ISPI))

A teljesítményfejlesztés modellálása egy fontos lépés volt a rendszer alkalmazásában és más tudományterületeken való adaptálásában.

A humán teljesítménytámogatási rendszer legismertebb modellje a ISPI szervezet által kidolgozott modell, amely egy folyamatábrán keresztül mutatja, be a teljesítménytámogatás legfontosabb lépéseit.



3. ábra A HPT-modell: a teljesítményfejlesztés modellje (saját fordítás)⁸

A HPT folyamata a jelenlegi és az elvárt állapot összehasonlításával kezdődik az egyén és szervezet szintjén, amely során azonosíthatóvá válnak a teljesítménykülönbségek. Az esetanalízis során sor kerül az okok elemzésére, melynek célja, hogy meghatározni, hogy a teljesítményre milyen hatást gyakorol a munkakörnyezet (információk, erőforrások, ösztönzők) és az egyéni tényezők (motívumok, egyéni készségek képességek és kompetenciák).

A teljesítménykülönbségek és az okok azonosítását követően a megfelelő beavatkozások tervezése és fejlesztés következik. Ezek közé tartozik a mérési és visszacsatolási rendszer

kialakítása, például a kompenzációs és jutalmazási rendszerek, valamint a képzés és fejlesztés kereteinek, hatókörének megadása. Ezt követően a beavatkozások végrehajtásra kerülnek, és a változásmenedzselés elkezdődik. A folyamat minden fázisa után értékelés zajlik. Kezdetben formatív értékelés történik (a teljesítményelemzés, az okok elemzése, a beavatkozás kiválasztása és a változás folyamata során). A második szinten zajló szummatív értékelés az alkalmazottak közvetlen visszajelzéseire fókuszál. A fenntarthatóságot mérő értékelés a hatékonyságot és az eredményességet monitorozza. A végső értékelés középpontjában az üzleti eredmények javítása áll, úgy, mint minőség, termelékenység, értékesítés, ügyfelek megtartása, nyereségesség, piaci részesedés), illetve a beavatkozások megtérülése. A végső értékelés tulajdonképpen egy értékelés az értékelésről (meta értékelés), amelyben a korábbi értékelések tapasztalatait is magában foglalja.

A rendszer egy dinamikus, körkörös folyamatként fogható fel, és fenntarthatósága miatt folyamatos tevékenységet igényel.

Bár a humán teljesítménytámogató technológia elsősorban az ipar és a termelési szektor területén terjedt el, a következő fejezetben jól láthatjuk, hogy az oktatás, és elsősorban az oktatási rendszerek és az egyén szintjén megjelenő hatékonyságvizsgálatokban megjelennek elemei.

4. A humántőke értékelése az oktatásban az oktatók oldaláról

A pedagógusi munka értékelése és a támogató rendszerek szerepe gyakori és időről időre visszatérő téma napjaink oktatáspolitikájában, elsősorban a tanulói teljesítménymérések elemzése során.

Sági Matild és mtsai.⁹ több kutatást végeztek a tanári munka értékelése és az iskolai eredményesség témakörben. A kilencvenes évek kutatásai már kimutatták, hogy diákok teljesítményét a családi háttér mellett az iskolai hatások közül leginkább a pedagógus munkájának minősége befolyásolja.¹⁰

Azonban, azt hogy milyen ismertetőjegyekkel rendelkezik a jó tanár, már nem egyszerű definiálni, hiszen a kutatások során arra jutottak, hogy nincs egyetlen olyan indikátor jellegű mutató sem, amely alapján a pedagógus teljesítménye egyértelműen, egyszerű eszközökkel értékelhető lenne.¹¹ Linda Darling-Hammond összefoglaló jelentésében megfogalmazta, hogy önmagában sem a pedagógus általános képességei és intelligenciája, sem a tantárgyi tudása, sem képzettségi-végzettségi szintje (diploma jellege és minősége), sem pedig a tanítási gyakorlatának –hossza – tehát egyetlen, egyszerűen mérhető és számon tartható tényező – sem mutatott jelentős kapcsolatot az általa tanított tanulók iskolai teljesítményével, vagyis a pedagógus munkájának eredményességével.¹²

A pedagógusok munkájának eredményességét tehát nem lehet egyszerű mérőszámokkal, indikátorokkal mérni, ebben különbözik az ipari szféra termelékenységi mérőszámaitól, ehhez komplexebb módszerek szükségesek, amelyet például a pedagógus életpálya-modell kínál. Azonban a humán teljesítménytámogató technológia rendszerében az értékelés során tetten érhetőek szakirodalomban is idézett, a pedagógusok munkáját minősítő értékelési típusok (formális, szummatív értékelés), sőt ritkább esetben a meta értékeléssel is találkozunk. Hasonlóan megjelenik a jutalmazás és a szankcionálás módszere is. A humán teljesítménytámogató technológia tehát kínál olyan módszereket, amelyek akár a pedagógusi munka értékelési-, visszacsatolási-, jutalmazási- és szankcionálási rendszerének hatékony rendszerét alakíthatja ki.

A következőkben a humán teljesítménytámogató technológia két lehetséges eszközét mutatjuk be, elsőként egy nemzetközi mérés eredményeit ismertetjük, majd a hazánkban, a tavalyi évben bevezetett pedagógus életpálya-modell kerül bemutatásra.

4.1 Nemzetközi tanárvizsgálat -TALIS (Teaching and Learning International Survey)

Az OECD – a fejlett országok gazdasági és társadalmi fejlődést támogató kormányközi együttműködési szervezete 2008-ban hívta életre TALIS (Teaching and Learning International Survey) néven oktatásügyi programját a tanári pályán dolgozó pedagógusok felmérésére. A nemzetközi tanárvizsgálatot az OECD Nemzetközi Indikátor Munkacsoportjának (INES) az oktatás körülményeivel foglalkozó szakértői hálózata kezdeményezte azzal a céllal, hogy a részt vevő országokban összehasonlítható mutatókkal megállapítsa a pedagógiai munka értékelésének jellemzőit, a tanárok továbbképzésének, szakmai fejlődésének jellemzőit, a tanároknak az iskolával, mint munkahellyel, illetve a tanítással kapcsolatos attitűdjeit, nézeteit és az iskolavezetés jellemző sajátosságait. (TALIS, 2009. 4.)

Az OECD TALIS-vizsgálat jelentőségét kiemeli, hogy ez az első olyan, kormányok által kezdeményezett nemzetközi vizsgálat, amelyik a tanárokat kérdezi meg a tanítás és a tanulás körülményeiről, munkájukról és az iskolához, mint munkahelyhez való viszonyukról.

A TALIS¹ (Teaching and Learning International Survey)¹³ tehát egy olyan nemzetközi vizsgálat, amely a pedagógusokról és a pedagógusoktól gyűjt információt. A felmérés azt mutatja be, hogy a tanárok munkakörülményei és a tanulási környezet, amelyben dolgoznak milyen jellemvonásokkal bír, illetve a felmérésben során arról kérdezték a pedagógusokat, milyen körülmények között dolgoznak, hogyan ismerik el munkájukat, vagy milyen szakmai fejlődési lehetőségeik vannak, és milyenek az iskolával és a tanítással kapcsolatos nézeteik. A vizsgálat célja, hogy érvényes, időszerű és összehasonlítható eredményekkel szolgáljon tanári szakmáról, és ez által fejleszthetővé tegye annak stratégiáját. Ezen túlmenően célja, hogy jobban meg lehessen érteni a tanári szakmát és definiálni kompetenciát, valamint, az oktatás hatékonyságát befolyásoló indikátorokat meg tudjuk határozni, valamint annak gyenge pontjaira is rámutat. ². A vizsgálatot részben összekapcsolták a PISA³¹⁴ kutatással is, hiszen feltételezték, hogy a tanulói teljesítményméréseken elért eredmények és a pedagógusok elégedettsége és megbecsültsége között kapcsolat van.

A felmérés fő csomópontjaiban⁴ jól visszaköszönnek a humán teljesítménytámogató technológia kritériumai, természetesen a kutatás az oktatási szintérré vetíti a modellben megjelenő indikátorokat.

A tanárkutatást eddig két alkalommal végezték el, 2008-ban⁵ 24, 2013-ban már 34 országban végezték el⁶. Magyarország csak a korábbi mérésben vett részt. A TALIS-vizsgálat rétegzett

¹ <http://static.publico.pt/DOCS/sociedade/TALIS.pdf>

² vö. Humán Teljesítménytámogató Rendszer céljai és modellje

³ A PISA a 15 évesek szövegértési képességeit, valamint matematikai és természettudományos tudását teszteli, és amelyen Magyarország sem 2 éve, sem 5 éve nem teljesített jól. A PISA 2012-felmérés eredményei szerint a magyar 15 évesek nem csak a szövegértésben, matematikában és a természettudományokban rontottak jelentősen, hanem a kreatív problémamegoldásban is, ugyanis nőtt a szakadék a gyengék és a jól teljesítők között, azaz a gyengék még gyengébbek lettek. A mérések során a Robinson-effektust alkalmazzák, amely komplex és kreatív problémamegoldást kívánja vizsgálni, valós helyzeteket modellezve.

⁴ A TALIS során a szakértők mérték a hatékony tanítási -tanulási folyamat feltételeit, az iskolai és a tanári munka értékelését és az ezekről való visszajelzést, a pedagógusok szakmai továbbfejlesztésének, továbbképzésének lehetőségeit, a tanítással kapcsolatos attitűdök és a tanítási gyakorlat helyzetét, a pályakezdő pedagógusok mentorálását, a pályakezdők tanítási munkájának értékelését, illetve az iskolai környezet jellemzőit.

⁵ Magyarország 2008-ban részt vett a felmérésben.

⁶ 2008-ban részt vett országok:

OECD-tagországok:

Ausztrália, Ausztria, Belgium, Dánia, Írország, Izland, Korea, Lengyelország, Magyarország, Mexikó, Norvégia, Olaszország, Portugália, Spanyolország, Szlovákia, Törökország

Partnerországok: Brazília, Bulgária, Észtország, Litvánia, Malajzia, Málta, Szlovénia

2013-ban részt vett országok:

véletlen mintavételen alapuló kérdőíves adatgyűjtés volt, amelynek elsődleges mintavételi egysége az iskola.⁷ Mindenhol 200 iskolát választottak ki, ahol húsz, 5.-8. osztályban tanító tanár és egy vezető töltötte ki a kérdőívet. Az OECD a résztvevő országokon kívül egy nemzetközi kutatási konzorciummal valamint társadalmi partnerekkel és az Európai Bizottsággal működik együtt.

A pedagóguspálya klímavizsgálata, valamint a tanárok saját életpályájának megítélése tekintetében a kutatás eredményei azt tükrözik, hogy tízből majdnem tíz tanár elégedett, nyolc újra ezt a foglalkozást választaná, de csak harmaduk érzi azt, hogy megbecsülik. Pedig azokban az országokban, ahol érzik a megbecsülést, a diákok is jobban teljesítenek, például a PISA-méréseken, derült ki az OECD nemzetközi felméréséből.

A 2008-as hazai eredmény szerint a magyar tanárok a többi ország átlagához viszonyítva elégedetlenebbek a munkájukkal, és kevésbé hatékonyak tartják a saját teljesítményüket. Ezen kívül az átlagnál jóval kisebb arányban mondták azt, hogy szakmai fejlődésükhöz több továbbképzésre lenne szükségük. Kiderült továbbá, hogy Magyarországon kiugróan magas azoknak a pedagógusoknak az aránya, akik a képzés ára miatt nem tudtak részt venni továbbképzésen. Ezen túlmenően a tanárkutatás eredményei mutatták meg, hogy a hazai iskolákban jellemzőbbek az ismereteket rendszerező tanórai tevékenységek és kevésbé jellemzőek a diákorientáltak, ráadásul a fiatalabb tanárok a konzervatívabbak. A magyar oktatáskutatók azt feltételezték, hogy a negatív önszelekció felerősödése, illetve a tanárképzés változásai állhatnak a háttérben.⁸

A következő mérés 2018-ban lesz, amelyben már 2013-ban bevezetett új minősítési rendszer, a pedagógus életpályamodell hatását is lehet(ne) mérni, ha hazánk újra csatlakozna a méréshez.

4.2 A pedagógusi munka minőségi értékelése-pedagógus életpálya-modell

A 2013-ban bevezetésre került pedagógusi munka komplex, minőségi értékelés nagy úrt tölt be a pedagógus teljesítményértékelés rendszerében. A pedagógusi munkaértékelésen korábban is megvalósult, azonban sok esetben inkább lokálisan, intézményi szinten, például formatív, igazgatói értékelések során. A mostani modell célja, hogy a „*pedagógus életpályamodellben a pedagógusképzésnek, a pályakezdők mentorálásának, támogatásának és a pedagógusok folyamatos – egész pályafutás alatt zajló – szakmai fejlődésének, továbbképzésének egységes rendszerré kell összeállnia*”.¹⁵

Az értékelés az egyéni teljesítményen alapul, amelyhez portfóliót⁹ kell a tanároknak készíteni, amelynek célja a pedagógiai fejlődés és tapasztalatok folyamatos rögzítése jelenti az életpálya

OECD-tagországok: Alberta (Kanada), Ausztrália, Chile, Csehország, Dánia, Anglia (Egyesült Királyság), Észtország, Finnország, Belgium, Franciaország, Izland, Izrael, Olaszország, Japán, Korea, Mexikó, Hollandia, Norvégia, Lengyelország, Portugália, Szlovákia, Spanyolország, Svédország, Egyesült Államok

Partnerországok: Abu Dhabi (Egyesült Arab Emírségek), Brazília, Bulgária, Horvátország, Ciprus, Litvánia, Malajzia, Románia, Szerbia, Szingapúr

⁷ „Teaching and Learning International Survey”: PEDAGÓGUSOK AZ OKTATÁS KULCSSZEREPLŐI: Összefoglaló jelentés az OECD nemzetközi tanárkutatás (TALIS) első eredményeiről.

Oktatáskutató és Fejlesztő Intézet Budapest, 2009 p. 5.

⁸ „Teaching and Learning International Survey”: PEDAGÓGUSOK AZ OKTATÁS KULCSSZEREPLŐI: Összefoglaló jelentés az OECD nemzetközi tanárkutatás (TALIS) első eredményeiről.

Oktatáskutató és Fejlesztő Intézet Budapest, 2009 p. 5.

⁹ A portfólió egy olyan dokumentumgyűjtemény, amely alapján végigkísérhető a pedagógus (vagy más szakember) szakmai útja, tevékenysége, fejlődése, nehézségei és sikerei egyrészt a tények tükrében, másrészt magának a pedagógusnak a reflexiói, értelmezése alapján. (Kotschy, 2013)

különböző szakaszaiban. ¹⁶ A portfólió a pedagógiai gyakorlatban már nemcsak a pedagógusok, hanem a tanulók fejlődését bemutató dokumentummá is vált.

Az iskolai gyakorlatban két kiemelten fontos funkciója van/lesz a portfóliónak. Egyrészt a szakmai fejlődés folyamatos segítése, másrészt az életpálya egyes szintjein a feljebb lépést lehetővé tevő minősítés megalapozása. A rendszeres szakmai ellenőrzés-értékelés során a szakértők és szaktanácsadók számára fontos információforrás a pedagógus portfóliója, amelyet a támogatás/segités szándékával elemeznek, értékelnek. ¹⁷

Az életpálya különböző szintjeire való pályázások során viszont az értékelés lesz jellemző. A kétféle megközelítés a portfólió két különböző típusát: a folyamatosan készített „munkaportfóliót”, illetve a cél által meghatározott „értékelési vagy bemutató portfóliót” részesíti előnyben. ¹⁸

A szakmailag kiváló portfólió nagyban segítheti a humán teljesítmény növelését és monitorozását, hiszen bemutatja a pedagógus személyiségét, a készítőjének a módszertani kultúráját, a pedagógus egyéni szemléletmódját, gyermekképét; a pedagógus elhivatottságát, valamint hogy a pedagógus magas szinten a birtokában van mind a nyolc kompetenciának ¹⁰.

A pedagógusok mellett a tanulókkal szemben is számos eljárás fogalmazódik meg a teljesítményükkel kapcsolatban, számos új kompetenciát el kell sajátítaniuk a jövőben a jobb humán teljesítmény elérése érdekében.

5. Elvárások a tanulói kompetenciák oldaláról

A Nemzeti Infokommunikációs Stratégia 2014-2020 egyik pillérét a digitális kompetenciák képezik, amely során a cél a „lakosság, a mikro-és középvállalkozások, illetve a közigazgatásban dolgozók digitális kompetenciáinak fejlesztése, az elsődleges (digitális írástudatlanság) és a másodlagos (alacsony szintű használat) digitális megosztottság 11 mérséklése, illetve a tartósan leszakadók részesítése a digitális ökoszisztéma előnyeiből (eBefogadás).” ¹⁹ Ahogyan a megfogalmazásból kitűnik, stratégiai elemmé válik a digitális kompetencia, amelyre erősen hat a technológiai determináció.

A stratégia horizontális céljai között jól kirajzolódik a digitális állampolgárság modelljének néhány aspektusa, például a biztonság vagy az eBefogadás.

A digitális állampolgárság modelljének rendszerét -több elméletet szintetizálva (Mossberger, Tolbert és McNeal)- az International Society for Technology in Education (ISTE) dolgozta ki, amely az alábbi kompetenciát tartalmazza: digitális hozzáférés, digitális műveltség, digitális kommunikáció, digitális felelősség, digitális etikett, digitális biztonság, digitális kereskedelem, digitális jog, digitális egészség és közérzet ²⁰

ISTE- International Society for Technology in Education Ribble -2011	ELTE PPK ITOK Digitális Állampolgárság modell 2013
---	---

¹⁰ A pedagógusoktól elvárt nyolc kompetencia : a tanuló személyiségének fejlesztése, az egyéni bánásmód érvényesülése; a tanulói csoportok, közösségek alakulásának segítése, fejlesztése; esélyteremtés, nyitottság a különböző társadalmi-kulturális sokféleségre; szaktudományos, szaktárgyi, tantervi tudás; a pedagógiai folyamat tervezése; a tanulás támogatása; a pedagógiai folyamatok és a tanulók személyiségfejlődésének folyamatos értékelése; kommunikáció és szakmai együttműködés; elkötelezettség, felelősségvállalás a szakmai fejlődésért.

¹¹ vö. second digital divide, vagyis a második szintű digitális szakadék. (Nemzeti Infokommunikációs Stratégia 2014-2020, 35.)

<ul style="list-style-type: none"> • digitális hozzáférés • digitális műveltség • digitális kommunikáció • digitális felelősség • digitális etikett • digitális biztonság • digitális kereskedelem • digitális jog • digitális egészség és közérzet 	<ul style="list-style-type: none"> • kommunikáció • hozzáférés • eszközhasználat • digitális egészség • digitális én-megjelenítés • digitális együttélés • értékteremtés • produktivitás • időgazdálkodás • tartalom-menedzsment
--	--

4. ábra Digitális állampolgárság kompetenciamodellek

A hazai modellben ²¹hasonló elemek érhetőek tetten, azonban kissé árnyaltabb képet mutatva: kommunikáció, hozzáférés, eszközhasználat, digitális egészség, digitális én-megjelenítés, digitális együttélés, értékteremtés, produktivitás, időgazdálkodás, tartalom-menedzsment.

A digitális állampolgárság ma már túlmutat az információs műveltség, médiaműveltség és a digitális írástudás fogalmain, azonban ezen komponensek nélkül nem valósulhat meg az új felfogása, amely az online produktív jelenlétet foglalja magába.

A DigComp 2013²² kutatás a digitális kompetencia keretrendszerét és megértésének segítségét tűzte ki célul Európában. A tanulmány egy hosszas társadalmi egyeztetés eredménye, amely részletes keretet ad a digitális kompetenciáról minden európai polgár számára. A kompetenciák részletes leírása mellett a jártasságok, a tudás, készségek és attitűdök is bemutatásra kerülnek. Három tudásszintet javasolnak minden kompetenciához és egy fejlesztési indikátorrendszer segítségével mindenki meghatározhatja saját digitális kompetenciaszintjét. Az információ, a kommunikáció, a tartalomlétrehozás, a biztonság és a problémamegoldás alkotja az öt pillért.

A fogalomhoz köthető a globális állampolgárrá/állampolgári nevelés koncepciója, amelyet az UNESCO hívott életre, és számos fórumon támogatja a kezdeményezést. A globális állampolgárrá nevelés ugyanis átalakítja a gyerekek gondolkodását, és egy igazságosabb és toleránsabb társadalmat hív életre.²³ A globális állampolgárrá nevelés csak az oktatáson keresztül valósulhat meg, ahogyan a 2013-as UNESCO kongresszuson, is hangsúlyozták: "Az oktatás a legerősebb fegyver, amely segítségével változtatni a világot."²⁴ és nagyban hozzájárul a békés társadalmak együttélésért.

6. Összegzés

A humán teljesítménytámogató technológia kezdetbe az oktatástechnológia területéről fejlődött ki, ma már az iparban történő alkalmazása a domináns. Ahogyan a tanulmányból láthatjuk az oktatásban ma is központi szerepet tölt be a teljesítmény és mérése, csak nem annyira tudatosan és szisztematikusan alkalmazott formában, mint a versenyszférában. A teljesítmény mérésére és támogatására számos alternatív megoldás bontakozott ki (pl. tanulói teljesítménymérések, pedagógus életpályamodell portfólió), amely a HPT elvein nyugszik.

Munkám során úgy vélem, hogy sikerült átfogó képet adnom a rendszer fogalmi struktúrájáról és a köznevelésben betöltött szerepéről. További elemzések során tervezem a tanulói teljesítménymérések és az életpályamodell humán teljesítménytámogatás szempontjai alapján történő elemzését.

7. Köszönetnyilvánítás

A kutatás a TÁMOP-4.2.4.A/2-11/1-2012-0001 azonosító számú Nemzeti Kiválóság Program – Hazai hallgatói, illetve kutatói személyi támogatást biztosító rendszer kidolgozása és működtetése konvergencia program című kiemelt projekt keretében zajlott. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Ábrajegyzék

1. ábra A Humán Teljesítménytámogató Technológia három összetevője	194
2. ábra A HPT tíz alapelve /	194
3. ábra A HPT-modell: a teljesítményfejlesztés modellje (saját fordítás).....	195
4. ábra Digitális állampolgárság kompetenciamodellek	200

Irodalomjegyzék

-
- 1 Dean, P. J. and Ripley, D. E. (Eds.) (1997). Performance Improvement Pathfinders. International Society for Performance Improvement (ISPI). Washington, DC
 - Gilbert, Thomas F. (2007). Human Competence: Engineering Worthy Performance , Pfeiffer, ISBN 978-0-7879-9615-4
 - Stolovitch, H., and E. Keeps (1999). What is Human Performance technology? In H. Stolovitch, and E. Keeps (eds), Handbook of Performance Improvement Technology (2nd ed., pp. 3–23). San Francisco: Jossey-Bass.
 - 2 Pershing, J.A. (2006). Handbook of Human Performance Technology: Principles Practices Potential. San Francisco: Pfeiffer. ISBN 0-7879-6530-8.
 - 3 ISPI What is Human Performance Technology? Letöltve: 2014. július 7. URL: <http://www.ispi.org/content.aspx?id=54>
 - 4 Chyung, S. Y. (2008). Foundations of instructional and performance technology. Amherst, MA: HRD Press.
 - ⁵ ISPI What is Human Performance Technology? retrieved June 12, 2012 from <http://www.ispi.org/content.aspx?id=54>
URL: <http://creativegapminding.com/category/performance-improvement/> (Letöltve: 2014. július 7.)
 - 6 Stolovitch, H., and E. Keeps (1999). What is Human Performance technology? In H. Stolovitch, and E. Keeps (eds), Handbook of Performance Improvement Technology (2nd ed., pp. 3–23). San Francisco: Jossey-Bass.
 - 7 Chyung, S. Y. (2008). Foundations of instructional and performance technology. Amherst, MA: HRD Press.

-
- 8 Van Tiem-J-L. Moseley-Joan C. Dessinger (2012): A teljesítményfejlesztés alapjai
URL: <http://www.ispi.org/images/HPT-Model/HPT-Model-2012.jpg> (Letöltve: 2014. július 7.)
 - 9 Sági Matild (2009): A tanári munka értékelése és az iskolai eredményesség
URL: <http://www.ofi.hu/tudastar/eredmenyesseg/sagi-matild-tanari-munka>
(Letöltve: 2014. július 7.)
 - 10 Goldhaber, D. (2002): The Mystery of Good Teaching. Surveying the Evidence on Student Achievement and Teacher Characteristics. *Education Next*, 2 (1), pp. 50–55.
 - Wright, P. S. – W. Sanders (1997): Teachers and Classroom Heterogeneity. Their Effects on Educational Outcomes. *Journal of Personnel Evaluation in Education*, 11 (1), pp. 57–67.
Wright, P. S. – W. Sanders (1997): Teachers and Classroom Heterogeneity. Their Effects on Educational Outcomes. *Journal of Personnel Evaluation in Education*, 11 (1), pp. 57–67.
 - 11 Sági Matild (2009): A tanári munka értékelése és az iskolai eredményesség
URL: <http://www.ofi.hu/tudastar/eredmenyesseg/sagi-matild-tanari-munka>
(Letöltve: 2014. július 7.)
 - 12 Darling-Hammond, L. (2000): Teacher Quality and Student Achievement. A review of State Policy Evidence. *Education Policy Analysis Archives*, vol. 8. No. 1. pp. 1–46.
 - 13 TALIS 2013 Results: An International Perspective on Teaching and Learning.
URL: http://www.keepeek.com/Digital-Asset-Management/oecd/education/talis-2013-results_9789264196261-en#page1 (Letöltve: 2014. július 7.)
 - 14 OECD. PISA 2012 Results: creative problem solving students's in tacking real-life problems (vol.5). OECD-jelentés, OECD, 2014.
 - 15 Thaisz Miklós (2013): Elkötelezettség és elhivatottság Pedagógus életpálya. Oktatásért Felelős Államtitkári Kabinet
URL: www.kpszt.hu/med/eletut/eletpalya_modell.ppt (Letöltve: 2014. július 7.)
 - 16 Falus Iván – Kimmel Magdolna 2003. A portfólió. ELTE BTK Neveléstudományi Intézet – Gondolat Kiadói Kör, Budapest.
 - 17 Kotschy Beáta (2013): A portfólió szerepe a pedagógusok minősítési folyamatában.
URL: https://www.oktatas.hu/pub_bin/unios_projektek/tamop_315_pedkepzes_fejl/3_1_5_1_hir_level_Kotschy_Beata__portfolio.pdf (Letöltve: 2014. július 7.)
 - 18 Kotschy Beáta (szerk.) 2011. A pedagógussá válás és a szakmai fejlődés sztenderdjei. Módszertani kiadványok. Eszterházy Károly Főiskola, Eger.
www.epednet.ektf.hu/eredmenyek/a_pedagogussa_valas_es_a_szakmai_fejlodes_sztenderdjei.pdf (Letöltve: 2014. július 7.)
 - 19 „Nemzeti Infokommunikációs Stratégia.” országos stratégia, Budapest, 2014-2020.

- 20 Ribble, M. Digital Citizenship in Schools. Second Edition. Eugene, Oregon, Washington, D.C.: International Society for Technology in Education., 2011.
- 21 Ollé, János, és mtsai. Digitális állampolgárság az információs társadalomban. Budapest: Eötvös Kiadó, 2014.
- 22 Ferrari, Anusca, és Yves-Brecko N., Barbara Punie. DigComp: A Framework for developing and Understanding Digital Competence in Europe. Report EUR 26035, Brüsszel: Joint Research Centre, 2013.
- 23 Pigozzi , Mary Joy A UNESCO view of global citizenship education. Volume 58, Issue 1, 2006
- ²⁴ UNESCO Forum on Global Citizenship Education: Preparing learners for the challenge of the 21st Century) http://www.huffingtonpost.com/sofia-gomezdoyle/at-the-unesco-forum-on-gl_b_4509311.html (Letöltve: 2014. július 7.)

Nem vagy Anonymus

You Are Not Anonymous

Bátfai Erika^a, Bátfai Norbert^b

^aDebreceni Egyetem Egyetemi és Nemzeti Könyvtár
ebatfai@lib.unideb.hu

^bDebreceni Egyetem Informatikai Kar
batfai.norbert@inf.unideb.hu

Absztrakt: Mindennapi életünk folyamatosan bővülő területe a közösségi média. A felhők és a különféle szociális média korában a hálózatok egyre több adatot tárolnak. Milyen valódi kapcsolatokat nyerhetünk ki innen? Miként profitálhat a társadalom ezekből? Hogy lehetséges a kapcsolati hálót reprezentálni? Egy valódi BigData megoldást mutatunk ezekre a kérdésekre.

Véleményünk szerint anonim kapcsolati hálózatokból előállíthatók valós közösségi hálók. Ehhez valós pártpreferencia adatokat gyűjtünk, hogy az azok alapján felrajzolt gráfokból egy adott nagyobb közösség jellemzőit tudjuk szemléltetni.

Fő módszernek a kapcsolatháló-elemzést választottuk, mivel az a módszer adja a leginkább megfogható eredményeket a készítendő programhoz. (Miért nem a szociometriát? A szociometriával a személyközi kapcsolatok kvantitatív feltárását végezhetjük. A szociometria elsősorban a szoros kiscsoportok vizsgálati módszere.)

Maga az anyaggyűjtés a *laza, személyes hálózati tőkén*k használatával folyik. A hálónkat nagyon egyszerűen készítjük el: nominalista megközelítéssel mi magunk húzzuk meg a hálónk határát, nem cél a teljes kapcsolati háló feltérképezése, csak egy bizonyos mennyiségű, valós adatra van szükség, ami nem tükör és nem következtetünk ez alapján egy nagyobb közösség véleményére, ám megadja a közösségi háló jóságának az alapját. Azaz egy centrális ego-hálónk lesz.

A megadott minták kerülnek be a YANonymous alkalmazásba és egy valós társadalmi hálót vázolunk a különböző névtelen egyirányú, egy távolságú kapcsolatokból.

Kulcsszavak: közösségi hálózatok, kapcsolatháló, kapcsolatháló-elemzés, BigData

Abstrakt: The social media are an expanding area of life nowadays. More and more data are stored in the nets in the age of clouds and miscellaneous social media. What real relationships can be recovered from? As the society can benefit from them? How is it possible to represent the social network? A real BigData solution is shown to these questions.

In our opinion it is possible to manufacture social network from anonymous social net. We collect anonymous party preference information and then we can map a wider community network by the software which has been developed for this goal.

As a main method we chose the social-network analysis, because this method provides the fairly tangible results to our program. (Why not sociometry? The interpersonal relationships are explored quantitatively by the sociometrics. Sociometry is primarily the method of close groups.)

The data collection is running in our loose, personal social capital. We concocted our net very simply: margin of our net mark out by ourselves with the nominalist approach. There isn't goal mapping of the whole network of

contacts, just an certain amount of real data are needed, which are no mirror and we don't conclude based on these to larger community reviews, however these gave basis of social net goodness'. That is we have a central ego-net.

The provided subgraphes are going into YaAnonymous application and there real social net will be outlined from the anonymous various using the one-way, one-distance relations.

Keywords: social networks, social network analysis, BigData

1. A közösségi hálózatok

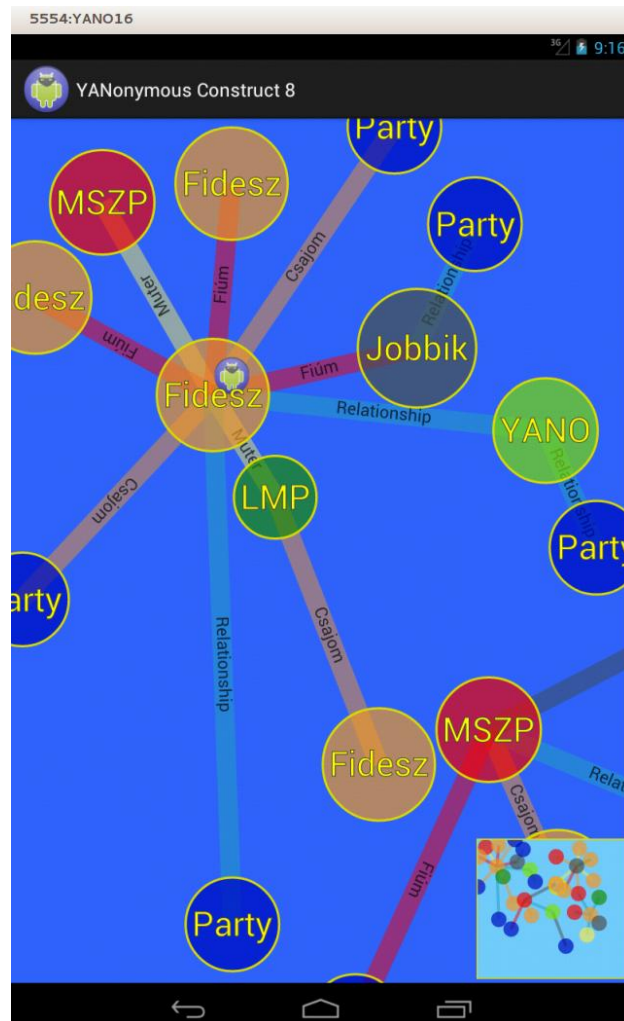
A közösségi hálózat mérhetőségét, azaz a “kis világ” szociális útvonalának kézfogásszámra váltását Karinthy [Láncszemek](#) c. novellájában vetette fel, amikor is arra jutott, hogy 2 egymás számára ismeretlen ember közötti távolság 5 másikon keresztül vezet. Érdekes módon matematikai [1][2], szociológiai és pszichológiai [3] kutatások nagyjából szintén ezt a számot hozták ki, a közösségi hálók hallatlan adattömegei is ezt igazolták.

Az 1995-ben induló [Classmates](#) után egymás után bújtak elő a különféle valós kapcsolati hálókból építkező online ismeretségi hálók, hogy mára a 2004-ben indult Facebook a kapcsolati tőkére épülő közösségi hálót [4] átértelmezve tartalomfogyasztási indexként is működjön – ezzel a funkciójával egyben kiszorítva a nem így működőket. [5]

2. YANonymous

Véleményünk szerint anonim kapcsolati hálózatokból előállíthatók valós közösségi hálók. Ehhez valós pártpreferencia adatokat gyűjtünk, hogy az azok alapján felrajzolt gráfokból egy adott nagyobb közösség jellemzőit tudjuk szemléltetni. A [YANonymous](#) = YANA (You Are Not Alone) + Anonymous = You Are Not Anonymous

A kapcsolati hálót a **YANonymous Construct** program építi fel (lásd 1. ábra [6]), amiből a **YANonymous Puzzle** készíti el a közösségi hálót. Az első hálónk tehát kicsi, lokális és anonim, az utóbbi globális és az előbbiből számított. (Az alábbi ábrán még a kapcsolatok normalizálása előtti állapot látható.)



1. ábra YANonymous Construct

Fő módszernek a kapcsolatháló-elemzést választottuk, mivel az a módszer adja a leginkább megfogható eredményeket a készítendő programhoz. Felvetődhet, miért nem (elég) a szociometria. A nemhez járjuk kicsit körül a szociometriát:

A szociometriával elsősorban szintén a személyközi kapcsolatok kvantitatív feltárását végezhetjük [7]. Ez az 1930-as évekbeli módszer, az 1940-es évekre mozgalommá nőtt; mindeközben Moreno kidolgozza a majdani alkalmazott szociálpszichológia csoport-pszichoterápia és pszichodráma lábait.[8][9]

A szociometria elsősorban a szoros kiscsoportok vizsgálati módszere, Moreno elképzelése szerint az együttélés, az együtt dolgozás és az együttes társas élet szociometriai kritériumokra használható és használandó. A szociometriai vizsgálat eredményei szociogrammal és/vagy kölcsönösségi szociomátrix-szal ábrázolhatóak.[10]

Mind az Erdős-Rényi-, mind a Watts-Strogarz-modellben a pontok száma fix, a hálózatok demokratikusak és minden pont véletlenszerűen alkot kapcsolatot. Mindeközben a különböző hálózatok jellemzője a növekedés és hogy egyes pontok kitüntetett szerepet játszanak - központiak és a kapcsolódás alapja a népszerűség.[11]

Maga az anyaggyűjtés a *laza, személyes hálózati tőkének* használatával folyik: pontosan tudjuk definiálni a hálózatba tartozók körét, a kapcsolatok típusát, kérdés, hogy az így létrejött mintázat mennyire fedi az országos, nem hálózati tőke hasznosításával megszülető mintát.

A hálónkat nagyon egyszerűen készítjük el: nominalista megközelítéssel egy mi magunk húzzuk meg a hálónk határát, nem cél a teljes kapcsolati háló feltérképezése, csak egy bizonyos mennyiségű, valós adatra van szükség, ami nem tükör és nem következtetünk ez alapján egy nagyobb közösség véleményére, ám megadja a közösségi háló jóságának az alapját. Azaz egy centrális ego-hálónk lesz.[12] Szintén lényegtelen a kapcsolatok sűrűsége vagy kölcsönössége.

3. A kapcsolatok normalizálása

A kapcsolatok normalizálása érdekében felállítottuk az elemi kapcsolattípusokat, aminek a segítségével elkerülhetőek/minimalizálhatóak a körök. Csak asszimetrikus kapcsolatok kerülhetnek szóba:

yanonymous → anya	yanonymous → apa
yanonymous → férj	yanonymous → feleség
yanonymous → gyerek	yanonymous → ismerős
yanonymous → barátnő	yanonymous → barát
yanonymous → szomszéd	yanonymous → lakótárs
yanonymous → iskolatárs	yanonymous → csapattárs
yanonymous → munkatárs	yanonymous → ex

A gyakorlatban ezt azt jelenti, hogy az sógornő a feleség/férj testvéreként; az unoka a gyerek gyerekeként képezhető le.

4. Összegzés

Az adatok behálózhatnak bennünket, a kapcsolati hálónk könnyen láthatóvá tesz bennünket akkor is, mikor az anonimitásba vonulunk. Jelen esetben az anonim megadott mintáink kerülnek be a YANonymous alkalmazásba és ezekből egy valós társadalmi hálót vázolunk a különböző névtelen egyirányú, egy távolságú kapcsolatok segítségével.

A Yanonymous egy közösségi fejlesztés (<http://sourceforge.net/projects/yanonymous/>) és ez nem csupán annyiban merül ki, hogy a projekt tárolójának a SourceForge ad otthont, hanem valóban egy valódi közösség gondozza. Az igazsághoz nyilvánvalóan hozzátartozik, hogy ezt (egyelőre) nem a projekt sikere, hanem annak szervezése biztosítja, miszerint ez a projekt a Debreceni Egyetem Magas szintű programozási nyelvek 2 kurzusának sztenderd példája is. Sőt, magát a projektet bemutató tudományos publikáció kéziratát [13] is közösségi fejlesztés formájában képzeljük el, hiszen ezzel arra teszünk kísérletet, hogy ugyanúgy a tárolóban gondozzuk, mint ahogyan magát a szoftvert is. Ebben önmagában nincs újdonság, hanem abban, hogy ez teljesen nyitott, olyannyira, hogy a kézirat is abszolúte szabadon hozzáférhető. Ez persze jelenthet majd megszorításokat a célfolyóirat kiválasztásánál, de maga ez a próba ennyi plusz erőfeszítést megér.

5. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Watts, Duncan J. - Strogatz, Steven H.: *Collective dynamics of 'small-world' networks*. <http://www.nature.com/nature/journal/v393/n6684/full/393440a0.html>
- [2] Albert Réka – Barabási Albert-László: *Statistical mechanics of complex networks*. <http://journals.aps.org/rmp/abstract/10.1103/RevModPhys.74.47>
- [3] Milgram, Stanley: *The small world problem*. http://measure.igpp.ucla.edu/GK12-SEE-LA/Lesson_Files_09/Tina_Wey/TW_social_networks_Milgram_1967_small_world_p_robblem.pdf
- [4] Ellison, Nicole B. – Steinfield, Charles – Lampe, Cliff: *The Benefits of Facebook 'Friends: ' Social Capital and College Students' Use of Online Social Network Sites*. <http://onlinelibrary.wiley.com/doi/10.1111/j.1083-6101.2007.00367.x/pdf>
- [5] Straub Ádám: *Tizenkét év után bezár az iWiW*. <http://www.origo.hu/techbazis/20140514-tizenket-ev-utan-bezar-az-iwiw.html>
- [6] YANonymous = YANA + Anonymous = You Are Not Anonymous http://progater.blog.hu/2013/09/26/yanonymous_yana_anonymous_you_are_not_anonymous
- [7] Szántó Zoltán: A társadalmi kapcsolatháló-elemzés szociometriai gyökerei. In. Letenyei L. (szerk.): *Településkutatás : a települési és térségi tervezés társadalomtudományos alapozása*. Budapest: L'Harmattan:Ráció, 2005. p. 649.
- [8] Mérei Ferenc: *Közösségek rejtett hálózata*. Budapest: Osiris, 2006. p. 44-49.
- [9] Freeman, Linton C.: *The development of social network analysis*. Vancouver: Empirical Press, 2004. p. 36-42.
- [10] Mérei Ferenc: i. m. p. 50-58.
- [11] Barabási Albert László: A hetedik láncszem - a gazdag egyre gazdagabb. In. Takács Károly: *Társadalmi kapcsolathálózatok elemzése*. http://www.tankonyvtar.hu/hu/tartalom/tamop425/0010_2A_08_Kapcsolathalo_elemzes_szerk_Takacs_Karoly/adatok.html p. 87-91.
- [12] Kürtösi Zsófia: A társadalmi kapcsolatháló-elemzés módszertani alapjai. In. Letenyei L. (szerk.): *Településkutatás: a települési és térségi tervezés társadalomtudományos alapozása*. Budapest: L'Harmattan:Ráció, 2005. p. 665-669.
- [13] Bátfai Norbert – Smajda Máté - Sziráki Szabolcs – Bátfai Erika: *Reproducing Social Networks from Anonymous Community Networks*. <http://sourceforge.net/p/yanonymous/code/ci/master/tree/Manuscript/> [készülő kézirat]

Az élethelyzetnek megfelelő aktív internet alkalmazások frissdiplomások körében

The use of active internet applications fitted to the present life situation in the group of young professionals

Eszenyiné Borbély Mária

Debreceni Egyetem Informatikai Kar
borbely.maria@inf.unideb.hu

Absztrakt: A Digitális Agenda, az Európai Unió digitális menetrendje, a digitális kompetencia fogalmi keretébe sorolja az élethelyzetnek megfelelő aktív internet alkalmazások csoportját is. Ebbe a körbe tartozik többek között a munka/hivatás, a tanulás (LLL), a kommunikáció, a társadalmi érintkezés, a szabadidő és a közösségi háló.

Sokan úgy gondolják napjainkban, hogy a digitális írástudás egyik fokmérője az lehet, hogy ki hány élethelyzetnek megfelelő alkalmazást használ rendszeresen. A jelen kutatás során felhasznált internet-használati mutatók az Eurostat szabadon hozzáférhető adatbázisaiból származnak. A szerző korábbi vizsgálatai kimutatták, hogy az alapvető informatikai jártasságra egyértelműen befolyással van az életkor és a nem. Egy adott életkori csoport alapfokú, középfokú és felsőfokú végzettséggel rendelkező tagjainak lényegesen eltérő hányada birtokolja a magas számítástechnikai jártassági szintet. Az aktív munkavállalók csoportjában, uniós átlagban az alapfokú végzettségűek 12%-a, a középfokú végzettségűeknek 25%-a, a felsőfokú végzettségűeknek pedig 52%-a rendelkezik magas szintű jártassággal. Magyarországon az alapfokon iskolázottak közül még annyian sem rendelkeznek magas szintű informatikai jártassággal, mint az uniós átlag (10%), viszont a közép-és felsőfokú végzettségűek nagyobb arányban birtokolják azt az átlagosnál. (33 és 65%)

A tanulmány azt vizsgálja, hogy a frissdiplomásokat milyen szintű internet-használati jártasság jellemzi, és melyek körükben a legszélesebb körben használt internet-alkalmazások. Van-e különbség az egyes uniós tagállamok fiatal értelmisége között internetezési szokásaik és kompetenciáik területén? Kimutatható-e eltérés a nemek között?

Kulcsszavak: internet-használat, diplomások, digitális írástudás

Abstract: The group of active applications fitted to the present life situation was classified as the part of digital competence by the Digital Agenda and European Union digital schedule. Work/ profession, learning (LLL), communication, social interactions, spare time and social network also belong into this group.

Many scientists consider the number of regularly used active internet applications fitted to the present life situation as the scale of digital literacy. Internet applications data used for the present study were gained from Eurostat free access databases.

The previous studies of the author, have shown the age and gender significantly influenced the basic informatics skills. Different ratio of a given age group possess the higher level of computer skills based on their qualifications. In the group of active workers (EU average) 12% with low, 25% with medium and 52% with high educations possess higher level of computer skills. In Hungary less people (10%) possess higher level of computer skills than the EU average, but people with medium or higher educations have better results (33% and 65%) than the EU average.

This study focuses on the level of internet skills the young professionals possess and the most often used internet applications. Also looking for the differences among the EU member state young professionals, their competences and the differences between genders.

Keywords: Internet use, young professionals, digital competence

1. Bevezetés

Napjaink frissdiplomásai jellemzően az 1990-1993 között született fiatalok csoportjához tartoznak. A fiatal felnőtteknek ez a csoportja a manapság nagy népszerűségnek örvendő generációs elméletek szerint az Y generációhoz sorolható. Az egyes kutatók véleménye megoszlik azzal kapcsolatban, hogy mely időintervallumban születtek alkotják az Y generációt. Greame Cordington felosztása szerint az Y generációhoz vagy más megnevezéssel élve, a millenniumi nemzedékhez az 1989-2008 között születettek tartoznak [1]. Konszenzus a téma kutatói között abban látszik kibontakozni, hogy a korai 80-as évektől a 2000-es évek elejéig születettek sorolhatók ebbe a körbe, és természetesen értelmetlen merev határokat húzni a generáció definiálásához.

A jelen vizsgálat fókuszában álló fiatal diplomás pályakezdők a generáció derékhadát alkotják. Nagy valószínűséggel rendelkeznek a millenniumi nemzedék minden főbb attribútumával. Mindannyian hasonló impulzusok között nőttek fel, elsősorban a technológiai hatások szerepe kiemelkedő a generáció egészét tekintve. Együtt születtek és nevelkedtek a világhálóval, a mobil telefonokkal, és számukra születésük pillanatától természetes az otthoni számítógép, a számtalan TV csatorna megléte. Az Y generáció tagjai a következő meghatározó és irányadó értékeket követik: optimizmus, magabiztosság, önbecsülés, a média- és szórakoztatóipar általi túlterheltség, szituációs tudatosság, konzervativizmus, egyediség, aktív hálózati tevékenység, felelős állampolgárság, etikus fogyasztás, teljesítményre törekvés, erkölcsösség, mesterkéletlenség, készség a változásra, technológiai érzék, világpolgárság széles látókörrrel. [1].

Természetesen az Y generáció sem tekinthető homogén csoportnak még az alapvető ismérvek tekintetében sem, hiszen nem vonatkoztathatunk el az egyes kontinensek, országok, szűkebb és tágabb közösségek, valamint a családok eltérő lehetőségeitől, hatásaitól.

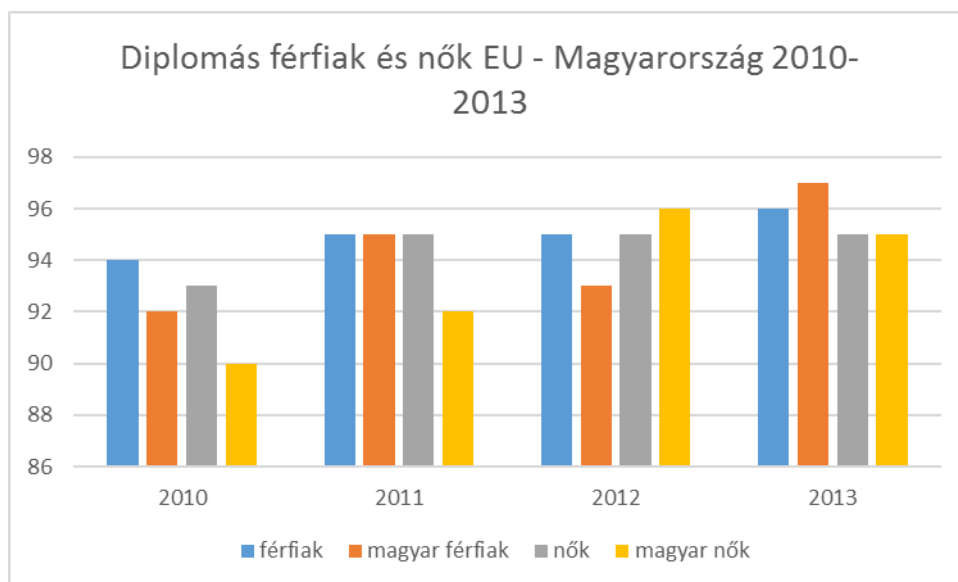
Korábbi vizsgálataimban többek között arra kerestem a választ, hogy az alapvető számítógépes jártasság és az iskolai végzettség között kimutatható-e valamilyen összefüggés az Európai Unió átlaga és az egyes tagállamok, elsősorban Magyarország vonatkozásában. Az adatok tanúsága szerint mindegyik korcsoportban a felsőfokú végzettségűek rendelkeznek nagyobb arányban magas szintű számítástechnikai jártassággal. Ezt követi a középfokú végzettségűek csoportja, majd az alacsony iskolázottaké. A 16-24 éveseknek egy jelentős hányada életkorából kifolyólag még nem rendelkezhet felsőfokú végzettséggel, de ennek ellenére már ebben az életkori csoportban is a felsőfokú végzettségűek birtokolják legkiterjedtebben a magas szintű informatikai kompetenciát. Az aktív munkavállalók csoportjában, uniós átlagban az alacsony fokú végzettségűek 12%-a, a középfokú végzettségűeknek 25%-a, a felsőfokú végzettségűeknek pedig 52%-a rendelkezik magas szintű jártassággal. Magyarországon az alacsony fokú iskolázottak közül még annyian sem rendelkeznek magas szintű informatikai jártassággal, mint az uniós átlag (10%), viszont a közép- és felsőfokú végzettségűek nagyobb arányban birtokolják azt az uniós átlagosnál. (33 és 65%) [2].

Kutatásomban arra keresem a választ, hogy a fiatal diplomások internet-használatának vannak-e korcsoport és végzettség specifikus jellemzői, és kimutathatók-e markáns különbségek az egyes európai uniós tagállamok frissdiplomásai között. Az adatok forrásául az Európai Unió Statisztikai Hivatalának, az Eurostatnak az információs társadalomra vonatkozó statisztikai táblázatai szolgáltak. [3].

2. Az internethasználat gyakorisága, helye és az elérés eszközei

Az internethasználat gyakoriságára, helyére és az elérés eszközeire vonatkozó megállapítások a legfrissebb, 2013-as Eurostat adatokra épülnek. Ma az EU tagállamaiban élők kiterjedt internethasználat jellemzi. A teljes lakosság 77%-a vallotta azt 2013-ban, hogy az elmúlt egy évben használt internetet. A lakosság internet-használati hajlandósága évről-évre folyamatosan emelkedő tendenciát mutat, a 2004-es 47%-os szintről napjainkig. Magyarország lakosságának 74%-a használt internetet az elmúlt évben. Ez alatta marad az EU átlagnak, viszont nagy előrelépés a 2004-ben mért 29%-os magyar mértékhez képest. Dánia, Luxemburg és Svédország lakossága használja a legnagyobb arányban a világhálót, 95%-uk. Dánia és Svédország adatai már 2004-ben is meghaladták a 80%-ot.

Az Eurostat által használt életkori csoportok közül a frissdiplomásokat a 16-24 évesek korcsoportjában találjuk. 2013-ban már 100%-uk vallotta azt, hogy az előző évben használt internetet. Az uniós átlag 2005-ben 95% volt, a magyar frissdiplomásoknak pedig 91%-a használt már akkor is internetet. A felsőfokú végzettség kis mértékben ugyan, de pozitív hatással van az általános internethasználatra. Ugyanezen korosztály középfokú iskolai végzettséggel rendelkező tagjainak 98%-a használt internetet az adatfelvételt megelőző évben. A legnagyobb különbségek a közép-és felsőfokú végzettségűek internet-használatában Bulgáriában, Csehországban és Romániában tapasztalhatók. A felsőfokú iskolai végzettséggel rendelkezőknek, valamennyi életkori csoportot figyelembe véve az EU 28-ak átlagában, 96%-a használ internetet napjainkban. A diplomások csoportjában a nemek közti különbségek teljes elmosódását figyelhetjük meg, bár ez a különbség soha nem is volt jelentős. (lásd 1. ábra) A magyar diplomás nők körében volt erősebb a lemaradás a korábbi években az EU átlag női internet-használatához mérten, de mára elérték azt.



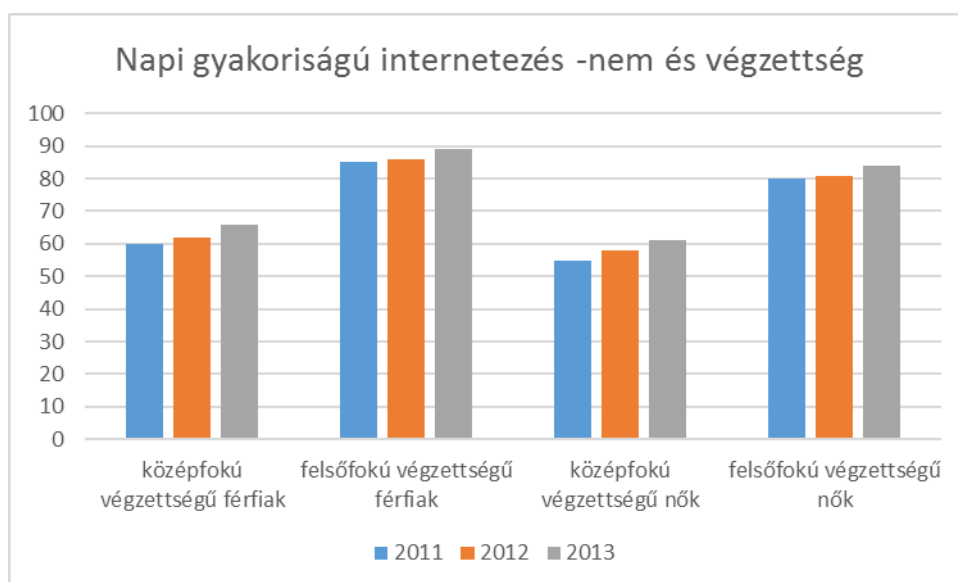
1. ábra. Diplomások internet-használatának változása nemek szerint

Napi rendszerességgel 2013-ban a 24 éves vagy annál fiatalabb diplomásoknak 97%-a használja az internetet az EU-ban. A magyar adat ebben az évben 100%, de egyetlen uniós tagállamban mért érték sem tér el kiugróan az uniós átlagtól. Romániában és Olaszországban a legalacsonyabb a mért érték, 90%, de a korábbi években minkét országban mértek már magasabb értéket is. Megállapítható, hogy az uniós országokban élő fiatal értelmiségiek közel

100%-a napi rendszerességgel használja a világhálót. Ezt a magas használati arányt nagyon rövid idő alatt érte el ez a csoport, mivel 2008-ban a frissdiplomásoknak még csak 80%-a vallotta magát napi rendszerességgel internet-használónak.

A diplomások teljes körét tekintve napjainkban a napi gyakoriságú internethasználat 86%-ukra jellemző. A növekedés itt is jelentős és dinamikus volt, hiszen 2008-ban még csak 70%-ot értek el.

A teljes életkori csoportot vizsgálva szembetűnő, hogy a napi periodicitású internetezést jelentősen befolyásolja a nem és az iskolai végzettség. (lásd 1. ábra) A középfokú végzettségű férfiak és nők közül lényegesen kevesebben interneteznek naponta, mint a felsőfokú végzettségűek közül. Mindkét végzettségi csoportban és mindkét nem esetében évről-évre emelkedik a naponta internetezők száma, de a köztük lévő különbség megmarad. A nemek között is van különbség, mind a középfokú, mind pedig a felsőfokú végzettséggel rendelkezők csoportjában a férfiak közül interneteznek többen naponta. Ugyanakkor a 2. ábrán az is látszik, hogy a végzettség lényegesen nagyon mértékben befolyásolja a napi internet-használatot, mint az, hogy a használó férfi vagy nő.



2. ábra. Napi gyakoriságú internetezés nem és végzettség szerint

Az internet kávézók a 2004-2010 közötti időszakban voltak a legnépszerűbbek a frissdiplomások körében. Néhány európai uniós tagállamban kifejezetten intenzíven használták ezeket a helyeket a fiatal értelmiségiek. Ilyen államok például Görögország, Spanyolország, Litvánia, Észtország, Magyarország, Lengyelország és Szlovákia. A vizsgált internet-használói körben a kávézók soha nem tettek szert nagy népszerűsége Belgiumban, Hollandiában és Svédországban. 2013-ban kiugróan sok személy, - a frissdiplomások több mint 15%-a - internetezett még ezeken a helyeken is Bulgáriában, Magyarországon, Portugáliában és Szlovákiában, és elsősorban férfiak. A felsőfokú végzettséggel rendelkező teljes lakossági csoport körében is ezekben az országokban használják a legtöbben az internet kávézókat, de arányuk nem magasabb, mint 9%. Egyébként ez az internet-elérési lehetőség a teljes lakosságot vizsgálva is csak maximum 5-6%-os, és több EU tagállamban egyáltalán nem is mérhető a használat.

Otthonról a teljes uniós lakosság 72%-a használja az internetet. Az internetelérésnek ez a formája a legnépszerűbb a dánok, luxemburgiak, svédek és hollandok körében, a legkevésbé pedig a bolgár, görög, olasz és román lakosság körében. Ma az uniós háztartások szélessávú internet ellátottsága 76%-os. Olaszország háztartásai csak kis mértékben maradnak el ettől, az

említett további három tagállam lemaradása viszont jelentős. Az Olaszországhoz hasonló, 70% körüli otthoni internet-ellátottság más országokban magasabb otthoni internetezési hajlandóságot indukál.

A frissdiplomások körében az otthoni internetezés nagyon népszerű, uniós átlagban 98%-uk használja otthonról is a világhálót. 11%-uk vallotta azt, hogy csak otthonról használja az internetet. A diplomások teljes körét tekintve is nagyon magas az otthonról vagy otthonról is internetezők aránya, 92%. A férfiak és nők között nincs lényeges különbség ebben a mutatóban.

A munkahely mint az internetezés színhelye nagyon változatos képet tár elénk. 2013-ban az aktív munkavállalók 49%-ának volt internet elérési lehetősége a munkahelyén uniós átlagban. Dánia, Hollandia, Svédország és Finnország munkavállalóinak több mint 70%-a bír ezzel a lehetőséggel. A felsőfokú végzettségű munkavállalók már lényegesen nagyobb arányban, 62%-ban rendelkeznek munkahelyi internet-eléréssel, ugyanakkor a pályakezdő diplomásoknak csak 48%-a. Erre a jelenségre magyarázatot valószínűleg a munka világában kell keresnünk. Az idősebb diplomás munkavállalók nagyobb arányban töltenek be magasabb beosztást, amelyhez értelemszerűen több jogosultság tartozhat, így a szabadabb internet-hozzáférés is. A munkahelyek nagy része napjainkra rendezte és szabályozta a munkahelyi internet-hozzáférés kérdését. A frissdiplomások munkaszerződésük aláírásakor már a korábbi időszak gyakorlatához mérten sokkal redukáltabb munkahelyi internet-használatot fogadnak el, és tekintenek magukra érvényesnek. A tanulmány elején, az Y generáció jellemzőiként felsoroltak nagy részének helytállósága, mint például a szituációs tudatosság, konzervativizmus, erkölcsösség, igazolódni látszik a munkahelyeken is. Napjainkban egyetlen uniós tagállam sincs, ahol a fiatal diplomások kizárólagos internetezési helyszíneként a munkahelyet nevezték volna meg. A 2006-os, 2007-es statisztikai adatok még arról tanúskodnak, hogy abban az időben sok pályakezdő diplomásnak (sok országban akár 20% feletti arányban is) internetezési lehetősége csak a munkahelyén volt.

A képzési helyszínt internetezési szintéreként az aktív munkavállalóknak mindössze 5%-a nevezte meg. Azokban az EU tagállamokban, ahol hagyományosan kiterjedt a dolgozók továbbképzése, sokan nevezték meg a képzési helyszínt internet-hozzáférési pontként. Ezek az országok Dánia, Hollandia, Svédország, Finnország és az Egyesült Királyság.

A fiatal diplomásoknak nagy hányada, 44%-a használja az internetet valamilyen oktatási helyszínen. Az ő esetükben, az életkori határokat figyelembe véve, elsősorban az lehet a nagy arányszám magyarázata, hogy ugyan már diplomás felnőttek az alapképzést követően is, de sokuk folytatja a felsőfokú tanulmányait mester képzésben, és azt nem feltétlenül fejezik be 24 éves korukig.

A nyilvános könyvtárak internet-hozzáférési lehetősége az unió teljes lakosságának átlagát tekintve alacsony, illetve számított átlag nincs is, mivel több tagállamban nem mérik ezt a mutatót. A legmagasabb értékeket Dánia és Svédország érte el 2013-ban, mindkét országban a lakosság 9%-a internetezett könyvtárból is. A teljes diplomás lakosságot vizsgálva Dániában, Svédországban és Portugáliában a legnépszerűbb könyvtárban internetezni, 12-13%-uk teszi ezt.

A frissdiplomások körében a lakosság egészéhez mérten lényegesen népszerűbb internetezési helyszínek a nyilvános könyvtárak. Belgiumban, Észtországban, Cipruson, Litvániában, Hollandiában és Lengyelországban a fiatal értelmiségieknek mintegy 10%-a, Bulgáriában, Magyarországon és Szlovákiában kb. 20%-a, Portugáliában pedig 31%-a használta internetezésre a könyvtárakat.

Az internethasználó teljes uniós lakosság 32%-a csatlakozott az internethez laptop, notebook, netbook vagy táblagép segítségével. A frissdiplomások körében népszerűbbek ezek az eszközök, 52%-uk használja internet-csatlakozáshoz, de a legfrissebb adatok csökkenő tendenciát jeleznek.

Kézben hordozható készülékkel ma már többen csatlakoznak az internethez mint a hordozható számítógépekkel, a lakosság 38, a frissdiplomásoknak pedig a 66%-a a jelenleg hozzáférhető 2012-es adatok szerint. A 2013-as mérések alapján ma az uniós lakosság 57%-a, a fiatal értelmiségieknek pedig 88%-a használ vagy hordozható számítógépet vagy valamilyen kézben hordozható készüléket az internethez történő csatlakozáshoz. A bevezetőben említett, az Y generációt jellemző változásra való készség, technológiai érzék nyer igazolást az előbbiek leírtak alapján is.

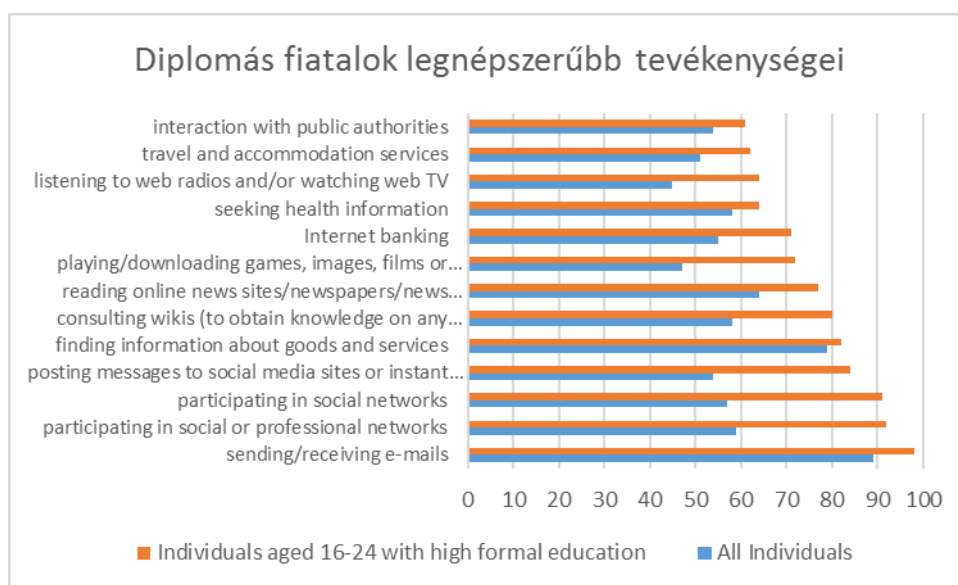
3. Élethelyzethez kapcsolódó internetes alkalmazások

Napjainkban sokan gondolják úgy, hogy a digitális írástudás vagy még inkább a digitális műveltség egyik fokmérője lehet, hogy ki, hányféle élethelyzethez kapcsolódó internetes alkalmazást használ, illetve képes annak használatára. Az Eurostat adatbázisaiban 2004-től kezdődően napjainkig találunk adatokat a különböző internet-alkalmazások használatára vonatkozóan a mindenkori tagállamok polgárainak önbevallása alapján. [3].

Az egyes internetes tevékenységek elemzésekor a frissdiplomások és a teljes lakosság adatait hasonlítom össze, esetenként a fiatal diplomások adatait a fiatalok teljes életkori csoportjának adataival. A vizsgálatban kizárólag olyan személyektől származó adatok szerepelnek, akik az adatfelvételt megelőző három hónapban használták az internetet, tehát nagy valószínűséggel rendszeres használónak tekinthetők. Az Eurostat napjainkban közel harmincféle, élethelyzethez kapcsolódó internetes tevékenységről gyűjt adatokat. A 3. ábrán a fiatal diplomások körében legkiterjedtebben használt tizenhárom tevékenység látható.

Az elektronikus levelezés a legkorábbi időktől kezdve, ami esetünkben ez 2004-et jelent, a legnépszerűbb internetes alkalmazás. Már 2004-ben is az internetet használó lakosság 81%-a küldött és fogadott e-maileket, 2013-ban pedig 89%-uk. A fiatal diplomásoknál ez az arány 98%, tehát teljes körben általános tevékenységnek tekinthetjük az elektronikus levelezést.

Az EU statisztikai hivatala külön-külön és együtt is megkérdezte a vizsgálatba vont személyeket, hogy tagjai-e valamilyen közösségi és/vagy szakmai hálónak.



3. ábra. Diplomás fiatalok legnépszerűbb internetes tevékenységei

A 3. ábra egyértelműen mutatja, hogy a közösségi vagy szakmai hálózati részvétel napjainkban még inkább a közösségi hálón való aktivitást jelenti a fiatal diplomások

esetében is. Körükben a szakmai hálózati tagság az általam vizsgált huszonhat tevékenység közül mindössze a huszonkettedik a sorban. (lásd 4. ábra) A közösségi hálóba való magas beavatottságból, behálózottságból egyértelműen következik, hogy az üzenetek posztolása is rendkívül népszerű köreikben, 84%. A teljes lakosság esetében ez még nem tartozik a legkiterjedtebben használt internetes tevékenységek csoportjába, de figyelemfelkeltő, hogy a 16-24 év közötti fiataloknak már 86%-a posztol rendszeresen.

Mind a teljes lakosság mind pedig a frissdiplomások körében is szokásos internetes tevékenység, hogy termékekről, szolgáltatásokról keresnek információt. Az elektronikus levelezéshez hasonlóan ez az internetes aktivitás is már az internethasználat kezdetétől töretlenül népszerű, és széles körben végzett.

Az Eurostat kérdése a wiki- rendszerű oldalak használatára irányult, amikor megfogalmazta a használókhoz intézett kérdését, miszerint használják-e a wikiket, ha valamilyen témában van szükségük ismeretekre, de valószínűleg a válaszadók szinte mindegyike a Wikipédiára gondolt. A Wikipédia indulása, 2001 óta egyre nagyobb népszerűségnek örvend, és az Eurostat adatsora is azt támasztja alá, hogy az egyetemekről, főiskolákról frissen kikerült fiatalok életében is nagy szerepet játszik. A frissdiplomások 80%-a használja ismeretszerzésre, míg a teljes lakoságnak az 58%-a. Az uniós állampolgárok közül a luxemburgiakat, finneket, németeket és svédeket tekinthetjük leginkább Wikipédia rajongóknak. Ugyanezen országok fiatal diplomása is nagy előszeretettel használják ezt az internetes lehetőséget, de felzárkóztak hozzájuk a dán, francia és holland kortársaik is.

Az internetes hírportálok, újságok, hírmagazinok olvasása a teljes lakosság körében is általános (64%), de a fiatal értelmiségiek még inkább élnek ezzel az informálódási lehetőséggel. (77%) A háztartások szélessávú internet-hozzáférésehez mérten viszonylag kevés francia (66%), belga (58%) és ír (52%) fiatal tájékozódik aktívan ezekből a hírforrásokból.

Az internetes bankolás évről-évre nagyobb népszerűségnek örvend az EU tagállamaiban. A 2004-es 35%-ról 2013-ra 56%-ra nőtt a használók aránya. A frissdiplomások közel háromnegyede (71%) bankol az interneten keresztül. Az észt, litván, holland, francia és osztrák fiatalok az átlagot jelentősen meghaladóan, a bolgár, görög, román fiatalok pedig jóval az alatt teljesítenek. A teljes magyar lakosságra még napjainkban sem jellemző az online bankolás, (36%) míg fiataljaink az uniós átlaghoz hasonló mértékben veszik igénybe azt.

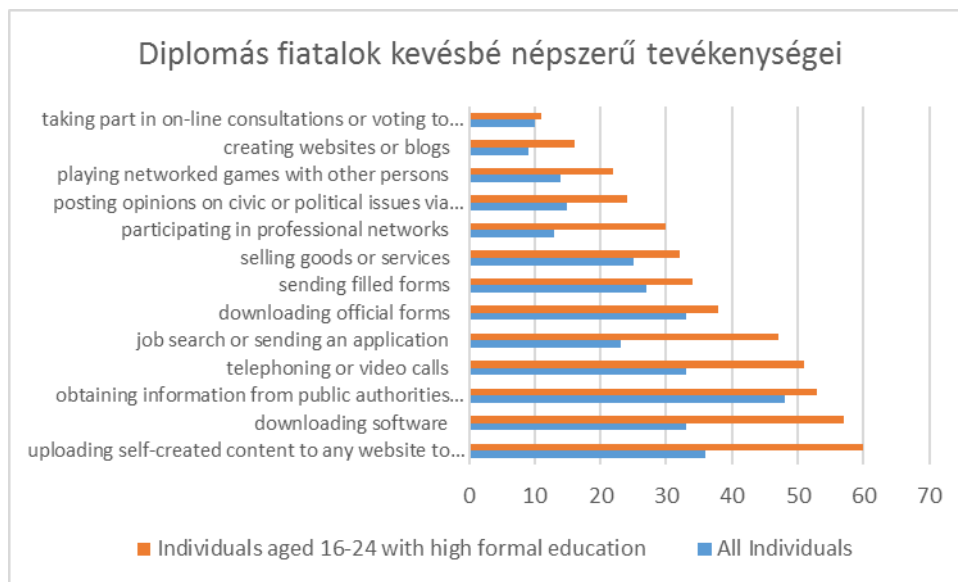
Az egészséggel kapcsolatos információk utáni kutakodás nagyon népszerű a teljes uniós lakosság körében, 2013-ban 58%-uk használta erre is az internetet. EU átlagot messze meghaladó mértékben foglalkoztatja az egészségük a finneket, szlovéneket, németeket, magyarokat és portugálokat. Azt gondolhatnánk, hogy a fiatal értelmiségi réteg számára ezek az információk még életkoruk miatt kevésbé relevánsak, de nagyot tévedünk. A frissdiplomások 64%-a bonyolít le egészséggel kapcsolatos internetes tevékenységeket, és különösen sokan teszik ezt a finn, magyar, német, spanyol, portugál és máltai fiatalok közül.

Web rádió hallgatás és web TV nézés az uniós lakosság 45%-ának, a fiatal diplomások 64%-ának viszonylag rendszeres és évről-évre egyre népszerűbb tevékenysége. A web médiumok használatában a svédek, hollandok, szlovének és a finnek járnak az élen a teljes lakosságot tekintve. Magyarország adatai átlagosak, a frissdiplomások körében kis mértékben meghaladja azt (67%), de a használat növekedésének tendenciája erőteljes.

Az utazási és szállás információkkal kapcsolatos internetes tevékenységek csökkenő, de legalábbis stagnáló aktivitást jeleznek mind a teljes lakosságot, mind pedig a fiatal értelmiségieket vizsgálva. Egyébként viszonylag népszerű tevékenységről van szó, hiszen a teljes lakosság 51, a frissdiplomások 62%-a él vele.

A hatóságokkal történő kommunikációban is az előzőekhez nagyon hasonló mértékben használja az internetet az uniós lakossága, 54 és 61%. A teljes lakosság vonatkozásában

Magyarország helyzete azonos az uniós átlaggal, viszont a fiatal diplomások közül kifejezetten sokan, 73%-uk használja az internetet a hivatalokkal történő kapcsolattartásban. Azok az élethelyzethez kötődő internetes alkalmazások, amelyeket a frissdiplomások 60%-a vagy annál kisebb hányada használ a kevésbé népszerű tevékenységek közé kerültek, és a 4. ábrán láthatóak.



4. ábra. Diplomás fiatalok kevésbé népszerű internetes tevékenységei

Természetesen a saját készítésű tartalmak feltöltése, a szoftverletöltés, a hivatali oldalakról történő információszerezés, az internetes telefonálás vagy videó hívás is fontos, a vizsgált csoport tagjainak több mint fele által végzett aktivitások.

Az internetes álláskeresés és az álláspályázatok internetes továbbításának lehetőségével a frissdiplomások 47%-a élt 2013-ban, a magyaroknak 63%-a. Élethelyzetükből adódóan ez természetes jelenségnek tekinthető.

Az elemzés fényt derített arra is, hogy melyek azok az aktivitások, amit nem szeretnek a fiatalok, és a 4. ábra tanúsága szerint általában a lakosság egésze sem. Nem szívesen osztják meg a véleményüket interneten keresztül állampolgári és politikai kérdésekben. Kifejezetten nem szeretnek online formában konzultálni vagy szavazni állampolgári vagy politikai kérdésekről, mint például várostervezés, petíciók támogatása. A magyar fiatalok az uniós átlaghoz képest is nagyon tartózkodnak ettől, 2%-uk él a lehetőséggel.

Irodalomjegyzék

- [1] Graeme Cordington: Understanding and connecting with people older and younger than yourself, *Tomorrowtoday* (2014) <http://www.tomorrowtoday.co.za/2014/06/09/keeping-generations-work/>
- [2] Eszenyiné Borbély Mária: [A digitális írástudás nemzetközi összehasonlítása](#), *Könyv és nevelés* **15** (2013), 20-35.
- [3] Eurostat http://epp.eurostat.ec.europa.eu/portal/page/portal/information_society/data/main_tables

Mit tudok informatikából?

What do I know in Informatics?

Csernoch Mária^a, Biró Piroska^b, Máth János^c és Abari Kálmán^d

^{a,b}Debreceni Egyetem, Informatikai Kar

csernoch.maria@inf.unideb.hu^a, biro.piroska@inf.unideb.hu^b

^{c,d}Debreceni Egyetem, Bölcsészettudományi Kar

math.janos@arts.unideb.hu^c, abari.kalman@arts.unideb.hu^d

Absztrakt: A 2011/2012-es tanévtől kezdődően teszteltük a Debreceni Egyetem Informatikai Karára beérkező elsőéves informatika szakos hallgatók informatikai ismereteit, terminológia használatát, algoritmikus készségét. A teszt eredményeit összehasonlítottuk a hallgatók informatikai érettségén szerzett pontjaival, a hallgatók önértékelésével, valamint általános és középiskolai informatikatanárok hallgatói teljesítmény értékelésével. A 2013/2014-es tanévben a vizsgálatainkat kiterjesztettük további három felsőoktatási intézményre (ELTE, NYF, EKF). A kutatásaink azt mutatják, hogy az elsőéves informatikushallgatók többsége nem rendelkezik megfelelő szintű algoritmikus készséggel és szegényes terminológia használat jellemzi. A hallgatók attitűdvizsgálatából egyértelművé vált, nincsenek felkészülve/felkészítve az Informatika Karon folyó magas szintű programozási ismeretek befogadására. Vizsgálataink eredményeként megfogalmazhatjuk továbbá, hogy a hallgatók előtérbe helyezik a felületi megközelítést szemben a mélyszerkezetű metakognitív problémamegoldó módszerekkel.

Kulcsszavak: algoritmikus készség, terminológia-használat, informatika didaktika, érettségi eredmények, önértékelés

Abstract: Since the 2011/2012 academic year we have been testing the first year students' knowledge in Informatics, their terminology usage and algorithmic skills at the University of Debrecen, Faculty of Informatics. The test results were compared to the students' results at the school leaving exams in Informatics, to their self-assessment values, as well as to the evaluation provided by the primary and secondary school teachers of Informatics concerning the students' results in the test. In the 2013/2014 academic year the project was further extended, and three higher education institutions (ELTE, NYF, EKF) joined it. Our research shows that most of the first year students of Informatics do not have the required algorithmic skills, and the majority provided evidence of their poor terminology usage. It is also clear from the results of our research that students rather put the emphasis on the surface-approach as opposed to the deep-approach metacognitive problem solving methods. From the attitude assessment of the students it became clear that they are not prepared for studying high level programming languages at the Faculty of Informatics.

Keywords: algorithmic skills, terminology usage, didactics of informatics, results of school leaving exam, self-assessment

Mottó: Miért leszek informatikushallgató?

Mit tudok informatikából?

Mit hiszek, hogy tudok informatikából?

Mit mutat az érettségi, hogy mit tudok informatikából?

1. Bevezetés

Az Algoritmikus és Alkalmazói Készségek Tesztelése (AéAKT Projekt – TAaAS Project – Testing Algorithmic and Application Skills) című projektet a 2011/2012-es tanév első félévében indítottunk, melynek célja az Informatikai Karon tanulmányaikat megkezdő hallgatók tesztelése [5]–[8],[11]–[14]. Arra kerestük a választ, hogy a hallgatók milyen szakmai ismeretekkel, milyen szintű algoritmikus készséggel rendelkeznek, hogyan értékelik saját tudásukat, továbbá, hogy mit jelent számukra az informatika, miért választották ezt a tudományterületet. A felmérést a következő két tanévben megismételtük a Debreceni Egyetemen, majd a 2013/2014-es tanévben további magyarországi felsőoktatási intézményeket sikerült bevonni a projektbe, ezáltal pontosabb képet kaphatunk az informatikushallgatók felkészültségéről.

2. Alkalmazói és Algoritmikus készségek tesztelése (AéAKT Projekt)

Minta

Az Alkalmazói és Algoritmikus készségek tesztelése című projektünket immár harmadik éve ismételtük meg az informatikushallgatók körében. A Debreceni Egyetem Informatikai Karán (DEIK) az elmúlt három tanévben összesen 950 Programtervező informatikus (PTI), Mérnök-informatikus (MI) és Gazdaságinformatikus (GI) vett részt a felmérésben (1. táblázat).

1. táblázat. A felmérésben résztvevő hallgatók száma a DE Informatikai Karán (DE_IK)

	PTI	MI	GI	Összesen
2011/2012	115	86	109	310
2012/2013	108	111	101	310
2013/2014	115	115	90	320
Összesen	338	312	300	950

A 2013/2014-es tanévben egy további magyarországi egyetemet és két főiskolát is sikerült bevonni a kutatásokba, ebben a tanévben 800 hallgató töltötte ki a tesztet. A felmérésben részt vett a Debreceni Egyetem mellett az Eötvös Lóránd Tudományegyetem (ELTE), a Nyíregyházi Főiskola (NYF) és az Eszterházy Károly Főiskola (EKF).

A teszt felépítése

A kutatásban használt teszt több részből épül fel, melyet az évek során folyamatosan javítottunk, fejlesztettünk. A hallgatók először egy Önértékelés lapot töltenek ki, amelyen meg kell jelölniük, hogy hogyan értékelik a tudásukat a felsorolt témakörökben. Ezen témakörök tesztelését végezzük az utána következő tesztlapon. A 2011/2012-es tanévben még nem volt önértékelési lap, ezt először 2012-ben töltöttük ki. Ebben a tanévben a hallgatóknak egy 6-fokozatú Likert-skálán kellett bejelölniük, hogy hogyan értékelik a tudásukat. Annak érdekében, hogy pontosabb eredményeket kapjunk a 2013/2014-es tanévben százalékos formában kértük a hallgatói önértékelést.

Az önértékelés után következik a teszt Alapinformációk része, ahol rákérdezőnk a hallgatók életkora mellett a számítógép használati szokásaikra, milyen IKT eszközökkel rendelkeznek,

tanultak-e programozást a középiskolában, milyen eredményt értek el az informatika és matematika érettségiben, informatika versenyeken, szereztek-e ECDL, informatikai OKJ okleveleket, milyen nyelveket beszélnek és mi a kedvenc hobbijuk.

Az alapinformációk után következtek az informatikai teszt feladatai, melynek az „Informatikai ismeretek” címet adtuk. A feladatokat a témakörök szerint a következő kategóriákba sorolhatjuk: billentyűzet- és egérhasználat, fájlkezelés és hálózati ismeretek, táblázatkezelés, szövegszerkesztés, logikai műveletek, pseudo kód megfejtése. A teszt kitöltését az „Attitűd vizsgálat” című rész zárja, ahol főként nyílt végű kérdéseket használva a hallgatóktól azt szeretnénk volna megtudni, hogy miért ezt a szakot választották, milyen területen szeretnének elhelyezkedni a tanulmányaik befejezése után, mit várnak az egyetemről és mit jelent számukra az informatika.

A teszt feladatai

Jelen dolgozatban a három tanévben elvégzett közös feladatok eredményeiről számolunk be: szövegszerkesztés (szg), táblázatkezelés (tbl), logikai operátorok (log) és pseudo kód megfejtése (kod), nagyobb hangsúlyt fektetve a logikai és pseudo kód megfejtése feladatok részletesebb eredményeinek bemutatására, a Debreceni Egyetem Informatikai Kar hallgatói körében.

	A	B	C	D	E
1	ország	foldresz	fovaros	terulet	fo_ezer_02
2	Afghanistan	Asia	Kabul	647500	27756
3	Albania	Europe	Tirana	28748	3545
4	Algeria	Africa	Algiers	2381740	32278
5	American Samoa	Oceania	Pago Pago	199	69
6	Andorra	Europe	Andorra la Vella	468	68
7	Angola	Africa	Luanda	1246700	10593
8	Anguilla	Amerika	The Valley	102	12

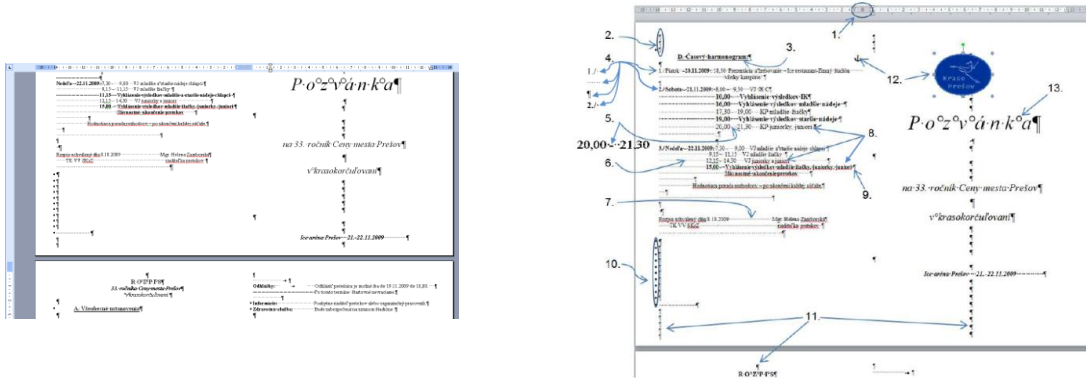
Az a)–d) és f) feladatokat Excel függvények felhasználásával oldja meg! A táblázat utolsó sora a 216. sor.

- a) Írassa ki a legnagyobb területű ország fővárosát
- b) Írassa ki az egyes országok népsűrűségét
- c) Írassa ki az afrikai országok számát
- d) Írassa ki azon országok átlagos területét, amelyek lakossága kisebb, mint egy, a H1 cellában tárolt érték
- e) Mit csinál az alábbi összetett Excel képlet: `=SZUM(HA(B2:B216="Europe";HA(BAL(A2:A216)="A";1)))` ?
- f) Írassa ki azon országok számát, amelyek területe nagyobb, mint egy, a G1 cellában tárolt érték

1. ábra. A táblázatkezelői (tbl) feladatok (jobb) és a feladathoz tartozó minta táblázat (bal)

A táblázatkezelői feladatok esetében a megadott mintatáblázat alapján kellett a hallgatóknak válaszolni a kérdésekre (1. ábra). Az a)–d) és f) feladatokban a hallgatóktól egy képletet vártunk, míg az e) pontban a megadott képlet megfejtése volt a feladat. Az e) pontban egy tömbképletet adtunk, ezáltal szeretnénk volna mérni, hogy ki ismeri fel, ki tanult már róla. Továbbá, aki ismeri a tömbképleteket azoknak nagy segítséget nyújthatott a c), d) és f) feladatok megoldásánál is. Úgy gondoljuk, hogy a tömbképletek ismerete [4], [5], [8], [13], a többszintű (multilevel), beágyazott képletek építése, a táblázatkezelés algoritmikus szemlélettel való tanítása sokkal hatékonyabb, mint az elterjedt, ám felületi metakognitív megközelítések. Az általunk kifejlesztett, korábbi elméletek ötvözésével született [25]–[27], [28], [29], [30], [33]–[35], CAAD-alapú (Computer Algorithmic And Debuging based) metakognitív módszer lényegesen eredményesebbnek bizonyult [6], [7], [11], [12], [14], mint a helyenként kritizált [19], [24], ám elterjedt TAEW (Trial-And-Error Wizard based) módszer. Az f) kérdés, ami a c) feladat általánosítása először a 2013/2014-es tanévben szerepelt a feladatsorban, annak tesztelésére, hogy mennyire tudják a hallgatók a változókat és az egyenlőtlenségeket kezelni a feltételes feladatokban.

A gyakorlati táblázatkezelési feladatokhoz egy elméleti jellegű feladatot is adtunk, amelyben a hallgatóknak a szigorú megköötésekkel használható FKERES(), VKERES() és az általános célú INDEX(HOL.VAN()) kétszintű függvényeket kellett összehasonlítaniuk.



2. ábra. Szövegszerkesztési (szg) feladatok mintaszövege a 2011/2012-es tanévben (bal), majd az azt követő tanévekben (jobb)

A szövegszerkesztés feladat az évek során a folyamatos fejlesztés következtében módosult. A 2011-es felmérésben csak a nemnyomtatódó karaktereket is tartalmazó minta szerepelt (2. ábra, bal), viszont az eredmények láttán próbáltunk segíteni a hallgatóknak, és egy új verziót készítettünk, ahol már bejelöltük mely hibák felismerését várjuk. Ezt használtuk a 2012/2013-as és a 2013/2014-es tanévben (2. ábra, jobb).

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.
helyes													
tördelési hiba													
formázási hiba													
szintaktikai hiba													
szemantikai hiba													
tipográfiai hiba													

3. ábra. A szövegszerkesztési (szg) feladat 2012/2013-as (bal) és a 2013/2014-es (jobb) értékelő lapja

A szövegszerkesztés feladathoz 2012/2013-as tanévtől kezdődően egy értékelő lapot is készítettünk, amelyen az egyes hibák típusát kellett megjelölnék a hallgatók (3. ábra, bal), valamint a 2013-ban azt is kértük, hogy a hibát is nevezzék meg (3. ábra, jobb).

Egy játékban két kártyát (X, Y) húzunk egy-egy pakliból. Mindkét pakliban található A-betűt, B-betűt és 0-s számjegyet tartalmazó kártyák. Add meg, hogy az alábbi algoritmus alapján milyen esetekben hány pontot kaphatunk!

	X	Y	Pont
	A	A	
	A	B	
	A	0	
	B	A	
	B	B	
	B	0	
	0	A	
	0	B	
	0	0	

V:=X="A" vagy Y="A"
W:=X="B" vagy Y="B"
Ha V és W akkor Pont:=3
különben Ha V akkor Pont:=2
különben Ha W akkor Pont:=1
különben Pont:=0

Mit csinálnak az alábbi programrészletek? Fogalmazza meg általánosan is, hogy a DB és az M változó mit tartalmaz!

N=50, a mért értékek:
500,500,500,500,500,600,600,650,700,750,820,880,930,1010,1050,980,930,830,780,720,710,700,
750,770,790,820,880,880,820,760,740,600,500,560,670,770,770,760,750,740,740,730,720.

DB:=0
a) Ciklus i=1-től N-ig
Ha X(i)>800 akkor DB:=DB+1
Ciklus vége

DB:=0
b) Ciklus i=2-től N-1-ig
Ha X(i)<X(i-1) és X(i)<X(i+1) akkor DB:=DB+1
Ciklus vége

M:=0
c) Ciklus i=2-től N-ig
Ha X(i)-X(i-1)>M akkor M:=X(i)-X(i-1)
Ciklus vége

4. ábra. A logikai (log) műveletek (bal) és pseudo kód (kod) megfejtése feladatok (jobb) [9], [10]

A logikai és a pseudo kód feladatait a Nemes Tihamér Országos Informatikai Tanulmányi Verseny – Programozás kategória, 2008/2009-es tanév, 1. forduló, 1. korcsoport (5–8. osztály) feladatsorából válogattuk [9], [10]. A pseudo kód megfejtése feladatoknál, azt vártuk a hallgatóktól, hogy fogalmazzák meg egy-egy egyszerű mondatban az algoritmusok lényegét. A programozási feladatok értékeléséhez a megértés különböző SOLO-kategóriáit használtuk [34]–[40]. A pseudo kód megfejtése feladatnál lényegesen egyszerűbb a logikai operátorok feladat, mivel ebben az esetben csak a négy lehetséges kimeneti érték közül kellett választani.

3. Hipotézisek

[H1] A programtervező informatikus szakra (PTI) jelentkező hallgatók teljesítenek a legjobban, majd ezt követően a mérnök informatikus (MI) és utána gazdaságinformatikus (GI) hallgatók.

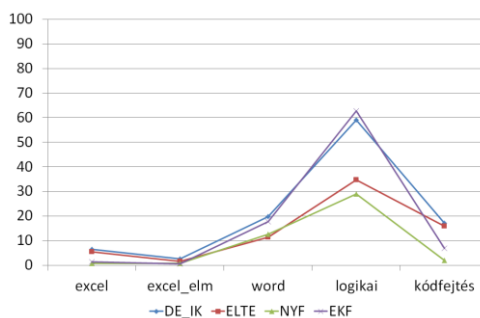
[H2] A hallgatók eredménye a 2013/2014-es tanévben jobb, mint a korábbi években, mivel a 2008-as kerettantervben megfogalmazottak alapján, nőtt az informatika órák száma a középiskolában [1], [2].

[H3] Magasabb érettségi eredmények magasabb önértékeléshez vezetnek, tehát az érettségi eredmények hatással vannak az önértékelésre.

[H4] A hallgatók jobb érettségi eredményei magasabb szintű algoritmikus készséget jelentenek, tehát jobban teljesítenek az egyes feladatokban.

4. Eredmények

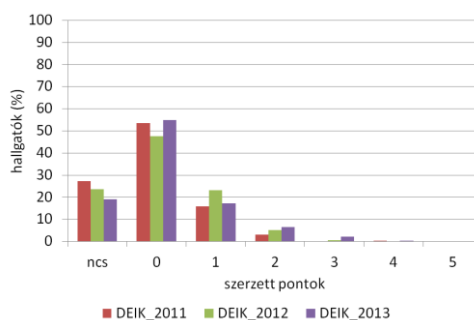
A 2011–2013 közötti évek felméréseinek eredményei



5. ábra. A 2013/2014-es felmérés intézményenkénti átlagos eredményei

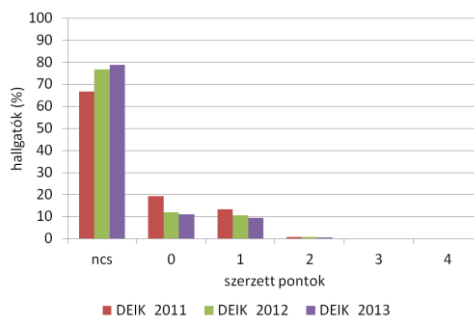
A 2013-as tanév eredményeit mutatja az 5. ábra, amely felmérésben a Debreceni Egyetem (DE_İK) mellett részt vett az ELTE, valamint az egri és a nyíregyházi főiskola (EKF, NYF) is. Az 5. ábra adatai alapján egyértelmű, hogy a hallgatók valamennyi intézményben nagyon gyenge teljesítményt értek el.

A logikai feladat volt a legnépszerűbb a hallgatók körében, ami a feladat jellegéből természetes módon adódik. Ennél a feladatnál érték el a legjobb eredményeket a vizsgált intézményekben (5. ábra) és a különböző tanévekben is (9. ábra). A logikai feladat után a szövegszerkesztés feladat eredményei jönnek, ezt követően a kódfejtés, amely már programozási ismereteket is igényelt, majd a sort a táblázatkezelés eredmények zárják (5. ábra).



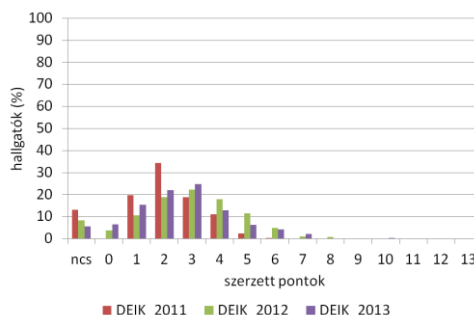
6. ábra. Táblázatkezelési feladat megoldása a DEIK három egymást követő évfolyamán

Az 'ncs' rövidítés az ábrákon a 'nem csinálta' – nem foglalkozott a feladattal – kategóriát jelenti (6–10. ábrák). A táblázatkezelési eredmények mutatják, hogy a leggyakoribb eredmény a 0 érték, mely arra utal, hogy a hallgató próbálkozott a feladattal, de nem sikerült megoldania (6. és 7. ábrák). Az 'ncs' és a 0 értékek száma lényegesen meghaladja az értékelhető megoldások számát. Ez az eredmény mindenképpen meglepő, mivel az informatikai érettséginek szerves része a táblázatkezelési ismeretek közép- és emelt szinten is, ugyanakkor összhangban van azokkal a mérésekkel, amelyek azt bizonyítják, hogy a táblázatkezelői dokumentumok hemzsegnak a hibáktól [19], [20]–[22], [17], [23].



7. ábra. Elméleti jellegű táblázatkezelési feladat a DEIK három egymást követő évfolyamán

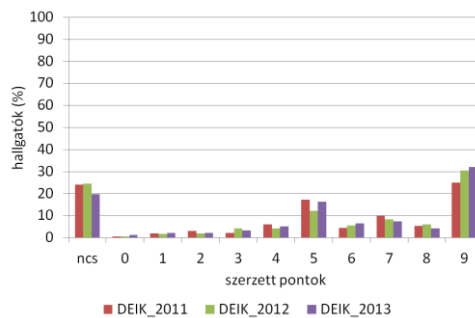
Az elméleti táblázatkezelési feladat megoldása során kapott eredmények nagyon meglepőek, mivel a legtöbb hallgató nem is foglalkozott a feladattal. A hallgatók valamivel több, mint 10%-a próbálkozott sikertelenül és további 10%-ának sikerült elérni 1 pontot a lehetséges 4 pontból. Ebben a feladatban teljesítettek a leggyengébben.



8. ábra. Szövegszerkesztési feladat megoldása a DEIK három egymást követő évfolyamán

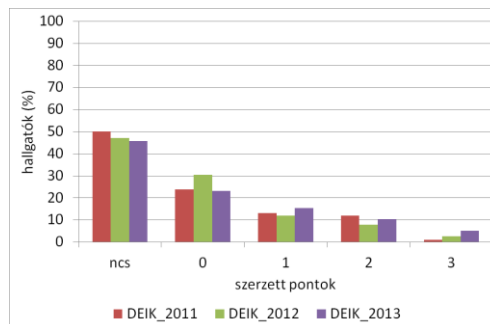
A szövegszerkesztési feladatot próbálták a legtöbben megoldani, ami egyenes következménye lehet a szövegszerkesztő programok népszerűségének. Ennek ellenére a jó megoldások száma rendkívül alacsony. A 13 részfeladatban a legnagyobb gyakorisággal 1–3 helyes megoldás érkezett a különböző tanévekben. A megoldások részletes elemzése alapján egyértelmű, hogy csak a legkirívóbb enter és szóköz hibákat képesek a hallgatók megtalálni, ezek közül sem

mindegyiket. További hibák felismerése azonban csak rendkívül alacsony határfokkal történik.



9. ábra. A logikai feladat megoldása a DEIK három egymást követő évfolyamán

A logikai feladat megoldása volt a legsikeresebb a hallgatók körében, de továbbra is magas azoknak a száma, akik nem foglalkoztak a feladattal. Nem szabad azonban elfelejtenünk, hogy ez volt az a feladat, ahol a megoldást mindösszesen négy lehetséges output közül kellett kiválasztani, tehát mint ilyen ez volt a teszt legkönnyebb feladata.



10. ábra. Programozási feladat megoldása a DEIK három egymást követő évfolyamán

Az önértékelések alapján egyértelmű, hogy a pseudo kód megfejtésével nagyon sokan azért nem foglalkoztak, mert nem tanultak programozást, ami meglepő, mivel a programozás része az utóbbi két évtized tanterveinek [1]–[3]. A feladat leírásánál említettük, hogy ezek általános iskolás első fordulás verseny feladatok [9], [10], tehát minimális programozói ismeret is elegendő lett volna a megoldásukhoz. Mindezeket figyelembe véve, rendkívül magas azon hallgatók száma, akik nem foglalkoztak a feladattal. Nagyon magas továbbá azon hallgatók száma is, akik ugyan próbálkoztak a feladattal, de nem tudták megoldani. Az 1, 2 és 3 helyes megoldást adó hallgatók száma folyamatosan csökken. Alig akadt hallgató, aki mindhárom részfeladatot meg tudta oldani, annak ellenére, hogy a DEIK hallgatók közel 80%-os, illetve 70%-os teljesítménnyel tettek közép- és emelt szintű érettségi vizsgát informatikából [15], [16].

Statisztikai elemzések – a DEIK hallgatóinak eredményei 2011-2013 között

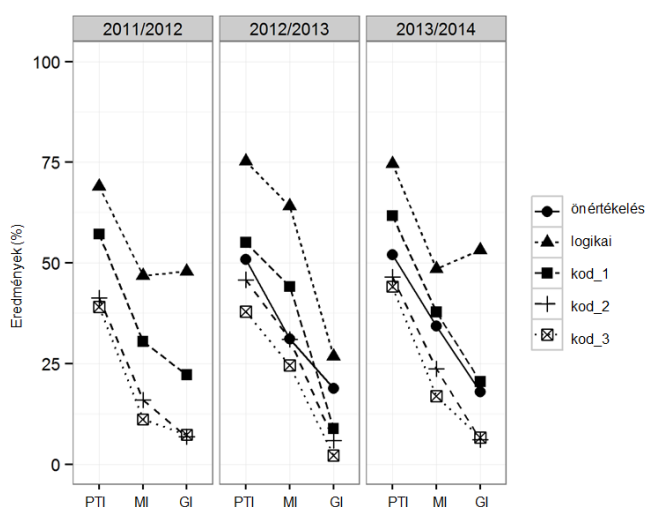
A továbbiakban a logikai és pseudo kód megfejtése feladatok megoldásainak eredményeiről számolunk be, a hallgatók önértékelését és érettségi eredményeit figyelembe véve.

Önértékelés

Megvizsgáltuk, hogy az önértékelés és a feladatokban nyújtott teljesítmény az egyes szakokon és az egyes években hogyan alakult. A feladatok nehézségi sorrendje a legkönnyebbtől kezd-

ve a legnehezebb felé haladva logikai, kod_1, kod_2, kod_3 (az eltérés a Friedman teszt szerint szignifikáns, $p < 0,001$). Ezen túlmenően, a logikai feladat szignifikánsan könnyebbnek bizonyult a másik három feladatnál (Mann-Whitney tesztek, $p < 0,001$) (11. ábra). Az ábráról a szakok sorrendje is leolvasható: legjobban a PTI teljesített, majd az MI és a GI, ezáltal beigazolódtott a [H1] hipotézisünk.

Ha az éveket összevontan kezeljük, akkor mindegyik részfeladat esetében szignifikánsan csökken a teljesítmény a szakok fenti sorrendjében (Jonckheere-Terpstra Test: $p < 0,001$ mindegyik részfeladatban). Az évek összehasonlításában szerény teljesítményemelkedést figyelhetünk meg minden részfeladat esetén, de ez a hatás nem szignifikáns, ha a kérdéses 2013/2014-es eredményt hasonlítjuk össze a megelőző két év átlagával (Mann-Whitney próbákkal a legkisebb $p = 0,201$). A [H2] hipotézis nem teljesült, mivel csak kis mértékű teljesítményemelkedés figyelhető meg, amely nem szignifikáns. Talán a következő évek eredményei alátámasztják a feltevéseinket.



11. ábra. Az önértékelés és az egyes feladatokban nyújtott teljesítmény

Hallgatói klaszterek kialakítása

A legsikeresebben megoldott feladat válaszaiból (logikai feladat, 9 változó 4 lehetséges értékkel: 0, 1, 2, 3), mint kategorikus változókból kiindulva klaszteranalízist végeztünk SPSS-ben (Klasztering of categorical data, similarity measures, log-likelihood measure, Two-step Klaszter Analysis in SPSS). A klaszterek létrehozásánál csak azon hallgatókat vettük figyelembe, akik mind a 9 részfeladatra adtak valamilyen választ. Azon hallgatók, akik nem vagy csak részlegesen oldották meg a logikai feladatot 0%-os megoldással szerepelnek az átlagban, ők alkotják a negyedik klasztert. A teljes megoldásokat figyelembe véve három klasztert találunk, melyek mindegyike tisztán értelmezhető – Klaszter 1–3 (2. táblázat).

A további elemzések során figyelembe vettük a hallgatók saját programozási ismeretére vonatkozó becslését is, a „Hogyan értékelné ismereteit a programozás témakörben?” kérdésre adott válaszokat. A 2. táblázat a logikai és a kódfejtés kérdésekre adott válaszok átlagait hasonlítja össze az értékelés pontjaival és az érettségi eredményekkel a négy klaszterben. Általánosan megfogalmazható, hogy a hallgatók az önértékelésüknél alacsonyabb teljesítményt értek el az egyes feladatokban. Ez alól kivétel a logikai feladat.

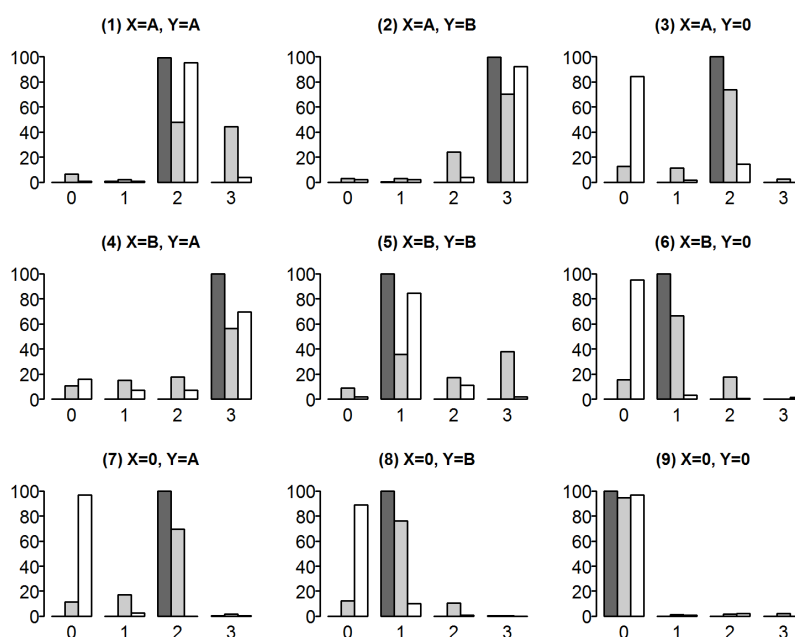
A logikai feladat eredményeit vizsgálva tapasztaltuk, hogy a Klaszter 1 azokat a hallgatókat tartalmazza, akik mind a 9 részfeladatban jó teljesítményt nyújtottak (99,7%, 2. táblázat „log” sor). A Klaszter 2-ben már megnő a hibás válaszok száma, de még mindig a helyes válaszok vannak túlsúlyban (65,6%). A Klaszter 3 hallgatói mindazon kérdésekre jól válaszolnak, amelyekben nincs 0 a részfeladat bemenő adatai között, ahol viszont van, ott ők rendszerint a

0 választ jelölik meg az algoritmus eredményeként. Véleményünk szerint úgy értelmezik az algoritmus által megfogalmazott szabályt, hogy az csak az A és B bemenő értékekre írja le a teendőket, de ha 0 is megjelenik bemenő adatként, akkor számukra az utolsó, „különböző” opció az irányadó. Ennek oka az lehet, hogy nincs a nullát tartalmazó, explicit feltétel az algoritmusban, az csak implicit módon, nagyobb erőfeszítéssel olvasható ki a feladatból.

2. táblázat. Hallgatói klaszterek teljesítménye a feladatokban

	Klaszter 1	Klaszter 2	Klaszter 3	Klaszter 4
N	279	168	195	308
önértékelés (%)	46,9	39,1	31,9	23,3
log (%)	99,8	65,8	52,1	0,0
kod_1 (%)	57,3	43,5	38,3	19,2
kod_2 (%)	46,1	27,5	19,7	10,3
kod_3 (%)	38,8	24,7	17,3	8,4
emelt szintű érettségi (%)	74,6 (N=85)	67,2 (N=35)	65,3 (N=27)	63,6 (N=33)
középszintű érettségi (%)	84,1 (N=154)	82,2 (N=110)	81,5 (N=135)	81,5 (N=206)

A fent leírtakat szemlélteti a 12. ábra, amely a 9 részfeladatra adott válaszok klaszterenkénti mintaszámainak százalékos arányát mutatja. Az egyes ábrák felirata az X és Y bemenő paraméterek értékeit jelentik. A helyes választ mindegyik részfeladat esetén pontosan megmutatja a Klaszter 1 (sötét szürke) oszlop helye, ugyanis ők válaszoltak helyesen a legnagyobb arányban. Többnyire vele együtt szerepel a Klaszter 2 (világos szürke) oszlop, és a Klaszter 3 (fehér) oszlop is A és B bemenő értékek esetén. Amint a bemenő értékek között megjelenik a 0 (X=0 és/vagy Y=0), akkor a Klaszter 3 válasza 0 lesz.



12. ábra. Klaszter 1 (sötét szürke), Klaszter 2 (világos szürke) és a Klaszter 3 (fehér) oszlopok

A 3+1 klaszter viszonyát megvizsgáltuk az év és a szak változóval. Azt találtuk, hogy a három változóra (év, szak, klaszter) illeszkedik a szak×klaszter év loglineáris modell ($\chi^2(2)=16,6$, $p=0,16$). Ez azt jelenti, hogy a klaszternek csak a szakhoz van köze. E kapcsolat lényege a következő keresztábrán látható (3. táblázat).

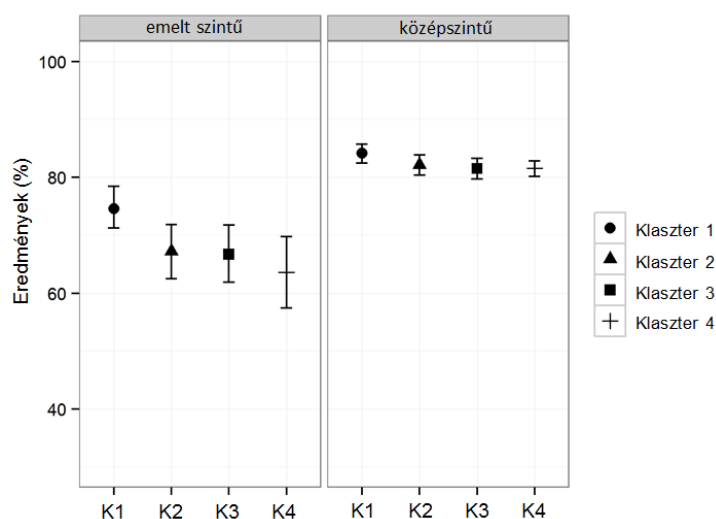
3. táblázat. Szakok és a klaszterek közötti kapcsolat

		Klaszter 1	Klaszter 2	Klaszter 3	Klaszter 4
GI	N	47	42	66	145
	%	15,7	14,0	22,0	48,3
MI	N	78	65	67	102
	%	25,0	20,8	21,5	32,7
PTI	N	154	61	62	61
	%	45,6	18,0	18,3	18,0
Összesen:	N	279	168	195	308
	%	29,4	17,7	20,5	32,4

A Klaszter 2 és a Klaszter 3-hoz tartozó hallgatók aránya hasonló mindhárom szakon. Számottevő különbség a Klaszter 1 és a Klaszter 4 hallgatóinak arányában van az egyes szakokon (Khi-négyzet próba: $\chi^2(6)=101,7$, $p<0,001$). A loglineáris elemzés azt is mutatja, hogy az évfolyam és a klaszter között nincs kapcsolat, tehát a kapott négy klaszter mindhárom évfolyamban hasonló arányban van jelen.

Klaszterek és érettségi eredmények

Az érettségi eredmények is mutatnak különbséget a klaszterek között: a Klaszter 1 hallgatói mindkét szinten szignifikánsan jobbak, mint a másik három klaszter, de ez az eltérés nagyobb az emelt szint esetén (Kruskal-Wallis próba emelt szinten: $\chi^2(3)=14,7$, $p=0,002$, közép szinten Kruskal-Wallis próba: $\chi^2(3)=8,7$, $p=0,033$) (13. ábra). Azonban a Klaszter 2, 3, 4 között szignifikáns eltérést egyik szinten sem találtunk.



13. ábra. Klaszterek és az érettségi eredmények

Az érettségi teljesítményben tehát nem különül el a három utolsó klaszter. A három féle „gyengeség” között az érettségi feladatok – mivel mindhárom hibákhoz vezethet – nem tudnak különbséget tenni.

Érettségi eredmények, önértékelés, teszt eredmények

Összehasonlítottuk a 2012/2013 és 2013/2014-es évfolyamokban a hallgatók önértékelését az érettségi eredményükkel [H3 hipotézis].

Emelt szinten közepes pozitív korrelációt mértünk (Spearman's $\rho_{2012/2013}=0,36$, $p_{2012/2013}=0,005$; $\rho_{2013/2014}=0,52$, $p_{2013/2014}<0,001$), míg középszinten, gyenge vagy elhanyagolható pozitív kapcsolatot találtunk (Spearman's $\rho_{2012/2013}=0,16$, $p_{2012/2013}=0,031$; $\rho_{2013/2014}=0,09$, $p_{2013/2014}=0,208$). A vizsgálatokhoz a Spearman-féle féle rangkorrelációs együtthatót használtuk. Ez azt jelenti, hogy a hallgatók az érettségi eredményeket nem tekintik mérvadónak saját teljesítményük, tudásuk megítélésében. Az emelt szintű eredményeket még valamelyest figyelembe veszik, de a középszintű érettségi eredmények hatása elhanyagolható.

Megvizsgáltuk továbbá, hogy az érettségi eredmények milyen mértékben befolyásolják a hallgatók az egyes feladatokban elért teljesítményét [H4 hipotézis].

Az emelt szintű érettségi esetén az egyes feladatokban elért teljesítménnyel közepes korrelációt mértünk (Spearman's $\rho_{\text{task } 1}=0,26$, $p_{\text{task } 1}<0,001$; $\rho_{\text{task } 2}=0,42$, $p_{\text{task } 2}<0,001$; $\rho_{\text{task } 2.1}=0,38$, $p_{\text{task } 2.2}<0,001$; $\rho_{\text{task } 2.2}=0,31$, $p_{\text{task } 2.2}<0,001$; $\rho_{\text{task } 2.3}=0,37$, $p_{\text{task } 2.3}<0,001$). Középszintű érettségi esetén a kapcsolat erőssége csökkent, csak gyenge hatást tudtunk kimutatni (Spearman's $\rho_{\text{task } 1}=0,09$, $p_{\text{task } 1}=0,022$; $\rho_{\text{task } 2}=0,22$, $p_{\text{task } 2}<0,001$; $\rho_{\text{task } 2.1}=0,23$, $p_{\text{task } 2.2}<0,001$; $\rho_{\text{task } 2.2}=0,20$, $p_{\text{task } 2.2}<0,001$; $\rho_{\text{task } 2.3}=0,17$, $p_{\text{task } 2.3}<0,001$). A vizsgálatokhoz a Spearman-féle féle rangkorrelációs együtthatót használtuk.

5. Összefoglalás

A 2011/2012-es tanévtől kezdődően az 'Algoritmikus és Alkalmazói Készségek Tesztelése' című projekt keretein belül teszteljük az elsőéves informatikushallgatók informatikai ismereteit, algoritmikus készségét, terminológia-használatát. A projekt keretein belül adatokat gyűjtünk a hallgatók informatika érettségi eredményeiről és önértékeléséről is, amelyeket összehasonlítjuk a tesztben elért eredményekkel. A felmérésben a 2011/2012-es tanévtől kezdődően részt vesz a Debreceni Egyetem Informatikai Kara, melyhez a 2013/2014-es tanévben csatlakozott az Eötvös Loránd Tudományegyetem, a Nyíregyházi Főiskola, valamint az Eszterházy Károly Főiskola, Eger.

Az informatika szakokon tanulmányaikat megkezdő hallgatók tesztelése azt mutatja, hogy a hallgatók rendkívül alacsony szintű algoritmikus készséggel, alkalmazói ismeretekkel rendelkeznek és nem tudják gondolataikat a szakma nyelvén megfogalmazni. A feladatok többségét el se tudják kezdeni és e mellett nagyon magas azon hallgatók száma, akik ugyan próbálkoznak a feladatok megoldásával, de teljesítményük értékelhetetlen.

A hallgatói eredmények statisztikai elemzése azt mutatja, hogy a hallgatók tudásuk alapján négy csoportba sorolhatóak. A legjobb és a leggyengébb teljesítményt elért csoportok között további két csoport helyezkedik el. Az egyik köztes csoportra a teljesen kiszámíthatatlan tudás jellemző. A másik köztes csoportról az elemzések alapján elmondható, hogy egy limitált tudással rendelkeznek, ami olyan formában jelenik meg, hogy egy bizonyos szintig megbízhatóan teljesítenek, utána viszont saját maguk által megfogalmazott menekülési útvonalakat választva próbálják a feladatokat megoldani.

A teszt eredményeit összehasonlítottuk a hallgatók informatika érettségi eredményeivel és önértékelésével. A kapott elemzések azt igazolják, hogy az érettségi eredmények, különösen

igaz ez a középszintű érettségi esetén, nincsenek hatással a hallgatók tesztben nyújtott teljesítményére. A legjobban teljesítő hallgatók esetén egy közepes kapcsolat mérhető a teszt és az emelt szintű érettségi eredményei között, de a gyenge teljesítményű csoportok között az érettségi nem tud különbséget tenni.

A hallgatói önértékelések vizsgálata során bizonyításra került a már korábban más tudományterületeken is tapasztalt negatív korreláció a hallgatói nem tudás és az önértékelés között [18]. A legjobban teljesítő csoport tudta leginkább megítélni tudását, míg a leggyengébben teljesítőknél volt a legmagasabb a felülbecslés. Az önértékelés és az érettségi eredmények összehasonlítása, a várakozásainkkal ellentétben, azt mutatja, hogy a hallgatók nem tekintik a középszintű érettségit a tudásuk mérésére alkalmas mutatónak, míg az emelt szintű érettségi és az önértékelés között egy közepes kapcsolat kimutatható. A vizsgálataink tehát a hallgatói véleményeket erősítették meg.

Az évek előrehaladtával egy gyenge emelkedés tapasztalható a hallgatók felmérésben nyújtott teljesítményében, azonban a 2013/2014-es tanév nem hozta meg a várva várt teljesítményemelkedést. A 2013/2014-es tanév kiemelt fontosságú abból a szempontból, hogy ekkor érkeztek/érkezhetnek először olyan hallgatók a felsőoktatásba, akik 2009-től a megemelt óraszámú informatika tantervek szerint tanulhattak középiskolában. A teljesítménynövekedés elmaradásának azonban több magyarázata is lehet: a 2013/2014-es tanévben felsőfokú tanulmányaikat megkezdő hallgatók nem feltétlenül 2009-ben kezdték középiskolai tanulmányaikat, hanem korábban, az informatikushallgatók között nagy számban találni olyanokat, akik már korábban is emelt óraszámú tanulták az informatikát, tehát a 2009-es óraszámemelés tekintetben nem hozott változást, és természetesen azt sem zárhatjuk ki, hogy az óraszám emelkedés önmagában nem elegendő a hallgatói tudásszint emelkedéséhez. Mindezeket figyelembe véve további mérésekre lesz szükségünk a hallgatói teljesítmények és az előző tanulmányaik összehasonlításához.

6. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

A kutatást részben támogatta az OTKA K-105262 számú projekt.

Irodalomjegyzék

- [1] Kerettanterv 2008. Magyar Közlöny, 20. szám II. kötet, 2008. február 8.
- [2] Kerettanterv 2013. <http://kerettanterv.ofi.hu/index.html>. Letöltés dátuma: 2013. 08. 30.
- [3] NAT 1995. 130/1995. (X. 26.) Korm. rendelet a Nemzeti alaptanterv kiadásáról. http://net.jogtar.hu/jr/gen/hjegy_doc.cgi?docid=99500130.KOR. Letöltés dátuma: 2013. 08. 30.
- [4] M. Csernoch és L. Balogh: Algoritmusok és táblázatkezelés. Tehetség gondozás a közoktatásban az informatika terén. *Magyar Tehetségsegítő Szervezetek Szövetsége*, Budapest, (2011).
- [5] M. Csernoch és P. Biró: Spreadsheet misconceptions, spreadsheet errors, *Oktatáskutatás határon innen és túl. HERA Évkönyvek I.*, ed. Juhász Erika, Kozma Tamás, Publisher: Belvedere Meridionale, Szeged, (2014), 370–395.

- [6] M. Csernoch és P. Biró: Teachers' Assessment and Students' Self-Assessment on The Students' Spreadsheet Knowledge. *EDULEARN13 Proceedings* July 1st-3rd, 2013 — Barcelona, Spain. Publisher: IATED, (2013) 949–956.
- [7] M. Csernoch és P. Biró: Button-up technikák hatékonyságának vizsgálata informatika szakos hallgatók táblázatkezelés-oktatásában. szerk. T. Kozma és I. Perjés, *Új kutatások a neveléstudományokban 2012*, ELTE Eötvös kiadó, (2013), 352–369.
- [8] M. Csernoch: Introducing Conditional Array Formulas in Spreadsheet Classes. *EDULEARN12 Proceedings*. Barcelona, Spain. 2-4 July, 2012. Publisher: IATED, (2012) 7270–7279.
- [9] Nemes Tihamér Országos Informatikai Verseny 2008-2009 I. forduló, I. korcsoport, 5-8. osztály. Feladatok. <http://nemes.inf.elte.hu/2009/nt09-1f1.doc>. Letöltés dátuma: 2013. 08. 30.
- [10] Nemes Tihamér Országos Informatikai Verseny, I. forduló, I. korcsoport, 5-8. osztály. Megoldások) in Hungarian. <http://nemes.inf.elte.hu/2009/nt09-1m1.doc>. Letöltés dátuma: 2013. 08. 30.
- [11] P. Biró és M. Csernoch: Deep and surface structural metacognitive abilities of the first year students of Informatics. *4th IEEE International Conference on Cognitive Informatics*, Proceedings, Budapest, (2013), 521–526.
- [12] P. Biró és M. Csernoch: Programming skills of the first year students of Informatics. *XXIII. International Conference on Computer Science 2013*, EMT, in Hungarian, (2013). 154–159.
- [13] P. Biró és M. Csernoch: Táblázatkezelés algoritmikus megközelítése. *Kiss Árpád Emlékkonferencia Tanulmánykötete 2013*, Debrecen, (2013). Elfogadva.
- [14] P. Biró, M. Csernoch, K. Abari és J. Máth: First year students' algorithmic skills in tertiary Computer Science education, *9th International Conference on Knowledge, Information and Creativity Support Systems*. Cyprus, November 6–8., (2014), submitted.
- [15] Érettségi eredmények 2007–2013. <https://www.ketszintu.hu/publicstat.php>. Letöltés dátuma: 2013. 08. 30.
- [16] Központi írásbeli feladatsorok, javítási-értékelési útmutatók. Korábbi vizsgaidőszakok vizsgaanyagai - vizsgaidőszakok szerint 2005–2013 <http://www.oktatas.hu/koznevelés/erettsegi/feladatsorok>. Letöltés dátuma: 2013. 08. 30.
- [17] Az olasz, magyar, görög után ezúttal egy holland tanulmány a tudatlanság áráról. Az ECDL Alapítvány március 9-i sajtóközleménye alapján. Mi újság. 2012. április. http://njszt.hu/sites/default/files/mi_ujzag_2012_aprilis.pdf. Letöltés dátuma: 2013. 08. 30.
- [18] J. Kruger és D. Dunning: Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments. *Journal of Personality and Social Psychology*. Vol. 77, No. 6, (1999), 1121–1134.
- [19] M. Ben-Ari: Bricolage Forever! PPIG 1999. 11th Annual Workshop. 5–7 January 1999. *Computer-Based Learning Unit*, University of Leeds, UK. <http://www.ppig.org/papers/11th-benari.pdf> (1999). Letöltés dátuma: 2013. 08. 30.
- [20] R. Panko és S. Aurigemma: Revising the Panko-Halverson taxonomy of spreadsheet errors. *Decis. Support Syst.* 49, 2 (2010): 235–244.
- [21] F. Tort: Teaching Spreadsheets: Curriculum Design Principles. In S. Thorne (Ed.), *Proceedings of the EuSpRIG 2010 conference: Practical steps to protect organisations from out-of-control spreadsheets*, (2010) 99–110.
- [22] F. Tort, F. Blondel és É. Bruillard: Spreadsheet Knowledge and Skills of French Secondary School Students. R.T. Mittermeir and M.M. Sysło (Eds.): *ISSEP 2008*, LNCS 5090, 305–316, (2008). Springer-Verlag Berlin Heidelberg.

- [23] A. Van Deursen és J. Van Dijk: CTRL ALT DELETE. Lost productivity due to IT problems and inadequate computer skills in the workplace. Enschede: Universiteit Twente. (2012). Letöltés dátuma: 2014. 05. 18. http://www.ecdl.org/media/ControlAltDelete_LostProductivityLackofICTSkills_UniverstiyofTwente1.pdf.
- [24] M. Gove: Michael Gove speech at the BETT Show 2012. <https://www.gov.uk/government/speeches/michael-gove-speech-at-the-bett-show-2012>. Letöltés dátuma: 2013. 08. 30.
- [25] J. M. Carroll: The Nurnberg funnel: designing minimalist instruction for practical computer skill, M.I.T. Press, Cambridge, Mass, 1990.
- [26] J. Nielsen: Usability Engineering, Academic Press, Boston, MA, 1993.
- [27] P. Warren: Learning to program: spreadsheets, scripting and HCI, in *Proceedings of the Sixth Australasian Conference on Computing Education*. Darlinghurst, Australia. Vol. 30, (2004), 327–333.
- [28] S. Papert és I. Harel: Situating Constructionism. Constructionism, Ablex Publishing Corporation: (1991) 193–206. Letöltés dátuma: 2014. 02. 02. <http://www.papert.org/articles/SituatingConstructionism.html>.
- [29] S. Booth: Learning Computer Science and Engineering in Context. Computer Science Education. Vol. 11, No. 3, (2001), 169–188.
- [30] R. E. Mayer: The Psychology of How Novices Learn Computer Programming, *ACM Comput. Surv.*, Vol. 13 (1), (1981), 121–141.
- [31] J. Case és R Gunstone: Metacognitive development as a shift in approach to learning: an in-depth study. *Studies in Higher Education*, 27(4), (2002) 459–470.
- [32] J. Case és R Gunstone: Approaches to learning in a second year chemical engineering course. *International Journal of Science Education*, 25(7), (2003) 801–819.
- [33] J. Case, R Gunstone és A. Lewis: Students' metacognitive development in an innovative second year chemical engineering course, *Research in Science Education*, 31(3), (2001) 331–355.
- [34] R. Lister, B. Simon, E. Thompson, J. L. Whalley és C. Prasad: Not seeing the forest for the trees: novice programmers and the SOLO taxonomy, in *Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education*, New York, NY, USA, (2006), 118–122.
- [35] R. Lister, E. S. Adams, S. Fitzgerald, W. Fone, J. Hamer, M. Lindholm, R. McCartney, J. E. Moström, K. Sanders, O. Seppälä, B. Simon és L. Thomas: A multinational study of reading and tracing skills in novice programmers, *SIGCSE Bull.*, Vol. 36 (4), (2004), 119–150.
- [36] E. Soloway, J. Bonar és K. Ehrlich: Cognitive strategies and looping constructs: an empirical study. *Communications of the ACM*. Vol. 26, No. 11 (1983), 853–860.
- [37] E. Soloway: Should we teach students to program? *Communications of the ACM*. Vol. 36, No. 10 (1993), 21–24.
- [38] J. B. Biggs és K. E. Collis: Evaluating the Quality of Learning: The SOLO Taxonomy. New York: Academic Press, 1982.
- [39] J. L. Whalley, R. Lister, E. Thompson, T. Clear, P. Robbins, P. K. A. Kumar és C. Prasad: An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies, in *Proceedings of the 8th Australasian Conference on Computing Education – Vol. 52*, (2006), 243–252.
- [40] J. Sheard, A. Carbone, R. Lister, B. Simon, E. Thompson és J. L. Whalley: Going SOLO to assess novice programmers, *SIGCSE Bull.* Vol. 40 (3), (2008), 209–213.

A kétfázisú HY-DE-modell: a hiper- és mélyfigyelem fázisváltásai a katedrától a hallgatói önfejlesztésig

Dani Erzsébet

Debreceni Egyetem Informatika Kar, Könyvtárinformatika Tanszék

dani.erzsebet@inf.unideb.hu

Absztrakt: Napjainkban egyre több szó esik az általam „bit-generációk”-nak nevezett Y- és Z- nemzedékek tanulási szokásairól, az IKT eszközök elterjedéséről, a digitális kompetenciák erősödéséről, és általában a digitális műveltség összetevőiről, miközben a szövegértelmezési képességek rohamos csökkenésének is szemtanúi vagyunk. A bit-nemzedékek napi szokásrendszere is folytonos mozgásban van, minek következtében idővel változni fog a kognitív működés a megértésben és a tanulásban, vagyis ennek révén az olvasásban is.

A felsőoktatást lassan elérő Z-generáció kommunikációs, információkeresési és tanulási szokásai sok tekintetben szignifikánsan eltérnek az előző generációkétól, attól tehát, ahogyan az előző nemzedékek reagáltak az információs társadalom kihívásaira. A kognitív folyamatokhoz elengedhetetlenül szükséges mélyfigyelem egyre inkább háttérbe szorul és helyét átveszi a hiperfigyelem (N. Katherine Hayles fogalmai). Utóbbi nélkül a több csatornán, multimédiásan érkező információ feldolgozhatatlan. A digitalizált világban a befogadó (olvasó) agy a sorokat ráadásul képként értelmezi. A képekben való gondolkodás pedig egyre inkább eluralkodik a „digitális bennszülöttek” körében. Így aztán a McLuhani kijelentés új értelmet nyer: a médium most már nem csak meghatározza az üzenetet, hanem befolyásolja annak tartalmát és célját is.

Jogosan tehetjük fel a kérdést: az új nemzedékek kommunikációját valóban az eszköz határozza meg? A digitális kor eszközei új tanulási környezetet teremtenek. Napjaink felsőoktatásában ki van hátrányban az információs szupersztrádán: a digitális bevándorló oktató vagy a digitális bennszülött hallgató? Tudjuk-e úgy használni a digitális környezet eszközeit, hogy azok tudatos előtérben tartásával, adott esetben visszább fogásával, a hiperfigyelmet megfelelő szinten tartva, elérjük azt a passzívan befogadó figyelmi állapotot, amelyben már az *egyén* kognitivitása a meghatározó, illetve megindulhat a mélyülő és önállósodó gondolkodás irányába?

Jelen tanulmányban nem célom a generációkat elemző nézetek ütköztetése. Objektív kutatási eredményekből kiindulva, az alábbi kérdésekre keresem a választ.

- (1) Hogyan kerül előtérbe a hiperfigyelem a mélyfigyelemmel szemben a bit-generációk tanulási szokásaiban?
- (2) Hogyan lehet a hiperfigyelmet kihasználni a mélyfigyelemre áthelyezni a hangsúlyt?
- (3) Utóbbi kérdés megoldására dolgoztam ki a kétfázisú **HY-DE-modellt**.¹ Mi a modell értelme és hogyan próbál a helyzeten segíteni?

Kulcsszavak: HY-DE oktatási modell, hiper- és mélyfigyelem, olvasási-tanulási narratívák

1. Az olvasás narratívája

A narratívaelméletek legszélesebb értelmében véve, az olvasásnak is megvan a maga narratívája (ha a szakirodalom eddig nem is közelített hozzá ilyen kontextusban). Különösen áll ez a mai korban, amikor az olvasásproblematika sokkal összetettebb kérdéssé vált, mint valaha volt. Amióta olvasás létezik az emberiség történetében, azóta beszélhetünk az olvasás narratívájáról is. Az olvasás kutatásával több tudományterület foglalkozik. Az

¹ A HY-DE elnevezést a hiper- és mélyfigyelem angol megfelelőinek („hyper-” és „deep attention”) első szótagjaiból alkottam.

olvasásszociológia, olvasásfilozófia, olvasáspszichológia, olvasáspedagógia vizsgálati tárgya maga az olvasás – ezeknek kialakult iskolái vannak. A biblioterápiának mint interdiszciplináris terápiás módszernek is megvan a maga narratívája. Jelen esetben azonban nem ezekkel a kérdésekkel foglalkozunk. És az olvasást nem is hermeneutikai értelemben közelítjük meg, bár, az értelmező olvasás, mint tevékenység, része annak, amiről itt szó van.

A mai olvasásnarratíva sem más, mint történet, sorstörténet, az olvasás sorsának története az információs társadalom és a digitális technika korában. Mi több, ha ránézünk az olvasás történetére, nagy művelődéstörténeti és kultúrtörténeti összefüggésben, az olvasásnak is megvannak a *nagy* narratívái. Ahogyan azt Jean François Lyotard megfogalmazza, a nagy narratívák egy-egy tudományterület meghatározó eszközei önmaguk legitimálásához. A posztmodern ember katasztrofális ideológiai rendszereket és történelmi helyzeteket tapasztalt meg, és ezek (például szélsőséges ideológiák, világháborúk, népiirtások révén) kiábránduláshoz vezettek. Ezáltal a nagy metanarratívák, melyek széles összefüggésekben értelmezik és törvényesítik az általuk képviselt beszédmódot és az abban foglalt hatalmi diskurzust a világban, a posztmodern korban delegitimálódnak, elvesztik addigi legitimitásukat [1].

Az olvasás tehát, létrejötté óta, maga is felfogható egy nagy (meta) narratívának. Ez a narratíva arról szól, hogy mi az olvasás, mit, hogyan és miért olvasunk és így tovább. Ezeknek a kérdéseknek külön elméletei, iskolái vannak (és a gutenbergi világgal kapcsolatosak), melyekkel jelen kontextusban nem foglalkozunk. Ugyanakkor az olvasás narratívája történetében is (mint ahogyan azt a kuhni paradigmaelmélet óta tudjuk) vannak paradigmák és paradigmaváltozások. A preGutenbergi és a postGutenbergi (mely Neuman-galaxisként is azonosítható) világ metszéspontján éles paradigmaváltás következik be: az e-világ térhódításának következtében a Gutenberg utáni olvasásnarratíva egyértelműen a digitalizáció körül forog. Következik ez mindabból, hogy a tudásalapú társadalmak nagy tempóban zúdítják az új nemzedékekre, elsajátítandó/elsajátítható narratívák áradatában az örökölt és az új tudást. A kibermédia világa ehhez soha nem tapasztalt gyorsasággal fejleszti ki az új információhordozó eszközöket, az adatkezelési-, továbbítási-, rögzítési technikákat.

Ezzel az egyre inkább gyorsuló folyamattal az olvasás narratívájában is gyökeres változások következnek be, delegitimálódik mindaz, ami a preGutenbergi korban hitelesítette az olvasást. Posztmodern viszonyrendszerben, az informatizált társadalomban, Lyotard szerint, „a tudás státusa megváltozik” [1]. A felgyorsult és egyre nehezebben követhető technikai átalakulások jelentős hatást gyakorolnak a tudás két alapfunkciójára: a kutatásra és az ismeretek átadására [1]. Ennek következtében a tudás természete is átalakul: „csak akkor haladhat az új csatornákon és válhat operacionálissá, ha az ismeret informatikai mennyiségekké alakítható” [1]. Ezen a ponton kapcsolódhat a mi gondolatmenetünk a lyotard-i eszmerendszerhez: az információ egysége a bit, a posztmodern állapotban a tudás bit-alapúvá válik – ennek a gondolatiságnak a jegyében használom az általam bevezetett bit-generációk elnevezést az Y és Z nemzedékekre, melynek tagjai az új technikák és alkalmazások igazi fogyasztói.

Kérdésként vetődik fel: az olvasás e-világbeli nagy paradigmaváltása mit tesz ezekkel a nemzedékekkel? Ha az olvasás posztmodern narratíváját tekintjük a vizsgálódás alapjának, kérdésként fogalmazódik meg: mit tesz a paradigmaváltás azzal, aki olvas, amit olvas és ahogyan olvas. Azt, hogy a paradigmaváltás mit tesz a figyelemmel, Hayles már

megállapította: létrehozza a hiperfigyelmet. Hayles hiperfigyelem-fogalma² a következő jellemzőkkel definiálódik: gyors fókuszváltás a különböző feladatok között, a többszörös információs folyam túlsúlya, magas stimulációs küszöb, alacsony tolerancia az unalommal szemben. Ez a fajta figyelem igen hasznos a gyors környezetváltást és gyors reakciókat igénylő helyzetekben [2]. Történetiségét tekintve a hiperfigyelem, Hayles szerint, korábban alakult ki, mint a mélyfigyelem. Utóbbi hagyományosan a kognitív megismerésre fókuszál és a bölcsészettudományokkal társul; hosszú időn keresztül képes egyetlen tárgyra koncentrálni, mely információ egyetlen forrásból érkezik, és kizár minden más környezeti forrást. Mindkét figyelemnek megvannak az előnyei és hátrányai. A szépirodalom megértéséhez elengedhetetlenül szükséges a mélyfigyelem, míg a hiperfigyelem hátránya egy viktoriánus novellához szükséges metafizikai gondolkodásban, vagy egy bonyolult matematikai probléma megoldásában egyaránt megmutatkozik [2].

Minek lehet nevezni a hiperfigyelmet az olvasás tekintetében? Továbbgondolva: hogyan jelentkezik az említett jelenség az olvasás-megértés-memorizálás hármasan keresztül a tanulás narratívájában?

A befogadás érzékszervei nem többszöröződtek meg, ahogyan a tudást befogadó agy sem – attól függetlenül, hogy a több csatornán érkező, multi/hipermediális környezet szinte ontja az információt. És a nagy tudományos vívmányként üdvözölhető, soha nem képzelt lehetőségeket jelentő médium így lesz annál is több, aminek McLuhan leírta (a „médium maga az üzenet”), mert ma már a közvetített tudásözönben nem csak az üzenet tartalmát és célját határozhatja meg, hanem a befogadó nemzedéket is. Legfőképp azzal, hogy az új médiák nyújtotta információs sávszélesség, gyorsaság, színesség, változatosság kifejleszti a hiperfigyelmet, és háttérbe szorítja az információs monoáramon érkező, gondolkodást igénylő olvasást (mely mélyfigyelmet előfeltételez), ezáltal a tanulás folyamatába is beleszólva. A szépirodalom-olvasás kifejezetten megszenvedi ezt, hiszen narratívái csak a mélyfigyelem segítségével tölthetik be az identitásképzésben nélkülözhetetlen szerepüket.

Nem szabad figyelmen kívül hagynunk azt sem, hogy „a tudás célja immár nem önmaga, hanem «elveszíti használati értékét»”, ahogyan azt Habermas megfogalmazta [1]. Napjainkban azonban a tudás informatikai áruvá válik, használati értéke pedig megtöbbszöröződik a bit-nemzedékek számára, hiszen az érvényesülés egyik eszközévé válik.

2. Szövegértés és tanulás

A tudás megszerzése a tanuláson át vezet, a tanulási folyamat pedig olvasáson, értelmezésen és memorizáláson alapul. A bit-generációk Y-tagjai már elérték a felsőoktatást, a Z-nemzedék most áll az egyetemek kapuja előtt. Világszerte megfigyelhető jelenség, hogy a fiatal nemzedékek olvasási hajlandósága csökken, a szövegértelmezési képességek évről-évre romló tendenciát mutatnak. Utóbbi ténytet látszik megerősíteni a magyar felsőoktatási intézményekben a Nagy Zoltán médiakutató által végzett kutatás is, mely szerint egy 12 soros, egyperces olvasásnyi tartalommal a hallgatók alig 60 %-a képes megbirkózni a megszabott idő alatt, a tartalmat reprodukálni pedig csak egyharmaduk tudja [3]. A szövegértés tekintetében nem szabad figyelmen kívül hagynunk a 2012-es nemzetközi oktatási vizsgálat, a PISA-felmérés eredményeit sem. Igaz ugyan, hogy a legfrissebb adatok szerint a magyar

² Hayles tanulmányának fordítása a szerző munkája.

diákok a matematikában estek vissza a legjobban, de a szövegértés területén is újra az OECD-átlag alá kerültek. A romló eredmények ellenére a jelentés Magyarországot azon országok közé sorolja, melyekben az első PISA-felmérés óta átlagosan javult a szövegértés tendenciája, de így is mélyen az OECD-átlag alatt helyezkedik el [4].

A fejlett olvasási képesség tehát iskolapolitikai, művelődés- és társadalompolitikai kérdés is, hiszen a generációs váltásoktól függetlenül, a hatékony tanulás egyik alapfeltétele még mindig a szövegértelmezés, úgy társadalomtudományi, mint természettudományi területen. Előállt viszont, ahogy azt már jeleztük, a hiperfigyelem jelensége, a maga pozitívumaival és negatívumaival együtt. Igen komolyan veendő fejlemény, különösen felsőoktatási kontextusban, az oktatási modellek és metodikák újragondolásában és alkalmazásában. Ráadásul a hiperfigyelem eluralkodásával a magyar felsőoktatásban is megjelent a neurodoppingok használata a tanulás stimulálására, és az Y-generáció tagjai körében egyre jobban terjed. Hayles szerint az angol felsőoktatásban is bevált szokás, hogy a jobb tanulmányi eredmény elérése érdekében a hallgatók Ritalint, Dexedrine-t vagy más serkentőszereket szednek.³ Ha ezt a problémát a hiper- és mélyfigyelem szemszögéből vizsgáljuk, egyértelművé válik, hogy a szer használata a mélyfigyelem stimulálására alkalmas. Vagyis, a komoly koncentrációt igénylő tanulási folyamatokban, a hallgatók doppingszerekkel törekednek a mélyfigyelem aktivizálására. A Z-generáció az információs monoáram hatásaira már nem reagál (legalábbis) a hiperfigyelem szintjén. Ez a nemzedék az általam „információs többlettöltetűség”-nek nevezett környezetben szocializálódott (mely többlettöltetűség lehet monomediális, vagy multi- és/vagy hipermediális), tehát, a korosztályra jellemző fiataloknak lehetősége sem volt arra, hogy megtanulja, hogyan fókuszáljon egyetlen dologra.

A hiperfigyelem jelenségének kutatásához Hayles a Kaiser Family Foundation eredményeit is felhasználta. Ez a cég publikálta a mai napig a legelfogadottabb felmérést a fiatal korosztály médiafogyasztási szokásairól. A reprezentatív felmérés 2032 fiatal bevonásával készült, akik közül 694-gyel intenzív napi médianaplót vezettek. Az eredmények a következők. Egy középiskolás tanuló egy nap átlagosan 6,5 órát tölt valamilyen média használatával, és mivel gyakori több média egyidejű használata, ezek bontott idejét szekvencializálva, a 6,5 óra napi 8,5 órára növekszik. A 8,5 órából 3,51 óra jut TV-re és DVD-re; 1,44 óra az mp3, zene CD és rádióra; 1,2 óra a web-böngészésre és más interaktív médiára; 0,49 óra a videójátékokra és végül mindössze 0,43 óra az olvasásra.

A felmérés arra a kérdésre is választ keres, hogy milyen körülmények között készítik el a diákok házi feladatukat. Az eredmény nem meglepő: 30 % legtöbbször TV, zene, és más média használata közben; 31 % teszi ugyanezt, de csak néha. Vagyis néha, vagy többnyire multitasking módon készülnek a zenét (30%-ban), a számítógépet (33%-ban), az olvasást (28%-ban) és a TV-nézést (24 %-ban) váltogatva. Hayles ezt abból a megfontolásból nevezi váltogató rendszernek, hogy a pszichológiai kutatások szerint a „multitasking voltaképpen a különböző feladatok közötti gyorsváltások sorozata”. Ebből arra következtethetünk, hogy „a multitaskinggal oly mértékben csökken a hatékonyság, hogy időkihasználás szempontjából hatékonyabb több feladatot szekvenciálisan végezni és nem szimultán módon (egyidejűleg)” [2].

³ A Ritalin és a Dexedrine az ADHD (mely neurológiai természetű betegség) gyógyszere. A gyógyszer hatására az agyban megemelkedő dopaminszint jótékonyan hat a tanulási képességekre.

Kérdéseim tehát a következők:

1. El kell fogadnunk, hogy a hiperfigyelem jelensége hangsúlyosan jelen van az új nemzedékekben, és tudomásul kell vennünk azt, hogy ha tudatosan nem használjuk ki a hiperfigyelem pozitívumait, ezek a nemzedékek még felületesebbé válhatnak? Esetleg visszafordíthatók-e azok az olvasási-megértési-tanulási fejlemények, melyek a digitalizációs paradigmaváltás következtében meghatározzák a bit-nemzedékeket ezen vonatkozásokban? Lehet-e ebbe a folyamatba beleavatkozni, ellene hatni, esetleg visszafordítani azt? Ugyanis nagy kérdés marad, hogy szabad-e a hiperfigyelem teljes visszaszorítására törekedni, hiszen az információáramlás egyre szélesebben hömpölyög, és ezt képtelenség mélyfigyelemmel feldolgozni.
2. Vagy inkább a kettő dinamikáján, egymáshoz való viszonyán kellene elgondolkodni? Azon, hogy hogyan hatnak egymásra, milyen fázisváltásaik lehetnek?
3. Miként befolyásolja az olvasásban érvényesülő és egymással interaktív kölcsönhatásban lévő, Peter J. Rabinowitz által megállapított négy szabályszerűséget a hiperfigyelem? A rabinowitzi szabályszerűségek részletes bemutatására terjedelmi okokból nincs lehetőség. A négy szabályszerűség a következő:⁴

(1) az olvasás közbeni figyelem és észlelés szabálya: az olvasó a szöveg bizonyos részeit fogja csak fel, a többit figyelmen kívül hagyja – itt derül ki, mennyit fog fel az olvasott szövegből;

(2) a jelentéstulajdonítási szabály (a szignifikáció): ami az előző lépésben megragadta az olvasó figyelmét, annak jelentőséget tulajdonít, tehát szimbolikus tartalommal ruházza fel;

(3) a konfigurációs szabály: az olvasó jelentésmintázatokat hoz létre, összeköti a különböző szövegtöredékeket;

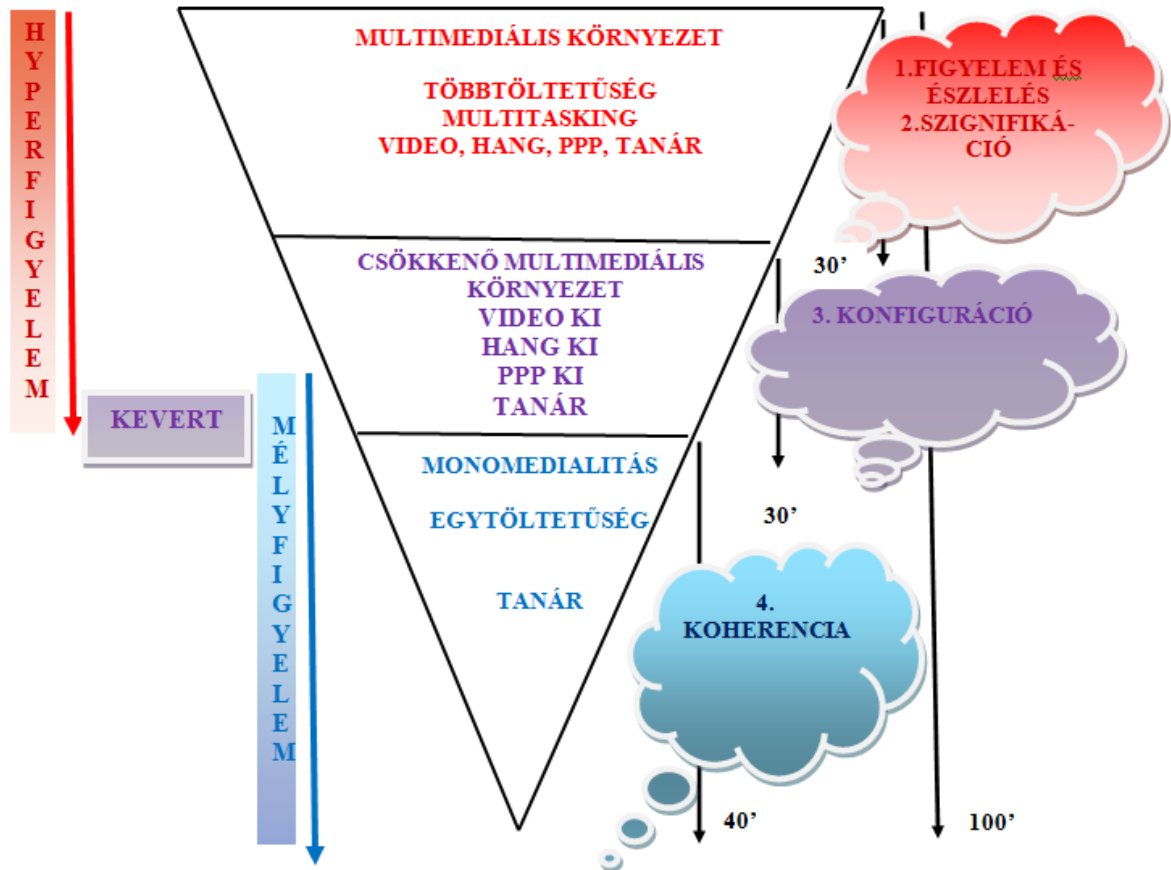
(4) a koherencia szabálya: az olvasó mindig arra törekszik, hogy a szöveget koherens egészé alkossa, akkor is, ha paradoxonok, devianciák, jelentéselhajlások vannak benne [5].

Az általam elképzelt és kidolgozott elméleti modell a fentebb felvezetett kérdések és elméleti belátások figyelembe vételével kísérletet tesz arra, hogy kövesse, esetleg tudatosan irányítani is próbálja a hiper- és mélyfigyelem fázisváltásait felsőoktatási tantermi környezetben.

⁴ A szerző fordítása.

3. A HY-DE modell (első fázis)

Javaslat egy új metodikára.



1. ábra. A HY-DE modell

A tananyagok elsajátításánál (a Rabinowitz-féle szabályszerűségekből kiindulva) az első két szinten (figyelem és észlelés, valamint szignifikáció), a hiperfigyelem túlsúlyának előnyeit kihasználva, az anyag átadása multimédiás felületen történne. Pl.: egy összevont egyetemi előadás (100 perc) első harmada (30 perc) a téma nagyvonalakban történő felvezetését tartalmazná, úgy, hogy a lehető legintenzívebb többlettöltetűséget alkalmazzuk, legyen az mono- vagy multimédiás. A multitasking (mely jellemzője a bit-generációknak és együtt létezik a hiperfigyelemmel) segítséget jelenthet ebben a fázisban. A hallgatók a hiperfigyelem segítségével nagy vonalakban áttekintést kapnak az órai anyagról, és a figyelem–észlelés szabályainak megfelelően alapvető terminusokat, információkat memorizálnak.

Ezt követően, a konfigurációs szabály törvényszerűségeit figyelembe véve, ezen a második szinten (a következő 30 percen), ugyanazon anyag előadásával, fokozatosan csökkentenénk a multimedialitást, az információs többlettöltetűséget (pl. megszűnik a hang, majd a mozgókép, és fokozatosan a bemutató diái). A hallgatók (hagyományos jegyzeteléssel) vázlatot készítenének – ez fokozatosan mélyfigyelemre szorítaná őket, és pedig úgy, hogy még (noha csökkenő intenzitással) a hiperfigyelem is jelen van.

A koherenciaképzés szakaszába érve (az előadás utolsó 40 perce – harmadik szint), teljesen megszűnne a többlettöltetűség, és az információs monoáram (ez esetben az oktató)

kerülne előtérbe, megpróbálva ezzel a mélyfigyelem aktivizálását segíteni. Ebben a folyamatban már a téma elmélyült taglalása, csak a tartalomra való koncentráció a jellemző. Az elképzelt modellben tehát a téma háromszintű mélységben jelenne meg: hiperfigyelmi, kevert, és mélyfigyelmi szinten, előadótermi környezetben.

4. Összegzés

Tanulmányom, a helyzet elemzését követően, a hiperfigyelem fokozatos visszaszorítását szolgáló felsőoktatási metodikára tesz javaslatot. Az új metodika célja kettős: elsősorban úgy próbálja edzeni és tudatosítani a hiperfigyelmet, hogy az minél szélesebb öleléssel legyen képes átfogni maga körül a szinte áttekinthetetlen információs világot, de oly módon, hogy meg tudjon ragadni a lényegből annyit, amennyi mélyfigyelemmel feldolgozható. Másodsorban, megpróbálja tudatosan úgy befolyásolni a hiperfigyelmet, hogy annak fokozatos visszaszorításával előtérbe kerüljön a mélyfigyelem, ahol már a tanulás a tudás felé halad. Ez a folyamat (is) hozzásegítheti a bit-nemzedéket ahhoz, hogy az e-világbeli identitásforgatókönyvek szerint megjelenő Akárki-, Senki-, Valaki-identitások közül, a tömegesen jelentkező Akárki- és Senki-identitás helyett minél nagyobb számban Valaki-identitású fiatalok hagyják el az egyetemeket, noha ezek között az agykutatás által elkülönített típusok között nincsenek éles választóvonalak (ezeket az ID-kategóriákat Susan Greenfield brit agykutató állította fel).

Terjedelmi okokból itt nincs lehetőség a különböző identitástípusok részletes taglására. Röviden összefoglalva:

- A Valaki-forgatókönyvben az egyént az elméje határozza meg. Az agy rugalmasan alkalmazkodik a digitális környezethez, állandóan változó „sejtszövetségeket” hoz létre [6]. A Valaki-identitásban rendkívül fontos szerepe van a szépirodalom olvasásának, leginkább a regénynek, mert segítségével az egyén fogalmi kereteket, narratívát alakít ki magának.
- A Senki-forgatókönyvben a képernyőkultúra jut elsődleges szerephez. A számítógépes játék egyrészt magányos tevékenység, másrészt a hangsúly a tartalomról a folyamatra tolódik – a logikai fogalmi keret nem tud kialakulni. A Senki-agy képi fantáziája hanyatlak, nehezen választja el a virtuálist a valóságtól, a cselekvésekben nem a cél dominál, hanem az automatizált folyamat. Ebben a forgatókönyvben nem jelentkezik az olvasás igénye, és nem szükséges hozzá huzamosabb idejű gondolkodás: az állandóan magasan tartott ingerküszöb az „itt és most” kábulatát jelenti, egyfajta függőséget – úgy, hogy az egyén mit sem tud erről a függőségről, és ez a dolog veszélyesebb része. A Senki-forgatókönyvben az olvasás nem jelent mást, mint személytelen, a befogadóban értelmes módon össze sem álló információfoszlányokat. Az „elme nélküli gondolkodásnak” két identitásváltozata van: a hamis (avatar) és a kollektív identitás [6].
- Az Akárki-forgatókönyvben a kollektív identitás játszik főszerepet az egyénnel szemben: az önálló identitás beleolvad a közösségi azonosságba. Az Akárki-agy egyik jellemvonása a rögzült állapot (a gondolkodási séma szinte megváltoztathatatlan) és a bejövő impulzusok iránti zártság. E forgatókönyv szerint az egyén olvas, de lényegtelen hogy mit: akármit. Nem saját meggyőződéséből válik olvasóvá, nem tudatosan olvas, nem kritizál, nem szintetizál, nem gondolkodik metafizikusan – optimális fogyasztóvá válik [7].

A HY-DE modell alkalmazása bizonyos mértékű korrekciós lehetőséggel kecsegtet, mert a hiperfigyelmű nemzedékekben lassítaná vagy megállítaná a narratívabefogadás elsekélyesedését, és visszavezetné őket a mélyfigyelemmel megszerezhető alaposabb és mélyebb tudás felé. Kitüntetett szerep juthat ebben a szépirodalmi narratíváknak. Vagyis egészséges irányban befolyásolhatnánk azt a narratívát, mely az olvasás és tanulás narratívája a digitális korban.

A kétfázisú elméleti (HY-DE) modell itt bemutatott része csak az első szakasza a végleges modellnek (a gyakorlatba való átültetéssel ez az első szakasz feltehetően tovább finomul/kristályosodik). Az oktatói szakasz, melyben az oktató feladata úgy előkészíteni az órai anyagot és vezetni az órát, hogy azzal tudatosan igyekezzon a hiper- és mélyfigyelem fázisváltásait befolyásolni. Az elméleti órákra kidolgozott modell második része a hallgatói, mely szemináriumi foglalkozás keretében, a mély- és hiperfigyelmi fázisváltást megfordítva (vagyis a mélyfigyelemből kiindulva fokozatosan haladva a hiperfigyelem felé) törekszik arra, hogy a hallgatók tanulási narratíváját fejlessze. Ennek bemutatása azonban egy másik tanulmány témája lesz.

Irodalomjegyzék

- [1] Lyotard, Jean-François: A posztmodern állapot. Budapest, Századvég, 1993.
- [2] Hayles, Katherine N.: Hyper and Deep Attention: The Generational Divide in Cognitive Modes. http://engl449_spring2010_01.commons.yale.edu/files/2009/11/hayles.pdf [2014. 05. 25.]
- [3] Dani Erzsébet: Neumann kontra Gutenberg-galaxis?: Különös tekintettel a generációs olvasási szokásokra. *Könyv és nevelés*, XIV. évf., 4 (2012), 48-59.
- [4] A magyar kamaszoknak nem megy a matek. <http://www.origo.hu/itthon/20131203-a-2012-es-pisa-felmeres-eredmenyei.html#> [2014. 06.25.]
- [5] Rabinowitz, Peter: Before Reading: Narrative Conventions and the Politics of Interpretation. Columbus, Ohio State University Press, 1987.
- [6] Greenfield, Susan A.: Identitás a XXI. században [ford. Garai Attila]. Budapest, HVG, 2009.
- [7] Steklács János: Személyiség, identitás, olvasás a jövő forgatókönyvében. *Könyv és nevelés*. XII. évf., 1 (2010), 36-42.

Startup, vagy munkahely? IKT vállalkozások feltételrendszerének egy modellje

Startup or a job? Conditions to start an ICT business

Dr. Dobay Péter^a, Dr. Kruzslicz Ferenc^b

^aPTE KTK Gazdaság-informatika Tanszék
dobay@ktk.pte.hu

^bPTE KTK Gazdaság-informatika Tanszék
kruzslicz@ktk.pte.hu

Absztrakt: Míg Magyarországon 10 végzős egyetemista közül 2-3 gondol vállalkozásra, addig az USA-ban, vagy Braziliában ez a szám 8 körül van. A hazai informatika szakos hallgatók zöme is úgy gondolja, hogy egy multinál kell kényelmes, alkalmazotti állásban elkezdni a karriert, és majd a lépcsőfokokon lehet felfelé haladni. Természetesen a lépcsőfokok nagyon hosszúak, s természetesen vannak, akik éppen egy „második Yahoo”-n, vagy egy újabb Angry Birds applikáción törik a fejüket – ez azonban egy sokkal rögzösebb, kockázatosabb út. Két megkérdezéssel vizsgálatot végeztünk informatikus hallgatók körében, s ennek eredményei eléggé alátámasztják a fentieket: még mesterszakosok és PhD-sek sincsenek tisztában azzal, hogyan is lehet betörni az IKT piacra, milyen elemi (és később: további!) feltételek szükségesek egy sikeres vállalkozáshoz, s mikor lesz mind-ebből egy valódi „startup”, egy felívelő, innovatív cég. Bemutatjuk azt a tervezett modellt, amely egyrészt egyéni és csoportos önvizsgálati eljárásokat, másrészt üzleti elemzési lehetőségeket ad egy IKT startup-ról való kezdeti gondolkodáshoz, majd a tényleges üzleti tervezés elindításához. Nyilván felvethető a kérdés, hogy az informatikai tantervek hogyan készítik fel minderre a hallgatót, miben tér el a mérnök-, a programtervező- és a gazdaság-informatikus ezirányú felkészültsége, s hogyan lehetne mindezeket javítani a felsőoktatásban: kötelező kurzusokkal, önkéntes tréningekkel, vállalkozási központok programjaival? S természetesen gondolnunk kell a hazai társadalom- és gazdaságpolitika mindennapos üzeneteire, a média által közvetített gazdasági világ- és ország-képre, a munka-, vagy tudás-alapú gazdaság közötti ellentmondásokra is. Az előadást néhány egyszerű javaslattal zárjuk.

Kulcsszavak: IKT vállalkozás, informatikai szakok, üzleti ismeretek, projektmunka

Abstract: Only 2-3 of 10 Hungarian university graduates has plans of self-employing or starting a business, whilst the same at the US or in Brazil is about 8. The majority of graduates intend to have a good job at a large – multinational – company, as being employed and creating a career step by step, even if rungs of a ladder are far from each other. Others think on creating a „second Yahoo” or an app to be a millionaire as soon as possible. We have executed two pilot surveys to find explanations whether students in their master courses or even at PhD education know enough about establishing and running a small business. Questions targeted whether they know market techniques, cost/benefit calculations, sources of financing an SME and managing a small company – maybe a startup. We show elements of a model to serve this evaluation process and present questions of how to embed information, knowledge and skill about SME development to ICT curricula. A further question can be raised as what is the difference when students on Computer Engineering, on Software Design or on Business Information Systems are involved. Practically there is no space for more compulsory courses – the solution could be introducing electives or separation of this type of trainings from the ordinary university curriculum. The contribution is closed with some proposals to solve these problems.

Keywords: entrepreneurship in ICT, curricula in ICT, business courses and trainings, projectwork

1. Bevezetés

A végzős egyetemisták vállalkozási hajlandósága minden országban a stratégiai tervek fontos eleme: politikusok, oktatási szakemberek, fejlesztési ügynökségek kíváncsiak arra, hogy vajon a friss diplomás alkalmazotti állást keres, vagy önállóan elindul az üzleti életben. Ma mindennapos a média és egyes felelős vezetők vélemény-nyilvánítása: „A felsőoktatás képtelen a munkaerőpiac igényeinek megfelelő tudást és leginkább kompetenciákat adni a kikerülő fiataloknak”. Ugyancsak lesújtó vélemények jelennek meg az innovációs készségek hiányáról, s megkérdőjelezik, hogy az ilyen irányú támogatások valóban hasznos kutatásokra fordítódnak-e. Eltekintve mindezek általános tárgyalásától, attól, hogy a kreativitás és innováció sehol sem hoz 100%-os beválási arányokat, valamint hogy sohasem volt közvetlen célja a felsőoktatásnak az azonnali munkába-állás követése, csak azt a kérdéskört elemezzük jelen dolgozatban, hogy milyen módon növelhető a nem-üzleti képzésekben résztvevők – speciálisan az „informatikusok” – vállalkozási hajlandósága, ilyen irányú készségeik fejlesztése. A kisvállalkozások ugyanis jelentős szerepet játszanak a foglalkoztatásban, s az igazán sikeresek, a „startup”-ok pedig az innovációs előrehaladásban, a műszaki-technológiai, vagy más irányú fejlesztési eredmények piacra vitelében. Akkor, amikor a nagyvállalatok és a közigazgatás, a szolgáltatások a racionalizálási törekvések eredményeképpen egyre kevesebb végzettet vesznek fel, mindenképpen fontosnak látjuk azzal foglalkozni, hogy elősegíthető-e a diplomások foglalkoztatása saját vállalkozások indítása, KKV-kban történő munkavégzés által.

Azokban a társadalmakban, ahol a gazdasági környezet valódi piacgazdasági – mai szitokszóval: szabadpiaci, liberális – elveken alapul, a vállalkozás szabadsága és védelme alaptörvény erősségű, a vállalkozóvá válás pedig vágyott és gyakorolt életforma, s mindennek következményeként gyermekkortól az ifjúság számára követendő példaként állított életpálya. A centrálisan tervezett-szervezett gazdaságú, kapcsolataikban zártnak nevezhető országokban (amelyre láttunk jó néhány évtizedes példát és gyakorlatot) ezt úgy próbálják helyettesíteni, hogy erre felkért vezetők és szakembereik többszintű hierarchiájú irányítási rendszerben mondják meg, hol, mit, miért és hogyan kell termelni – elosztani – elfogyasztani, azaz vállalkozási szándékokra, innovációra, kreativitásra nincs szükség.

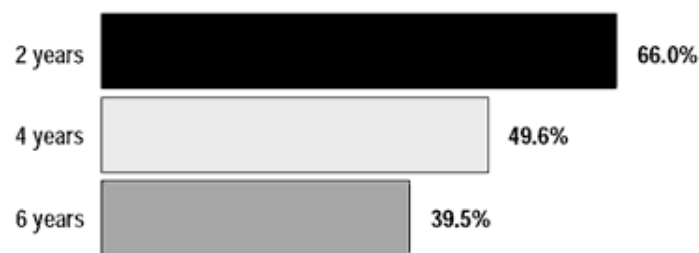
Ma az európai munkaerő mintegy 85%-át a kis- és középvállalkozások (KKV) foglalkoztatják. Az OECD-országokban 95% körüli a KKV-k részesedése az összes vállalkozásból. A rendszerváltás után a kényszernek köszönhetően hatalmas felfutás volt a vállalkozások létszámában, de ugyanakkor még a kétezres évek közepén is a közgazdasági képzések közép-pontjában a nagy (termelő-szolgáltató) vállalatok munkájának szervezése, menedzselése, információs hálózatainak működtetése álltak. Európa innovációs lemaradása többek szerint annak is eredménye, hogy viszonylag kevés a jelentős újdonságokat felmutató kisvállalkozás, kevés az igazi siker, s így a fiatal generáció sokkal kevésbé hajlandó ezt a kockázatosabb, noha ígéretesebb utat választani.

Az EU-ban évente 4-5 millió új KKV (50- 250 fő alatti vállalkozás) jön létre, s kb. ugyanannyi meg is szűnik: ez mutatja, mennyi mozgékony, mennyire labilis ez a vállalkozási szegmens. Ugyanakkor az Európa Tanács statisztikái szerint a működő mintegy 23 millió KKV a munkaerő-piac mintegy 75-85%-át tudja lefedni, a működő cégek csaknem 99%-a ilyen kisebb szervezet – nyilvánvaló tehát a közösségi szándék az ilyen szervezetek támogatására.¹

¹ lásd pl. <http://www.elections2014.eu/hu/top-stories/content/20131014TST22248/html/Foglalkoztat%C3%A1s>

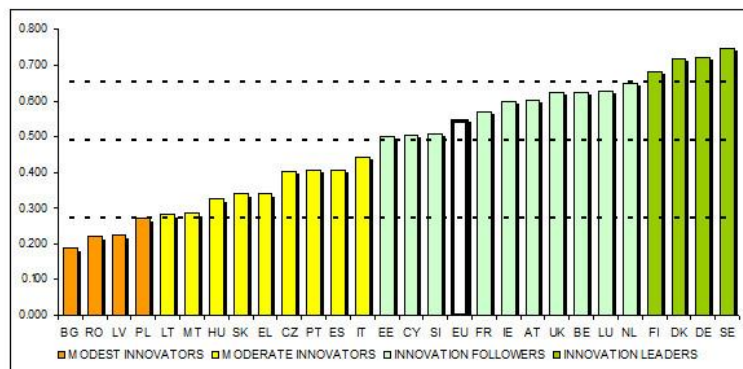
„Itt kell tartanunk a startup-okat, mert munkájuk, tudásuk mindannyiunk javára válhat, nagy hatással lehetnek a gazdaságra, nemcsak tagállami, de európai szinten is.” – nyilatkozta Neelie Kroes, az Európai Bizottság digitális politikáért felelős alelnöke a Startup Európa kezdeményezésről a Világgazdaságnak.² „... a Startup Európa, a közösség digitális napirendjének egyik kezdeményezése, amelyet ICT és webes vállalkozások elindításának és fellendítésének ösztönzése érdekében indítottunk el. ... A program keretében több akció indul: a Startup Európa Akceleratorainak Közgyűlése, ... a Startup Európa Befektetői Fórum, amely megnöveli a kockázati tőke kompetenciáját a tech startupok esetében.”

Minél kisebb a KKV, annál fontosabb a tudástőke szerepe. Az iparági szakismeretek, a vállal-kozási készségek és képességek alapvetően meghatározzák az erőforrás-hiányos mikrovállalkozások túlélési arányát. Az amerikai piac statisztikái azt mutatják, hogy az átlagos cég-életciklus kb. 11.2 év³, ugyanakkor a túlélés esélyeit az alábbi ábra mutatja:



1. ábra: Amerikai start-up vállalkozások túlélési aránya
(Forrás: SBA, "Small Business by the Numbers," May 2002)

Évekig azt tartották, hogy a „vállalkozás” létrehozása és sikere egyenlő az ötletadó „vállalkozóval”, plusz az ő tevékenységével. Az informatikai iparban sokkal inkább azt kell hangsúlyoznunk, hogy a piaci környezet alapos és naprakész ismerete, a jó menedzsment és marketing módszerek együttese (beleértve ebbe a finanszírozási politikát is) hozza meg a piaci sikert – egyszerűen meg kell értenünk, hogy ezernyi, hozzánk hasonló „kreatív elme” próbál meg ugyanerre a piacra betörni.



2. ábra: EU tagországok innovációs teljesítménye (24 paraméter alapján)
(Forrás: EC Innovation Union Scoreboard, 2013)

² Világgazdaság, 2013. 11. 8.

³ IDC, "Web Site Development in U.S. Small Businesses," June 2004

A jelek szerint az EU és a nemzeti kormányok közös erőfeszítése lesz a jövőben az innováció és az informatikai jellegű fejlesztések-kutatások támogatása. Az European Commission “Innovation Union Scoreboard 2013” kiadványa szerint Magyarország innovációs szempontból az utolsó harmad végén tartózkodik⁴. Az emberi tényezők területén Magyarországon a doktori képzésre jelentkezők számát, valamint az érettségivel és diplomával rendelkezők arányát veszik figyelembe. Magyarország gyengesége egyértelműen a kis és közepes vállalkozások innovációs képessége.

Munkánknak két célkitűzése volt:

A/ Ismeretszerzési modellt akartunk kipróbálni az egyének és hallgatói csoportok vállalkozási és piac-érettségi szintjének értékeléséhez, valamint

B/ Kísérletet kívántunk végezni vállalkozás-közeli tematika beillesztésére gazdaság-informatikai mesterszakos tantervbe, majd ennek alapján előzetes javaslatokat megfogalmazni tantervi változtatásokra.

A vizsgálatok kiváltó oka egyrészt a Bevezetésben felemlített általános vélekedések voltak, miszerint haszontalanul sokrétű elméleti tudást nyújtunk (különösen mesterszakon), s a végzetek nemhogy alkalmazottként nem tudnak azonnal teljesítményt nyújtani, hanem kisebb vállalkozások indításában és menedzselésében se jeleskednek ismeretek és gyakorlat híján. Másrészt számos külföldi – és néhány hazai – példa nyomán azt látjuk, hogy vannak szerteágazó próbálkozások a gyakorlatiasság magasabb szintjének elérésére (ilyenek például a duális képzések, ipari részvétellel), illetve a vállalkozási hajlandóság növelésére tantervekben szimulált projektmunkákkal, cég-indítási gyakorlatokkal.

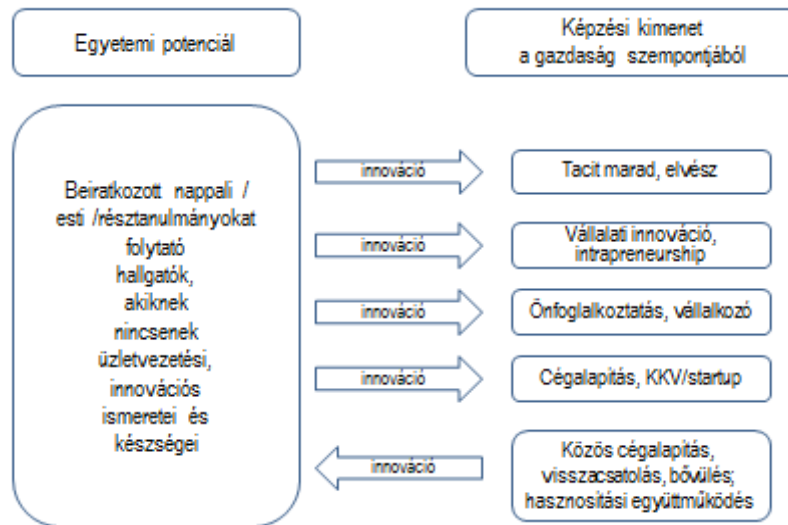
Az világosan látható volt, hogy mindkét célkitűzést ellenállás kíséri:

- A hallgató előzetes „iskolai szocializációja” és a média-háttér ma Magyarországon nem kedvez a kreativitásnak, hanem sokkal inkább azt „üzenni” a hallgatónak, hogy legyen engedelmes beosztott, ismerje ki egy nagyobb hivatali-vállalati hierarchia belső viszonyait, s ott próbáljon előrejutni, karriert építeni;
- Ha pedig tantervi változtatásokra teszünk javaslatot, egy lazább ipar-egyetem-támogatások körében szervezünk (alapvetően kompetencia-fejlesztő) képzési modult, akkor beleütközünk a képesítési követelmények merev rendszerébe, a tudományosokdás megkövetelt szintjét prezentáló oktatói habitusokba.

Mindezek ellenére úgy gondoljuk, hogy az informatikai területeken mindenképpen szükséges ilyen készségek kialakítása, legalább az alábbi okok miatt:

- Az ICT iparok a globális és regionális átlagokat jóval meghaladó ütemben bővülnek, beleértve a válság időszakát is, ahol szinte csak ez az iparág maradt talpon;
- Az ICT jellegű mikrovállalkozás igen kis anyagi-fizikai erőforrás felhasználásával elindítható, s noha nehezen lehet a milliárdossá váló ICT startup-ok történetét megismételni, azért a tudásalapú indulás nem túl erőforrás-igényes;
- A cégeket „kicsiben” lehet tartani ezen a területen, a munkatársak akár távmunkában foglalkoztathatóak, a bukás „ára” alacsony;
- Az indulást követő bővülés erőforrás-korlátokba lényegében nem ütközik.

⁴ http://ec.europa.eu/enterprise/policies/innovation/files/ius-2013_en.pdf



3. ábra: A felsőoktatási innováció lehetséges gazdasági kimenetei

Mindezek alapján úgy látjuk, lehetséges nem-üzleti tantervekbe illeszteni olyan képzési modulokat, amelyek a 3. ábra mentén „üzleti kimenetek” létrehozását fogják támogatni.

2. A probléma háttere

A téma szakirodalma a szokásos módon hatalmas, a pszichológiai tényezőktől az egyetemi tantervekig terjedően - itt csak néhány elemet említünk, némileg önkényesen. Nyilván első-sorban az amerikaiak kedvenc témája a vállalkozási hajlandóság, s annak korai felismerése, támogatása. Azt kell valamiképpen megérteni, hogy hogyan indul el az ilyen gondolkodás a fiatalok körében, s hogyan lehet mintegy „kiemelni” a potenciálisan sikeres jövőbeni vállalkozókat a tömegből. Természetesen lehet mindenféle kurzusokat szervezni, s talán ezek gyakorlati elemei a legjobbak arra, hogy a készségfejlesztés során felismerjük, kiben rejlik ilyen irányú tehetség, nyitottság, kreativitás. Már a nyolcvanas években 24.6%-os „hajlamot” mértek amerikai hallgatóknál⁵, s ezt nyilván befolyásolta az ezredfordulós gazdasági boom kialakulása. A kezdeti felmérések nyomán Wang, P. és K. Wong mérnöki-, természettudományi és informatikus egyetemi hallgatók vállalkozási orientációját vizsgálták a Harvardon, s –csodák csodájára – 90%-os „ön-foglalkoztatási szándékot” találtak ([14]). Cikkünkben részletes szakirodalmi áttekintést adnak a témáról, s ezekre alapozott paramétereik: nemi identitás, családi indíttatás, etnikai hovatartozás, pénzügyi feltételek elérhetősége, valamint a kockázatkerülés mértéke.

Ami számunkra talán megdöbbentő: a középiskolások (!) 67%-a vállalkozási jövőt képzelt el magának már 1998-ban! Az angol egyetemisták körében ez 35-40% volt a 90-es években. Vigyázzunk: a szándék és a tény messze van egymástól mindenhol, általában a végzetek 4-8

⁵ Scott, M. F. & Twomey, D. F (1988). The long term supply of entrepreneurs: students career aspirations in relation to entrepreneurship. *Journal of Small Business Management*, 26(4) 5 - 14.

%-a indít ténylegesen vállalkozást 2-3 éven belül. Sok publikáció elemzi a család, a környezeti kultúra szerepét, de meglepő, hogy például a fizető iskolákba, kurzusokra járók messze jobban törekszenek vállalkozásokra, mint szerencsésebb társaik.

David H. Hsu *et al* ([9]) cikkükben a mérnök-karokon követett gyakorlatról írnak, 1950-től kezdve elemezve az MIT „teljesítményét” hardver, szoftver és egyéb kapcsolódó vállalkozási területeken. Fő problémájuk: vajon a mérnöki-innovációs teljesítmény a szükséges, ami képes a fennmaradást, a KKV túlélését biztosítani – vagy milyen szerepe van ebben a tudástőkének, az üzleti szakértelemnek, az ezirányú egyetemi képzésnek? Felméréseikben kb. 60%-ra teszik az „ipari innovációs megrendelés” típusú egyetemi vállalkozásokat, s 40%-ra a kutatási-fejlesztési „belső innováció” szerepét. A tudástőke jelentősége – több publikáció szerint – különösen a millennium környékén kapott nagyobb hangsúlyt, amikor a dotcom buborék kipukkanása⁶ különösen érzékennyé tette a KKV-k sikerét és bukását vizsgáló kutatókat. Charney ([3]) áttekinti az amerikai egyetemeken szokásos „elkülönített” vállalkozói program történetét a University of Arizona esetében. Leszögezi: 10-15 év alatt hatalmas fejlődésen ment át az ilyen „Entrepreneurship Center” és hasonló beépülése az egyetemi campusok-ba.⁷ Az ilyen irányú összehasonlító elemzések közül kiemelik a kormányzati támogatású ún. Small Business Development Center-ek regionális szerepét (mintegy 600 ilyen működik az USA-ban); hivatkoznak olyan tanulmányokra, amelyek a helyi SBDC és a tudásközpontként szolgáló regionális egyetem együttműködését elemzik. A Karl Eller Centerben 16 éve futó ún. Berger-programjuk tanterve alapján 15 év alatt kb. 600 tanuló kapott elméleti és gyakorlati vállalkozási ismereteket. Kutatási eredményeik bemutatják, hogy kb. háromszoros azon alumni hallgatók aránya, akik sikeres KKV-kat menedzselnek, ha az egyetemen a Center-ben ilyen képzésben vettek részt. Ugyancsak háromszoros a különbség a két csoport között, ha a vállalkozás-alapításra, az ön-foglalkoztatásra kérdeznék rá, szemben az alkalmazotti életpályával. Igencsak érdekes eredmény, hogy a vállalkozó alumni rétegben 27%-kal nagyobb az egyetemi tréningekben résztvettek fizetése, s vállalkozásaik vagyona átlagosan 62%-kal nagyobb! 34 dékán nyilatkozata szerint az ilyen „campus-based” vállalkozási központ jelenléte jelentősen befolyásolta a tantervek átalakulását (különösen az MBA és hasonló programokban) – és például az UofA Center vállalati támogatása nagymértékben, több millió dollárra nőtt...

A National Council for Graduate Entrepreneurship (UK) 2006 -os áttekintése szerint ([8]) azt is érdemes megvizsgálni, hogy a szakmaterületek szerint hogyan alakul a későbbi karrier. Így például a művészeti diplomások közül kétszer annyian lesznek vállalkozók, mint alkalmazottak, a bölcsészek és orvosok között kb. fele-fele, a természettudományban kb. negyedével több az alkalmazotti karrier, s meglepő módon az üzleti szakokon végzettek közül kétszer annyi az alkalmazott, mint az önálló vállalkozó.

Sabrina Di Addario és Daniela Vuri olaszországi vizsgálatai szerint ([1]) a 3 éve végzett hallgatók vállalkozási sikeressége nagyban függ a környezettől: nagyobb városokban a siker valószínűsége kisebb. Az olasz vállalkozók között csak 11% a főiskolát-egyetemet végzettek aránya, és a szerzők szerint e szegmens kiemelt vizsgálatával talán sikerül elkerülni azt a szokásos hibát, hogy a rendszerint alulképzett kényszer-vállalkozókkal torzítják el felmérések eredményeit.

6 Lásd pl. a London Silicon Roundabout nevű, gyorsan növekvő „európai” próbálkozást, <http://www.siliconroundabout.org.uk/>

⁷ Egyetemünk az említett UofA-val és pl. az Ohio University (at Athens)-sel is kapcsolatban van ilyen téren: az utóbbi által két évtizede működtetett „Voinovich Center” például egyetemi posztgraduális (MA/MSc, MBA) hallgatók csoportjaival párhuzamosan dolgozik ipari-vállalati megrendeléseken, vállalkozás-alapításon, innovációs inkubációs projekteken.

Ismert és sokat idézett az 1999 óta folyó Global Entrepreneurship Monitor (GEM⁸) kutatás – ennek során több tucat országban – hazánkban is –, több ezer vállalkozás ismétlődő megkérdezésével elemzik a vállalkozási szándékok eltéréseit, változását. Raffai Mária már 1996-ban az IF Konferencián elemezte azt a kérdéskört⁹, hogy az akkor tervezés alatt álló gazdaság-informatikus szak milyen társadalmi igényekkel találkozik. Az akkori felmérésben egyaránt részt vettek vállalkozások vezetői és végzett hallgatók – ők a tantervben megfogalmazott tárgyak vélt és tapasztalt hasznosságát értékelték. Már ott 43% jelölte meg a (tanult!) gazdálkodási ismereteket fontosként – jelentsen ez bármit is. Ugyanakkor a már foglalkoztatottak 29%-a “hiányolta a képzésből” a mélyebb vállalkozási-pénzügyi-számviteli és marketing ismereteket. Szirmai Péter és Csapó Krisztián a Corvinus Egyetemen folyó oktatómunkájuk tapasztalatait írják le 2009-ben ([13]). A tanulmány széleskörűen indokolja a vállalkozói képzés szükségességét, s kísérleti programot is ad – természetesen ez közgazdasági képzési környezetet feltételező helyzet. Kiszl Péter ([10]) a könyvtár-informatikusok képzésének nemzetközi példái alapján adott javaslatokat arra, hogy a tantervekben hogyan kell bemutatni, hogy a könyvtár, mint hagyományosan nonprofit intézmény hogyan lehet mégis profit-center, milyen információ-szolgáltatásokkal lehet bevételeket generálni korszerű ICT háttérrel. Tanulmányában bemutatja a társ-intézmények tanterveiben alkalmazott elemeket, s részletesebben kitér a speciális “Információ-brokering” vállalkozási forma lehetőségeire. Hazai kutatók 2011-ben részletes tanulmányt közöltek ([2]) a Szegedi Tudományegyetem hallgatói körében végzett vizsgálatokról, amelynek alapvető célja a szándékok, a karrier-elképzelések feltárása volt. A szerzők idézik a Global University Entrepreneurial Spirit Students’ Survey (GUESSS) kutatási projektet, amelyet a svájci St. Gallen egyetem fog össze, több tucat ország több száz egyeteme százezernyi hallgatójának megkérdezésével. A TÁMOP projekt megállapításai szerint a “vállalkozóvá válás” hatásmechanizmusa lokális-kulturális elemeken, személyiségen, kényszerítő tényezőkön (munkanélküliség, helyhez-kötöttség, stb.) és támogató tényezőkön (pozitív minták, szakmai támogatás, stb.) múlik. Itt a megkérdezettek 43%-a a TTK Informatikai Kar hallgatója volt, így az eredmények számunkra érdekesek lehetnek. Az 5 éven belüli karrier-elképzelések 44.5%-a alkalmazotti létet feltételez, 22.1% egyetemi-kutatási jövőt képzel el és 18.6% a közsférában “szolgálna”. Mindössze 4.3%-ban alapítanának “új céget”, s további 1-2% “tovább dolgozna családi cégben”. Ez a hatalmas méretű kulturális tradicionalitás kissé meglepő 2010-2011-ben. Tény az is, hogy mintegy 17.3% gondolja úgy, hogy 5 év tapasztalatszerzés után már bátrabban indítana saját vállalkozást, s 65% válaszolta azt, hogy “már elgondolkodott a vállalkozáson, mint karrier-lehetőségen”.

Egy érdekes mellék-szál az ilyen kutatásokban a nők részvétele. A közismert Global Entrepreneurship Monitor (GEM¹⁰) kutatások keretében is elemezték a női vállalkozók helyzetét, s eszerint minden vizsgált európai országban a nők érintettsége a korai vállalkozási kezdeményezésekben, a megalapított vállalkozásokban és a túlélő, stabil KKV-kben szinte mindenhol fele a férfiakénak. Szerb László ([12]) szerint nálunk a felnőtt népességből 17.1% a férfi és 8.53% a női vállalkozók aránya. A doktoranduszok – mint az innováció jelentős forrásai – körében a nők jelenléte nálunk jelentős hátrányt mutat: a természettudományok, matematika és számítástudományok esetében a szegregáció 37.5%, a mérnöki-termelési és konstrukciós területeken 29.7% a nők aránya, az üzleti tudományokban is mindössze 25%.

⁸ Lásd pl. itt: <http://www.gemconsortium.org/docs>

⁹ Lásd pl. itt: <http://rs1.szif.hu/~raffai/hinfok96.htm>

¹⁰ http://www.gemconsortium.org/download/1201794307439/GEM_2006_Womens_Report_May_07.pdf

Az Európai Bizottság 10 éve gyűjti a vállalkozások számának, összetételének és fejlődésének adatait. A legutóbbi Flash Eurobarometer¹¹ 27 tagállam és 13 további ország elemzése között Magyarországról azt állapítja meg, hogy a megkérdezett felnőttek 55%-a alkalmazotti állásra vágyik, míg 39% inkább vállalkozna. Több adat alaposan eltér az EU átlagtól (zárójelben):

- “szabadság ígérete” helyszínben, időbeosztásban 16% (EU 30% “nomádok”)
- magasabb jövedelem elérése 20% (EU 16%)
- a foglalkoztatás bizonytalanságának elkerülése 11% (EU 3% !)
- unalmas, semmitérvő alkalmazotti munka elkerülése 7% (EU 2%)
- családtagjaimat foglalkoztathatom 4% (EU 1%), stb.

Ebben a környezetben kell meggyőznünk az egyetemi hallgatót arról, hogy érdemes egy kockázatos területre belépnie - miközben azzal, hogy egyetemre jelentkezett, s 3-5 éve ezt az életet éli, már egyszer választott életpályát (hiszen alaposan elhalasztotta a munkába állást, annak minden következményével).

Az EU következő tervezési periódusában több kiemelt irány (pl. a HORIZON 2020 egyes területei) támogatják az ICT fejlesztéseket, a fiatalok munkába állását akár vállalkozások útján, a KKV-k indulását és fennmaradását. Az ún. “Entrepreneurship Action Plan” például egy olyan keretrendszer, amelynek közvetlen célja a vállalkozási kedv élénkítése, az akadályok elhárítása, a vállalkozási kultúra oktatása-terjesztése. A tervezet három pillére (lásd az EU szerverein¹²):

1. Vállalkozási ismeretek oktatása, készségekre irányuló tréningek
2. A KKV-k létrejöttét és fennmaradását szolgáló környezet kialakítása
3. Alulreprzentált társadalmi csoportok bevonása a vállalkozói körbe, támogatásuk a fennmaradási siker érdekében

Egyetemeink szempontjából fontos látni, hogy a fentiekkel kapcsolatos összegeket nem “közvetlenül” az érintett csoportoknak gondolják juttatni, hanem olyan szervezeteknek, amelyek el tudják érni ezeket a csoportokat, képesek képzéseket szervezni és lebonyolítani, s támogatni a készségek folyamatos alakítását és a vállalkozások kezdeti munkáját.

3. A kutatási módszer

Két pilot-felmérést hajtottunk végre annak érdekében, hogy induló modellt tudjunk adni a vállalkozási érettség felmérésének technikájára ebben a környezetben, valamint egy tantervi kísérletet dolgoztunk ki (és hajtottunk végre) Gazdaság-informatika szakos hallgatókkal.

Az egyik vizsgálat olyan hallgatói csoportra irányult, amelynek tagjai diplomások, doktoranduszok, még ki nem alakult kutatási témákkal, de vállalkozási jövőt (is) ígérő innovációs potenciálokkal, kutatási-fejlesztési részeredményekkel és gyakorlattal. Itt elsősorban az érdekelt bennünket, hogy fel tudják-e ismerni az érintett hallgatók fejlesztéseik értékét, s meg tudják-e fogalmazni azokat a szükségleteket, emberi és fizikai erőforrásokat, amelyek ahhoz kellenének, hogy ebből működőképes vállalkozás legyen. A kibocsátott anonim kérdőíveket 9 kutatócsoport tagjai kapták meg. Két online kérdőív készült:

- Az A/ kérdőív szerepe az volt, hogy feltárjunk néhány, a *munkacsoportokra*, finanszírozásukra, vezetési és általános munka-módszereikre vonatkozó tényezőt;

¹¹ No.354 “Entrepreneurship in the EU and beyond”, pl. itt:

<http://ec.europa.eu/enterprise/policies/sme/facts-figures-analysis/eurobarometer/>

¹² http://ec.europa.eu/enterprise/policies/sme/entrepreneurship-2020/index_en.htm

rákérdeztünk a munkavégzés módjára, a helyszín meghatározó szerepére, s megpróbáltuk összevetni mindezeket az eredményességgel, a potenciális vállalkozási lehetőségekkel. A kérdőív első két kérdéscsoportja a munka jellegére, a munkacsoport működés módjára koncentrált, míg a második két csoport a szakmai és üzleti kapcsolatrendszerre, gazdálkodási és piaci ismeretekre, vezetésre kérdez rá.

- A B/ kérdőív egyénenként igyekezett rákérdezni azokra a kompetenciákra, amelyek egy-egy csoportban a szakmai munkán kívül a piacismereti, piac-befolyásolási, gazdálkodási és értékesítési munka megkezdését segíthetik.

A második vizsgálat végzős alap- és mesterképzéses Gazdaság-informatika szakos nappali hallgatókra fókuszált. A pilot felmérés kérdőíve először a tanulmányok és a munkavégzés-vállalkozás párhuzamosságára kérdezett, majd a személyes kompetenciákat vizsgáltuk. Ezután az időgazdálkodás, az üzleti-menedzseri kompetenciák, finanszírozási ismeretek következtek, míg végül négy kérdéscsoportban jövőbeli elképzeléseikről kérdeztük a hallgatókat (tanulmányok alatt és utána). Végül lehetőségük volt tantervi javaslatokra a fentiekkel kapcsolatban.

A kérdőíveket a kis mintanagyságok okán egyszerű gyakorisági és kereszt-táblás elemzésnek vetettük alá, ezekből néhány kiugró megállapítást közlünk a továbbiakban.

4. Az eredmények elemzése, javaslatok

A doktoranduszok projekt-team-jei esetében figyelemreméltónak látjuk a következőket:

- a) A válaszadók mindegyike egyetemi és más finanszírozású K+F projektmunkában, csoportban dolgozik
- b) A csoportok 69%-ban egyszerű projekt-tervezéssel haladnak előre
- c) 54,5% egy-két éves munka után tudott előállítani egy deszkamodellt¹³
- d) 63% nemzetközi szintű, de csak K+F jellegű kapcsolatrendszerrel rendelkezik
- e) 64,5% arányban a csoportban van 1-2 személy, aki használható üzleti gyakorlattal rendelkezik, 27,3% semmilyen, 9,1% valamiféle tanult ismeretekkel dolgozik
- f) 63% ismeri a fejlesztés iparjogi konzekvenciáit
- g) A szervezeti tudás-menedzselés viszonylag jól működik: 27% részletesen dokumentál, 18% ezen felül „explicitte” teszi a tartalmakat, közös rendezvényeken, vitákon
- h) Vállalkozási szempontból jó, hogy a csoportok 72,7%-a vegyes összetételű, magasan kvalifikált szakemberekből áll
- i) 54,5%-ban semmiféle vállalkozási formán még nem gondolkodtak, biztonságos (bár véges) projekt-finanszírozásuk van
- j) 36,7%-ban nincs neve a terméknek, nincs piaci elképzelés, „csak” fejlesztés folyik
- k) 18,2%-nak semmiféle üzleti elképzelése nincs, 54,5% valami alapszintű tervdokumentummal rendelkezik (finanszírozás, marketing)
- l) 18%-nak nincs semmiféle piaci elképzelése – de 63,6% egyértelműen tengerentúli piacra szánja a fejlesztés eredményét

Maguk a résztvevők a második részben nyilatkoztak:

- a) 85,7% beosztottként dolgozik

¹³ Minimal Viable Product: befektetőnek-vevőnek bemutatható, működő termék

- b) 21,4% a megkívántnál jobb-több szakmai kompetenciával rendelkezik, sőt, 17,8% szerint „további területen, ki nem használt kompetenciái vannak”
- c) Vállalkozási szempontból rossz jel, hogy 28% csak „tanulmányok mellett, időnként” tud dolgozni a fejlesztéseken, további 21% más munka mellett tud erre 2-3 napot szánni
- d) 46,4%-nak semmiféle üzleti ismerete-gyakorlata nincs, s csak 7% számol be korábbi, közvetlen üzleti tevékenységről
- e) 60,7%-nak semmiféle finanszírozási ismerete, elképzelése nincs, 21,4% legalább „pályázati ismeretekkel rendelkezik”
- f) Kissé meglepő, de 46+14% alkalmasnak tartja magát vezetőnek, s további 11% „kiemelten” vezetői szerepre „teremtett”.
- g) 4% tartja magát „tárgyalóképesnek”, 64% „inkább távol tartaná” magát ilyesmitől
- h) Ami veszélyes még: 32+25% lényegében nem foglalkozik a (potenciális) megrendelők körével, s további 25% „semleges”, nem érdekli ez a téma
- i) 30-30% „nem rendelkezik”, ill. „esetleg” dolgozik ipari-szakmai kapcsolattal
- j) És végül: 36% nem ismeri a „startup” kérdéskört, további 39% „hallott már erről”.

A kivonatos eredményekből is látható, hogy a kifejezetten kutatás-fejlesztésre orientált csoportok jelen állapotukban nem fogják tudni menedzselni a fejlesztési eredményeket piacra-jutási értelemben. Kérdés, mi történik majd a prototípusokkal, amikor gyártásról, vevőkapcsolatról, piacismeretről kell gondoskodni.

A Gazdaság-informatika szakos hallgatók esetében érdeklődésre tarthatnak számat például az alábbiak:

- a) 59% csak a tanulmányaival foglalkozik, nem vesz részt más munkákban, további 22% hobbiként végez „másfajta szakmai munkát”
- b) Ami fontos: 47% „tudna szakmai irányítással fejlesztőmunkát végezni”, és további 21%-nak „van olyan szakmai részterülete, amiben profi”
- c) Elgondolkodtató, hogy 40% „a tanulás mellett akár 2-3 napot is tudna dolgozni”, s 5%-nak már ajánlottak is ilyen munkát és „akár abba hagyom tanulmányaimat”
- d) 14%-nak „Nincs semmilyen üzleti ismeretem”, 57% a nem túl biztatóan hangzó „volt ilyen tárgyunk...” megjelölést választotta
- e) 67%-nak „semmiféle pénzügyi ismerete nincs”, de legalább 21% „hallott már KKV-k pályázati és EU támogatásáról”
- f) A fentiek után 45% „alkalmas lenne vezetőnek munkacsoportban”, sőt, 10% „kiemelten vezetőnek való személyiségnek” vallotta magát
- g) Tárgyalási és prezentációs készségeket 40-69%-ban tartanak jellemzőnek
- h) 28% nem tudja, mire való a „befektető”, 14% „hallott róla”, 40% „semleges”
- i) Nappali tagozatosoknál életszerű, hogy 38%-ra „nem jellemző” a szakmai ipari kapcsolat, 21%-ban „van ilyen valamiféle ismeret, kapcsolat”
- j) 36% „nem hallott” startup technikákról, 28% „már hallottam róla”
- k) Biztató, hogy csoportmunka területén 45% jónak, 22% kiválónak tartja magát
- l) Számviteli ismeretei 16%-nak nincsenek, 28% minimális szinten áll
- m) 47%-ban „szívesen dolgoznék szakmai projekten tanulmányaim alatt is”, 22% „szakmai téren megállnám a helyem” és 14% „Keresem a vállalkozás lehetőségét tanulás mellett”
- n) és végül: 28% „beosztott informatikus leszek”, 41% „multinál keresek helyet”, 19% „kisvállalkozásnál, családban akarok dolgozni” – s 7% „irány külföld”...

Úgy gondoljuk, hogy a fenti kivonatos ismertetőből is jól kirajzolódik az a hiány, ami a ténylegesen esetleg létező, „üzleti-gazdálkodási” tartalmú tantervi tárgyak „teljesítése” és a vállalkozásokban szükséges kompetenciák között fennáll. A szokásos vállalkozás-fejlesztési tréningek, a jó üzleti iskolák pontosan tudják, mire lenne szükség: marketing, piacismeret, pénzügyiek-finanszírozás, versenytárs-elemzés, szervezetfejlesztési kérdések, üzleti tervezés, csoportmunka és prezentációs készségek, kreativitás-fejlesztő technikák, kommunikációs gyakorlatok, információszerzési készségek.

Néhány javaslat:

- 1) Explicit módon meg kell jeleníteni gyakorlati vállalkozás-fejlesztési ismereteket és készség-fejlesztési eljárásokat tartalmazó kurzusokat, legalább a végzős évfolyamon
- 2) Jól válogatott hazai és külföldi esetekkel kell megismertetni a hallgatókat, találkozni kell „élő” vállalkozókkal
- 3) Az ICT piac és mikro-vállalkozások speciális ismereteit és gyakorlatát be kell mutatni
- 4) Önbevallásos és más személyiség- és kompetencia-felmérésekkel tudatosítani kell az ismerethiányokat, a vállalkozási inkompetencia szintjét
- 5) Időt kell szánni szimulált vállalkozás-indítás gyakorlására, lehetőleg vegyes szakmai összetételű csoportokban (közgazdász + mérnök + informatikus + bölcsész?)
- 6) Országos szinten egyeztetett, standard módon kell kompetencia-szintfelmérést végezni, s közösen, szakmai szervezetekkel kidolgozni tartalmi-formai megoldásokat.

Több egyetemen indultak már “vállalkozási ismeretek” és hasonló kurzusok, de akár még a közgazdasági karokon is kuriózum lehet, hogy közvetlen készségeket, vállalkozás-alapítást támogató-fejlesztő kurzusra tesznek javaslatot. A PTE KTK keretében a mesterszakos Gazdaság-informatika csoport a Projektmunka I. keretében helyi mikro-vállalkozások gazdálkodási-informatikai problémáin dolgozott szigorú projektmenedzsment keretek között, majd a Projektmunka II. során igyekeztek megszerzett készségeiket (most már egyénileg) szakdolgozati munkává fejleszteni. Sajnos, a zsúfolt „akadémiai” tantervben ennél többre nincs idő: ha a diák kreditet nem kap, akkor igencsak vonzó (és ingyenes) tartalmat, formát kell kitalálni. Az ilyen kurzusok „nem olcsók”: igen sok tutori felkészülés szükséges, külsővel kell dolgozni, vezetni kell az ötletelő-beszámoló értekezleteket (a Kar két oktatót rendelt a csoporthoz, s négy további, külső mentorral dolgoztunk). S nem utolsósorban: minden tutornak, mentornak „hitelesnek” kell lennie, sokoldalú üzleti-projektvezetési-vállalkozási tapasztalattal...

5. Összefoglalás és nyitott kérdések

A hallgatói hiányok világosan látszanak. Szóban, irányelvekben, stratégiai tervekben mindenhol és mindenki beszél a kisvállalkozások szükségességéről, az innovációs rangsorokban elfoglalt szégyenletes helyzetünkről. Látnunk kell, hogy amennyiben nem tudjuk megcélozni az egyetemeken munkára fogható, vagy az onnan éppen kilépő réteget, a kreatív életkorban lévő fiatalokat, akkor mindez nem fog sikerülni: 50-60-éves kényszervállalkozóktól ne várjunk jelentős innovációkat – főleg az ICT területeken nem.

A felsőoktatásba most bekerülő „Z-generáció” igényei, ismeretszerzési módszerei és tanulási hajlandósága merőben más lesz, mint amit a konzervatív egyetemek megszoktak. Az ICT területeken igen szemléletes példák állnak a „digitális bennszülöttek” előtt- ki kellene használnunk ezt az érdeklődést, aktivizálnunk kell ezeket az energiákat.

A legjobb szándékok mellett is vannak súlyos, **nyitott kérdések** – megválaszolásuk viták, szakértői elemzések, kari-egyetemi döntések feladata kell legyen:

- a) Hol vannak a vállalkozási ismeretek oktatására képes felsőoktatási oktatók az érintett karokon? Életpályájuk jelenlegi követelményei tudományos publikációk és könyvek „gyártására” kötelezi őket, idejük és módjuk sincs ipari-vállalati gyakorlati munkák kivitelezésére, s az ezek egyetemi elismertetése helyett kapott kritikák elviselése sem egyszerű.
- b) Ha az oktató nem méreti meg tudását gyakorlati munkákban – miért is adna ilyesmit hallgatóinak? A Bologna-BA/BSc deklarált célja a gyakorlatias képzés volt – a tantervi eredmények, a képzési kimenet közismert. A mesterszakost már a tudomány felé szorítják és a PhD programok egyértelműen kutatási-publikálási orientációjúak. Hiába készít innovatív terméket, szoftver-alkalmazást a doktorandusz, erre nem fog doktori címet kapni.
- c) Vajon a felsőoktatás a jó környezet a vállalkozói képzésekre, vállalkozási gyakorlóterepre, inkubációs központok kialakítására, vagy ez inkább a felnőttképzések, a kamarák, szakértő magánvállalkozások és befektetői társulások feladata?
- d) Az EU 2020-ig tervezett HORIZON és más programjai (lásd pl. az EU website-okon¹⁴) elkülönített alapokat biztosítanak KKV-k fejlesztésére, s mindez jól kombinálható az ICT + innovációs szándékú alapokkal, valamint a fiatal kori munkanélküliség leküzdésére irányítandó összegekkel. Ezeket a pályázatokat meg kell ismerni, fel kell tárni, segíteni kell a hallgatókat – milyen egyetemi szervezetek fogják elvégezni ezt a munkát? S milyen referenciákat tudnak felmutatni az egyetemek a sikeres pályázáshoz - szakértelem, szervezetek, speciális kurzusok, hallgatói csapatok, ipari kapcsolatok?
- e) Melyik tagozaton, milyen módon építhetők be ezek az ismeretek a zsúfolt tantervekbe? Nyugaton tipikus az MBA-beágyazás, hiszen itt mérnökök, agrár, bármilyen műszaki, a termelésben, vagy akár a közigazgatásban dolgozó, nem-közgazdasági végzettségűek iratkoznak be az MBA kurzusokra, szöges ellentétben a hazai MAB-előírásokkal.
- f) Nyilvánvaló, s felmérésekkel alátámasztott, hogy az „alkalmazotti” lét és életpálya messze más készségeket és képességeket követel, mint a vállalkozói. Milyen módon „adagoljuk”, s milyen kurzusokban e kétféle tudásanyagot?

Természetesen a tantervi változások nem rendelkeznek el felülről. Segíthet a kamara, a miniszteriális vezetés, a kancellári rendszer által diktált gazdasági kényszerűség – mindezek elmozdíthatják a szigorúan poroszos, tantermi tudásközvetítő, tankönyv-centrikus képzést egy fokkal praktikusabb irányba. Mindennek azonban sok hátulütője van, s életveszélyes kísérlet lenne a felsőoktatást egészében valamiféle szakfőiskolai – tanfolyami szintre szorítani holmi aktuális gazdasági szirénhangok nyomására. Szakmunkások, vagy betanított call-centeres ügyintézők seregének kiképzése nem lehet az egyetemek feladata.

Irodalomjegyzék

- [1] Sabrina Di Addario - Daniela Vuri: Entrepreneurship and Market Size: The Case of Young College Graduates in Italy, *Leibniz Information Centre for Economics*, Discussion paper series, No. 5098 July 2010

¹⁴ HORIZON 2020 http://ec.europa.eu/enterprise/policies/sme/files/entrepreneurship-2020/final-report-pub-cons-entr2020-ap_en.pdf

- [2] Bajmócy, Zoltán – Imreh, Szabolcs: A Szegedi Tudományegyetem hallgatói, mint potenciális tudás-intenzív vállalkozók sajátosságai, *TÁMOP-4.2.1-08/1-2008-0002 projekt, B/2*, 2011.
- [3] Charney, A. – Libecap, G. D.: The Impact of Entrepreneurship Education: An Evaluation of the Berger Entrepreneurship Program at the University of Arizona, 1985-1999; Karl Eller Center No. 85721, November 29, 2000, University of Arizona, Tucson
- [4] EUROBAROMETER Survey on Entrepreneurship, 2012, letöltés: <http://ec.europa.eu/enterprise/policies/sme/facts-figures-analysis/eurobarometer/>
- [5] European Commission: Mini-companies in Secondary Education, letöltés: http://europa.eu.int/comm/enterprise/entrepreneurship/support_measures/training_education/doc/mini_companies_en.pdf, (2005)
- [6] Flash Eurobarometer 354, Entrepreneurship: COUNTRY REPORT HUNGARY, June 2012
- [7] (ed. Silvia Ganzerla): Evaluation on Policy: Promotion of Women Innovators and Entrepreneurship, Final Report, EC DG Enterprise and Industry, *E E C (GHK, Technopolis)* ENTR/04/093-FC-Lot 1, 25 July 2008
- [8] F. J. Greene – G. Saridakis: Understanding the Factors Influencing Graduate Entrepreneurship , Research Report 001/2007, National Council for Graduate Entrepreneurship, Centre for Small and Medium Sized Enterprises, Warwick Business School University of Warwick, March 2007
- [9] Hsu, David H. – Roberts, Edward B. – Eesley, Charles E.: Entrepreneurial Ventures from Technology-Based Universities: Evidence from MIT, June 2008, *Wharton School of Business*
- [10] Kiszl, Péter: Business: informatikus könyvtáros hallgatóknak – A vállalkozói, üzleti információ oktatása a felsőfokú könyvtáros-képzésben, *Tudományos és Műszaki Tájék.*, 51./11, 2004
- [11] Raffai, Mária: *Társadalmi-gazdasági elvárások az informatikus szakemberekkel szemben.* IF'1996 Konferencia, Debrecen
- [12] Szerb, László: Women entrepreneurs and firms owned by women in Baranya county 2001-2004, University of Pécs, 2007
- [13] Szirmai, Péter - Csapó, Krisztián: Gyakorlati vállalkozásoktatás, OFI, 2009. jún. 17.
- [14] Wang, Clement K. – Wong, Poh-Kam: Entrepreneurial interest of university students in Singapore, *Technovation* 24 (2004) 163–172; www.elsevier.com/locate/technovation

Algoritmuskötő-készség újszerű vizsgálata

A new kind of examination of algorithm component skill

Godó Zoltán Attila^a, Kocsis Dénes^b, Kiss Gábor^c, Stóka György^d

^a University of Debrecen, Faculty of Informatics – Department of Information Technology, Debrecen, Hungary
magortalto@gmail.com

^b University of Debrecen, Engineering Faculty - Department of Environmental Engineering, Debrecen, Hungary

^c University of Óbuda, Institute of Machine Design and Safety Engineering, Budapest, Hungary

^d Eszterházy Károly College, Faculty of Comenius - Department of Informatics, Sárospatak, Hungary

Absztrakt: Az informatikaoktatás teljes vertikumában tetten érhető, hogy az egyes hallgatók algoritmizáló képessége igen nagy differenciát mutat. Ez kihat a programozás készség szintű elsajátítására és végső soron az eredményes oktatásra egyaránt. Mint minden készségben genetikai, szociális és tanulási folyamatok egyaránt, de különböző mértékben szerepet játszhatnak. Az algoritmuskötő-készség már kisgyermekkorban megfigyelhető túlnyomóan öröklött összetevők révén, amelyet az életkor előrehaladtával egyre több szerzett készség módosít. Célul tűztük ki egy olyan tesztrendszer, hardver és szoftver kidolgozását, amely már az írástudás megjelenése előtt képes mérni az algoritmuskötő-készséget, valamint ugyan ezen rendszerrel mérhetőek az alanyok egészen az egyetemi szintig. A teszt univerzalitása a különböző életkorokban teljes statisztikai összehasonlíthatóságot tesz lehetővé. A rendszer az általunk definiált ún. fényprogramozás elvére épül, amelyhez célorientált hardvert és szoftvert fejlesztünk

Kulcsszavak: algoritmus, készség, fényprogramozás

Abstract: In the practice of IT higher education, we can see big differences in algorithmisation ability of the students. It affects the acquisition of programming and the successful education as well. Just like in every skill, genetic, social, and learning progressions all play part, but in different rates. The algorithm component skill can be discovered even in little childhood, mostly because of inherited things, which is modified by various other skills by the time later. Our aim is making a test system, software and hardware, which can measure the algorithm component skill before the literacy, and which can be used right until university level. The universality of the system makes statistical comparability completely possible in different ages as well. The system is built on the theory of light-programming (defined by us), which we are developing aim oriented software and hardware to.

Keywords: algorithm, ability, light-programming

1. Bevezetés

Az informatikaoktatás teljes vertikumában tetten érhető, hogy az egyes hallgatók algoritmizáló képessége igen nagy differenciát mutat. Ez kihat a programozás készség szintű elsajátítására és végső soron az eredményes oktatásra egyaránt.

A mindennapi életben, folyamatosan algoritmusokat készítünk és hajtunk végre, tevékenységsorozatokot, információ-áramlási folyamatokat tervezünk, melynek megértéséhez tisztában kell lenni annak elemi felépítésével [2]. A problémamegoldó készség fejlesztését az oktatás kiemelten fontos feladatának kell tekintenünk. Ennek eszközüül hagyományosan elsősorban a matematika szolgált hosszú időn keresztül. Az informatikaoktatásának bevezetésével azonban egy olyan újabb lehetőség jelent meg, amely sok esetben kevésbé

igényli az absztrakt gondolkodást, ugyanakkor a problémák és a megoldások tekintetében is lényegesen kézzelfoghatóbb [4].

Mint minden készségben genetikai, szociális és tanulási folyamatok egyaránt, de különböző mértékben játszhatnak szerepet. Az algoritmusalkotó-készség már kisgyermekkorban megfigyelhető túlnyomóan öröklött összetevők révén, amelyet az életkor előrehaladtával egyre több szerzett készség módosít. A differencia azonban az oktatás minden szintjén szembetűnő [5]. Egyes hallgatók -akár a felsőoktatást tekintve is- szinte azonnal ráéreznek és hamarosan bámulatos dolgokat alkotnak, mások olyan elemi szintű algoritmusokat is képtelenek megérteni, amelyet egyes kisiskolások is könnyedén megoldanak. Az algoritmus alkotása összetett, magas szintű idegrendszeri tevékenység eredője. Az elemzés során a célhoz vezető lépéssorozatot valamint az egyes részfeladatokat is értelmezni kell, amely együtt építi fel a megoldást[6].

Célul tűztük ki egy olyan tesztrendszer, hardver és szoftver kidolgozását, amely már az írástudás megjelenése előtt képes mérni az algoritmusalkotó-készséget, valamint ugyan ezen rendszerrel mérhetőek az alanyok egészen az egyetemi szintig. Ezért rendkívül egyszerű eszköztrendszert kerestünk, amely minél kevésbé függ a képességek tanult összetevőitől és a rövidtávú memóriától (short term memory - STM). Kutatásunk tervezésénél kiemelt prioritást szántunk a didaktikai szempontból alapvető jelentőségű nyomon követésnek [1]. Így ugyan azon tesztalannyal különböző életkorokban el kívánjuk végeztetni a tesztünket, a hozzá kapcsolódó egyéb dedikált tesztekkel. A tesztrendszer univerzalitása a különböző életkorokban teljes statisztikai összehasonlíthatóságot tesz lehetővé. [3]. A rendszer az általunk definiált un. fényprogramozás elvére épül, amelyhez célorientált hardvert és szoftvert fejlesztünk

2. Életkor független problémamegoldás

Az első, fényvel és hanggal megvalósított teszter a „Simon mondja”, 1978-ban Ralph H. Baer és Howard J. Morrison ötlete alapján, Assembly nyelven készült el [8]. A játék sikerével a '80-as évek pop szimbólumává nőtte ki magát. „Do what Simon Says: Follow the pattern of lights and sounds for as long as you can... If you can!”. Az egymás után felvillanó fényeket kell a játékosnak egyre hosszabban utánoznia. Azonban Simon csak a memóriát teszteli, habár nem utolsó sorban fejleszti is azt. Erre épült a Cogmed agy stimulációs programcsomagja, amellyel ADHD szindrómás (figyelemhiányos hiperaktivitás-zavar: Attention Deficit Hyperactivity Disorder), gyermekeket is sikerrel kezelnek [9]. Az algoritmizáló készség vizsgálatánál azonban tovább kellett lépniünk a memóriateszten és a kreatív problémamegoldást kell vizsgálnunk.

Olyan tesztrendszert kívántunk fejleszteni, amely már kisgyermekkorban képes mérni az algoritmusalkotó-készséget, azonban a felnőtt korosztály is ugyan ezzel az eszközzel vizsgálható. Az így nyomon követett alanyok fejlődése, öröklött és szerzett készségeinek elemzése új távlatokat nyithat az algoritmizáló képesség jobb didaktikai megértéséhez.

Az óvodás korúak és kisiskolások nem tudnak még írni/olvasni olyan szinten, amely a programozáshoz szükséges. Nem is várhatjuk, hogy programot fejlesszenek, de az algoritmikus gondolkodásra való hajlam sokukban már adott. Színes fényforrásokat kigyújtani és eloltani azonban írás nélkül, nyomógombokkal is lehetséges. Ha pedig ezt tervszerűen felépítve egymás után teszik, ugyan azt érik el mintha programot írnának rá. Mindössze a megfelelő hardvert kell kialakítani hozzá, amely a gombnyomás variációkat memóriába helyezi és képes visszajátszani azokat. A fényforrásként alkalmazott LED-ek felgyulladását

vezérlő gombnyomás variáció ez esetben a programmal, a visszajátszás a futtatással egyenértékű. A feladat rendkívül látványos, szórakoztató és játékos, amely kihívást jelent a teljes vizsgálandó életkor intervallumban. A digitális alapú, játékos oktatási rendszereknek kiemelt hatékonyságot tulajdonít a szakirodalom [7].

3. Bináris gondolkodás

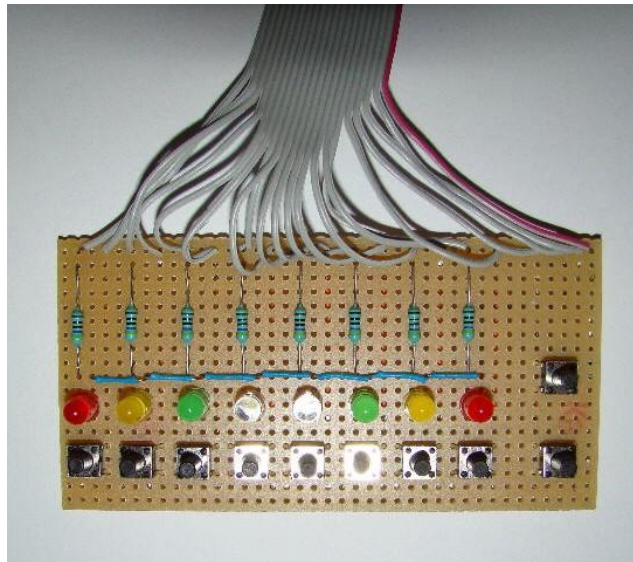
Az informatikaoktatás egy fontos területe a digitális technika elsajátítása. Itt jobbra bináris értékekkel dolgozunk, amelynek megértése még nagyobb szórást eredményez a tanulók között. Magasabb oktatási szinten olyan program elkészítése a cél, amely egy bináris variációt állít elő vagy egy bináris bejövő jelet dolgoz fel. A bináris jelrendszer azonban közvetlenül értelmezhető a LED-ek emittáló és sötét állapotának. A cél, hogy képes legyen egy végeredmény elérését lépésekre bontani, tehát algoritmizálni. Habár a magas szintű programozás egy jóval összetettebb gondolkodást igényel, a végeredmény tulajdonképpen ugyan az, vagyis hogy a LED-ek adott variációban gyulladjanak ki (bináris 1 / 0 megértésére történő rávezetés). Az így vizualizált adatfolyam látványosan jelzi, hogy a vizsgált személy milyen szinten képes az adott feladatot megoldani. Magasabb oktatási szinten ezeket a kombinációkat decimálisan értelmezve megalkothatja a tanuló az ezt létrehozó kódot. Így a fényprogramozás szoros analógiát mutat a digitális technika didaktikai problémakörével.

4. Hardver

A rendszer két részből épül fel. Egy kezelő felületből, ahol a tesztelő vezérli és ellenőrzi a feladatokat valamint visszakapja az eredményeket. A mérési oldal a vizsgált személyé, ahol tetszőleges fényvariációt tud létrehozni, eltárolni és visszajátszani.

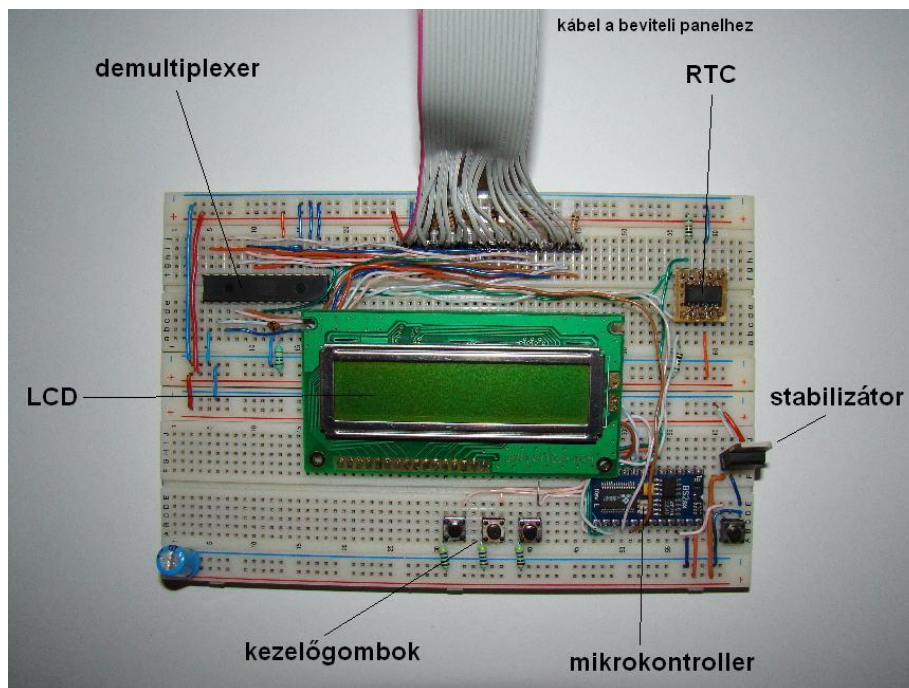
A mérési oldalon 8 LED és előtte 8 nyomógomb található. Az egyes nyomógombokkal a hozzá tartozó LED-eket tudjuk be illetve ki kapcsolni tetszőlegesen, korlátlan próbálkozási lehetőséggel. A velük párhuzamosan elhelyezkedő nyomógommbal a kialakított bináris variációt tudjuk memóriába helyezni. A legfelső nyomógommbal pedig a letárolt mintákat időrendben, ismétlődően futtatni, azaz a feladatot „beadni”.

A kezelőfelület elektronikáját szándékosan nem rejtettük műszerdobozba. Célunk volt a nem virtuális eszközök használata. Az alkatrészek látványának, a hozzányúlás megszokásának kialakulása. (lásd 1. ábra).



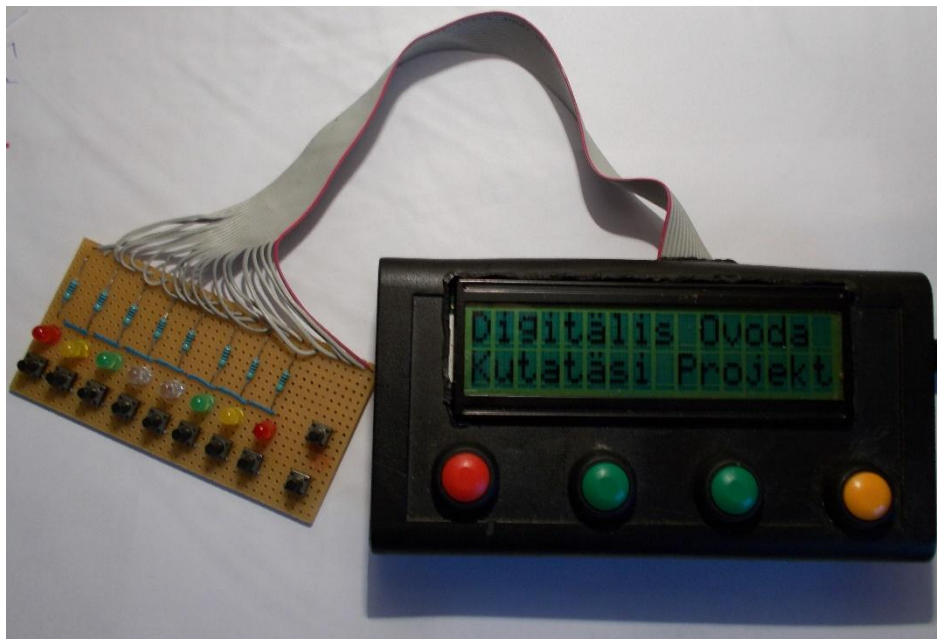
1. ábra. A teszter mérési felülete

A kezelő felület 16 I/O lábbal rendelkező, RISC processzoros, BS2SX mikrokontrollerre épül. A perifériák sokszorozását kétirányú, digitális, I2C kommunikációs alapú, MCP23016 demultiplexer végzi. A feladatok időzítéséhez és az értékelési adatokhoz PCF8583 századmásodperc pontosságú, I2C buszos RTC-t alkalmaztunk. A kijelzést 4x20 felbontású, karakteres LCD panel végzi (lásd 2. ábra).



2. ábra. Deszka modell, a rendszer felépítése

Végül műszerdobozt kapott. Külső és beleső tápellátással, PC felé adat porttal (lásd 3. ábra).



3. ábra. Műszerdobozba építve

5. Mérés

Első lépés a tesztalany lehetőségeinek megismerése. A LED-ek fel, le kapcsolása, a minta elmentése és visszajátszása.

5.1. Minta utánpótlás

A rendszer bemutat egy statikus mintát egy adott ideig, majd a bemutatás megszűnik. Ezután elindul a mérés és az alanynek le kell másolnia a korábban látott mintát. A feladatok előrehaladtával egyre összetettebb minta mutatható be. A bemutatás időtartama megváltoztatható, amely nagyban nehezíti a feladat megoldását.

Számszerűsíthetően mérhető a megoldásban lévő hibák száma, a megoldás „beadásáig” eltelt idő, a megoldás során történő hibás próbálkozások száma valamint az összes lenyomott billentyűk száma a megoldásig.

A mérés a bináris minta leképzésének és visszaadásának képességét hivatott vizsgálni. Kevésbé orientál a vizuális memóriára, hiszen minden feladat összesen csak 8 fényből felépülő mintát használ.

5.2. Minta követés

A rendszer egy időben változó fényjátékot mutat be. A feladat, hogy a tesztalany a mintát lépésekre bontsa és leprogramozza.

Az első minta egy egyszerű fény átfutás az egyik oldalról a másikra. Ezután fokozatosan egyre bonyolultabb feladatok jöhetnek egészen a 20-30 lépéses, összetett variációkig. Például két irányból összefutó és egymáson áthaladó, egymástól visszapattanó, több fényből álló csóvát húzó, több pontból induló, egymást kioltó stb. fényjátékok. A tesztek száma 100-as nagyságrendig terjedhet (lásd 4. ábra).



00000001 = 1	11000000 = 192
00000011 = 3	10000000 = 128
00000110 = 6	11000000 = 192
00001100 = 12	01100000 = 96
00011000 = 24	00110000 = 48
00110000 = 48	00011000 = 24
01100000 = 96	...

4. ábra. Egyszerű feladat implementálása

Két pontból ellentétes irányba kiinduló-növekvő és a szélekhez érve ismételtlen csökkenő méretű, összetett mintakombináció megvalósítása összetett problémamegoldó készséget igényel:

00100100 = 36
01111110 = 126
11100111 = 231
01000010 = 66
00100100 = 36
00011000 = 24
00111100 = 60

...

Objektíven mérhető a megoldásban lévő hibák száma, a megoldás „beadásáig” eltelt idő, a megoldás során történő hibás próbálkozások száma valamint az összes lenyomott billentyűk száma a megoldásig.

6. Kiértékelés

Az adatok a mikrokontroller saját memóriájában tárolódnak. SPI buszon keresztül kiolvashatóak és PC-n feldolgozhatóak. A statisztikai kiértékelés így széleskörűen elvégezhető. A fényprogramozás tesztből igen sok, sokféle és rendkívül pontos, automatikusan mentett adat feldolgozására van lehetőség. A különböző lépés és hibaszámokon kívül századmásodperces pontosságú idő adatok állnak rendelkezésre.

Az adatok összehasonlításra kerülnek ugyan azon alanyokon elvégzett pszichológiai tesztekkel. Kisgyermeknél az iskolaérettségi, motilitás, agresszivitás, kreativitás tesztek létjogosultságát vizsgáljuk, idősebb korosztálynál a különböző intelligencia tesztek. Ezek eredményeinek korrelációját kívánjuk összevetni a saját tesztrendszerünk méréseivel.

A kutatás jelenleg a szoftver fejlesztésénél valamint a pedagógiai és pszichológiai kiértékelési modell kidolgozásánál tart.

Irodalomjegyzék

- [1] Bakonyi Anna: Az óvodás gyermek fejlődésének nyomon követése, a kompetencia alapú óvodai programcsomag gyermekek megfigyelésén alapuló mérési és értékelési rendszere, Educatio Társadalmi Szolgáltató Nonprofit Kft (2009).

- [2] Cláudio Amorim: Beyond Algorithmic Thinking: An Old New Challenge for Science Education. Eighth International History, Philosophy, Sociology & Science Teaching Conference, July 15 to July 18, 2005, University of Leeds, England.
- [3] Eilers, Merry L.: Older adults and computer education: "Not to have the world a closed door.", *International Journal of Technology & Aging*, Vol 2(1), (1989), 56-76.
- [4] Geda Gábor, Hernyák Zoltán: Algoritmizálás és adatmodellek, *Educatio Kht.*, (2011)
- [5] James A. Kulik, Chen-Lin C. Kulik, Robert L. Bangert-Drowns: Effectiveness of computer-based education in elementary schools, *Computers in Human Behavior* V. 1, I. 1, (1985), P. 59–74
- [6] Juraj Hromkovic: *Algorithmic Adventures – From Knowledge to Magic* (2009), Springer.
- [7] Marina Papastergiou: Digital Game-Based Learning in high school Computer Science education: Impact on educational effectiveness and student motivation, *Computers & Education* V. 52, I. 1, (2009), P. 1–12
- [8] Ralph H. Baer and Howard J. Morrison: *Simon Says*, Marvin Glass & Associates (1980), US4207087(A)
http://worldwide.espacenet.com/publicationDetails/biblio?CC=US&NR=4207087&KC=&FT=E&locale=en_EP
- [9] Toni Clarke: Brain fitness seen as hot industry of the future (2008), Reuters,
<http://www.reuters.com/article/2008/03/12/us-brainfitness-idUSN1218668920080312?feedType=RSS&feedName=healthNews&pageNumber=1&virtualBrandChannel=0>

Szövegbányászati módszerek alkalmazása távoktatási fórum üzenetek elemzésében

Application of text mining methods in the analysis of forum messages of distance education

Nyéki Lajos

Széchenyi István Egyetem, Tanárképző Központ

nyeki@sze.hu

Absztrakt: A Széchenyi István Egyetem 2004 óta foglalkozik e-learning alapú távoktatással, azóta a Coedu e-learning távoktatási keretrendszert használjuk. A távoktatási tananyagokat a Coeditor programmal készítjük. A Coedu1 szerver a tananyagokat teszi elérhetővé, a Coedu2 szerver feladata a vizsgák lebonyolítása. Az e-learning távoktatási keretrendszer alapvető részei a tananyag-megjelenítő felület, a belső levelező alrendszer és a fórum. A tananyag-megjelenítő felület biztosítja a felhasználó jogosultságának megfelelő kurzusok elérését. A belső levelező alrendszer biztonságos levelezést tesz lehetővé a regisztrált felhasználók számára. A fórum a tanulással kapcsolatos problémák megbeszélésére szolgál. A fórum tantárgyhoz kapcsolódik, automatikusan hozzá vannak rendelve a tárgy tutorai és hallgatói. A fórum résztvevőinek joga van témát létrehozni és az egyes témákhoz hozzászólásokat küldeni. A fórum üzenetek tartalmának elemzése lehetőséget ad az egyetem vezetése számára a tipikus tanulási problémák megismerésére, a tutorok és a távoktatási központ tevékenységének ellenőrzésére, valamint a tananyagok minőségének, használhatóságának értékelésére. A fórum üzenetek tartalmának elemzésére az SPSS Modeler adatbányászati szoftver Text Analytics szövegbányászati alkalmazását és az ahhoz készített, a magyar nyelvű szövegek feldolgozását lehetővé tevő CLEMTEXT modult használtuk fel. A kutatás során létrehoztuk a fórum üzenetek elemzéséhez szükséges szótárt. Ez teszi lehetővé a fogalmak azonosítását, a kategóriák és a klaszterek meghatározását. A szövegben rejlő összefüggések feltárásához magyar nyelvű vélemény szótárt készítettünk. A részletes szöveg-kapcsolati elemzés elvégzése még további kutatást igényel.

Kulcsszavak: szövegbányászat, e-learning, távoktatás, SPSS Modeler

Abstract: Szechenyi Istvan University deals with e-learning based distance education since 2004, thence we use the Coedu e-learning distance education framework. We prepare e-learning curricula with the Coeditor software. Curricula are reachable on the Coedu1 server, the task of the Coedu2 server is the transaction of exams. The fundamental parts of the e-learning distance education framework are the curriculum indicator surface, the inner mail subsystem and the forum. The curriculum indicator surface authorizes the access of courses according to the authority of the user. The inner mail subsystem enables a secure correspondence for the registered users. The forum makes for discussion of problems connected to learning. The forum joins to the course, the tutors and the students of the course are automatically assigned. The participants of the forum have right to create a topic, and send messages to the topic. The analysis of the content of forum messages enables the leaders of the university to get acquainted with the typical learning problems, to control the activity of tutors and distance education centre, and to evaluate the quality and usability of curricula. We have used for the analysis of the content of forum messages the Text Analytics text mining application of the SPSS Modeler data mining software, and the CLEMTEXT module made for it enabling to process Hungarian texts. During the research we have created a dictionary necessary for the analysis of forum messages. This allows the identification of concepts, and the definition of categories and clusters. To explore the potential relationships residing in the text we have prepared a Hungarian opinion dictionary. To carry out a detailed text-link analysis requires further research.

Keywords: text mining, e-learning, distance education, SPSS Modeler

1. Bevezetés

A Széchenyi István Egyetem 2004 óta foglalkozik e-learning alapú távoktatással, azóta a Coedu e-learning távoktatási keretrendszert használjuk. A távoktatási tananyagokat a Coeditor programmal készítjük. A Coedu1 szerver a tananyagokat teszi elérhetővé, a Coedu2 szerver feladata a vizsgák lebonyolítása.

A távoktatási kurzusok általában moduláris szerkezetűek. A modulok leckékből állnak, a lecke végén önellenőrzésre szolgáló feladatsorral. A lecke záró feladatsor többször is kitölthető, ennek eredményét nem rögzíti a keretrendszer. A modul végén modulzáró feladatsor teszi lehetővé a tutor és a hallgató számára az előrehaladás értékelését. A modulzáró feladatsor csak egyszer tölthető ki, a kitöltés eredményét a Coedu1 szerver tárolja.

A távoktatás a győri és a budapesti konzultációs központok köré szerveződik. Bizonyos, feladatmegoldást igénylő tárgyakból (pl. matematika, mechanika, fizika) csoportos konzultációkat is biztosítunk. A keretrendszerben elérhető tananyagokat a távoktatásos képzés mellett a levelező oktatásban is használjuk.

A távoktatásos hallgatók a Proszeminárium tantárgy keretében sajátítják el a Coedu keretrendszer használatát. Ennek során kötelező feladatokat kell megoldaniuk: levelet kell írniuk a tutornak, fórum témakört és hozzászólást kell létrehozniuk és ki kell tölteniük a modulzáró feladatsort. A kutatás során feldolgozott 18147 fórum üzenetből 7543 (ez az összes üzenet 42 %-a) a Proszeminárium kurzushoz kapcsolódik.

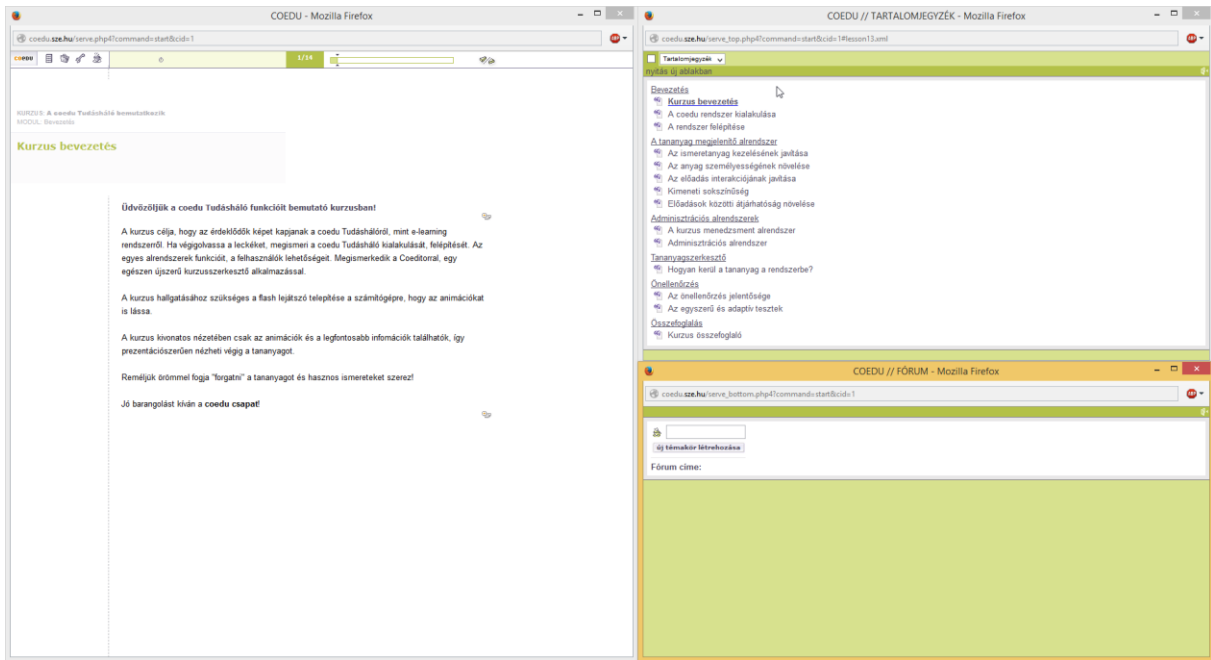
A Coedu távoktatási keretrendszer 2008 óta Creative Commons Open Source. Működtetéséhez nincs többé támogatás. Emiatt kurzusainkat fokozatosan SCORM kompatibilissé kell alakítanunk, hogy át tudjunk térni a Moodle vagy a napjainkban egyetemünkön EU támogatással fejlesztés alatt álló HunLine keretrendszer használatára.

A fórum üzenetek tartalmának elemzése lehetőséget ad az egyetem vezetése számára a tipikus tanulási problémák megismerésére, a tutorok és a távoktatási központ tevékenységének ellenőrzésére, valamint a tananyagok minőségének, használhatóságának értékelésére. A hallgatói problémák általában a távoktatási keretrendszer működésével, bizonyos tananyagrészek meg nem értésével, a modulzáró feladatok megoldásának módjával, a távoktatási központ tevékenységével és a tutorok munkájával kapcsolatosak.

2. A Coedu keretrendszer

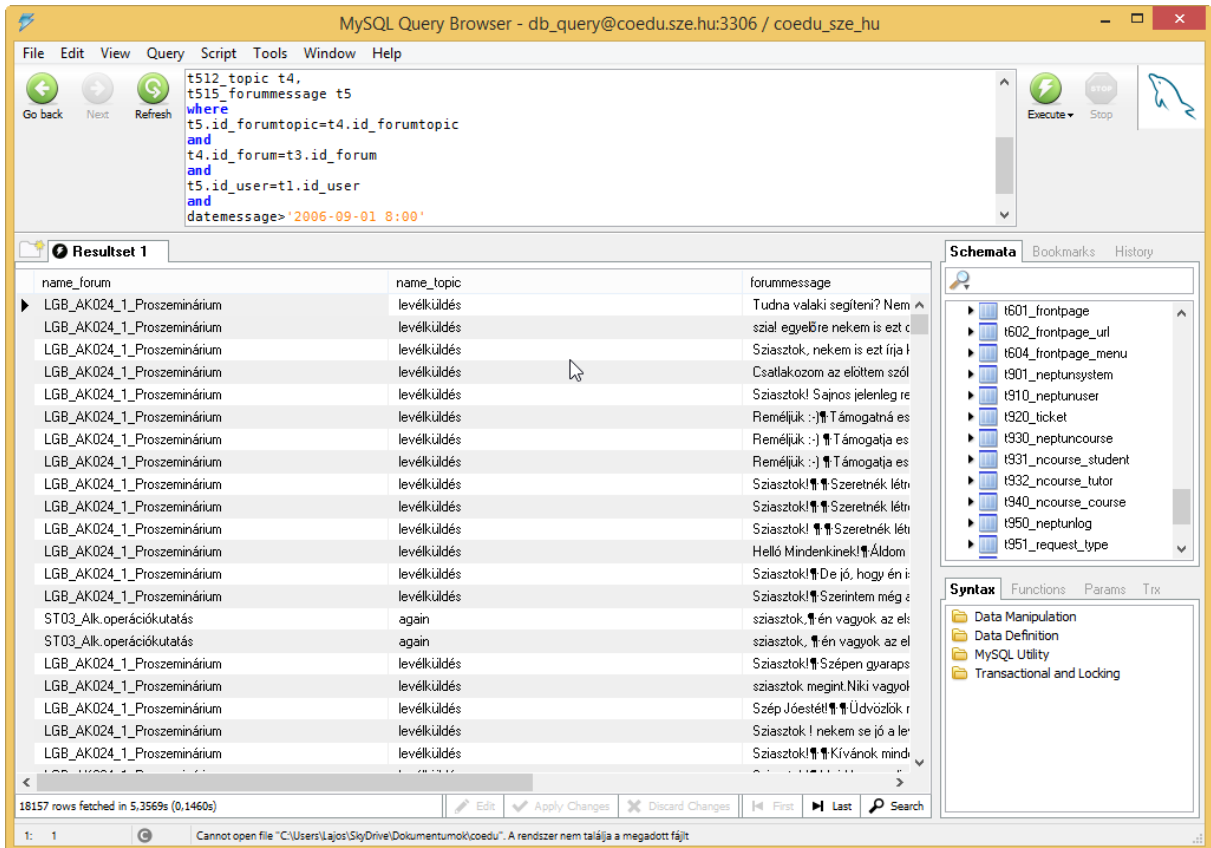
Az e-learning távoktatási keretrendszer alapvető részei a tananyag-megjelenítő felület, a belső levelező alrendszer és a fórum. A tananyag-megjelenítő felület biztosítja a felhasználó jogosultságának megfelelő kurzusok elérését. A belső levelező alrendszer biztonságos levelezést tesz lehetővé a regisztrált felhasználók számára. A fórum a tanulással kapcsolatos problémák megbeszélésére szolgál. A fórum kurzushoz kapcsolódik, automatikusan hozzá vannak rendelve a tantárgy tutorai és hallgatói. A fórum résztvevőinek joga van témát létrehozni és az egyes témákhoz hozzászólásokat küldeni.

A Coedu keretrendszer tananyag-megjelenítő felületén a bal oldali keretben látható a tananyag, a jobb felső keretben van a kurzus tartalomjegyzéke, a jobb alsó keretben helyezkedik el a fórum. Az egyes keretek átméretezhetőek (lásd 1. ábra).



1. ábra: A Coedu keretrendszer tananyag-megjelenítő felülete

A fórum üzeneteket a Coedul szerver tárolja, az adatbázis csak Cisco VPN klienssel érhető el. A Coedu adatbázist a MySQL Query Browser használatával kérdezzük le (lásd 2. ábra).



2. ábra: A Coedu adatbázis lekérdezése

A lekérdezés eredményét csv, xml, html vagy xls szövegfájlba exportáljuk. A feldolgozáshoz a csv szövegfájlt célszerű felhasználni.

A fórum üzeneteket tároló adattábla szerkezetét a 3. ábra mutatja. Az első mező a fórum neve, a második a témakör, a harmadik a fórum üzenet, a negyedik a dátum és az ötödik a login név.

name_forum	name_top	forummessage	datumszall	loginname
LGB_AK024_1_Proseminarium	levélküldés	Tudna valaki segíteni? Nem tudok a levelezőbe belépni mert mindig azt a hibáüzenetet írja ki. HIBA: A kapcsolatot az imap szerver eldobta. Kovács...	2006-09-26 16:14:23	e-ZKL2YX
LGB_AK024_1_Proseminarium	levélküldés	Szia! egyelőre nekem is ezt dobja be, szerintem hamarosan megoldódik a probléma. Mészáros Zoltán	2006-09-26 16:24:58	e-P3W3CX
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok, nekem is ezt írja ki, remélem holnapra redbe lesz, viddák mária	2006-09-26 16:39:52	e-XOF5HF
LGB_AK024_1_Proseminarium	levélküldés	Csatlakozom az előttem szólókhoz, nekem sem jó! Szabó Tibor	2006-09-26 16:45:38	e-I110E1
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! Sajnos jelenleg reménytelen a levelezőt használni, de tapasztalataim szerint este 8 után már minden hasonló program hatékonyabban...	2006-09-26 16:46:50	e-HMWH...
LGB_AK024_1_Proseminarium	levélküldés	Reméljük :-). Támogatná esetleg valaki azt, hogy létrehozzunk egy külső levelezőlistát? Ennek lényege az, hogy ha valaki küld rá levelet, akkor a list...	2006-09-26 17:10:17	e-HZULF3
LGB_AK024_1_Proseminarium	levélküldés	Reméljük :-). Támogatja esetleg valaki azt, hogy létrehozzunk egy külső levelezőlistát? (ha még nincs) Ennek lényege az, hogy ha valaki küld rá lev...	2006-09-26 17:15:18	e-HZULF3
LGB_AK024_1_Proseminarium	levélküldés	Reméljük :-). Támogatja esetleg valaki azt, hogy létrehozzunk egy külső levelezőlistát? (Feltéve, ha még nincs) Ennek lényege az, hogy ha valaki k...	2006-09-26 17:18:59	e-HZULF3
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! Szeretnék létrehozni egy levelezőlistát az évfolyamunk részére. Mire jó ez? Ennek segítségével kényelmesebben értesülhetek fontos tudniva...	2006-09-26 17:55:32	e-HZULF3
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! Szeretnék létrehozni egy levelezőlistát az évfolyamunk részére. Mire jó ez? Ennek segítségével kényelmesebben értesülhetek fontos tudniva...	2006-09-26 17:56:05	e-HZULF3
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! Szeretnék létrehozni egy levelezőlistát az évfolyamunk részére. Mire jó ez? Ennek segítségével kényelmesebben értesülhetek fontos tudniva...	2006-09-26 17:59:20	e-HZULF3
LGB_AK024_1_Proseminarium	levélküldés	Helló Mindenki! Áldom az eget, hogy benéztem a fórumra: majdnem frászt kaptam, mert nem működött a levelezőm... Mostmár megnyugodtam...	2006-09-26 19:13:15	e-F04I2W
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! De jó, hogy én is beléptem ide, mert nálam sem működik a levelező. Sok sikert mindenkinek a továbbiakhoz. Görbe Nikolett	2006-09-26 19:51:49	e-UHQNS3
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! Szerintem még az informatikusok dolgoznak a rendszeren. Nézzünk vissza később! Pintérmé Demeter Henrietta	2006-09-26 19:52:08	e-BUOGDP
ST03_Aik-operációkutatás	again	sziasztok, én vagyok az első :) remélem mindenki jól van, erővel feltöltve...na persze...ezt én sem komolyan írom...a második felét persze! újra itt...	2006-09-26 19:53:50	e-CUDF81
ST03_Aik-operációkutatás	again	sziasztok, én vagyok az első :) remélem mindenki jól van, erővel feltöltve...na persze...ezt én sem komolyan írom...a vessző utáni részét természet...	2006-09-26 19:55:38	e-CUDF81
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! Szépen gyarapszik a létszám, én szoroklok. Unger László	2006-09-26 19:57:03	e-Z11JL0
LGB_AK024_1_Proseminarium	levélküldés	sziasztok megint Niki vagyok, azt szeretném kérdezni, hogy van-e valaki közületek, aki szólna, vagy közvetlen környékbeli településről való, ugyani...	2006-09-26 20:03:33	e-UHQNS3
LGB_AK024_1_Proseminarium	levélküldés	Szép Jóestét! Üdvözlök mindenkit és eredményes tanulást kívánok mindenkinek és előre is nagy kalappal a vizsgáthoz! Sebástén Zoltán	2006-09-26 20:20:35	e-UH1106
LGB_AK024_1_Proseminarium	levélküldés	Sziasztok! nekem se jó a levelezés, de aztán benéztem ide és láttam, hogy másmak se jó. Remélem hamarosan jó lesz :) Ladiszlai Péter	2006-09-26 20:25:10	e-MJTWD6

3. ábra: A fórum üzenetek szerkezete

Vizsgálataink során főként a fórum üzenet mezőben levő információkat használtuk fel.

3. A fórum üzenetek jellemző sajátosságai

A fórum üzenetek hasonlítanak az elektronikus levelekre, bár nyelvi igénytelenségük, sietős elküldésük miatt inkább az SMS műfajához állnak közel. A fórum üzenetek részei:

- Megszólítás vagy üdvözlés (pl. Tisztelt Tutor, vagy Hali, Szia, Sziasztok, Üdv).
- Érdeklődés, bemutatkozás, segítségkérés, teljesítés jelzése (pl. a feladat teljesítve).
- Köszönetnyilvánítás, véleményközlés, várákozás jelzése (pl. válaszáat várom).
- Zárás (Üdv, Üdvözlettel, Szia, Sziasztok).

Megszólítás általában a tutornak vagy a tutor által írt levelekben fordul elő, a hallgatók egymás között az üzenet elején és végén is valamilyen üdvözlési formát alkalmaznak. A fórum üzenetek elején használt megszólítási vagy üdvözlési formák alapján a fórum üzenetek típusai:

- tutor minden hallgatónak, 1002 db, 5,5 %;
- tutor valamelyik hallgatónak, 210 db, 1,2 %;
- hallgató minden hallgatónak, 4001 db, 22,0 %;
- hallgató tutornak, 1951 db, 10,8 %;
- hallgató valamelyik hallgatónak, 1525 db, 8,4 %;
- egyéb, 9458 db, 52,1 %.

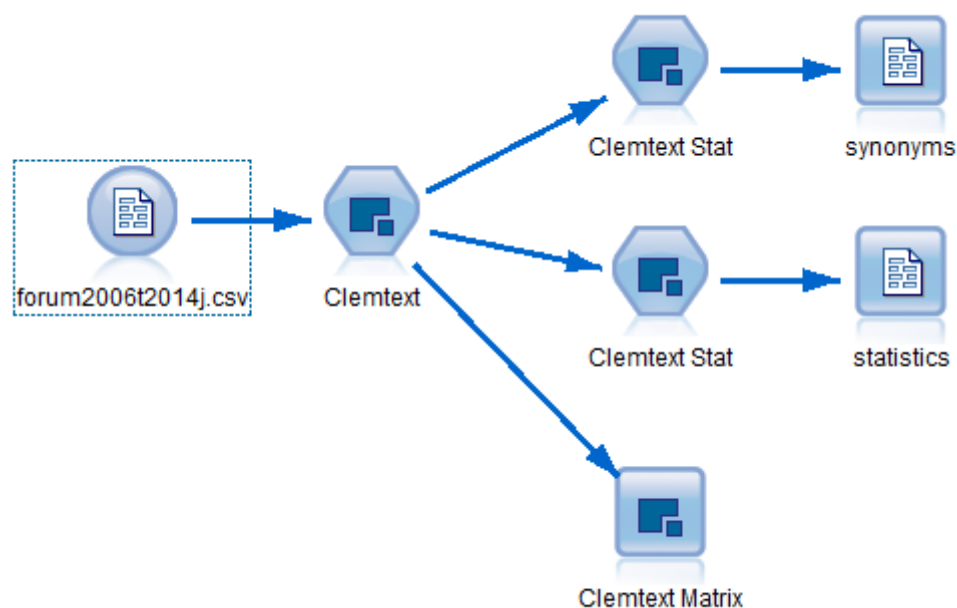
Láthatóan a fórum üzenetek több mint a fele megszólítás vagy üdvözlés nélkül kezdődik. Ezek típusa sztring műveletekkel nem állapítható meg, nagyobb részük hallgatói, kisebb részük az aláírásból ítélve tutori üzenet. Az arányokból is látható, hogy a hallgatók többször fordulnak egymáshoz, mint a tutorokhoz, a tanulmányi problémáikat először egymás között próbálják megoldani. Ezt korábban web bányászati elemzéssel is kimutattuk [3].

4. A fórum szótár létrehozása

A fórum üzenetek tartalmának elemzéséhez az SPSS Modeler adatbányászati szoftver Text Analytics szövegbányászati alkalmazását és az ahhoz készített, a magyar nyelvű szövegek feldolgozását lehetővé tevő CLEMTEXT modult használtuk fel.

Az SPSS Text Analytics fejlett nyelvészeti és NLP (Natural Language Processing) technológiát alkalmaz a szöveges állományok gyors feldolgozására. Alkalmas a kigyűjtött alapszavak és fogalmak kategóriákba sorolására. A nagy nyugat-európai nyelvekhez beépített erőforrásokat (könyvtárakat, szótárakat) biztosít.

Beépített magyar nyelvi erőforrások híján először magyar szótárt kell létrehoznunk. Ez a CLEMTEXT alkalmazás segítségével lehetséges. A CLEMTEXT a fórum üzenet mezőben levő szöveget mondatokra, a mondatokat szavakra bontja, majd szótövezéssel állítja elő a nyelvi alapegységeket. A nyelvi alapegységek szófaját a Morphdb.hu adatbázis alapján állapítja meg. A CLEMTEXT három különböző kimenetet képes előállítani: a szinonima párok listáját, a nyelvi alapegységek statisztikai jellemzőit megadó listát és a szó-dokumentum mátrixot (lásd 4. ábra) [5].



4. ábra: A CLEMTEXT 2.0 kimenetei

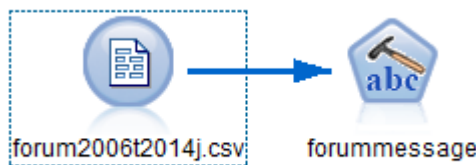
A nyelvi alapegységenként megadott statisztikai jellemzők a szó előfordulásainak száma a korpuszon belül (Term Frequency), azon dokumentumok száma, amelyekben a szó előfordult (Document Frequency), a szó súlya (TF-IDF) és a szófaj (Part-of-Speech tag).

A nyelvi alapegységek statisztikai jellemzőit megadó listából szófajonkénti listákat készíthetünk. Az így kapott szófajonkénti listák segítségével építhetjük fel a fórum szótárt. A szinonima listát a szinonima szótár építéséhez használhatjuk fel. A szó-dokumentum mátrix különböző szövegbányászati modellek bemeneteként alkalmazható.

Az SPSS Text Analytics képes az ékezetes betűk kezelésére, karakterhelyesen beolvasa az UTF-8 kódolású, ékezetes betűket tartalmazó szöveget. A beolvasott szöveget ISO-8859-1 kódolásúvá alakítja. Az ISO-8859-1 kódoláshoz nem illeszkedő karaktereket szóközzel helyettesíti. Az eredmények megjelenítésekor bizonyos ékezetes betűk (ő, ü) nem láthatók, emiatt a szótárakban és feldolgozandó szöveges állományokban ezeket célszerű az ö és az ü betűkkel helyettesíteni.

A CLEMTEXT szoftvert természetesen még az eredeti UTF-8 kódolású szövegen kell alkalmazni, hiszen a szófajok felismerése csak így lehetséges. A szófajonkénti listákat célszerű alaposan áttanulmányozni, mivel az eredeti szövegben levő elírások, helyesírási hibák helytelen szófaj megjelölést is eredményezhetnek. Ehhez járul még hozzá a Tordai-féle magyar szótövező algoritmus Heavy változatának túltövezésre való hajlandósága [4]. A legfontosabb szófajok szövegbányászati szempontból a főnevek, az igék és a melléknevek. Ezért ezek szó-listáit különös gondossággal kell ellenőrizni.

A fórum üzenetek feldolgozását a Text Mining node-ot tartalmazó stream végzi (lásd 5. ábra).



5. ábra: A Text Mining stream

A kutatás során létrehoztuk a fórum üzenetek elemzéséhez szükséges szótárt. Ez a Text Mining node Interactive Workbench üzemmódjában lehetséges (lásd 6. ábra).

The screenshot shows the 'Interactive Workbench - forummessage' window. The main area displays a list of terms and their synonyms. The 'Resources' table lists terms like 'abbat', 'abbaméret', 'abbat', etc., with their types (Entire Term, NOUN) and libraries (Forum Library (Hungarian)). The 'Target' table lists terms like 'budapest', 'coedu', 'eredmény', etc., with their synonyms and libraries. The 'Synonyms' table lists terms like 'budapest', 'coedu', 'eredmény', etc., with their synonyms and libraries. The 'Library' column indicates the source of each term, mostly 'Forum Library (Hungarian)'. The 'Exclude List' column has checkboxes for each term, and the 'Library' column has a dropdown menu for each term.

6. ábra: Az Interactive Workbench

Az 5. ábra középső keretében a főnevek (Noun), a jobb oldali keretében a kizárt szavak, a bal alsó keretében pedig a szinonimák láthatók. A magyar szótár teszi lehetővé a leggyakoribb fogalmak, kifejezések azonosítását (lásd 7. ábra).

Concept	In	Global	Docs	Type
üdvözet		5 407 (4%)	5 345 (29%)	<NOUN>
sziasztok		3 729 (2%)	3 718 (20%)	<UTT-INT>
köszönet		3 097 (2%)	3 035 (17%)	<NOUN>
2		3 154 (2%)	2 576 (14%)	<Weights-Measures>
feladat		2 519 (2%)	2 173 (12%)	<NOUN>
vizsga		1 573 (1%)	1 390 (8%)	<ADJ>
tisztelt tutor		1 294 (1%)	1 291 (7%)	<Unknown>
jegyzet		1 185 (1%)	975 (5%)	<NOUN>
szia		972 (1%)	962 (5%)	<UTT-INT>
1		1 348 (1%)	908 (5%)	<Weights-Measures>
3		1 048 (1%)	819 (5%)	<Weights-Measures>
kérdés		839 (1%)	739 (4%)	<NOUN>
coedu		791 (1%)	713 (4%)	<NOUN>
úr		717 (0%)	708 (4%)	<NOUN>
tutor		665 (0%)	648 (4%)	<NOUN>
4		824 (1%)	638 (4%)	<Weights-Measures>
lecke		686 (0%)	614 (3%)	<NOUN>
válasz		652 (0%)	602 (3%)	<NOUN>
tisztelt hallgatók		578 (0%)	578 (3%)	<Unknown>
előre		575 (0%)	573 (3%)	<ADV>
tantárgy		605 (0%)	541 (3%)	<NOUN>
5		669 (0%)	536 (3%)	<Weights-Measures>
tananyag		580 (0%)	528 (3%)	<NOUN>
remél		477 (0%)	477 (3%)	<VERB>

7. ábra: A leggyakoribb fogalmak

A kivonatolt fogalmak alapján a rekordok (a fórum üzenetek) kategóriákba sorolhatók. Mivel a Text Analytics angol nyelvészeti eszközöket használ a mintázatok és a szabályok megállapítására, a kapott kategóriák manuális finomítására szükség van. Egy érdekes, a szoftver által megállapított szabály a következő: fx [györ|györi|gyors|gyorsan], vagyis egy kategóriába sorolja a györ és a gyors fogalmakat. A leggyakoribb kategóriák a tutor, a hallgató, a vizsga, a feladat, a segítség és a megoldás.

A fogalmak közötti kapcsolatok alapján klaszterek képezhetők. A csoportképzés egy komplex algoritmus alapján történik, amely többek között azt vizsgálja, hogy két fogalom milyen gyakran fordul elő együttesen vagy önállóan. A klaszterek grafikusán is megjeleníthetők. Vizsgálatunk során a klaszterek képzése nem hozott új, felhasználható eredményeket.

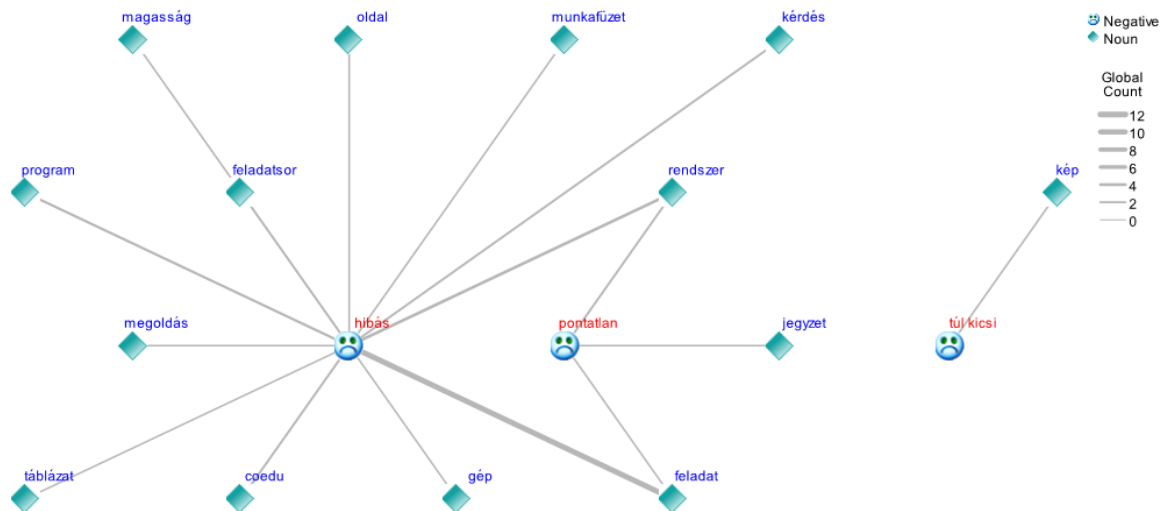
5. A vélemények vizsgálata

A szövegben rejlő összefüggések feltárásához Text Link Analysis (TLA) elvégzésére van szükség. Ehhez olyan nyelvi erőforrásokra van szükség, amelyek beépített TLA mintázatokat tartalmaznak. Mivel magyar nyelven ilyen nem áll rendelkezésre, magunknak kell azt elkészítenünk. Ki kell választanunk egy TLA mintázatokat is tartalmazó angol nyelvű szótárt, és el kell készíteni annak magyar változatát.

Mivel a hallgatói vélemények vizsgálatával is foglalkozni akartunk, ezért az angol Opinions nyelvi erőforrást választottuk. A magyar változat elkészítése két feladat megoldását tette szükségessé. Magyar nyelvű vélemény szótárt kellett készítenünk és az alapvető angol TLA mintázatok alapján meg kellett írunk azok magyar megfelelőit.

Az egyes vélemények előfordulásának gyakoriságát az ábrán a vonal vastagsága jelzi. Láthatóan a „tantárgy lényeges” és a „tanulás szükséges” vélemény gyakoribb a többinél.

A negatív véleményeket vizsgáló „főnév + negatív” mintázat eredménye a következő (lásd 10. ábra):



10. ábra: Negatív vélemények

Láthatóan a „feladat hibás” vélemény a leggyakoribb.

Az „Unknown + Q” mintázat érdekes eredményre vezetett, 7 rekordban fordult elő a „no para” kifejezés, 1 üzenetben pedig a „ne parázzon” változat. Ez a szleng kifejezés a „ne légy paranoid” vagyis „ne félj feleslegesen” rövidebb formája.

A tutori munkával kapcsolatos kritikai észrevételek is feltárhatók a TLA mintázatok segítségével. A „jegy nincs beírva” panasz 34 esetben fordult elő. A „tutor nem válaszol” észrevétel 30 üzenetben szerepelt.

6. További feladatok

Az idő előrehaladtával a fórum hozzászólások száma is növekszik. A fórumban új, eddig nem használt fogalmak, szleng kifejezések jelennek meg. Ezért időről időre el kell végezni a fogalmak kivonatolását, és a fórum szótár naprakész állapotba hozását. A CLEMTEXT 2.0 végleges változatának elkészültével a szófaj listák és a szinoníma listák készítése várhatóan hatékonyabbá válik. A végleges változat már a stoplistát is kezelni fogja, ezért az azon szereplő nyelvi elemek szótövezését már nem fogja elvégezni. Ez a futási idő csökkenését fogja eredményezni.

Mivel a szövegbányász szoftver angol nyelvészeti eszközöket használ, a fogalmak alapján kapott kategóriákat és klasztereket még kézi eszközökkel finomítani kell. Ennek jelenleg a

szinoníma szótár manuális, időigényes, elemenkénti bővítése az akadály. A szinoníma lista importálásának lehetősége jelentősen gyorsítaná a szótárépítést.

A részletes TLA elvégzése még további kutatást igényel. Az eddig létrehozott alapvető makrók és szabályok körét tovább kell bővíteni, ez jelentős fejlesztő munkát igényel.

További feladat egy olyan szövegbányász stream létrehozása, amely vélemény kategóriák szerint csoportosítja a fórum hozzászólásokat. Ez módot ad majd pl. az elégedett és a panaszos üzenetek rendszerezésére. A távoktatás minőségének biztosításához a hagyományos elégedettségmérő kérdőívek kitöltésén túl a hallgatók véleményének vizsgálata is hasznos adalékokat szolgáltathat.

7. Köszönetnyilvánítás

Ezúton is szeretnénk köszönetet mondani Körmendi Györgynek, a Clementine Consulting (korábbi nevén SPSS Hungary) ügyvezetőjének, hogy a kutatásainkhoz díjmentesen rendelkezésünkre bocsátotta az IBM SPSS Modeler 15.0, az IBM SPSS Text Analytics 15.0 szoftvereket és a CLEMTEXT 2.0 fejlesztői változatát. E szoftverek nélkül a kutatás elvégzése nem lett volna lehetséges.

Irodalomjegyzék

- [1] Miner, G. et al.: Handbook of statistical analysis and data mining applications, Academic Press, Burlington, 2009
- [2] Miner, G. et al.: Practical text mining and statistical analysis for non-structured text data applications, Academic Press, Waltham, 2012
- [3] Nyéki Lajos: A Coedu távoktatási keretrendszer használatának elemzése, in: Balogh Imre – Horváth Ádám (szerk.): Felhasználói viselkedés elemzése webes környezetben, DSGI Ergonómiai Mérnöki Iroda Kft, Budapest, 2010, 99.
- [4] Tikk Domokos: Szövegbányászat, Typotex Kiadó, Budapest, 2007, 49.
- [5] Tikk Domokos: User and Developer Manual for CLEMTEXT 2.0, TextMiner Ltd, Budapest, 2014, 13.

ERP-választás KKV környezetben

ERP selection at SMEs

Erdős Ferenc

Széchenyi István Egyetem, Informatika Tanszék
erdosf@sze.hu

Absztrakt: Napjainkban egyre több kis- és közepes vállalkozás (kkv) mérlegeli egy integrált vállalatirányítási rendszer bevezetését, vagy a meglévő rendszerének bővítését, illetve cseréjét. Ezen rendszerek bevezetése sok esetben külső pályázati forrás bevonásával történik, informális kiválasztás útján. Ennek ellenére az előadás kísérletet tesz a különböző rendszerek közötti választás szempontjainak áttekintésére és az elemzési folyamat során a kkv-k körében alkalmazható monetáris és hasznóérték-elemzésen alapuló technikák ismertetésére.

Kulcsszavak: ERP, választás, KKV

Abstract: Improvement of IT infrastructure has gained increased attention of Hungarian executives in order to enhance their competitiveness in the European marketplace. A key segment of these developments is the Enterprise Resource Planning (ERP). Today there is a wide range of ERP systems available for managers of SMEs, albeit in most cases it proves difficult to distinguish the adequate ones. This study attempts to analyse some of techniques to calculate the benefits of IT investments and investigates the practice of the companies.

Keywords: ERP, choose, SME

1. Bevezetés

Napjainkban a különböző beruházások előzetes gazdasági elemzése a vállalati pénzügyek, mint tudományterület alapvető tárgyterületei közé tartozik. A konvencionális (termelő) beruházások esetében kialakult ezen számítások jól kiforrott vállalati gyakorlata. Az is megfigyelhető, hogy a különböző gazdálkodó szervezetek viszonylag magas arányban használják is ezeket, és túlnyomórészt azonos mutatók és módszerek alkalmazása mellett teszik le a voksukat.

Mindez az informatikai beruházások és ezen belül is az integrált vállalatirányítási rendszereket (ERP – Enterprise Resource Planning) érintő beruházások esetében egy teljesen más, ambivalens képet mutat. A vállalati pénzügyekkel foglalkozó gazdasági szakemberek többnyire nem rendelkeznek megfelelő szakértelemmel az ilyen típusú IT-beruházások speciális költségeinek és gyakran rendkívül nehezen monetarizálható hozamainak beazonosításához és számszerűsítéséhez, valamint az ilyen elemzések elvégzéséhez szükséges módszertani ismereteik is hiányosak. A másik oldalon viszont megfigyelhető, hogy az informatikai projektekkal foglalkozó szakemberek rendszerint projektmenedzsment értelemben közelítik meg ezen beruházásokat és nehezen azonosulnak a projekttel, mint az IT-beruházás által generált változásalapú pénzáramokkal.

A tanulmány kísérletet tesz bemutatni az elméleti szakirodalmakban javasolt mutatók és módszerek vállalati gyakorlatban való elterjedtségét mind a konvencionális, mind pedig az informatikai befektetési körben.

2. A beruházásértékelési módszerek és azok alkalmazásának fejlődése

Az üzleti gazdaságtanhoz sorolható három tudományterület (számvitel, adótan, vállalati pénzügyek) közül a vállalati pénzügyek a legfiatalabb, amely mindössze a múlt század 60-as éveitől tekinthető önálló tudományágnak. Kezdetben, egészen a 80-as évekig a vállalati pénzügyes megközelítés minden olyan beruházást megvalósítandónak tartott, amely számviteli értelemben nyereséges volt. Egyes vállalati kutatók már a 20. század utolsó harmadától kezdődően elkezdtek fókuszálni a különböző beruházási projektértékelési módszerek gyakorlati alkalmazásának vizsgálatára. Ezek kezdetben a – kor paradigmájának megfelelően – kizárólag a nagyvállalatokra koncentráltak és a belső kamatláb (IRR – Internal Rate of Return) alkalmazásának elsődleges prioritását hangsúlyozták, jóllehet az IRR csak rendkívül korlátozottan, a legkritikább esetben alkalmas több beruházási alternatíva komparatív vizsgálatára. Gitman és Forrester 1977-es felmérésében [8] 103 nagyvállalatot vizsgáltak, ahol a megkérdezettek mindössze 9,8%-a alkalmazta a nettó jelenérték (NPV – Net Present Value) számítást, ezzel szemben 53,6 % az IRR-t. Stanley és Block 1984-es kutatása [19] szerint a vállalatok 65% használta előzetes beruházásértékeléshez az IRR módszert.

Később a 80-as évektől paradigmaváltás következett be a vállalati pénzügyek területén. Az új, a mai kor szellemének megfelelően már azt a beruházást javallott megvalósítani, ami a többi hasonló beruházási lehetőséghez képest nagyobb hozamot hoz. Moore és Reichert 1983-as felmérésében [14] a Fortune 500 -as listából 298 vállalatot kérdeztek meg, amely szerint a cégek 86%-a alkalmaz beruházási döntéseinek meghozatala előtt valamilyen diszkontált pénzáramon (DCF – Discounted Cash Flow) alapuló gazdasági elemzést. Bierman 1993-ban megállapította [1], hogy a Fortune 100-as listából megkérdezett 74 cégből 73 használ DCF megközelítést, ami az ilyen típusú módszerek erőteljes terjedését jelzi. A 20. század végén ezt a területet vizsgáló számos egyéb kutatás is hasonló eredményre jutott.

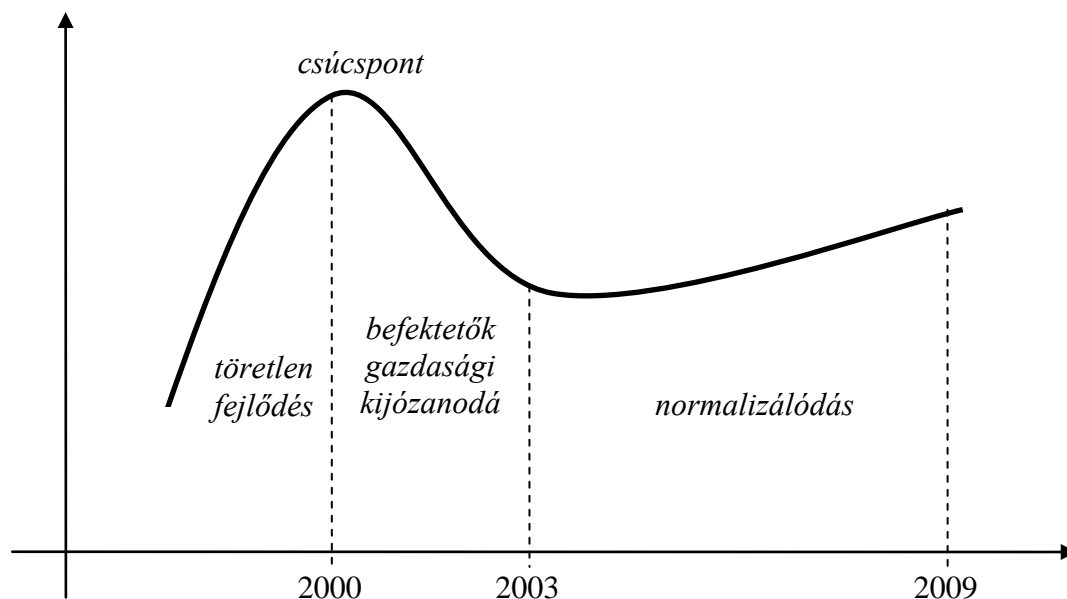
A hazai szakirodalomba és üzleti életbe ez az új megközelítés csak némiképp később, a rendszerváltás után fokozatosan ivódott át. Ezt jól jelzi, hogy a vállalati pénzügyek területén alapműnek tekinthető, magyarra is lefordított és hazánkban is kiadott „Modern vállalati pénzügyek” című könyv [3] eredeti angol címében [4] nem szerepel a modern szó, mivel ez a fajta megközelítés csak hazánkban volt újszerűnek tekinthető. A hazai vállalatok tehát csak jóval később kezdték alkalmazni ezt a megközelítést beruházási döntéseik meghozatalakor. Ezzel kapcsolatban a külföldi érdekltségű vállalatok más helyzetben voltak, mivel azok rendszerint a menedzsmentmódszerekkel egyetemben a vállalati pénzügyi kultúrát is egy az egyben az anyavállalattól vették át.

3. A beruházásértékelési módszerek és azok alkalmazásának fejlődése

A beruházás-gazdaságosság kérdése az informatikai célú beruházások esetében is természetesen felmerül. Kezdetekben a múlt század harmadik negyedében még rendkívül egyszerűek voltak ezek a számítások, ugyanis az automatizálás korszakában a számítógép alkalmazása nagyrészt élőmunka kiváltására szolgált, s az így kalkulálható munkaerő megtakarítás jól alátámasztotta a beruházás gazdaságosságát. [2] Később a menedzserek, majd egyre több alsóbb szintű vezető asztalain megjelenő személyi számítógépek a személyes hatékonyságot növelték, amelyet tovább erősített a belső hálózatok, integrált vállalatirányítási rendszerek és a külső üzleti modellek megjelenése. Ebből adódóan a számítások egyre bonyolultabbá váltak, a beruházások rendkívül sok nehezen számszerűsíthető minőségi jellegű hozamokat mutatnak. Megfigyelhető volt, hogy korábban, a kilencvenes években a fejlett országok vállalatainak többsége nem fordított kellő hangsúlyt az informatikai beruházások gazdaságosságának vizsgálatára, mivel ezen investíciók realizálását kötelező

jelleggel, az esetleges versenyhátrány elszűnésének kiküszöbölése céljából, sokszor mindennemű gazdaságossági kontroll nélkül valósították meg [20].

A beruházás gazdaságossági számítások szerepe jellemzően forráshiányos vállalati környezetben válik hangsúlyosabbá. Ilyen esetekben a vállalatvezetésnek jól megfontolt döntést kell hozni, mire költi a szűkösen rendelkezésre álló forrásokat. Az ezredforduló után az ún. „dot com lufi” kipukkanása és a világgazdaságban bekövetkezett recesszió miatti forráshiány előtérbe helyezte a beruházás gazdasági vizsgálatok jelentőségét az információ és kommunikációtechnológiai investíciók terén is. Az 1. ábra a Gartner Group e üzletvitelre vonatkozó életgörbéjét szemlélteti, amely szintén rámutat az egyes korszakok jellemzőire.

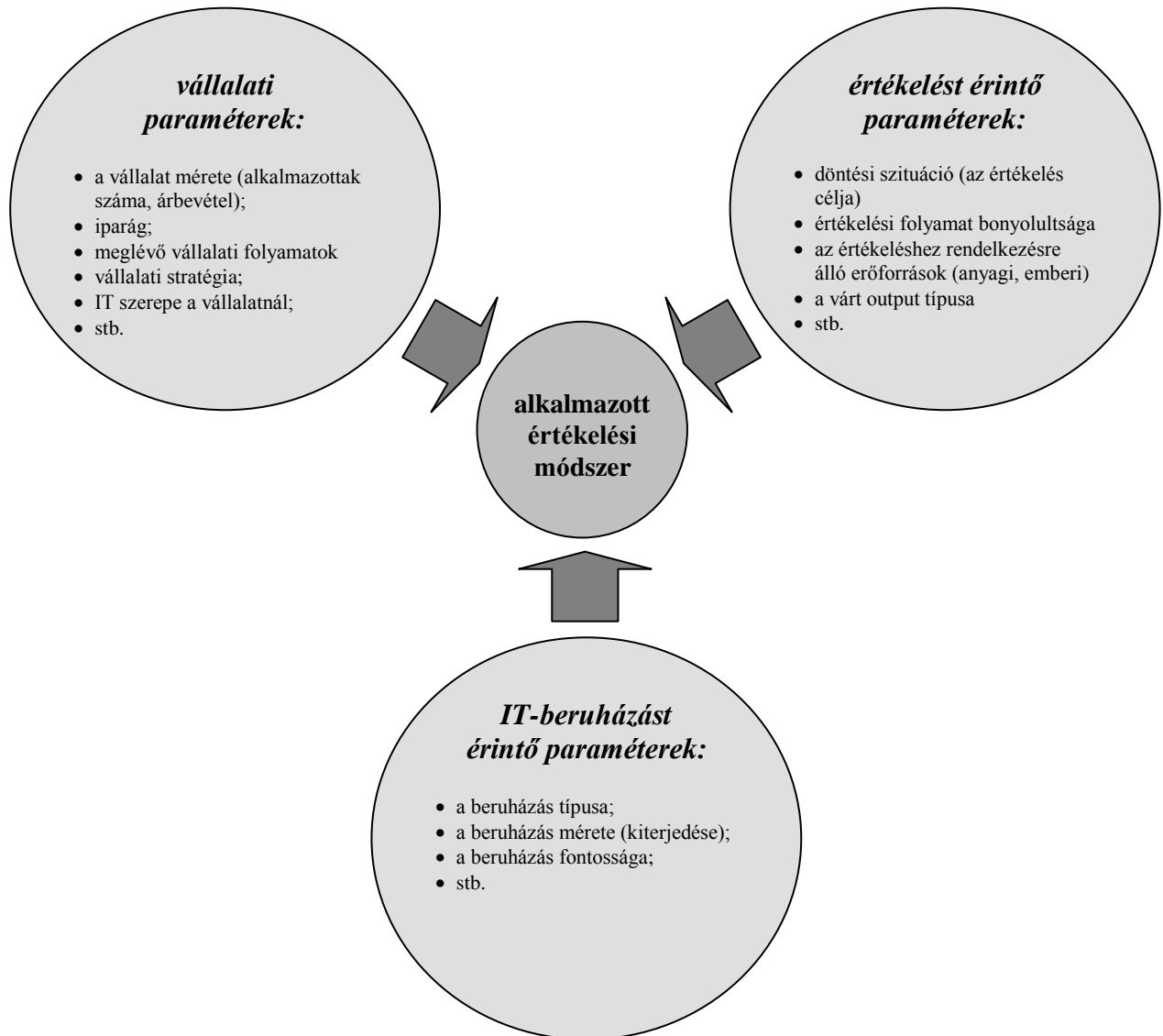


1. ábra: Az elektronikus üzletvitel fejlődésének életgörbéje

Forrás: [16]

Napjainkban már számos kidolgozott eszköz létezik a vállalatok IT-beruházásainak értékelésére, a viszonylag egyszerű (leegyszerűsített) eljárásoktól kezdődően a komplex módszertanokig bezárólag. A pénzügyi jellegű, kvantitatív módszereknél megfigyelhető, hogy minden, a beruházással kapcsolatban felmerülő tényezőt megpróbálnak, még esetleg akár inadekvát módon is, valamilyen pénzügyi mennyiségre váltani. Természetesen léteznek különböző egyéb kvantitatív és kvalitatív módszerek, amelyek segítségével az IT-beruházások értékelhetők, azonban ezek által a hasznok közvetlenül nem monetarizálhatók [6]. A nem pénzügyi kvantitatív módszerek (pl.: pontozáson alapuló értékelési rendszer) valamilyen nem pénzügyi számértékeket képeznek, míg a kvalitatív módszerek (pl.: CSF, portfólió elemzés) a probléma megértésében és a stratégiai illeszkedés vizsgálatában nyújthatnak további segítséget. Léteznek többtényezős, összetett értékelési eljárások is, amelyek lényege, hogy valamilyen többdimenziós szempontrendszer alapján történik az adott beruházási projekt értékelése, az investíció üzleti értékének meghatározásával. A pénzügyi, kvantitatív módszerek használatával az egyes IT-beruházási alternatívákkal összefüggésben a becsült diszkontált pénzáramok alapján gyakorlatilag mindegyik pénzügyi mutató (pl.: NPV, ROI, IRR, PBT stb.) viszonylag könnyen képezhető.

Az egyes módszerek közötti választást számos tényező befolyásolhatja, amelyek közül a legfontosabbakat a 2. ábrán foglalom össze:

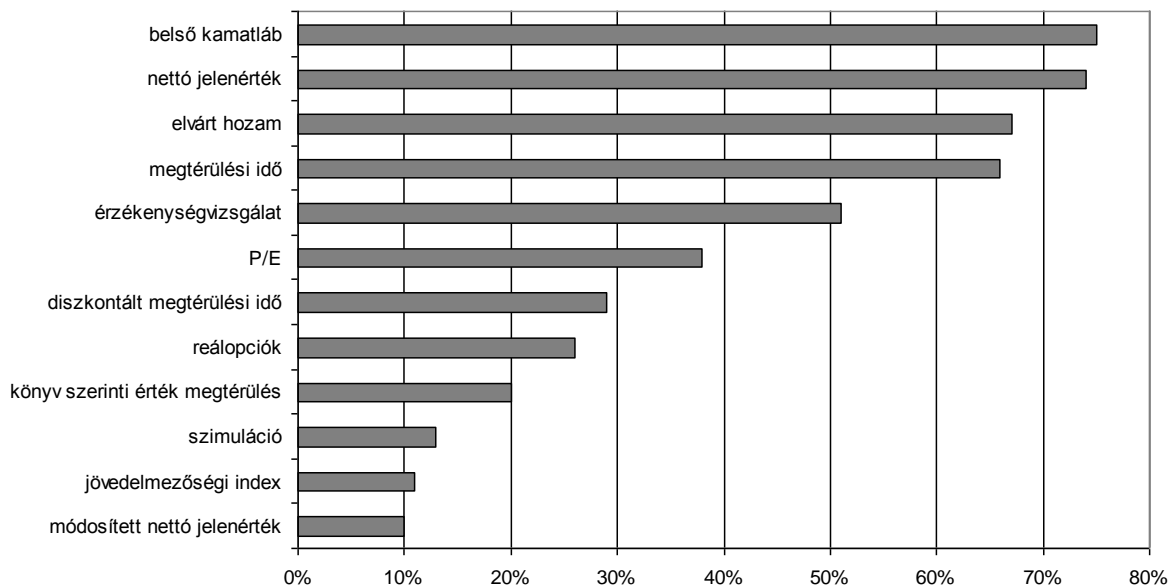


2. ábra: Az IT-értékelési módszerek közötti választást befolyásoló tényezők

Forrás: [7]

4. A vállalati gyakorlatban alkalmazott módszerek

Az általánosan használt beruházásértékelési módszerek vizsgálatakor a vállalati pénzügyekkel foglalkozó szakirodalmak rendszerint Graham és Harvey ezredfordulón készített felmérését [10] említik, amely közel négyszáz amerikai vállalatot vizsgált. A 3. ábra diagramja ezen kutatás eredményét mutatja.



2. ábra: A vállalati gyakorlat során alkalmazott beruházásértékelési módszerek, USA (2000)

Forrás: [10]

A kutatás megállapítja, hogy az NPV és az IRR gyakorlatilag azonos arányban alkalmazott mutatók. A felmérés kiterjedt a kis- és közepes vállalkozásokra is. Amíg a nagyobb vállalatok sokkal inkább alkalmazzák a nettó jelenérték számításokat, valamint a tőke költség meghatározásához a tőkepiaci árfolyamok modelljét (CAPAM – Capital Asset Pricing Model) használják, addig a kisebbek relatíve gyakrabban alkalmaznak valamilyen a pénz időértékét figyelmen kívül hagyó megtérülési idő számítást, vagy fixen elvárt hozamot [10]. A reálopcióknak a gazdaságelemzési számításokba való beépítése már a fent bemutatott 2001-es felmérés során is megközelítette a 30%-ot. Ezen reálopciók alkalmazása kifejezetten a stratégiai beruházások értékelése során rejt magában komoly lehetőségeket. [18]

Megfigyelhető, hogy az informatikai beruházásokat érintő kutatások csak rendkívül korlátozottan vizsgálják a beruházásgazdaságosság kérdését, az általános beruházási felmérések pedig csak a legkritkább esetben terjednek ki az IT-beruházásokra, mivel azok rendszerint a konvencionális beruházásokra koncentrálnak. A kifejezetten az IT-beruházások gazdaságossági elemzésének vállalati gyakorlatát boncolgató vizsgálatok¹ során pedig a hangsúly alapvetően nem a számított mutatókra (NPV, IRR, ROI, PI, PBT stb), hanem elsősorban a költség-haszon elemzések használata során alkalmazott technikák mennyiségi vagy minőségi voltára helyeződik. Hallikainen és szerzőtársai például a skandináv államok² körében végzett kutatásaik [11] során megállapították, hogy az alkalmazott monetarizált, kvantitatív eljárások aránya rendkívül alacsony és a vállalatok közel fele esetében az IT-beruházási döntések mindenféle számítással való alátámasztás nélkül a menedzserek belátása alapján születnek, ami egybevág más országokban végzett korábbi kutatások³ eredményeivel. Történtek olyan vizsgálatok is, amelyek a számszerűsítésen alapuló módszerek során alkalmazott pénzügyi mutatókat vették górcső alá. Egy az USA informatikai biztonsági beruházásait vizsgáló tanulmány [17] szerint az investíció előtt gazdasági elemzést végző

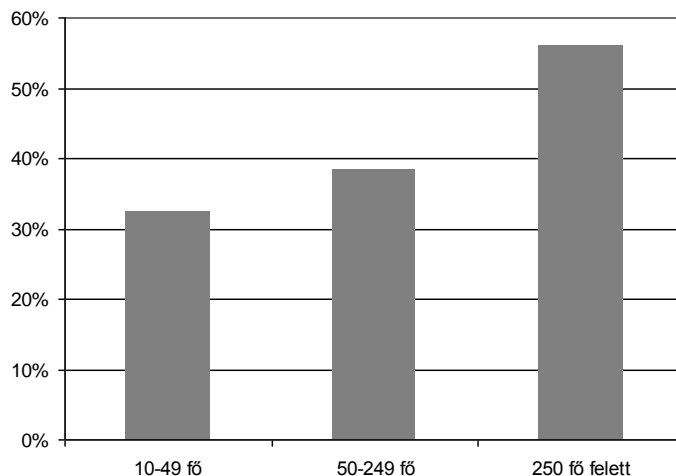
¹ Ilyenek például: Hallikainen et al, 2006; Lin–Pervan, 2003; Love et al, 2005; Norris, 1996.

² A felmérés Svédországra, Norvégiára és Finnországra terjedt ki.

³ Ilyenek például: Lin–Pervan, 2003; Love et al, 2005; Norris, 1996.

vállalatok több mint fele számol ROI-t, valamint negyedük az IRR, további negyedük pedig az NPV mellet teszi le voksát.

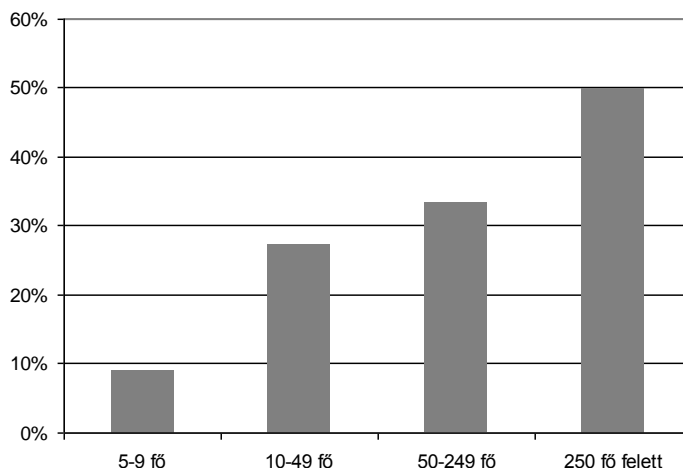
A hazai felmérések csak rendkívül ritka esetben térnek ki az IT-beruházási döntések esetén alkalmazott megtérülés-számítási módszerek alkalmazásának vállalati gyakorlatára. A BellResearch által minden évben publikált Magyar Infokommunikációs Jelentés is mindössze az ilyen számításokat végző vállalatok arányát (4. ábra) vizsgálja, az alkalmazott módszerek esetében mindössze a ROI típusú (költségek és várható hozamok számbavétele), TCO típusú (csak a költségek figyelembe vétele) és egyéb kategóriákat különbözteti meg.



4. ábra: Az IT-beruházások előtt megtérülési számításokat végző vállalatok aránya Magyarországon (2006-ban)

Forrás: BellResearch 2006 alapján saját szerkesztés

Megfigyelhető, hogy a vizsgált vállalatok körében csak a nagyvállalatok közül alkalmaz legalább minden második valamilyen számítást, a kisebb vállalatoknál ez az arány jóval alacsonyabb. Egy másik, ugyanezt a kérdést vizsgáló, de szűkebb területi dimenzióban elvégzett felmérés [7] szintén hasonló eredményt mutat (5. ábra).

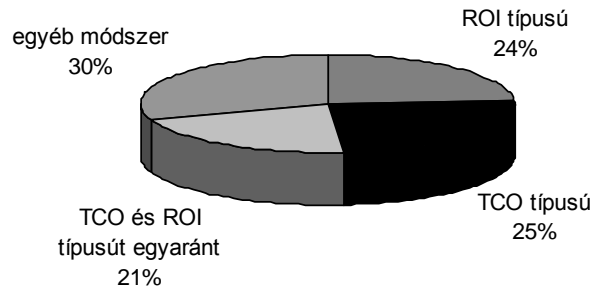


5. ábra: A Nyugat-dunántúli régióban működő ERP-rendszerrel rendelkező vállalatok közül IT-beruházásgazdaságossági számításokat végzők aránya vállalati méret szerinti bontásban (2007-ben)

Forrás: [7]

A hazai kis- és középvállalkozások többségénél megfigyelhető forráshiány [9] alapján a szűk anyagi lehetőségek a különböző fejlesztések előtti gazdaságossági számítások elvégzésének egyik legfőbb moztatórugói lehetnének. Ennek ellenére mindkét idézett elemzés esetében megfigyelhető, hogy a kkv-k beruházási döntéseik előtt jóval ritkábban alkalmaznak beruházásgazdaságossági számításokat, mint a nagyvállalatok.

Az egyes felmérések azonban sajnos nem vizsgálják az alkalmazott módszerek részletes körét. A BellResearch erősen leegyszerűsített kategóriái szerinti vizsgálati eredményeket a 6. ábra szemlélteti.



6. ábra: Az IT-beruházások előtt megtérülési számításokat végző vállalatok által alkalmazott módszerek megoszlása Magyarországon (2006-ban)

Forrás: BellResearch 2006 alapján saját szerkesztés

A BellResearch által vizsgált módszerek tekintetében megfigyelhető, hogy a vállalatok közel öt százaléka beruházásgazdasági elemzés keretében mindössze csak a költségek egzakt meghatározására törekszik, ami alapján önmagában semmiféle megtérülés nem számítható.

5. Összefoglalás

Összességében megállapítható, hogy az informatikai, és ezen belül az ERP-beruházásokat érintő kutatások csak rendkívül korlátozottan vizsgálják a beruházásgazdaságosság kérdését, az általános beruházási felmérések pedig csak a legritkább esetben terjednek ki az IT-beruházásokra, mivel azok rendszerint a konvencionális (termelő) beruházásokra koncentrálnak. Az elemzett felmérések többnyire rámutatnak, hogy az IT-beruházási döntések meghozatalát jóval ritkábban támasztják alá különböző számításokkal. Minden bizonnyal a vállalati pénzügyes és informatikus szakemberek között megfigyelhető kommunikációs rés is hozzájárul ahhoz a tényhez, hogy az informatikai investíciók esetében arányaiban jóval ritkábban alkalmaznak valamilyen pénzügyi megtérülésre irányuló gazdasági elemzést, mint a termelő beruházásoknál, valamint ezek gyakorlati megvalósítása is sokszor rendkívül elnagyolt és durva becsléseken alapul. A sok esetben pályázati támogatással megvalósított beruházások fejlesztőinek kiválasztása még jobb esetben is csak valamilyen pontozáson alapuló, súlyozásos kritériumrendszert alkalmazó haszonérték-elemzéssel történik, amely nem számol a pénzügyi megtérülés mutatóival.

Irodalomjegyzék

- [1] Bierman, H.J.: Capital budgeting in 1992: a survey. *Financial Management* 1993/22.
- [2] Bőgel Gy.: Üzleti elvárások – informatikai megoldások. HVG Kiadó, Budapest, 2009.
- [3] Brealey, R.A., Myers S.C.: *Modern vállalati pénzügyek*. Panem, 2005, Budapest.
- [4] Brealey, R.A., Myers, S.C.: *Principles of Corporate Finance*. Fifth Edition; McGraw Hill, 1983, New York.
- [5] Bruner, R.F., Eades, K.M., Harris, R., Higgins, R.C.: Best practices in estimating the cost of capital: survey and synthesis. *Financial Management* 1998/27., p. 13-28.
- [6] Erdős F., Raffai M., Varga Á.: The Benefits of IT Investments – Methodology and Analytic Approach – Chroust G.– Doucek P. (Hrsg.) *Schriftenreihe Informatik* 2008/25. Trauner Verlag, Linz. pp. 275-288.
- [7] Erdős, F.: A kis- és közepes vállalkozások informatikai beruházásai és azok megtérülési lehetőségei Magyarországon – doktori értekezés, Széchenyi István Egyetem, Regionális- és Gazdaságtudományi Doktori Iskola, 2009.
- [8] Gitman, L.J., Forrester, J.R. Jr.: A survey of capital budgeting techniques used by major U.S. firms. *Financial Management* 1977/6., pp. 66-71.
- [9] GKI Évkönyv 2002. GKI Gazdaságkutató Rt., Budapest, 2003.
- [10] Graham, J.R., Harvey, C.R.: The theory and practice of corporate finance: Evidence from the field. *Journal of Financial Economics* 2001/61., pp. 187-263
- [11] Hallikainen, P., Hu, Q., Frisk, E., Eikebrokk, T. R., Päivärinta, T., and Nurm, A.: The use of formal IT investment evaluation methods in organizations: A survey of European countries. *Proceedings of the 12th Americas Conference on Information Systems (AMCIS)*. Acapulco, México, August 4-6, 2006.
- [12] Lin, C., Pervan, G.: The practice of IS/IT benefits management in large Australian organizations. *Information & Management* 2003/41, pp. 13-24.
- [13] Love, P. E. D., Irani, Z., Edwards, D. J.: Researching the Investment of Information Technology in Construction: An Examination of Evaluation Practices. *Automation in Construction* 2005/14, pp. 569-582.
- [14] Moore, J.S., Reichert, A.K.: An analysis of the financial management techniques currently employed by large U.S. corporations. *Journal of Business Finance and Accounting* 1983/10., pp. 623-645.
- [15] Norris, G. D.: Post-Investment Appraisal. In Willcocks (ed.): *Investing in Information Systems*, Chapman and Hall, London, 1996. pp. 193-223.

- [16] Raffai, M.: Az információ - Szerep, hatás, menedzsment. Palatia, 2006, Győr.
- [17] Richardson R.: CSI Computer Crime & Security Survey. Computer Security Institute, 2008.
- [18] Rózsa, A.: A reálopciók lehetőségei és korlátai a stratégiai beruházások értékelésében. – Majoros P. (szerk.): Budapesti Gazdasági Főiskola Szakmai Füzetek. BGF, 2007. pp. 50-63.
- [19] Stanley, M.T., Block, S.B.: A survey of multinational capital budgeting. The Financial Review 1984/19., pp. 36-54.
- [20] Strassman, P. Why ROI ratios are now crucial to IT investment. Butler Group Review 2002/9. pp. 5 7.
- [21] Trahan, E.A., Gitman, L.J.: Bridging the theory-practice gap in corporate finance: a survey of chief financial officers. Quarterly Review of Economics and Finance 1995/35., pp. 73-87.

Termelésinformatikai eszközök hazai gyárak működésében

Production informatics tools in operation of domestic factories

Gludovátz Attila^a, Bacsárdi László^b

^a Nyugat-magyarországi Egyetem, Simonyi Károly Műszaki, Faanyagtudományi és Művészeti Kar, Informatikai és Gazdasági Intézet

gludovatza@inf.nyme.hu

^b Nyugat-magyarországi Egyetem, Simonyi Károly Műszaki, Faanyagtudományi és Művészeti Kar, Informatikai és Gazdasági Intézet

bacsardi@@inf.nyme.hu

Absztrakt: A gyárakkal rendelkező vállalatok legfontosabb értékteremtő folyamata a termelés. A termelésmenedzsment feladata az input erőforrásokat termékeké vagy szolgáltatásokká konvertáló rendszer menedzselése. Ehhez a funkcionalitáshoz szorosan kapcsolódnak olyan informatikai megoldások, amelyek támogatják a precíz irányítást, valamint a folyamatok minél pontosabb megtervezését elsősorban az átfutási idők és a költségek előrejelzése szempontjából. Továbbá segíti még az utólagos adattárolást, feldolgozást, elemzést és ezáltal lehetőségünk nyílik a termelési folyamatok optimalizálására. Ezeknek az informatikai szoftvermegoldásoknak rendkívül széles spektrumát alkalmazzák az egyes vállalatoknál kezdve az egyszerű irodai programcsomagoktól egészen a professzionális adatbányászati alkalmazásokig. Az ipari informatikai megoldások bemutatása és megismertetése egyik fontos célja az oktatásunknak a Nyugat-magyarországi Egyetemen is. Az elmúlt évek során lehetőségünk nyílt megismerni néhány olyan szoftvert, amelyeket Magyarországon működő cégeknél használnak. Munkánk során ezen szoftvereket elsősorban abból a szempontból vizsgáltuk, hogyan épülnek be a termelési folyamat támogatásába. Cikkünkben bemutatjuk néhány kiválasztott hazai gyárnál működő megoldást, vizsgáljuk azokat, és azt is bemutatjuk, hogy a kutatásból származó tapasztalatokat hogyan lehet beépíteni a gazdaságinformatika szak oktatásába.

Kulcsszavak: termelésinformatika, termelésmenedzsment, szoftver-elemzés

Abstract: The production is the most important process related of making valuables. The solutions of industrial engineering (standards and algorithms) support the production in our country as well. Main tasks in this term: prediction of lead time and costs, control the work of capital goods, store the data of output etc. In our state of art article, we describe some software, which support production in the manufactories starting with simple methods through complex production algorithms ending with complete industrial solutions. We bring into focus the Hungarian applications. Our experiences can be also used in our education.

Keywords: informatics in production, industrial engineering, software analyzing

1. Bevezetés

Napjainkban az egyes vállalatoknál rendkívül színes az alkalmazott szoftverek palettája. Kezdve a leginkább elterjedt irodai alkalmazásoktól, a specifikus termelést irányító szoftvereken át a menedzserek munkáját támogató rendszerekig. Ebből a sokrétű listából mi most leginkább a termelésirányító szoftverekre koncentrálunk. Ezek a rendszerek legtöbbször

valamilyen nagyobb vállalatirányítási rendszer részei (SAP, Microsoft Dynamics AX stb.). A vállalatirányítási rendszereket már rengetegen megvizsgálták. Az egyik fontos probléma ezekkel kapcsolatban rögtön a bevezetés fázisa előtt felmerül: örök kérdés a szoftverválasztással kapcsolatban, hogy a standard (dobozos) termék vagy az egyedi fejlesztés mellett döntsünk. A kettő közötti különbségekbe nem célunk belemenni ezen tanulmányban, mivel ezt már előttünk sokan felmérték különböző szempontok alapján. Mivel azonban előbbiről rengeteget tudhatunk meg a különböző tanfolyamokon, interneten keresztül, folyóiratokból, ezért a második bemutatására helyezzük a fő hangsúlyt. A saját fejlesztésű szoftverek sokkal egyedibb és izgalmasabb megoldások a magyarországi termékek piacán. Néhány esettanulmányt fel is villantunk ezzel kapcsolatban, melyeket hazai cégeknél fejlesztettek ki és/vagy hazai cégeknél használnak azokat.

Mielőtt azonban megvizsgálánk a konkrét rendszereket, érdemes áttekinteni, hogy milyen újfajta megközelítéseket, konvenciókat vezettek be az idők során a termelésirányítás területére. Az mindenki számára világos lehet, hogy a termelésirányítási folyamatok összehangolása meglehetősen szerteágazó és sok feladattal járó, magas odafigyelést igénylő tevékenység. Emiatt a gépesített termelésben professzionális mérnökök csoportja végezte ezt a munkát. Napjainkban azonban, az informatika előretörése nagyban segíti a vállalatok hasonló tevékenységeinek működését. Ehhez azonban szükség van olyan fejlett programozókra is, akik a hatékony algoritmusokat a számítógépek nyelvére képesek voltak átültetni.

A tanulmány utolsó részében megvizsgáljuk azt, hogy ezeket az elméleti és gyakorlati tapasztalatokat, hogyan érdemes az oktatásba beleilleszteni, mivel az egyetemeken a BSc (Bachelor of Science) és az MSc (Master of Science) rendszerek bevezetése után is küzdünk azzal, hogy az oktatási anyagokat megfelelő mélységű gyakorlati ismeretekkel töltsük fel.

2. Gyártással kapcsolatos tevékenységek, fejlődési irányok áttekintése

Az ipari korszakban (2000-es évekig bezárólag) a termeléssel kapcsolatos különböző technológiák és fejlesztések élen jártak az innovációs tevékenységek területén. Ennek egyik jeles képviselője a Toyota gyártási rendszere volt, amely számos új, addig nem alkalmazott technikát vezetett be a termelési folyamatba. A tömegtermelés, futószalagon termelés már a második világháború idején is elterjedt volt az ipari vállalatoknál, azonban a Toyota cégnél fejlesztették ezt professzionálisra azáltal, hogy sokkal magasabb szintre emelték az alábbi elvek gyakorlati alkalmazását: folyamatos gyártás elve, szabványosítás, automatizálás stb. A japánok hatékony megoldást adtak az állandóan változó világ kihívásaira: az egyre nagyobb mennyiségek és egyre szerteágazóbb termékek előállítására kialakították a Toyota Termelési Rendszerét [1].

A 2000-es évek végén kialakuló globális pénzügyi válság talán mondhatjuk, hogy leginkább az autópárba gyűrűzött be [2]. Válsághelyzetben pedig válságmegoldásokra van szükség. A Toyota cég még napjainkban is úttörőnek számít a fejlesztési irányvonalak területén. Míg a nagy európai versenytársaik – élükön a Volkswagennel – a terjeszkedéssel és az új gyárak építésével akarnak minél nagyobb nyereséget elérni, addig a Toyota vezetői úgy gondolkodnak, hogy inkább a már meglévő gyáraikat próbálják még inkább hatékonyabbá tenni és óvatosabban tekintenek a jövőbe. Ezáltal nagy kockázatot vállalnak, hiszen le is maradhatnak a többiektől, de ha bejön nekik ez a stratégiai húzás, akkor ismét ők jöhetnek ki győztesen a versenyfutásból [3].

A Toyotát mindig jellemezte az effajta kockázatvállalás és új, ismeretlen irányba való előrehaladás. Ezek által voltak annyira sikeresek az előző évtizedekben, mert mindig arra törekedtek, hogy időt és ezáltal pénzt spóroljanak. Technikájukat a világon rengeteg cég

követte. Például egy hazai biztonsági öveget gyártó cégnél néhány éve vezettek be egy elektronikus javaslati rendszert. Ennek segítségével az alkalmazottak javaslatokat tehetnek a gyártási folyamat optimalizálására, és ha a vezetőség megvalósíthatónak, ténylegesen időspóroló fejlesztésnek találja ezt a javaslatot, akkor a dolgozót pénzügyi jutalomban részesíti, az újítást pedig bevezetik éles keretek közé.

A gyártási folyamatok optimalizálásának célja a termeléshez kapcsolódó költségek minimalizálása, továbbá a változó kereslethez való alkalmazkodás. Ez többféleképpen történhet:

- Munkaerő változtatása: a kereslet ingadozásának megfelelően a létszám szabályozása – kis szaktudás-igény és nagy kereslet-ingadozás esetén lehetséges.
- Munkaidő változtatása: állandó munkaerő mellett a munkaórák számát módosítjuk (rugalmas munkahét, túlórák) – nagy szaktudás-igény esetén célszerű.
- Készletszintek szabályozása: állandó létszám, fix munkaidő és kibocsátási szint esetén a készletek mennyiségének változtatásával tudunk alkalmazkodni az ingadozó kereslethez.
- Szállítási határidő (rendelésállomány) módosítása: a kereslet növekedésével több munkát vállalunk, nagyobb határidővel, és fordítva.

Ezek az optimalizálási tevékenységek nem feltétlenül kockázatmentesek, hiszen a következő költségvonatokkal járhatnak:

- Termelési költségek: az ingadozó kibocsátás a változó termelési költségeket módosítja.
- Munkaerő változtatásával kapcsolatos költségek: felvétel, betanítás, elbocsátás stb.
- Készlettartás költségei.
- Határidő növelésének költségei (hiányköltség): a jó hírnév vagy a vevő elvesztése stb.

Azonban a fejlődés érdekében mindig fejleszteni kell a cég folyamatait. Ennek legkevésbé kockázatos eszköze az lehet, ha a gyártási üzem területén módosítjuk a termelőgépeink elhelyezkedését, ezáltal spórolhatunk az árumozgatás költségein, vagy pedig különböző ütemezési algoritmusokat vezetünk be arra, hogy egy-egy munkadarabot hogyan, mikor ütemezzünk egy gyártósorra, annak érdekében, hogy a termelés egyenletes legyen és a gépeinket maximálisan kihasználjuk anélkül, hogy túlságosan le lenne terhelve valamelyik részleg. Erre találták ki szintén a japánok a Just-In-Time (JIT) elvet, ami azt foglalja össze, hogy minden a megfelelő időben történjen (se túl korán, se túl későn) [1]. A kifejezés magába foglalja még azt is, hogy minden a megfelelő helyen, megfelelő mennyiségben és minőségben álljon rendelkezésre. Ezek együttes alkalmazása jelenti a JIT elvet. Azt, hogy mit jelent a „megfelelő” kifejezés, azt mindig a vevő határozza meg. Látható, hogy ennek az elvnek az alkalmazása arra törekszik, hogy a készleteket kiiktatva áramoljanak a termékek a gyártósorokhoz (ahol ez nem lehetséges, ott pedig arra kell törekedni, hogy minimalizáljuk a készletszinteket). Ennek az elvnek az alkalmazása kapcsán azonban meg kell jegyezni – szintén egy másik hazai vállalat példáját idevéve –, hogy nagyfokú kockázattal is jár. Egy magyarországi passzív elektronikai alkatrészek gyártásával foglalkozó vállalat esetében konkrétan felmerült ennek az elvnek az alkalmazása, talán azért is leginkább, mert egy japán konszern felvásárolta. Azonban a cég menedzserei néhány hónap – ezen elven történő – működés után azt tapasztalta, hogy nem tudja időben teljesíteni a megrendeléseit a vevői számára és ez vagy presztízsveszteséggel járt számára, vagy pedig kötbérfizetéssel.

Ahogy azt már említettük, a termelésirányítás területén a nagy cégek fejlesztőmérnökei időről-időre találtak fel új, egyre hatékonyabb algoritmusokat. Kutatásaink során megvizsgáltunk számos algoritmust, melyek közül néhányat röviden bemutatunk itt, de ennek a tanulmánynak nem célja, hogy az algoritmusok mélységeibe is belemenyjünk:

- *Johnson-féle* megoldás olyan egy utas sorba rendezési feladatok kivitelezését teszi lehetővé, melyeknél két egymást követő géphez kell rendelni n számú munkát. Speciális esetben ez három gépre is kiterjeszhető (dinamikus programozás) [4].
- Egy Shop jellegű probléma megoldására *Jackson* dolgozta ki a kétgépes ütemezést végző algoritmust [5]:
 - a Shop jellegű ütemezési algoritmusok általános formájában n munkát kell elvégezni m rendelkezésre álló gép segítségével,
 - minden egyes munka több műveletből állhat, melyeknek adott a végrehajtási ideje,
 - adott munkára vonatkozó műveletek sorrendje tetszőleges is lehet.
- Léteznek az úgynevezett egyutas, előzésmentes nagyfeladatok, amelyek esetén a feladatokat végrehajtása több mint két gépet, berendezést igényel, ám minden feladat megmunkálási sorrendje azonos, így az alkatrészeknek ugyanolyan sorrendben kell végighaladniuk minden egyes gépen. Erre a problémára egy egyszerű és gyors megoldást nyújt *Palmer* módszere [6].
- Ide tartoznak még a diszkrét matematika területéről is ismert és elfogadott eljárások:
 - A *PERT*¹ módszer, amely a Program becslési és elemzési technikája eljárást jelenti.
 - A *CPM*², amely a Kritikus Út Módszer, ami egy hálós tervezési eszköz.

Az itt felsorolt eljárások a téma ismerői számára mind ismertek. Léteznek még további algoritmusok, melyeket még vizsgáltunk a terület mélyebb megismerése miatt: Lomincki megoldása [7], Campbell, Dudek és Smith módszere [8], Dannenbring algoritmus [9].

Végül röviden áttekintjük a történetét a termelésmenedzsment megoldásoknak [10]:

- 1910-es évek: munkavégzés tudományos elemzése - Taylor, szerelőszalag - Ford, ütemezési diagram – Gantt diagram;
- 1947: lineáris programozás – Dantzig;
- 1950-60-as évek: operációkutatási módszerek fejlődése (szimuláció, sorbanállás, döntéselmélet, matematikai programozás, hálózattervezés);
- 1970-es évek: számítástechnikai eszközökön termelésprogramozás, MRP³ rendszer;
- 1980-as évek: automatizálás, TQM⁴, JIT, CIM⁵, CAD/CAM⁶;
- 1990-es évek: korszerű eszközök (informatika, algoritmikus szoftverek) elterjedt alkalmazása, szolgáltatások termelése sokkal elterjedtebbé vált.

3. Termelésinformatikai szoftverrendszerek és esettanulmányok

A termelésirányításhoz szorosan kapcsolódnak a vállalatirányítási rendszerek (ERP⁷), többnyire azok egy konkrétan körülhatárolható része. A vállalatirányítási rendszerek moduláris felépítése miatt a termelésirányítás is integrálódik ebbe a környezetbe, általában egy konkrét modulként. Ez a modul nem feltétlenül része az alaprendszernek, mivel csak a

¹ Program Evaluation and Review Technique

² Critical Path Method

³ Manufacture Resource Planning (Gyártási erőforrás tervezése)

⁴ Total Quality Management (Teljeskörű minőség menedzsment)

⁵ Computer Integrated Manufacturing (Számítógéppel integrált termelés)

⁶ Computer-Aided Design/Computer Aided Manufacturing (Számítógéppel segített tervezés/számítógéppel segített gyártás)

⁷ Enterprise Resource Planning

termeléssel foglalkozó vállalatoknak lehet rá szüksége. Ekkor ők ezt a plusz modult megvásárolhatják az alaprendszerükhöz.

A vállalatirányítási rendszer kiválasztása egy fontos kérdés a vállalat életében. Nem mindegy, hogy egy standard rendszert választ ki a cég, vagy egyedileg fejleszti/fejleszteti le a saját cégére tökéletesen illeszkedő ERP megoldást. Általánosan elmondható, hogy az egyedi fejlesztés elsősorban a kis- és középvállalkozásoknak érheti meg napjainkban. Az ERP rendszerek piaca egyre inkább telítődik és mindenfajta vállalat számára készülnek ilyen dobozos megoldások. Tehát csakis akkor érdemes egyedi fejlesztésbe kezdeni, ha a vállalat annyira specifikus, hogy az nehezen lenne kezelhető egy ilyen standard rendszer esetében, esetleg túl drága lenne lefejleszteni a cég speciális körülményeihez kapcsolódó problémák megoldását elvégző bővítményeket.

Az ismert és népszerű vállalatirányítási rendszereket manapság már mindenki megismerheti különböző könyvek [11], tanfolyamok keretein belül. Ezek közé tartoznak a legismertebb SAP, Microsoft Dynamics NAV / AX, IFS Applications stb. rendszerek. Ezeknek a szoftvermegoldásoknak többnyire léteznek honosított verziói is. Ez annyit jelent, hogy létezik hozzájuk magyar nyelvű kezelőfelület, követik az adott ország jogi szabályozásait, valamint az országban több partnercég foglalkozik ezek értékesítésével, támogatásával, valamint plusz modulok egyedi fejlesztésével is.

Az általunk leginkább ismert Microsoft Dynamics NAV alkalmazásnak is léteznek gyengeségei: csak alapvető funkciókkal rendelkezik a termelésirányítás területén, mint például darabjegyzékek és művelettervek definiálása, műhelyek és munkanaptárak meghatározása, munkák FIFO elv szerinti adagolása stb. De nem rendelkezik azokkal az új algoritmusokkal, amelyek elvégzik például összetett munkák több gépre való ütemezését. Ezeket a bonyolultabb algoritmusokat külön hozzá kell fejleszteni a rendszerekhez, feltéve, ha a licencünk lehetőséget ad erre, illetve képesek vagyunk az adott fejlesztőkörnyezetben algoritmusokat implementálni.

3.1 Hazai fejlesztésű standard rendszer bemutatása

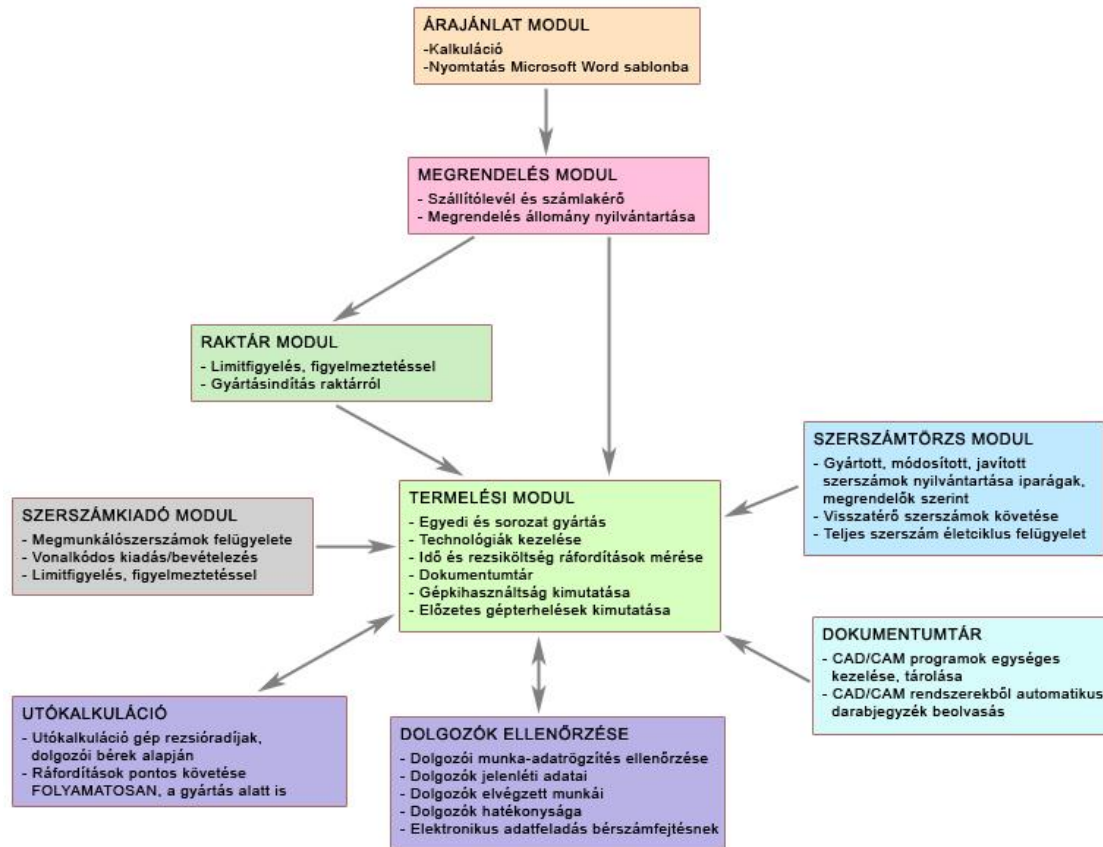
Magyarországon azonban nem csak a külföldi multinacionális cégek kínálják termékeiket, hanem hazai vállalkozások is megjelentek a vállalatirányítási és termelésirányítási rendszerek piacán saját termékeikkel.

Ebben a témakörben a *Norma Termelésirányítási Rendszert* mutatjuk be [12]. A rendszer fejlesztését 2003-ban kezdték meg, amelyhez hozzájárult a készítő 30 éves termelésvezetői tapasztalata, valamint az a motiváció, hogy fiatal programozókként a hazai piacon egy egyedülálló, magas színvonalú termelésirányítási szoftvert készítsenek el. A rendszer elsősorban az egyedi gyártás kezelésére lett optimalizálva úgy, hogy konkrétan a termelésre magára koncentrál. Továbbá a rendszer megoldást nyújt speciálisan a szerszámgyártásra, alkatrészgyártásra, újra- és sorozatgyártás nyomon követésére, irányítására. Lehetővé teszi még a termelés dokumentálását, visszakereshetőségét. Kimutatások készíthetők vele a dolgozók hatékonyságáról, a munkavégzésre fordított időkről, valamint a gyártógépek kihasználtságáról. Megtehetjük azt is, hogy a tervezés során megállapított normaidőt és normaköltséget a termelés végén összevessük a végső tényidővel és tényköltséggel, így mérhető a dolgozók termelési hatékonysága.

A Norma Termelésekövető rendszer ideális megoldást nyújt a következőkre:

- Árajánlat készítés, kiadott árajánlatok nyilvántartása (Ajánlat modul);
- Megrendelések és az elvégzett munkák nyilvántartása (Megrendelés modul);
- Elvégzett munkák valós idejű követése;
- Egyedi- és sorozatgyártás támogatása;
- Projektekre, megrendelésekre fordított idők, erőforrások mérése, nyilvántartása (Jelenléti ív, Szerszámkiadó, Dokumentumtár, Darabjegyzék modul);

- Dolgozók teljesítményének mérése, előzetes illetve utólagos munkaidő utalványozás (Dolgozói munkaidők, Dolgozói teljesítmények mérése);
- Visszatérő megrendelések kezelése;
- Árak, rezsiköltségek, megtérülési hatékonyság kimutatása (Pénzügyi adatok).



1. ábra. A rendszer működésének vázlata

3.2 Standard rendszereket kiegészítő saját fejlesztés

Ahogy azt már korábban említettük, a termelésirányítással foglalkozó szoftverek általában egy vállalatirányítási rendszer részeként kerülnek megvalósításra, azok extra moduljaként. Léteznek azonban olyan vállalatirányítási rendszerek, amelynek az adott, esetleg régebbi verziója nem rendelkezik termelésirányítási modullal. Az a cég, aki rendelkezik vállalatirányítási rendszerrel, de szüksége lenne a hiányzó termelésirányítási funkciókra, valószínűleg rosszul választotta meg a megvásárolt rendszerét. Egy másik opció lehet, hogy ha a vállalatirányítási rendszer mégis bővíthető lenne termelésirányítási funkciókkal, de az túl drága és/vagy túl általános a cég számára, akkor azt nem veszi meg. Ekkor történhet meg az, hogy az adott vállalkozás nem veszi meg a vállalatirányítási rendszer bővítményét (ha nincs ilyen, akkor nem is tudná megvenni), hanem a meglévő rendszere mellé egy cégre szabott megoldást fejleszt vagy fejlesztet. Ilyenkor legtöbb esetben biztosítani kell a lehetőséget, hogy az adott termelésirányítási funkciókat ellátó alkalmazás valamilyen interfészen keresztül képes legyen kommunikálni a már meglévő vállalatirányítási rendszerrel. Ehhez azonban előbb létre kell hozni azt az alkalmazást, amely képes vizsgálni az adott gyártási gépek működését, esetlegesen egyéb gépeknek jelez, hogy mit és mikor kellene működésbe lépnie. Ezután történhet meg az adatok biztonságos összegyűjtése és adott helyen való eltárolása. Majd egy esetleges visszajelentés történhet a vállalatirányítási rendszer felé, hogy éppen melyik fázisában áll a termelési folyamat (opcionális lépés). Ha ezek már mind sikeresen

rendelkezésünkre állnak, akkor kerülhet sor olyan kimutatások készítésére, amelyekre a cég vezetőségének szüksége van. A termeléssel kapcsolatos adatok elemzésére a jelenlegi informatikai megoldások adnak lehetőséget: kezdve az alap statisztikai/matematikai kimutatásokkal, egészen a nagy műveletigényű, hosszú lefutási idejű adatbányászati feladatok megoldásáig. Nekünk lehetőségünk volt egy nagyon hasonló projekt egyes feladataiban részt venni, így testközelből láthattuk ennek a bonyolultságát egy faipari/bútoripari vállalat esetében.

3.3 Standard rendszereket kiegészítő gyári fejlesztés

A következő hasonló, ám egy kicsit egyszerűbb megoldás egy elektronikai részegységek, berendezések félkészre, illetve készre szerelését, csomagolását végzi. Ez a cég több termelésirányítási informatikai rendszert használ egyidejűleg. A rendszerek modulokból épülnek fel, amelyek egymás mellett különböző területek munkáit segítik. Mi ebben az alfejezetben most egy olyan megoldásra térünk ki, amely egy adott eszközhöz készült szoftvert mutat be. A cégnek a jó sorkihasználtság és még jobb termelékenységi érdekében szüksége volt egy jó monitorozó szoftverre, amivel nyomon követhető a gyártósorok minden fontos adata. Ehhez a Siemens cég beültető gépét alkalmazták, valamint a hozzá fejlesztett szoftvert. Ez a szoftver biztosítja a gépkezelőknek az összes fontos üzemi adatot:

- rendelkezésre állási idő,
- kapacitáskihasználtság,
- selejt mennyisége.

A rendszer nagyon fontos része a trendmutatók kezelése. Ezek figyelik, és ha kell, figyelmeztetik a gép kezelőjét, hogy valami nem a rendeltetésének megfelelően működik. Lehet ez valamilyen előre beállított küszöbérték túllépése, például túl sok alkatrész elszórása stb.

A rendszer szerver-kliens struktúrájú. Minden beültető sorhoz tartozik egy úgynevezett sorvezérlő számítógép, amely minden fontos gyártási adatot tartalmaz. Általában erre a PC-re szokták a szerver szoftverét telepíteni, míg minden egyes beültető gépre kliens szoftver kerül.

A szoftver segítségével a kezelőjének lehetősége nyílik a következőkre:

- biztonsági hozzáférések beállítása,
- különböző határértékek beállítása
 - hulladékszázalék: az alkatrészek elszórási mutatójáról ad információt (TOP 3),
 - elérhetőség: a gépidő rendelkezésre állásáról kapunk információt (fut, várakozik, blokkolja valami, meghibásodott stb.),
- monitorozási időszakok beállítása,
- hálózati kapcsolatok beállítása.

Látható tehát, hogy nem csak vállalatirányítási rendszerek nyújthatnak olyan szolgáltatásokat, amelyek segítségével termelésirányító, monitorozó funkciókat tudunk ellátni, hanem például az itt röviden ismertetett szoftver is segítségünkre lehet ebben a problématerben.

4. Tapasztalatok beépítése az oktatásba

Természetesen céljaink között szerepel, hogy a kutatáshoz kapcsolódó anyagokat beépítsük az oktatásunkba. Elsősorban a Termelésinformatika c. tantárgy kapcsán merülhet fel ennek szüksége. Szerencsére elkezdődött egy olyan jótékony folyamat, amely előremutató lesz a tantárgyunk és annak tematikájának fejlődésében. Segítségünkre volt és lesz is a szorgalmas

diákok csoportja Sopronban és Zalaegerszegen is. Mindkét helyen történtek olyan együttműködések, amelyek támogatták nem csak ennek a kutatásnak a megalapozását, hanem a Termelésinformatika tantárgy elméleti anyagának bővítését is. Ez a tárgy a BME mérnök-informatikus hallgatóinál oktatott korábbi tananyag kivonatára épült, de szerencsére, egyre inkább képes önmagában – a BME-s alapoktól elkülönülten – kerek egészet alkotni.

Jelen kutatásunkban nem mindenhol nevesítettük a vizsgált cégeket közvetlenül, ennek több oka is van. De megjegyeznénk, hogy van több olyan vállalkozás is, amely jelezte felénk, hogy az oktatás területén, tájékoztató jelleggel megismertethetjük a cégek termelésirányítási rendszerével a hallgatóinkat. Sajnos erre csak az elméleti oktatás területén történhetnek próbálkozások, mivel általában ezek a rendszerek annyira robosztusak, hogy mindegyiknek külön-külön speciális környezetet kellene kialakítani. Gondolunk itt elsősorban a platformfüggőségre, ami annyit tesz, nem mindegy, hogy az adott rendszer Windows vagy valamilyen Unix operációs rendszeren képes futni. Ez azonban még talán csak a legkisebb, orvosolható probléma, mivel egyetemünk nem rendelkezik az adott szoftverek oktatási célú verziójával – soknak nem is létezik ilyen verziója. Egy lehetőség lehetne, hogy ha – akár csoportosan is – megkezdődhetnének céglátogatások és ott a rendszer egy éles működését mutathatnák be a helyi szakértők hallgatóinknak. Ennek viszont főleg anyagi korlátait látjuk jelen pillanatban.

Ahogy azt említettük, egyetemünk a Microsoft Dynamics NAV vállalatirányítási rendszer egy oktatási verziójával rendelkezik, amely tartalmazza a termelésirányítási modulhoz való hozzáférést is. Ehhez a rendszerhez még 2007-ben készített hallgatók és oktatók egy lelkes csoportja gyakorlati feladatsorokat. Ezeket az idők során, ahogy új verziók jelentek meg az NAV rendszerből, mindig frissítettük. Ennek a kutatásnak köszönhetően ráláthattunk olyan problémákra is élesen használt rendszereknél, amelyek szintén beépíthetők a Termelésinformatika tantárgy gyakorlati anyagába.

5. Összefoglalás

Tanulmányunkban egy átfogó kutatás bevezető részét mutattuk be. Céljaink között szerepel egy hatékony, hazai gyártást támogató program tervezése, amelyhez először felmértük a hazai piac helyzetét. A röviden bemutatott megoldások mind egy-egy ilyen problémakört oldanak meg a gyárak működésében. Bemutattunk néhány fejlődési irányvonalat, amelyet az elmúlt évtizedekben alkalmaztak, kitértünk néhány termelésirányítási algoritmusra, amelyek szintén az optimalizálást segítik. Majd rátértünk a konkrét termelésirányítási rendszerek bemutatására, megemlítve a dobozos, standard rendszereket, valamint konkrét példákkal szemlélítve a hazai saját fejlesztésű szoftvereket. Végül pedig a szerzett tapasztalatok oktatásba való beépítéséről vázoltunk fel egy tervet, amelynek a részleges végrehajtása már meg is kezdődött.

6. Köszönetnyilvánítás

Ez a tanulmány a TÁMOP-4.2.2.C–11/1/KONV-2012-0015 számú (Föld-rendszer) projekt keretében, az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

Irodalomjegyzék

- [1] Taiichi Ōno: Toyota Production System: Beyond Large-Scale Production, *Productivity Press* (1988).
- [2] Az autópárt is padlóra küldte a válság, *mfor.hu - Menedzsment Fórum*, 2008. október 28. http://www.mfor.hu/cikkek/Az_autoipart_is_padlora_kuldte_a_valzag.html
Utolsó megtekintés dátuma: 2014. június 19.
- [3] A Toyota ismét felrázhatja a világot, *Portfolio.hu*, 2014. május 28. http://portfolio.hu/vallalatok/cegauto/a_toyota_ismet_felrazhatja_a_vilagot.199468.html
Utolsó megtekintés dátuma: 2014. június 19.
- [4] Kulcsár Gyula: Ütemezési modell és heurisztikus módszerek az igény szerinti tömeggyártás finomprogramozásának támogatására, PhD értekezés, Miskolci Egyetem, (2007)
http://ait.iit.uni-miskolc.hu/~kulcsar/nyilvanos_vedes/KulcsarGy_ertekezes.pdf
Utolsó megtekintés dátuma: 2014. június 5.
- [5] Jackson J. R.: An Extension of Johnson's Results on Job Lot Scheduling, *Naval Research Logistics Quarterly*, **Vol. 2** (1956), pp 201-203.
- [6] Palmer, D. S.: Sequencing jobs through a multi-stage process in the minimum total time-- a quickmethod of obtaining a near optimum. *Operational Research Quarterly*16, **Vol. 1** (1965),101-107.
- [7] Lomincki Z. A.: A Branch and Bound Algorithm for the Exact Solution of the Three-Machine Scheduling Problem, *Operational Research Quarterly*, **Vol. 16** (1965), pp. 89-100.
- [8] Campbell H. G., Dudek R. A., Smith M. L.: A Heuristic Algorithm for the n-Job m-Machine Sequencing Problem, *Management Science*, **Vol. 16** (1970), No. 10, pp. 630-637.
- [9] Dannenbring D. G.: An evaluation of flow shop sequencing heuristics. *Management Science* **23** (1977), pp. 1174-1182.
- [10] Koltai Tamás: Termelésmenedzsment, *Tyotex* (2006)
- [11] Heteyi József: ERP rendszerek Magyarországon a 21. században, 2. kiadás új rendszerekkel, *Computerbooks* (2009).
- [12] Norma Termelési Rendszer – Hatékonyság a szerszám- és alkatrészgyártásban, <http://www.vallalat-iranyitas.hu/>
Utolsó megtekintés dátuma: 2014. június 22.

Kollaboratív környezet és ismeretbázis építése a mezőgazdasági erdészeti oktatására az AgroFE Leonardo nemzetközi projektben

Collaborative Environment and Building Learning Knowledge Base for Agro-Forestry Trainings

Herdon Miklós^a, Burriel, Charles^b, Tamás János^c, Várallyai László^d, Lengyel Péter^e, Pancsira János^f

^{ade}Debreceni Egyetem/Gazdálkodástudományi Kar

^aherdon@agr.unideb.hu, ^dvaral@agr.unideb.hu, ^elengyel@agr.unideb.hu, ^fpancsira@agr.unideb.hu

^bAgrosup Dijon

charles.burriel@educagri.fr

^cDebreceni Egyetem/Mezőgazdaság-, Élelmiszertudományi és Környezetgazdálkodási Kar

tamas@agr.unideb.hu

Absztrakt: Az egyes európai országokban az egyes kormányok, szakmai szervezetek és a képzésekben érdekelt oktatási intézmények különböző hangsúllyal megpróbálják újraéleszteni, fejleszteni a mezőgazdasági erdészeti oktatását a különböző képzési formákban és képzési szinteken. A projekt célja ennek a folyamatnak az erősítése, serkentése a következő főbb feladatok megoldásával: a.) az igények és elvárások szintetizálása meglévő képzések figyelembevételével; b.) képzési programok és tananyagok kifejlesztése; c.) egy új innovatív kontextus függő, moduláris, IKT alapú képzési rendszer kialakítása; d.) kollaboratív rendszeren alapuló kommunikációt, fejlesztést és interaktív szolgáltatást támogató rendszer és ismeretbázis létrehozása. A rendszer architektúra alapja a nyílt forráskódú koncepció, amely a nyílt forráskódú alrendszerek (kereső, tartalomkezelő, munkafolyamat menedzselő, publikáló, e-learning) integrálásával valósul meg. A rendszerben jellegénél fogva sokféle fájl- és adatformátum kezelése szükséges. A projektben 6 európai országból 14 partner vesz részt.

Kulcsszavak: mezőgazdasági erdészeti, e-learning, oktatás, tudásmenedzsment

Abstract: Depending on the European countries, states or professional organizations and training actors try to reintroduce Agro forestry in the course of training and qualification in initial training and in adult education. The main objectives are a.) to make a synthesis of needs and expectations, b.) based on present the existing training actions set up a common framework; c.) to build an innovative training system (contextualized, modularized trainings, use of ICT, professionals participation); d.) to create a technical collaborative support for the implementation of the project with communication tools (information of partners and promotion) and for providing access to the resources and training services during and after the project (knowledge databank, interactive services). The system architecture is based on open source concept. Open source subsystems are the Search Engine (harvester), Digital Asses Management System (KDB), Content Management System for internet portal, Editorial Chain (workflow) Management, E-Learning platform. 14 partner from 6 European countries have been involved in the project.

Keywords: Agro-forestry, e-learning, trainings, knowledge management

1. Bevezetés

A mezőgazdasági ágazatban erőteljes lemondás volt tapasztalható a mezőgazdasági erdőgazdálkodásról (Agroforestry) a 20. században, ezért ma már csak néhány millió hektár ilyen terület van Európában. Az elmúlt években néhány szakember figyelme a mezőgazdasági erdészeti tudományos kutatása, fejlesztési struktúrák kialakítása felé fordult, amely találkozott és elismerést vívott ki nemzeti és európai szinten egyaránt. A tudományos kutatási és fejlesztési eredmények alapján kísérleti tanfolyamokat szerveztek a farmerek számára a különböző országokban, köztük Belgiumban, Franciaországban, az Egyesült Királyságban. A projektben részt vevő partnerek rövid időtartamú képzési igényeket azonosítottak. Ezen képzési igények, illetve szükségletek célcsoportjai a projekt tagjai, a jövőbeli képzést végzők, a felnőttek és tanulók / diákok, tanárok, valamint tanácsadók. Ezek az igények különböző szintű képzési programokhoz (L4 és L5/L6), végzettségekhez kapcsolódnak [5]. Rövidtávon a projekt nyilvános képzési programok létrehozását tűzte ki, innovatív megoldások kialakításával. Szerencsére a fejlett és rohamosan fejlődő IKT eszközök és módszerek, az e-learning és az e-együttműködést biztosító megoldások egyre szélesebb tárháza áll rendelkezésre [1, 3, 4]. A projektben használt és tervezett rendszer architektúrát ismertetjük a következő fejezetekben.

2. Célkitűzések

A projekt fő célkitűzése, hogy a képzési igények és elvárások szintézise alapján a jelenlegi meglévő képzési programok figyelembevételével hozzon létre egy közös keretet. Ebben alakítson ki egy újszerű képzési rendszert (moduláris képzés, IKT szakemberek részvétel), hozzon létre egy technikai együttműködési támogatást a projekt megvalósításához kommunikációs eszközökkel (tájékoztatás a partnerek számára és promóció) és biztosítson hozzáférést a projekt alatt és után a tudásbázishoz és a képzési szolgáltatásokhoz (tudás adatbank, interaktív szolgáltatások). Ezen célok elérése érdekében a következő fő tevékenységeket kell elvégezni:

- Hasznosítani az elérhető eszközöket és szolgáltatásokat.
- Kiepíteni ki egy kollaboratív munkakörnyezetet.
- Megtervezni egy az oktatás és képzés igényeit kielégítő architektúrát.
- Megvalósítani az e-learning környezetet.
- Megtervezni azon multimédiás környezetet, hogy a rendszer tartalma és szolgáltatásai elérhetőek legyenek a tanulók és oktatók számára.

Ezen célokra és feladatokra alapozva építünk egy közös munka környezetet a projektpartnerek és a szereplők számára, akik csatlakoznak ehhez a tudásbázishoz és információs szolgáltatáshoz. Ennek megvalósítását a következő módszerekkel (tevékenységekkel) fogjuk elérni:

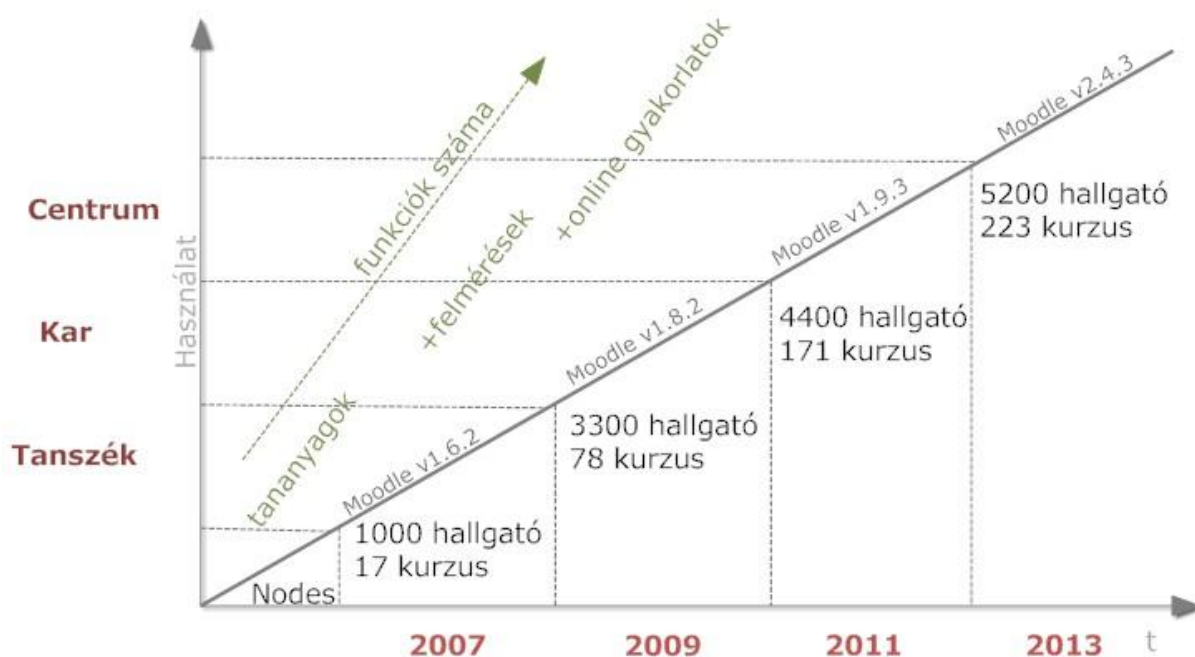
- A korábbi projektek és gyakorlati tapasztalatok alkalmazása.
- Az új technológiák és módszerek tanulmányozása.
- Az új IKT eszközök és alkalmazások értékelése.
- A felhasználásra alkalmas (a célnak és elvárásoknak legjobban megfelelő) eszközök kiválasztása.

3. Eredmények

A közös kollaboratív munkakörnyezet (virtual working space) kialakítása meglévő nyílt forráskódú és ingyenes szolgáltatásokat (open source, open access) alkalmazzuk. Az egyik alapvető megoldás volt a Moodle rendszer legújabb verziója. Egy kiválasztási szempont volt, hogy már több mint 7 éves tapasztalattal rendelkezünk az alkalmazásáról, amely funkcionálisan kielégíti a szükséges közös projekt munka követelményeit. Az kialakítandó rendszer egy tudásbázison alapuló szolgáltatásra épül, amelyben a mobil (terepi munka) alkalmazások és szolgáltatáselérések is alapvető követelmények Android, IOS, Windows platformon egyaránt.

3.1 A Moodle rendszer használata virtuális munkakörnyezetként

Sok más felsőoktatási intézményhez hasonlóan már 2007-ben bevezettük a Moodle rendszert. A kari vezetők elismerték a korábbi projektjeink kutatási eredményeit és felismerték azt a tényt, hogy a modern technológiákat az oktatásban be kell vezetni, amely meg is valósult a Moodle rendszer bevezetésével [2, 6]. A bevezetés fázisai az 1. ábrán látható.



1. ábra: e-Learning rendszer implementációjának fázisai a Debreceni Egyetem Agrártudományi Centrumában.

Tanszéki használata és tesztelése 2008 januárjában kezdődött a Gazdálkodástudományi és Vidékfejlesztési Karon. Bevezetése óta a Moodle rendszer folyamatosan frissül. A rendszer által támogatott képzések és kurzusok gyorsan növekedtek és vele együtt természetesen a felhasználók (tanárok és diákok) száma is. Az informatikus és szakigazgatási agrármérnök szak 2009-ben került bevezetésre, így a kurzusok és felhasználók száma tovább növekedett. A rendszer frissítésére, az új verziók bevezetésére többször került sor, annak érdekében, hogy az újabb funkciókat tudjuk alkalmazni az e-learning még hatékonyabbá tételé érdekében. Az e-Learning rendszerek mint kollaboratív rendszerek [7] alkalmasak a fejlesztés, megvalósítás és az oktatás fázisaiban is az együttműködések, a közös munka támogatására. A múltban is több európai projektben, a felnőttképzésben is alkalmazott Moodle rendszert az előzetes

tapasztalatok alapján alkalmasnak ítéltük, hogy az AgroFE projektben a közös virtuális munkakörnyezetet a rendszerrel valósítsuk meg, így ez bevezetésre került (<https://moodle.agr.unideb.hu/agrofe/>).

3.2 Videokonferencia rendszer használata

A másik funkció, hogy támogatnunk kell a virtuális találkozókat, videokonferenciákat. Az NIIF nagy sebességű kutatási és oktatási hálózat, lehetővé teszi ennek biztosítását. A rendszer által biztosított két fontos együttműködési szolgáltatás (<http://www.niif.hu/en/>):

- Video- és asztali konferencia rendszer (IP alapú videokonferencia), amely bárhol bárhová, bármilyen résztvevő száma számára biztosítja a szolgáltatást az NIIF intézmények számára.
- A Videotorium egy videó megosztó portál a felsőoktatási, a kutatási és a közgyűjteményi intézmények és alkalmazottaik számára, amellyel akár nagy felbontású (HD) minőségű kutatási és oktatási felvételek oszthatók meg az érdeklődő felhasználók körében.

A videó hálózatban jelenleg található mintegy 140 tárgyalóterem terminálok és egy asztali videokonferencia rendszer is rendelkezésre áll. Az AgroFE projektben használjuk az MCU (Multipoint Control Unit) rendszert, amely HD szolgáltatást, képes felvenni és / vagy közvetíteni az interneten. Az asztali (Desktop) rendszer is használható, mert minden partner csatlakozhat egy virtuális tárgyalóteremhez, amely az összes résztvevővel csatlakoztatható az MCU szerverhez. Ez az asztali rendszer a Vidyo szoftverrel működik.

3.3 A Vidyo rendszer

A Vidyo portfólió (<http://www.vidyo.com/>) mindent tartalmaz, amit telepíteni kell: a HD video együttműködéstől a központi infrastruktúra megoldásokig. A Vidyo úgy működik, ahogy mi akarjuk. Az általunk használt eszközök mindegyikén fut, az intelligens telefonoktól a tableteken keresztül, az asztali videó-rendszerekig HD minőségű videót biztosít minden résztvevőnek. A VidyoDesktop alkalmazás kiterjeszti a kiváló minőségű videokonferenciát Windows, Mac és Linux számítógépekre, így e rendszerek felhasználói is részt vehetnek videokonferencián az irodában, otthon, vagy akár útközben is. Támogatásával full HD és dual-screen multipoint video is készíthető. A legfontosabb funkciói:

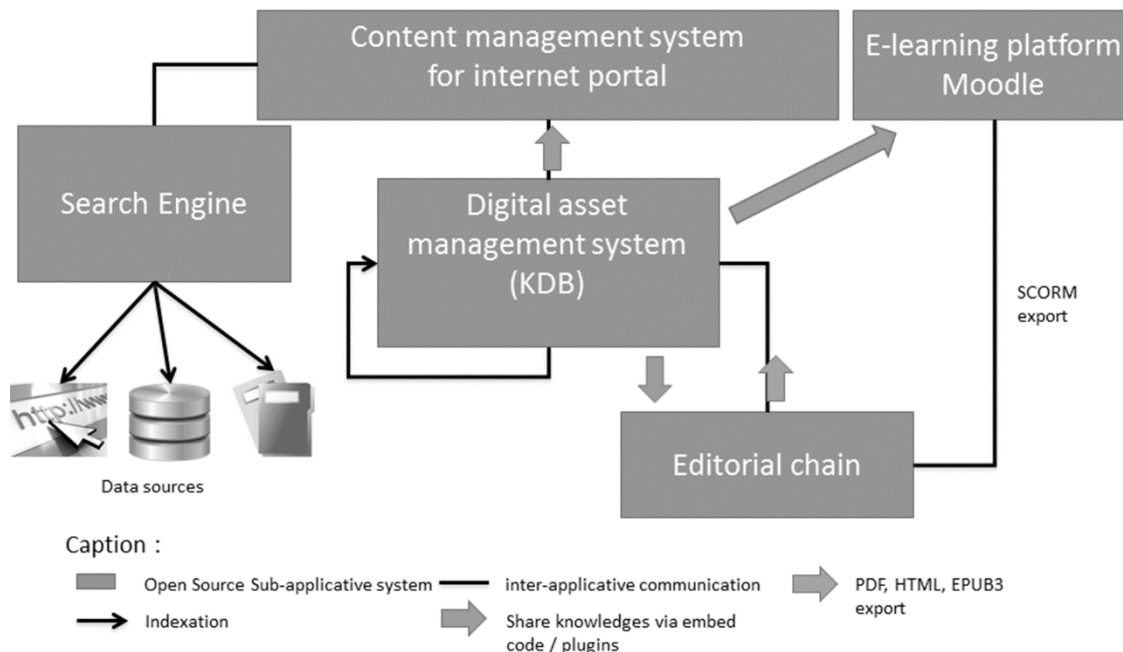
- 2X Extreme Definition (XD): Ez jeleníti meg a résztvevőket és a tartalmakat egy vagy két monitoron, egyenként akár 1440p felbontással.
- Több-felhasználós választható elrendezések a folyamatos jelenlétre, az aktív hangfalra és a megosztott tartalomra.
- Nyilvános és magán szöveges „chat”.
- Megosztható bármilyen alkalmazás vagy asztal a konferenciában, és válthatunk a több megosztott tartalom között.
- Távoli kamera vezérlés.

A VidyoMobile kiváló minőségű videokonferencia alkalmazás népszerű mind Apple és Android tableteken, okostelefonokon is. Részt vehetünk személy-személy vagy multi-party videokonferencián az irodában, otthon, vagy utazás közben mind a vezeték nélküli szélessávú és mind Wi-Fi kapcsolat használatával.

3.4 A rendszer felépítése

Korábbi e-learning kutatásra és fejlesztésre, az AgroFE projekt koncepciójára és előzetes munkákra alapozva megalkottuk a rendszer funkcionális modelljét (2. ábra), mely következő elemekből épül fel.

- Tudás-tár
- Tartalom menedzser
- Szerkesztői rendszer
- Keresőmotor.



2. Ábra. A tudásalapú szolgáltató rendszer felépítése (Forrás: Charles Burriel)

3.4.1 Tudásbázis rendszerek

A tudásbázis és a tudás bank egy speciális adatbázis a tudásmenedzsmenthez. A tudásbázis egy olyan adattár, amely eszközöket biztosít az információ gyűjtéséhez, a szervezéséhez, megosztásához, kereséséhez és felhasználásához. Ez lehet akár gép által olvasható vagy emberi felhasználásra szánt tudásbázis. Az adatbank háttérében egy adatbázis-kezelő rendszer van. Néhány tudásbázis szoftver a következő (<http://www.capterra.com/>):

- PHPKB Knowledge Base Software
- OPEN KM Knowledge Management
- Phraseanet.

A **PHPKB**-t egy teljes tudásmenedzsment megoldásra tervezték. PHP-ban fejlesztették MySQL adatbázisra építve és teljes forráskód elérhető, nem titkosított. Az **OpenKM** (<http://www.openkm.com/en/>) egy web alapú dokumentum menedzsment alkalmazás, amely szabványokat és a nyílt forráskódú technológiákat használ. Az OpenKM teljes körű dokumentum menedzsmentet biztosít beleértve a verziókövetést, a fájl történetet, a metaadatokat, a keresést, stb. A **Phraseanet** (<https://docs.phraseanet.com/3.8/en/>) egy Digital Asset Management rendszer (DAM) célja a szakemberek igényeinek kiszolgálása. A Phraseanet tud tárolni, kezelni és megosztani különböző digitális forrásokat, mint a kép, videó, hang vagy hivatalos dokumentumokat különböző méretű munkacsoportokban. Mivel webes alkalmazás, minden modern böngészővel futtatható. Mivel szabványokra (MySQL, a

PHP és Apache alapokra) épült, így telepíthető Linux, Mac OS X vagy Windows rendszerekre is.

3.4.2 Tartalom-menedzsment rendszerek

Napjaink vezető tartalomkezelő rendszerei a Drupal, a Joomla és a WordPress. Mind a háromnak megvan a saját felhasználói csoportja és mindegyiknek meg van a maga előnye és hátránya. A három rendszer között nincs túl nagy szakadék, persze van amelyik egy bizonyos feladatra jobb mint a másik, viszont ettől még nem jelenthetjük ki hogy jobb is lenne a többinél.

A legnépszerűbb közülük WordPress, mely 2003-ban indult. A koncepció egyszerű volt, kellett a blogolási platform. Az elején valójában, csak egy blog volt, majd a későbbiekben fejlődött CMS-é. Ennek köszönhetően még napjainkban is előfordul, hogy a WordPress-t csak egy blog rendszernek tekintik, pedig egy ideje már CMSként funkcionál. Mi sem bizonyítja jobban, mint a Widget rendszer, melynek segítségével "Fogd és Ejtsd" módszerrel tudjuk bővíteni az oldalunk funkcióját különböző modulokkal, mint egy lapozó, plusz menü vagy éppen egy Twitter feed. Egy Joomla-ról átpártolónak okozhat némi fejtörést a menü kezelés, tekintettel arra, hogy a mag egy blog motor, amiben szükségtelen volt a komplikált menü rendszer.

3.4.3 Opale rendszer

Az **Opale** egy publikációs rendszer, mellyel egyetemi képzési dokumentumokat és multimédiás anyagokat lehet menedzselni. A rendszerrel a következő dokumentumtípusok készíthetők: prezentáció videó vetítéshez (Flash), nyomtatható dokumentumok (Open Office ODT és PDF), web dokumentumok (XHTML vagy HTML 4.01 Transitional - W3C szabványok), exportálható tartalmak egységes sémában (XML).

3.4.4 Keresőmotorok

A keresőmotor az informatikában egy program vagy alkalmazás, amely bizonyos feltételeknek (többnyire egy szónak vagy kifejezésnek) megfelelő információkat keres valamilyen számítógépes környezetben. A **Sphinx** egy úgynevezett full-text keresőmotor, ami rendkívül alkalmas nagyobb mennyiségű szövegben történő keresésre. Ha nagyobb mennyiségű szövegben keresünk, akkor mindenképpen jó választás, de akkor is, ha címke szerint szeretnénk megjeleníteni a keresési eredményeket, továbbá jó lehet azokban az esetekben is, amikor az adott szövegekörnyezet alapján a minél relevánsabb találatok megjelenítése a cél, ugyanis a nyers teljesítmény mellett ez a Sphinx másik erőssége. Számátalan keresési lehetőséget és többféle súlyozási algoritmust kínál. Az **OpenSearchServer** egy nagy teljesítményű, vállalati szintű kereső program. A webes felhasználói felület, a robotok (web, fájl, adatbázis, stb.), és a REST / REST API használatával integrálni lehet a fejlett full-text keresést bármely alkalmazásba. Az OpenSearchServer funkciói a következők:

- A teljes körű keresési funkciók
- Saját indexelési stratégia
- Teljesen integrált megoldás
- Teljes szöveges adatok elemzése
- A robotok mindent indexelhetnek.

Az **Elasticsearch** egy rugalmas és hatékony nyílt forráskódú, elosztott, valós idejű kereső és elemző motor. Felépítése, az alapoktól kezdve használható elosztott környezetben, ahol a megbízhatóság és bővíthetőség alapvető szempont.

3.4.5 Moodle tanulásmenedzsment rendszer – új funkciók

Azt tervezzük, hogy a Moodle rendszert fogjuk használni a projekt keretein belül az e-learning megvalósításához. Azt több éves tapasztalatainkra, a rendszer referenciákra, a rendszer széles funkcionalitására és új funkcióira alapozzuk. Néhány új, a projekt szempontjából is fontos funkció:

- Kapcsolat felhő szolgáltatással
- Mobil eszközök támogatása.

A Moodle összeköthető egy a felhő szolgáltatással. Több funkció van, melyek lehetővé teszik, hogy a Moodle-t több millió tanuló számára kibővítsük. Jobb adatfelhasználást és jelentést építettek a rendszerbe. Ez azt jelenti, hogy a fejlesztő szakemberek több információt nyerhetnek ki a Moodle-ből a tanulási tapasztalatokról. Ez a funkció a naplózó keretrendszer használja, és ez lehetővé teszi, hogy a Moodle közösség jobb jelentéseket tudjon összegyűjteni. A mobil eszközök támogatásával kapcsolatosan talán a legizgalmasabb újdonság, hogy elérhető bárhol a "push" elv alkalmazásával, amely az agrárgazdaságban is fontos alkalmazási technológia és terület [8].

4. Következtetések és jövőbeli feladatok

Az agrár-erdészeti tevékenység több szempontból fontos a vidéki térségek és a kisgazdaságok számára. Környezeti, gazdasági, mezőgazdasági termelési tevékenységekben és feladatokban fontos szerepet játszhat. A projekt résztvevői a szakterületre több képzési szintre tantervek kidolgozásával és képzési anyagokkal segítik a szakterület fejlesztését. Eddigi eredménye a projektnek, hogy a DE GVK és MÉK karain C-típusú tárgyként elkészült a „Mezőgazdasági erdészet az agrárkörnyezet gazdálkodásban”, amelyet a két kar kari tanácsa elfogadott. A kollaboratív munkakörnyezetet a Moodle rendszerben hoztuk létre, amely széles körű funkcionalitásával megfelelően támogatja a projekt munkát, a virtuális együttműködést. E mellett a virtuális találkozókra videokonferencia rendszereket használunk. A korábban használt Marratech és Skype rendszer mellett jelenleg az NIIF MCU és Desktop videokonferencia rendszerét alkalmazzuk. A projektben a nyílt forráskódú eszközökre és rendszerekre alapozva kívánjuk létrehozni a képzést európai dimenzióban támogató tudásbázist és szolgáltató rendszert. Az elképzelt rendszer architektúra építőelemeire alkalmas nyílt forráskódú rendszereket elemeztük, amelyből az e-Learning feladatokra a Moodle rendszert választottuk. Meggyőződésünk, hogy az innovatív technológiák és megoldások megfelelően fogják segíteni a projekt célok megvalósulását.

5. Köszönetnyilvánítás

A publikációban megjelentetett kutatást és annak megjelenését támogatta EU Leonardo Innovations Transfer “Agroforesterie Formation en Europe - AgroFE” Ref. Number: 2013-1-FR1-LEO05-48937 projekt.

Irodalomjegyzék

- [1] Bustos E. S., F. S. Zazueta és H. B. Howard: Rapid prototyping of learning objects and their implementation using ontology editor. *In Conference on Agricultural Economics, Rural Development and Informatics*. Debrecen, Hungary. (2007), 59-60.
- [2] Charles B.: NODES – E-learning aspects and accessibility. *In International Conference on Agricultural Economics, Rural Development and Informatics*, Debrecen, Hungary. (2007), 83-91.
- [3] Herdon M. és P. Lengyel: Building and Using Knowledge Repositories for Agriculture: An Innovation Case Study. *In: EFITA WCCA CIGR 2013 Conference: Sustainable Agriculture through ICT innovation*, Torino, Italy. (2013). 180-187.
- [4] Herdon M. és T. Rózsa: Knowledge dissemination on innovative information technologies in agriculture. *Journal of Ecoagritourism* **8:(1)** (2012), 301-306.
- [5] Herdon M., R. Szilágyi és L. Várallyai: ICT Tools for Implementation the European Qualification Framework in the Agricultural Sector. *Journal of Agricultural Informatics* **2:(1)** (2011), 29-40.
- [6] Lengyel P. és M. Herdon: Implementing learning design by LAMS to improve teaching and learning. *APSTRACT - Applied Studies in Agribusiness and Commerce* **3:(5-6)** (2009), 21-24.
- [7] Manouselis N., R. Vuorikari és F. V. Assche.: Collaborative recommendation of e-learning resources: an experimental investigation. *Journal of Computer Assisted Learning* **4** (2010), pp 227-242.
- [8] Szilágyi R.: New information and communication technologies in agriculture - factors, drivers and application possibilities. *Journal of Agricultural Informatics* **1** (2012), pp. 10-18.

Time-lapse felvételek elemzése fonalgomba növekedésének vizsgálatához

Analysis of time-lapse video recordings for investigation of fungal hyphae

Mátyás Anita^a, Dr. Szemán Nagy Gábor^b, Szeghalmy Szilvia^c

^aDebreceni Egyetem/Informatikai Kar/Gazdaságinformatikus
anita_6@freemail.hu

^bDebreceni Egyetem/Mikrobiális Biotechnológiai és Sejtbiológiai Tanszék
szeman-nagy.gabor@science.unideb.hu

^cDebreceni Egyetem/Komputergrafika és Képfeldolgozás Tanszék/Egyetemi tanárségéd
szeghalmy.szilvia@inf.unideb.hu

Absztrakt: Jelen dolgozat témája a Debreceni Egyetem Mikrobiális Biotechnológiai Tanszékének laborjában készült time-lapse felvételek feldolgozása. A time-lapse felvételek széles körben használhatók, készítésük során a szokásos (jellemzően 24 vagy 30 fps) helyett jóval hosszabb időközönként kerülnek rögzítésre a képkockák. Napjainkban mind gyakrabban alkalmazzák a time-lapse mikroszkópokat tenyésztett sejtpopulációk hosszú-távú megfigyeléséhez is. Alkalmazásukkal nagy, de kezelhető mennyiségű képi adathoz juthatunk, mely lehetőséget ad a sejttenyészetek időbeli viselkedésének elemzésére is.

A célunk egy olyan C++ nyelvű szoftver készítése, mely lehetővé teszi a laborban készült felvételek hatékonyon elemzését. A cikkben a szoftver fonalgombák növekedésének elemzésével foglalkozó részét mutatjuk be. Kitérünk a megvilágításból eredő problémák megoldására, melyet frekvenciatérbeli szűréssel oldottunk meg, és bemutatjuk a fonalgombák területének meghatározására fejlesztett algoritmust, mely jól ismert képfeldolgozási módszerek sorozataként áll elő.

Kulcsszavak: képfeldolgozás, time-lapse mikroszkópia, fonalgomba

Abstract: The purpose of this paper is to present a method for processing time-lapse microscopy images recorded in the Laboratory of Department of Microbial Biotechnology, University Debrecen. The time-lapse imaging is a widely used method, where the frame rate (fps) of time-lapse recording is typically lower than the usual fps of movie. Nowadays, the time-lapse microscopes are increasingly being used for long-time monitoring cultured cell populations. This method provides large, but manageable amounts of image data. It gives an opportunity to examine the temporal behaviour of cell populations.

Our goal is to develop a C++ software, providing an effective analysis of records made in the lab. In this paper we introduce the software module which is designed for analysis of growth of *Candida albicans* strains. We discuss a solution of the problems deriving from the poor lighting condition and we introduce our algorithm based on a sequence of well-known image processing operators.

Keywords: image processing, time-lapse microscopy, yeast

1. Bevezetés

A time-lapse eljárás nagyszerűen alkalmazható olyan hosszabb távú megfigyeléseknél, amikor a változás sebessége lehetővé teszi, hogy viszonylag nagy időközönként készítsünk képet a vizsgálat tárgyáról. A sejtbiológia területén elsősorban a sejttenyészetek megfigyelésére alkalmazzák a módszert. Nagy előnye, hogy a felvételek elemzésével objektív módon írható le a sejttenyészetek viselkedése, és egy-egy szer sejttenyészetre gyakorolt hatása.

A képek elemzése gyakran manuálisan, vagy félautomatikus módon történik. Előbbi rendkívül időigényes, utóbbi megoldás hátránya pedig az, hogy az eredmény függhet a felvételeket elemző személytől, bár ez a protokoll rögzítésével kiküszöbölhető.

Jelen cikkben egy olyan fejlesztést mutatunk be, amely az emberi szervezetben leggyakrabban előforduló gomba, a *Candida albicans* növekedésének automatikus vizsgálatát teszi lehetővé.

A következő fejezetben röviden ismertetjük a kapcsolódó irodalmat, majd a felvételek készítésének körülményeit és az automatikus eljárás lépéseit mutatjuk be.

2. Kapcsolódó irodalom

A *Candida albicans* sejtek viselkedése hosszú idő óta vizsgálat tárgyát képezi. A sejtfaalak növekedésének megfigyelésére már az 1980-as években is használtak time-lapse felvételeket [10], azonban akkoriban csak néhány egyedi sejt vizsgálatáról volt szó. Manapság jellemző a fluoreszcens festés használata, mely megkönnyíti az automatikus feldolgozást. Megemlíthetjük például Mech és társai munkáját [4], akik 6 órás fertőzésről készítettek felvételeket bizonyos időközönként, legalább 100 véletlenszerűen kiválasztott *Candida albicans* sejtről. A szerzők a felvételek elemzését az általuk korábban fejlesztett szoftverrel (Definiens Developer XD [3]) végezték. A vad típusú és nem vad típusú gombák megkülönböztetésére differenciál festést alkalmaztak és a hifák (fonalak) növekedés elemzésére matematikai modellt is adtak. A módszer természetesen kihasználja a fluoreszcens festés előnyeit, ezért az általunk használt felvételek elemzésére nem alkalmazható.

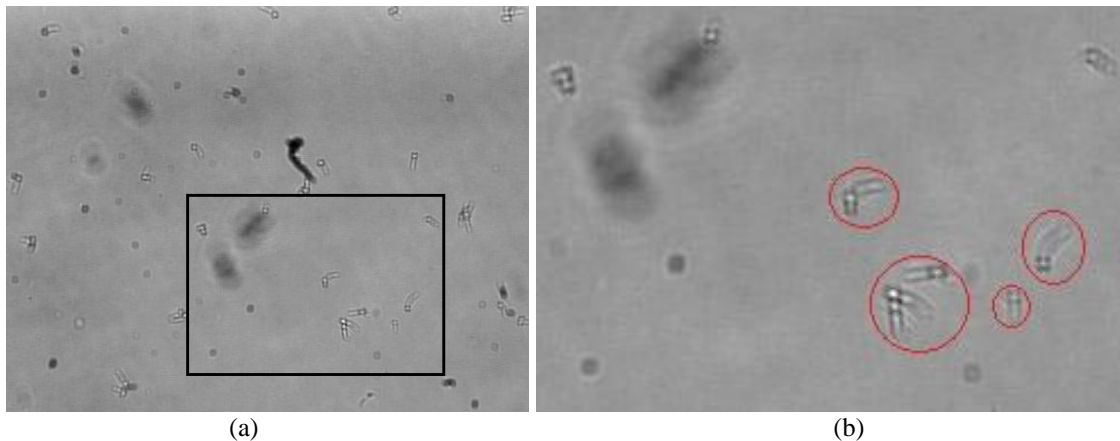
A DE Mikrobiális Biotechnológiai tanszékének laborjában jelenleg az ImageJ [8] szoftverrel történnek az elemzések a Nagy és társai által megadott módszer alapján [5]. A precízen leírt lépések, biztosítják, hogy a módszer reprodukálható legyen más laborokban is, viszont a lépések végrehajtása után manuálisan kell kiválasztani azt a részt, ahol a gombák területe stabilan mérhető.

3. Anyag és módszertan

A sejttenyésztés módja és a felvételek készítésének részletesebb leírása megtalálható az [5]-ben, ebben a fejezetben csak a fontosabb tudnivalókat és azokat az információkat emeljük ki, melyek alapvetően befolyásolják a képek minőségét.

A felvételek DE Mikrobiális Biotechnológiai Tanszékének laborjában kialakított CO₂ inkubátorba épített számítógéppel vezérelhető videó mikroszkóppal készültek vad típusú és KO mutáns *Candida albicans* gombákról, melyekből egy-egy gén ki lett ütve, hogy

elemezhető legyen a növekedés dinamikája a vad típusú egyedekhez képest. A képeket egy átalakított webkamera szolgáltatta, melyből a szűrő és az objektív el lett távolítva. Helyette egy Carl Zeiss akromatikus objektív ($\times 10$: 0.25 NA) látja el a feladatot. Annak érdekében, hogy a sejteket ne érje túl nagy fényterhelés, a 940nm-es hullámhosszú megvilágítás csak a képek készítésekor aktív (konkrét esetben 3 percenként), majd a scan rendszer egymás után több felvételt készít a tenyészetéről közeli infravörös tartományban, és az ezekből készített átlagképet menti le. Ennek hátránya, hogy a felvételek kissé elmosódottak, (1. ábra) és a megvilágítást sem lehet ideálisnak nevezni (1. ábra), melyeket néha a tápfolyadék hullámzása is tetézt.



1. ábra: Candida albicans fonalagombáról készült felvétel (a) és a megjelölt rész nagyított változata (b)

Az általunk használt képsorozat 161 darab 1600×1200 -as felbontású képet tartalmaz, PNG formátumban.

4. Képelemzés

A felvételek elemzéséhez C++ nyelvű programot fejlesztettünk OpenCV [2] felhasználásával. Az algoritmus fontosabb lépéseit a 2. ábra mutatja.

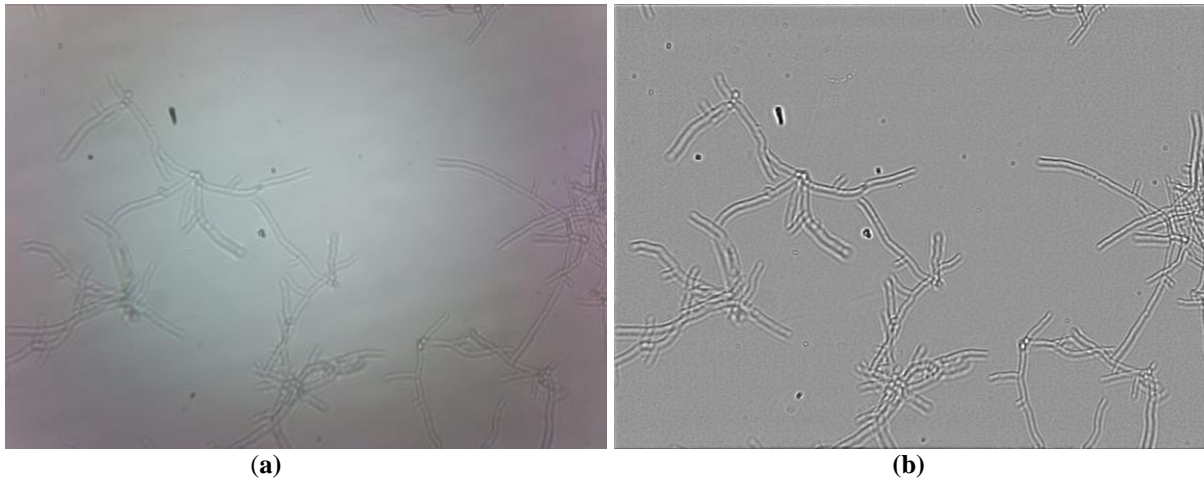


2. ábra: A képelemzéshez fejlesztett program főbb lépései

Előfeldolgozás

A feldolgozás kezdetén az 24 bites, RGB képeket 8 bites szürkeárnyaltos képé konvertáljuk. A 3. fejezetben már említett átlagolási és „bulb-effect” megvilágítási problémát frekvenciatérbeli szűréssel, sávszűréssel [9] szokás megoldani. Egy megfelelően megadott

sávszűrő képes eltávolítani mind a hullámok okozta, mind a megvilágításból eredő zajokat (3. ábra).

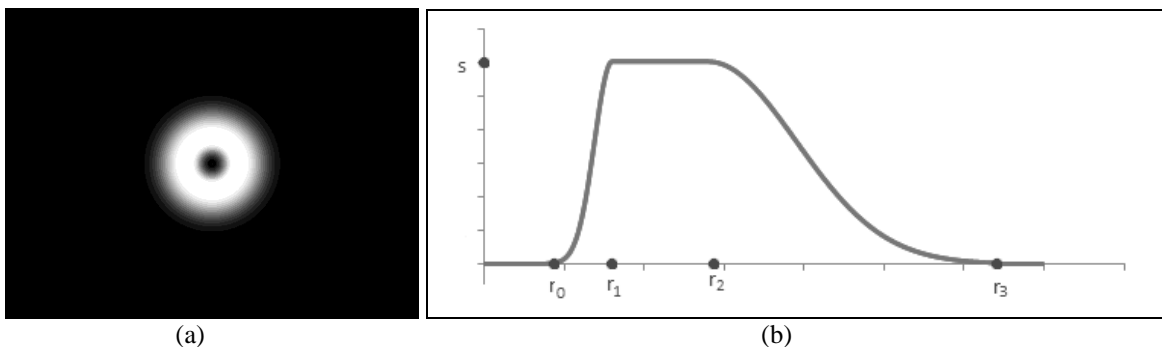


3. ábra: *Candida albicans* fonalagomba felvétele sávszűrés előtt (a) és szűrés után (b)

A szűrő létrehozásához egy egydimenziós aszimmetrikus Gauss függvény szolgáltatja az alapot [6]. A kétdimenziós szűrő képlete az alábbi:

$$\text{maszk}(p) = \begin{cases} s \cdot \exp\left(-\frac{(r_1-d)^2}{2r_1^2\sigma^2}\right) & \text{,ha } r_0 < d < r_1 \\ s & \text{,ha } r_1 \leq d \leq r_2 \\ s \cdot \exp\left(-\frac{(r_2-d)^2}{2\sigma^2}\right) & \text{,ha } r_2 < d < r_3 \\ 0 & \text{,egyébként,} \end{cases}$$

ahol $s = \frac{2}{\sqrt{2\pi} \sqrt{\sigma^2(r_i+1)}}$, d a p pont és a maszk középpontjának távolsága, $r_0 < r_1 < r_2 < r_3$ és r_i a „gyűrű” formáját meghatározó paraméterek. Az r_1 és r_2 paraméterek között egy konstans részt illesztünk az aszimmetrikus Gauss függvény két fele közé. A σ paraméter segítségével pedig a szokásos módon szabályozhatjuk a Gauss függvény csúcosságát, az r_i pedig az aszimmetriáért felel. Az egyes paraméterek szerepének megértéséhez tekintsük a szűrő keresztmetszeti képének jobb oldalát (4.b ábra).



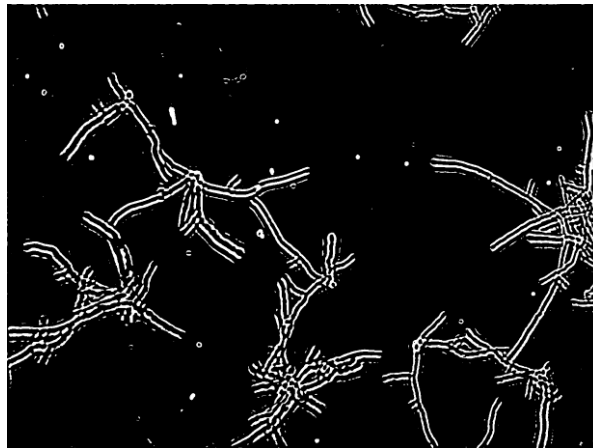
4. ábra: A szűréshez használt maszk (a) és a szűrő keresztmetszeti képének jobb oldala (b)

A háttér eltávolítása

Minden sorozat elején szerepelnek olyan képek, melyen csak a sejtektől mentes háttér látható, azonban tapasztalatunk alapján 20-30 képkocka után már deflicker használata mellett is zajos marad háttérlevonás utáni kép. Az előzetes kísérletek alapján a háttérrel mindig az aktuális képből alakítjuk ki, egy nagyméretű (41×41) mediánszűrő segítségével, melynek a feladata a gombák és egyéb foltok eltávolítása a képről. Az előteret az aktuális képből egy kisméretű mediánszűrővel alakítjuk ki (7×7), ennek szerepe csak a szokásos zajszűrés. Ezek után a háttérképből kivonjuk az előtér képet, az eredményképen tehát a magas intenzitású részek jelentik a gombák körvonalát.

Küszöbölés

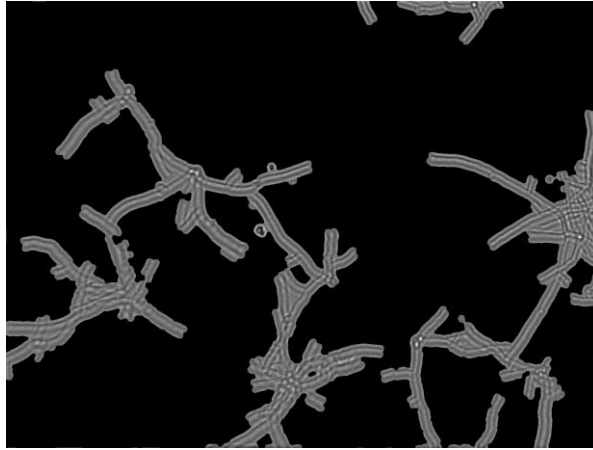
A gomba körvonalak meghatározásához az előzőleg nyert különbségképre küszöbölést alkalmazunk. A vágási határt az Otsu [7] módszerrel határozzuk meg, mely úgy alakítja ki az előtér és a háttér osztályokat, hogy az abba eső intenzitásértékek szórása minimális legyen. A határ meghatározásánál csak a pozitív értékeket vesszük figyelembe. A különbségképre alkalmazott vágás eredménye az 5. ábrán látható.



5. ábra: Az Otsu küszöbölés eredménye a különbségképre. Fehérrel a gombák körvonalai.

A gomba maszkjának meghatározása

A gomba terület meghatározásához dilatációt hajtunk végre a küszöbölt képre egy igen nagyméretű struktúra elemmel (25×25), melynek eredményeként a gomba körvonalai összemosódnak, majd az apró lyukakat feltöltjük, így a következő lépésben alkalmazott erózió (11×11 -es kernel) csak a külső részről törölje le a felesleges területet. Végül eltávolítjuk az apró objektumokat a képről. A maszk alatti terület képe a 6. ábrán látható.



6. ábra: A gombaként detektált terület az eredeti képen

Növekedés vizsgálata

A képen található gombák növekedését akár a maszk alatti teljes területtel is jellemezhetjük. Ennek a módszernek az előnye, hogy a korai összenövés ellenére is leírható a növekedés üteme. Az összenövés megjelenése előtt egyedileg is nézhetjük a gombák növekedését. Ehhez az aktuális és az azt megelőző képkockán detektált összefüggő komponenseket kell egymáshoz rendelnünk. Annak érdekében, hogy az összenövés tényét is automatikusan megállapíthassuk, mindig a korábbi képen lévő komponensekhez keressük a párt, távolság és terület, majd ha ezek alapján nem találtunk párt, átfedés vizsgálat alapján.

5. Összefoglalás

A cikkben bemutattuk a fonalas gombákat ábrázoló felvételek készítésének körülményeit, és a gomba területének meghatározásához fejlesztett módszert. Az algoritmust C++ nyelven implementáltuk. Ismertettük a gomba területének meghatározásához fejlesztett algoritmust, melynek a megvilágítási problémákat kezelő része várhatóan a laborban, más sejtípusokról készült felvételek esetén is használható lesz.

Az elért eredmény mindenképpen előrelépést jelent a kézi és a félautomatikus elemzéshez képest, azonban a hifák legvégének pontos meghatározásával kapcsolatban maradt egy kis hiányérzetünk. A nehézséget az okozza, hogy ezek a képen szabad szemmel is rosszul láthatóak. Úgy gondoljuk, hogy a probléma legkönnyebben a felvételek készítése során orvosolható majd.

6. Köszönetnyilvánítás

A publikáció elkészülését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

A kutatás a TÁMOP 4.2.4.A/2-11-1-2012-0001 azonosító számú Nemzeti Kiválóság Program – Hazai hallgatói, illetve kutatói személyi támogatást biztosító rendszer kidolgozása és működtetése konvergencia program című kiemelt projekt keretében zajlott. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] M. Baatz, N. Arini., A. Schäape, G. Binnig, B. Linssen: Object-oriented image analysis for high content screening: Detailed quantification of cells and sub cellular structures with the Cellenger software, *Cytometry Part A* **69** (2006), 652–658.
- [2] G. Bradski: The OpenCV Library, *Dr. Dobb's Journal of Software Tools* (2000).
- [3] F. Mech, A. Thywissen, R. Guthke, A. A. Brakhage, M. T. Figge: Automated image analysis of the host-pathogen interaction between phagocytes and *Aspergillus fumigatus*, *PLoS One* **6** (2011).
- [4] F. Mech, D. Wilson, T. Lehnert, B. Hube, M. T. Figge: Epithelial invasion outcompetes hypha development during *Candida albicans* infection as revealed by an image-based systems biology approach, *Cytometry Part A* **85** (2014), 126–139.
- [5] G. Nagy, G. W. Hennig, K. Petrenyi, L. Kovacs, I. Pocsi, V. Dombradi, G. Banfalvi: Time-lapse video microscopy and image analysis of adherence and growth patterns of *Candida albicans* strains, *Applied Microbiology and Biotechnology* **98** (2014), 5185–5194.
- [6] T. Kato, S. Omachi, H. Aso: Asymmetric Gaussian and Its Application to Pattern Recognition, *Structural, Syntactic, and Statistical Pattern Recognition* (2002), 405–413.
- [7] N. Otsu: A threshold selection method from gray-level histograms, *IEEE Transactions on Systems, Man, and Cybernetics* **9** (1979), 62–66.
- [8] W. S. Rasband, ImageJ, U. S. National Institutes of Health, Bethesda, Maryland, USA, <http://imagej.nih.gov/ij/>, 1997–2014.
- [9] M. Sonka, V. Hlavac, R. Boyle: Image Processing, Analysis and Machine Vision (1993), 1–555.
- [10] M. Staebell and D. R. Soll: Temporal and spatial differences in cell wall expansion during bud and mycelium formation in *Candida albicans*, *Journal of general microbiology* **131(6)** (1985): 1467–1480.

Tudás disszemináció és innovatív információs technológiák a mezőgazdaságban

Knowledge dissemination and innovative information technologies in agriculture

Szilágyi Róbert^a, Herdon Miklós^b, Lengyel Péter^c, Rózsa Tünde^d

^{a b c d}Debreceni Egyetem Gazdaságtudományi Kar

szilagyi@agr.unideb.hu

herdon@agr.unideb.hu

lengyel@agr.unideb.hu

tunde.rozsa@gmail.com

Absztrakt: A felsőoktatási és kutatási intézmények, valamint a különböző szakmai szervezetek nagyon fontos szerepet töltenek be az agrárgazdasági IKT kutatásokban. Az agrár- élelmiszeripari ágazat számára fontos, hogy ezen intézmények és szervezetek a kutatási eredmények disszeminációjával, az ágazat innovációs képességének növelésében is tevékenyen részt vegyenek. A Magyar Agrárinformatikai Szövetség (MAGISZ) az elmúlt 16 évben a felsőoktatási intézményekkel együttműködve számos kezdeményezéssel és projekttel járult hozzá az ágazati K+F valamint az agrárinformatikai oktatás fejlődésének és fejlesztésének támogatásához. Korábbi TÁMOP projektnek köszönhetően létrehozásra került egy tudományos portál, a nemzetközi szerkesztőbizottsággal és tanácsadó testülettel elindított Journal of Agricultural Informatics folyóiratnak 8 száma került eddig kiadásra, számos tudományos konferencia került megrendezésre, a fiatal kutatók, diákok részvételével. A hallgatói munka támogatására szolgál az évente kiírásra kerülő diplomadolgozat pályázat. A MAGISZ számos műhelymunka rendezvény és konferencia szervezése mellett nagy hangsúlyt helyez az oktatásban való közreműködésre. A 2014-ben elindult szakigazgatás-szervező és informatikus agrármérnöki mester szak kialakításában is fontos szerepet játszott.

Kulcsszavak: disszemináció, mezőgazdaság, információs technológiák

Abstract: Higher education and research institutions, and professional organizations play a very important role in the agricultural economy of ICT research. Important for the agri-food sector to disseminate the results of research institutions, organizations, increasing the innovative capacity of the sector to be actively involved in this. The Hungarian Association of Agricultural Informatics (HAAI) for the past 16 years in collaboration with higher education institutions and a number of initiatives to support the project contributed to the R & D sector and the agricultural development of IT education and development. Thanks to the former TÁMOP project has been created a scientific portal, the international editorial advisory board and committee launched the Journal of Agricultural Informatics magazine has published eight issues, a number of scientific conferences were held for young researchers and students in attendance. Support for the students' work to be announced in a year dissertation proposals. The event HAAI numerous workshops and conferences with a high emphasis on education to contribute. In the new Agricultural Engineer in Agricultural Public Administration and Agri-informatics MSc degree program the HAAI has been played an important role in the development.

Keywords: dissemination, agriculture, information technologies

1. Bevezetés

A kutatási eredmények disszeminációja fontos a kutatók és a felhasználók számára egyaránt. Az információ a hatékonyság növelésének fontos eszköze [4] [5]. Információra van szüksége a kutatónak, aki felhasználva az eddigi eredményeket, új eljárásokat módszereket fejleszt ki, szüksége van a gazdálkodónak, aki a gyakorlatban igyekszik beépíteni a kutatók eddigi eredményeit, szüksége van az igazgatási szerveknek, hogy olyan szabályozásokat alakítsanak ki, amelyek az ágazat előrelépését segítik, szüksége van a bankárnak, aki megfelelő anyagi háttérrel képes biztosítani a kutató, vagy a gazdálkodó számára, szüksége van a tanácsadónak, aki tevékenységével szintén hozzájárul az agrár-ágazat sikerességéhez [1]. A megfelelő időben, a megfelelő formában a megfelelő információ megtalálása nem egyszerű feladat. Ezért különösen fontos a specifikus portálok működése [3], hiszen az Internet adta lehetőségek kihasználásával rövid időn belül megtalálhatók a megfelelő információ források. A Magyar Agrárinformatikai Szövetség egy olyan Egyesület, melynek magánszemélyek és szervezetek a tagjai, akik egyaránt érdekeltek az agrárinformatikai terület fejlődésében. A szervezet azzal a céllal jött létre, hogy képviselje azoknak az érdekeit, akik az agrár-ágazatban alkalmazni, használni akarják az információ technológia korszerű eszközrendszerét. Az Egyesület alapszabályában az alábbi célokat fogalmazta meg:

- Az agrárinformatika hazai szakembereinek meghatározó szervezeteként járjon hozzá az információs társadalom, azon belül agrárgazdasági kibontakozásához. Emellett ápolja a magyar agrárinformatikai, számítástechnikai hagyományokat.
- Legyen a szakmai élet aktív résztvevője: vegyen részt szakmai kutatásokban, alkalmazás-fejlesztésekben, szakvélemények készítésében; a szakmai közélet fórumainak kialakításával és rendezvények szervezésével segítse elő az agrárinformatikai szakemberek tájékozódását, valamint ösztönözze a szakmai életben való konstruktív részvételt: írjon ki pályázatokat, alapítson, illetve ítéljen oda díjakat.
- Az informatikai tájékozottság növelése érdekében vállaljon kezdeményező szerepet az informatikai ismeretek fejlesztésében, az Internet-kultúra terjesztésében. Szenteljen figyelmet a tehetséggondozásnak: rendezzen, illetve támogasson országos versenyeket az agrárinformatika területén.
- Ápoljon hatékony együttműködésen alapuló kapcsolatokat az informatika területén a hazai állami és vállalkozói szférában működő, valamint nemzetközi szervezetekkel.
- Nyújtson szolgáltatásokat egyéni és jogi tagjainak, képviselje a tagok érdekeit.
- A szakmai feladatokkal és a Szövetség szolgáltatásaira való társadalmi igény növekedésével arányban biztosítsa a taglétszám növekedésének feltételeit, és fordítson különös gondot a fiatalok szakmai életbe történő bevonására.

Figyelembe véve a megfelelő információ agglomerációs igényét, a szervezet 2009-ben elindította a „Innovatív információtechnológiák agrárgazdasági kutatási, fejlesztési, alkalmazási eredmények disszeminációja” című projektjét.

A projekt az információterjesztés különböző formáinak segítségével tette lehetővé a felsőoktatásban keletkezett új és újszerű eredmények megismerését. A tudományos portál célja azon információk és információforrások összegyűjtése, melyek az agrárinformatika területén érdekesek és értékesek a szakemberek számára. Az elektronikus folyóirat az új eredmények publikálásának teremtette meg a lehetőségét. A tanulmánykötetben kiforrott esettanulmányok publikus megjelenését biztosítottunk. A tudományos konferenciák a fejlődés elengedhetetlen színterei, a személyes vita a szakemberek között mindig jótékony hatással van az eredményekre és azok korrekációjára. A fiatal kutatók fejlődése a jövő zálogát jelenti,

és biztosítja a tudományterület folytonosságát, ezért az egyesület kiemelt figyelmet fordított a fiatalok támogatására.

2. A Magyar Agrárinformatikai Szövetség tevékenységének társadalmi és környezeti háttere

Az Európai Unió életében a mezőgazdasági támogatások kiemelt szerepet játszanak. A közösség költségvetésének jelentős részét jelenleg is a mezőgazdaság és a vidékfejlesztés területe teszi ki. A szervezet bővítésével és a belső változások szükségszerű előtérbe kerülésével, az agrárpolitika is jelentős változáson ment keresztül, sőt, ez a folyamat, részben a globális kihívásokra reagálva, részben a nemzetközi válságra tekintettel továbbra is tart. Mindeközben az EU a világ többi részével szemben is meg szeretné őrizni, illetve szeretné javítani pozícióit, az információs korszak kihívásainak is meg kell felelnie, ha állni akarja a versenyt. Egyértelműen látszik, hogy a mezőgazdasággal kapcsolatos további diskurzusoknak és kezdeményezéseknek a jövőben is kiemelten kell számolniuk az újabb Információs és Kommunikációs Technológiákkal (IKT). A fenntartható, multifunkcionális, globális környezetben létező mezőgazdaság sikerességéhez elengedhetetlenek a modern IKT-eszközök alkalmazása.

A gazdálkodás számos összetevőjéhez kapcsolódhatnak ezek az eszközök, szakmai információk könnyebb elérését tehetik könnyebbé (az időjárástól a szaktanácsadásig), a termékek értékesítésében, új piacok felfedezésében, vagy éppen a támogatások bonyolult rendszerében történő eligazodást és a támogatások kérelmezésének bonyolult folyamatát egyszerűsíthetik. Ez utóbbi kérdéskör vezet el két, az Európai Unióban kiemelten kezelt terület összefonódásához. A mezőgazdasági politika mindig is kiemelt szerepet játszott a Közösség életében (noha újabb és újabb, egyre komolyabb korrekcióra szorul), míg az újabban információs társadalomnak nevezett jelenségkör, az ezzel kapcsolatban álló versenyképességi kérdések egyik alappillére, az elektronikus közigazgatás, a közsféra modernizálása is jelentősen felértékelődött az utóbbi évek során. Az adatok alapján a bővülés üteméből arra következtethetünk, hogy a magyarországi információs társadalom fejlődése (az infrastruktúra fejlesztése) révén, a terjedési folyamatok felgyorsultak (11 százalékponttal nőtt az számítógéppel felszerelt háztartások aránya, és 14 százalékponttal az interneteléréssel rendelkező otthonoké) – és ez jótékony hatással van az e-közigazgatás fejlődésére is.

A vonatkozó magyar kutatásokban folyamatosan megállapításra kerül, hogy hazánkban igen komoly digitális megosztottságot észlelhetünk az információs technológiákhoz történő hozzáférés tekintetében. Jelentős különbségek mutatkoznak a társadalomban településméret és iskolai végzettség szintjén is. Ez azért fontos, mert a magyar gazdátársadalom demográfiai jellemzőit ismerve könnyen beazonosítható, hogy a vidéken élő, az átlagosnál alacsonyabb végzettséggel rendelkező rétegről van szó, amely ezen tulajdonságaiból kifolyólag nem ideális célcsoport egy olyan sikeres e-közigazgatási projekt megvalósításához, mint az agrár-támogatások elektronikus kérelmezése. Település-méret alapján több mint húsz százalék különbség van az internethasználók arányában a községek és a főváros között, míg az előbbieken a lakosság alig harmada használja a világhálót, a fővárosban ez az arány jóval meghaladja a lakosság felét.

Az internet használatával kapcsolatos készségeknél az e-mail küldése, fogadása a gazdák háromnegyede számára nem jelent problémát. Fontos tudni, hogy az internethasználó gazdák komoly információforrásként tekintenek a világhálóra, ezt bizonyítja, hogy közülük több mint

80 százalék keres munkájával kapcsolatos információkat a neten. Programokat letölteni már jóval kevesebben szoktak, az internetező gazdák kevesebb, mint fele, míg bankügyleteket negyedük-ötödük végez el elektronikusan. Hazánkban is tapasztalható tehát az a nemzetközi trend, mely szerint a gazdálkodók az egyszerűbb, üzenetváltással és információkereséssel kapcsolatos eljárásokra használják leginkább a világhálót.

A magyar gazdaságszervelet számos problémával küzd, melyek jelentős része az információ hiányából fakad [1]. Ez az információs deficit a modern információs- és kommunikációs technológiákkal komoly mértékben csökkenthető, ám ehhez a tudatosság felkeltésére, valamint a készségek növelésére is szükség van. Az előbbi talán a legnehezebb feladat, mely egy sikeres e-közigazgatási projekttel lendületet kaphat, hiszen a kérelmek pontos kitöltése, azok gyors elbírálása és ezáltal a támogatások időben történő kifizetése neuralgikus pont. Egy jól működő rendszer ráirányíthatja a gazdák figyelmét az életüket megkönnyítő alkalmazások használatára.

A MAGISZ által elérhető célcsoportokat két részre oszthatjuk:

- Egyik célcsoport azon felsőoktatási intézmények szervezeti egységei, amelyek a projekt szempontjából releváns kutatási témákkal foglalkoznak és a szakmai szervezet jogi tagjai, valamint a felsőoktatásban dolgozó oktató/kutató személyek és doktori iskolák hallgatói [1]. A Magyar Agrárinformatikai Szövetség személyi és jogi tagjainak meghatározó többsége azon felsőoktatási intézmények köréből kerül ki, amelyek felsőoktatási alapszak, egyetemi szak, doktori iskola keretében a szakterület oktatásával foglalkoznak és kutatási témákat művelnek. Ezen intézmények a Debreceni Egyetem, Budapesti Corvinus Egyetem, Szegedi Tudományegyetem, Károly Róbert főiskola, Pannon Egyetem, Szent István Egyetem, Nyugat-Magyarországi Egyetem.
- A másik célcsoport az agrárgazdaság kormányzati, szakigazgatási és vállalati szereplői, akik számára a projekt révén a kutatási eredmények nagyobb hatékonysággal eljuttathatók, hasznosíthatók.

Az ágazatirányítás, a köz- és szakigazgatás fontos feladata az információs rendszerek fejlesztése, nagyobb hatékonyságú működtetése, az e-szolgáltatások nyújtása saját szervezeten belüli felhasználóik, valamint külső partnereik, ügyfeleik, a gazdasági szereplők, a gazdálkodók számára. E feladatban résztvevők a rendszerek és szolgáltatások fejlesztése, működtetése során hasznosítani tudják a kutatók eredményeit, tanulmányait. A gazdálkodók számára a fenti rendszerek használata, saját gazdasági tevékenységük fejlesztése, hatékonyabbá tétele, a szükséges információk megszerzése rendkívül fontos, amelyet az portál információs szolgáltatása támogat.

2.1. Tudományos információs portál

A tudományos információs portál (<http://tamop.magisz.org/>) lehetőséget biztosít a felsőoktatásban keletkezett kutatási-fejlesztési és innovációs eredmények gazdasági szektor felé történő elterjesztésére, megismertetésére. Tagjai és a szakterületen dolgozó oktatók, kutatók, szakemberek számára biztosítja a szakmai életben való aktív részvételt. Szolgáltatásokat nyújt az egyéni és jogi tagok, valamint az érdeklődők részére.

A portál tervezésnél és fejlesztésnél az alábbi követelményekre különös figyelmet fordítottunk:

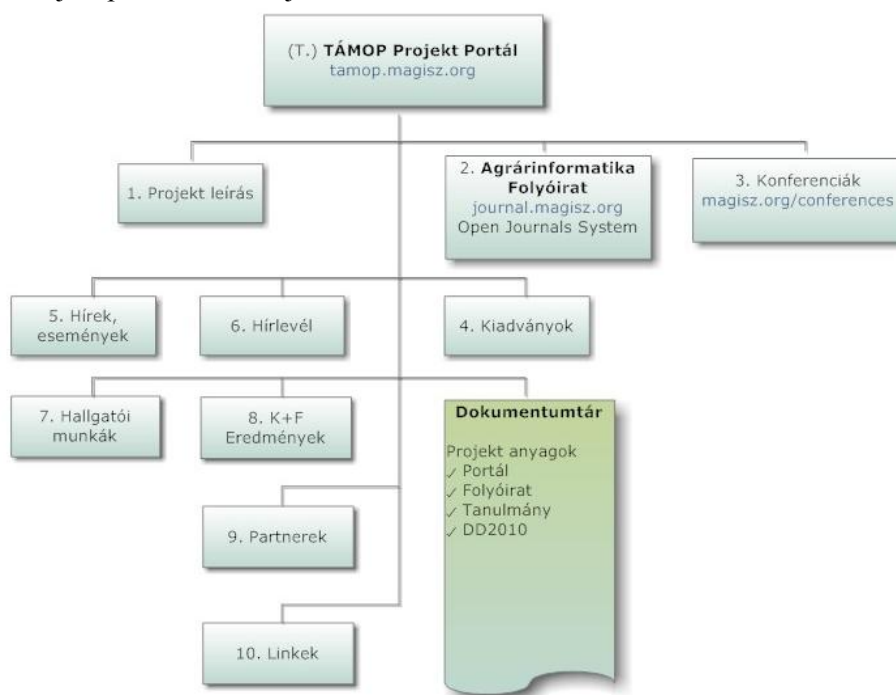
- interoperabilitás,
- skálázhatóság,
- modularitás,
- könnyű használhatóság,

- linkek biztosítása a közösségi hálózatokhoz,
- Web 2.0 alkalmazások (Facebook, RSS).

A portál főmenüje az alábbi kategóriákat tartalmazza:

- Projekt leírás, amely tartalmazza az alapvető célkitűzéseket.
- Agrárinformatikai Folyóirat: átirányítja az érdeklődőt a tudományos folyóirat oldalaira,
- Konferenciák: a szakterülethez kapcsolódó saját és társzervezetek által szervezett konferenciák, szakember találkozók felsorolása található, a rendezvény oldalra mutató linkekkel.
- Kiadványok: a szakterületen megjelenő kiadványok felsorolása és azokra vonatkozó információk, linkek, illetve teljes kötetek található a bejegyzés alatt. Ilyenek például az agrárinformatikai tanulmány kötetek, konferencia kiadványok, egyéni kiadványok, szakkönyvek.
- Hírek, események: itt kerülnek publikálásra a legújabb hírek, események.
- Hírlevelek: itt olvashatók a meghatározott időközönként kiküldött hírlevelek.
- Üzenőfal: lehetőséget biztosít a szerkesztők fele történő üzenetküldésre.
- Hallgatói tudományos munkák: a meghirdetett hallgatói pályázatra beérkezett tudományos művek felsorolását tartalmazza.
- Partnerek: az együttműködő szervezetek felsorolását biztosítja.
- Dokumentumtár

Az 1. ábra mutatja a portál struktúráját.



1. ábra. A TAMOP portál struktúrája

A tudományos portál működése nyomán vélhetően javul a szakterület információ ellátottsága. Az agárgazdaságban alkalmazott információ technológiákról, alkalmazási módszerekről, esettanulmányokról, kutatási eredményekről szélesebb kör számára nyílik lehetőség ismeretszerzésre, információ eléréshez. Ilyen fontosabb területek: e-kormányzati, és e-szakigazgatási rendszerekről információk, termék azonosítás és nyomon követés információtechnológiai és rendszerei, e-kereskedelem, vállalati információs rendszerek, statisztikai adatbázisok, klímaváltozás és hatásai, precíziós gazdálkodás, mobil internet stb. Ez azért fontos, mert az előző tématerületekről igen szegényes az elérhető tartalmak köre. A

portál egyenletes látogatottsággal rendelkezik, a tartalomfejlesztés és a közösségi médiákon való megjelenés folyamatos.

2.2. Journal of Agricultural Informatics

Magyarországon a szakterülettel kapcsolatos tudományos folyóiratok száma kevés, az agrárinformatika témakörben pedig nem is volt. Ezért tekintjük a létrehozott folyóiratot hiánypótló jellegűnek. A folyóirat közzéteszi a mezőgazdaságban történő fejlett információs technológiák alkalmazásához kapcsolódó kutatási és az alkalmazási eredményeket. A eredmények közzétételével a folyóirat növeli a tudományos ismeretek terjesztését és fejleszti az innovációs folyamatokat. Magyarországon az utóbbi években, a makró-rendszerekben történő fejlesztések terén érték el jelentős sikereket, azonban a vállalkozások, gazdálkodók jelentős lemaradásban vannak az IKT és a rá épülő alkalmazások használatában. A cél, hogy a folyóirat kitöltse ezt az űrt azáltal, hogy a kutatási eredmények és a legjobb gyakorlatok a vállalatok és a gazdaságok számára is elérhető legyenek. Számos kutatási intézmény, egyetemi doktori iskola foglalkozik az agrárinformatikához kapcsolódó tématerülettel. A folyóirat nemzetközi szerkesztőbizottsága garanciát jelent arra, hogy minőségi publikációk jelenjenek meg. Hazai viszonylatban az agrárinformatikai területen elsősorban a konferencia előadások és kiadványok jelentették és jelentik a publikálási lehetőséget. A konferencia kiadványok nagyon fontosak és nagyon hasznosak, viszont hiányzik egy megfelelő kontroll, egy lektorálási folyamat, amelynek hiánya hátráltatja a fiatal szakemberek minőségi publikálását. E folyóirat a tudományos eredmények közzététele mellett javítaná a külföldi publikációk, nemzetközi folyóiratokban való publikációs lehetőségeket. E publikálási forma elsősorban az új tudományos eredmények közzétételéhez, disszeminációjához, járul hozzá, amely széles felhasználói kör számára elérhető.

Az e-folyóiratnak 8 száma jelent meg eddig, melyben helyet kaptak angol és magyar nyelvű publikációk valamint az erre érdemes hallgatói munkák is. 2013-tól a folyóirat szerkesztőbizottsága átalakult. A folyóirat szerkesztőségében nagyrészt képviselnek az European Federation for Information Technology in Agriculture, Food and the Environment (EFITA) nemzeti tagjai is. A folyóirat tematikus különszámokat is jelentett már meg. A folyóirat honlapját tartalmazza a 2. ábra.

The screenshot shows the website for the Journal of Agricultural Informatics (JAI). The header includes the journal's name in English and Hungarian, its publisher (HAAI), and its ISSN (2061-862X). It also mentions support from the European Federation for Information Technology in Agriculture, Food and the Environment (EFITA). The navigation menu includes Home, About, Log In, Register, Search, Current, Archives, Támpo, HAAI, EFITA, and CALLS. The main content area displays the current issue (Vol 4, No 2 (2013)), a table of contents, and back issues. A sidebar on the left contains user login fields, information links, a search bar, and a call for special issues. The footer lists the advisory board members and the editor in chief.

2. ábra. A TÁMOP portál struktúrája

2.3. Tanulmánykötetek megjelentetése

Az agrárinformatikai tanulmányok a témakörben elért kutatási eredmények publikálásával segíti az informatikai tájékozottság növelését. Segíti az agrárinformatikai szakemberek tájékozódását. A szakterületen tevékenykedő oktatók/kutatók számára kutatási eredményeik megjelentetésére nagyon korlátozott lehetőségek állnak rendelkezésre. Mivel általában a publikálási költségforrások meglehetősen korlátozottak, az angol nyelvű források nem tartalmaznak hazai környezetre vonatkozó kutatási eredményeket, ezért az oktatók/kutatók, PhD kutatások eredményeinek közzététele, publikálása tanulmánykötetekben hasznos lehetőséget biztosít. A tanulmánykötetből négy került kiadásra.

2.4. Konferenciák, rendezvények

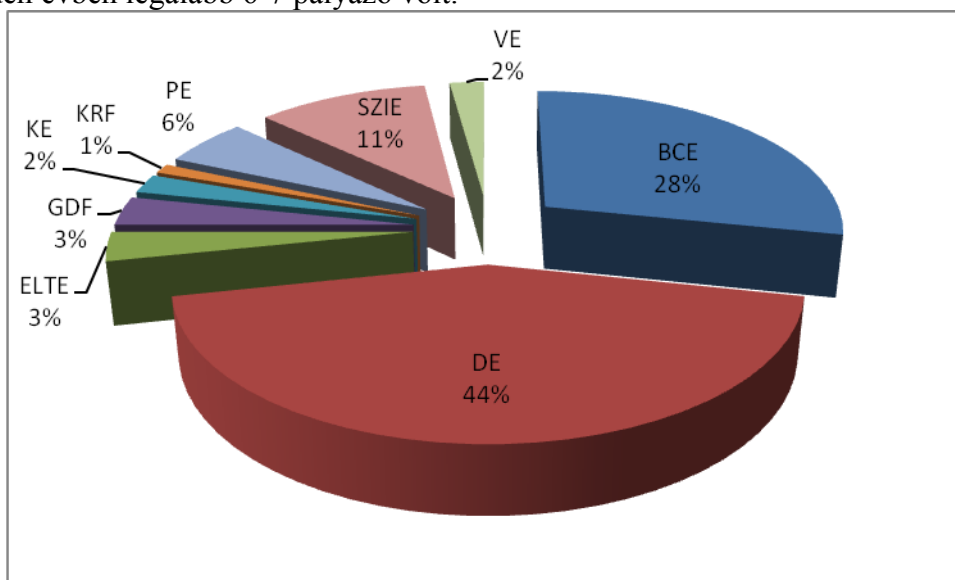
A tudományos nemzetközi konferenciák, hazai rendezvények szervezésével fejleszti, elősegíti a hazai non-profit és vállalkozói szféra szakembereinek tapasztalatcseréjét, nemzetközi kapcsolatok fejlődését. A Magyar Agrárinformatikai Szövetség elmúlt 11 éves tevékenysége során rendszeresen szervezett hazai konferenciákat, valamint társszervezőként több rendezvény előkészítésében és lebonyolításában vett részt. A MAGISZ nevéhez köthetők az Agrárinformatika konferenciák, és az Agrárinformatikai Nyári Egyetem rendezvények, melyek évente megrendezésre kerülnek. A MAGISZ számos műhelymunka rendezvényt is tartott. Ezen rendezvények résztvevői nagyrészt a hazai felsőoktatási intézmények oktatói, kutatói.

2.5. Fiatal kutatók és hallgatók támogatása

A hallgatók és fiatal kutatók tudományos munkájának támogatását pályázatok kiírásával, díjak alapításával és odaítélésével segíti. A tehetséggondozás támogatására országos és helyi rendezvények szervezését támogatja. A szervezet immár öt éve ír ki diploma/szakdolgozat és TDK dolgozat pályázatokat. Ma már egyre több dolgozat születik olyan témakörben, melyek az informatika agrárgazdaságbeli alkalmazások különböző területével foglalkoznak. Az egyesület tevékenységének kibővítése e területre három szempontból fontos.

- Ösztönözi a színvonalas kutatómunkák végzését.
- Lehetőséget biztosít a fiatal kutatók (PhD hallgatók) szakmai elismerésére.
- A tudományos eredmények szélesebb körű megismertetését biztosítja.

A MAGISZ által kiírt diplomadolgozat pályázatra, amelyet 2004-ben írtak ki először, eddig összesen 92 pályamunka került beadásra. Évente változó számú pályamunkát adtak be, de eddig minden évben legalább 6-7 pályázó volt.



3. ábra. A MAGISZ diplomadolgozat pályázatra beadott pályamunkák alakulása intézményenként

Az 3. ábra alapján elmondható, hogy a pályázók jellemzően a Budapesti Corvinus Egyetemről és a Debreceni Egyetemről adták be pályázataikat, de a Szent István Egyetemről is több pályamunka érkezett be. Fontos megemlíteni, hogy más felsőoktatási intézmények hallgatói munkái is beadásra kerültek.

3. Következtetések

A társadalmi és tudományos szereplők számára egyaránt fontos az innováció az agrárgazdaságban, legyen szó mezőgazdasági vállalkozókról, kutatókról, tanácsadókról, döntéshozókról, szállítókról, feldolgozóiparról, kiskereskedelem szereplőiről, vagy akár az végfelhasználókról. Az innovációs folyamatban lényeges szereplők a felsőoktatási intézmények. Ezért különösen fontos, hogy a keletkezett eredmények közkinccsé váljanak. Ebben játszik fontos szerepet a Magyar Agrárinformatikai Szövetség tevékenysége, mellyel hozzájárul az innovációs folyamatok hatékonyságának növekedéséhez. E tevékenység

azonban nem érhet véget a projekt befejezésével, hiszen továbbra is rendkívül fontos a tevékenységek folytatása, a létrehozott rendszerek működtetése. A feladatok ellátása érdekében azonban együttműködés szükséges a felsőoktatási, a kutatási, intézményi és gazdálkodási szervezet között. Ezt a hídszerepet hivatott ellátni az Egyesület a jövőben is.

4. Köszönetnyilvánítás

A korábbi projekt a TÁMOP 4.2.3-08/1-2009-0004 ("Innovatív információtechnológiák agrárgazdasági kutatási, fejlesztési, alkalmazási eredmények disszeminációja" támogatásával valósult meg.

Irodalomjegyzék

- [1] Herdon M.: Agri-informatics Curriculum and Education. Why and how we need training agri-informatics experts? Demeter Conference. European Higher Education Conference on Virtual Mobility - Information and Communication Technologies in Agriculture and Related Sciences (Video conferencing), Gent-Copenhagen-Montpellier, June 16-17,1997. DEMETER Proceedings (1997), pp. 61-69.
- [2] Kaaya J.: Role of information technology in agriculture. Proceedings of FoA conference Vol.4, (1999) 315.
<http://www.tzonline.org/pdf/roleofinformatontechnologyinagriculture.pdf>.
Letöltés:2011.07.25
- [3] Klerkx, L., Leeuwis. C.: Balancing multiple interests: Embedding innovation intermediation in the agricultural knowledge infrastructure. Technovation 28 (2008) 364–378
- [4] Li, J., Kozhikode R. K.: Developing new innovation models: Shifts in the innovation landscapes in emerging economies and implications for global R&D management. Journal of International Management, Volume 15, Issue 3, (2009), Pages 328-339
- [5] März, S., Friedrich-Nishio, M., Grupp, H.: Knowledge transfer in an innovation simulation model Technological Forecasting and Social Change, Volume 73, Issue 2, (2006), Pages 138- 152

Python kontra ArcGIS

Python vs. ArcGIS

Zichar Marianna

Debreceni Egyetem Informatikai Kar
zichar.marianna@inf.unideb.hu

Absztrakt: Az ESRI ArcGIS desktop termékei igen népszerűek a térinformatikai szoftverek körében. Az egyéni igények kódolással való megvalósításához is támogatást adó termék 10-es verziójának megjelenésével egy új korszak indult el azzal, hogy fokozatosan megszűnik, a korábban igen népszerű, VBA támogatás. E verzió telepítőanyaga már nem tartalmazza a korábban beintegrált fejlesztő környezetet, s az egyéni igények, geofeldolgozások kielégítéséhez egy új programnyelv irányába, a Python felé terelgeti a felhasználókat, fejlesztőket. A Python egy olyan általános célú programozási nyelv, melynek felhasználása rendkívül sokszínű: webes alkalmazásoktól kezdve, a grafikus felhasználói felületeken, tudományos számításokon keresztül most már a térinformatikáig is elér. Népszerűsége egyre nagyobb, mellyel párhuzamosan felhasználói közössége is egyre több tagot számlál. További előnyös tulajdonságai: nyílt forráskódú, ingyenes, folyamatosan fejlődik, könnyen tanulható, számtalan (online) oktatási anyag, példatár áll rendelkezésre. Jelen tanulmány feltárja az ArcGIS programozásához használt régi és új programnyelv használata közötti különbségeket, külön figyelmet szentelve az alkalmazhatóság körére, a technológiai jellemzőkre és nem utolsósorban az oktatásban való megjelenés tapasztalataira.

Kulcsszavak: ArcGIS, Python, geofeldolgozás

Abstract: The desktop products of ESRI are rather popular in the field of geographic information systems. These products support the customizations by enabling to develop custom codes. The appearance of version 10 indicates a new era by starting to eliminate the support of VBA, which was rather popular previously. The install set of this version does not contain the VBA integrated developing environment any more, but pushes the users and developers to use Python if customization is needed. Python is a general-purpose, high level programming language which usage ranges from creating web applications, graphical user interfaces till performing scientific computation and even to geoinformatics. Its popularity keeps increasing implying also the growth of Python fan communities. It is open source, free; its developing does not stop. It is easy to learn and wide range of online tutorials, sample codes are available. This paper explores the difference between the previous and the actual programming language used for creating custom codes for ArcGIS and also pays attention to discuss their usability, technical properties and the experiences collected in the education.

Keywords: ArcGIS, Python, geoprocessing

1. Bevezetés

A térinformatikai szoftverek piacán az ESRI ArcGIS termékcsaládja már hosszú évek óta meghatározó szerepet tölt be. Az asztali, server és mobil eszközökre fejlesztett különböző termékei lehetővé teszik a térbeli információval rendelkező adataink elemzését, megértését, vizualizációját elősegítve helyes, földrajzi elhelyezkedéssel kapcsolatos döntések meghozatalát. Az ArcGIS for Desktop több mint száz eszközt biztosít az elemzéseken és megjelenítéseken kívül különböző automatizálási feladatokhoz is. Az egyéni igények kielégítéséhez saját alkalmazást is fejleszthetünk többféle platformon. A 10-es verzióig a telepítő anyag részét képezte a VBA integrált fejlesztői környezet, mely a futó asztali alkalmazásokból közvetlenül elérhető volt. Nagyon népszerűnek bizonyult, mégis úgy döntött

az ESRI, hogy fokozatosan megszünteti a támogatását. A 10-es verzióhoz már csak külön telepíthető a VBA, majd pedig ez a lehetőség is megszűnik, tehát mindenkinek új nyelvet kell választani. Több lehetőség adott, de valószínűleg a 9-es verzióval már 2004-ben debütáló Python veszi át a VBA szerepét.

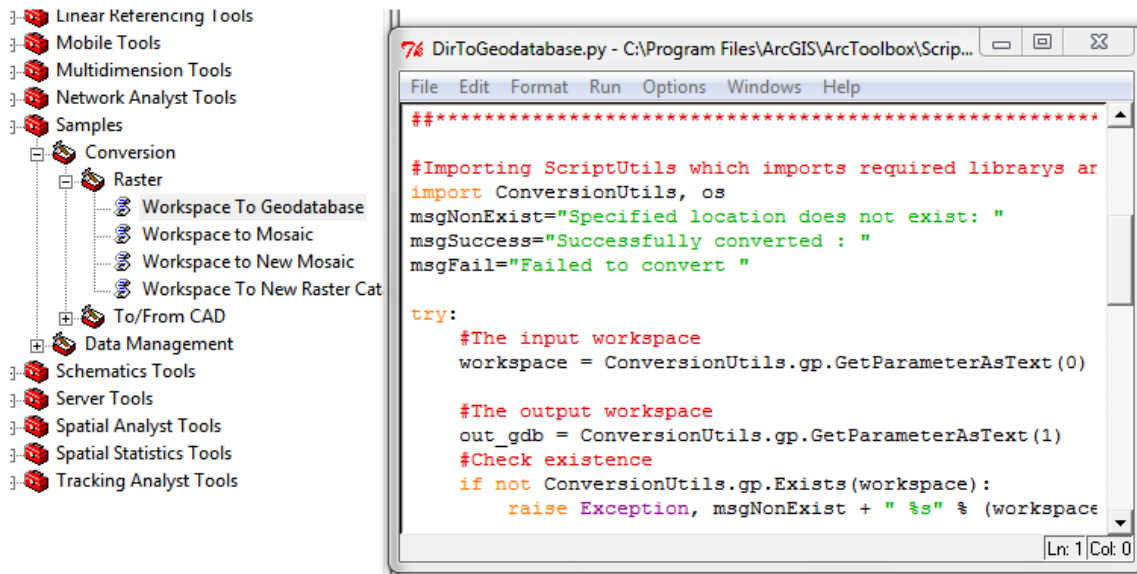
A nyílt forráskódú alkalmazások megjelenése nem új keletű a térinformatikában. Az asztali alkalmazások körében az utóbbi években például dinamikusan nő a QGIS (<http://www.qgis.org/>) népszerűsége, melynek már szerver változata is elérhető. Sőt webes kliensen keresztül elérhető alkalmazásra is találhatunk már hazai példát, mellyel a Mars egyik kráterét fedezhetjük fel [1]. A QGIS terjedését egyértelműen elősegítette az MS Windows operációs rendszerre készült változatok megjelenése, mellyel egyébként már a nagy vetélytárs, a GRASS (<http://grass.osgeo.org/>) is rendelkezik. Az asztali alkalmazások körében a 2000-ben, kimondottan MS Windows platformra fejleszteni kezdett MapWindow (<http://www.mapwindow.org/>) is dinamikusan fejlődik, felhasználói közössége is egyre nagyobb [2]. Az online térképek körében megemlítendő az önkéntesek által készített OpenStreetMap (OSM), mely szabadon felhasználható és szerkeszthető, sőt maga az adatbázis is letölthető és felhasználható a forrás korrekt megjelölésével. Egészen összetett alkalmazások is készülnek már nyílt forráskódú szabad szoftverek felhasználásával [3]. A térinformatikában használható nyílt forráskódú szoftvertermékek teljes körű áttekintését most mellőzzük, csupán a tudományterületen megfigyelhető tendencia érzékeltetése volt a cél. Az aktuális trendekről [4]–ben kaphatunk további információkat.

Ezek után talán már nem meglepő, hogy egy kereskedelmi szoftver nyílt forráskódú, szabad felhasználású programozási nyelvet beemel az egyéni alkalmazások készítéséhez támogatott nyelvek körébe. Akár jelzés értékűnek is tekinthetnénk, hogy nem ellenségnek, hanem egymást kiegészítő lehetőségeknek kell tekintenünk a kereskedelmi és a szabad szoftvereket a jövőben, mely megközelítés mindenképpen üdvözlendő lenne nem csak a térinformatikai termékek piacán.

2. Python térinformatikai célok szolgálatában

A Python egy általános célú, magasszintű, platform független (Unix/Linux, Windows, Mac OS, stb.) programozási nyelv, mely nyílt forráskódú, szabad felhasználású. Guido van Rossum, holland programozó tervezte a 80-as évek végén. Az interpreteres nyelv fő tervezési szempontja a könnyű olvashatóság volt. Kiegészítő termékekkel (pl. Py2exe, Pyinstaller) futtatható programot is készíthetünk a szkriptünkből. Multiparadigmás nyelvként támogatja az objektum orientált, imperatív és a funkcionális programozást is. Dinamikus típusok használata és automatikus memóriakezelés jellemzi. Jelentős mennyiségű standard könyvtára van a Pythonnak, melyek között számos térinformatikai témájú is akad. Elsajátításához könnyen találunk online segédanyagokat, videókat, illetve a támogató közösségek is aktív tevékenységet folytatnak segítve egymást.

Már az ArcGIS 9.0-ás verziójában megjelent a Python, azaz egy teljes telepítés a Python installálásán túl a standard könyvtárak és a NumPy csomag elérhetőségét is eredményezte. A 9.2-es verziótól külön telepítés után akár már a kényelmesebb PythonWin környezetben lehetett kódolni. A geofeldolgozási feladatok elsődleges szkript nyelvének tekintette és tekinti az ESRI a Pythont, hiszen az ArcToolbox számos eszköze is Python kódot takar, melyek akár megtekinthetők, sőt szerkeszthetők (lásd 1. ábra). Az önálló tanuláshoz kiváló ez a lehetőség, ugyanúgy, mint az egyéb internetes honlapokon fellelhető további kódok tanulmányozása. Az átlag felhasználó által legismertebb asztali termék még további helyeken is lehetővé teszi Python kódok elhelyezését (pl. FieldCalculator).



1. ábra. Az ArcToolbox néhány Pythonban íródott eszköze és egynek a forráskódja

A geográfus szakirányos hallgatók egy félév erejéig ismerkednek az egyéni igények kielégítéséhez szükséges programozói ismeretekkel. Döntő többségük még soha nem kódolt, nem tervezett algoritmust, így gyakran komoly kihívás, még heti 5 óra mellett is, a tárgy követelményeinek teljesítése. Az általános programozási ismereteken túl szakirányos, elsősorban az ArcGIS-hez köthető kisebb alkalmazások fejlesztése a cél. A VBA ArcGIS-be integrált fejlesztőkörnyezete és a nyelv MS termékekben való elérhetősége miatt évekig ezt a nyelvet használtuk az órákon, míg a korábban már felvázolt támogatottság megszűnése miatt váltásra nem kényszerültünk. A következő fejezetben a két nyelv, tapasztalataim szerint releváns, tulajdonságait vetem össze.

3. VBA kontra Python

A két nyelv nem ugyanarra a célra lett tervezve, ezért akár el is tekinthetnénk a szigorú értelemben vett összehasonlításuktól. Azonban a geoinformatika szakirányos hallgatók *Szakági programozás* kurzusán egymást cserélte a két nyelv, ezért nem haszontalan a hallgatói erőfeszítések szempontjából egy kis összevetést készíteni. A részletezés előtt azonban meg kell még jegyeznünk, hogy alapvető funkcionális különbségek vannak a két nyelv között. VBA esetén könnyedén adott volt a lehetőség nem csak saját eszköztárat létrehozni, hanem helyi menükben is elhelyezni saját menüpontjainkat. A felhasználóval való kommunikációhoz eseményvezérelt párbeszédablakokat tervezhattunk, ahol a vezérlők különböző eseményeinél használtunk VBA kódot. Nem utolsó sorban közvetlenül lehetőségünk volt az ArcObjects komponens könyvtár használatához, ami gyakorlatilag korlátlan lehetőséget jelentett a térinformatikai adataink menedzselésére. Igaz, mindez csak az ArcGIS környezetben volt kivitelezhető. A Python elsősorban geofeldolgozásokhoz kiváló és nem feltétlenül igényli a futó ArcGIS-t, de a programból elérhető eszközök mögé is fejleszthetünk Python szkripteket.

A Pythonban nem kell deklarálni a változókat, míg VBA-ban nem kötelező. Praktikus okokból én elvártam VBA-ban a deklarációt, ezzel tudatosítva, hogy deklaráció híján *Variant* típusú változónk lesz, melynek valódi típusa az első értékadáskor dől el. (Ennek is van külön haszna, de érdemes tudatosan használni.) Ennek ellenére számonkérésnél előfordultak

deklarátlan változók, azaz terhet jelentett időnként a változók típusának deklarálása. Ezzel teljes mértékben egybecseng, hogy már a Python tanuló hallgatóknak nem volt hiányérzete a deklarációk elmaradása végett. Meg kellett azonban barátkozniuk azzal, hogy pl. az osztás eredményének típusa függ a művelet operandusainak típusától. További odafigyelést igényelt, hogy a kis- és nagybetűket megkülönbözteti a Python, ellentétben a VBA-val. Pythonban ez egyébként számtalan hiba forrása, melynek a felderítése is kihívás egy darabig a hallgatóknak, de jól előkészíti az XML alapú nyelveket. VBA-ban ajánlott ugyan a változónevekben nagybetűket is használni, de gépeléskor elegendő csak kisbetűket leütöni. Ha létezik már a változó, akkor a szerkesztő helyreállítja a kis- és nagybetűk rendjét ugyanúgy, mint a parancsszavak esetében. Ez gyakorlatilag egy lehetőség a nem deklarált változók felderítésére. VBA-ban az objektumváltozók deklarálása két lépésben történik, Pythonban viszont fel sem merül ez a kérdés.

Kezdő fejlesztők nem mindig hajlandók a kódok horizontális tagolására, holott kétséget kizáróan megkönnyíti az olvashatóságot. Pythonban ez nem ajánlás, hanem kötelező érvényű előírás, hiszen egyéb utasítás-zárójel hiányában a *Tab* billentyűvel kivitelezett behúzásokkal jelölhető csak az összetartozó utasítások sorozata (lásd 2. ábra). Időnként gondot okoz az is, hogy a behúzások szóközökkel nem keverhetőek. VBA-ban az utasításoknak van lezáró párjuk, tehát a horizontális tagolás nem kötelező, viszont többszörös egymásba ágyazáskor a lezáró elemek sorának megkeresése fejtörést okozhat, ha elmulasztjuk a tagolást.

```
if rettegSzam == 1:
    uzenet = 'A térképnek egy rétege van.'
else:
    uzenet = ''
    OK = 1
print uzenet
```

2. ábra Python kódrészlet

A programozói gyakorlat nélküli geográfusoknak az egyenlőségjel két szerepkörének a megkülönböztetése szokott még egy kis fennakadást okozni az első időkben Pythonban. Hasonlító operátorként ugyanis duplán kell használni, míg szimplán csak értékadást jelöl. VBA-ban mindkét esetben csupán egyetlen jelet kell használni. Egyik nyelvben sem kell azonban az utasítás-sorokat külön karakterrel lezárni.

Megemlítendő még, hogy az általános célú Pythonhoz fejlesztett modulok teszik lehetővé a különböző téma specifikus felhasználást. Így az ArcGIS 9.x verziókhöz az *arcgisscripting* modul, míg a 10.x verziókhöz már az újabb fejlesztésű *ArcPy* modulra van szükségünk (melyek természetesen a programmal együtt telepítésre kerülnek ugyanúgy, mint a támogatott verziójú Python is).

4. Összegzés

Egyéni alkalmazások fejlesztésére mindig lesz igény, ezért a térinformatikai szoftverek gyártóinak érdemes támogatnia ezeket a törekvéseket. Az ArcGIS esetében a Python váltotta le az eddigi VBA-t, mely új helyzet elé állította fejlesztői közösséget (és a geoinformatika szakirányos hallgatókat). Szintaktikailag kötöttebb a nyelv, de sok tekintetben rugalmasabb, mint a VBA. Külön erénye, hogy további térinformatikai célra is használható [5], hiszen az elérhető modulok száma jelentős.

5. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Sik András: QGIS Server és QGIS Web Client szoftverkörnyezetben készített webtérkép-alkalmazás a marsi Gale-kráterről, *Az elmélet és a gyakorlat találkozása a térinformatikában. V. Konferencia és Szakkiállítás kötete* (2014), 307–315.
- [2] Meems, Paul: MapWindow Open Source GIS, *Az elmélet és a gyakorlat találkozása a térinformatikában. V. Konferencia és Szakkiállítás kötete* (2014), 205–209.
- [3] Szabó, György and Wirth, Ervin: Decision Support in Uncertain Real Estate Transactions, *Az elmélet és a gyakorlat találkozása a térinformatikában. V. Konferencia és Szakkiállítás kötete* (2014), 351–358.
- [4] Siki Zoltán: Trendek a nyílt forráskódú térinformatikai fejlesztésekben, *Az elmélet és a gyakorlat találkozása a térinformatikában. V. Konferencia és Szakkiállítás kötete* (2014), 325–330.
- [5] Zichar, Marianna: Python in GIS, *Az elmélet és a gyakorlat találkozása a térinformatikában. V. Konferencia és Szakkiállítás kötete* (2014), 429–433.

Storage szolgáltatás bővítése az NIIF Intézetnél

Enhancing the storage service at NIIF Institute

Kazinczy Tamás^a

^aNemzeti Információs Infrastruktúra Fejlesztési Intézet
tamas.kazinczy@niif.hu

Székelyi Szabolcs^b

^bNemzeti Információs Infrastruktúra Fejlesztési Intézet
szekelyi@niif.hu

Absztrakt: Felhasználóink számára 2011. első negyedévében vált elérhetővé az NIIF Intézet iSCSI storage szolgáltatása. Az első néhány vállalkozó kedvű felhasználó után a Networkshopon elhangzott előadás hozta meg az áttörést: a jelentkezők száma és így az igényelt területek összege is jelentősen megugrott. 2013. április elejére pedig elérkezett az ideje, hogy kitegyük a megtelt táblát, holott ekkor is érkeztek még új megkeresések. Már a hirtelen megnövekedett érdeklődés általi gyors szabad kapacitás csökkenés korai stádiumában megfogalmazódott bennünk, hogy az infrastruktúrát a későbbiekben bővítenünk szükséges. Később ez kiegészült azzal az ötlettel, hogy a szolgáltatást is bővítsük. Fontos számunkra meglévő és potenciális felhasználóink véleménye, ezért a lehetséges bővülési irányok kijelölése előtt kérdőívben érdeklődtünk, mire van/lenne/lehet szükségük. A felmérés főbb eredményei: a közvetlen iSCSI elérést minden – kérdőívünket kitöltő – felhasználónk megtartaná; felhasználóink fele használna NFS elérést, ha lehetőség lenne rá; mentés/archiválás a legjellemzőbb igény; az összesített terület igény a következő két évre meghaladhatja a 3PB-ot. Az igények, valamint a saját korábbi elképzeléseink alapján az alábbi irányokat vázoltuk fel: iSCSI storage szolgáltatás meglévő kapacitásának bővítése; HSM vagy HSM-szerű szolgáltatás bevezetése; mentés/archiválás (akár a meglévő Archiver szolgáltatás felhasználásával, akár attól függetlenül) szolgáltatás bevezetése.

Kulcsszavak: storage, adattárolás, niif, iscsi, hsm, mentés, archiválás

Abstract: It was in Q1 of 2011 when NIIF launched its iSCSI storage service. Starting with a few pilot users the service quickly became popular after a speech held at Networkshop: both the number of applicants and the total requested capacity increased rapidly. Though requests were continually arising we had to close the application form in April 2013. The idea of further expanding our infrastructure had come up early in the days of the request boom and along with it additional types of service were identified. We had always thought that customer remarks are essential so we had done a survey before we made our final decisions. Results showed that our users want to keep existing (iSCSI) service; would use NFS if it were possible; the main usage scenario is backup/archiving; and that total required capacity would likely to exceed 3PB in two years. Our customers' needs and our own conceptions have synthesized in the following directions: expansion of existing iSCSI storage capacity and establishing new services (an HSM or HSM-like one and a backup/archival – either or not based on our Archiver solution).

Keywords: storage, niif, iscsi, hsm, backup, archive

1. Bevezetés

Felhasználóink számára 2011. első negyedévében vált elérhetővé az NIIF Intézet iSCSI storage szolgáltatása. Az első néhány vállalkozó kedvű felhasználó után a Networkshopon elhangzott előadás hozta meg az áttörést: a jelentkezők száma és így az igényelt területek összege is jelentősen megnőtt.

2. Az NIIF Intézet iSCSI storage szolgáltatása és a bővítés igénye

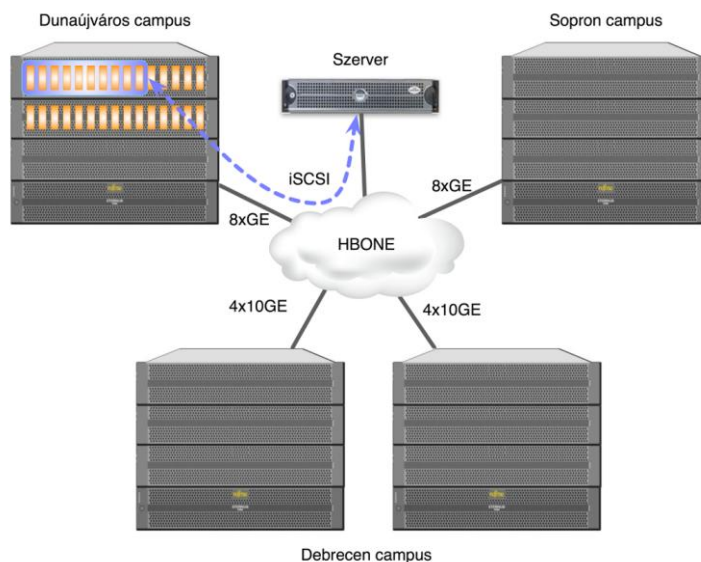
Mi mindenre használják jelenleg a szolgáltatást?

A teljesség igénye nélkül:

- könyvek, folyóiratok, térképek, plakátok, hanganyagok, filmek digitalizált változatai másodpéldányának,
- tudományos mérési adatok,
- diagnosztizálható kórszöveti metszetek szkennelt adatállományainak,
- napi biztonsági mentések (offsite) tárolására, valamint
- szolgáltatás tárhelyként

Az iSCSI storage szolgáltatás három helyszínen biztosít a fenti feladatok kiszolgálásához erőforrásokat: Debrecenben, Dunaújvárosban és Sopronban.

Az infrastrukturális alapokat ehhez jelenleg négy Fujitsu Eternus DX90 S2 típusú midrange adattároló eszköz adja, amelyek egyenként, egyforma 2TB-os NL-SAS diszkekkel felszerelve, 240TB bruttó névleges összkapacitás biztosítására képesek. Debrecenben két, míg Dunaújvárosban és Sopronban egy-egy ilyen adattároló került elhelyezésre (ld. 1. ábra).



1. ábra. Az NIIF intézet iSCSI Storage szolgáltatása

A szolgáltatás népszerűségének következtében 2013. április elejére elérkezett az ideje, hogy kitegyük a megtelt táblát, holott ekkor is érkeztek még új megkeresések.

Már a hirtelen megnövekedett érdeklődés általi gyors szabad kapacitás csökkenés korai stádiumában megfogalmazódott bennünk, hogy az infrastruktúrát a későbbiekben bővítenünk szükséges. Később ez kiegészült azzal az ötlettel, hogy a szolgáltatást is bővítsük.

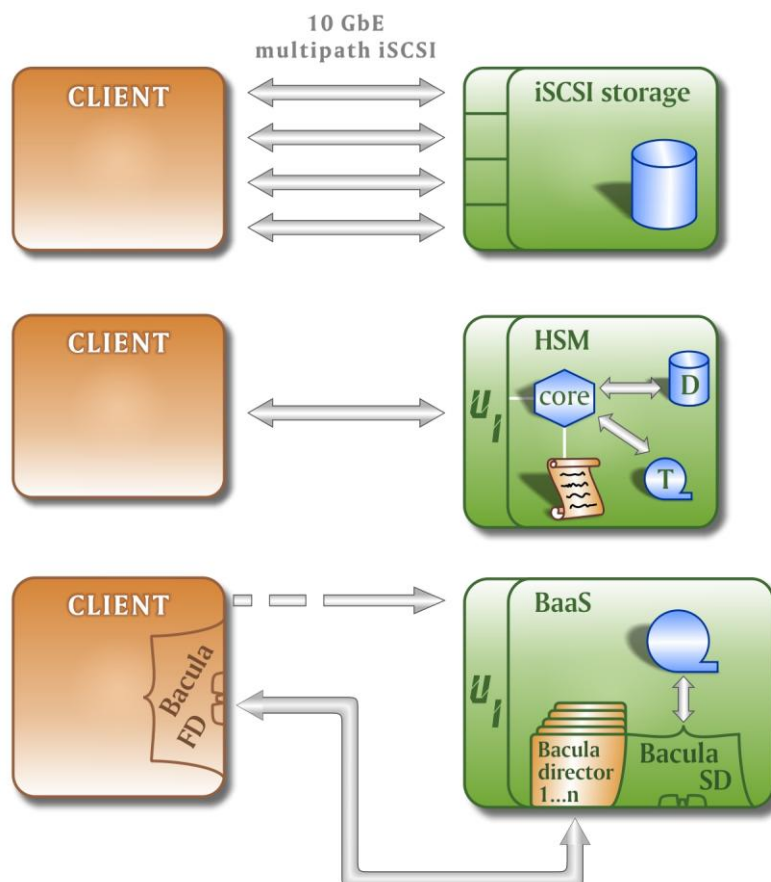
Fontos számunkra meglévő és potenciális felhasználóink véleménye, ezért a lehetséges bővülési irányok kijelölése előtt kérdőívben érdeklődtünk, mire van/lenne/lehet szükségük.

A felmérés főbb eredményei (melyben tehát kizárólag a kérdőívet kitöltő felhasználóink jelennek meg) közé tartozott, hogy: mindenki megtartaná a közvetlen iSCSI elérést; a felhasználók fele használna NFS elérést, ha lehetőség lenne rá; mentés/archiválás céljára felhasználható területekre van a legnagyobb szükség; valamint, hogy az összesített terület igény a következő két évre meghaladhatja a 3PB-ot.

Az igények, valamint a saját korábbi elképzeléseink alapján természetesnek tűnt, hogy az infrastruktúrát, valamint a kapcsolódó szolgáltatást az alábbiak szerint fejlesztjük tovább:

- 1) iSCSI storage szolgáltatás meglévő kapacitásának bővítése
- 2) HSM vagy HSM-szerű szolgáltatás bevezetése
- 3) mentés szolgáltatás bevezetése

3. Fejlesztési tervek



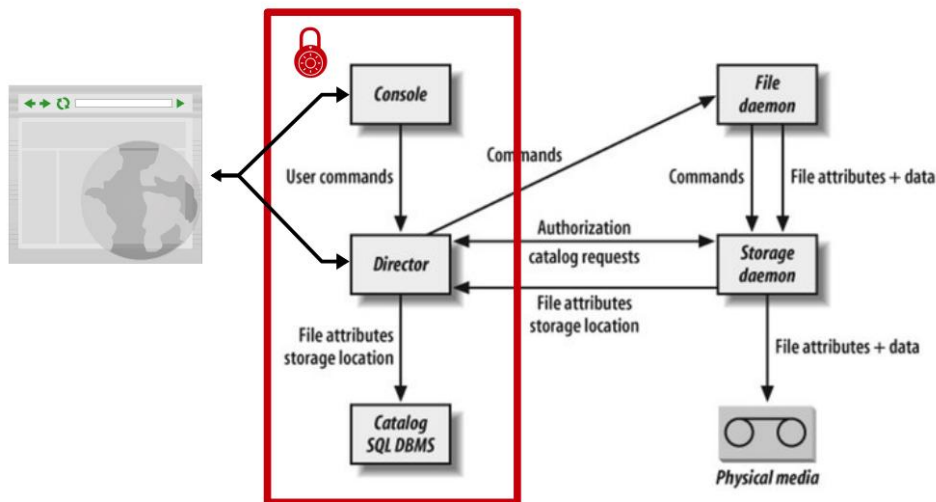
2. ábra. Storage szolgáltatás bővítése az NIIF intézetnél

Fejlesztési terveinket a 2. ábra foglalja össze.

Az iSCSI storage szolgáltatás esetében 1PB nettó formázott kapacitás bővülést céloztunk meg. Ennél a felhasználható kapacitás természetesen kevesebb lesz a redundancia (RAID kötetekbe szervezés) valamint a spare diszkek használata miatt.

A HSM(-szerű) szolgáltatás esetén - terveink szerint – felhasználóink egy pontosabban még meg nem határozott felületen keresztül tölthetik majd fel/le fájljaikat, és azok egységes nézetét láthatják, miközben adataik a háttérben egy policy rendszer működése folytán automatikusan a megfelelő adattárolási rétegre kerülnek. A tervezés jelenlegi fázisában, figyelembe véve a tényt, hogy a rendszert a felhasználók többsége WAN-on keresztül éri majd el, alapvetően két réteggel (diszk és szalag) számoltunk. Ezek együttesen várhatóan 5PB nyers kapacitást nyújtanak majd.

Tervezett mentés szolgáltatásunk (Backup as a Service) keretében a mentett adatokat szalagokon tárolnánk, ezek fölé pedig olyan interfészt illesztenénk, melyen egyszerűen konfigurálhatók a mentendő adatterületek és a mentések üzemeltetése. Az interfész és a szalagok között a Bacula nevű, Intézetünkön belül évek óta meglegedéssel használt enterprise-szintű mentőmegoldást használnánk a 3. ábra szerinti felépítésben. Ez egyben azt is jelenti, hogy a mentendő gépeken a megfelelő Bacula komponensnek (File Daemon, FD) kell futnia. Ez operációs rendszerek és architektúrák széles skálájára érhető el. Az FD-n szinte semmilyen konfigurációt nem kell elvégezni, minden egy központi helyen intézhető. A felhasználók közötti szeparációról az intézményenként egyedi, egymástól független Bacula példányok gondoskodnak saját, szintén felhasználónként dedikált adatbázissal, melyek egy közös tárolót (Storage Daemon, SD) használnak. Mivel azonban a kazetták mindig Bacula példányokhoz dedikáltak, az egyes példányok nem látják egymás kazettáit sem.



3. ábra. Baas szolgáltatás belső felépítése

A megújuló iSCSI storage-, valamint az újonnan bevezetésre kerülő szolgáltatások reményeink szerint elnyerik majd jelenlegi és jövőbeni felhasználóink tetszését.

Irodalomjegyzék

- [1] W. Curtis Preston: Backup & Recovery: Inexpensive Backup Solutions for Open Systems, O'Reilly Media (2007)
- [2] dr. Stefán Péter, Kazinczy Tamás: A NIIFI storage szolgáltatása, <http://www.niif.hu/szolgáltatások/szuperszamitastechnika/storage>

Kollaborációs eszközök alkalmazása K+F projekt portfólió kezelésében

Dr. Dinnyés Álmos^a, Doktor Zsuzsanna^b

^aSzéchenyi István Egyetem
dinnyes.almos@sze.hu

^bSzéchenyi István Egyetem
doktorzs@sze.hu

Absztrakt: A felsőoktatási intézmények fejlesztésének, működtetésének finanszírozásában a pályázati támogatásból megvalósuló projektek jelentősége megnőtt. A pályázatok pénzügyi és adminisztrációs előírásainak való megfelelés, a szakmai munka támogatása és felügyelete a projektek esetében a normál működéstől eltérő folyamatokat és eszközöket igényel. A Széchenyi István Egyetemen működő Járműipari Kutatóközpont a felügyelete alá tartozó K+F projektjei tervezésére és megvalósításuk támogatására MS SharePoint alapú kollaborációs projekt portfólió kezelő rendszert alakított ki. A rendszer jelenlegi változata egységes sablonokra épülő projekt oldalakkal áll. A sablonok tartalmazzák a hagyományos kollaborációs eszközökön – csoportnaptár, címtár, dokumentumtár – túl pénzügyi, szerződéskezelő modulokat, valamint a szakmai munka követésére kialakított szakmai előrehaladási beszámoló tárhelyet. Az egyes projektoldalak adat- és információtartalmának közös adatbázisba kapcsolása lehetővé teszi aggregát riportok készítését tervezéshez, beszámolóhoz, valamint támogatja az adatcserét más egyetemi rendszerekkel (t.k. munkaidő nyilvántartó rendszer). Az eddigi tapasztalatok alapján folyamatban van a következő verzió kialakítása, amelyben megjelennek a jóváhagyási folyamatot támogató work-flow –k, nagyobb hangsúlyt kapnak az egyedi űrlapok, valamint sor kerül a metaadatok konszolidációjára.

Kulcsszavak: projektmenedzsment, csoportmunka támogató portál, sharepoint, projekt portfólió

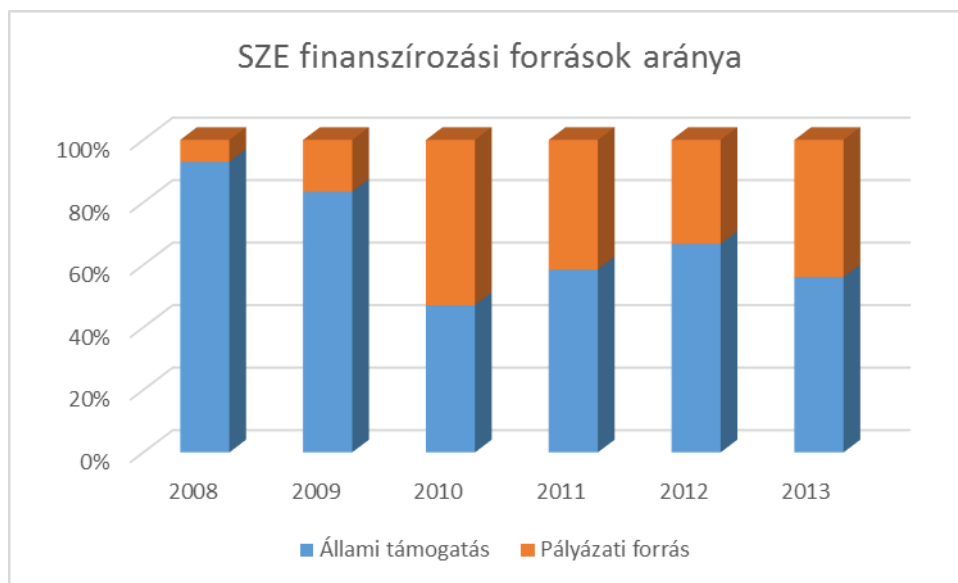
Abstract: The significance of the EU financed tenders has been increased in the operation and development of the higher education institutions. To assure the compliance of the financial and administrative regulations and to support and supervise the professional in projects requires different methods than the ones used in the everyday operation. The Research Centre of Vehicle Industry at the Széchenyi István University has developed a MS SharePoint based collaboration portal for managing its R&D project portfolio. The current system is built on template based project pages. These templates contain beyond the traditional collaboration tools – team calendar, site pages, shared documents – financial and contract handling modules and a customised library to track the progression of the research by form based reports. By joining the databases of the single projects enables to generate aggregate reports for programming, reporting and it supports the data exchange with other internal online systems (like the timesheet system). Based on the experiences gained from the current version the next version of the system is under development. The new version will contain more approval work-flows, the importance of the single forms will be increased and the meta-database will be consolidated.

Keywords: project management, collaboration portal, sharepoint, project portfolio

1. Háttér

A Széchenyi István Egyetem (továbbiakban Egyetem) éves költségvetésében – hasonlóan a többi hazai felsőoktatási intézményhez – a pályázati projektekből származó források aránya megnőtt, az Egyetem esetében 2008. és 2013. között 7 %-ról 44 %-ra, egyúttal az állami támogatás mértéke 1,2 milliárd Ft-tal csökkent. Ez a helyzet az Egyetem működési folyamatainak és informatikai rendszereinek átalakítását tette szükségessé, hogy a pályázati

projektek tervezésével, megvalósításával és fenntartásával kapcsolatos kihívásoknak meg tudja felelni.



1. ábra: A finanszírozási források aránya 2008. és 2013. között a Széchenyi István Egyetemen [1]

Olyan működési modellt kellett kialakítani, amely biztosítani képes több párhuzamosan futó projekt hatékony menedzselését a szakmai munka, a munka szervezés valamint a pályázati és pénzügyi adminisztratív folyamatok esetében. Egy ezeknek az elvárásoknak megfelelő mintamodell kialakítására az Egyetemen a Járműipari Kutatóközpontban (JKK) került sor, ahol három – összesen több mint 3 milliárd összköltségvetésű – K+F projekt esetében 2012 óta sikeresen alkalmazzák. Az elmúlt időszakban már az Egyetem más szervezeti egységéhez kapcsolódó pályázati új projektek esetében is alkalmazni kezdték a teljes modellt vagy annak egyes elemeit.

2. Projekt portfólió menedzsment elvárásai

A JKK jelenleg három K+F projektet valósít meg pályázati forrásból. Ezekben a projektekben konzorciumi tagként hat intézmény vesz részt, és összesen közel 60 részprojektet hajtanak végre. Ennek a projekt portfóliónak a kezelésére alakult ki az alábbiakban ismertetett módszertanának és eszköztár, amelynek kialakításának alapjául a hazai és nemzetközi szakirodalomban, valamint a projektmenedzsment stáb tagjai által megismert ipari és felsőoktatási jó gyakorlatok jelentik. Ez a fejezet mutatja be az előkészítés során azonosított igényeket.

Szakmai munka támogatása

Tudásmegosztás és kommunikáció

A projektekben résztvevő összesen közel 200 munkatárs különböző konzorciumi tagokat képvisel, így a közös munka során a földrajzi távolságot kellett áthidalni, és meg kellett oldani az információ áramlását, a tudás megosztását is. Az egyes projektekben végzett kutatómunka több szempontból is összefügg egymással – közös erőforrások (humán, labor stb.), egymásra

épülő részprojektek, több megoldást integráló prototípus fejlesztése – ezért biztosítani kellett, hogy legyen egy közös tudásbázis.

Olyan megoldást kellett tehát találni, amely a projektekben összegyűlt tudáshoz és szellemi termékekhez biztosítja a tagok számára a szabályozott hozzáférést a projekten belül és projektek között is.

Előrehaladás mérése

A projektekben külső – a támogató által definiált – és belső – a szakmai vezetés által kitűzött – célokot kell elérni. Ezeknek a projektcéloknak az elérése úgy biztosítható, ha teljesítésük időarányosan mérhető. Fontos szempont több projekt esetében a szinergiák feltárása, amelyek révén felgyorsítható az előrehaladás. Ilyen lehetőség az egymással összefüggő – akár eltérő projektekben lévő – kutatási tevékenységekből elkészülő publikációk és más szellemi termékek (pl. know-how –k).

Az előrehaladás mérésére egy olyan többszintű – havi/negyedéves és személyes/részprojekt szintű – beszámolási rendszert kellett kialakítani, amely csak a szükséges mértékben terheli adminisztratív feladatokkal a kutatókat, ugyanakkor elegendő információt szolgáltat a projektek szakmai vezetői számára.

Emellett igényként fogalmazódott meg az is, hogy a projekt során elkészülő beszámolók részévé váljanak az előző pontban felvázolt tudásbázisnak.

Erőforrás kezelés

A projektekben egyes szerződéstípusok esetében sok esetben szükséges a projektre fordított munkaidő központi nyilvántartása és tervezése. Ennek kiemelt szerepe az emberi erőforrások tervezésénél van az esetleges erőforráshiány megelőzésére különösen azokon a területeken, ahol speciális szakértelemre van szükség.

Kockázatkezelés

Pályázati adminisztrációs folyamatok támogatása

A pályázati projektek sikeres megvalósításának kritikus eleme a pénzügyi és kapcsolódó adminisztráció (pl. szerződéskezelés) hatékonysága és megbízhatósága.

Pénzügyi folyamatok

Bár a jelenlegi projekt webhely sablon alkalmas több konzorciumi tag költségvetésének kezelésére, jelenleg csak a konzorciumvezető esetében került bevezetésre.

A pénzügyi folyamatok nyomon követése szükséges projekten belül, illetve projekt portfólió szinten is, hogy biztosítható legyen – különösen az utófinanszírozású projektek esetében – a számlák, megbízási szerződések és a bérek időben történő kifizetése.

Projekt szinten az esetleges változásokat követve kell a rendelkezésre álló pénzügyi kereteket a támogatási szerződésnek megfelelő felosztásban nyilván tartani. Ehhez kell hozzárendelni a költségeket, illetve kapcsolódó kifizetéseket oly módon, hogy az aktuális állapota – pl. kötelezettség vállalás, szerződéskötés, teljesítés igazolás, kifizetve, támogatás igénylése, támogatás utalása – követhető legyen.

Projekt portfólió szinten a projekt szintű adatokat kell összesíteni, és oly módon értékelni, hogy az Egyetem likviditásának tervezéséhez megfelelő adatokat szolgáltatson.

Dokumentáció kezelés

A projektekben keletkező dokumentációnak egyrészt különféle tartalmi és formai szabályozásoknak kell megfelelnie. Olyan megoldásra van szükség, amely előre definiált sablonok alkalmazásával biztosítja az arculatra vonatkozó előírások teljesülését, tartalmi szempontból a szükséges és elégséges információk bevitelét oly módon, hogy a több projektben résztvevő munkatársak számára minden projektben azonos elvek szerint tölthetőek.

Kockázatkezelés

Több projekt esetén a tervezési szakaszban, illetve a megvalósítás során célszerű kockázatelemzést végezni, és a magas kockázatot jelentő projektek esetében kockázatcsökkentő intézkedéseket tenni.

Az egyedi projektek esetében a legfontosabb szempontok a kockázat meghatározásához egy belső, egyetemi elemzés alapján:

- az intézményi stratégiának való megfelelés,
- az erőforrások rendelkezésre állása,
- a pénzügyi keretek nagysága és a finanszírozás típusa (elő- vagy utófinanszírozás, előleg és önerő mértéke),
- a konzorciumi partnerekkel való együttműködés korábbi tapasztalatai.

Természetesen az Egyetemi szintű kockázatértékeléshez az egyedi projektkockázatokat összesíteni kell, és ez alapján lehet megtervezni a kockázatkezelési stratégiát.

3. Megvalósított informatikai háttér

A fenti igényeknek megfelelő megoldás kialakítása során számos tényezőt – idő, költségek, betanítás, kapcsolódó folyamatok – figyelembe véve egy olyan fejlesztési roadmap alakult ki, amelynek első fázisában három technológiát alkalmazunk, amelyeket a jövőben az igények tisztulását és a technológiai lehetőségek fejlődését figyelembe véve lehet egy következő fázisban integrálni.

SharePoint alapú megoldások

A fent bemutatott igények jelentős részének való megfelelést egy Microsoft Office SharePoint Server 2010 (a továbbiakban MOSS2010) kiszolgáló biztosítja. Ez az az alaprendszer, amelyen az egyes projektek számára önálló webhelycsoportokat alakítottunk ki, amelyeknek az adattartalma egy MS Access adatbázisba csatolva érhető el, lehetővé téve aggregát kimutatások készítését. A rendszer testreszabása részben a MOSS2010 webes felületén keresztül történt, az alapértelmezett űrlapok testreszabását az InforPath Designer 2013 alkalmazással végeztük, a munkafolyamatok fejlesztése pedig SharePoint Designer 2010 használatával zajlott.

Jogosultságok kezelése, hozzáférés

A projektekben számos olyan információ keletkezik, amelyekhez való hozzáférést korlátozni szükséges. A pénzügyi és különösen a bérekhez, megbízásokhoz kapcsolódó adatok csak a

projektvezetés, valamint az illetékes egyetemi vezetők számára lehetnek elérhetőek. Ezt a MOSS2010 rendszer alapfunkcióinak megfelelő beállításával lehet biztosítani.

A szakmai anyagokat tartalmi szempontból kell értékelni, hogy meghatározható legyen a besorolásuk: védett (pl. szabadalom előkészítéséhez kapcsolódó anyagok), konzorciumon belül elérhető anyagok, valamint szabadon hozzáférhető információk. A szakmai munkát végző személyek feladata a fenti kategóriákba sorolás, a rendszer támogatja a célcsoportok kialakítását, így biztosítva szükséges védelmet.

Szakmai dokumentáció kezelése

A K+F projektekben nagy mennyiségű szakmai anyag keletkezik, amelyeket több szempont szerint kereshető strukturáltan kell tárolni. Ezt a rendszer oly módon teszi lehetővé, hogy minden projektben központilag karbantartott háttértáblákból (tulajdonság-mátrix) választhatóak minden dokumentumhoz meta adatok, amelyek segítenek a keresésben. A keresését támogatja a MOSS2010 teljes szöveges keresés funkciója is.

Mivel számos kutató több projektben is dolgozik, számukra biztosítani kell, hogy minden projektben azonos lehetőségek álljanak rendelkezésre, át tudják tekinteni akár az összes dokumentumukat. Központilag kialakított és karbantartott sablonokkal érhető el, hogy minden projektben hasonló lehetőségek és funkciók álljanak rendelkezésre. Az összes dokumentum és kapcsolódó információ eléréséhez az MS Outlook jelenti az integráló felületet, ide csatolható be a munkavégzéshez szükséges összes dokumentumtár.

Pénzügyi és adminisztratív modulok

A projekt menedzsmentet támogató legfontosabb modul a pénzügyi adatokat tartalmazza. Az aggregát riportok készítéséhez ez a modul minden projektben azonos felépítésű. Az itt tárolt információk a személyes adatok védelme miatt csak projekt menedzsment stáb számára érhetőek el.

A rendszer tartalmazza a projekt támogatási szerződésének megfelelő költségvetés keretszámait, amihez minden számlát és szerződést rögzíteni kell, megelőzve ezzel a költségvetési keretek átlépését. Mivel az adatok exportálhatóak, egy változás előkészítése során nagymértékben támogatja a költségvetési sorok közti átcsoportosítások modellezését és tervezését MS Excel alkalmazásban.

Ez a modul tartalmazza a szerződések alapadatait, valamint az aláírt szerződések szkennelt változatát is. Ezek az információk több szintén a projekt webhelyén tárolt listához kapcsolódnak: tk. a szerződések teljesítésigazolását jelentő beszámolókhöz, az utalások nyilvántartásához.

A rendszerben minden adminisztrátornak, pénzügyi munkatársnak a saját munkájának hatékony elvégzéséhez szükséges nézetek érhetőek el. Ezeken mindig csak a szükséges és elégséges információk jelennek meg.

Munkafolyamatok

A rendszer munkafolyamatok (ún. work-flow –k) révén épülő jóváhagyási folyamatokat tartalmaz. Egyelőre a belső egyetemi szabályozás a pénzügyi folyamatok esetében ezeket nem támogatja, így a szakmai munka felügyeleténél van jelentősége. Ilyen elektronikus jóváhagyási folyamat biztosítja, hogy egy szakmai grémium felügyelje, hogy mely cikkek jelenhetnek, biztosítva, hogy az esetleges szabadalmi bejelentéseknél ne legyen újdonságrontás.

A jövőben elektronikus aláírások (tanúsítványok) révén várhatóan lesz ezen a téren előrelépés a pénzügyi folyamatok terén is.

Munkaidő nyilvántartó rendszer

A projekt portfólió kezelést támogató MOSS2010 rendszer mellett a munkaidő nyilvántartását egy külön alkalmazás végzi. Ennek oka, hogy nem csak a pályázati projektek esetében van szükség központi adatbázisra, így szélesebb felhasználói kör számára kell elérhetőnek lennie. Ezt a funkciót egy MySQL adatbázist használó, PHP nyelven megírt, az Egyetem portálrendszeréhez illesztett webes alkalmazás látja el. A rendszer bevezetésénél szempont volt, hogy a felhasználók számára ne kelljen új azonosítót és jelszót generálni, így a megvalósult fejlesztés a felsőoktatási intézmények shibboleth autentikációs rendszeréhez csatlakozik.

Kockázati menedzsment rendszer

A kidolgozott eszköztárnak ez az egyetlen nem web alapú eleme. Ennek oka, hogy a kockázatértékelési és -kezelési elvek további finomításra szorulnak, így egyelőre ez egy szűk szakértői réteg számára használható modellező eszköz.

A rendszernek két fő eleme van. Egy MS Access alapú adatbázis, amely a projekt portfólió kockázatainak nyilvántartására, valamint az adatok úrlapon történő rögzítésére alkalmas. A másik elem egy MS Excel alapú modellező eszköz, amely az adatbázisban rögzített pénzügyi adatok (várható kiadások és bevételek) alapján segít megtervezni az egyes projektek optimális kezdési dátumát, a projektek hosszát és a kifizetési kérelmek ütemezését.

4. Összefoglalás

Tapasztalatok, fejlesztési lehetőségek

A projektek szakmai megvalósítása során megállapítható, hogy jelentősen javult a kommunikáció és az információ megosztás hatékonysága. Az egymástól földrajzilag külön dolgozó munkatársak között a hatékony információáramlásnak köszönhetően a közös munkák eredményesség nőtt a korábbi projektekhez képest.

Fejlesztési lehetőség az adminisztratív folyamatok – különösen a jóváhagyások – területén van lehetőség például elektronikus aláírásokat (tanúsítványokat) használó work-flow –k révén egy papírmentes iroda koncepció megvalósításával.

További fejlesztési lehetőség a munkaidő és a kockázati menedzsment rendszer integrálása a SharePoint környezetbe.

Hosszabb távon javasolt a MOSS újabb verziójára áttérni, mivel számos olyan funkciót tartalmaz, lehetővé teszi a csak webes működést. További funkciókat tesz elérhetővé, amennyiben az Egyetem jelenleg használt levelező szerverét az extra szolgáltatások is biztosító MS Exchange használatára tér át.

Visszajelzések külső féltől

A projekteket ellenőrző közreműködő szervezet (2013. végéig az ESZA Nonprofit Kft.) évenkénti ellenőrző projekt látogatásai során elfogadta, és pozitívan értékelte az alkalmazott rendszert és módszertant. Az ellenőrzések időigénye jelentős mértékben csökkenthető volt

azáltal, hogy a dokumentumok jelentős része elektronikus formában, strukturáltan, kereshető módon a webes felületen keresztül elérhető volt.

5. Köszönetnyilvánítás

A cikkben bemutatott módszertanok kialakítása és bevezetése az alább felsorolt, a Magyar Állam és az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával megvalósuló projektekből történt, valamint a konferencián való részvételt a Széchenyi István Egyetem Informatika Tanszéke tette lehetővé.

- TÁMOP-4.1.1.C-12/1/KONV-2012-0002: Járműipari Felsőoktatási és Kutatási Együtműködés
- „TÁMOP-4.2.2.C-11/1/KONV-2012-0012: „Smarter Transport” - Kooperatív közlekedési rendszerek infokommunikációs támogatása
- „TÁMOP-4.2.2.A-11/1/KONV-2012-0012: Hibrid és elektromos járművek fejlesztését megalapozó kutatások

Irodalomjegyzék

[1] Széchenyi István Egyetem éves költségvetési beszámoló 2008-2013

[2] Mihály Görög: A strategic oriented implementation of projects, Projekt Irányítási Kézikönyv

[3] Széchenyi István Egyetem Járműipari Kutatóközpont belső szabályzata

[4] Halász Gábor (2011): A felsőoktatási stratégiai gondolkodás helyzete a világban: intézményi és ágazati gyakorlatok. In: AVIR Tanulmánykötet, Stratégiai gondolkodás a felsőoktatásban, 71-93. oldal, Educatio Társadalmi Szolgáltató Nonprofit Kft.

Sivák és Brahmák a BigData korában

Szabó Bernadett

SAS Institute Kft.

Bernadett.Szabo@sas.com

Absztrakt: A SAS-nál fontosnak tartjuk, hogy az egyetemről kikerülő új szakemberek tudása találkozzon ügyfeleink munkaerő igényével.

Álláshirdetések között keresgélve, különösen az üzleti intelligencia területén, azt tapasztalhatjuk, hogy rendszerint több kereskedelmi szoftver ismerete is az előnyök között szerepel. Azaz az informatikai vagy analitikai tudás mellett a megfelelő szoftverek ismeretére is szükség lenne.

Kulcsszavak: BigData, analitika, SAS, adatbányászat, SQL, Egyetemi program

1. Bevezetés

A SAS-nál fontosnak tartjuk, hogy az egyetemről kikerülő új szakemberek tudása találkozzon ügyfeleink munkaerő igényével.

Álláshirdetések között keresgélve, különösen az üzleti intelligencia területén, azt tapasztalhatjuk, hogy rendszerint több kereskedelmi szoftver ismerete is az előnyök között szerepel. Azaz az informatikai vagy analitikai tudás mellett a megfelelő szoftverek ismeretére is szükség lenne.

Természetesen az egyetemek igyekeznek ezt biztosítani, de főként csak a nyílt forráskódú vagy ingyenesen elérhető szoftverek körében teszik/tudják. Ez nem is lenne gond, amíg csak a kis- esetleg középvállalkozások körében maradnak a végzős hallgatók. Viszont a nagy pénzügyi, államigazgatási és szolgáltatási szektorba kerülve ez már nem elegendő. Itt elengedhetetlen már a kereskedelmi szoftverek ismerete. A kutatásfejlesztés falain túl csak ezek képesek megfelelően együttműködni az adatbiztonsági szoftverekkel, ezek tudják az auditálást biztosítani vagy kezelni, lekérdezni a BigData-t.

Persze mondhatjuk, hogy majd az adott intézmény továbbképzzi frissdiplomásunkat, ha neki egyedi igényei vannak. De ma már a SAS alapjainak az ismerete sem egyedi igény! Így sokszor a próbaidő jelentős részét teszi ki a szoftver Önálló elsajátítása a szakmai tudás maximális megnyilvánulása helyett. Ráadásul, többször szembesülünk azzal, hogy az így megszerzett tudás nem megfelelő, hiányos, rendszerezést igényel. Legtöbbször a nagy adatmennyiségre való alkalmazás esetén válik szembeszökővé. Viszont ügyfeleinknél, oktatási keret alaptudásra már nem áll rendelkezésre, csak a specializáció támogatott.

A SAS az alap és mester képzésben, az oktatók, hallgatók és a kutatók támogatására hozta létre a SAS Global Academic Programot (GAP), amely lehetőséget teremt minden olyan SAS tudás megszerzésére még a felső- és felnőtt oktatásban, amit ügyfeleinknél tömegesen használnak a SAS szoftvereiből a világban.

2. SAS Global Academic Program

Oktatási segédletek és konzultáció

Oktató anyagaink (könyv, slide, gyakorló feladatok...) elérhetők az egyetemi oktatók számára, ahhoz hogy SAS kurzust tarthassanak. Az anyagok teljes egészben vagy részleteiben is felhasználhatók a diplomát adó képzések hallgatóinak oktatásában. [1]

Konzulenseink segítenek minden egyetemi kurzus során, ahol a SAS-t szeretnék használni. Tematika áttekintése, feladatok, gyakorlatok tervezése egy féléves tárgyhoz vagy akár több szemesztert átölelő képzéshez, hogy a tudás célba érjen.

Elérhető szoftverek

- **SAS University Edition**

Egy ingyenes csomag ajánlat a SAS szoftver alapmoduljainak eléréséhez. Hallgatók, oktatók, kutatók és örökké tudásra szomjazók részére, azaz bárkinek elérhető. Magába foglalja a következőket:

- Base SAS (SAS programozási nyelv);
- SAS/STAT (statisztikai módszerek);
- SAS/IML (mátrix algebra nyelve);
- SAS Studio (interaktív felhasználói felület a SAS-hoz)
- SAS/Access Interface to PC files (pl.: Excel, JMP, Paradox,SPSS, Stat, DBF)

A modulok letölthetők Pc-re, Mac-re vagy Linuxra. [2]

- **SAS OnDemand for Academics**

hozzáférést biztosít a SAS számos képességének eléréséhez az interneten keresztül, ingyenesen. A feladat végrehajtás a SAS által biztosított szerveren történik saját internet kapcsolatunkon keresztül. Teljes frissítés és bővítés 2014. augusztus 18-tól érhető el. SAS OnDemand az alábbi legnépszerűbb SAS szoftverek elérhetőségét teszi lehetővé:

- SAS Enterprise Guide – point and click interface to power of SAS;
- SAS Enterprise Miner – SAS hatékony adatbányászati megoldása;
- SAS Forecast Server – Előrejelző modellek;
- SAS Studio – Platform-független kódszerkesztő

SAS OnDemand for Academics csak az egyetemi oktatásban érhető el. Házi feladatokra, beadandók, vizsgamunkák készítésére valamint a levelező hallgatók igényeire tervezve. [3]

- **SAS Visual Analytics**

esettanulmányokon keresztül ismerhető meg, költségtérítés nélkül, a Teradata University Network (TUN) [4] és SAS között létrejött partnerség révén. A SAS Visual Analytics lehetőséget biztosít az oktatóknak és hallgatóiknak az adatfeltáráshoz, riportkészítéshez és az eredmények megosztásához a Weben vagy mobil eszközökön. A szoftverhez ingyenes oktatóanyag is kapcsolódik a TUN-ban. Használatához internetkapcsolat szükséges.

- **SAS Education Analytical Suite**

A SAS analitikai megoldásai egy csokorban oktatáshoz és kutatáshoz tervezve. Kutatói, tanszéki, kari vagy egyetemi formában igényelve telepíthető az egyetem adott számú számítógépére. Legmegbízhatóbb óraszervezést teszi lehetővé és az Educational Analytical Suite [5] igénylésével most ingyenesen elérhetővé válik számos SAS e-learning kurzus [6] a következő témákhoz: SAS programozás, SAS makró nyelv, SAS SQL, a SAS Enterprise Guide használata, Statisztika: ANOVA, regresszió, logisztikus regresszió és prediktív modellezés logisztikus regresszióval. Kampusz szinten licenselt Enterprise Miner esetén a SAS Enterprise Miner szoftverhez kapcsolódó alkalmazott analitika és rapid prediktív modellezés kurzusok is elérhetőek.

- **JMP for Statistical Discovery (mérnökképzésekben ajánljuk)**

Vizuális, interaktív és könnyen elsajátítható analitikai alkalmazás a statisztikai összefüggések gyors feltárára. Táblázatos és grafikus eredmények azonos ablakban. A hallgatók point and click és drag and drop technikán alapuló egyszerű szűrésekkel, eljárások tovább finomításával, 3D-s és mozgó grafikonok készítésével juthatnak közelebb a statisztikai eredményeik megértéséhez. JMP Win és Mac platformokra igényelhető. [7]

3. Minősítések

A diplomához SAS minősítést is szerezhetnek a hallgatók, ha az egyetemi képzéssorozat megfelel a SAS minőségbiztosítási előírásainak. Az egyetem akkreditálhatja statisztikai, adatbányászati vagy programozási képzéssorozatát a Joint Certificate Program-ban. [8]

4. SAS diák nagykövet program

Ez a program lehetőséget biztosít a hallgatóknak, hogy SAS-sal készült eredményeiket, angol nyelven publikálják az éves, tavaszi, nemzetközi SAS Global Forum-on, amennyiben annak eredményét a döntőbizottság újszerű megoldásnak, újszerű felhasználásnak, ügyes tippnek vagy trükknek ítéli. A kiírás minden év negyedik negyedévében jelenik meg. [9]

5. Összefoglalás

Sokszor vádolnak minket, hogy túl zártak és elérhetetlenek vagyunk. Remélem ezzel az összefoglalással sikerült rációfolni erre és hamarosan ügyfeleink elégedettségén keresztül mérhetjük majd le az egyetememen induló SAS programok eredményességét. Valamint az álláshirdetésekből bátran az elvárt kategóriába írhatják az igényelt kereskedelmi szoftverek ismeretét az előny helyett.

Irodalomjegyzék

- [1] SAS Global Academic Program elérhető oktató anyagainak listája:
<http://support.sas.com/learn/ap/tkit/list.html>.
- [2] SAS University Edition információi: www.sas.com/universityedition
- [3] SAS OnDemand for Academics információi: www.sas.com/academics
- [4] Teradata University Network: <http://www.teradatauniversitynetwork.com/>
- [5] Educational Analytical Suite:
http://www.sas.com/resources/asset/sas_education_analytical_suite.pdf
- [6] SAS e-learning kurzus információi:
<https://support.sas.com/edu/elearning.html?ctry=us&productType=library>
- [7] JMP for Statistical Discovery információi: www.jmp.com/academic/
- [8] Joint Certificate Program: <http://support.sas.com/learn/ap/prof/index.html#t4>
- [9] SAS diák nagykövet program információi:
<http://support.sas.com/learn/ap/student/amb.html>

Ansible

Tóth Sándor

Nemzeti Információs Infrastruktúra Fejlesztési Intézet

toth.sandor@niif.hu

Absztrakt: Az informatikai infrastruktúra általa nyújtott lehetőségeket egyre többen szeretnék kihasználni egy-egy intézményen belül. Ami egyfelől örvendetes, ugyanakkor az egyre sokrétűbb és egyre több szolgáltatáshoz egyre nagyobb és komplexebb infrastruktúrákat kell felépítenie és fenntartania az informatikát üzemeltető szakembereknek. Ennek a kihívásnak a kezelésére az egyik lehetséges megoldás a szolgáltatások indításának, illetve beállításainak egységesítése majd automatizálása. Ebben tud segíteni az Ansible. Az Ansible egy nyílt forrású, telepítő és konfiguráció menedzsment eszköz, amelyet Michael DeHaan írt 2012-ben Python nyelven. Van egy konzervatívabb hozzáállással fejlesztett magja, amelyhez API-n keresztül kapcsolódik nagyon sok modul. Tulajdonképpen ezek a modulok valósítják meg az egyszerű kezelhetőséget és működést. A szerverek konfigurációjának menedzseléséhez emberi szemmel is jól olvasható és érthető, egyszerű szöveges fájlokat kell létrehozni, majd parancssorból futtatni. Ugyanakkor az Ansible nem klasszikus szerver-kliens architektúra, mivel a menedzselte eszközökre nem kell telepíteni semmilyen ügynök programot, illetve Ansible démonok sem fognak sehohly futni. Az Ansible SSH-n keresztül kommunikál és csak addig "él", amíg a feladatot végrehajtja.

Kulcsszavak: Ansible, konfiguráció menedzsment, nyílt forráskód

Abstract: Within a single institution there are more and more people who want to take advantage of the opportunities provided by the IT infrastructure. This is rejoice on the one hand, but on the other hand the IT professionals should build and maintain bigger and bigger and more and more complex infrastructures for the increasingly diverse services of which there are more and more. To manage this challenge one of the possible solutions is to standardise the launch and settings of services and then to automate them. Ansible can help with it. Ansible is an open source, deployer and configuration management tool written by Michael DeHaan in Python in 2012. There is a core developed by a more conservative approach, to which several modules are connected through an API. Actually, these modules ensure the ease of use and functionality. To manage the configuration of servers you should be created human eye readable and understandable simple text files then run it from the command line. However, Ansible is not a classic client-server architecture, because it does not need to install any agent programs on the managed machines, and Ansible demons will not run anywhere either. Ansible communicates via an SSH and will only be "alive" until the job is performed.

Keywords: Ansible, configuration management, open source

1. Bevezetés

A felsőoktatási intézményeknek számos sajátossága van, amelyek nagyméretű és magas színvonalú informatika üzemeltetését követelik meg. Ezen sajátosságok egyike a magyarországi léptékkal mérve jelentős felhasználószám, ami ugyan az utóbbi években csökkent, de az informatikával szembeni igények ennek ellenére egyáltalán nem csökkentek. Ennek egyik oka lehet, hogy a mai kor informatikai felhasználói sokkal több, sokkal szerteágazóbb, sokkal magasabb minőségi igényeket fogalmaznak meg. Például ma már közhely, hogy, aki nincs az Interneten, az nem is létezik, tehát szinte alapértelmezett igény lett, hogy minden szervezeti egységnek legyen saját web oldala, vagy egy-egy projektnek,

pályázatnak is kötelező eleme a webes megjelenés. Természetesen rengeteg további olyan igény is felsorolható lenne, amely néhány évvel ezelőtt még nem is létezett.

A fentiekkel párhuzamosan ugyanakkor az is igaz, hogy a rendelkezésre álló pénzügyi, műszaki vagy humán erőforrás kapacitások korlátosak, ami nyilvánvalóan feszültséget fog keletkeztetni az informatika felhasználói és az informatika üzemeltetői között. Ezt a feszültséget többféleképpen is kezelhetjük: több erőforrást biztosítunk az informatikának, növeljük a hatékonyságot vagy valamilyen módon egységes kereteket alakítunk ki, azaz standardizáljuk az üzemeltetést és a szolgáltatásokat. Szerverek esetében ezt lehet például egy server policy, vagy szolgáltatások esetében egy SLA (Service Level Agreement – Szolgáltatás Szint Megállapodás). A standardizálás pozitívuma, hogy az ilyen működés könnyen automatizálható, azaz informatikai szemszögből leprogramozható, így ugyanakkor a feladatnak az ismétlései már sokkal kevesebb erőforrásból elvégezhetőek.

A standardizált működést hivatottak támogatni a konfigurációmenedzselő alkalmazások. A konfiguráció menedzsment általánosan elfogadott definícióját nehéz megadni, mivel ha csak a wikipedia.org [1] weboldal kapcsolódó szócikként megnézzük, ott is több mint 20 irányelv illetve szabvány van felsorolva. Szoftverek esetében azt mondhatjuk, hogy a konfiguráció menedzsment célja, hogy egy adminisztratíván előre meghatározott működési állapotot elérjünk, illetve azt folyamatosan fenntartsuk. A konfigurációmenedzselő alkalmazások tulajdonképpen egy absztrakciós réteget biztosítanak számunkra, amelyekben le tudjuk képezni a server policy-kban vagy az SLA-kban megfogalmazott adminisztratív elvárásokat. Azonban a konfigurációmenedzselő alkalmazások kezdeti fejlesztését alapvetően nem, az akkor még alig létező, elméleti megfontolások indukálták, hanem az olyan gyakorlati kihívások, mint az időhiány, az unalmas ismétlődések, a nagyszámú eszköz azonos állapotban való tartása, stb. Ezekre válaszul mind a nagy szoftvergyártók, mind nyílt forráskódokat fejlesztő közösségek elkészítették a saját megoldásaikat. A nyílt forráskódú alkalmazások közül a Cfengine a legkorábbi (1992), de ma is aktívan fejlesztett megoldás, de mellette a Puppet, a Salt vagy a Chef is széles körben ismertek.

Az Ansible a szoftverkonfigurációkat menedzselő alkalmazások közül az egyik legújabb. A 2012-ben kiadott, 0.01-es verziótól [2] kezdődően nyílt forráskódú, GNU GPL licenclésű [3], Python nyelven íródó alkalmazás, amelynek megalkotója, és mai napig fő fejlesztője Michael DeHaan. A fejlesztés fő célja az volt, hogy nem csak a használatában, hanem a felépítésében és a működésében is egy egyszerű, könnyen tanulható szoftver legyen az Ansible [4].

2. Miért Ansible?

A Nemzeti Információs Infrastruktúra Fejlesztési Intézet (NIIFI – niif.hu) helyzetéből eredően nagyon nagyszámú eszközt és szolgáltatást működtet, amelyek konfigurációjának kezelése jelentős feladat. Ennek hatékonyabb és koherensebb menedzseléséhez kerestünk egy jól használható szoftvert. Az Intézet profilját markánsan meghatározza a szabadszoftveres filozófia, így alapvető szempont volt, hogy open source megoldás legyen, amely korrektül képes együttműködni Linux/Unix operációs rendszerekkel és a legelterjedtebb open source alkalmazásokkal. Ennek a szempontnak több, fentebb is említett szoftver megfelelt, ugyanakkor az alább részletezett három szempont miatt az Ansible-re esett a választásunk.

2.1. Nincs kliens program

A szoftverek konfigurációjának menedzselésében általánosságban három alapelemet különíthetünk el:

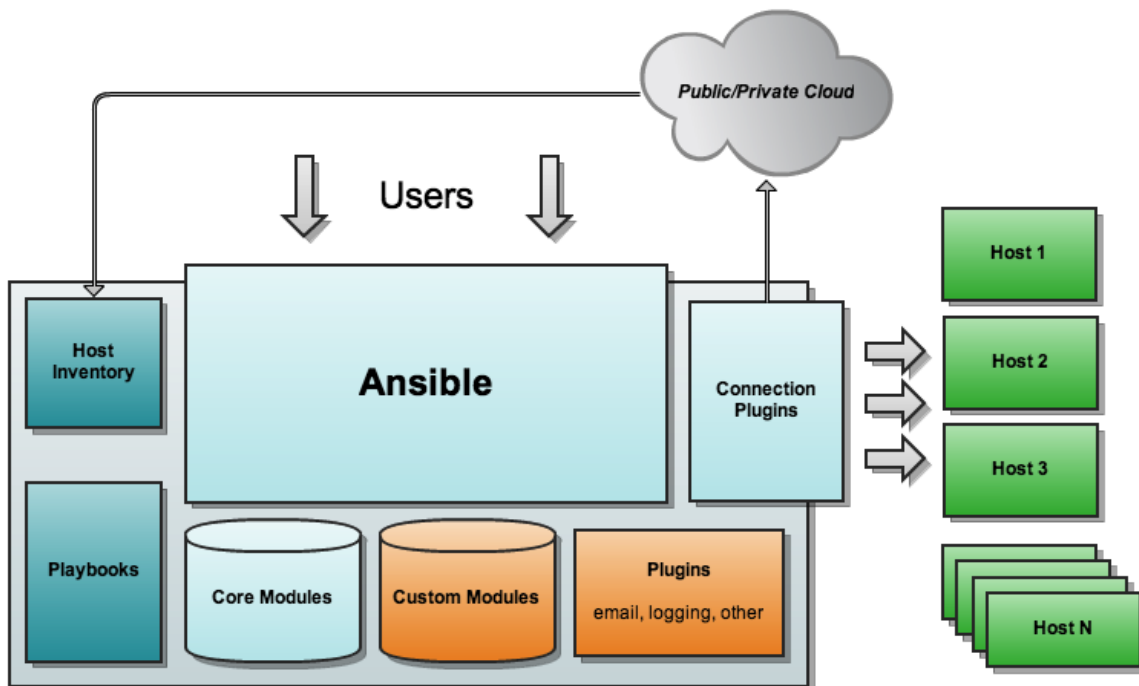
- a menedzsert,
- a klienst,
- a kommunikációt a menedzser és a kliens között.

A legtöbb konfigurációmenedzselő alkalmazás a klienseken úgy valósítja meg a konfigurációkezelést, hogy egy úgynevezett ügynök programot (agent) telepít oda, amely kommunikál a menedzserrel és a tőle kapott információk illetve utasítások alapján beállítja a kliens konfigurációját. Ennek a működési modellnek számos pozitívuma van, de nyilvánvalóan ugyanazokat a többletfeladatokat is magával vonzza, mint amelyek a menedzselni szándékozott szoftverekre jellemzőek. Azaz az ügynök program is megkövetel a működéséhez bizonyos szoftverkönyvetet, az ügynök program frissítésekor ezt a környezetet is a frissített igényekhez kell igazítani, a működését célszerű folyamatosan monitorozni, a különböző verziójú ügynök programokhoz különböző lehet az utasítások szintaxisa, stb.

Az Ansible ennek a helyzetnek a kialakulását úgy kerüli el, hogy nem használ ügynök programot a klienseken. Annyit követel csak meg, hogy a kliens gépeken legyen SSH és a Python legalább 2.6-os verziója. Az Ansible a konfigurálás végrehajtásához szükséges modulok bináris verzióit egy ideiglenes könyvtárba a kliens gépre másolja SCP-vel, majd a feladat végeztével törli azokat [5].

2.2. Moduláris

Az Ansible tervezésénél az egyszerűség minden rétegben fontos szempont volt. A szoftver architektúrájában is igyekeztek ezt az egyszerűséget, átláthatóságot megtartani, ahogy az az 1. ábrán látható.



1. ábra Az Ansible architektúrája (Forrás: <http://terry.im/wiki/terry/Ansible.html>)

Az Ansible felépítésére a modularitás a jellemző. Van egy konzervatív szemléletben fejlesztett „core” modulja és további egyedi modulok. Tulajdonképpen ezekben az egyedi modulokban testesülnek meg azok a funkcionalitások, amelyek segítségével az Ansible a klienseken parancsokat tud futtatni, illetve a megfelelő módon kezelni tudja a kliensen futó alkalmazásokat. Az egyes modulokról az Ansible weboldalán lehet részletes leírást találni (http://docs.ansible.com/list_of_all_modules.html). Ha az általunk elvégzendő feladathoz nincs megfelelő modul, akkor magunk is írhatunk egyet bármilyen nyelven, és csak annyi a megkötés, hogy JSON formátumot adjon vissza, amelyet az Ansible fel tud dolgozni.

Itt érdemes megjegyezni, hogy az Ansible-nek nagyon gyors a fejlesztése, szinte minden hónapban adnak ki új verziót, így a modulok száma rohamosan növekszik. Ennek a fejlesztési aktivitásnak azonban egyik következménye, hogy a korábbi verziókban csak a kritikus hibák kerülnek javításra, azaz érdemes mindig a legfrissebb verziót használni.

Korábban csak Unix/Linux rendszerek menedzselésében lehetett használni az Ansible-t, de a legújabb fejlesztések eredményeképpen egyre több modul segíti a Microsoft Windows rendszerek kezelését is [6].

2.3. Egyszerű szerkeszthetőség

Az Ansible-ben a kliensen végrehajtandó feladatok egy szöveges fájlban kerülnek leírásra, amit „playbook”-nak neveznek. Ezt a szöveges fájlt bármelyik, általunk kedvelt szövegszerkesztővel létrehozhatjuk azzal a megkötéssel, hogy a szöveg szintaxisának a YAML [7] nyelvnek kell megfelelnie. A YAML nyelven írott szöveges fájlok emberi szemmel is viszonylag jól olvashatók és könnyen átláthatóak. Ennek köszönhetően a tanulási görbéje kedvező, különösen egy olyan környezetben, mint az NIIFI, ahol az üzemeltetési feladatokat gyakran parancssoros környezetben kell végezni, sokszor szöveges konfigurációs állományok szerkesztésével.

Ha szükséges, akkor sablonok is készíthetők, amelyhez a Jinja2 [8] használható. Ez egy sablonkészítő modul a Python nyelvhez, BSD licenccel [9].

3. Az Ansible rétegei

Az Ansible használatához a telepítését követően csak egy fájlt kell szerkesztenünk. Ez a fájl alapértelmezetten a `/etc/ansible/hosts`. Ebben a fájlban tudjuk megadni a kliens gépeket névvel vagy IP címmel, de csoportosíthatjuk is őket, illetve a csoportok is csoportosíthatóak [10]. Egy példakonfigurációt az alábbi 2. ábra szemléltet.

```
$/etc/ansible/hosts
```

```
192.192.92.2  
mail.example.com
```

```
[teszt]  
93.121.169.136
```

```
[webservers]  
foo.example.com  
bar.example.com:5555 ansible_ssh_host=192.168.1.50
```

```
[database]
foo.example.com
www[01:50].example.com
```

2. ábra Az Ansible `/etc/ansible/hosts` fájl példakonfigurációja

A kliens gépek menedzselésére, illetve a velük kapcsolatos feladatok szervezésére/rendezésére az Ansible három rétegben biztosít lehetőséget, amelyek az alábbiakban kerülnek részletes bemutatásra.

3.1. Ad hoc parancsok

A `/etc/ansible/hosts` fájlba felvett klienseket az Ansible legelső rétegében egyszerűen parancssorból is elérhetjük az `$ ansible` parancs segítségével. Ez a lehetőség olyan feladatok elvégzésében segíthet, amelyeket nem akarunk menteni vagy egy alkalommal egyszerre sok gépen szeretnénk lefuttatni (pl. leállítás). Jó hasznát vesszük akkor is, ha csak tesztelni szeretnénk a kapcsolat meglétét a kliens és a menedzser gép között, amit a 3. ábrán bemutatott parancs szemléltet.

```
$ ansible teszt -m ping (-vvvv)
93.121.169.136 | success >> {
  "changed": false,
  "ping": "pong"
}
```

3. ábra Az `$ ansible` parancs használata

A fenti ábrán egy parancssorból kiadott `$ ansible` ad hoc parancs segítségével a korábban az `/etc/ansible/hosts` fájlba felvett `[teszt]` csoport gépein (amely csoport történetesen egy gépet jelent) az Ansible `ping` modulját (`-m`) futtatjuk le. Ha a modul sikeresen lefut a teszt csoport gépén, akkor a „ping”-re egy „pong” választ kapunk, azaz a kliens és a menedzser gép között a kapcsolat hibátlan.

Amennyiben részletesen is meg szeretnénk tudni, hogy mit is csinál a `ping` modul a kliens gépen, akkor a zárójelektől megszabadított `-vvvv` kapcsolóval adjuk ki újra a fenti parancsot.

3.2. Playbook

Az Ansible valódi lehetőségeit azonban a `playbook`-okon keresztül lehet kihasználni. Ezek a szöveges fájlok `Yaml` nyelvűek és `.yml` kiterjesztésűek a konvenciók szerint. Tulajdonképpen ezekben a fájlokban történik meg azoknak a feladatoknak a leírása, amelyeket a klienseken szeretnénk elvégeztetni. A 4. ábra mutat egy példa `playbook`-ot.

```
$ apache.yml
- hosts: teszt
  remote_user: toth
  sudo: yes
```



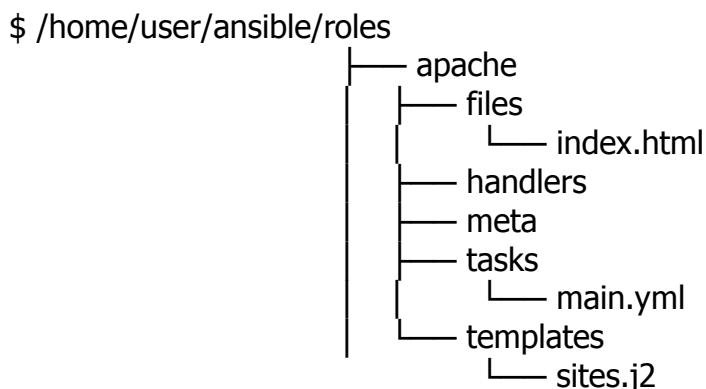
```
tasks:
- name: apache2 telepites
  apt: name=apache2
      state=installed
  notify:
  - start apache2
- name: index.html feltoltese
  copy: src=files/index.html dest=/var/www/index.html
- name: index.html jogok beallitasa
  file: path=/var/www/index.html owner=www-data
```

4. ábra Példa Ansible playbook fájl

A fenti `apache.yml` playbook egy apache webszerver telepítését mutatja be a `[teszt]` csoportba tartozó klienseken (`hosts`). A klienseken a `toth` felhasználó nevében fogunk tevékenykedni, és a `sudo` lehetőség engedélyezett. Az elvégzendő feladatokat (`tasks`) elnevezzük (`name`), ami segíti az átláthatóságot és a nevek szerint kapunk visszajelzést a feladatok végrehajtásakor azok sikerességéről. A nevek alatt kerülnek meghívásra az egyes modulok (pl.: `apt`, `copy`, `file`), illetve a modulok által elvégzendő konkrét feladatok.

3.3.Role

Természetesen a több tucat szerver, szolgáltatást üzemeltető szervezeteknél hamar beleütközünk olyan kihívásokba, hogy egyetlen playbook-ba nem lehet minden feladatot belecsúfolni vagy, hogy egyes részfeladatokat ugyanúgy kellene végrehajtani minden esetben, míg másokat csak egy-egy kliensen, azaz a playbook-nak csak egyes részei különböznek, míg sok részük azonos, stb. Az ilyen esetekben lehet számunkra hasznos az Ansible harmadik rétege, a `role`. Ez a réteg leginkább feladatszervezésbeli eltérést jelent a playbook-hoz képest, mivel ebben a megközelítésben nem egyetlen fájlba írunk bele minden feladatot, hanem a jól elkülöníthető részfeladatokat külön-külön playbook-okba írjuk le, illetve az ezekhez a részfeladatokhoz kapcsolódó feltölthető fájlokat vagy sablonokat egy egységes könyvtárhierarchiába szervezzük. Erre az 5. ábra mutat példát.



5. ábra Példa egy Ansible apache role könyvtárhierarchiára

A fenti példa egy `apache` role könyvtárhierarchiát mutat be. A `file` alkönyvtárba a kliens gépre feltöltendő fájlok kerülnek, míg a `templates` alkönyvtárba a kliens gépre feltöltendő

sablonok. A **tasks** alkönyvtárba szokott kerülni az a **playbook**, amely az adott role feladatait írja le és ennek **main.yml** lesz az elnevezése. Egy role-t tetszés szerint elnevezhetünk, de a a role könyvtárstruktúrájába tartozó alkönyvtáraknak és fájloknak bizonyos konvenciókat követniük kell [11].

A role-okba szervezett részfeladatok később egy-egy **playbook**-ban összerendezhetjük, így egészen komplex feladatokat is el lehet végezni. Természetesen az egyes role-ok kialakításánál ügyelni kell rá, hogy ezeket úgy írjuk meg, hogy önmagukban és más role-okkal kombinálva is mindig ugyanazt az eredményt adják, azaz idempotensek legyenek.

A következő fejezetben néhány hasznos linket összegyűjtöttem a további információszerezéshez.

4. Hasznos linkek

Ansible	http://www.ansible.com
Ansible forráskód	https://github.com/ansible
Ansible dokumentáció	http://docs.ansible.com/index.html
Ansible könyv	http://www.amazon.com/Ansible-Configuration-Management-Daniel-Hall-ebook/dp/B00GUKM7U0
Ansible playbookok, sablonok, példák	https://galaxy.ansible.com
YAML nyelv	http://www.yaml.org/spec/1.2/spec.html
Konfiguráció menedzselő eszközök	http://en.wikipedia.org/wiki/Comparison_of_open_source_configuration_management_software

Irodalomjegyzék

- [1] Wikipedia: Configuration management
http://en.wikipedia.org/wiki/Configuration_management (2014.06.20.)
- [2] <https://github.com/ansible/ansible/releases> (2014. 06. 20.)
- [3] GNU General Public License: <http://www.gnu.org/copyleft/gpl.html> (2014. 06. 23.)
- [4] Ansible in depth: <http://www.ansible.com/ansible-in-depth-whitepaper> (2014.06.20.)
- [5] The Benefits of Agentless Architecture: <http://www.ansible.com/benefits-of-agentless-architecture> (2014. 06. 23.)
- [6] Michael DeHaan (2014): Windows Is Coming, <http://www.ansible.com/blog/windows-is-coming> (2014. 06. 23.)
- [7] YAML Ain't Markup Language: <http://www.yaml.org/spec/1.2/spec.html> (2014. 06. 23.)
- [8] Jinja2: <http://jinja.pocoo.org/> (2014. 06. 23.)

[9] The BSD 3-Clause License: <http://opensource.org/licenses/BSD-3-Clause> (2014. 06. 23.)

[10] Ansible Documentation: Host and Groups,
http://docs.ansible.com/intro_inventory.html#id5 (2014. 06. 23.)

[11] Playbook Roles and Include Statements: http://docs.ansible.com/playbooks_roles.html
(2014. 06. 24.)

A bilineáris leképezéseken alapuló hibrid mixnet alkalmazása az e-vizsga rendszerekben

Applying of the bilinear pairing based hybrid mix in e-exam systems

Kovács Zita^a

^aDebreceni Egyetem, Informatikai Kar

kovacs.zita@inf.unideb.hu

Absztrakt: Egy e-vizsgáztatási rendszer kidolgozásakor a szakembereknek igen körültekintően kell eljárni a biztonsági kérdéseket illetően. Míg hagyományos esetben például a személy azonosítására már jól kiforrott protokoll alkalmazható – fényképes igazolvány és a személy összevetése -, addig elektronikus esetben ez kriptográfiai építőegységek felhasználásával megoldandó feladat. Igényként jelent meg, hogy se a dolgozat írója, se a dolgozat javítója ne legyen ismert (azaz legyen anonim) a többi résztvevő előtt, így megadva a korrekt és kényszerítéstől mentes értékelés esélyét ezen résztvevők számára. Egy anonimitást megvalósító eszköz a mix network, melyet először Chaum írt le 1981-ben. Ezek hálózatba kötött szerverek, melyek egy üzenet küldőjét rejtik el. Az anonimitás mellett egyéb funkciók jelentek meg: küldhetünk választ az anonim feladó részére, s ha szükséges, akkor a lebonyolító szervezetek közösen felfedhetik az anonim küldők valódi azonosítóját. A mai gyakorlatban egy vizsga esetén szükség van a résztvevők - vizsgázási, javítási - jogosultságának ellenőrzésére is. Feltétel lehet, hogy egy vizsgázó csak egy dolgozatot küldhessen be a határidőig vagy, hogy a vizsgát lebonyolító szervezet üzenhessen a vizsgázónak, (többkörös vizsga, pótlások/javítások kérése). Az eredmény beírásához tudni kell a vizsgázó azonosítóját, még akkor is, ha a vizsgázó nem akar hozzájárulni egy számára előnytelen értékeléshez. Ismerni kell a javító azonosítóját, a munka díjazásához, vagy ha reklamációra kell reagálnia. Az e-vizsga rendszerek készítésekor bevett szokás, hogy a különféle szolgáltatásokat megvalósító primitíveket ollózzák össze a rendszer megalkotói, így azonban biztonsági rések keletkezhetnek, s így támadhatóbb lesz a rendszer, még ha a primitívek igen megbízhatónak és erősnek is számítanak. A bilineáris leképezéseken alapuló hibrid mixnetünk éppen ennek a kiküszöbölésére is készült: úgy nyújt anonim szolgáltatást, hogy lehetőség van a korábban felvázolt egyéb igényeket is kielégíteni, úgy mint a résztvevők jogosultságának ellenőrzése, az anonim üzenetre való válaszolás, az anonim személy azonosítójának felfedése. A mixnet első a hibridek között, mely a kriptográfiailag igen jó tulajdonságokkal bíró bilineáris leképezésekre épít. A hibriditásnak köszönhetően hosszfügetlen üzeneteket küldhetünk hatékonyan.

Kulcsszavak: hibrid mixnet, bilineáris leképezés, e-vizsga

Abstract: When we construct an electronic-exam (e-exam) system we should be taken of the security issues very carefully. While in the case of classic exams we have sophisticated protocols to identify an examinee – comparison of the person and his photo ID -, in the case of e-exams it is a problem to be solved using cryptographic primitives. Sometimes it is necessary that nor the examinee nor the corrector of the exam won't be known ie they are anonymous to the other participants. Because of this we can avoid bribery and subjective evaluation as well. In 1981 chaum proposed a mix network which provide anonymity. A mix network has several proxy server and it hides the sender of the message. After a while other functions have appeared besides anonymity: to send a reply to the anonymous sender, and if it is necessary we can reveal the real ID of the senders. In practical systems there is a need to verifying the eligibility of the participants. Even if the examinee is not cooperative then we still have to enter his bad grade. Another requirment is that for every examinee should send at most one examination paper till the deadline. Sometimes the study committee wants to send a message to the anonymous examinee (multiround exam, supplement/correct request). Often when someone construct an e-exam system then he put together the cryptographic primitives thus formed vulnerabilities regardless how strong was the primitive. The bilinear pairing-based hybrid mixnet – proposed by A. Huszti and Z. Kovacs – eliminates these weak points because the optional functions such as eligibility verification of the participants, reply to an anonymous message, anonymity revocation are built in. This is the first hybrid mix which based on the bilinear

maps. The bilinear maps have really good properties and provide great services for three party protocols. Thanks for the hybrid property we can send efficiently long messages as well as short messages.

Keywords: hybrid mixnet, bilinear pairing, e-exam

1. Bevezetés

Az elektronikus oktatás azaz az e-learning elterjedésével az elektronikus vizsgáztató rendszerek kidolgozása kiemelt feladata a szakembereknek. Különböző igények jelentek meg, magától a vizsgától függően, s emellett a biztonsági követelmények teljesülését is garantálnunk kell. Hagyományos vizsgarendszerek esetén már sok-sok évnyi tapasztalat áll mögöttünk, kiforrott technikákkal rendelkezünk a vizsgafolyamat minden lépéséhez. Az elektronikus vagy e-vizsga esetén azonban szembetaláljuk magunkat a modern kor minden előnyével és hátrányával, így új ötletekkel és megoldásokkal kell előállnunk az egyes lépések megtételéhez, a különböző csalások kizárásához. Vizsgáról van szó mindkét – azaz hagyományos és elektronikus – esetben, azonban a rendszer kidolgozóinak teljesen más problémákkal kell szembenézniük az egyes esetekben. Elektronikus vizsgáztatáskor éppen a vizsga elektronikus mivoltát kihasználva szeretnék a vizsgázók előnyökre szert tenni, jogtalanul jegyet vagy jobb jegyet szerezni. Magyarországon még nem terjedt el, hogy egy e-vizsga az otthonunkból lehetőleg legyen, hiszen akkor a vizsgázó helyett más is ülhetne a számítógép mellett. Amennyiben a vizsga mégis így történik, akkor a bizonyítványt bekérő munkáltató tisztában van ezen vizsga értékével. De gondolhatunk továbbképzésekre is, amikor nem cél tudás nélkül jegyet szerezni, mert azt a tudást utána használnunk kell. Ebben az esetben sem gond, hogy otthonról jelentkezik be a vizsgázó. Ettől függetlenül mi olyan vizsgakörnyezetet képzelünk el, amikor a vizsgázók összegyűlnek egy felügyelt helyen és ott vizsgáznak, azonban elektronikusan. Ilyen például a KRESZ vizsga, de az érettségit is ide gondolhatjuk, hiszen az iskolában letölthető a vizsga, de mégis felügyelet mellett vizsgáznak a tanulók. Természetes igényként merült fel, hogy a vizsgázók ne legyenek beazonosíthatók, s ezért általában név nélkül (másnéven anonimán), egy sorszámmal vagy pszeudonimmal vannak azonosítva az értékelés végéig. Amikor a kapott érdemjegy beírásra kerül, akkor derül ki, hogy melyik érdemjegy kihez tartozik. Ekkor azonban van olyan résztvevője a vizsgafolyamatnak, akiben meg kell bízunk, hiszen bármikor elárulhatná, hogy egy adott pszeudonim kihez tartozik. Vannak olyan e-vizsga rendszerek (pl. [5]), amelyek úgy lettek megvalósítva, hogy senki se legyen megbízható résztvevő azaz megbízható harmadik fél (angol terminológiával Trusted Third Party) nélküliek, mi is erre törekszünk. Érdekes elvárás, hogy a vizsgát javító személyek se legyenek ismertek az értékelés végéig. Mit érünk el ezáltal? Azt, hogy a tanuló nem tudja megkeresni a javítót, hogy megvesztegessen vagy megfenyegetse – közös néven kényszerítse – egy jobb jegy érdekében, hiszen nem tudja, hogy az ő dolgozatát ki fogja javítani. Ez persze akkor működik a valóságban is, ha nem egy tanár javít, hanem több. A biztonsági rendszerek építésekor feltételezzük, hogy a résztvevők minden csoportjában (vizsgázók, javítók) vannak megbízható felek, így ők bejelentenek egy ilyen megkeresést, tehát a tanulóknak nem éri meg találmásra megvesztegetni a javítókat. Azt mondtuk, hogy a tanulók anonimak, nem a saját nevükkel dolgoznak, így, ha meg is találják az ő vizsgadolgozatukat javítót, hogyan mondják meg, hogy melyik dolgozatról van szó? Egyszerű a válasz, odaadják a pszeudonymjuket vagy ha ez nem megoldható, akkor egyszerűen olyan utalást tesznek a dolgozatban, ami által felismerhetővé válik, például az á betűről lebegyja az ékezetet háromszor egymás után vagy duplapontot tesz egy mondat végére. Sajnos előfordul a szubjektív értékelés is, hiszen emberek vagyunk azaz, hogy egyes javítók jobb vagy rosszabb jegyet adnának egy számukra kedvelt vagy nem kedvelt vizsgázónak, már ha tudnák, hogy melyik az ő dolgozata. Azzal, hogy nem ismert a vizsgázó,

objektív értékelés válik lehetővé. Előfordulhat olyan eset is, amikor egy vizsga több körből áll, azaz az első rész elkészítése után kapja meg a második részt a vizsgázó, azaz miután benyújtott egy megoldást, várja a következő feladatsort. Ha ő anoniman vizsgázik, akkor nem tudjuk, hogy kinek kell válaszolnunk. A lényeg, hogy semmilyen azonosításra alkalmas lépés ne történjen a vizsgázás során, tehát nem azonosíthatja a vizsgázót az a számítógép, ahol ül, az első beküldött feladatsora, az üzenetienk egyike sem. A [2]-ben leírt hybrid mixnet megoldja ezt a kérdést, ők megvalósították az *anonim válasz* lehetőségét úgy, hogy rendszerükben nincs megbízható fél. Az ő rendszerük olyan, hogy folyamatos kommunikáció alakulhat ki egy anonim és egy nem anonim résztvevő között. Chaum 1981-ben javasolt egy új kriptográfiai eszközt [4], amelyet *mix network*nek nevezett el, az ő megoldásában egyetlen válaszra volt lehetőség. Lépünk vissza egy kicsit. Mit értünk az alatt, hogy az egyik fél nem anonim. Az egyik fél, aki anonim az vagy a vizsgázó vagy a javító, akiknek ők üzenetet küldenek vagy akitől fogadnak, azok valamilyen hivatalok, akiknek nem szükséges anonimnak lenni, hiszen ők ismertek. Több funkciót is megvalósítanak a hivatalok: az egyik regisztrál, a másik kérdéseket küld, dolgozatokat fogad, értékelésre dolgozatot kiküld, értékelést fogad, végül regisztrálja a jegyet és a javítót. Az utolsó feladat azt is jelenti, hogy szükségessé válik megtudni, hogy kinek írja be az érdemjegyet/értékelést. Tehát egy bizonyos határidő után már az *anonimitást felfedjük*, hogy a vizsga eredményét rögzíteni tudjuk. A javító anonimitását is fel kell fednünk, hiszen a javított dolgozatok számától függ az ő fizetése. Az anonimitás visszahívását szintén megvalósítja a [2]-ben leírt hybrid mixnet. Továbbá [2]-ben belátták, hogy a mixnetük valóban teljesíti az anonimitást, az anonimitás visszahívását, az anonim választ, illetve korrektül működik. A mixnet azt is felkínálja, hogy csak *jogosult felhasználók üzenetét* vegye figyelembe a fogadó fél. Ezt a mixnetet felhasználva - a vizsgadolgozat beküldésétől - biztosíthatjuk, hogy elérjük az általunk is kívánt követelményeket. Némi módosítással azt is garantálhatjuk, hogy a vizsgát *egyszer* lehessen *beküldeni*. Vagy ha a vizsga olyan típusú, hogy a határidőig akárhányszor próbálkozhatnak a vizsgázók, akkor a legutoljára beküldött vizsgadolgozattal tekintjük érvényesnek. Ha jól megnézzük a felépített hybrid mixnet rendszert, akkor láthatjuk, hogy biztosítja a *letagadhatatlanságot* is, azaz a vizsgázó és a javító nem tudja utólag letagadni, hogy az adott üzenetet ő küldte. Ebben a dolgozatban azt a kérdést szeretném felvetni, hogy hogyan lehetne megvalósítani a *globális és egyéni (azaz individuális) ellenőrizhetőséget*. Ezeknek lényege, hogy valaki valamit tudjon ellenőrizni. Azaz globális esetben akár egy külső szemlélődő is meggyőződhet arról, hogy a rendszer jól működik, a beküldött dolgozatokat kijavítják, a jegyet beírják. Az egyéni ellenőrzés az annyit takar, hogy minden vizsgázó, minden javító visszajelzést kaphat, s abból tudhatja, hogy a dolgozata/értékelése beérkezett, valóban az lett értékelve, valóban az az értékelés lett bejegyezve. Dreier és társai 2014-ben megvizsgálták egy hagyományos és egy elektornikus vizsgát az ellenőrizhetőséget helyezve a középpontba [3]. Előtte el is készítették ezt az e-vizsga rendszert, melynek a Remark! nevet adták. Alapvető különbség, hogy más számítási modellen alapul, mint amit a hybrid mixnet használ, illetve az ő esetükben mindenkinek van tanúsítványa. Mi el szeretnénk kerülni ezt az igen költséges megoldást, tehát a széleskörű felhasználhatóságon kívül a költségre is odafigyeltünk. A hibrid mixnet épp ezért alkalmas számunkra, bilineáris párokon alapul, ami olyan költségmegtakarítást tesz lehetővé, mint az elliptikus görbék alkalmazása az utóbbi időben a kriptográfiában. A hibrid elnevezés mögött egy igen fontos tulajdonság bújjik meg: tetszőleges hosszúságú üzenetet lehet hatékonyan kezelni, küldeni. Mindegy, hogy az üzenet egy szóból áll, vagy hosszabb kidolgozás, mindkét esetben a lehető legtöbb költség megtakarításra kerül. Általában egy rövid üzenet küldésekor kiegészítik “szeméttel” az üzenetet, hogy akkora legyen, amekkorát a rendszer megkíván, a hibrid esetben ilyesmire nincs szükség.

2. Alapfogalmak

2.1. Résztevők

Az elektronikus vizsga résztvevői az alábbiak.

A **vizsgáló (V)** az a résztvevő, aki a tudásáról számot ad.

A **javító (J)** az a résztvevő, aki a vizsgáló által készített vizsgadolgozatot értékeli. A válaszok azon részét javítja ki, amely nem javítható automatikusan, szoftver által, például egy vizsga esetén fogalmazás javítása.

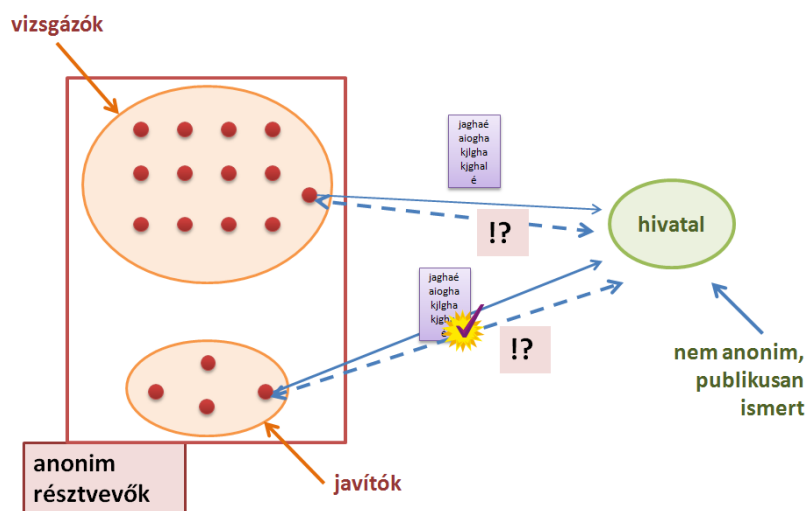
A **regisztrációs hivatal (R)** feladata, hogy a vizsgára jelentkezők és a javításra jelentkezők regisztrációját elvégezze, továbbá nyilvántartsa a rendszer jogosult felhasználóit. Ő ad aláírást, mellyel igazolja, hogy az adott üzenetet valóban egy jogosult felhasználó akarja elküldeni.

A **tanulmányi hivatal (T)** végzi a vizsga lebonyolítását, kiküldi a kérdéseket, összegyűjti a vizsgadolgozatokat, amelyeket javításra továbbít, majd begyűjti az értékeléseket. Szükség esetén kommunikál a vizsgálóval, javítóval. Rajta keresztül tud kommunikálni a vizsgáló és a javító is.

A **hirdetőtábla (HT)** egy olyan felület, amely bárki számára információkat szolgáltat, melyeket az arra jogosultak helyeznek el rajta. Az ellenőrzéseket a hirdetőtáblán lévő információk alapján tudják elvégezni az illetékesek.

A **mixszerverek (M_1, \dots, M_N)** olyan szerverek, melyek az anonim küldés szolgáltatást valósítják meg. Rajtuk keresztül történik a kommunikáció, miközben az anonim felhasználók kilétét elrejtik. Az anonim felhasználók mindig valamelyik publikusan ismert hivattal kommunikálnak, amely az utolsó mixszerver szerepét is betölti. A mixszerverek vesznek részt az anonimitás felfedésében is, a határidő lejárta után, közösen a tanulmányi hivattal.

A vizsgálók és a javítók anonimak, a hivatalok nem anonimak. A vizsgálók és a javítók az tőlük elvárt üzeneteken (vizsgadolgozat, értékelés) kívül feltehetnek kérdést is egymás számára, amelyet a hivatalon keresztül küldenek el. Ezáltal kialakulhat egy ellenőrzött kommunikáció is, amennyiben a vizsga olyan, hogy erre lehetőséget kapnak a résztvevők. A résztvevők közötti kommunikációt az alábbi ábra szemlélteti (lásd 1. ábra). Amennyiben az *anonimitást* is igénybe vesszük, akkor az üzenetek a *mix szervereken keresztül* közlekednek.



1. ábra. E-vizsga résztvevőinek kommunikációi

2.2. Biztonsági kritériumok

Titkosság: A vizsgadolgozat, illetve az értékelés titkosak. Fontos, hogy bizalmas információk ne kerüljenek illetéktelenek kezébe.

Anonimitás: A vizsgázók és a javítók anonim módon vesznek részt a vizsgafolyamatban, kilétük nem ismert. Ezzel elérjük, hogy kényszerítésmentes, objektív értékelések történjenek.

Hitelesség: A kérdések, a válaszok és az értékelések hitelesek. Nagyon fontos elvárás, hogy biztosítsuk a hitelességet, vagyis, hogy a küldött információ a küldés során nem változott, illetve az információ valóban a küldőtől származik.

Egyszeri beküldés: Egy beküldött vizsgadolgozatot tekintünk érvényesnek. A vizsgázás során elkészített dolgozatot egyszer küldheti be a vizsgázó, ha többet is beküld a megadott határidőig, akkor az utoljára beküldöttet veszi figyelembe a rendszer, amennyiben olyan a vizsga menete.

Jogosultság: Csak az arra jogosultak vehetnek részt a vizsgán. Csak jogosult vizsgázó küldhessen be dolgozatot és csak jogosult javító küldhessen be értékelést. Külön érdekesség, hogy az anonimitás és a jogosultság együttesen kell teljesüljön, s ennek megvalósítása a regisztrációs hivatal által végrehajtott bilineáris vak aláírás által történik.

Letagadhatatlanság: A vizsgadolgozat és értékelés beküldője ne tudja letagadni tettét, azaz, hogy ő volt az, aki az adott dokumentumot elküldte.

Egyéni és globális ellenőrizhetőség: Fontos követelmény, hogy minden résztvevő egyénileg tudja ellenőrizni, hogy az általa beküldött dolgozat/értékelés megfelelően lett-e feldolgozva, beleértve, hogy a vizsgázó, illetve a javító visszajelzést kap az adatok sikeres beküldéséről. A globális ellenőrizhetőség teljesül, ha egy külső megfigyelő is meggyőződhet arról, hogy szabályosan zajlott-e le a vizsga.

2.3. Bilineáris párok, hibrid mixnet

A [2]-ben leírt bilineáris párokon alapuló hibrid mixnetet és az általa nyújtott szolgáltatásokat alkalmazzuk az e-vizsgára, a mixnet az alábbi definícióra épít.

Definíció: Legyenek G_1 és G_2 két q rendű csoport, ahol q egy nagy prímszám. Egy $e : G_1 \times G_1 \rightarrow G_2$ leképezést **elfogadható bilineáris leképezés**-nek nevezünk, ha teljesíti az alábbiakat:

- *Bilineáris:* Egy $e : G_1 \times G_1 \rightarrow G_2$ leképezés bilineáris, ha $e(aP, bQ) = e(P, Q)^{ab}$ minden $P, Q \in G_1$ és minden $a, b \in \mathbb{Z}_q^*$ esetén.
- *Nem-degenerált:* A leképezés nem minden $G_1 \times G_1$ -beli párhoz rendel G_2 -beli elemet identikusan. Mivel G_1, G_2 prímrendű csoportok, ha P generátora G_1 -nek, akkor $e(P, P)$ generátora G_2 -nek.
- *Számítható:* Létezik hatékony algoritmus, mely kiszámítja $e(P, Q)$ -t minden $P, Q \in G_1$ -re.

A Weil és Tate pairing bizonyítja ilyen konstrukciók létezését. Tipikusan, G_1 egy elliptikus görbe csoport és G_2 egy véges test. Az ilyen leképezések bilineáris tulajdonsága adja a hibrid mixnetben a számítások, ellenőrzések alapját. Ezek a leképezések nagyszerű lehetőséget biztosítanak a háromrésztvevős protokollok kulcsmegosztásához, a három résztvevő titkos kommunikációjában. A 2000-es évek elejéig a Weil párokat elliptikus görbe rendszerek támadására használták, azonban 2000-től egyre több rendszerben ismerték fel a párok alkalmazásának előnyeit, például Joux, aki egy egykörös háromrésztvevős Diffie-Hellman protokoll elkészítésére használta [6]. Az egyes mixnetek különböznek például abban is, hogy

a mix szerverek milyen kriptográfiai műveletet végeznek, azaz titkosítanak, visszafejtenek, újratitkosítanak, vagy abban, hogy szimmetrikus vagy aszimmetrikus kriptográfiát alkalmaznak. A hibrid mixnet azért hibrid, mert épít mindkettőre, ezáltal éri el azt, hogy a tetszőleges hosszú üzenetek költséghatékonyan küldhetőek legyenek. Nekünk ez éppen megfelelő, mert vizsgáktól függően többféle hosszúságú üzenetet küldenek a résztvevőink. A dolgozat is lehet egészen kicsi, ha csak tesztről van szó, de kifejtős is, ami akár egész hosszúra nyúlhat. Az értékelés is lehet akár egy darab érdemjegy, míg az is lehet, hogy egy hosszabb szöveges értékelést készít és küld a javító.

3. Az e-vizsga

Tipikusan a vizsgák és az e-vizsgák lépéseit fázisokkal adják meg. A következő részben a javasolt e-vizsga **vizsgázó** szemszögéből tekintett fázisait írtuk le. A vizsgázó valamilyen képzésének végén (pl: önképzés, tanfolyam, egyetem) vizsgára jelentkezik. Ezt egy adott vizsga esetén megteheti például úgy, hogy kitölt egy formanyomtatványt, jelentkezik online. Egyszerű esetben a vizsgadíj befizetésekor máris jogosulttá válik a vizsgára. Személyes megjelenésével igazolja magát a regisztrációs hivatalnál, ezzel bekerül a vizsgára (illetve vizsgáztatásra) jogosultak adatbázisába. Ezután a vizsgázó megjelenik a korábban felvett vizsga időpontjában a felügyelt teremben, igazolja magát és helyet foglal. A kérdéseket megkapja, s elkezd dolgozni.

Előkészítő fázis Ebben a fázisban történnek meg az alábbiak: a rendszer nyilvános paramétereinek legenerálása, egyes résztvevők nyilvános-titkos kulcspárjainak legenerálása, a mix szerverek és a tanulmányi hivatal által megosztott kulcsok legenerálása, a vizsgadolgozat elkészítése (megj: a vizsga kérdéssorát valamilyen módon, akár nyilvánosan letölti a vizsgázó, majd elkészíti a dolgozatát, ehhez még nem szükséges az anonimitásnak teljesülnie)

Regisztrációs fázis A vizsgázó elkészült a dolgozatával, amit szeretne beküldeni a tanulmányi hivatal részére. Ehhez szüksége van egy hitelesítésre, amelyet a regisztrációs hivaltól tud megszerezni. A bilineáris vak aláírást használva eléri, hogy a vizsgadolgozatának kivonatán a regisztrációs hivatal aláírása szerepeljen.

Beküldés és keverés fázisa A vizsgadolgozat beküldéséhez a vizsgázó kiszámolja a szükséges értékeket és elkészíti a hibrid mixnet leírásában megadott üzenetét, majd elküldi azt az első mix szervernek. Az első mix szerver begyűjti a vizsgákat, majd ahogy a többi mix szerver kever és számol, majd továbbküldi az üzenetet a következő mix szervernek, végül utolsóként a tanulmányi hivatal fogja megkapni a vizsgázók üzenetét.

Fogadás fázisa Az utolsó mix szerver a tanulmányi hivatal, amely szintén végez számításokat, hogy megkapja a vizsgázó üzenetét. Ezután két ellenőrzést végez, melyekkel megbizonyosodik róla, hogy a vizsgázó jogosult volt beküldeni a dolgozatot, illetve, hogy az anonimitás felfedése megvalósítható lesz-e. Amennyiben a válaszok pozitívak, akkor a jogosult vizsgázóhoz bejegyzések kerülnek, az üzeneteit és néhány kiszámolt értéket tárolja el a tanulmányi hivatal.

Anonim válasz fázisa (opcionális) Ha további körök következnek, vagy valamilyen kérdés merült fel, vagy pótolnia kell valamit a vizsgázónak, akkor a tanulmányi hivatal válaszüzenetet fogalmaz meg, melyet az egyik tárolt érték alapján a mixszerverek kézbesíteni képesek az anonim vizsgázónak. Ez akár igény szerint többször is megtörténhet.

Anonimitás visszahívása A vizsga (vizsgázás és értékelés) végeztével szükség van az értékelés rögzítésére, így az anonim vizsgázó kilétét felfedi közösen a tanulmányi hivatal és a mix szerver. Ha a vizsgázó együttműködik, akkor lehetőség van költségmegtakarításra, mert ebben az esetben jóval gyorsabban és hatékonyabban felfedhető a valódi azonosító.

Láthattuk, hogy a fázisok során mindvégig a vizsgázóra összpontosítottunk. A **javító** esete is pontosan így zajlik. A tanulmányi hivatal a vizsgadolgozatok értékelését megkezdi egy adott időpontban, vagy akár rögtön a vizsga után. A jogosult vizsgázókat nyilvántartja a regisztrációs hivatal. A vizsgajavításkor a tanulmányi hivatal valamilyen véletlen sorsolással kiosztja a dolgozatokat a javítók között (akárcsak a vizsgázók esetén a kérdéseket), akik elvégzik a javítást. Az értékelés beküldéséhez futtatják az előbb leírt *regisztrációs és beküldés és keverés fázisokat*, így eljuttatva az értékelést a tanulmányi hivatalnak, anonim módon. Ha már nem merül fel egyik oldalról sem kérdés és nem hamarabb, akkor elindulhat az *anonimitás visszahívása fázis*, mind a vizsgázók, mind a javítók személyére vonatkozóan.

4. Biztonsági értékelés

Az *egyszeri beküldés* és az *egyéni és globális ellenőrizhetőség* kritériumok kivételével az összes többi biztosítja az alkalmazott hibrid mixnet. Ezért a két kivétellel fogunk foglalkozni.

Az egyszeri beküldés követelményét úgy tudjuk teljesíteni, hogy a tanulmányi hivatal egy vizsgadolgozat fogadásakor ellenőrzi, hogy adott vizsgázó nem küldött-e már be dolgozatot. Ha többszöri is küldhet be dolgozatot, melyek körül az utolsó az érvényes, akkor felülírjuk a tárolt dolgozatot. Mivel a vizsgázó hitelesítést egyszer kérhet a regisztrációs hivataltól (ezt a hivatal tudja kezelni, hiszen a hitelesítéshez elküldi a saját azonosítóját a vizsgázó), így a μ_i érték azonosítja a dolgozatot, vagyis a tanulmányi hivatal ki tudja keresni, hogy szerepel-e már ilyen érték az adatbázisában, azaz a vizsgázó küldött-e már be dolgozatot. Tehát ezen követelmény teljesülését a hibrid mixnet protokolljának egy kis módosításával elérhetjük, ha a tanulmányi hivatal ezt a plusz ellenőrzést beiktatja.

Az egyéni és globális ellenőrizhetőség teljesüléséhez a [3]-ban leírt fogalmakat tekintem alapul. Az egyéni ellenőrzést a vizsgázó szempontjából tekintjük, s az alábbiakat jelenti. Minden vizsgázó szeretné ellenőrizni, hogy a vizsgadolgozata korrektül lett értékelve. Ez jelenti azt, hogy a neki kiküldött kérdések hitelesek, az őáltala beküldött dolgozatot elfogadta a tanulmányi hivatal és azt értékelésre megkapta egy javító, értékelte a dolgozatát egy javító, a kapott értékelés korrekt, a dolgozatra adott értékelést valóban a dolgozat készítője kapja meg, a vizsgázót értesítik a kapott értékelésről. A globális ellenőrizhetőség egy külső résztvevő által történik, s az alábbiakat takarja. Csak jogosult vizsgázók által beküldött dolgozatokat tárolja a rendszer, minden jogosult vizsgázó által beküldött dolgozat és csak ezek vannak kiküldve javításra, valamint ezek vannak értékelve, csakis ezek a dolgozatokra kapott jegyek vannak figyelembe véve. Ezen ellenőrizhetőségek (egyéni, globális) teljesülését többféle módon tudjuk belátni. A [3]-ban Proverif technikát alkalmazva szeretnénk megvizsgálni a mi esetünkben mennyire teljesülnek, illetve milyen módosítással teljesülnének ezek a feltételek. Jelenleg még annyit mondhatunk, hogy az egyéni ellenőrzések némi módosítás után könnyen teljesülnek, úgy mint a tanulmányi hivatal a vizsgadolgozat beérkezésekor egy visszaigazolást küld a hirdetőablára, aminek a segítségével a vizsgázó megbizonyosodhat róla, hogy a dolgozat beérkezett, stb. Ezen megoldásokat bemutattuk már egy korábbi dolgozatunkban, 2011-ben [1]. A hibrid mixnet működése olyan, hogy biztosítja

a jogosultság folyamatos ellenőrzését, ezért a globális ellenőrizhetőség egyes részei szintén teljesülnek. Ennek a formális leírása, Proverif alkalmazás a következő feladatunk, terveink között szerepelnek.

5. Összefoglalás

A dolgozatban felvázoltunk egy olyan e-vizsgát, amely a [2]-ben leírt bilineáris leképezéseken alapuló hibrid mixnetet alkalmazza a dolgozatok és értékelések beküldéséhez, aholis szükség van a küldő felek anonimitásának biztosítására. A mixnet által biztosított szolgáltatások éppen azok, amelyekre az e-vizsga rendszerekben is szükség van. Nami módosítással mindenféle egyéb speciális igény (egyszeri beküldés, ellenőrizhetőségek) teljesülését garantálhatjuk. Az egyéni és globális ellenőrizhetőséget részletesebben körül fogjuk járni, mivel vannak már ehhez használatos módszerek, azokat alkalmazzuk. A [3]-ban leírt e-vizsga hasonlóan működik, viszont a miáltalunk bemutatott rendszer költségmegtakarítást is lehetővé tesz. Terveink között szerepel ebből a szempontból történő összehasonlításuk is.

6. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg. Továbbá szeretnék köszönetet mondani Dr. Huszti Andreának kutatásaink során tett segítő megjegyzéseiért, munkájáért és odaadó támogatásáért.

Irodalomjegyzék

- [1] Huszti Andrea és Kovács Zita: Univerzális kriptográfiai protokoll e-felmérésekhez, *Informatika a felsőoktatásban Konferencia Kiadványa*, Debrecen, (2011), 592--600.
- [2] Huszti Andrea és Kovács Zita: Bilinear Pairing-based Hybrid Mixnet with Anonymity Revocation, *(megjelenés alatt)*
- [3] J. Dreier, R. Giustolisi, A. Kassem, P. Lafourcade, G. Lenzini: On the Verifiability of (Electronic) Exams, *Verimag Research Report n^o*, Grenoble, (2014)
- [4] David Chaum: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms, *Commun. ACM* 24(2), (1981), 84-88.
- [5] Huszti Andrea és Pethő Attila: A secure electronic exam system, *Publ. Math.* 77/3-4 Debrecen, (2010), 299-312
- [6] A. Joux: A One Round Protocol for Tripartite Diffie-Hellman, *Algorithmic Number Theory, 4th International Symposium, ANTS-IV Leiden, The Netherlands, July 2-7, 2000, Proceedings, LNCS 1838*, (2000) ,385-393

Védekezés az RFID rendszerekben a lehallgatás ellen

Defending in the RFID system against eavesdropping

Radványi Tibor^a

^aEszterházy Károly Főiskola
radvanyi.tibor@ektf.hu

Absztrakt: Ebben a cikkben a napjainkban legdinamikusabban fejlődő, az automatikus azonosítás témaköréhez tartozó RFID technológiával foglalkozunk. Bemutatjuk, hogy milyen lehetőségek vannak ezen technológia támadására hangsúlyosan az UHF és a HF/NFC frekvencia tartományokra fókuszálva. Bemutatjuk, hogy milyen lehetőségek vannak a védekezésre. Kiemelt hangsúlyt teszünk a lehallgatás elleni védekezés elemzésére és a védekezés lehetőségeire. Kitérünk rá, hogy melyek azok a kriptográfiai algoritmusok, melyek használhatók és melyek azok, melyek a technológia korlátai miatt nem. Nagyon erős szelektivitással rendelkezik ebben a kérdésben az RFID bélyegek (tagek)-ben implementált memória nagysága és a lehetséges számítási műveletek végrehajtására képes chip hiánya vagy megléte. A passzív UHF és HF/NFC tag-ek nem tartalmaznak számítási kapacitással rendelkező kontrollert, csak az adatok tárolására alkalmas memória chippet. Ebben az esetben a védekezés lényegesen nagyobb problémát okoz, mint az aktív eszközök esetén, ahol a rendelkezésre állnak a számítási feladatok ellátására alkalmas hardver eszközök. Ellenben ezek a transzponderek akár két nagyságrenddel is olcsóbbak lehetnek az aktív változathoz képest. Várhatóan az európai unió ajánlását követve az „Internet of Things” irányvonal mentén az előttünk álló években több tízmilliárd transzponder kerül felhasználásra, és ezek nagyrészt passzív tag-ek lesznek. Ezzel egyre nagyobb igény generálódik a védendő adatok felhasználására és forgalmazására, holott a védelmük nem, vagy csak nehezen megoldható.

Kulcsszavak: RFID, adatbiztonság, kriptográfia, adatvédelem, UHF, HF/NFC

Abstract In this article we are going to deal with today’s most dynamically improving RFID technology connected to the topic of automatic identification. We are going to introduce the possible attacks, emphasizing the UHF and the HF/NFC frequency ranges. Different defend methods are also going to be mentioned. Analysing the possibilities of defeating eavesdropping will be highlighted as well. The suitable cryptographic algorithms are going to be listed besides the ones which are non-suitable according to the boundaries of the technology. The size of the implemented memory and the persistence or non-persistence of a chip which is able to solve the possible calculation operations in the RFID tag strongly influences this issue. Passive UHF and HF/NFC tags do not contain a controller with calculation capacity, only a memory chip to store data. In this case defense means a reasonably bigger problem than if we are working with active devices where hardware tools able to serve calculation purposes are available. On the other hand these transponders might be extremely cheaper than the active ones. In the next few years tens of billions of transponders are expected to be installed according to the European Union’s “Internet of Things” plan, and most of these transponders are going to be passive tags. This project is going to generate a large demand for using and distributing data that should be protected, in spite of the fact that their protection is not solvable or very hard to solve.

Keywords: RFID, data security, cryptography, data protection, UHF, HF/NFC

1. Bevezetés

Napjainkban széles körben megtalálhatóak az azonosítási rendszerek különböző formái. Egy olyan kód- és kommunikációs rendszert értünk ez alatt, amely személyeket, tárgyakat, eseményeket egyedileg azonosít. A legfiatalabb és legdinamikusabban fejlődő azonosítási módszer az RFID. Különböző típusú szenzorokkal illetve helymeghatározó rendszerekkel párosítva széles felhasználási területen alkalmazható. Találkozhatunk vele az autógyártásban,

logisztikában, gyógyszer- és hadiiparban valamint számos más helyen is. Lehetővé teszi a közúti, légi és vízi szállítás teljes nyomon követését, és nem utolsó sorban ellenőrizhetővé válik annak minőségi állapota a szállítás folyamán. A technológia azonosítási- és biztonsági lehetőségeit egyre inkább kihasználják a modern útlevelek, a digitális azonosítók és nem utolsó sorban még a legújabb fizetési megoldások is. [1][2]

Egy termék számtalan veszélynek van kitéve, ameddig a gyártótól el nem jut a fogyasztóhoz. A gyárból átkerül egy átmeneti raktárba, innen a nagykereskedő, majd a kiskereskedelmi cég elosztó központjába, végül pedig az áruházak polcaira. Ez elég hosszú folyamat, amely során az áruk elveszhetnek, összecserélődhetnek, ellophatják őket.

Magyarországon jelenleg is folynak az egyeztetések a mobillal való fizetési lehetőségekről, melynek elterjedése egy hatalmas mérföldkő lehet a fejlődésben. A felhasználók nincsenek tisztában ennek veszélyeivel, legtöbbjük nem tud, vagy nem is akar foglalkozni ehhez hasonló problémákkal. Így a gyártóknak folyamatosan figyelniük kell a biztonságra, figyelemmel kell követniük azokat a lehetőségeket, melyek bármilyen sérülést, vagy adatlopást tudnánk okozni a felhasználót körülvevő rendszerekben. A csökkenő előállítási költségek révén az olcsó passzív RFID rendszerekre jellemző adattárolási limit is előbb vagy utóbb megszűnik. Felválthatják az aktív címkék, melyek már sokkal nagyobb biztonsággal használhatóak, és nem kell speciális algoritmusokat kidolgozni, hogy működni tudjanak az egyszerűbb rendszereken is.

Végül is hamar belátták azt, hogy első szempont mindig az információkezelés biztonságos és akadálymentes kezelése legyen, s a hatékonyságot ezzel a háttérbe szorították. Véleményünk szerint is a legfontosabb az adatok biztonságban tartása, főként az olyan rendszerek esetében, ahol nélkülözhetetlen a titoktartás. Pl. egy banki szolgáltatás inkább legyen lassabb, és biztonságosabb, mint legyen gyors. [3][10]

2. Támadás az RF interface-en keresztül

Az RFID rendszerek elleni egyik jellemző támadási módszer az RF interfészen keresztül érkező támadás. Az RFID rendszerek rádió rendszerek és elektromágneses hullámok segítségével kommunikálnak közelre és távolra egyaránt. Így a támadónak lehetősége van a rádiófrekvenciás interfészen keresztül is támadást indítani, mivel nincs szükség az olvasó vagy a transzponder fizikai hozzáféréshez. Ennek a támadástípusnak számos alete ismert, a következőkben ennek bemutatására teszünk kísérletet. [11,12,13,14]

Nagy hatótávolságú RFID rendszernek nevezzük azt, melyben a két eszköz közötti távolság nagyobb, mint 1 méter. Általában UHF (868 vagy 915 MHz) vagy mikrohullámú frekvenciatartományban (2,5 vagy 5,8 GHz) működnek. Ha a tag kikerül a rendszer olvasási tartományából, szintén két lehetőség merül fel az adás megszakítására. Az egyik ok, hogy a tag nem kap elég áramot az antennából a működéshez. A másik lehetőség pedig, ha a visszavert teljesítmény kevés ahhoz, hogy az olvasó érzékelti tudja. A távolság növeléséhez emelni kell az olvasó átviteli teljesítményét. Ahhoz, hogy az olvasási távolság a kétszeresére nőjön az olvasó teljesítményét a négyszeresére kell emelni. Ha szeretnénk megtartani a visszaszórás teljesítményének mértékét kétszeres olvasási távolságon, akkor az olvasó teljesítményét már tizenhatszorosára kellene növelni. 2005-ben sikerült a Yagi-Uda antennával 21 méterről sikeres támadást végrehajtani. [4][10]

3. A lehallgatás (eavesdropping)

A kommunikáció lehallgatása az olvasó és a transzponder között történik. Az RFID rendszerek hatótávolsága pár centimétertől (pl.: 13.56 MHz) több méterig terjedhet (pl: 868 MHz). Finke and Kelter megállapították, hogy 13.56 MHz-es induktív csatolású rendszer, akár 3 méterről is lehallgatható.[4] A vevő pár kHz-es sávzélességen az olvasó modulálatlan jelét több 100 méterről is érzékelheti. Nagyobb távolságnál a jelet zavarhatják fém tárgyak, mint például kerítések, alumínium tárgyak, de akár nagyobb épületek is torzíthatják. Mitől függ, hogy a támadó sikeresen lehallgassa az eszközeink (olvasó és transzponder) közötti kommunikációt? A befolyásoló tényezők száma nagyon nagy. [7][8]

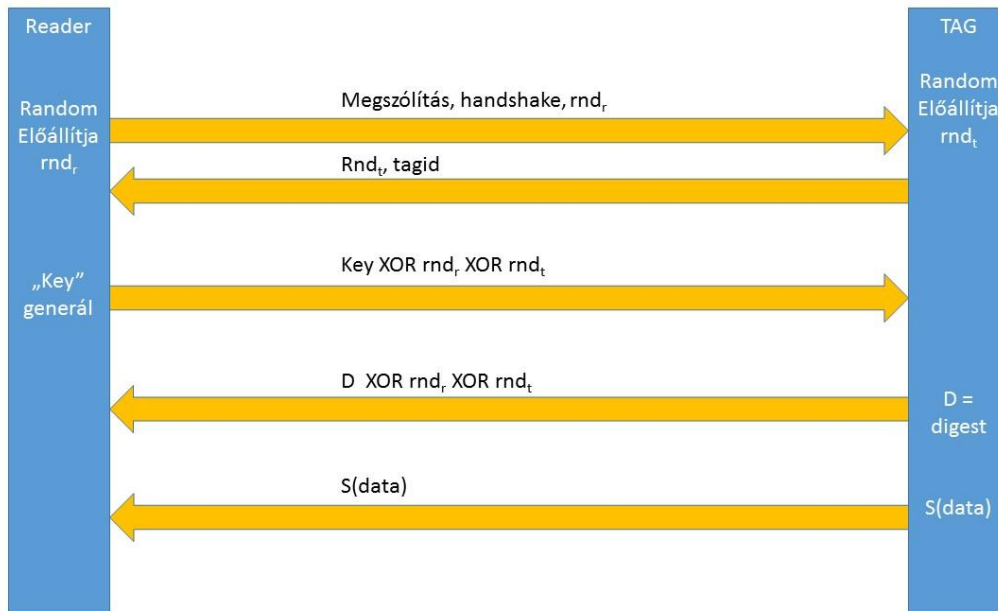
- Függ a RF tér karakterisztikájától. Ezt az olvasó antennájának geometriája, belső felépítése és a kibocsátási energia határozza meg.
- Fontos tényező az olvasó és a transzponder közötti térben elhelyezkedő zavaró tárgyak, fém felületek mérete, elhelyezkedése.
- Befolyásolja a támadó lehallgató eszközének minősége, felépítése, geometriája. Nagyban függ az olvasó által kibocsátott, generált tér energiájától.
- Fontos befolyásoló tényező, hogy az RF kommunikációban passzív vagy aktív transzponderek vesznek részt. Ha a tag passzív, akkor az olvasó által gerjesztett tér energiáját használja, így a visszasugárzott, hasznos információ alacsonyabb energiával vesz részt a kommunikációban. Ez az UHF tag-ek esetében (868 MHz – 915 MHz) 1-3 méter. Amennyiben a tag aktív, esetleg szemipasszív, azaz rendelkezik saját energiaforrással, az e távolság megnőhet akár 10-30 méterre is. Aktív esetben a tag által kisugárzott információ nagyobb energiája miatt könnyebben fogható a támadó eszközeivel. Illetve ez az eszköz a nagyobb támadási térben jobban rejtve maradhat. Támadási térnek nevezzük azt a térrészt, ahol a támadó adott jellemzőkkel rendelkező lehallgató eszközt elhelyezve, még sikeres támadást tud végrehajtani.

A lehallgatás során az adatokat a következő támadások érhetik:

- Titkos vagy személyes adat kerül illetéktelen kézbe. Ekkor a kommunikációra nincs hatással a támadó. A jelenlétét nagyon nehéz, ha nem lehetetlen felfedezni. Ebben az esetben az adatok közvetlen védelme, kriptográfiai protokollok használata segíthet.
- A lehallgatott adatot a támadó megváltoztatja és a megváltoztatott, hamis adat jut el az olvasóhoz. Nehezen megvalósítható, komoly támadó eszközt feltételező támadás. Az adat módosításhoz szükséges időablaknak be kell férnie a kommunikációs adatcsere folyamatába.
- Egy másik lehetőség, hogy ha a támadó nem módosítja a lehallgatott adatot, hanem a helyére új adatot szűr be. Ez akkor lehetséges, ha a transzponder viszonylag sok adatot küld az olvasónak. Így a kommunikációra sok idő szükséges. A fenti támadási módok esetén a támadót felfedezhetik, az adatait kivédhetik. Ellenőrző összegek használata és a kriptográfiai algoritmusok és protokollok kombinálása lehetővé teszi ezt.
- A lehallgatásos támadás egy bonyolultabb, komoly technikai felkészülést igénylő módja a „relay attack” támadás. Ebben az esetben a támadó nem csak lehallgatja a kommunikációt, hanem a megszerzett adatokat egy másik csatornán pl. wifi nagyobb távolságra továbbíthatja. A másik helyen egy eszközzel az adatok felhasználásra kerülhetnek pl egy vásárlás során. Ezt a támadási módot nehéz kivédeni az érintésmentes fizetési lehetőségek tulajdonságai miatt. Egyelőre jó lehetőséget biztosít az egyéb azonosítóval való kiegészítése a folyamatnak. A legegyszerűbb a pin kód használata, de bármely személyhez, helyhez kötött biometrikus azonosító is kiváló lehet. [15][16]

Láthatjuk, hogy a lehallgatás bizonyos esetekben könnyen elvégezhető támadási mód. Viszonylag sok lehetőséget tartalmaz a támadó számára és elég nehéz észlelni és kivédeni.

Az adatok védelme érdekében, ha nem tudjuk megvédeni a kommunikációs csatornát, akkor az esetlegesen lehallgatott információt tegyük nem vagy csak nehezen felhasználhatóvá a támadó számára. Ehhez nyújt segítséget, ha kriptográfiai protokollt használunk az információváltás során.



1. ábra. reader és transzponder kommunikáció

4. Biztonsági lépések

A rejtjelezés alapvetően a passzív támadások ellen véd, az aktív támadások elleni védekezéshez kriptográfiai protokollokat használunk, ami előre meghatározott üzenetcsere-folyamatot jelent. Ennek során észleljük az aktív támadásokat, és kivédjük azok káros következményét.

A publikált protokolloknak sok közös vonásuk van. [6] Fő lépéseik:

1. Az olvasó kérést sugároz a tag-nek
2. A tag azonosítja magát az olvasónak (megadja a tárolt adatokat)
3. Az olvasó továbbítja az adatokat a háttér szervernek
4. A szerver adatbázisa alapján feldolgozza az adatokat
5. A szerver elküldi a hitelesítést és a feldolgozott adatot

A különbség a különböző szinteken kriptográfiai primitívek alkalmazásában van. [5] A tag hash-seli az adatokat mielőtt továbbítja az olvasónak. A háttér szerver a közös kulccsal visszafejti az üzenetet, adatbázisában megkeresi és feldolgozza azt.

A lehallgatás ellen az 1, 2 pontok által leírt folyamatot kell részleteznünk és erősíteni. Lehetséges az erős védelmi rendszerbe bevonni a háttér szervert is. Így két lehetőségünk adódik. Az egyik, amikor a kriptográfiai protokoll mindhárom réteget érinti, azaz a transzpondert, az olvasót és a háttérszervert is. A másik lehetőség, hogy próbáljuk a tag és az olvasó közötti kommunikációra szorítani a védelmet, feltételezve, hogy az olvasó és a háttérszerver közötti, többnyire belső, védett hálózatban futó adatforgalom már biztonságban van.

Természetesen erősen függ a követető protokoll attól, hogy a kommunikációban passzív vagy aktív tag-ek szerepelnek. Már a meglévő követelmények is eltérnek, és a rendelkezésre álló számítási kapacitás is jelentős különbséget mutat.

Tekintsük át az 1. ábrán is szemléltetett kommunikációs sémát.

Látható, hogy a titkosításhoz egyrészt XOR függvényt használunk, ami könnyen implementálható hardver szinten is, így a passzív tag-ben nincs akadálya a használatának.

A XOR protokoll különböző kulcsokat használ, különböző irányban. [17,18]

$$\begin{aligned} R \rightarrow T : x \oplus k_1 \\ T \rightarrow R : x \oplus k_2 \end{aligned}$$

Biztonságos megoldás, hogy a k_1 és k_2 kulcsokat véletlenszerűen választjuk minden egyes futtatáskor. Az egyik lehetőség ennek megvalósítására a XOR kulcsgenerálás, melyben i változó alapján R véletlenszerűen választ új $k(i)$ kulcsot és XOR titkosítást hajt végre a $k(i-1)$ kulccsal. Ezáltal a következő protokollt kapjuk:

$$\begin{aligned} R \rightarrow T : a(i) = x(i) \oplus k(i), k(i) = k(i-1) \\ T \rightarrow R : b(i) = x(i) \oplus k(i) \end{aligned}$$

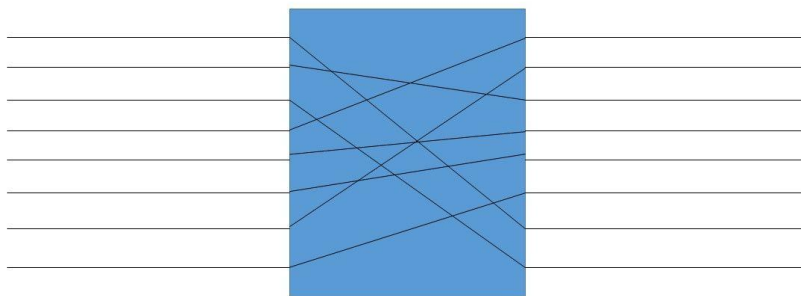
ahol $i = 2, 3, \dots$ egy számláló, minden futásnál egyel növelve. $x(i)$ az i -edik véletlenszám és $k(0)$ és $k(1)$ előre beállított osztott kulcsok. $k(1), k(2), \dots$ sorozat nem változik véletlenszerűen, csak az értéküket nem tudja követni a támadó.

Valamint megjelenik egy S függvény, amit kicsit részletezni szükséges.

Először tekintsük az úgynevezett P és S dobozokat. Ezek alapjait képezik a kriptográfiai algoritmusoknak. Előnyük, hogy elektrotechnikailag könnyen megvalósíthatóak. Így a passzív tag-ek rendkívül korlátozott eszközkészletéhez is integrálhatóak lesznek. Aktív tag-ek esetén ez nem jelent gondot, hiszen a tag tartalmaz intelligenciát, azaz programozható processzort, így a teljes AES algoritmus megvalósítható viszonylag kis energia ráfordítással és rövid idő alatt.

A passzív tag-ek esetén használjuk a P és S dobozok kombinációját.

A P doboz egy 8 bit bemenő adatból 8 bit kimenő adatot előállító függvény. Egy gyors és egyszerű elektronikai eszköz, mely ha ismerjük a P doboz hozzárendelési szabályát, akkor az inverz függvény is elkészíthető. A feladata a 8 bit valamilyen keverése, egy bitpermutáció előállítása.



2. ábra. egyféle P doboz leképezés

Az S dobozok egy 6 bemenő bitből 4 bitet előállító nem lineáris függvényt megvalósító eszközök. [19] Az S dobozok működését egy 4 sorból és 16 oszlopból álló táblázat írja le. Minden egye S doboznak más és más a táblázata. Ezek felhasználásával lehetséges az S dobozok kódolása. A bejövő 6 bit első és hatodik bitje adja a sorindexet, míg a 4 középső

bitnek megfelelő decimális szám az oszlopindexet. Így kapjuk a kimenő 4 bitet a táblázat megfelelő cellája alapján.

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

3. ábra. egyféle S doboz leképezés

Az 1. ábrán látható S függvény, mely előállítja a user memória tartalmának S(data) képét, és elküldi az olvasónak, egy S és P dobozokból álló összetett függvény. A használt S és P dobozok táblázatait ismerve az

$$S^{-1}(S(\text{data})) = \text{data}$$

alapján visszkapjuk a tag-ben tárolt adatot. Hogy melyik S és P dobozt használjuk, azt a reader által küldött *key* kulcs fogja meghatározni. A reader egy célszámítógép, mely minden olyan számítási kapacitással és tárral rendelkezik, mely szükséges a kulcs előállítására és a kapott S(data) visszafejtésére. A tag-ek az S dobozok és a P dobozok elektronikus megvalósításait.

Az ellenőrzéshez a tárolt *data*-ról egy lenyomatot, „*digest*”-et készítünk. Ehhez kiválóan megfelelnek a közismert HASH függvények. A tag-ekben implementáljuk a HASH függvények egyikét, pl a MD5 függvényt. Ennek alkalmazásával lehetséges az elküldött, titkosított adat visszafejtés utáni ellenőrzése. Egy plusz védelmet nyújt az adatváltoztató, esetleg adat beszúrásával élő támadások ellen.

5. Következtetés

Az RFID rendszerek használata napjainkban is folyamatosan változik. Számos új technológia jelenik meg és a gyártók, multinacionális cégek törekednek arra, hogy ezeket az újdonságokat eljuttassák a felhasználóig. A transzponderek napról napra egyre kisebb kivitelben és olcsóbban kerülnek ki a gyárakból, mely szintén segíti annak elterjedését. A széles körű elterjedésnek köszönhetően, egyre több szegmensben találhatóak meg ezek a rendszerek, így annak veszélyeire, sebezhetőségeire is sokkal nagyobb hangsúlyt kell fektetni.

A fent vázolt rendszer alapot biztosíthat az egyszerű, olcsó transzponderek hordozta adatok védelmére. Hozzáteve, hogy ennek alkalmazásához meg kell változtatni a jelenleg használt Class1Gen2 tag-ek protokollját, és be kell építeni a megfelelő kommunikációs és S illetve P dobozokat megvalósító részeket.

6. Köszönetnyilvánítás

Ezúton szeretnék köszönetet mondani a Nemzeti Kiválóság programnak, hogy lehetővé tette a kutatás folytatását.

Dr Radványi Tibor publikációt megalapozó kutatása a TÁMOP 4.2.4.A/2-11-1-2012-0001 azonosító számú Nemzeti Kiválóság Program – Hazai hallgatói, illetve kutatói személyi támogatást biztosító rendszer kidolgozása és működtetése országos program című kiemelt projekt keretében zajlott. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Dr. Imre Sándor, Kis Zoltán, Molnár László, Pogácsa Attila, Schulez Róbert, Tóth Gábor – RFID rendszerek vizsgálata felhasználás és technológia szempontjából - <http://www.rfid.answare.hu:8080/site/kutatasi-erdmenyeink/radios-megoldasok/2006/rfid-rendszerek-vizsgalata-felhasznalas-es-technologia-szemponthabol.pdf/view>.
- [2] Klaus Finkezeiler – RFID Handbook, Third Edition, 2010
- [3] Pedro Peris-Lopez, Julio Cesar Hernandez-Castro, Juan M. Estevez Tapiador and Arturo Ribagorda – LMAP: A Real lightweight Mutual Authentication Protocol for Low-cost RFID tags - <http://events.iaik.tugraz.at/rfidsec06/program/papers/013%20-%20lightweight%20mutual%20authentication.pdf>
- [4] Hee-Jin Chae, Daniel J. Yeager, Joshua R. Smith, and Kevin Fu (University of Massachusetts) Maximalist Cryptography and Computation on the WISP UHF RFID Tag 2007
- [5] Sindhu Karthikeyan and Mikhail Nesterenko_Kent State University, RFID Security without Extensive Cryptography 2005
- [6] M. McLoone and M.J.B. Robshaw (Queen’s University, Belfast, U.K.) Public Key Cryptography and RFID Tags 2008
- [7] Ernst Haselsteiner, Klemens Breitfuß: Security in Near Field Communication (NFC) Philips Semiconductors Mikronweg 1, 8101 Gratkorn, Austria
- [8] Radványi Tibor, Biro Csaba, Király Sándor: RFID tagek elleni támadás és a védekezés lehetőségei, Attack against the RFID tags and possibilities of the defense, Networkshop 2014 Pécs, , ISBN: 978-963-88335-5-6, elektronikus kiadás.
- [9] Radványi Tibor, Bíró Csaba: Az adatvédelem helyzete az RFID-ban, . SzamOkt 2013. október 10-13, Nagyszeben (Sibiu, Románia), ISSN 1842-4546 283-289 oldal
- [10] Jung-Sik Cho, Sang-Soo Yeo, Sung Kwon Kim: Securing against brute-force attack: A hash-based RFID mutual authentication protocol using a secret value, Computer Communications 34 (2011) 391–397

- [11] A. Juels, RFID security and privacy: a research survey, *Selected Areas in Communications* 24 (2) (2006) 381–394. February.
- [12] S.S. Yeo, S.K. Kim, Scalable and flexible privacy protection scheme for RFID systems, *European Workshop on Security and Privacy in Ad hoc and Sensor Networks – ESAS’05 LNCS*, 3813, Springer, 2005, pp. 153–163.
- [13] S. Weis, S. Sarma, R. Rivest, D. Engels, Security and privacy aspects of low-cost radio frequency identification systems, in: *International Conference on Security in Pervasive Computing*, March 2003, pp. 201–212
- [14] S.A. Sarma, S.E. Weis, D.W. Engels, RFID systems and security and privacy implications, cryptographic hardware and embedded systems – *CHES 2002, LNCS*, vol. 2523, Springer, 2002. August, pp. 454–469.
- [15] S. Yu, K. Ren, W. Lou, A privacy-preserving lightweight authentication protocol for low-cost RFID tags, in: *IEEE MILCOM 2007*, October 2007, pp. 1–7.
- [16] Y.-C. Lee, Y.-C. Hsieh, P.-S. You, T.-C. Chen, An improvement on RFID authentication protocol with privacy protection, in: *Third International Conference on Convergence and Hybrid Information Technology – ICCIT 2008*, vol. 2, November 2008, pp. 569–573.
- [17] Jeongkyu Yang, Jaemin Park, Hyunrok Lee, Kui Ren, Kwangjo Kim (KOMSCO, ICU, WPI Mutual Authentication Protocol for Low-cost RFID 2005
- [18] NIST. Recommendation for block cipher modes of operation: Galois/counter mode (GCM) and GMAC. NIST Special Publication 800-38D, 2007.
- [19] Lauren De Meyer, Beg Bilgin, and Bart : Extended Analysis of DES S-boxes, *Proceedings of the 34rd Symposium on Information Theory in the Benelux*, 30-31 May 2013, Leuven, Belgium (pp. pp. 140-146).

Digitálisan aláírt dokumentumok hosszú távú tárolása

Long-term preservation of digitally signed documents

Gyurák Gábor^a

^aPécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Kar
gyurak@pmmik.pte.hu

Absztrakt: A digitális dokumentumok egyre fontosabb szerepet töltenek be a mindennapi életünkben. Ezen dokumentumok biztonságos tárolása önmagában sem egyszerű feladat és nagyban nehezíti a helyzetet, ha a dokumentumok hosszú távú tárolásáról is gondoskodni kell. További problémákkal találjuk szembe magunkat, amikor a digitális aláírással ellátott dokumentumok megőrzéséről van szó, amelyeknek hosszútávon biztosítani kell a hitelességét is. Ilyen dokumentumok például az elektronikus számlák, amelyekkel egyre többet fogunk találkozni, hiszen az Európai Bizottság szeretné elérni, hogy 2020-ra az elektronikus számlázás általánossá váljon Európában. A cikk bemutatja a digitálisan aláírt dokumentumok hosszú távú tárolásával kapcsolatban felmerülő problémákat és azok megoldásának lehetséges módjait. A cikk vizsgálja a téma kapcsolódását a hatályos magyar jogszabályokhoz, és ismerteti az elektronikus aláírással ellátott PDF (Portable Document Format) dokumentumok hosszú távú hitelességét biztosító, ETSI (European Telecommunications Standards Institute) által kidolgozott PAdES (PDF Advanced Electronic Signature) megoldást.

Kulcsszavak: adattárolás, digitális aláírás, PAdES, pdf

Abstract: Digital documents play an increasingly important role in our lives. Reliable digital storage of these documents is complicated and we have to deal with other problems if we would like to store these documents for a long time. Some documents, especially the most important documents are electronically signed. The long-term storage of electronically signed documents is more difficult, because we have to ensure the long-term validity as well. Electronic invoices (e-invoices) are also electronically signed documents and their role is getting more important. The proposal of the European Committee on e-invoicing aims to facilitate the use of e-invoices. By 2020 e-invoicing will be general in the EU. This paper describes the problems in connection with long-term storage of digitally signed documents. Possible solutions are also presented. In connection to this, the regulation of preserving electronically signed documents is also examined from the point of view of hungarian legislation. Finally it is shown how could ETSI's (European Telecommunication Standards Institute) PAdES (PDF Advanced Electronic Signature) support the long-term validity of e-documents using the widely used portable document format (PDF).

Keywords: data storage, digital sign, PAdES, pdf

1. Bevezetés

Az emberiség történetében mindig fontos szerepet töltött be az információk rögzítése és megőrzése az utókor számára. A legrégebbi emlékek az i.e. 13 000 körül készített barlangrajzok formájában maradtak ránk, amelyek közül a leghíresebbek a franciaországi Lascaux-ban találhatóak. Az időszámításunk előtti harmadik évezredben egy új korszak kezdődött az írás kialakulásával. Ötezer év elteltével is képesek vagyunk az uruki agyagtáblákon található információk értelmezésére, négyezer év után is el tudjuk olvasni Hammurapi kőbe vésett törvénykönyveit és az i.e. II. századból származó papirusz tekercseket. Könyvtárainkban őrizzük az elmúlt évezredek írásos emlékeit.[1]

A technológiai fejlődés az elmúlt évtizedekben alapvetően megváltoztatta az évezredek hagyományokat és az agyagtáblára, papiruszra és papírra vetett képi és írásos jeleket

felváltotta a mágnesszalagokra, optikai lemezekre és elektronikus hordozókra rögzített bitek sorozata.

Az új technológiák vitathatatlan előnyeinek elismerése mellett mindenképpen meg kell említeni két új – a korábbi megoldásokra nem jellemző – tulajdonságukat. Míg hagyományos adathordozókról (pl.: a papírról) az emberi érzékszervekkel elolvasható volt az adat, addig az új adathordozók (pl.: egy Blu-ray lemez vagy egy pendrive) tartalmának megismeréséhez speciális eszközökre van szükségünk. A másik fontos újítás, hogy amíg a hagyományos módon tárolt adatok értelmezéséhez nincs szükség speciális tudásra, addig a bináris formátumban lévő adatok értelmezése, az információ megjelenítése további nehézségeket okoz. Ezen új kihívások szemléltetéseként gondoljunk bele, hogy a Boticelli által 1486-ban vászonra festett képet bárki élvezheti, aki ellátogat a firenzei Uffizi galériába. Egy 1990-es években készített és Dr. Halo CUT¹ fájlformátumban floppy lemezre mentett kép megtekintésével kapcsolatban azonban már nehézségekbe ütközhetünk. Az első probléma, hogy napjainkban már csak elvétve találkozni olyan floppy meghajtóval, amely el tudná olvasni a lemezt. Ha mégis el tudjuk olvasni a lemezt, akkor nehéz lesz olyan szoftvert találni, amely értelmezni tudja az elavult formátumot és meg tudja jeleníteni a képet.

Napjainkban a legtöbb új információ digitális formában keletkezik, hiszen ezáltal kényelmesebben lehet létrehozni, módosítani és továbbítani az adatokat. Olyannyira így van ez, hogy egy ideje a papír alapú dokumentumokat is elektronikusan hozzuk létre, majd kinyomtatjuk őket, sőt a régebbi papír alapú dokumentumainkat elektronikus formátumúvá alakítjuk (pl.: szkenneljük) a könnyebb kezelhetőség érdekében.

A digitális formátumnak azonban nem csak az információk praktikusabb kezelése, hanem megőrzése szempontjából is jelentősége van. Egyes információk különösen fontosak egy személy számára: a megismételhetetlen események emlékeit őrző fényképektől a nyugdíj összegét meghatározó foglalkoztatottsági és társadalombiztosítási adatokig bezárólag. Emellett az olyan információkról sem szabad megfeledkezni, amelyeket nem az egyének, hanem a jövő nemzedékei számára kell megőrizni. Ilyenek lehetnek a tudományos, kulturális, örökségvédelmi információk [2] és ide sorolhatók akár az atomrobbantási kísérletek eredményei is.

Függetlenül attól, hogy egyéni vagy közösségi érdekről van szó, vannak olyan adatok amelyek tárolását évtizedekig, esetleg évszázadokig biztosítani kell. Hogy pontosan mennyi idő után beszélhetünk hosszú távú adattárolásról, azt nehéz általánosan megmondani, mert nem létezik éles határ. Az alkalmazás jellegétől függően akár néhány év is hosszú táv lehet (lásd 5. fejezet), de az évtizedekig tartó adattárolás már biztosan hosszú távnak tekinthető.

2. Hosszú távú adattárolás

Az előző fejezetben felvázoltak alapján meghatározhatók a digitális dokumentumok hosszú távú adattárolásának legfőbb problémái. A technológia fejlődése azt eredményezi, hogy az adatok fizikai tárolását és olvasását végző hardver eszközök, az adatok értelmezéséhez szükséges szoftver eszközök és adattárolási formátumok gyorsan elavulnak. Emellett meg kell említeni, hogy napjaink adattároló eszközei csak korlátozott ideig képesek megőrizni a rajtuk elhelyezett adatokat. Egy átlagos felhasználásra szánt optikai adattároló lemez (pl.: CD, DVD, Blu-ray) csak néhány évig képes megbízhatóan tárolni a rajta elhelyezett adatokat, míg a speciális bevonatokkal ellátott prémium kategóriás lemezekről sem várhatunk el 10-15 évnél

¹ Az 1997-ben népszerű PaintShop Pro képszerkesztő program által támogatott fájlformátum

többet.[3] Ennél többre nincs is szükség, mert a technológia elavul, és már nem lesznek olvasó eszközök, amelyekkel a lemezek tartalma visszanyerhető lenne.²

A hosszú távú adattárolás gyakorlati megoldásaiban két irányvonal rajzolódik ki. Az egyik a migrációs, a másik az emulációs technika. [4]

A *migráció* lényege, hogy adatainkat időközönként átalakítjuk az új technológiáknak megfelelő fizikai és logikai formába. Logikai átalakításnál a régi formátumról átalakítást végzünk az új - szabványos - formátumokra (pl.: Word'97 formátumú dokumentumot átkonvertálunk Word'2013 formátumba). Fizikai szinten a migrációnál biztosítani kell azt is, hogy időközönként áttérjünk új adattároló technológiára (pl.: floppyról CD lemezre másoljuk adatainkat), kivédve ezzel a technológia elavulását. Az új technológiákra való áttérés mellett érdemes fontolóra venni azt a lehetőséget is, hogy hagyományos (nem digitális) módon is tároljuk az adatainkat (pl.: mikrofilm³, papír⁴).

A másik hosszú távú adattárolási megoldás az *emuláció*, amelynek során az adatokat meghagyjuk az eredeti formájukban. Ezek elavulását pedig úgy küszöböljük ki, hogy az új rendszerekben emuláljuk a régi (elavult) rendszer hardver és szoftver környezetének működését, amely meg tudja jeleníteni az információkat. Erre a célra fel lehet használni a különböző virtualizációs technikák nyújtotta lehetőségeket.

3. Elektronikus formátumú hiteles dokumentumok

A korábbiakban általánosságban a dokumentumokról beszéltünk, nem esett szó azok tartalmáról. Az írás kezdete óta léteznek olyan „dokumentumok”, amelyek fontos információkat tartalmaznak. A történelem során kialakultak olyan technikák, amelyek az ilyen dokumentumok biztonságát, azon belül is a bizalmasságát, integritását és hitelességét hivatottak megvédeni.

A bizalmasság lényege, hogy illetéktelenek ne ismerhessék meg a dokumentum tartalmát. Ehhez titkosítást alkalmaztak, amelynek története egészen Szun-ce⁵ idejéig vezethető vissza.

A dokumentum hitelessége azt jelenti, hogy a dokumentum készítőjének a személye bizonyítható. A hagyományos dokumentumokat a készítő ellátta kézjeggyel (kézzel írott aláírásával), így vált a dokumentum hitelessé. Mivel napjainkban a legtöbb információ elektronikus formában áll rendelkezésre, célszerű lenne a hiteles dokumentumokat is elektronikusan kezelni. Azonban ezen a területen napjainkban még inkább a papír alapú dokumentumok dominálnak, mivel az elektronikus dokumentumok előnyei - egyszerű létrehozni, másolni, módosítani őket – a hitelesség szempontjából könnyen a hátrányukká válhatnak.

Az információs társadalom kialakításának egyik fontos eleme a „papírmentes” közigazgatási rendszer kialakítása (e-kormányzat) és az elektronikus ügyintézés lehetőségének megteremtése. Az ügyletek jelentős része megköveteli a hiteles dokumentumok használatát, amelyek közül kiemelt helyen szerepel az elektronikus szerződéskötés és az elektronikus

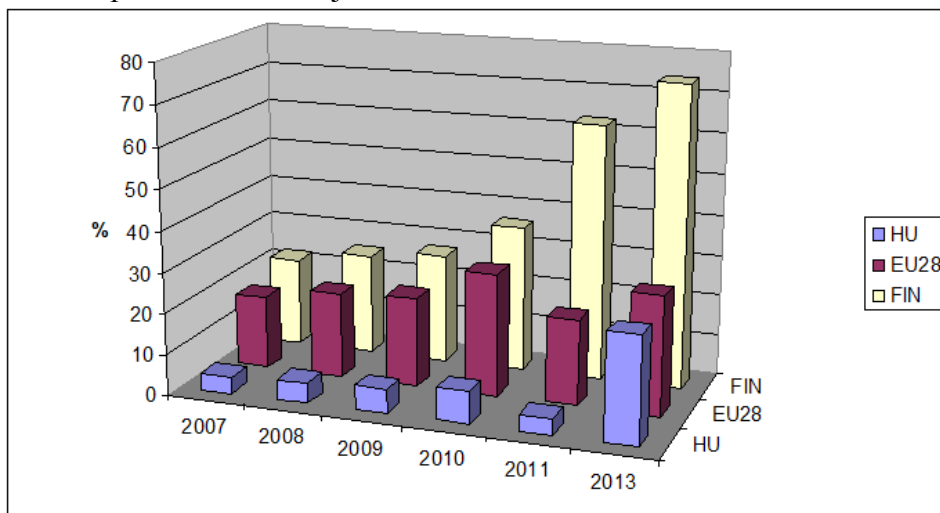
² Vannak olyan új fejlesztésű lemezek is, amelyek a gyártó ígérete szerint 1000 évig képesek tárolni a felírt adatokat. Ilyen a Millenniata cég által fejlesztett M-DISC. (www.mdisc.com)

³ A legrégebbi mikrofilm több mint 70 éves.

⁴ A legrégebbi papír alapú dokumentum több mint ezer éves.

⁵ Szun-ce i.e. V. században élt hadtudós és matematikus

számlázás. Az „Európa 2020 stratégia” egyik pillérét alkotó európai digitális menetrend⁶ a modern és versenyképes uniós gazdaság nélkülözhetetlen elemeként határozza meg az e-kormányzatot. Ennek keretében az Unió jelentős erőfeszítéseket tesz az elektronikus ügyletek elterjesztése, többek között az elektronikus számlázás általánossá válása érdekében. Az Európai Bizottság „Az elektronikus számlázás előnyeinek kiaknázása Európában”⁷ című közleményében felszólította a tagállamokat, hogy 2020-ig tegyék az elektronikus számlázást a számlakiállítás alapértelmezett módjává.



1. ábra. Elektronikus számlázásra képes vállalkozások aránya⁸

A fenti ábrán az Eurostat felmérése azt mutatja, hogy a vállalatok milyen arányban biztosítanak elektronikus számlázás szolgáltatást. Magyarországon a 2013-as esztendőben történt nagy áttörés, amikor hazánk felzárkózott az EU átlaghoz.

Jól látható, hogy az elektronikus számlázás növekvő tendenciát mutat, és az érdekelt felek is lépéseket tesznek annak érdekében, hogy a 2020-ra kitűzött cél megvalósuljon. Természetes dolog, hogy a hiteles dokumentumokat is tárolni szeretnénk, és sokszor célunk, sőt legtöbbször jogszabályi kötelezettségünk is, hogy ezeket a dokumentumokat hosszú távon tároljuk. A hiteles elektronikus dokumentumok hosszú távú tárolásakor a 2. fejezetben bemutatott problémákon túl új kihívásokkal is meg kell küzdeni, így biztosítani kell a dokumentumok *hosszú távú hitelességét* is. A következő fejezetekben erről lesz szó.

4. Hitelesség elektronikusan

Technikai háttér

Az elektronikus dokumentumok hitelesítésének ugyanaz az alapelve, mint a hagyományos dokumentumokénak: a dokumentumot aláírásunkkal látjuk el, amely azonosítja az aláíró személyét. Az elektronikus dokumentumok hitelesítésénél az a különbség, hogy hitelesítő megoldásként kriptográfiai algoritmusokkal létrehozott *elektronikus aláírást* alkalmazunk.

⁶ <http://ec.europa.eu/digital-agenda/> (Utoljára látogatva: 2014. június 14.)

⁷ COM (2010) 712, Brüsszel, 2010.12.02. A Bizottság közleménye az Európai Parlamentnek, a Tanácsnak, az Európai Gazdasági és Szociális Bizottságnak és a Régiók Bizottságának

⁸ <http://epp.eurostat.ec.europa.eu/tgm/table.do?tab=table&plugin=0&language=en&pcode=tin00114>

Az elektronikus aláíráshoz a PKI (Public Key Infrastructure) eszköztárát használjuk fel. A rendszer lényege, hogy minden érintett fél egy-egy kulcspárral rendelkezik. Az egyik kulcs titkos (privát kulcs), azt a tulajdonos nem adhatja át másnak, a másik kulcs nyilvános (publikus kulcs), bárki számára hozzáférhető. Az aláírás gyakorlatilag a dokumentum hash lenyomatának titkosítása az aláíró privat kulcsával. Az aláírás ellenőrzése úgy történik, hogy az aláíró publikus kulcsával dekódolják az aláírásból a dokumentum lenyomatát, amelyet már össze lehet hasonlítani a dokumentumról készített újabb lenyomattal. Az aláírást csak az tudja elkészíteni, akinek a birtokában van a privat kulcs. Ha a privat kulcsot illetéktelenek megszerzik (kompromittálódik a kulcs), akkor a továbbiakban már nem lehetünk biztosak az aláíró személyében.

A rendszer részét képezik még a hitelesítő szervezetek (CA, Certificate Authority), amelyek a velük szerződésben álló ügyfelekről tanúsítványokat (certificate) adnak ki. A tanúsítvány egy érvényességi idővel rendelkező elektronikus aláírt igazolás, amelyben a hitelesítő szolgáltató igazolja, hogy egy adott nyilvános kulcs egy adott szereplőhöz tartozik. A tanúsítványon szereplő aláírás a CA privat kulcsával készül, tehát ezt annak publikus kulcs párjával lehet ellenőrizni. A CA publikus kulcsát egy másik CA hitelesíti, így egy tanúsítvány lánchoz jutunk. A CA feladata továbbá az is, hogy az érvénytelen tanúsítványokról visszavonási listát (CRL, Certificate Revocation List) adjon ki, illetve online ellenőrizhetővé tegye a tanúsítványok érvényességét (OCSP, Online Certificate Status Protocol).

A rendszer további elemei az időbélyegek (timestamp), amelyek időbélyeg szolgáltatók által kiállított kriptográfiai elemek annak bizonyítására, hogy a dokumentum egy adott időpontban már létezett. Technikai értelemben az időbélyeg is egy aláírás, amelyet az időbélyeg szolgáltató a dokumentum hash lenyomatából és az aláírás időpontjából számít.

Jogszabályi háttér

Az előző pontban ismertetett technikai megoldás műszaki értelemben megoldja a dokumentum-hitelesítési problémát, azonban a gyakorlatban mindaddig nem volt használható, amíg jogszabályi szinten nem került elismerésre. Az Európai Parlament és a Tanács korán felismerte az elektronikus aláírásban rejlő potenciált, és már 1999-ben iránymutatást adott a tagállamoknak az 1999/93/EK irányelvvel.⁹ Ennek alapján Magyarországon is megszületett az elektronikus aláírásról szóló 2001. évi XXXV. törvény (a továbbiakban: Eat.), amely megteremtette a megfelelő jogszabályi háttérrel az elektronikus hitelesítéshez.

A törvény négy szolgáltatást határoz meg:

- hitelesítés szolgáltatás;
- időbélyegzés szolgáltatás;
- eszköz szolgáltatás;
- elektronikus archiválás szolgáltatás.

Az Eat. megkülönböztet minősített és nem minősített szolgáltatókat, és ezzel párhuzamosan beszélhetünk minősített elektronikus aláírásról, fokozott biztonságú elektronikus aláírásról és egyéb, az előző kettő kategóriába nem sorolható elektronikus aláírásról.

⁹ Az Európai Parlament és a Tanács 1999/93/Ek irányelve az elektronikus aláírások közösségi programjáról, 1999. december 13.

5. Hitelességet érintő kihívások hosszú távon

Egy elektronikus dokumentum aláírásakor az aláíró kötelezettséget vállal annak tartalmáért. Egy dokumentum hitelességének ellenőrzésekor megvizsgáljuk, hogy a dokumentumon elhelyezett aláírás érvényes-e. Az Eat. csak az érvényes aláíráshoz rendel jogkövetkezményt.¹⁰

Az ellenőrzés az alábbi lépésekkel történik:

- elkészítjük a dokumentum hash lenyomatát $H(M)$;
- az aláíró publikus kulcsával visszafejtjük az aláírt hash lenyomatot $H(M)'$;
- ha a két lenyomat megegyezik ($H(M)=H(M)'$), akkor arra következtetünk, hogy az aláírás készítője birtokában volt a publikus kulcs párjának (privát kulcs).

Az eddigi lépésekkel az bizonyítható, hogy a dokumentum nem változott meg az aláírás óta és az aláírás a nyilvános kulcshoz tartozó privát kulccsal készült. A továbbiakban azt kell ellenőrizni, hogy a kulcspár kihez tartozik és a privát kulcs az aláíró kizárólagos tulajdonában volt-e az *aláírás időpontjában*. A hangsúly a kötelezettségvállalás időpontján van, tehát nagyon fontos, hogy az akkori körülményeket kell ellenőrizni függetlenül attól, hogy az ellenőrzést közvetlenül az aláírás után vagy akár évtizedekkel később végezzük. A nyilvános kulcs tulajdonosának igazolására a tanúsítványok szolgálnak és a tanúsítványok hitelességéért a CA-k felelnek. A feladat tehát annak ellenőrzése, hogy az aláírás időpontjában érvényes volt az aláíró tanúsítványa, valamint létezett olyan tanúsítványlánc, amely visszavezethető egy root CA tanúsítványáig és a láncban lévő összes elem érvényes volt (a tanúsítványok nem voltak felfüggesztve vagy visszavonva).

Látható, hogy a hitelesség ellenőrzése igen összetett folyamat, amely számos adat ellenőrzését teszi szükségessé. Ha mindez röviddel az aláírást követően történik, akkor viszonylag problémamentes az ellenőrzés. Ezt a rövid időtartamot nehéz nevesíteni, de ha a tanúsítványok alapértelmezett érvényességi idejét tekintjük, akkor 1-2 éves tartamról beszélhetünk (feltéve persze, hogy közben nem vonják vissza a tanúsítványt, mert akkor a visszavonás időpontja lehet vízvonalzó).

Ha a tanúsítvány érvényessége lejár vagy visszavonják, attól az aláírás még érvényes marad, de az érvényesség igazolása válik szükségessé. Az igazolás során az alább ismertetett problémák merülhetnek fel.

Az aláírás időpontja

Amennyiben az aláírás időpontját nem tudjuk hitelesen bizonyítani, akkor az aláírás csak addig érvényes, amíg a tanúsítvány érvényes (nem járt le az érvényességi ideje és nem vonták vissza a tanúsítványt). Az érvényességi időt úgy tudjuk növelni, ha bizonyíthatóvá tesszük az aláírás időpontját, azaz időbélyeggel látjuk el. Ettől kezdve az időbélyeg érvényessége is fontos szempont lesz a hitelességhez.

¹⁰ Eat. 4-5. §

Visszavonási információk

Egy tanúsítvány érvényessége az érvényességi időn belül is felfüggeszthető, illetve visszavonható. Tipikusan ez történik akkor, ha felmerül annak gyanúja, hogy a privát kulcs kompromittálódott. Ha az aláírást időbélyeggel látták el, akkor a tanúsítvány visszavonása esetén is megállapítható annak érvényessége. Ha a visszavonás előtt történt az aláírás, akkor az érvényesnek tekinthető. A visszavonási információkat CRL-ek formájában teszi közzé a hitelesítés szolgáltató, illetve OCSP lekérdezéseket tesz lehetővé. A közzétett adatok csak az aktuális visszavonási információkról szólnak, az idő múlásával ezek már nehezen, sőt egy idő után már egyáltalán nem hozzáférhetők. Az Eat. vonatkozó rendelkezése alapján a szolgáltatók a tanúsítvány érvényességének lejáratát után még tíz évig (illetőleg az elektronikus aláírással, illetve az azzal aláírt elektronikus dokumentummal kapcsolatban felmerült jogvita jogerős lezárásáig) kötelesek tárolni a fenti adatokat.¹¹ Ez azt is jelenti, hogy ha ennél hosszabb távon szeretnénk biztosítani a hitelességet, akkor össze kell gyűjteni a visszavonási információkat, és gondoskodni kell annak hosszú távú megőrzéséről.

CA információk

Az aláírás érvényességének megállapításához a tanúsítványláncban található összes szereplő adatait ellenőrizni kell. A hitelesítés szolgáltató szervezetek tanúsítványai időközben lejárhathatnak és a szervezetek meg is szűnhetnek. Ez a kérdés érinti az időbélyegek hitelességét is, hiszen az időbélyegzést legtöbbször ugyanazok a szervezetek végzik, mint a tanúsítványok kezelését. A hosszú távú hitelesség érdekében szükséges ezen adatok összegyűjtése és tárolása is.

Elavult algoritmusok

A elektronikus aláírás hátterében kriptográfiai eljárások, titkosító algoritmusok (pl.: RSA), lenyomatkészítő hash algoritmusok (pl.: MD5, SHA-512) működnek. Ezek olyan tulajdonságokkal rendelkeznek, amelyek lehetővé teszik a rendszer biztonságos működését; azaz a kor technológiai fejlettségét figyelembe véve nem létezik hatékony módszer és elegendő számítási kapacitás a rendszer kompromittálására. Jelenlegi tudásunk szerint nem létezik megfelelő módszer az egész számok faktorizációjára.[6] Ez az alapja az RSA-titkosításnak, ezért fogadjuk el biztonságos módszernek az RSA alapokon nyugvó elektronikus aláírási rendszert.

Ismét egy olyan ponthoz érkeztünk, amelynél figyelembe kell vennünk az időtényezőt. Azok az algoritmusok, amelyeket ma használunk talán már néhány év múlva, de évtizedekkel később már biztosan elavultnak számítanak.

A kriptográfiai építőelemek elavulásának egyik jó példája lehet az MD5¹² hash algoritmus, amelyet az ezredfordulón még előszeretettel alkalmaztak. A dokumentumokból képzett MD5 hash lenyomatokat látták el elektronikus aláírással. A hash algoritmusok alkalmazhatóságának egyik kritériuma, hogy ütközés-ellenállónak kell lenniük. Az ütközés-ellenállóság azt jelenti, hogy nehéz két olyan üzenetet találni, amelyeknek azonos lenne a lenyomata. Mint kiderült, az MD5 nem teljesíti ezt a tulajdonságot, így kriptográfiai alkalmazásokra nem alkalmas. Egy másik széles körben használt hash algoritmus az SHA-1¹³, amelynek kriptográfiai célokra

¹¹ Eat. 9.§ (7) bekezdés

¹² RFC 1321

¹³ RFC 3174

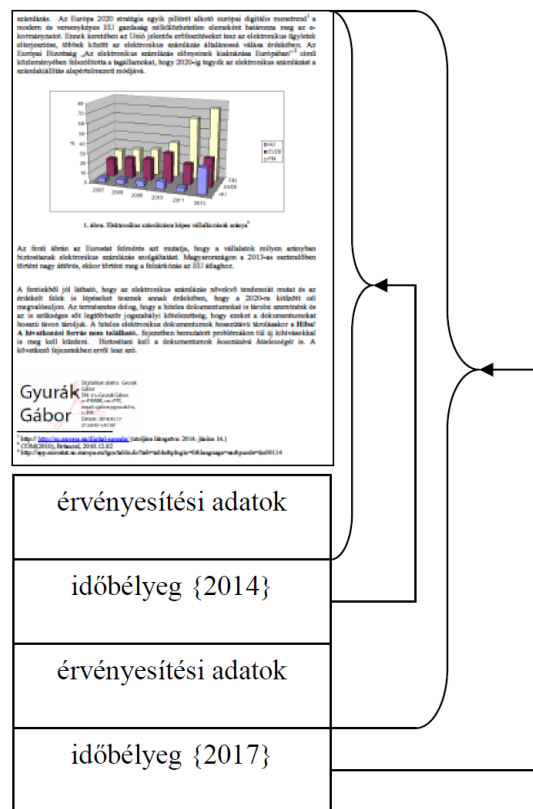
történő használata Magyarországon 2011. december 31. után már nem engedélyezett a CA szervezetek számára.¹⁴

Mindemellett a technológia fejlődése is egyre nagyobb számítási kapacitást tesz lehetővé, amelynek révén lehetővé válik egyes megoldások kimerítő keresés (bruteforce) alapú támadása. Szerencsére a szabályozási oldal is felismerte ennek jelentőségét, és a jogszabályok előírják a felügyelő szerveknek, hogy ellenőrizzék és utasítsák a szolgáltatókat a biztonságos algoritmusok és megfelelő hosszúságú kulcsok használatára.

6. Megoldások

Az előző fejezetben bemutatott problémákra olyan megoldást kell találni, amely műszaki és jogszabályi szempontból is képes biztosítani az elektronikusan aláírt dokumentumok hosszú távú hitelességét.

Az ETSI által kidolgozott PAdES (PDF Advanced Electronic Signatures) 4. fejezete a PAdES-LTV (Long Term Validity) egy olyan fejlesztés, amely kiegészíti a PDF formátumot olyan képességekkel, amelyek lehetővé teszik az elektronikus aláírással ellátott dokumentum hitelességének hosszú távú ellenőrizhetőségét.[7]



2. ábra PAdES dokumentum

¹⁴ A Nemzeti Média és Hírközlési Hatóság határozatban értesíti a CA szervezeteket az egyes algoritmusok felhasználásának korlátozásáról. Az NMHH az EF/26838-8/2011. ügyiratszámú határozatában értesíti a Netlock Kft.-t az SHA-1 algoritmus használatának korlátozásáról. A határozat alapját az ETSI által kiadott ETSI TS 102 176-1 v 2.1.1 jelentette.

A kiindulási alapot egy elektronikus aláírással és időbélyeggel ellátott PDF dokumentum jeleni. Ez kiegészítésre kerül egy új konténerrel, amely minden olyan kiegészítő adatot tartalmaz, amelyet felhasználunk az aláírás ellenőrzésére (érvényesítési adatok):

- a tanúsítványláncban szereplő hitelesítő szervezetek tanúsítványai;
- visszavonási információk (CRL-ek és OCSP válaszok);
- az időbélyeg szolgáltató tanúsítványa.

A 2-es ábra szerint mindezt egy időbélyeggel látjuk el, amely az időbélyeg érvényességi idején belül biztosítja a dokumentum hitelességét akkor is, ha már lejárt az aláíró tanúsítványának érvényessége. Az aláírás ellenőrizhető akkor is, ha időközben a CA-k adatai vagy a visszavonási információk már nem lennének elérhetők, hiszen ezeket csatoltuk a dokumentumhoz.

Ha hosszú ideig akarjuk tárolni a dokumentumot, akkor az előbbi lépések megismétlésével (érvényesítési adatok hozzáfűzése és időbélyegzés) biztosítható a hitelesség. Csupán arra kell ügyelni, hogy a „frissítés” az utolsó időbélyegző tanúsítványának érvényességi ideje alatt történjen és az újbóli időbélyegzésnél alkalmazott algoritmus megfeleljen a kor követelményeinek (megfelelően erős algoritmus és hosszú kulcsok használata).

A műszaki megoldás mellett akkor működőképes ez a rendszer, ha megfelelő jogszabályi háttér támogatja. Magyarországon a digitális archiválás szabályairól szóló 114/2007. (XII. 29.) GKM rendelet szabályozza az elektronikus aláírással ellátott dokumentumok hosszú távú tárolását.

A rendelet 4.§ (4) bekezdése a megőrzési kötelezettség időtartama alapján megkülönbözteti a hosszú távú tárolást, amely a hatályos jogszabály alapján 11 éven túli megőrzést jelent. Ebben az esetben a megőrzésre kötelezett feladata, hogy

- gondoskodik az elektronikus aláírás hosszú távú érvényesítéséhez szükséges információk beszerzéséről és megőrzéséről ;
- minősített szolgáltató által kibocsátott időbélyegzőt helyeztet el az érvényességi láncon;
- az előző pontban meghatározott időbélyegzést megismétli akkor, ha az alkalmazott kriptográfiai algoritmusok elavulnak.

Az előírásoknak eleget tehetünk saját magunk vagy a tárolással megbízhatunk archiválási szolgáltatót is. Utóbbi esetben vélelmezni kell, hogy a szolgáltató jól végzi feladatát, ezért egy esetleges jogvita esetén az ellenérdekelt félnek kell bizonyítani a dokumentum hitelességével kapcsolatos problémát.

A számvitelről szóló törvény¹⁵ alapján az elektronikus számlák megőrzési kötelezettsége nyolc év. Az archiválás szabályai szerint ez nem minősül hosszú távú tárolásnak, tehát nem vonatkoznak rá a szigorú szabályok. Ennek ellenére műszaki szempontból mindenképpen indokolt lehet a fenti archiválási megoldások alkalmazása nyolc éven belül is, hiszen az 5. fejezetben felvázolt problémák ebben az időintervallumban is érinthetik az aláírt dokumentumainkat.

¹⁵ A számvitelről szóló 2000. évi C. törvény 169.§ (2) bekezdés.

7. Összegzés

Az előző fejezetek alapján látható, hogy a hosszú távú adattárolás és különösen a hiteles dokumentumok tárolása nehéz és igen költséges feladat. A fenti technológiákkal biztosítható a hosszú távú hitelesség, de nem szabad megfeledkezni a hosszú távú adattárolás általános problémáiról sem. A hitelesség biztosítása mellett az eredeti dokumentum értelmezését is meg kell valósítani, így visszajutottunk a 2. fejezetben tárgyalt kérdéshez.

A 114/2007. (XII.29.) GKM rendelet 2.§ (2) szerint:

„A megőrzésre kötelezett köteles biztosítani, hogy az őrzött elektronikus dokumentumok értelmezhetősége (olvashatósága) – a dokumentum megjeleníthetőségét lehetővé tevő szoftver- és hardverkörnyezet biztosításával – a megőrzési kötelezettség időtartama alatt megmaradjon.”

A fentiek szerint a migrációs technikára logikai szinten nincs lehetőség, hiszen a dokumentumot eredeti formátumban kell megtartani. A problémára az emulációs technika, illetve az eredeti hardver és szoftver környezet megőrzése jelenthet megoldást.

8. Köszönetnyilvánítás

A kutatást támogatta a „Szuperszámítógép, a nemzeti virtuális laboratórium” című TÁMOP-4.2.2.C-11/1/KONV-2012-0010 számú pályázat.

Irodalomjegyzék

- [1] Várkonyi Nándor: Az írás és a könyv története, Széphalom Könyvműhely (2001)
- [2] Mingquan Zhou, Guohua Geng, Zhongke Wu: Digital Preservation Technology for Cultural Heritage, Springer (2012)
- [3] Hans Peek, Jan Bergmans: Origins and Successors of the Compact Disc, Springer (2009)
- [4] U.M. Borghoff, P. Rödig, J. Scheffczyk, L. Schmitz: Long-Term Preservation of Digital Documents, Springer (2006)
- [5] Dr. Berta István Zsolt: Nagy E-szignó könyv, Microsec Kft. (2011)
- [6] Buttyán Levente, Vajd István: Kriptográfia és alkalmazásai, Typtotex (2012)
- [7] ETSI TS 102 778 (2009-07) Electronic Signatures and Infrastructures
- [8] 2001. évi XXXV. törvény az elektronikus aláírásról
- [9] 114/2007. (XII. 29.) GKM rendelet a digitális archiválás szabályairól
- [10] Sangchul Song, Joseph Jaja: Techniques to audit and certify the long-term integrity of digital archives, Springer Verlag (2009)
- [11] Nick Pope: Protecting Long Term Validity of PDF documents with PAdES-LTV, Securing ELeCtronic Business Processes, Vieweg pp. 320-327 (2009)
- [12] Jean-Francois Blanchette: The digital signature dilemma, Ann. Télécommun, 61, n 7-8 pp. 908-923 (2006)

Radioaktív hulladékokat tároló rendszer adatainak hosszútávú megőrzése.

Realization of long-term data preservation at a radioactive waste storage facility

Pandur Béla

Pécsi Tudományegyetem, PMMIK, Rendszer és Szoftvertechnológiai Tanszék
pabe@t-email.hu

Absztrakt: Az adattároló rendszer tervezésénél követelmény volt az adatok minimálisan 75 éves megőrzése. A keletkező adatokat olyan formában kell tárolnunk, hogy a dokumentumok elérhetők legyenek a tervezett élettartam alatt, továbbá az adatállományok bővíthetők legyenek a később meghatározott leíró adatokkal. Az elkészült rendszer figyelembe veszi a már létező szabványokat, elsősorban az ISO 14721:2012 (Open Archival Information System Reference Model) javaslatait. A metaadatok tervezésénél a „Dublin Core” ajánlást tekintettük a minimális követelménynek, a későbbi interoperabilitás érdekében. Az adattartalom meghatározásánál az informatikai tervezés támaszkodott a Radioaktív Hulladékokat Kezelő Nonprofit Kft korábbi, több mint tíz éves adatfeldolgozási tapasztalataira, illetve a „Stratégiai Mérnöki Iroda” által megfogalmazott követelményekre. A megvalósult rendszer részfolyamatai az adatok feldolgozásának speciális igényei miatt kissé eltérnek OAIS -ben meghatározott sémától, de nem kerülnek vele ellentmondásba. Az archív állományokat mágneslemezen, egyszer írható mágnesszalagon, illetve papír hordozón tároljuk. Az állományok kizárólag ASCII kódokból épülnek fel a biztonságosabb visszaállítás érdekében. A mágnesszalag és a papír alapú hordozókon a várható élettartam migráció nélkül is tudja teljesíteni az élettartamra vonatkozó előírásokat. A tartalom megjeleníthetőségét az operációs rendszer és adatbázis független tárolás segíti elő. Szerző a rendszer követelményspecifikációját készítette, és szakértőként működött közre a megvalósításban.

Kulcsszavak: hosszú távú adattárolás, adatmegőrzés, tudományos adatok, mérnöki adatok, OAIS

Abstract: The requirement for the design of the data storage system was that the data must be preserved for a minimum of 75 years. The resulting data is to be stored in a form that the documents are available during the planned life span of the system, and the data files can be upgraded later with descriptive data to be specified later. The completed system takes into account existing standards, especially the ISO 14721:2012 (Open Archival Information System Reference Model) recommendations. For the design of the metadata, the "Dublin Core" recommendation was considered the minimum requirement for the purpose of future interoperability. For the definition of the data content, the information design relied on the over ten years of previous data processing experience of the RHK (Public Limited Company for Radioactive Waste Management), as well as on the requirements of the "Strategic Engineering Office" (Stratégiai Mérnöki Iroda). Due to the special data processing requirements, some of the processes of the implemented system differ slightly from the model specified in OAIS, but they are not in conflict with it. The archive files are stored on disks, on read-only magnetic tapes and on paper. The files are built exclusively of ASCII codes for safer recovery. The expected life spans of the special magnetic tapes and of the paper carriers meet the related specifications without migration. The content display is facilitated by the operating system and by database independent storage. The author set up the requirement specifications of the system and participated in the implementation an expert consultant.

Keywords: long-term, data storage, scientific data, engineering data, preservation, OAIS

1. Bevezetés

A programcsomag a Radioaktív Hulladékokat Kezelő Közhasznú Nonprofit Kft megrendelésére, a radioaktív hulladékok kezelésének igényeit figyelembe véve készült. Az elkészült rendszer kielégít a hasonló alkalmazásra szánt szoftverek általános elvárásait, és az adatbeviteli modul módosításával

más szakterület igényeihez is könnyen adaptálható.

A radioaktív hulladékok tárolásának egyik fontos jellemzője, hogy a tárolást hosszú időtávra tervezik. A kis és közepes aktivitású tárolókat a tervek szerint 50-100 évig monitorozzák, majd lezárják. A hulladék véglegesen a tárolás helyén marad. A radioaktív hulladék helyét, összetételét, a tárolás műszaki körülményeit tartalmazó dokumentumokat legalább a monitorozás időszakában elérhető formában kell tárolni.

A radioaktív hulladékok adatainak jelentős részét az Országos Atomenergia Hivatal útmutatója [1] a „végleges”, 30 évet meghaladó megőrzési kategóriába sorolta. A céltervben meghatározott megőrzési idő 75 év. Amíg a tárolóban aktív munka folyik a dokumentumokra könnyen elérhető formában van szükség. Elvárható, hogy a lényeges dokumentumok a céltervben meghatározottnál hosszabb ideig is olvashatók és értelmezhetőek maradjanak. A tároló helyét, és paramétereit több száz év múlva is jó lenne ismerni, annak ellenére, hogy a tárolt anyagok sugárzási szintje már veszélytelen értékre csökkent és a tárolót lezárták.

A létesítéssel és üzemeltetéssel kapcsolatos adatvagyon jelentős értéket képvisel, és máshonnan nem megszerezhető ismereteket tartalmaz. Az adatok jelentős része meg nem ismételhető mérések eredménye, nem pótolható. A tároló létrehozását megelőző állapotok soha többé nem hozhatók létre, a paraméterek nem mérhetőek újra. A geológiai és térképészeti adatok a további tárolóterek kialakításának tervezéséhez, biztonsági elemzésekhez nélkülözhetetlenek. Az üzemeltetés során keletkezett adatokból létrehozott tudásbázis hasznos az üzemeltetőnek, és fontos lehet a későbbi létesítmények tervezőinek és a tervezési paramétereket előíró szervezeteknek is.

Rövid távon a tároló üzemeltetéséhez, a gépészeti, villamos módosításokhoz napi szinten szükség van a létesítmény kiviteli terveinek elérhetőségére, azok aktualizálására. Az aktualizálhatóság feltétele az, hogy a dokumentumok elérhetőek legyenek, és szerkeszthető formában is rendelkezésre álljanak.

A cél tehát egy a kiválasztott dokumentumok hosszú idejű tárolását biztosító rendszer létrehozása, ami emellett a napi igényeket (technológia felügyelete, jelentések) is kiszolgálja. A feladat megoldására nincs kialakult gyakorlat, hiszen a digitális adattárolás jelenlegi hardver és szoftver eszközei legfeljebb 10-15 éves múltra tekinthetnek vissza. A jövő informatikájáról 50 éves távlatban csak sejtéseink vannak.

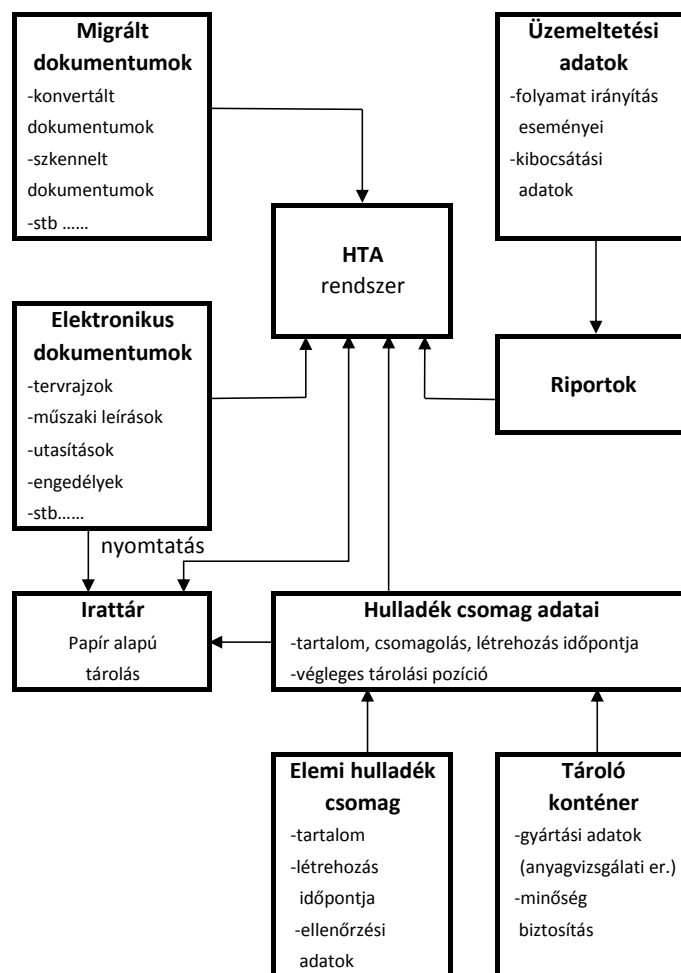
2. Az adatok forrásai

A megőrzendő dokumentumok forrása és formátuma nagyon változatos. Az iratok egy része csak papír alapon létezik. Ezek főként a korábban keletkezett jelentések, térképek, elemzések. A csak papír alapon létező dokumentumok digitalizálását RHK folyamatosan végzi. Az új tervek, dokumentumok már digitális formában kerülnek a rendszerbe. A dokumentumok eredeti formája is digitális. A szövegszerkesztővel, vagy számítógépes tervező rendszerrel készült dokumentumok eredetijének a digitális változatot tekintjük.

A korábbi elektronikus rendszerekben tárolt adatokat az új adatállományokba be kell illeszteni, és tartalmukat ellenőrizni kell. A migráció jó alkalom az adatok szakmai ellenőrzésére is. A migrálás során elvégezhető néhány formai és határérték vizsgálat, de az ellenőrzés sajnálatosan nem automatizálható teljes mértékben. A mérési metodika változásából, vagy a mértékegység eltérésből eredő hibák nem mindig ismerhetők fel automatizáltan.

A kis és közepes aktivitású tároló építése során RHK Kft (Radioaktív Hulladékokat Kezelő Közhasznú Nonprofit Kft) minden résztvevőtől elektronikus formában is bekérte a dokumentumokat, ami egyszerűsíti a további feldolgozást. A korábbiakhoz képest az elektronikus formában is rendelkezésre álló dokumentáció megléte nagy előrelépés. A tapasztalatok alapján célszerű lett volna a formátumok pontosabb meghatározására a szerződésekben (Pl.: AutoCad verzió, formátum, PDF verzió). A beszállítói szerződések megkötésekor azonban komplett informatikai specifikáció még nem állt rendelkezésre. A látszólag kompatibilis verziók egységes kezelése a gyakorlatban többlet feladatot jelent a rendszer üzemeltetői számára. Az adatformátumok sokfélesége, és esetlegessége azzal a következménnyel jár, hogy nem minden dokumentumban valósítható meg a tartalom szerinti keresés. A dokumentumok tartalmára csak a meta adatok utalnak. A megvalósult rendszerben lehetőség van tartalom szerinti és meta adat alapú keresésre is. A meta adat alapú keresés lehetővé teszi az azonos témához tartozó dokumentumok gyors összerendelését.

Az információs rendszer adatforrásai (lásd 1. ábra).



1. ábra. Adatforrások

Az üzemeltetést támogató Real - Time rendszer adatai nem kerülnek be eredeti formájukban a hosszú távú adattároló rendszerbe. Az üzemeltetés adatigényéhez képest a megőrzendő adatok kis hányadot képviselnek. A megőrzendő adatok többnyire pillanatértékekből összeállított, hosszabb időszakra vonatkozó átlagok, összegek. Ezek az adatok képezik jórészt a hatósági jelentések alapját is.

A jövő számára a hulladék csomagok adatai a legfontosabbak. A „hulladék csomag” fizikailag egy hordókat tartalmazó konténer, amit a föld alatti tárolótérben helyezünk el.

A dokumentáció tartalmazza egyrészt a konténer gyártására, minőségellenőrzésére vonatkozó adatokat, másrészt a konténerbe kerülő hulladék jellemzőit. Dokumentálnunk kell a konténer végleges helyének térbeli pozícióját is. A technológia informatikai rendszere természetesen folyamatosan regisztrálja a konténer helyét és állapotát, de a végleges elhelyezés után ezeket az adatok hosszú távon nem kell megőrizni. A feladat egy raktár gazdálkodási folyamathoz hasonló, de számos különlegessége is van. A radioaktív hulladék tartalmaz közvetlenül nehezen mérhető elemeket is. Ezek mennyiségére a radioaktív bomlási folyamatok fizikája alapján, számítással lehet következtetni.

A sugárzó anyagok radioaktivitása a spontán bomlás miatt számítható módon, folyamatosan csökken. Ez tehát egy olyan „raktár”, ahol bizonyos mennyiségeket csak számítással határozzuk meg, és a raktárból matematikailag leírható módon „tűnik el” készlet. Az algoritmusokat az RHK Stratégiai Mérnöki Irodája nemzetközileg elfogadott módszerek alapján határozta meg. Az eljárás egy része újdonság az iparágban. A számításoknak a tárolóban lévő összaktivitás meghatározásánál van szerepe, ami a tároló egyik fontos jellemzője. A számítás akkor végezhető el helyesen, ha ismerjük a tárolt hordókban (ez a kezelt legkisebb egység) lévő hulladék kezdeti izotóp összetételét, az azóta eltelt időt, és az algoritmust is.

Egy raktárkészlet lekérdezés annyiban tér el a megszokottól, hogy a kezdeti állapot és az idő függvényében számítással kell meghatároznunk a pillanatnyi állapotot. A beépített eljárások lehetővé teszik az aktuális állapot meghatározását, és alkalmasak egy jövőbeni állapot modellezésére is.

3. A tervezésnél figyelembe vett ajánlások

A HTA rendszer kialakításánál az Open Archival System [2], a Dublin Core [3], MSZ ISO 15836:2004 Dublin Core Metadata elemkészlete [4] és a MoReq [5] [6] elveit vettük figyelembe. A rendszer tervezésénél messzemenően figyelembe vettük az itt fel nem sorolt nyelvi, kódolási, hozzáférés védelmi szabványokat, ajánlásokat is.

A világ vezető számítástechnikai cégeinek ajánlatában szerepelnek hosszú távú adatmegőrzést támogató termékek. A nagy szoftver cégek hosszú távú adattárolást támogató szoftverei számos jó megoldást tartalmaznak, de nehézkes a testre szabásuk, és nem veszik figyelembe a radioaktív hulladékkezelés sajátosságait. Az eljárások általában a könyvtári archiválás követelményeit tekintik meghatározónak. Egyedi matematikai eljárások nehezen, vagy egyáltalán nem integrálhatók a rendszerekbe. A lehetséges gyártó specifikus alternatívákkal szemben az OAIS elveken alapuló, saját fejlesztés választása tűnt a legkedvezőbbnek. Az OAIS szabványos elvi kereteket tartalmazó, széles körben elfogadott megoldás. Több mint tíz éves tapasztalat bizonyítja az OAIS koncepció helyességét. Az EU által támogatott CASPAR [7] projekt is az OAIS referencia modellt alkalmazza. Az iratkezelés, hozzáférés védelem követelményeit a MoReq specifikáció nagyon körültekintően szabályozza. Implementálása sok olyan problémát előz meg, ami a fejlesztők figyelmét esetleg elkerülte volna, és a későbbi alkalmazás során okozna nehézségeket. Az egyedi fejlesztésű, de a szabványokat tiszteletben tartó program a feladat jó megoldásának bizonyult.

Az OAIS főbb funkcionális elemei:

A 1. ábra erősen egyszerűsített vázlata a rendszer működésének.

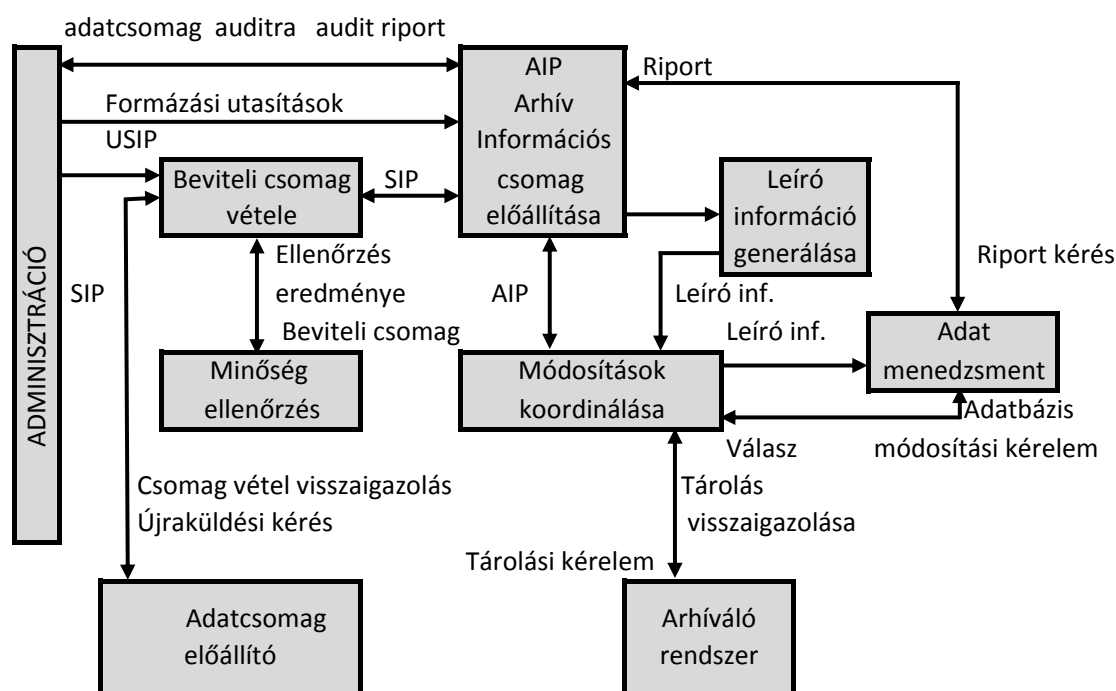
Főbb funkcionális feladatok:

- adatbevitel
- adatállományok minőség ellenőrzése, javítása
- archív adatsomag előállítása
- archív adatsomagok kezelése
- riportok létrehozása
- adminisztráció

A ISO:14721: 2012 szabvány még számos további modult tartalmaz [8], de ezek a működés lényegét nem érintik (pl.:az adott felhasználásban nem tekintettük megvalósítandónak a dokumentumok fizetés ellenében történő elérésének és a pénzügyi tranzakciók kezelésének megoldását).

A gyakorlati megvalósítás során nyilvánvalóvá vált, hogy a minőségbiztosítási és a beviteli modul kapcsolata kiemelt fontosságú. Valós igény a belső használat során a munkaterületen tárolt adatok különböző szempontok szerinti elérése is. A lekérdezések sok esetben jelentős mennyiségű számítással járnak, amit tárolt eljárások valósítanak meg. Ezeket a feladatokat az archív állományokkal nehézkes megvalósítani. A feladatok a munkaterületen oldhatók meg célszerűen.

Az OAIS funkcionális egységei:



Rövidítések:

- SIP Forrás adatsomag (Submission Information Package)
- USIP Módosított forrás adatsomag (Updated Submission Information Package)
- AIP Archív információs csomag (Archive Information Package)

2. ábra. OAIS vázlatos felépítése

Az archív állományoknak jelenleg elsősorban biztonsági szerepe van. A jövőben fontossá fog válni az archív állományok program és operációs rendszer függetlensége, mivel a mai operációs rendszerek és adatbázis kezelők várhatóan már 10-20 év múlva sem lesznek elérhetők, és felülről kompati-

bilitást sem biztosítják a gyártók. A tervezés során olyan megoldásokat kerestünk, melyek valószínűleg használhatók lesznek 50 év múlva is. Vélhető, hogy egy tisztán ASC-II karakterekből álló sorozatot visszaállítható lesz, ha van a hordozót leolvasó, működő hardver. Remélhetjük, hogy egy kétdimenziós vonalkódot olvasni tudó hardver/szoftver együtttest az akkori informatikusok is létre tudnak hozni, ha leírjuk a kód szerkezetét is dokumentáljuk, és elhelyezzük a tudásbázisban.

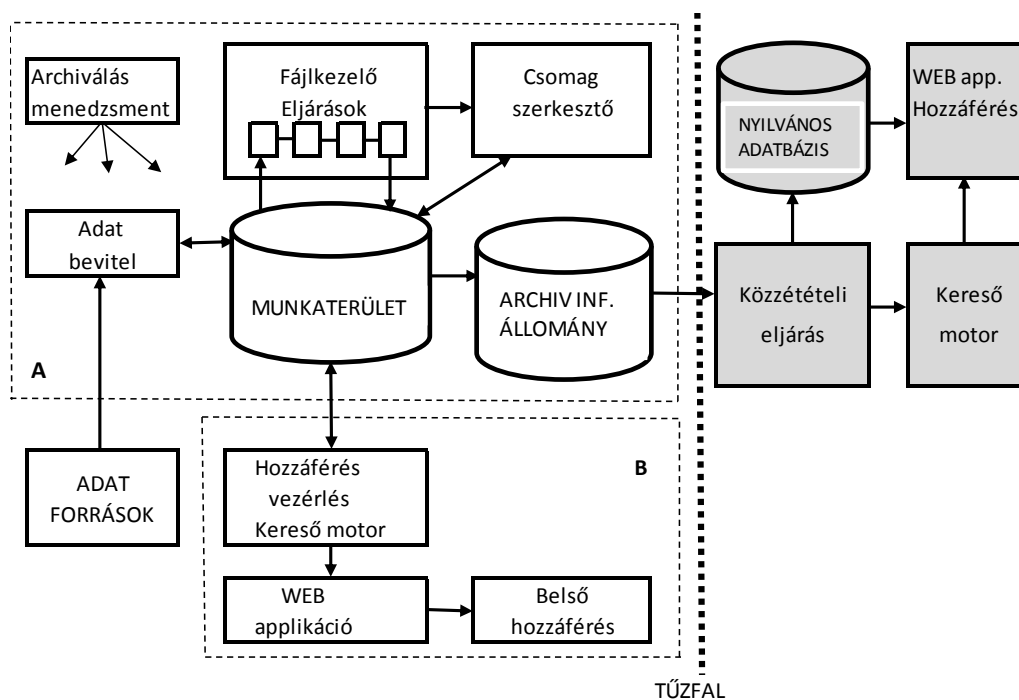
4. Megvalósított rendszer

4.1. Elvi felépítés

A megvalósított rendszer megfelel az OAIS elveinek. Nincs olyan művelet ami a szabványnak el-
lentmond. A jól látható (lásd 1. ábra), hogy az OAIS beviteli moduljának minden funkciója megma-
radt, és bővült a munkaterület funkcionalitása. A munkaterület elérhető WEB felületről is, és megfe-
lelő jogosultsággal módosítható a tartalom is.

A kezelőfelület lehetővé teszi viszonylag bonyolult lekérdezések egyszerű paraméterezéssel történő
megvalósítását, az eredmények grafikus megjelenítését. A egyszerű kezelés és WEB felület támad-
hatósága a hozzáférések precíz szabályozását követeli meg.

Adattárolás sémája:



3. ábra. Hosszú távú adattárolás sémája

Az archív állományok nem védhetők titkosítással, hiszen semmi nem garantálja, hogy 70 év múlva
valaki ismerni fogja a kulcsokat. Az adatok elérésének szabályozása a „címkézett hozzáférés véde-
lem” elvei szerint történik. A dokumentumokhoz és a mappákhoz biztonsági szintjüket meghatározó
címkét rendelünk. Ezt viszik magukkal a fájlok az átalakítások és kódolások során is. Adott bizton-
sági szintű dokumentumok nem helyezhetők el tetszőleges biztonsági szintű mappában. Például egy
alacsonyabb biztonsági besorolású dokumentum elhelyezése egy magasabb biztonsági szintű map-

pában azt eredményezi, hogy a dokumentumot a jogos felhasználók egy része nem fogja elérni. A felhasználókat is biztonsági szintekbe soroljuk. A jogosultságot a felhasználók és az objektumok biztonsági szintjeiből képzett mátrix határozza meg. Ha a hosszú távú adattárolás archív csomagjaiból egy új adatbázist hozunk létre, akkor a biztonsági címkék alapján az eredeti dokumentumok elérési szintjének megfelelő szerkezet helyreállítható.

A felhasználó lehetséges viszonya az objektumokhoz:

- létrehozhat;
- módosíthatja;
- olvashatja;
- megtudhatja a dokumentum tárolási helyét és címét, de nem érheti el;
- csak a dokumentum létezéséről értesül;
- semmilyen információt nem tud meg a dokumentumról.

Egy dokumentum fizikailag egy példányban létezik a rendszerben. Ennek ellenére létrehozhatunk tematikus mappákat is, melyek az ügyszámhoz tartozó valamennyi dokumentumot tartalmazzák. A dokumentumok ilyen esetben is egy példányban tárolódnak, a rendszer csak hivatkozást hoz létre egy már létező dokumentum esetén. Ezzel a dokumentumok tartalmi azonossága biztosítható.

Nyilvános adatbázis létrehozása jelenleg nem kötelező. A nemzetközi együttműködés várhatóan igényelni fogja valamilyen meta adatokra épülő, közösen használható adatbázis létrehozását.

Jelenleg is tartalmazza a rendszer a meta adatokra épülő keresést, ami hozzá kapcsolható egy közös meta adattárat kezelő rendszerhez (Pl.: iRODS, Integrated Rule-Oriented Data System). Ha szükségessé válik, a közös elérést biztosító modul implementálható a meglévő keretekbe.

A meta adattár kialakítása megfelel az MSZ ISO 15836:2004 elvárásainak. A szabvány a kötelező elemkészletet határozza meg, és megengedi az elemkészlet bővítését. A megvalósítás lehetővé teszi, hogy egy grafikus felületen további elemeket hozzunk létre, vagy a meglévők közül válasszunk.

A meta adatok tetszőleges bővíthetősége azért is fontos, mert a radioaktív hulladék tárolók igényeire szabott meta adattár kidolgozása most van folyamatban.

A meta adatok kitöltése az egyik allergikus pontja az adatbevitelnek. A feltöltők nem érzik ennek a fontosságát, és ennek megfelelően kezelik. Mindent meg kell tenni azért, hogy a feltöltők számára a bevitel egyszerűsödjön. A több dokumentumban előforduló közös meta adatokat (pl.: egy mérőhely meta adatai, amiről rendszeresen töltünk fel adatot) célszerű csoportokba rendezni. A feltöltéskor így elegendő a csoportot hozzáadni a meta adatokhoz. A csoportképzést ebben az implementációban az teszi lehetővé, hogy létrehozhatunk hierarchikus elemeket is. A meta adatok csoportosíthatósága a jövőben valószínűleg szabvány követelmény lesz. A NASA módszertani ajánlása már tartalmazza ezt a javaslatot.[8]

4.2. Archiválás adatformátumai

A HTA archív rekordjai majdnem minden ismert adatformátumot tudnak tárolni. A jövőbeni visszaállíthatóság, és megjeleníthetőség követelménye azonban jelentősen szűkíti az ésszerű választékokat. A rajzok és a szöveges dokumentumokat célszerű PDF-A formátumban tárolni. A PDF-A széles körben támogatott, jól specifikált formátum, melyben egységesen tárolhatók képi, rajz és szöveges dokumentumok is. Előny, hogy a PDF a szöveges dokumentumokat kereshető formában is tudja tárolni. A PDF széles körű felhasználás miatt a felületre a jövőben is érdemes lehet megjelenítő szoftvert létrehozni, a felhasználó nem marad egyedül a problémával. A rajzi objektumok kereshető-

sége a PDF formátumban nem valósítható meg. A rajz tartalmát leíró meta-adatok jelentősége itt is jelentősen megnő.

A tervek jelentős része AutoCad –el készült. Az AutoCad állományok kompatibilitása az új verziókkal várhatóan nem biztosítható hosszú távon (ha még létezni fog egyáltalán a jelenlegihez hasonló szoftver termék). A fájlok tartalmának megjelenítése vélhetően csak úgy valósítható meg, ha a teljes szoftver környezetet (virtuális gép, operációs rendszer, felhasználói szoftver) is archiváljuk az adatállományokkal együtt. Erre az archiváló rendszer alkalmas.

A bináris állományok operációs rendszer és hordozó függetlensége nehezen garantálható. A bináris állományokat ezért BASE64 eljárással kódolja a rendszer. Az eredmény így egy csak ASC-II karaktereket tartalmazó fájl, amit nagy biztonsággal kezel szinte minden hardver/szoftver eszköz. A BASE64 ismert, egyszerű, könnyen implementálható algoritmus, amit ma is használ szinte minden levelező rendszer a bináris adatok továbbítására.

4.3. A használt hordozók.

A számítástechnika eddigi története olyan rövid, hogy 75 éves adathordozókkal gyakorlatilag nem találkozunk. A kezdetben használt lyukkártya, vagy lyukszalag élettartama megfelelő lenne, de az alacsony adatsűrűség miatt nem jöhetnek szóba. A gyorsított öregedésvizsgálatokra támaszkodó katalógus adatok szerint az egyszer írható mágnesszalagok megfelelő tárolás esetén teljesítik az elvárt élettartam követelményeket. Hardver oldalról a gyártók ígéretei szerint még 20 évig rendelkezésre fognak állni megfelelő olvasók. Tehát mintegy 30 évig biztosítható a mai szalagok olvashatósága. Ekkor érdemes újragondolni a tárolt anyagok sorsát.

A legfontosabbnak ítélt dokumentumok papír alapon is irattárba kerülnek. Az ember által közvetlenül olvasható dokumentum (nyomtatott szöveg) tartalmát kétdimenziós vonalkóddá alakítjuk, és nyomtatva a dokumentumhoz csatoljuk. A vonalkód lehetővé teszi az eredeti elektronikus dokumentum bit pontosságú, automatizálható visszaállítását. (A nyomtatott szöveg, táblázat optikai felismerő rendszerrel történő visszaállítása nem alkalmas az eredeti digitális dokumentum bit pontosságú visszaállítására). A papír alapú tárolás eddigi története megnyugtatóan biztosítja az elvárt élettartamot.

5. Törlés és módosítás

Az archív információs csomag kezelésénél néhány esetben nehezen egyeztethető össze az archiválás filozófiája a törvényi szabályozással. Az archiválás minden lépése arra irányul, hogy a tárolt anyag változatlan legyen az „idők végezetéig”.

Előfordulhat, hogy valamilyen adatot törvényi szabályozás alapján törölnünk kell a rendszerből. A törlés egy olyan rendszerben, amit arra szerveztünk, hogy az adatok megmaradjanak, nehezen elvégezhető művelet. A munkaterületen a törlés könnyen elvégezhető. A törölt adatokból a dokumentum címe, a törlést elvégző azonosítója, és a törlés jogcíme tárolódik a továbbiakban.

Az archív információs csomagok átmeneti mágnesszalagos tárolóján is elvégezhető a törlés. A mágnesszalagokon már bonyolultabb a helyzet. Az egyszer írható szalag nem módosítható. A megváltozott tartalmú szalag csak úgy állítható elő, hogy újraírjuk a teljes szalagot. Ezzel a lépéssel azonban elveszítettük a hitelességet, hiszen ez már fizikailag egy új állomány, aminek a tartalma az eredetivel nem egyezik. Az eltérés mértéke nem ellenőrizhető, ha az eredetit nem őrizzük meg. Ha megőrizzük, akkor nem semmisítettük meg az adatállományt. Az ilyen eseményekről felvett jegyzőkönyv sokak szára nem meggyőző bizonyíték. Az elvi aggályokon túl a gyakorlatban további prob-

lémákkal kerülünk szembe. Biztos, hogy minden másolatot regisztráltunk és megtaláltunk? Ezekre a kérdésekre nem tudunk egyértelmű válaszokat adni. Látható, hogy a jogi szabályozás is csak akkor működhet, ha a szereplők betartották az ügyrendet. A felírt mágnesszalag rendszeren kívüli objektum, ami a rendszer eszközeivel nem kezelhető. Az ügyrend maradéktalan betartása a gyakorlatban ellenőrizhetetlen. Az elektronikus dokumentumok illetéktelen másolása a tapasztalatok szerint nem akadályozható meg teljes biztonsággal, így a törlések mindig tartalmazznak bizonytalanságot.

6. Eredmények

A HTA rendszer üzemeltetési tapasztalatai alapján megállapítható, hogy a kialakított rendszer megfelel a jelenleg elvárható műszaki színvonalnak, és kielégíti a céltervben meghatározott követelményeket. Várhatóan alkalmas a dokumentumok megőrzésére és megjelenítésére a tervezett időtávon. A továbbfejlesztés lehetséges irányai:

Célszerű lenne a meta adatok bevitelének fejlettebb támogatása, például sablonok létrehozása gyakran előforduló dokumentum típusokhoz. Az adatsorokhoz tartozó meta adatokat célszerű lenne csoportokba rendezni. A meta adatok csoportba rendezésének elve megjelent a NASA ajánlásai között is [8]. A meta adatok hierarchikus felépítése a megvalósított rendszerben már lehetővé teszi csoportok létrehozását, de nem támogatja a sablonok alkalmazását.

Az ismert, nagy adattároló rendszerek fejlesztése a dokumentum táruk összekapcsolására, az információ közös felhasználására irányul. A programcsomag ezekre a későbbi feladatokra interfész felületet biztosít, és lehetővé teszi a megvalósítást a létrehozott rendszer megtartása mellett.

Irodalomjegyzék:

- [1] Országos Atomenergia Hivatal : A dokumentálás szabályozása és nyilvántartások 2006, verzió2.
- [2] OAIS ISO:14721:2003, 2012 Reference Model
- [3] Dublin Core Metadata Initiative. ISO 15836:2009
- [4] MSZ ISO 15836:2004 Dublin Core Metadata elemkészlete
- [5] European Commission Archival Policy : Model Requirements for the Management of Electronic Records (MoReq), an Commission by Cornwell Management Consultants plc 2002.
- [6] European Commission Archival Policy: MoReq2 specification, 2010
- [7] CASPAR, Cultural, Artistic and Scientific knowledge for Preservation, Access and Retrieval Doc.Identifier: CASPAR-D1101-TN-0101-1_0, Submission Date: 15-05-2007.
- [8] Consultative Committee for Space Data Systems: CCSDS recommended practice for an OAIS Reference Model, CCSDS 650.0-M-2 Page F-2, 2012
- [9] NASA Earth Science Data and Information System (ESDIS), Goddard Space Flight Center (GSFC) Code 423.0 : NASA ESD Base Metadata Requirements_V1_20110922_0.doc

OLVASÁS 2.0 – AZAZ AZ OLVASÁS FELVILLANYOZ

az e-könyvek helye az oktatásban és a könyvtárakban

READING 2.0 – THE READING GETS ELECTRIFIED

the place of e-books in the education and in the libraries

Bátfai Erika^a, Bátfai Norbert^b, Kőrösi Krisztina^c

^aDebreceni Egyetem Egyetemi és Nemzeti Könyvtár

ebatfai@lib.unideb.hu

^bDebreceni Egyetem Informatikai Kar

batfai.norbert@inf.unideb.hu

^cCEU Library

Korosik@ceu.hu

Absztrakt: Az első e-könyvet 1971-ben írta Michael Hart. A netgeneráció életének (és így nyilvánvalóan a tanulásának/oktatásának is) része az elektronikus irodalom.

Mi a feladata az egyetemi könyvtáraknak? Kizárólag az állományokba integrálódó e-könyvekkel törődjenek vagy inkább/és a tartalomfejlesztésük kiterjedjen-e erre a területre is? Egyetemi könyvtárként feladatunk továbbá a digitális írástudás közvetítése az eszközkölcsönzéstől az e-könyvek-beli hivatkozás tanításán át a szabad gyűjtemények megismertetéséig.

Meglátásunk szerint a könyvtárak az oktatókkal karöltve az oktatás segítésének egyik új eszközeként használhatják az e-könyveket és az e-könyv-olvasókat; szó szerint zsebkönyvtárat szolgáltatva a hallgatók számára.

Jelen előadásban néhány példán keresztül (CEU Library, DE IK) azt vizsgáljuk, hogy jelenleg, 2014-ben, hogyan jelenik meg az e-könyv a hazai könyvtári állományban, a könyvtár tartalomfejlesztőként, támogatóként az e-könyves területen és milyen szabad gyűjteményeket használhatunk az oktatásban.

Kulcsszavak: e-könyvek, egyetemi könyvtárak, oktatástámogatás, e-könyv-kölcsönzés, eszközkölcsönzés, felhasználóképzés

Abstract: The first e-book was written by Michael Hart in 1971. Electronic literature is part of the net generation's life including teaching and learning.

What is the main role of a university library? Do they only have to deal with integrating e-books to their catalogues or serve as a content provider? As a university library we should concentrate on transmitting digital literacy, including device lending, teaching the citation of e-books and drawing attention to free e-book sources.

We think, that university libraries with Faculty members together will use e-books and e-book devices as new tools in education that will serve as a pocket library for students.

In this article we examine how are e-books present in Hungarian libraries (CEU Library, DE IK) in 2014, which open sources are available for teaching and learning, how the libraries will serve this new situation.

Keywords: e-books, university libraries, e-books lending, device lending, patron training

1. Bevezetés

A Gutenberg-galaxis válságáról hallhatunk évtizedek óta; miközben az írott kultúra újabb győzelmet aratott – amennyiben az írottat nem papírra írottként értelmezzük. Sebestyén György az elektronikus írásbeliséget egyenesen a harmadik forradalomnak nevezi (az alfabetikus írás és a sokszorosított könyvnyomtatás után).[1]

Az e-könyv tömegeket adott vissza az olvasásnak [2]. Jelen dolgozatban azt vizsgáljuk, hogy ezek a tömegek miként csábíthatók a könyvtárakba; hogyan jelenhet meg az e-könyv a könyvtári állományban és a könyvtár tartalomfejlesztőként, -támogatóként az e-könyves területen.

Meglátásunk szerint a könyvtárak, különösen az egyetemi könyvtárak, az oktatókkal karöltve az oktatás segítésének egyik új eszközeként használhatják az e-könyveket és az e-könyv-olvasókat; szó szerint zsebkönyvtárat szolgáltatva a hallgatók számára.

Jó ideje *digitális hajléktalanok*[3] vívják harcukat a *netgeneráció*[4] homo informaticus-aival[5], akiknek az információfeldolgozási sebessége – a gépekéhez hasonlóan – felgyorsult. Van, akik szerint nem csupán új embertípus, de új társadalmi osztály jelenik meg: a *netokrácia*, aminél az információ az új tőke, életterük virtuális, miközben lakhelyük lehet bárhol. Mivel a netokraták elsősorban önmagukért küzdenek, a klasszikus nemzet, nemzetállam, nemzetgazdaság berendezkedés értelmét veszti... Akik osztályon kívüliek, azok szinte nem is léteznek, sorsuk sanyarú, főleg, mert a legjobbjai folyamatosan beolvadnak a nemzetközi netokráciába...[6]

Mindeközben az információhordozók, az azokra való igény és a használatukhoz vezető szocializáció változnak – amire a könyvtáraknak fel kell készülni, ahogy ezt évezredek óta meg is teszik.

2. Kihívás – az e-book

Az e-könyvek területén egyértelmű lemaradásban vagyunk - közben egyre többet észlelünk és olvashatunk a frissebb nemzedékek megváltozott, felgyorsult észlelésről; találkozhatunk az olvasás 2.0 fogalmával is. [7]

Nézzük meg, most hogyan kerülhet e-könyv egy hazai könyvtárba:

1. Az e-könyvek egy része szabadon használható, gondoljunk pl. a [Projekt Gutenberg](#)-re, a [DigiLibraries](#)-re, [Free-ebooks](#)-ra, [Google Books](#)-ra, [BookZZ](#)-ra; a [MEK](#)-re. Ezeket a legtöbb esetben „csak” használja a könyvtár, esetleg a meglévő hagyományos példány rekordján át elérhető egy kapcsoló mező (marc 21: 856) beillesztésével; illetve könyvtárközi kérésnél felajánljuk a kérőnek az e-változatot.
2. Másik nagy része aggregátorokon, előfizetés ellenében, weben keresztül érhető el. Az oktatásba, tanulásba ezek nyilván szakterületfüggően épülhetnek be, a térhódítás azonban ténykérdés mind a lakossági penetráció[8], mind könyvtári viszonylatban[9].
3. Egyre több hazai kiadó, könyvforgalmazó kínál e-könyveket, azonban ezek itthoni könyvtári használata még gyerekcipőben jár.
4. Az „on Demand” szolgáltatások a pillanatnyi olvasói igényt kívánják kiszolgálni. Ilyen on Demand másolatszolgáltatás az [eBooks on Demand](#), amiben jelenleg az OSZK és a Magyar Tudományos Akadémia Könyvtára vesz részt. Könyvtárak és olvasók, kutatók számára kétségtelenül előnyös szolgáltatás, ám a használók köre (az 1500-1930 közötti kiadási intervallum miatt) nem számottevő.
5. Saját digitalizáló projektekkel (ritkán munkakörben, leginkább pályázati forrásokból), digitális gyűjtemények építésével a tartalomfejlesztést célozzák meg az intézményeink -

ám gyakran ez nem túl jó minőségű .pdf-eket jelent; valamint nem feltétlenül a széles olvasói igényeket szolgálják ki.

3. Könyvtári kezelésük

Mit kezdünk az e-könyvekkel kapun belül:

- **Készüljön-e katalógusbeli leírás?** Ha igen, akkor jelenleg a 2000-ben elkészült *KSZ/2 Bibliográfiai leírás: elektronikus dokumentumok: könyvtári szakirodalmi tájékoztatói szabályzatot* használjuk. Az azóta született egyéb elektronikus anyagokkal lehetne frissíteni (pl. elektronikus tananyagok, flipbook-ok); e szabvány szellemében nem túl tökéletes leírásokat készíthetünk, és ami ennél nagyobb gond, hogy *a visszakeresést sem szolgálják* az így megszületett tételek. Következő kérdés, hogy **milyen adatmezőket** használunk. A DEENK által évek óta szervezett [Elektronikus dokumentumok katalógizálása](#) c. tanfolyam aktualizálása kapcsán új belső szabályzatot készítettünk. Ennek értelmében a számítógépfájl jellemzőkbe (marc21 256-os mező) írjuk az „ekönyv” megjelölést, ez esetben írunk fizikai jellemzőket (300 \$aElektronikus dokumentum (59 p.)), a rendszeradatoknál (538 \$aazw, x-ray) feltüntetve, milyen formátumú az adott könyv és mely egyéb technikai jellemzői vannak (mindezek mellett a rekordfejben (007) az eddigi gyakorlatnak megfelelően kódoljuk a fizikai jellemzőket).
- Amennyiben **nem**, akkor is nyilvánvalóan szolgáltatjuk ezeket, sarkalatos pont, hogy milyen kölcsönzési politikával, valamint, hogy mennyire tanítsuk olvasóinkat a használatra. Az e-könyvek-beli pontos hivatkozás, az x-ray, a kiemelések/jegyzetek letöltési lehetőségei – mind olyan új ismeretanyag, amit feltétlenül és könyvtárként közvetítenünk kell; egyben belső továbbképzéssel a kollégákat is felkészíteni.
- Hogyha a könyvtár **kölcsönözni kívánja**, akkor ez belső hálózaton vagy példánykölcsönzéssel működjön?
 - Ha példánykölcsönzéssel, akkor milyen modullal oldjuk meg? Ha webalapú-kölcsönzés, saját fejlesztés vagy valamelyik „nagy” szolgáltatóhoz csatlakozzon az intézmény, ahol viszont a dokumentumvásárláson felül borsos fenntartási költséggel találkozhatunk? Meddig lesznek a könyvtár tulajdonában az így „megvásárolt” könyvek?
 - Legyen hálózati-alapú (is)?
 - Használatalapúnál kit terheljen a költség?
 - Intézményhez kötött nagyvonalakban a kicsit kiterjesztett kolostori könyvtárakat idézi – ám ahogy pl. a CEU-s gyakorlat mutatja, annak (is) van létjogosultsága.
- **Kell-e kölcsönözni?** Érdemes elgondolkodni azon is, kell-e kölcsönöznünk e-könyveket; ill. valós tömegigény van-e erre?

Kétségtelen, hogy az e-könyvek miatt nem csak új kiadói[9], de új könyvtári modellek is születnek, kezdve a tárolásuk/kölcsönzésük új politikájával.

3.1. Néhány hazai példa

3.1.1. Eszközkölcsönzés

A könyvtár a digitális írástudás elsajátításának helyszíne is, feladata a digitális írástudás közvetítése, aminek része az eszközökkel való megismertetés. Erre több példát is találunk a hazai gyakorlatban:

- Az **OSZK**-ban *2010 decemberétől* 10 db. helyben használható, MEK-es tartalommal feltöltött My EBook e-könyv-olvasóval ismerkedhetnek az olvasók.
- *2012.04.16-ától* a **BGF Gazdálkodási Kar Zalaegerszeg Könyvtár**ában elindult e-könyv-olvasók (36 db Prestigio PER3162BN) kölcsönzése. Az olvasók folyamatosan forgalomban vannak, jelenleg a szolgáltatás tabletekre való kiterjesztését tervezik.
- A **DEENK Társadalomtudományi Könyvtár**ban inkubátor szolgáltatásként vezettük be Kindle Paperwhite-k kölcsönzését *idén januártól*[10]. A pozitív tapasztalatok fényében a szolgáltatás bővítését tervezzük a hallgatók felé (is), lehetőleg minél több egyetemi tartalommal.

3.1.2. ELDORADO

Reménykedhetünk az ELDORADO-ban, jelenleg úgy látszik, hogy (várhatóan a [Digitális Könyvtár](#) tapasztalatait felhasználó) olyan akadálymentes, webes felületről elérhető közös könyvtári szolgáltatás lesz, amivel könyvtári beiratkozás nélkül akár otthonról is kereshetők a hazai könyvtári rendszerben meglévő anyagok (kiv. audio- és AV), megrendelhető digitális másolatuk, lehetséges online használatuk (streaming), letöltésük vagy akár e-könyvként is kölcsönözhetjük az e-formában meglévőket, valamint értéknövelt szolgáltatásokat kérhetünk róluk (pl. PoD, OCR), mindezen folyamatokba a jogtisztázás is helyet kap. Maga a szolgáltatás könyvtári eHub-ként a MOKKA-ODR katalógusra épül majd[11].

Mely művekre gondolhatunk? A szerzői jogvédelem alatt állókra, az árva művekre, a kereskedelmi forgalomba nem állókra, a szerzői jogi védelem alatt nem állókra – mindezek pontos behatározására munkacsoportok dolgoznak, a fő hangsúlyt a „szigorú jogosultságkezelésre és az adatbiztonságra”[12] helyezik. Kellőképpen hangsúlyozva, hogy a súlypontot javarészt a nehezen beszerezhető, a kereskedelmi forgalomban nem kapható kiadványokra helyezik – azaz a legoptimálisabb 2014 őszi induláskor is marad tartalmi piaci rés.

3.1.3. E-Könyv-kölcsönzés

E-könyvek a CEU Könyvtárában

(Előljáróban leszögeznénk, hogy a fejezetcím a CEU Könyvtár esetében nem állja meg a helyét, mert jelenleg nem kölcsönözzük e-könyveinket. Ennek két oka van: adatbázisokból nem kölcsönözhetünk, egyedileg megvett könyvek esetében pedig az a célunk, hogy az itteni kurzusokhoz szükséges könyvekhez állandó elérést biztosítsunk egyidejűleg, minden diák számára.)

Az e-könyv fogalmát bővebben értelmezzük, mint a hagyományos felfogás. Számunkra nemcsak azok az e-könyvek, melyeket e-könyv olvasóval lehet olvasni, hanem beleértjük az összes elektronikus formában terjesztett könyvet.

E-könyves állományunkat kb. 10 ezer könyv jelenti, ennek nagy többsége adatbázis-előfizetésből származik, míg kisebb részét az egyedileg megvett e-könyvek teszik ki. Előfizetett adatbázisaink közül kifejezetten e-könyveket tartalmaznak az ACLS Humanities, valamint az Oxford Scholarship Online (OSO), de szép számmal találunk e-könyvet ezekben az adatbázisokban is: Business Source Complete, CIAO, OECD Taxation, Science Direct, Westlaw UK. Az adatbázisok (és minden más elektronikus forrás) IP-cím alapján használhatóak, távolról is. Az adatbázisokban lévő e-könyvek fejezetenként letölthetőek, lehet belőlük nyomtatni, nem sújtja őket a DRM, de kölcsönzést nem tesznek lehetővé.

2011-ben kezdődött el az e-könyvek egyedi beszerzése. Az igény a tanszékek részéről merült fel, illetve „kötelező olvasmány”-ként vásárolt a könyvtár e-könyveket három nagyobb szolgáltatótól: DawsonBooks, EBSCO (Netlibrary), ill. Proquest (Ebrary). Egyedileg megvett e-könyveink többségét korlátlan számú felhasználó használhatja egyidejűleg; de van néhány, melyet három felhasználó, illetve egy felhasználó használhat. Mindegyik IP-alapon férhető hozzá, kereshetőek a Comprehensive Search-ben. A rendelést követően megkapjuk a MARC-rekordokat, ezek bekerülnek a katalógusba. Az e-könyvek használatának mérését a szolgáltatóktól kapott statisztika segíti. Lenne rá lehetőség, hogy az ezeket az e-könyveket letöltsék maguknak a diákok (kikölcsönözzék), de akkor a kölcsönzés alatt a többi diák nem férne hozzá, így pont az az elv sérülne, hogy a kötelező irodalom mindig mindenki számára elérhető legyen.

Össességében közel 60 e-könyvet vásárolt eddig az intézmény e három szolgáltatótól és ezeket az elmúlt tanévben átlagosan 9-15 alkalommal használták. Ez igen szép mutató, de nem szabad arról elfeledkezni, hogy ezek mindegyike kurzushoz kötődő kötelező irodalom. Hátrányként említeném meg, hogy a könyvekből a lementhető oldalak száma és a nyomtatás mennyisége is limitált.

A két fő forráson kívül az elmúlt években beszereztünk még mintegy 50 Kindle-könyvet is. Ennek többsége könyvtárközi kölcsönzés keretében merült fel igényként, de nem aratott osztatlan sikert a felhasználók körében. Egyrészt, mert a hallgatók a saját Kindle-jükre szerették volna letölteni a könyvet, ez nem megengedett, s így a jegyzetelés sem megoldott. Másrészt a Kindle-készülékek lineáris olvasást tesznek lehetővé, nem látni egyszerre az adott oldalt és a könyv teljes struktúráját. A Kindle-könyvek valójában nem könyvtári célokat szolgálnak[13][14].

Üzleti kiadványok esetében megint másképp kellett eljárunk, mivel ezek egyik jól bevált szolgáltatóknál sem voltak hozzáférhetőek. A CEU Business School kurzusain tanulók a tananyagot a CourseSmart elnevezésű e-learning felületen érik el, melyhez hozzáférési kódot kapnak a tandíj fejében. Az online felületen nemcsak a könyvhöz férnek hozzá, hanem kiegészítő anyagokhoz, példákhoz is. A beszerzésben a könyvtár nem játszik aktív szerepet, s ezek az e-könyvek csak a BS diákjai számára érhetőek el. Kölcsönzésre itt sem kerül sor, mivel a hozzáférési kóddal bármikor elérhetik e-könyveiket.

Jövőbeni terveink, hogy növeljük e-könyveink számát, valamint, hogy továbbra is tartjuk a megkezdett irányt: 24 órás hozzáférhetőséget biztosítunk, de nem kölcsönözzük e-könyveinket. E-könyveink számának bővítése érdekében 2014 februárjában bekapcsolódtunk a Knowledge Unlatched-projektbe. Induláskor közel 300 könyvtár csatlakozott a projekthez: együttesen veszünk meg kiadóktól e-könyveket, a kiadó által megszabott áron, s a könyvet szabadon hozzáférhetővé tesszük az OAPEN-en keresztül.

A Goethe Intézet Könyvtára és a Francia Intézet Médiatára

Idehaza elsőként a [Goethe Intézet Könyvtára](#) indította el az e-könyv-kölcsönzést 2011 szeptemberében [15]. A szolgáltatást bárki igénybe veheti, aki állandó magyarországi lakcímmel rendelkezik és a Goethe Intézet könyvtárának regisztrált tagja (hasonlóan működik a Francia Intézet Médiatárának kölcsönzése is). A DiViBib kínálatából a Goethe Intézetek önállóan válogatnak az [onleihe](#) szolgáltatásába; a kemény DRM-mel ellátott .epub formátumú könyvek mellett folyóiratokat és AV anyagok is kölcsönözhetőek; a dokumentumok többnyire egy felhasználósak.

A [Francia Intézet Médiatárának Digitális Könyvtár](#)ából a NumiLog-on keresztül 2013 óta kölcsönözhető e-könyvek, a többi e-dokumentum (pl. folyóirat, film) stream módban használható. Az intézet a gyűjtőköréhez igazodva válogat a kínálatból, ill. hozzáférnek az anyaintézmény által előfizetett címekhez is; a gyűjteménybe került anyagok védelme szintén erős.

Mindeközben hazánkban többszázezer e-könyv-olvasót használnak (egy nagyobb egyetemi könyvtár beiratkozott olvasóinak a száma), tabletek, egyéb okoskészülékek beleszámításával ez a szám mára 2.5 millió feletti[16].

4. E-könyv-tár

Mindazonáltal mire számítottunk? A könyvek jó ideje számítógéppel íródnak, távmunkában végzik a lektorálást/tördelést, a komplett könyvtári munkafolyamataink is évtizedek óta ILL-kel működnek – várható volt, hogy mindez nem szimpla gépesítés lesz. És vártuk is, hiszen a könyvtár 2.0 él (gondoljunk pl. a san antonio-i [BiblioTech](#)-re).

4.1. DocBook könyvek

Az elmúlt évtized bővelkedett az egyetemi közegben elérhető tankönyvfejlesztési pályázatokban (pl. a Nemzeti Kutatási és Technológiai Hivatal, DIGITÁLIS SZAKKÖNYV, DIGIT 2005; TÁMOP-4.1.2-08/1/A-2009-0046 Kelet-magyarországi Informatika Tananyag Tárház; TÁMOP-4.1.2.A/1-11/1-2011-0103 Gyires Béla Informatika Tananyag Tárház, TÁMOP-4.1.2.A/1-11/1-2011-0063 Sokprocesszoros rendszerek a mérnöki gyakorlatban). Ezek közös jellemzője, hogy az eredménytermékeknek (azaz a könyveknek) DocBook XML formátumban kellett előállniuk[17]. A mi meglátásunk szerint ez remek és időszerű kezdeményezés volt, bár megjegyezhetjük, hogy ugyanúgy a DocBook XML 4.4 szerinti validitást kérni, azaz ennek a verzióknak megfelelően elkészíteni a könyvet 2005-ben és 2011-ben is, magyarázatra szorul. Ennek oka szerintünk az lehetett, hogy a pályázati eredmények elérhetőségét szolgáltató apparátus (www.tankonyvtar.hu) az első pályázat ideje alatt ennek a változatnak a kezelésére készült fel, s nem tudott alkalmazkodni a folyamatos változásokhoz. A DocBook XML használatának ide vonatkozó aspektusa, hogy a főleg informatikai-tartalmi annotálásban erős jelölőnyelvi szövegnek számos transzformációja elérhető (például böngészhető HTML, nyomtatható PDF, MS Word DOC, e-könyv-olvasóra .epub vagy .mobi). Korábban az HTML transzformáció volt a normatív a Tankönyvtárban, ami adott esetben kívánni valót hagyott maga után, ennek egyik oka a megjelenítő transzformációk, megjelenítő szoftverek fejletlensége is volt. (Az O'rielly kiadó támogatja a DocBook formátumot, sőt a *DocBook 5.1*[18] jó precedens arra, hogy – igaz ritka esetekben, de ebben éppen – az informatikus is olvashat naprakész könyvet, hiszen a könyv maga a szabvány bemutatása).

Napjainkra már a forrás XML állományt és annak erőforrásait is közzéteszik a Tankönyvtárban, így elvben van lehetősége a hozzáértő érdeklődőnek, hogy a konverzióit a maga személyére szabja. Teszik ezt több esetben maguk a szerzők is, könyveik úgynevezett “szerzői kiadásában”, az e-könyv-olvasós tesztjeinket mi is ilyen kiadásokra alapoztuk, konkrétan a következő listába szedett könyvekre:

- *Párhuzamos programozás GNU/Linux környezetben: SysV IPC, P-szálak, OpenMP*
- *Paternoster of Programmers Reloaded: C, C++, Java, Python and AspectJ Case Studies*
- *Programozó Páteroszter újratöltve: C, C++, Java, Python és AspectJ esettanulmányok*
- *Mesterséges intelligencia a gyakorlatban: bevezetés a robotfoci programozásba*
- *Mobil programozás*
- *Javát tanítok*
- *A Debreceni Egyetem programozói évkönyve [UDPROG][19]*

(A Tankönyvtárba való kikerülés sokszor éveket vesz igénybe, mivel azonban az elkészült könyveket az szerzők használják az egyetemi kurzusaikon, ezért pl. a *TÁMOP-4.1.2-08/1/A-2009-0046 számú Kelet-magyarországi Informatika Tananyag Tárház* projekt keretében készült tankönyvek „házon belüli” használatát elősegítendő WebPac katalógusbeli rekordok is készültek.)

Az említett könyvek – saját terminológiájuk szerint – közösen egymás környezetét alkotják, s szerzői kiadásukat nem csupán a megjelenítésük igénye, hanem a folyamatos naprakészen tartásuk is indokolja.

Informatikai könyvek lévén ez alapvető fontosságú, sőt a szerzői kiadásokkal nem kizárólag egy szükséges API frissítéséhez tudjuk módosítani a példákat, hanem akár egy tárgyévfolym aktuális igényeihez is szabhatjuk az adott könyv mindenkori tartalmát. A nyers XML forrásokat nem tesszük közzé a jegyzetek környezetében (de ezek a Tankönyvtárban egyébként is elérhetőek), hanem csak az xsltproc-kal legenerált HTML kimenetet, a többféle stílusban a dlatex-el legenerált PDF illetve EPUB formátumokat. Az UDPROG könyv a legfrissebb DocBook tankönyvünk, ezzel végeztük a legtöbb kísérletet.

4.2. A Debreceni Egyetem programozói évkönyve

Ez a könyv sok mindenben eltér az említett környezetétől. Fontos, hogy ezt nem pályázati támogatásból, hanem önálló érdeklődésből készítjük. Nem 4.4 formában, hanem a legaktuálisabb 5.1 szabványnak megfelelően. Magát a könyvet még a klasszikus könyv tagadásának is felfoghatjuk: nem egy vagy kevés szerző ír egy könyvet egy szélesebb közönségnek, hanem sok szerző (gyakorlatilag az aktuális évfolyam) tipikusan a szerzői körnél alig szélesebb közönségnek. A könyv tulajdonképpen egy feladatkiírás- és megoldásgyűjtemény, amit évkönyv jellegű funkciókkal és a kurzusszervezési információkkal is tömünk, hiszen ez a dokumentum az alapja a Debreceni Egyetem *Magas szintű programozási nyelvek* kurzusainak. Funkcionális megközelítésben az évkönyvet a neki megfelelően fejlesztett szoftverek egyfajta indexének is felfoghatjuk. Ez a [YouTube videó](#) látványosan mutatja a könyvön történő valójában konkurens, a Git verziókezelő támogatata munkát.

4.3. Oktatási tapasztalatok

A szóban forgó környezetet (kiemelten az UDPROG könyvet) egy DPS 800 és egy Kindle Paperwhite gépen teszteltük a Debreceni Egyetem *Magas szintű programozási nyelvek* kurzusai közben, illetve némely laboron és vizsgán. Ezek a kurzusok alapvető és bevezető, de “hardcore” programozás tárgyak. A mi előadásunkban alaptulajdonság, hogy bármit és bármikor lehet használni; a vizsgázás különösen elképzelhetetlen internetes és programozásra is alkalmas (ezért legfőképpen Linuxos) számítógép használata nélkül.

Legfontosabb tapasztalatunk, hogy a tipikusan szekvenciális olvasású regényekhez képest az e-ink alapú könyv olvasó eszközök adta felhasználói élmény elvész. Ennek fő oka, hogy tankönyv/szakkönyv lévén jellemzően gazdag kereszthivatkozásban, ami az ezek menti (nem szekvenciális, hanem programozott sorrendű) olvasást indikálja. A konkrét olvasók esetében a Kindle a **.mobi** állománnyal már képes követni a dokumentum belső linkeit, de az élmény elmarad pl. a PDF PC-s olvasási élményétől.

Megoldás lehet, hogy az említettél lényegesen kisebb egységeket érdemes célzottan elektronikus könyv olvasóra kiadni, ilyen lehet akár egy esszé szerűen kidolgozott vizsgatétel.

4.4. Könyvtári szolgáltatás

Az egyetemi szintű gyűjteményépítés látszik hosszabb távú megoldásnak, amiből akár a hallgatók maguk állíthatják össze az adott kurzushoz szükséges oktatási csomagjukat.

Ajánlatos gondolni a letölthetőség mellett a streaming olvasás biztosítására is, mivel a hallgatók nem feltétlenül a könyvbeli jegyzetelést preferálják, ill. nem minden anyag igényel több órás folyamatos olvasást. Az on-line szolgáltatás a formátum- és eszköztöredezettséget is enyhíti.

A vázolt szolgáltatás újdonsága az oktatók munkájának segítése, valamint a hallgatók is könnyebben boldogulnak egy kész csomaggal, aminek az aktualitását a könyvtár garantálja. Mindezek mellett a könyvtár a digitális írástudás elsajátításának helyszíne is, így feladatunk az olvasókat megismertetni mind az e-könyv-olvasókkal, mind az olvasáshoz municiót biztosító szabad gyűjteményekkel.

A szabadon elérhető kurzuscsoomagok előnye, hogy a jegyzetek intézményeken túlnyúló oktatástámogatási forma, egyben ingyenes egyetemi reklámja az adott kurzusnak, képzésnek.

5. Összegzés

Az e-könyv olvasóknak köszönhetően új értelmet nyer a zsebkönyvtár is, hiszen ennek az eszköznek köszönhetően a könyvtárunk a zsebünkbe költözhet. Eközben a könyvtár tervezi az oktatók és hallgatók munkáját egyaránt hatékonyan segítő oktatási gyűjtemények építését.

Jelenleg a hazai könyvtárak tömegesen szeretnék beszerezni és szolgáltatni az e-könyveket, ám a meglévő jogi, gazdasági (ÁFA[20]) és kiadói környezetben elnagyolva két csoportot láthatunk:

- Ahol szeretnének, de még nincs/csak külföldi nagy gyűjtemény hálózati alapú szolgáltatása van (pl. CEU).
- Ahol valóban van napi szintű e-könyv-kölcsönzés (ezekben az esetekben nem állami/önkormányzati fenntartású intézményekről beszélhetünk, amiknek

többnyire az anyaintézménye egy-több külföldi szolgáltatóval áll szerződésben: pl. Goethe Intézet, Francia Intézet).

Meglátásunk szerint az egyetemi könyvtárak a tartalomépítéssel áthidalhatják a meglévő problémák megoldásáig hátralévő időt, akár katalizálva is a folyamatot – mindeközben érdemes átgondolni a szolgáltatás modelljét is.

Irodalomjegyzék

- [1] Sebestyén György: *A Gutenberg-galaxis és a digitális kultúra szintézise: az elektronikus könyvtár*. In <http://epa.oszk.hu/00400/00413/00001/11sgy.html> (2013.05.03.)
- [2] Kerekes Pál – Kiszl Péter – Takács Dániel: *E-könyvészet: a digitális könyvkultúra alapvonásai*. Budapest: ELTE BTK KII, 2013. p. 11-18.
- [3] Tapscott, Don: *Digitális gyermekkor*. Budapest: Kossuth, 2001. p. 353-354.
- [4] V. ö. Tapscott: i. m. p. 25-47.
- [5] V. ö. Mérő László: *Az élő pénz*. Budapest: Tericum, cop. 2004. p. 119-124.
- [6] V. ö. Bard, Alexander - Söderquist, Jan: *Netocracy*. Stockholm: 2002. p. 51-72.
- [7] Merkoski, Jason: *Burning the page: the ebook revolution and the future of reading*. Naperville: Sourcebooks, 2013. [Kindle ed.] Loc. 2112
- [8] *E-Reading rises as device ownership jumps*. http://www.pewinternet.org/files/old-media//Files/Reports/2014/PIP_E-reading_011614.pdf [2014.02.05.]
- [9] Henke, Harold: The global impact of ebooks on e-publishing. In *Proceedings of the 19th annual international conference on Computer documentation (SIGDOC '01)*. New York: ACM, 2001. p. 172-180.
- [10] *Kindle e-könyv-olvasó kölcsönzése*. http://hu.wiki.lib.unideb.hu/index.php/Kindle_ek%C3%B6nyv-olvas%C3%B3_k%C3%B6lcs%C3%B6nz%C3%A9se#1_k.C3.B6lcs.C3.B6nz.C3.A9si_politika [2013.12.29.]
- [11] Dancs Szabolcs: *ELDORADO*, *végállomás*. http://tmt.omikk.bme.hu/show_news.html?id=5876&issue_id=556 [2014.04.17.]
- [12] Amberg Eszter: *Az ELDORADO-projekt szerzői jogi aspektusai*. http://tmt.omikk.bme.hu/show_news.html?sid=a2c3a13ab71a7f8ea254222eb27e7cc6&id=5877&issue_id=556 [2014.04.17.]
- [13] Goodwin, Susan – Shurtz, Suzanne – Gonzalez, Adriana – Clark, Dennis: *Assessing an e-reader lending program: from pilot to mainstream service*. <http://www.emeraldinsight.com/journals.htm?issn=0024-2535&volume=61&issue=1&articleid=17009937&show=html&PHPSESSID=0f3qsav9f90t5o2st7s0cumid7> [2014.03.10.]
- [14] *Amazon's Kindle Swindle*. <http://www.defectivebydesign.org/amazon-kindle-swindle> [2014.03.10.]
- [15] *Onleihe e-kölcsönzés*. http://www.goethe.de/ins/hu/bud/kul/bib/onl/huindex.htm?wt_sc=budapest_e-kolcsonzes [2014.02.06.]

- [16] eNET – TELECOM: *Már okostelefon-felhasználó a magyar lakosság több mint ¼-e.*
<http://www.enet.hu/hirek/mar-okostelefon-felhasznalo-a-magyar-lakossag-tobb-mint-%C2%BC-e/?lang=hu> [2014.05.20.]
- [17] Markója Szilárd: *A Felsőoktatási Digitális Tankönyvtár és a DocBook XML leíró nyelv.*
<http://epa.oszk.hu/00100/00143/00055/markoja.html> [2014. 05.21.]
- [18] Walsh, Norman: *DocBook 5.1: the definitive guide.*
<http://docbook.org/tdg51/en/html/docbook.html> [2014.05.20.]
- [19] Bátfai Norbert [et. al.]: *A Debreceni Egyetem programozói évkönyve.* [UDPROG]
<http://sourceforge.net/projects/udprog> (.epub, .mobi, .pdf)
- [20] A várakozásokkal ellentétben 2015. jan. 1-jétől az e-könyvek sajnos változatlanul nem kerülnek át a könyvek ÁFA körébe, a célország törvényei lesznek az irányadóak.
<http://www.taxand.com/taxands-take/media/2015-eu-vat-changes-update-ready-install>

Kulturális örökség és virtuális valóság – az ókori Alexandriai Könyvtár

Cultural heritage and virtual reality – the ancient Library of Alexandria

Boda István^a, Tóth Erzsébet^b, Bényei Miklós^c, Csont István^d

^aDebreceni Egyetem, Informatikai Kar

boda.istvan@inf.unideb.hu

^bDebreceni Egyetem, Informatikai Kar

tother12@gmail.com

^cDebreceni Egyetem, Informatikai Kar

benyei@unideb.hu

^dDebreceni Egyetem, Informatikai Kar

ist.csont@gmail.com

Absztrakt: Az előadásunkban egy három dimenziós virtuális könyvtári modellt mutatunk be, amely a VirCA rendszer lehetőségein alapul. Az ókori Alexandriai Könyvtár Kallimakhosz-féle hierarchikus osztályozási rendszerét alapul véve az i.e. 3. század görög költészetének és drámairodalmának néhány jeles és ismert alkotóját, valamint egyes műveiket (ill. ezek meghatározott részleteit) választottuk ki a virtuális könyvtárban hozzáférhetővé tett tartalom számára. Annak érdekében, hogy a megjelenített tartalom a három dimenziós virtuális térben kereshető, ill. megtalálható legyen a felhasználók számára, meghatározott metaadatokat kapcsolunk a kiválasztott ókori görög szövegek angol fordításaihoz. Mindezekkel egy jól kereshető és rendkívül látványos térbeli (spatial) hipertext rendszert alakítottunk ki. Ahogy korábban már említettük, az általunk kifejlesztett három dimenziós virtuális könyvtári modell a VirCA rendszer prezentációs és navigációs lehetőségein alapul. A rendszer alapvető egységének, a virtuális szobának az előterében meghatározott számú fülkét, „kabinetet” alakítottunk ki, és ezekben helyeztük el az általunk kiválasztott és a virtuális környezetben megjelenített könyvtári tartalmakat. Az általunk kiválasztott jeles görög alkotók és műveik Kallimakhosz hierarchikus osztályozási rendszerében a „költők” kategória minden alcsoportját lefedik – ennek megfelelően jelenítettük meg az alkotók műveit (ill. ezek meghatározott részleteit) tartalmazó fülkék felett a megfelelő alcsoportok megnevezését mint alapvető, a könyvtári tartalmak keresését és megtalálását lehetővé tevő metaadatokat. Követve Kallimakhosz katalógusának szerzőközpontú osztályozási sémáját, a három dimenziós virtuális könyvtári környezetben nemcsak költői, ill. drámai műveket tettünk hozzáférhetővé, hanem az egyes szerzők rövid életrajzát is, kiegészítve a szerzők képével, és/vagy mellszobrával. Mindezzel azt a tartalomközpontú elrendezést – és ennek jellegzetes hangulatát – próbáltuk minél jobban visszaadni, amit Kallimakhosz korabeli katalógusa és az ennek megfelelő ókori könyvtári rendszer képviselt. A három dimenziós virtuális könyvtári modell mellett elkészítettük a rendszer ma már „hagyományosnak” tekinthető, hipertext alapú (kétdimenziós) reprezentációját is, amely véleményünk szerint még jobban kiemeli a három dimenziós könyvtári modell által nyújtott új lehetőségeket. A hipertext reprezentáció elkészítésekor megkíséreltük a lehető legjobban megközelíteni azt a környezetet, amit a három dimenziós tér nyújt. Ehhez ugyanazokat a tartalmakat ábrázoltuk, lényegében ugyanabban az elrendezésben, mint a virtuális könyvtári modellben. Amellett, hogy így többé-kevésbé hasonló atmoszférát kapunk, reményeink szerint a hipertext reprezentáció potenciálisan lehetővé teszi azt is, hogy a két megjelenítési formát összehasonlítsuk, és ennek az eredményeit felhasználjuk a rendszer továbbfejlesztése – és a megfelelő elméleti következtetések levonása – során.

Kulcsszavak: virtuális könyvtári modell, térbeli hipertext, VirCA, Alexandriai Könyvtár

Abstract: In our lecture we would like to present a three-dimensional model of a virtual library created by exploiting the capabilities of the Virtual Collaboration Arena (VirCA) system. Based on the classification system elaborated by Callimachus we focus on the field of Greek poetry and drama in the 3rd century BC, and select some illustrious ancient authors and texts for the content of a virtual library. In order that the presented content should be appropriately searchable we attach certain (verbal and/or multimedia) metadata to the English translation of the selected ancient Greek texts and using them to form a well-searchable spatial hypertext system. In the implementation of our three-dimensional virtual library model we have used the 3D presentation and navigation features of the Virtual Collaboration Arena (VirCA) system. In the front part of the virtual room we constructed several cabinets containing the selected and presented library content. We chose prominent Greek ancient authors who would represent each sub-group of Callimachus' hierarchical structure within the „poets” main category and presented above each cabinet the title of the corresponding sub-group as metadata. We followed the same author's approach as Callimachus applied in his catalogue by associating the author's image or his bust with the author's name, and attached a short biography to him. Using this approach we imitated the content layout that the Pinakes (i.e. name of Callimachus' catalogue) had in ancient times. In addition to our three dimensional virtual library model, we have created its hypertext (two dimensional) representation which can further emphasize the additional features that the three dimensional representation provides for us. In the hypertext representation we tried to create an environment which is, in some respect, similar to that of the 3D space — by providing the same content, presenting an equivalent data structure or architecture, and attempting to convey more or less the same impression and atmosphere. According to our hopes, the hypertext representation will potentially also make possible to compare the two representations with each other and whose findings we can use during the further development of the system, in reaching the appropriate conclusions.

Keywords: virtual library model, spatial hypertext, VirCA, Library of Alexandria

1. Miért választottuk az Alexandriai Könyvtárat?

Az antik (görög római világ) legjelentősebb és leghíresebb könyvtára, bár semmi sem maradt belőle, csak a korabeli írók munkáiból ismert. Híre a későbbi korokban is fennmaradt, de mindmáig jelképes erőt képvisel, mintegy a könyvtár-eszmény megtestesítője. Az Alexandriai Könyvtárban gyűjtött görög irodalmi művek örök értékek az emberi kultúra és művelődés számára. Figyelembe vettük, hogy a görög költői művek elemzése nyitott kérdés maradt napjainkban. Ezenkívül kutatásunkban ezeket az ókori műveket új formátumban és környezetben kívántuk bemutatni. Annak érdekében, hogy a megjelenített tartalom a 3D-s virtuális térben kereshető, ill. megtalálható legyen a felhasználók számára meghatározott metaadatokat kapcsoltunk a kiválasztott ókori görög szövegek angol fordításaihoz.

A makedón Nagy Sándor (III. Alexandrosz) kiterjedt birodalma egyik utódállamának, a hellenisztikus Egyiptomnak az első uralkodói, I. Ptolemaiosz Szótér és fia, II. Ptolemaiosz Philadelphosz hozták létre a Kr. e. IV-III. század fordulóján Alexandriában a Muszeion (múzsák szentélye) elnevezésű tudományos intézményt. A királyi kincstárból fenntartott Muszeion egyik legfontosabb része volt a könyvtár. Feladata a görög nyelven írt valamennyi dokumentum összegyűjtése, tehát a görög írásbeliség területén a teljességre törekvés volt. Hamarosan több százezer papirusz tekerceset őrzött.

A könyvtár kiadói programja a görög költők alexandriai kánonjának megteremtését szolgálta. A múzeum és a könyvtár több évszázadig maradt fenn, azonban elpusztult az Aurelianus római császár idején vívott polgárháborúban a Kr. u. 3. század végén. 2002-ben az egyiptomi kormány felavatta az új könyvtár épületét, a Bibliotheca Alexandrinát, amely az ókori tudományos intézmény helyére épült [1], [2].

2. Az osztályozási rendszer felépítése

Állományát a könyvtár harmadik vezetője, Kallimakhosz rendezte Kr. e. 260-240 között, és az ő irányításával készült el a gyűjtemény katalógusa, a Pinakész (Fatáblák). Az elnevezés onnan eredt, hogy a katalógus megfelelő részeit papiruszlapokon fatáblákra ragasztották (esetleg írták), és azokat ráakasztották a szekrényekre (vagy állványokra), hogy megkönnyítsék a keresést és a visszaosztást. A katalógus teljes címe: „*Mindazok táblázatai, akik az irodalom valamely területén jelentőset alkottak.*” (Más fordításban: „*Táblák minden kimagasló alakról az irodalom minden változatában és írásában.*”) A görög eredetiben az „irodalom” szó tágabb értelmű: nevelés vagy műveltség, vagyis Kallimakhosz az uralkodói célhoz igazodva a hellén kultúra teljességét fel akarta ölelni a jegyzékben. A szerzői megközelítést alkalmazva minden névhez rövid életrajzot illesztett, továbbá jellemezte az egyes írások eredetiségét, s mivel a cím (ha létezett egyáltalán) ritkán volt egyértelmű, mindig hozzáfűzte a mű első sorát, és közölte, hány sorból áll a mű.

A katalógus százhusz könyvből (azaz tekercsből) állt, kb. kétszázezer tekercset vett nyilvántartásba. Mivel a Muszeion könyvtára gyakorlatilag teljes volt a görög irodalom műveiben, a katalógus egyúttal a görög irodalom bibliográfiája is. Életrajzi jegyzetei révén őse az irodalom- és tudománytörténetnek is [3]. Egy ilyen terjedelmű katalógus megalkotásához előbb rendezni kellett az egész állományt, vagy a gyűjteményt eleve szisztematikus felállításban raktározták.

A katalógus szerkezete a korabeli tudósok és irodalmárok gondolarendszerét követve két szintre tagolódott:

- az egyik (a felső) az írók csoportosítása – ez adta a felosztás vázát,
- a másik az egyes fejezeteken belüli szerzői betűrend.

Kallimakhosz a könyvtár anyagát két fő részre bontotta: költőkre és prózairókra. Mindkettőben hat-hat csoportot alakított ki:

- költők: epikusok, elégikusok, szatirikusok, lírikusok, tragédiairók, komédiairók
- prózairók: történetírók, szónokok, filozófusok, orvosok, természettudósok, vegyes (itt voltak a szakácskönyvek is „vacsorázó írók” alcímmel) [4].

3. Tartalmak a könyvtári modellben

Három jeles ókori görög szerzőt választottunk, akik minden egyes csoportot külön-külön képviselnek a „költők” fő kategórián belül. Ugyanazt a szerzői megközelítést követtük, mint amit Kallimakhosz használt katalógusában összekapcsolva a szerző képét vagy hermáját annak nevével, valamint egy rövid életrajzot kapcsolva hozzá. Ezzel a megközelítéssel ugyanazt a tartalmi elrendezést követtük, mint amivel a Pinakész (azaz a Kallimakhosz-féle katalógusnak a megnevezése) rendelkezett az ókori időszakban.

A tartalmak könyvtári modellbe történő integrálásának lépései a következők voltak:

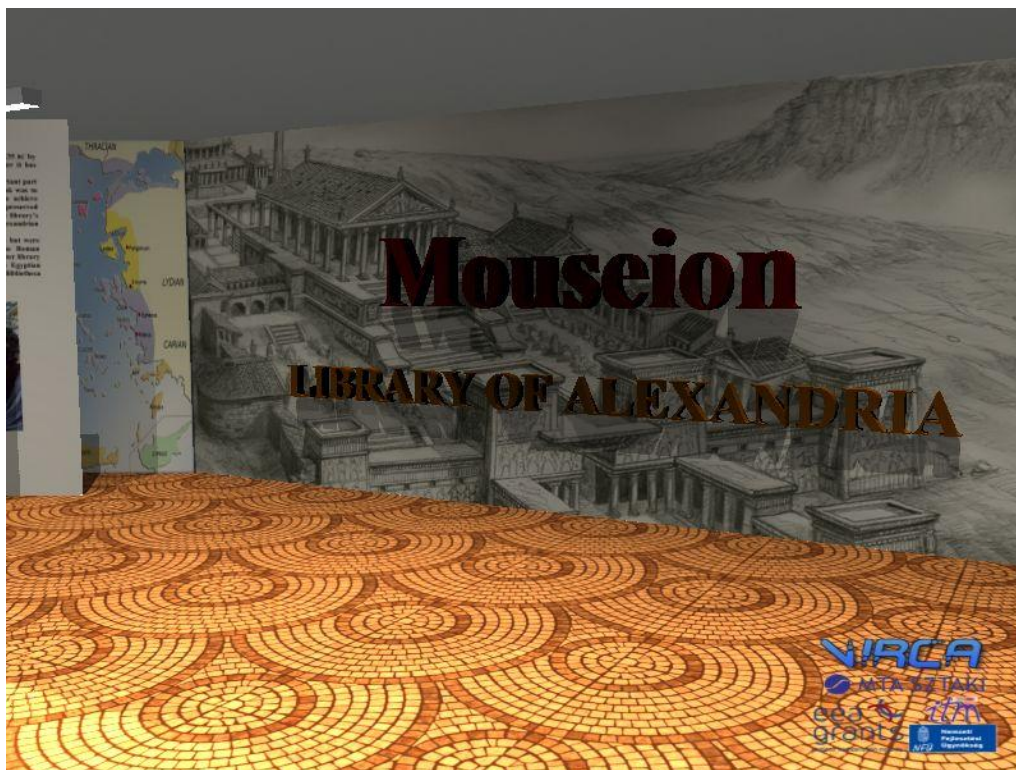
1. A szerzők életével és költői műveikkel kapcsolatos képekre kerestünk.
2. A kiválasztott szerzők rövid életrajzát összegyűjtöttük és megszerkesztettük.
3. Figyelembe véve a szerzői joggal kapcsolatos kívánalmakat, a görög költői művek megfelelő angol nyelvű fordításaira kerestünk.
4. Az összegyűjtött tartalmakat betöltöttük a VirCA rendszerbe a szoftver sajátosságainak megfelelő szigorú logikai sorrendben és eszközökkel.

4. A három dimenziós környezet bemutatása

A három dimenziós környezet előnyeit a következőképpen foglalhatjuk össze:

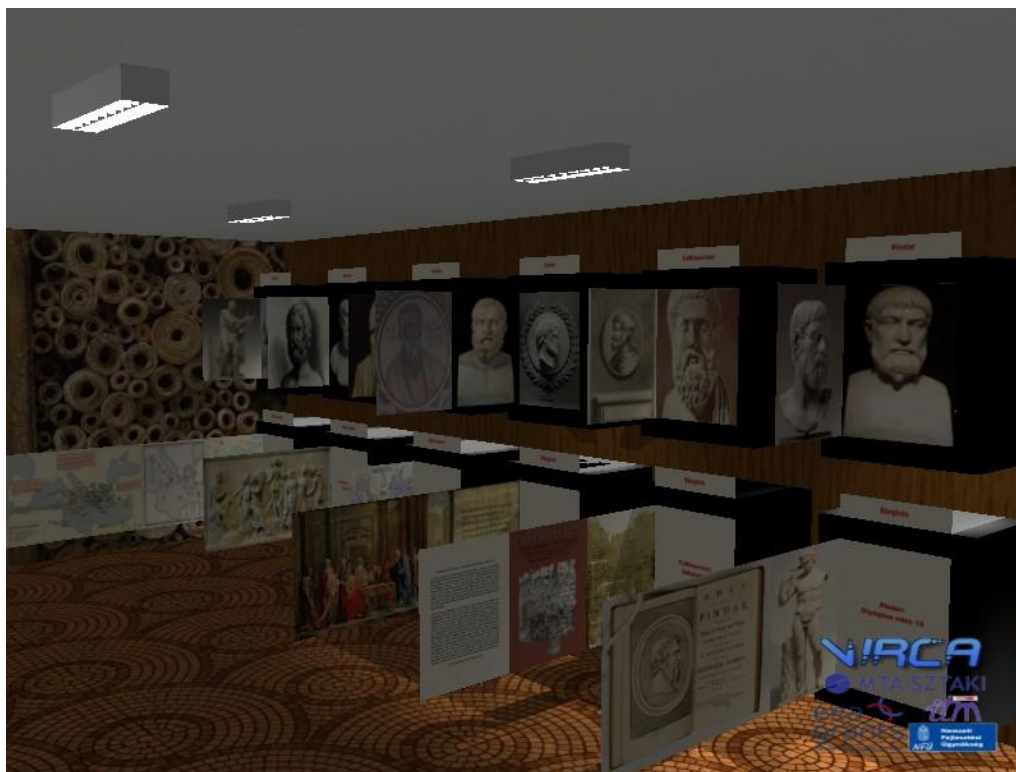
- speciális *megjelenítési és navigációs* lehetőségeket biztosít: bármilyen irányban mozoghatunk, körbejárhatjuk és megvizsgálhatjuk az objektumokat stb.
- lehetővé teszi a 3D környezetben reprezentált objektumok közötti kapcsolatok egészben történő, „*holisztikus*” áttekintését,
- a virtuális valóság erős érzelmi hatást vagy „*qualia*”-t eredményez.
- a három dimenziós környezet egyes tulajdonságai *akadálymentességi* szempontból kifejezetten előnyösek, ugyanis a felhasználó választhatja meg a reprezentált tartalom olvasásához szükséges optimális távolságot, és ehhez elegendő pusztán a kurzorbillentyűvel történő navigálás.

A VirCA (*VIR*tual *CO*llaboration *ARE*na) egy 3D környezetet biztosító informatikai keretrendszer, amelynek fejlesztője az MTA SZTAKI Kognitív Informatikai Kutatócsoportja [5], [6]. Kutatásunk szempontjából a VirCA 3D megjelenítési képességei alapvetőek. A VirCA-ban a megjelenítés alapegysége a *virtuális szoba*, amelyet berendezhetünk és dekorálhatunk lényegében bármilyen 3D objektummal (pl. kabinetekkel, képfolyosókkal, navigátorokkal, panelekkel stb.). Kifejlesztett könyvtári alkalmazásunk esetében a virtuális szoba (vagy könyvtár) hátsó falán jelenítjük meg a könyvtárban reprezentált tartalmakra utaló *Mouseion* feliratot („A Múzsák Szentélye”) és az egykori Alexandriai Könyvtár rekonstruált képét (lásd 1. ábra).



1. ábra. A virtuális könyvtár felirata és háttérképe

A felirattól balra és jobbra tájékoztató posztereket helyeztünk el, amelyek bemutatják az ókori Alexandriai Könyvtár rövid történetét, egy térképet a könyvtár környezetéről, valamint a könyvtári rendszert, amelyet a híres tudós és könyvtáros, Kallimakhosz dolgozott ki. A korhű hangulat megteremtése érdekében az oldalsó falakat papiruszok ismétlődő képei dekorálják, a padlót pedig ókori mintázatok díszítik. A felirattal szemközti falon max. 18 *kabinet* helyezkedik el két párhuzamos sorban (lásd 2. ábra). A kabinetek koncepciója alapvető a virtuális könyvtári modell szempontjából, mivel a kiválasztott könyvtári tartalmakat (a modell jelenlegi verziójában az adatbázisban tárolt műveket vagy a művek szerzőinek rövid életrajzát) ezekben jelenítjük meg.



2. ábra. A kiválasztott könyvtári tartalmakat tartalmazó kabinetek

A kabinetek tetején egy felirat („címke”) mutatja – a kabinetben megjelenített tartalomtól (egy kiválasztott mű szövege vagy ennek egy részlete, ill. egy szerző életrajza) függően – a szöveghez tartozó kategóriát vagy a szerző nevét. A kabinetek elejét („ajtaját” vagy „bejáratát”) egy függöny vagy „fátyol” díszíti, amely a szöveg címét vagy pedig a szerző képét mutatja. A kabinetek „mennyezetén” egy lámpát helyeztünk el, hogy megvilágítsuk a benne megjelenített tartalmat. A kabinetekhez kétoldalt egy max. 4 képből álló *képfolyosót* kapcsoltunk, amely segíti a kabinetben megjelenített tartalom megtalálását a virtuális környezetben, és multimédiás (képi) információk hozzáadásával bővíti a kabinetekben reprezentált verbális tartalmakat.

5. A modell implementációja

A virtuális szobát és tartalmát (esetünkben pl. a kabineteket, a képeket, a szöveges tartalmakat stb.) a VirCA rendszerben egy jól dokumentált XML fájl írja le. A virtuális

szobában a fentiekben leírt módon először létrehoztuk a kabinetek egy előre definiált elrendezését, majd egy Java nyelvű programmal konvertáltuk a kiválasztott könyvtári tartalmak egyszerű szöveges leírását VirCA alapú XML reprezentációra. Ezután a hozzáférhető könyvtári tartalmakat adatbázisba szerveztük, és egy interaktív (webes) kereső felületet készítettünk a releváns információ visszakeresésére. A felületen jelenleg kategóriák, szerzők és címek szerint kereshetünk, a keresési eredményeket pedig egy PHP program szolgáltatja. Jelenleg ezek a következő formátumokban állnak rendelkezésre:

- a visszakeresett könyvtári tartalmak egyszerű szöveges leírása;
- a VirCA rendszerben történő megjelenítéshez szükséges, a virtuális szobát leíró XML fájl;
- az XML fájl közvetlen webes megjelenítéséhez készített XSLT stíluslap;
- a virtuális szoba elrendezését követő weblap, amely a 3D virtuális könyvtári modell (weben is hozzáférhető) hipertext reprezentációját valósítja meg.

6. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Csont István (Ph.D hallgató Debreceni Egyetem Informatikai Kar) közreműködésével lett fejlesztve a jelenlegi HTML/JavaScript alapú felület. Kutatása a TÁMOP 4.2.4.A/2-11-1-2012-0001 Nemzeti Kiválóság Program című kiemelt projekt keretében zajlott. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Grüll Tibor: *Könyvtárak és könyvkiadás az ókorban*. In: Bevezetés az ókor tudományba 1., Debrecen: Kossuth Egyetemi K., 1996, 223-231.
- [2] Guglielmo Novelli: *Alexandriai Könyvtár. Egyiptom, Alexandria*. In: Az építészet csodái: Az ókortól napjainkig / [összeáll., szerk. Alessandra Capodiferro]; [ford. Getto Katalin] Pécs : Alexandra, 2010, 164-167.
- [3] R. MacLeod: *The Library of Alexandria: Center of Learning in the Ancient World*, New York: I. B. Tauris & Co Ltd., 2005.
- [4] M. El-abbadi: *The Life and Fate of the Ancient Library of Alexandria*, 2nd illustrated ed., Unesco/UNDP, 1990.
- [5] VirCA: 3D Virtual Collaboration Arena. <http://virca.hu/> (2014-07-04)
- [6] P. Galambos and P. Baranyi: VirCA as Virtual Intelligent Space for RT-Middleware. In: *AIM 2011. Proceedings of the 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. Budapest, Hungary, July 3-7 2011. 140–145.

A plakátok leíró sémája

Posters and their description schema

Némethi-Takács Margit

Debreceni Egyetem Informatikai Kar Könyvtárinformatika Tanszék

`takacs.margit@inf.unideb.hu`

Absztrakt: Bár a könyvtárak meghatározó kiadványtípusai évszázadok óta a könyvek és a folyóiratok, a könyvtárak gyűjtőköre más dokumentumtípusra is kiterjedhet. Az egyes dokumentumtípusokból a dokumentumok számától függően különgyűjtemények szerveződhetnek, ahol helyet kaphatnak az aprónyomtatványok is. Ennek a kiadványtípusnak az egyik legtöbbet kutatott és leggyakrabban kölcsönzött része a plakátanyag. A plakátokból kialakított gyűjteményekben történő kutatások során azonban nehézségekbe ütközhetünk, hiszen nem biztos, hogy elegendő a mű címének és szerzőjének ismerete, sok esetben szükség lehet még a plakáton látható kép megtekintésére is. Ezért előfordulhat, hogy egy releváns mű megtalálásához több plakátot is meg kell nézni, ami a plakátok tárolási és típusbeli sajátosságait tekintve problémás lehet. Erre megoldást jelenthet a plakátok digitalizált változatának elkészítése és elektronikus gyűjteménybe szervezése. Digitalizált plakátgyűjteményekre többszörösen igaz az, hogy működ(tet)ésükhöz nélkülözhetetlen a metaadatok használata, hiszen ezek segítségével valósulhat meg az elektronikus dokumentumok keresése. A problémát az jelenti, hogy a formai feltárást magára vállaló könyvtárak eltérő metaadat elemkészlettel dolgoznak. Természetesen a legfontosabb adatok, mint cím, szerzőség, megjelenés, terjedelem felvételre kerülnek, de a plakátokra jellemző adatok megjelenítését csak kevés könyvtár tudja felvállalni. A probléma hátterében az is áll, hogy nem létezik olyan szabvány, amely a plakátok bibliográfiai leírását szabályozná. Épp ezért fontosnak tartok egy olyan elemkészlet kidolgozását, amely a plakátok formai sajátosságait tükrözi, ugyanakkor magában hordozza az ISBD alapú leírások jellemzőit, továbbá a plakátok digitalizált változatára vonatkozó adatokat is tartalmazza. Ebben a tanulmányban bemutatásra kerülő, saját fejlesztésű PoMeDS (Poster Metadata Description Schema) névre keresztelt, a plakátok metaadatait tároló XML alapú séma ilyen elemkészlettel dolgozik.

Kulcsszavak: plakátok, metaadat, XML séma

Abstract: Though books and journals are determinant document types in libraries since centuries, the collection interest of the libraries can include other document types too. Special collections can be organized from certain document types depending on the number of documents where small prints can be placed as well. Posters represent the most investigated and often borrowed part of this publication type. During conducting research in poster collections we, however, can face difficulties, since it is not sure that it is enough for us to know the title and author of the work. In several cases it might be even necessary to have a look at the image itself on the poster. In order that a relevant work should be found we must look at more posters, which can be rather problematic in regard to storage and type of the posters. Preparing the digitized version of the posters and organizing them into an electronic collection can be a solution to this task. It is definitely true for the digitized poster collections that the use of the metadata is essential for their operation as with the help of the metadata the electronic documents can be retrievable. The problem arises from the fact that the libraries taking on cataloguing apply a different set of metadata elements. Of course the most significant bibliographic data, such as title, author, publication, number of pages are recorded, but the presentation of the data which are typical of the posters is undertaken only by a few libraries. In addition, there is no standard which regulates the bibliographic description of the posters. For this reason we consider it important to elaborate such a set of metadata elements, which reflects the formal characteristics of the posters, at the same time bears the characteristics of the ISBD descriptions and also contains the data characterized by the digitized version of the posters. In this study the Poster Metadata Description Scheme (PoMeDS) that we have developed will be presented. It is an XML-based scheme storing the metadata of the posters and working with such a set of metadata elements.

Keywords: posters, metadata, XML schema

1. Bevezetés

A könyvtáraknak a múlt század végétől újabb és újabb kihívásokkal kell szembenézniük. Először a hagyományos cédulakatalógusokat kezdték el felváltani számítógépes katalógusokkal, majd pedig a számítógépes hálózatok kialakulása gyakorolt nagy hatást a könyvtárak életére.

A számítástechnika valamint a telekommunikáció fejlődésének és találkozásának köszönhetően az egyes számítógépek elszigeteltsége megszűnt. Ezáltal lehetőség adódott a számítógépek közötti kapcsolatok kiépítésére, és így akár a világ másik feléről is le lehetett kérdezni egy könyvtár állományát.

A következő nagy lépést az jelentette, amikor megjelentek az úgynevezett közös könyvtári katalógusok. Ezek a közös könyvtári katalógusok a rendszerhez csatlakozó könyvtárak állományának akár egészéről is kínálhatnak információt.

A technika és a technológia fejlődésének köszönhetően a könyvtárak számára lehetőség nyílt arra, hogy az állományukban lévő dokumentumokat digitalizálják, és az így létrejött digitalizált dokumentumokból elektronikus gyűjteményt hozzanak létre.

Az elektronikus könyvtárak számának növekedésével párhuzamosan a digitális gyűjteményekben megjelentek a nem hagyományos könyvtári dokumentumok is. Manapság pedig egyre több olyan elektronikus könyvtárat hoznak létre és működtetnek, amelyeknek gyűjteménye döntően speciális dokumentumtípusokból áll. Ilyen speciális dokumentumtípus a plakát is, amely a vizuális alkotások közé sorolható.

A különböző plakátgyűjteményekben történő keresés nem biztos, hogy könnyedén megvalósul. Egyrészt a nyomtatott plakátok általában nagyméretűek, ezért megtekintésük elég helyigényes. Másrészt tárolásuk jobb esetben, fiókos szekrényekben történik, sokszor azonban tékába kötve vagy dobozban tárolják ezeket a dokumentumokat. Utóbbi esetben nem csak a plakátok megtekintése problémás, hanem a tékák és dobozok közötti eligazodás is.

Mivel a plakátgyűjtemények használata elég nehézkes, ezért fontos, hogy minél több plakátnak készüljön el a digitális változata. A digitalizált plakátokból kialakított gyűjtemények használatához nélkülözhetetlen a metaadatok, mert csak ezek segítségével valósulhat meg a digitális dokumentumok keresése. A plakátok bibliográfiai leírását meghatározó szabvány hiányában szükséges egy olyan elemkészlet kidolgozása, amely a plakátok formai sajátosságait tükrözi, ugyanakkor magában hordozza az ISBD (International Standard Bibliographic Description) alapú leírások jellemzőit. Ebben a tanulmányban bemutatásra kerülő, saját fejlesztésű PoMeDS (Poster Metadata Description Schema) névre keresztelt, a plakátok metaadatait tároló XML (Extensible Markup Language) alapú séma ilyen elemkészlettel dolgozik és ezeken túlmenően a digitalizált plakátok leíró adatait is tartalmazza.

2. Hagományos könyvtárártól a digitális könyvtárárkig

A könyvtáraknak a múlt század végétől újabb és újabb kihívásokkal kell szembenézniük. Először a hagyományos cédulakatalógusokat kezdték el felváltani számítógépes katalógusokkal, majd pedig a számítógépes hálózatok kialakulása gyakorolt nagy hatást a könyvtárak életére. Mára a csak helyben használható papíralapú (nyomtatott) dokumentumok szolgáltatásától eljutottunk a digitális könyvtárárkig.

A múlt század végéig a könyvtári szolgáltatások papírtechnológiára épültek. Az informatika fejlődésének köszönhetően azonban kialakultak az automatizált könyvtárak. Az automatizált könyvtár fogalma alatt olyan könyvtárat értünk, amelyben a könyvtári dokumentumok elsősorban papíralapúak, de a könyvtári munkafolyamatok számítógépesítve

vannak.[6] Ez azt is jelenti, hogy a hagyományos cédulakatalógusokat folyamatosan váltották fel a számítógépes katalógusok.

Később a számítástechnika valamint a telekommunikáció fejlődésének és találkozásának köszönhetően megjelent az internet, ami nagy változást jelentett a kommunikáció és az információ szerzés világában. Könyvtári szempontból ez azt jelentette, hogy az alapvető szolgáltatásokat szép lassan kivitték az épületen kívülre.

Az internet megjelenésének másik nagy jelentősége, hogy az egyes számítógépek elszigeteltsége megszűnt. Ezáltal lehetőség adódott a számítógépek közötti kapcsolatok kiépítésére, és így akár a világ másik feléről is le lehetett kérdezni egy könyvtár állományát, azaz a felhasználó nincs elválasztva a katalógustól. [6]

A következő nagy lépést az jelentette, amikor megjelentek az úgynevezett közös könyvtári katalógusok. A felhasználó már nem csak az adott könyvtár dokumentumait tudták elérni, hanem más könyvtárak dokumentumaihoz is hozzá tudtak férni.

A dokumentumhasználatban a jelentős változást az elektronikus dokumentumok megjelenése hozta. A technika és a technológia fejlődésének köszönhetően ugyanis a könyvtárak számára lehetőség nyílt arra, hogy az állományukban lévő dokumentumokat digitalizálják, és az így létrejött digitalizált dokumentumokból elektronikus gyűjteményt hozzanak létre. Azaz létrejöttek a digitális könyvtárak.

Kezdetben a digitális könyvtárak hagyományos könyvtári dokumentumok elektronikus változatához nyújtottak elérést. A digitális könyvtárak számának növekedésével párhuzamosan azonban a digitális gyűjteményekben is megjelentek a nem hagyományos könyvtári dokumentumok. Manapság pedig egyre több olyan digitális gyűjteményt hoznak létre, amely döntően speciális dokumentumtípusokból áll. Ilyen speciális dokumentumtípus a vizuális alkotások közzé sorolható plakát is.

3. A plakát

A plakát – falragasz, hirdetmény – a kis- és aprónyomtatványok körébe tartozó kiadvány. A kisnyomtatvány meghatározás általában 4 ívet meg nem haladó terjedelmű nyomtatványokat jelöl, míg az aprónyomtatvány fogalma alatt az egyleveles nyomtatványokat értjük. Mivel a plakátok egyleveles nyomtatványok, így ezeket az aprónyomtatványok közzé soroljuk.

A mai értelemben vett plakáttal, falragasszal csak a betűnyomtatás megjelenése utáni időszakban találkozhatunk. Eleinte jórészt szöveges plakátok jelentek meg az utcákon, a későbbiekben ezek a szövegek képekkel egészültek ki. A grafikai plakátoknál az áttörést az jelentette, amikor a műfaj találkozott *Senefelder* találmányával, a litográfiával. A plakátműfaj meghonosodását, elterjedését tulajdonképpen a mindennapi élet szükségletei eredményezték.

A betűnyomtatás felfedezése utáni, tipográfiával készült korai szöveges hirdetmények elsősorban közérdekű tudnivalókról tájékoztatták a lakosságot. Így váltak ismertté a törvények, a közigazgatási intézkedések és a hadihirdetmények, de készültek árverési plakátok és toborzó hirdetések is. Majd szép lassan megjelentek az első kereskedelmi plakátok, először kikalakú röplapok formájában.[3]

A gyors és olcsóbb litográfiai eljárásnak köszönhetően napvilágot láttak a nagyvárosi utcákon a művészi igényességgel festett litografált falragaszok. Az új típusú képes műfaj először a kereskedelem, az ipar és a kultúra, majd később a politikai meggyőzés hatékony vizuális eszközévé vált. A múlt század végére a kereskedelmi plakátok kerültek túlsúlyba, és megjelentek az óriási multiplakátok.

Formai sajátosság szempontjából a plakátoknak három fajtáját különböztetjük meg:

- *Szöveges plakát:* Nevéből adódóan csak szöveget tartalmaz. A szöveges plakát legtöbbször pontos, precíz, részletekbe menő, sok információt tartalmazó, adatokban gazdag dokumentumtípus. Főleg egy hivatalos szerv, vagy intézmény egyik legfontosabb tájékoztatási eszköze volt a város lakossága felé. Korabeli szerepe a szűkebb, vagy tágabb közösséggel való tényközlés volt. E szerepet azonban ma már leginkább más kommunikációs csatornák látják el, ezért is csökkent jelentősen a szöveges plakátok száma.
- *Figurális plakát:* csak és kizárólag grafikai elemeket tartalmaz, vagyis csak rajzok és képek szerepelnek a plakáton.
- *Szöveggel ellátott figurális plakát:* előző kettő kombinációja, és mindenesetben a leghatásosabb eszköz. Az így készült falragaszra a kép és a szöveg összehatása jellemző. A grafikai plakátok főleg kereskedelmi és/vagy kulturális céllal jöttek létre, de céltól függetlenül minden grafikai plakátra a figyelemfelkeltés és a tömör közlés módja jellemző.

Akár szöveges, akár grafikai plakátról legyen szó, mindig célja van: az érdeklődés felkeltése, a tájékoztatás vagy a meggyőzés kép és/vagy szöveg segítségével. A plakátok képeiből és szövegéből a múlt eseményei mozaikszerűen rajzolódnak ki, ezért rajtuk keresztül megismerhetjük a társadalmi és politikai eseményeket, az ipar, a kereskedelem és a kulturális élet változásait. A grafikai plakátok a művészeti élet értékes dokumentumai, a vizuális kultúra részeként kultúrtörténeti, művészettörténeti, tartalmuknál fogva fontos kortörténeti dokumentumok is.

Egy-egy komolyabb plakátgyűjtemény nélkülözhetetlen forrása lehet a történeti, ipartörténeti, biográfiai, művészettörténeti, de más irányú kutatásoknak is, hiszen ezek a nyomtatványok politikai harcok, forradalmak és ellenforradalmak híreit, eseményeit, rendeleteit közzé tették a tömeggel. Ezeken túlmenően a plakátokon keresztül megismerhetjük a társadalmi és politikai eseményeket, az ipar, a kereskedelem és a kulturális élet változásait is. [7] Ez mind alátámasztja azt, hogy a plakátoknak van és kell, hogy legyen helye a közgyűjteményekben.

A különböző plakátgyűjteményekben történő kutatások során azonban nehézségekbe ütközhetünk, mert nem biztos, hogy elegendő a mű címének és szerzőjének (ha tudunk róluk) ismerete, sok esetben szükség lehet még a plakáton látható kép megtekintésére is. Ezért előfordulhat, hogy egy releváns mű megtalálásához több plakátot is meg kell nézni, ami a plakátok tárolási és típusbeli sajátosságait tekintve gondot okozhat. Ugyanis a nyomtatott plakátok általában nagyméretűek, ezért megtekintésük elég helyigényes. Tárolásuk jobb esetben, lapos fiókos térkép/rajztároló szekrényekben vagy polcos rekeszekben történik. A szekrényekben tíz-húsz darabos kötegekben kerülhetnek elhelyezésre a plakátok. Azonban a könyvtárak többségében tékába kötve vagy dobozban tárolják ezeket a dokumentumokat. Utóbbi esetben nem csak a plakátok megtekintése lehet problémás, hanem a tékák és dobozok közötti eligazodás is.

A technika fejlődésének köszönhetően megoldódni látszik ez a probléma, ugyanis lehetővé vált a dokumentumok digitalizálása és elektronikus tárolása, azaz elektronikus archívumok létrehozása. Az elektronikus archívum működ(tet)éséhez nélkülözhetetlen a metaadatok használata, hiszen ezek segítségével valósulhat meg a digitális dokumentumok tárolása és keresése.

4. Metaadatok világa

A metaadatok „szűkebb értelemben az internetforrások intellektuálisan vagy automatikusan előállított másodlagos adata”. [14] A használhatóság érdekében a metaadatoknak követni kell a szabványokat és a szabályokat, hogy az információforrásokban keresők ugyanazokat a metaadat jelöléseket és jellemzőket használhassák, mint azok, akik létrehozták és fenntartják az információforrást.

Ahhoz, hogy a metaadatok feladatukat el tudják látni, ahhoz szükség van arra, hogy ezeket valamilyen számítógépes környezetben tárolják. A hagyományos bibliográfiai adatok számítógépes formában történő kezelése a különböző MARC (Machine-Readable Cataloging) formátumokban történik.

Az elektronikus dokumentumok feldolgozása a formai feltárás hagyományos eszközeivel viszont nem valósulhat meg, ugyanis ezek a dokumentumok megjelenésükben, formájukban jelentősen eltérnek a megszokott papíralapú dokumentumoktól. Ennél fogva az elektronikus dokumentumok leíró adatainak tárolására a MARC formátum nem használható megfelelően. Az elektronikus dokumentumok bibliográfiai adatainak tárolására XML nyelvet kezdték el alkalmazni. Mára a metaadatok tárolásának legelterjedtebb formája az XML séma lett.

Az XML előnye, hogy nyílt rendszer, azaz bárki tudja alkalmazni. Egy olyan jelölő nyelv, amely az SGML (Standard Generalized Markup Language) továbbfejlesztésével jött létre az 1990-es években, és aminek a segítségével újabb jelölőnyelvek készíthetők. Többek között olyan területen alkalmazzák ezt a metanyelvet, mint matematika, nyomdaipar, közgyűjtemények stb.

Az XML-lel általános célú elemkészletet és dokumentumszerkezetet adhatunk meg, amelyet együttesen XML alkalmazásnak nevezünk. XML alkalmazást definiálhatunk DTD (Documentum Type Definition) létrehozásával vagy XML sémával. Ma már az XML sémák alkalmazása jobban elterjedt, ugyanis a DTD-nek vannak bizonyos korlátai. Ilyen korlát például a névtér-érzékenység, ami azt jelenti, hogy a DTD-nek az összes névtér valamennyi elemének és attribútumának összes lehetséges kombinációját ismernie kell, és az alkalmazott előtagokat előre kell definiálni. A DTD-k másik gyengesége, hogy saját szintaxisuk van, ami szükségtelen lenne, hiszen az általuk létrehozott modelleket normális XML-dokumentum jelöléssel is le lehetne írni.

A hátrányok kiküszöbölésére 2001-ben a W3C konzorcium kiadta az XML sémaszabványt. Egy XML séma meghatározza azokat az elemeket és jellemzőket valamint ezek hierarchiáját, amelyeket egy megfelelő dokumentum tartalmazhat. A séma tulajdonképpen egy speciális típusú XML dokumentum, amelyet külön tárolunk az adott sémával leírt XML dokumentumtól, és ami meghatározza ezeknek a dokumentumoknak a szerkezetét és érvényességét.

A sémákban úgy biztosítjuk a sémaelemek jelölését, hogy azokat a gyökérelemben deklarált névtérhez rendelve, minősített névvel szerepeltetjük. A séma állomány az xsd:schema elembe ágyazva elemek, attribútumok, és értékek definícióit tartalmazza. [2]

Az XML sémaszabvány, akárcsak a legtöbb katalogizálási szabvány szintaktikai és szűkebb értelemben szemantikai szabályokat ír elő, ezért az XML séma alkalmas a különböző bibliográfiai szabályok gépre történő átültetésére. Lehetőség van a sémákban az egyes adatelemek sorrendjének a meghatározására, ami a bibliográfiai leírás egyik meghatározója, továbbá az adatelemek kötelezőségéről is tudunk rendelkezni. Mivel az elemek egymásba ágyazhatóak, ezért lehetőség van az összetartozó adatelem ismétlésére. Nagy előnye még, hogy nem előre definiált elemkészlettel dolgozik, így az egyes metaadat-rendszerek saját

magukra szabva formálhatják. Talán ez az oka, hogy a metaadatok tárolására az egyik leggyakrabban alkalmazott szabvány az XML sémaszabvány. [12]

Az előbb említett jellemzők miatt az XML nyelv könyvtári alkalmazása egyre inkább elterjed. Ennek a jelölő nyelvnek az alkalmazására jó példa a MODS (Metadata Object Description Schema).

A MODS bibliográfiai adatelemkészletet a Kongresszusi Könyvtár Hálózatfejlesztő és MARC21 Szabványhivatala (Library of Congress Network Development and MARC21 Standards Office) fejlesztette ki azzal a szándékkal, hogy „alkalmas legyen már létező MARC21 rekordok válogatott adatainak továbbítására csakúgy, mint eredeti információforrásokról új bibliográfiai rekordok készítésére. Tartalmazza a MARC21 mezők egy részét, és inkább nyelvre alapozott mezőket használ, mint számmal ellátottakat, bizonyos esetekben újracsoportosítva a MARC21 bibliográfiai formátum elemeit”. [5]

A MODS 20 felső szintű elemmel dolgozik, amelyeknek egy része tényleges értékeket tartalmaz, és vannak olyanok, melyek alelemekkel rendelkeznek. A MODS hierarchikus rendszer, amelyben a szülő-gyermek kapcsolat érvényesül. A séma gyakran használ konténer elemeket, amiknek a feladata az alelemeket tartalmazása. Ezek a konténer elemek nem tartalmazzak metaadat értékeket. A MODS felső szintű elemeinek nagyjából fele ilyen konténer elem, míg a többi felső szintű elem nem konténer elem, azaz közvetlenül tartalmaz metaadat értékeket alelem nélkül. A következő példa két felső szintű elemet mutat be, amelyből az egyik konténer elem beágyazott alelemmel, és a másik egy felső szintű nem konténer elem.

```
<originInfo>
  <place>
    <placeTerm>Washington, D.C.</placeTerm>
  </place>
  <publisher>Library of Congress</publisher>
  <dateIssued>1998</dateIssued>
</originInfo>

<typeOfResource>still image</typeOfResource>
```

A MODS XML attribútumokat is használ az elemek pontosítására, az elemek értékének kódolására és egyéb hasonló célokra. Ezek az attribútumok a kezdő tagban szerepelnek. Az egyszerű elemek egynél több attribútumot is tartalmazhatnak. A MODS-ban a kontrollált szótárak az authority attribútummal kerülnek jelölésre, mint például a subject elemnél: <subject authority="lcs" >. [9]

A MODS-on kívül más fontos metaadat-rendszerek is az XML sémát használják metaadatainak tárolására, ilyen többek között a Dublin Core vagy a VRA Core (Visual Resources Association) is.

5. Plakátokat leíró adatok

Digitalizált plakátgyűjteményekre többszörösen igaz az, hogy működ(tet)ésükhöz nélkülözhetetlen a metaadatok használata, hiszen ezek segítségével valósulhat meg az elektronikus dokumentumok keresése. Mivel nem létezik olyan szabvány, amely a plakátok bibliográfiai leírását szabályozná, ezért a cél az, egy olyan elemkészlet kialakítása, ami ISBD alapú, a plakátok formai sajátosságait tükrözi és a plakátok digitalizált változatára vonatkozó információkat is megjeleníti.

Ahhoz, hogy ezt a megfelelő metaadat-rendszert megtaláljuk, először meg kell határozni azoknak az adatoknak a körét, amelyek a nyomtatott plakátok bibliográfiai leírására szolgálnak. A plakátok szabványos leírásának az ISBD/NBM lehet az egyik alapja, így viszont a formai feltáráshoz használt metaadatok köre nem lehet teljes, mivel az ISBD/NBM-

ben nem található meg a plakátokat leíró speciális metaadatok. Bár a hagyományosan megjelent plakátok formai feltárását és katalogizálását meghatározó szabvány nem áll rendelkezésre, viszont léteznek olyan útmutatók, szabályzatok, amelyek a képek és a grafikus művek feldolgozására jöttek létre. A plakátok bibliográfiai leírására használható szabályzatok, útmutatók tehát a következők:

- Állóképek bibliográfiai leírása : útmutató / [összeáll. Varga Ildikó]. - Bp. : OSZK-KMK, 1981. - 33 p.
- Graphic materials : rules for describing : original items and historical collections / Elisabeth W. Betz. - Washington D.C. : LC, 1982. <http://www.loc.gov/rr/print/gm/GraMatWP8.pdf>
- ISBD/G: General International Standard Bibliographic Description
- ISBD/NBM: International Standard Bibliographic Description for Non-Book Materials

A különféle útmutatók és szabályzatok használatának is köszönhető, hogy a könyvtárak nem mutatnak egységes képet a plakátok feldolgozása terén. Épp ezért a plakátok sajátosságait figyelembe véve ki kell alakítani egy bibliográfiai leírásra alkalmas metaadat elemkészletet. Az ismertetésre kerülő PoMeDS (Poster Metadata Description Schema) névre keresztelt, a plakátok metaadatait tároló XML alapú séma ilyen elemkészlettel dolgozik.

6. Poster Metadata Description Schema

A PoMeDS kialakításánál a fő szempont volt a szabványos bibliográfiai leírás követése. Ennél fogva nagy jelentősége volt az ISBD szabványoknak, továbbá az egyes metaadatrendszerekkel és azok sémáival kapcsolatban megjelent tanulmányoknak. Tervezés során fontos szempont volt, hogy ne csak egy leírás legyen elkészíthető, hanem a leírást ki lehessen egészíteni egységesített besorolási adatokkal, melyeknél az IFLA (International Federation of Library Associations) útmutatásai követendők.

A sémában használt elemek a következő csoportba sorolhatók:

- bibliográfiai leírás adatelemeinek megfeleltethető elemek,
- bibliográfiai leírás egyezményes jeleinek megfeleltethető elemek,
- bibliográfiai leírás szerkezetét meghatározó elem,
- besorolási adatoknak megfeleltethető elemek.

A besorolási adat elemei három csoportra bonthatók:

- egységesített név
- kiegészítő adatok
- kiegészítő adatok egyezményes jelei.

Besorolási adatok esetén a különböző típusú nevek egységesített alakjának meghatározását a PoMeDS-hez illeszkedő bibliográfiai rekordokat készítő feldolgozó könyvtáros feladata.

PoMeDS rendelkezik az elemek sorrendjéről, mivel a bibliográfiai leírás adatelemeinek kötött a sorrendje. A séma rendelkezik továbbá az elemeket megelőző egyezményes jelekről is, ugyanis ezek az egyezményes jelek nemzetközi szinten egységesek és elfogadottak, és ezek a jelek biztosítják a bibliográfiai leírás nemzetközi érthetőségét és értelmezhetőségét. Az

egyezményes jelek önálló, üres XML elemként lettek definiálva. Egyezményes jelek esetén szintén a feldolgozó könyvtáros feladata azoknak a jeleknek a megadása, amelyek a leírásban megjelenő konkrét bibliográfiai adathoz kapcsolódnak (ilyen például a három pont, ami az egyes adatsorozatban a szövegelhagyást jelöli).

```
<xs:element name="colon">
  <xs:complexType/>
</xs:element>
```

A nyomtatott plakátokhoz kapcsolódó bibliográfiai leírások adatelemeinek megfeleltetett PoMeDS elemek elnevezése az ISBD/NBM terminológiája alapján készült, szem előtt tartva a nemzetközi felhasználhatóságot. Párhuzamos bibliográfiai adatelemeknek megfeleltetett PoMeDS elemek nevei a *p* betűvel kezdődnek az angol parallel szó rövidítéséből adódóan. A digitalizált dokumentumok azonosíthatóságát biztosító adatelemeknek megfeleltetett PoMeDS elemek többségénél a Dublin Core és a VRA Core terminológiája lett alapul véve.

```
<xs:element name="ptitleproper" type="xs:string"/>
```

A PoMEDS egy olyan hierarchikus séma, amiben konténer elemek kerültek alkalmazásra. Ilyen például a <poster> elem, amely tartalmazza az <originaldocument> és a <digitizeddocument> alelemeket, vagy az <edition> elem, amely az <editionstatment>, a <pointspacedashspace> és a <statofresprelatingtotheedition> alelemeket tartalmazza. A metaadat értékeket az alsó szintű alelemekbe tudjuk beírni. A következő példa a <series> konténer elemet mutatja be és az abban lévő alelemeket:

```
<series>
  <pointspacedashspace/>
  <roundbracket/>
  <titleproperofseries> O.T.I. Balesetelhárítási Propagandája
</titleproperofseries>
  <numberingwithinthseries>
    <semicolon/>
    <number>215.</number>
  </numberingwithinthseries>
  <roundbracket/>
</series>
```

A PoMEDS XML attribútumokat is használ az egyes elemek pontosítására, bizonyos kódolások vagy kontrolált szótárak jelölésére és egyéb hasonló célokra. Az attribútumok használatával kiküszöbölhetők a minősítők okozta olyan kellemetlenség, mint az újabb és újabb adatelemek definiálása. Továbbá így könnyebbé válik a könyvtárak közötti adatcsere, mivel elkerülhető a különböző minősítők használatából adódó elemkészletbeli különbségeket. Ezek az attribútumok a kezdő tagban szerepelnek. Az elemek egynél több attribútumot is tartalmazhatnak.

```
<notes>
  <firstnote type="description">
    <newindent/>
    <note>Teherhordó elefánt gázol át egy folyón, hátán egy indiai,
      turbános férfival, hatalmas ládákkal. A háttérben dombok,
      hegyek és palmafák.
    </note>
  </firstnote>
</notes>
```

Ez a példa is jól mutatja az attribútum fontosságát. Elég egy megjegyzés elemet deklarálni és az attribútumok segítségével pedig pontosítani lehet, hogy az adott megjegyzés mire vonatkozik, így nem kell például külön leírás elemet létrehozni.

A digitalizált plakátok bibliográfiai leírását segítő PoMeDS két részből épül fel: az egyik segítségével magát az eredeti dokumentumot dolgozzuk fel, a másik rész pedig az előzőt egészíti ki a digitalizált változatra jellemző adatokkal. A következő táblázat bemutatja a <originaldocument> és a <digitizeddocument> felső szintű elemek által tartalmazott alelmeket.

PoMEDS felső szintű elemei és alelemei	
Felső-szintű elem	Alelemek modellje (első szinten)
originaldocument	mainentry{0,1} , titleandstatofresp , edition{0,1} , publication , physicaldescription , series* , notes{0,1} , distributiondatas{0,1} , subjectaddedentry* , subjectheading+ , addedentry* , repository
digitizeddocument	title , statofresp+ , date+ , identifier+ , access{0,1} , right{0,1} , technique{0,1} , systemrequirements{0,1} , filecharacteristics , polarity , generation{0,1} , relations{0,1} , origdocbibldescr , cataloger
{0,1}: nem kötelező és nem ismételhető elemek +: kötelező és ismételhető elemek *: nem kötelező elemek, amelyek ismételhetőek jelölés nélküli elemek kötelező és nem ismételhető elemek	

Az eredeti dokumentum leírása a tizenkét nagyobb egységre tagolódik a bibliográfiai leírás adatszoportjainak és besorolási adatainak megfelelően:

- <mainentry> testületek, személyek, címek egységesített alakját tartalmazza, amelyek alapján az adott bibliográfiai leírást besoroljuk.
- <titleandstatofresp> cím és szerzőségi közlés adatainak megfeleltetett PoMeDS elemek, ahol plakát címadata és a mű létrejöttében, szellemi megformálásában részt vevők, közreműködők kerülnek felvételre.
- <edition> kiadásjelzés és az ehhez kapcsolódó szerzőségi közlés vehető fel ebben az adatszoportban.
- <publication> megjelenéssel, terjesztéssel kapcsolatos adatelemeket tartalmazza.
- <physicaldescription> olyan fizikai jellemzőkre vonatkozó adatokat tartalmaz, mint a fizikai egységek száma, egyéb fizikai jellemzők (színesség, előállítási technika), plakát terjedelme.
- <series> amennyiben a plakát sorozat tagjaként jelenik meg, akkor ebben az adatszoportban kerül felvételre a sorozat cím és szerzőségi adata.
- <notes> a bibliográfiai leírás adatait kiegészítő, pontosító megjegyzéseket, valamint a plakátra vonatkozó megjegyzéseket tartalmazza.
- <distributiondatas> a plakát terjesztésével kapcsolatos adatokat tartalmazza.
- <subjectaddedentry> a plakát tárgyát képező személy, testület, rendezvény földrajzi hely és cím egységesített alakját tartalmazza.
- <subjectheading> dokumentum tartalmát jellemző tárgyszavak.
- <addedentry> személy, testület, rendezvény, földrajzi hely és cím egységesített alakját tartalmazza.
- <repository> a plakátot őrző intézmény adatait tartalmazza.

A PoMeDS másik részében a digitalizált plakátra jellemző adatok jelennek meg úgy, mint cím, szerzőség, fájljellemzők, jogok és így tovább. A digitalizált plakátokra vonatkozó adatok tizennégy kisebb részből épülnek fel:

- <title>: A plakát digitalizált változatának adott cím vagy egyedi azonosító, ami megegyezhet az eredeti plakát címével.
- <statofresp>: A digitalizált plakát szerzőségi adata (pl. a digitalizálást végző személy vagy testület).
- <date>: A digitalizált plakátra vonatkozó dátum adat (ilyen lehet a digitalizálás időpontja, a közzététel időpontja stb.).
- <identifier>: A digitalizált plakát azonosítója (ez lehet ID, url stb.).
- <access>: A digitalizált plakát hozzáféréseire vonatkozó információ, mint a hozzáférés korlátja vagy a feltétele.
- <right>: A digitalizált plakátra vonatkozó jogok közlése.
- <technique>: A digitalizálás során használt technika.
- <systemrequirements>: A digitalizált kép meg tekintéséhez szükséges működési követelmények közlése.
- <filecharacteristics>: A digitalizált képet tartalmazó fájl jellemzői (pl. fájl név, formátum, méret stb.).
- <polarity>: A digitalizált kép polaritására vonatkozó információ, azaz a kép színei megfelelnek-e a valóságnak.
- <generation>: A generáció elemmel információt lehet rögzíteni arra vonatkozólag, hogy a digitalizált képet hányszor vitték át az egyik hordozóról a másikra.
- <relations>: A digitalizált kép kapcsolataira utaló információ.
- <origdocbibldescr>: Az eredeti dokumentum bibliográfiai leírására mutató OPAC URL vagy BIBID.
- <cataloger>: A digitalizált dokumentum adatait felvevő szakember megnevezése.

PoMeDS struktúrájáról teljesebb képet kapunk egy példán keresztül. A példában szereplő plakát egy Meinl teát reklámozó kereskedelmi plakát, amelyet Otto Exinger készített. A nyomtatott plakát leírása megtalálható a Debreceni Egyetem Egyetemi és Nemzeti Könyvtárának számítógépes katalógusában, digitalizált változata pedig a Debreceni Egyetem Elektronikus Archívumában. A következőkben látható a plakát bibliográfiai leírása, a digitalizált változat adatainak egy részlete és végül a digitalizált plakát adatai PoMeDS-ben.

```
000 01039nkm 2200277 i 4500
001 bibDEK00712409
005 20120127171508.0
008 081202s1933 hu hun d
040 $aHuDeKLEK
100 1 $aExinger, Otto$d(1897-1957)
245 10 $aMeinl Indiai Tea$h[vizuális dokumentum] /$cExinger
260 $a[Budapest] :$b[Floderer Vilmos],$c[1933]$f([Budapest] : Bakács
Litográfia Offsetnyomda)
300 $a1 plakát :$bgrafikus, színes ;$c63 x 47 cm
520 0 $aTeherhordó elefánt gázol át egy folyón, hátán egy indiai, turbános
férfival, hatalmas ládákkal. A háttérben dombok, hegyek és palmafák.
583 $xD1/ko
600 1 $aMeinl, Julius$d(1824-1914)
610 2 $aJulius Meinl (Budapest)
695 $akereskedelmi plakát
856 40 $3A plakát képe$uhttp://hdl.handle.net/2437/89952
949 $a50244$c1050672$d1D1/PL$tmp$z9990000000170168
```

```
<postercollection ID="bibDEK00712409">
<poster ID="PC0001">
<originaldocument type="graphic"
ID="bibDEK00712409">
<mainentry
<newindent/>
<personname standard="MSZ3440/2">
```



```

<modernantiquemedievalname>
<uniformname>Exinger, Otto</uniformname>
<nameaddinfs>
<nameaddinfsvar1>
<roundbracket/>
<nameaddinf type="date">
<addinf>1897-1957</addinf>
</nameaddinf>
<roundbracket/>
</nameaddinfsvar1>
</nameaddinfs>
</modernantiquemedievalname>
</personname>
</mainentry>
<titleandstatofresp>
<ptitleandstatofresp>
<newindent/>
<titlevar1>
<titleproper>Meinl Indiai
Tea</titleproper>
<generalmaterialdesign>
[vizuális dokumentum]
</generalmaterialdesign>
</titlevar1>
<diagonalslash/>
<firststatofresp>
Exinger</firststatofresp>
</ptitleandstatofresp>
</titleandstatofresp>
<publication>
<pointspacedashspace/>
<publisher>
<firstpublisher>
<placeofpublication>
[Budapest]
</placeofpublication>
<colon/>
<publisher>
[Floderer Vilmos]
</publisher>
</firstpublisher>
</publisher>
<dateofpublication>
<date>
<comma/>
<year>[1933]</year>
</date>
</dateofpublication>
<printer>
<roundbracket/>
<placeofprinter>[Budapest]</placeofprint
er>
<colon/>
<nameofprinter>Bakács Litográfia
Offsetnyomda</nameofprinter>
<roundbracket/>
</printer>
</publication>
<physicaldescription>
<pointspacedashspace/>
<specificmaterialdesignationandextent>
<numberofphysicalunits>1</numberofphysic
alunits>
<specificmaterialdesignation>
Plakát</specificmaterialdesignation>
</specificmaterialdesignationandextent>
<colon/>
<otherphysicaldetails>
<colours>színes</colours>
</otherphysicaldetails>
<semicolon/>
<dimension>63x47 cm</dimension>
</physicaldescription>
<notes>
<firstnote type="description">
<newindent/>
<note>Teherhordó elefánt gázol át egy
folyón, hátán egy indiai,turbános
férfival, hatalmas ládákkal. A háttérben
dombok, hegyek és pálmafák.</note>
</firstnote>
</notes>
<subjectaddedentry>
<star/>
<personname standard="MSZ3440/2">
<modernantiquemedievalname>
<uniformname>Meinl, Julius</uniformname>
<nameaddinfs>
<nameaddinfsvar1>
<roundbracket/>
<nameaddinf type="date">
<addinf>1824-1914</addinf>
</nameaddinf>
<roundbracket/>
</nameaddinfsvar1>
</nameaddinfs>
</modernantiquemedievalname>
</personname>
</subjectaddedentry>
<subjectaddedentry>
<star/>
<corporatename standard="MSZ3440/3">
<corporatenameentryword type="official
full name">
<entryword>Julius Meinl</entryword>
</corporatenameentryword>
<corporateaddinfs>
<roundbracket/>
<corporateaddinf type="chief town">
<addinf>Budapest</addinf>
</corporateaddinf>
<roundbracket/>
</corporateaddinfs>
</corporatename>
</subjectaddedentry>
<subjectheading subjectheadings="other">
<subject>kereskedelmi plakát</subject>
</subjectheading>
<repository>
<nameofrepository>Debreceni Egyetem
Egyetmi és Nemzeti Könyvtár
</nameofrepository>
<callnumber>50244</callnumber>
</repository>
</originaldocument>
<digitizeddocument>
<title>
<titlepropr>Meinl Indiai Tea
</titlepropr>
</title>
<statofresp>
<corporate>
<corporateentryword type="official
full name">Debreceni Egyetem
Egyetemi és Nemzeti Könyvtár

```

```
</corporateentryword>
</corporate>
</statofresp>
<date type="access">2009-11-05</date>
<identifier>
<locationofmasteritem>
http://dea.lib.unideb.hu/dea/
bitstream/2437/89952/1/DSCF2646.JPG
</locationofmasteritem>
</identifier>
<filecharacteristics>
<filename>DSCF2646.JPG</filename>
<format>
<formatname>JPEG</formatname>
</format>
<filesize type="filesize">1.88
MB</filesize>
</filecharacteristics>
<polarity>
<polaritytype>positive

</polaritytype>
</polarity>
</origdocbibldescr>
<bibid>bibDEK00712409</bibid>
<opacurl>
http://corvina.lib.unideb.hu:8082
/WebPac/CorvinaWeb?action=onelong&
showtype=longlong&idno=bibDEK00712409
</opacurl>
</origdocbibldescr>
<cataloger>
<person>
<uniformname>Komán Magdolna
</uniformname>
</person>
</cataloger>
</digitizeddocument>
</poster>
</postercollection>
```

A PoMeDS segítségével jelenleg a nyomtatott és a digitalizált plakátok bibliográfiai leírását lehet megadni. A séma továbbfejlesztése szükséges, hiszen digitális környezetben is jelennek meg plakátok. Így egy továbbfejlesztési irány lehet a digitális plakátok formai feltárását segítő séma kialakítása.

7. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2C-11/1/ KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

Irodalomjegyzék

- [1] BETZ Elisabeth W.: Grafic materials: rules for describing : original items and historical collections. Washington D.C., LC, 1982. 174 p.
<http://www.loc.gov/rr/print/gm/GraMatWP8.pdf> [2014. július 7.]
- [2] Bradley, Neil: Az XML kézikönyv, Bicske, Szak, 2005, p. 179-203.
- [3] Cseh Mária: Az utca művészete: képes plakátok Benczúr Gyulától Konecsni Györgyig., Cseh Mária (szerk.): A nemzeti könyvtár kisnyomtatványtára. Budapest, Osiris, 2002. p. 12–35.
- [4] Dublin Core honlapja. <http://dublincore.org> [2014. július 5.]
- [5] El-Sherbini, Magda; Klim, George: Metaadatok és katalogizálási gyakorlatok, Tudományos és műszaki tájékoztatás, 53. évf. (2006), 2. sz. – http://tmt.omikk.bme.hu/show_news.html?id=4306&issue_id=469 [2014. július 7.]

- [6] Koltay Tibor: Virtuális, elektronikus, digitális, Budapest, Typotex Elektronikus Kiadó Kft., 2007 <http://www.tankonyvtar.hu/hu/tartalom/tkt/virtualis-elektronikus/adatok.html> [2014. július 7.]
- [7] Magyar Plakát Ház. <http://kanizsaikultura.hu/plakat> [2014. július 5.]
- [8] Metadata Object Description Schema honlapja: <http://www.loc.gov/standards/mods/> [2014. július 7.]
- [9] Miller, Steven J.: Metadata for digital collections, New York, Neal-Schuman Publishers, cop. 2011, p. 166-167.
- [10] MSZ 3424/1-1978 Bibliográfiai leírás. Könyvek
- [11] Országos Széchényi Könyvtár Plakát- és Kisnyomtatványtár. http://www.oszk.hu/plakatok_kisnyomtatvanyok [2014. július 7.]
- [12] Salgáné Medveczki Marianna: Az XML: új perspektívák a könyvtár-informatikában, Tudományos és Műszaki Tájékoztató, 51. évf. (2004), 2. sz., p. 61-71.
- [13] Salgáné Medveczki Marianna: Egy új jelölő nyelv (metaadat-rendszer) kidolgozása a számítógépes bibliográfiai adatfeldolgozáshoz. PhD értekezés, Debrecen: Debreceni Egyetem, 2006. 141 p.
- [14] Ungváry Rudolf, Vajda Erik : Könyvtári információkeresés, Budapest : Typotex, 2002. 169 p.
- [15] VARGA Ildikó (összeáll.): Állóképek bibliográfiai leírása : útmutató. Budapest, OSZK-KMK, 1981. 33 p.
- [16] VRA Core honlapja. <http://www.vraweb.org/projects/vracore4> [2014. július 7.]
- [17] Young, Michael J.: XML lépésről lépésre, Bicske, Szak, 2002, p. 147-168.

Statisztikai adatok közötti véletlen összefüggések tanulmányozása

Random correlation analysis between statistics data

Bencsik Gergely^a, Bacsárdi László^b

^aNyugat-magyarországi egyetem, Informatikai és Gazdasági Intézet
bencsikg@inf.nyme.hu

^bNyugat-magyarországi Egyetem, Informatikai és Gazdasági Intézet
bacsardi@inf.nyme.hu

Absztrakt: Napjainkban számos elemző módszer áll rendelkezésünkre hogy a különböző tudományterületekről származó adatokat klasszikus és újszerű modellekkel írjuk le. Az adatelemzési folyamat során a modell és a kapcsolatok megtalálása már önmagában eredményt jelent, ezután következik a - lehetőleg minél pontosabb - jóslás a jövőre nézve. Ezen kategóriába tartoznak a statisztikai próbák, adatbányászati algoritmusok, regressziós elemzések.

Kutatásunk az adatok karakterisztikájára vonatkozik. A fenti különböző adatelemzési módszerek önmagukban nem képesek értelmezni az eredményeket, vagyis az adott matematikai képlettel csak kiszámoljuk az eredményeket. A módszerek önmagukban nem ítélik meg, a megállapításokat, miszerint elfogadjuk vagy sem az adott összefüggést, mindig egy elemző személy teszi meg.

Kutatásunkban azt vizsgáltuk, hogy mi történik akkor, ha az elemző által keresett összefüggés matematikailag bizonyítható, de az adott elemzési döntés mégsem helytálló, mivel a matematikai összefüggés a mért adatok olyan véletlenszerűségéből adódik, amelyet az elemző személy sem ismerhet. Az olyan összefüggéseket, amelyek a véletlenszerűség következményeként létrejönnek, véletlenszerű kapcsolatoknak nevezzük. Munkánkban bemutatjuk, milyen módszerekkel lehet a fenti véletlenszerű kapcsolatokat megközelíteni; ismertetjük azon körülményeket, ahol ilyen kapcsolatok létrejöhetnek, és értelmezzük az ebből levonható következtetéseket.

Kulcsszavak: döntéstámogatás, véletlen kapcsolatok, statisztikai adatelemzés

Abstract: There are several methods to analyze data coming from different scientific fields. Following different predictions, correlations are found in these processes. However, these analyzing methods cannot interpret the results, this can be done by an expert as part of the knowledge management process. The given results is judged as satisfied or not. This category contains statistics tests, data mining techniques and regression analysis. Our research is focuses on data characteristics and their relations with different methods. We focuses on cases when the correlation can be proofed mathematically but the thesis are not valid since the results can be derived by only randomization which could be unknown for the experts as well. If the correlations are happened only by some unknown random effects, we talk about random correlations [RC]. We introduce RC analyze techniques and review the circumstances when RCs can occur. In this paper, a standard framework is defined, which can support to examine different RC cases. We summarize the results performing RC analyses.

Keywords: decision support, random correlations, statistical analysis

1. Bevezetés

Kutatásunkat kezdetben döntéstámogató rendszerekben végzett elemzésekkel kezdtük meg. A növekvő számú és különböző tudományterületekről származó adatsorok elemzése közben egyre többször fogalmazódott meg, hogy ezen módszerek használata mennyire ad megbízható eredményt. Az eredmények ellenőrzésének két lehetséges módja van: (1) elő tudunk állítani egy olyan objektív mérőszámot, amely megmondja, hogy az adott eredmény mennyire helytálló, vagy (2) szakértők bevonásával és a tudásmenedzsment eszközeivel fogadjuk, vagy utasítjuk el a talált összefüggést. Ezen túlmenően az elemzések iterációs végrehajtása során általában mindig kaptunk olyan eredményt, amelyet különböző technikákkal [pl.: paraméterek javítása, outlier analízis] javítani is tudtunk.

A tudományos kutatások általános kivitelezésére több publikáció is megjelent, és sok ilyen értekezés adott tudományterület speciális tulajdonságait is figyelembe veszi [1-4]. Korábbi kutatásainkban követtük ezen state of art jellegű szabályokat, ugyanakkor az adatelemzés iterált végrehajtása eredményeképpen előbb utóbb megkapott megfelelő eredmény azt a feltételezést indukálta, hogy az összefüggések jelentős része véletlenszerűségből adódhat. Másképp megfogalmazva képesek vagyunk úgy manipulálni az egyébként szabálykövető kísérleti eredményeket, hogy azok, ha nem is teljes mértékben, de meggyőzően hozzák az általunk várt eredményeket. Ez nem szándékos torzítást, hanem olyan véletlenszerű kapcsolatok létét feltételezi, amelyet a kísérletet végző személyek sem ismerhetnek. Ez lehet az oka, hogy sok esetben hasonló módszerekkel, ugyanazon kérdésre vonatkozó, de egymásnak ellentétes eredményeket olvashatunk tudományos cikkekben. Ezek száma növekvő és az ellentét nagymértékű, teljesen kizáró. Ha megvizsgáljuk biológia területén a sófogyasztás hatásait, akkor teljesen eltérő magatartást látunk. Létezik olyan publikáció, amely abszolút pozitívan nyilatkozik a sófogyasztásról és nem hozza összefüggésbe betegségekkel, mint például a magas vérnyomás [5]. Ugyanakkor létezik olyan eredmény is, amely azt mutatja, hogy nem csak a magas vérnyomásra van hatással a só, hanem sokkal több betegséget, például vesebetegségeket is okozhat [6]. Több tudományterületen hasonló eredményeket tapasztalhatunk. Az előzőekkel összefüggésben kételkedés merült fel, hogy az „elfogadott” eredmények mindig hozzák-e az elemzés utáni jó validációs és/vagy jóslási eredményeket is. Létezik olyan megállapítás, miszerint a matematikailag kimutatott összefüggés még nem jelenti a tényleges kapcsolatot az adatok között. Ezzel egyetértünk, ugyanakkor ennek ellenére sokszor tényszerűen kezeljük az eredményeket. Ezt az is erősíti, hogy semmilyen kutatást nem találtunk a véletlenszerűségek elemzésére és kiszűrésére vonatkozóan, bár tapasztaltunk olyan ökölszabályszerű kijelentéseket, miszerint tíz esetből kettő véletlenül is összefügghet. A tényszerűség és a véletlenszerűség közötti rést akarjuk áthidalni kutatásunkkal.

Kutatásunkban arra a kérdésre kerestük a választ, hogyan lehetséges pontosan definiált kutatási környezetben mért adatokon végzett elemzések „elfogadott”, de véletlenszerűen létrehozott eredményeit kivitelezni. Az ilyen kapcsolatokat véletlen összefüggéseknek neveztük el. A cikk foglalkozik a véletlen összefüggések osztályaival, valamint definiálunk egy olyan egységes keretrendszert, amelyen belül tanulmányozhatóak a véletlen összefüggések. Továbbá meghatározzuk azon paramétereket és körülményeket, ahol ilyen összefüggések létrejöhetnek. Végül két módszer érzékenységét teszteljük véletlenszerűség szempontjából és értelmezzük az eredményeket. Bemutatásra kerül két olyan technika is, amellyel a véletlen összefüggéseket más módszerek esetén is lehet tesztelni.

2. Véletlenszerű összefüggések és osztályai

Általánosságban azon módszert illetve módszerek sorozatát keressük a kutatás során, amellyel összefüggéseket tudunk találni a különböző változók között úgy, hogy azok véletlenül fűggnnek csak össze. Arra vonatkozóan vonunk le következtetéseket, hogy adatsorok [mint valószínűségi változók] a kimutatott [matematikai] kapcsolaton túlmenően valójában nem biztos, hogy összefűggnnek. Ebben a fejezetben az ilyen összefűggnések egyes irányait foglaljuk össze.

Ahogy korábban megfogalmazztuk, viszonylag sok tudományos módszer létezik az adatsorok elemzésére vonatkozóan és amennyiben nem találunk összefűggnést egy adott módszerrel, akkor addig próbálkozunk más módszerekkel, amíg összefűggnést nem találunk. A célunk hogy az adatsorokat elemezve egy adott matematikai módszerrel kapcsolatot találjunk, és a keresést addig folytatjuk, amíg csak lehetséges. Másik oldalról nem definiáljuk [nem tudjuk definiálni], hogy a két adatsor mikor nem függ össze. A módszerek számát még növeli a paramétereizhetőségük változatosága, illetve a hibahatár változatos megválasztása, amely 20% is lehet. [Például egy determinisztikus egyűthatható 0,8-as értéke jó eredménynek számít, de tapasztaltunk már 0,4-es r^2 értéket, mint elfogadott eredmény.] Problémaként fogalmazzuk meg, hogy ugyanazon adatsor több és több algoritmussal történő elemzésének eredményeképpen nem a tényleges összefűggnést találjuk meg, hanem csak véletlenszerűen találunk egy összefűggnést. Úgy is fogalmazhatunk, hogy az elemzések számának növelésével biztos lesz legalább egy olyan módszer, amely jó eredményt ad [nagy számok törvénye]. Ez azért is lehetséges, mert a „nem összefűggnést” csak az adott, de összefűggnést nem produkáló módszer szempontjából tudjuk definiálni. Viszont fordítva is igaz, az összefűggnés csak az adott módszer értelmezésében összefűggnés.

Az előző osztálynál nagyobb problémát jelent, ha létezik két olyan módszer, amely egymással szemben mutat ki összefűggnést. Mindkét módszer matematikailag helyes, mégis egymásnak eltérő ítéleteket vonhatunk le. Mivel az elemzések nagy része abbamarad, amikor már az adott módszerrel megtaláltuk az adott összefűggnést, ezért sok esetben nem is derűlhet ki, hogy egy másik megközelítésben épp az előző eredmények alapján történő magatartás ellentéte is kimutatható. Inkább az adott módszer pontosabb felparaméterezésének kutatása jellemző. Ezen iránynak két aletese lehetséges:

- (a) Amikor két [vagy több] módszer egymás mellett ad matematikai összefűggnést;
- (b) Amikor a két [vagy több] módszer egymásnak ellentétes [egymást kizáró] eredményt ad.

Az (a) eset nagyon jó, hiszen ugyanaz az összefűggnés több nézőpontból is helyes, ezzel is erősítve a kapcsolatot a kettő vagy több adatsor között. Ennél rosszabb a (b), az egymást kizáró eredmények megléte, hiszen itt nem lehet kialakítani közös ítéletet. Mivel feltételezzük, hogy a mért adatok tulajdonképpen véletlenül is adhatnak összefűggnéseket, ezért létezheth kettő [vagy több] olyan módszer, amely ugyanazon adathalmazok esetén mindig ellentétes eredményt mutat. A kutatás másik célja ilyen módszerek felkutatása, amely egyben a véletlenszerűen kialakuló kapcsolatok ezen osztályának speciális eseteit is bemutatja:

- (1) Ha létezik egy adott módszer az összefűggnés kimutatására, akkor találunk legalább még egy módszert, amely legalább olyan jó eredményt ad;
- (2) Amennyiben egy adott módszer rossz eredményt ad, egy másik módszer nagy valószínűséggel jó eredményt fog adni. [Saját értelmezésben ezek az ún. inverz módszerek.]
- (3) Önmaga az adott módszer az „összefűggn” vagy „nem függ össze” ítélete is értelmes, azaz mindegyik változatnak van tudományos eredménye.

A későbbiekben a (3) esettel fogunk foglalkozni.

A harmadik főosztály az adatok növekvő tömegére vonatkozik. Általános az a megfogalmazás, hogy minél több adatunk van, annál pontosabb eredményeket tudunk generálni. Másik oldalról ellentmondásos az a megközelítés, miszerint az adott adatsor egyes része [például az adatgyűjtés kezdetétől a T . időpillanatig] más eredményt ad ugyanazon adatsor egy nagyobb részéhez képest [például az adatgyűjtés kezdetétől a $(T + K)$. időpillanatig beszerzett adatértékek]. Ez azért kritikus, mert nem tudhatjuk, hogy az adatgyűjtésben pontosan hol tartunk, így – bár az elemzés pontos és „jó” eredményt ad adott ideig – a jövőre nézve a felállított modell nem hozza a kívánt pontosságú eredményeket. A jelenségnek két megközelítése lehet értelmezésünkben: (1) az adatsor elemeinek a száma [illetve számának változtatása] önmagában okozza a véletlen összefüggést, valamint (2) rejtett paraméterrel vagy paraméterekkel magyarázható ez az osztály. Ha a rejtett paraméter, amely nem szerepelt a kezdeti tényezők halmazában, tudomásunk nélkül változik meg, akkor a modell összeomlik. A jelenben csak azzal az adathalmazzal dolgozhatunk, amely rendelkezésünkre áll, ezért szükséges olyan kritérium megfogalmazása, amely biztosítja, hogy nem pusztán véletlenszerű összefüggésekről van szó az adott adathalmazra vonatkozóan. A véletlenszerű kapcsolatok kialakulásának kivédése felé egy utat jelenthet a következő megállapítás: amennyiben egy adatsor véletlenszerűen kiválasztott részhalmazai nem teljesítik az egyébként teljes adathalmazra vonatkozó összefüggés meglétét, úgy pusztán nagy valószínűséggel véletlenszerű kapcsolat áll fenn. Ennek bizonyítására megfelelő eszköz a keresztvalidáció.

Összefoglalva három fő osztályt definiálunk véletlenszerű kapcsolatok szempontjából:

- (1) Minél több módszert alkalmazunk egy adott adatsoron, annál nagyobb a valószínűsége, hogy találunk összefüggést az adatok között;
- (2) Léteznek olyan módszerek, amelyek egymás mellett [megalapozott döntés], vagy egymással szemben [nem megalapozott döntés] mutatnak ki matematikailag helyes összefüggéseket;
- (3) Az adathalmazok részhalmaza más eredményeket mutat az adott módszert tekintve.

A véletlenszerű kapcsolatok létrejöttét még a korábban említett módszertanok is – tudtukon kívül – segítik. Előfordul, hogy még statisztikailag ellentmondó eredmény is konszenzusra hozható [7]. A korábban megfogalmazott sófogyasztási szokásokkal kapcsolatban is sok cikk kerül bizonyos paramétereket, amelyek növelhetik [növelték] a véletlenszerűség bekövetkezését, pedig a tényleges kérdés az lenne, hogy valóban káros vagy valóban ártalmatlan a magas sófogyasztás.

3. Véletlenszerű kapcsolatok elemzési keretrendszere

A korábban megfogalmazott irányokat egy egységes környezetbe helyezzük el, amelyben meghatározzuk azokat a paramétereket, amelyeket figyelembe véve értékelhetőek a véletlen összefüggések. Ez a paraméterhalmaz a következő:

- (1) Adatsorok száma (k);
- (2) Adatértékek száma adott adatsor esetén (n);
- (3) Adatértékek terjedelme (r);
- (4) Módszerek száma (t).

Az adatok terjedelme az a tartomány, ahová az egyes mért értékek eshetnek. Megjegyezzük, hogy a tartomány korlátozása fontos feltétel, de elemzéseknél nem jelent feltétlenül nagy megkötést. Sok esetben az intervallum „megkötés nélkül” is korlátos. Például erdészetben egy fa nem nőhet egy évben egy métert, hasonlóan olyan is ritka, hogy semennyit sem nő, ezért található alsó és felső korlát, vagyis a mérési értékek korlátosnak tekinthetők. Hasonlóan van a serdülőkorban lévő gyerekekkel is. Ugyanez mondható el szociológiában és

orvostudományban is. Amennyiben egy orvosság hatékonyságát vizsgálják, akkor pontosan előírt kísérlettervezés után adott korlátos számokkal mutatják ki, hogy a jelenlegi gyógyszerek mellett az új orvosság hatékonyabb-e vagy sem. A korlátosság megfigyelhető a fizika és az elektromágnesesség területén is. Ehhez szorosan kapcsolódik, hogy nagyon sokszor a kiugró eredményeket mérési hibáknak [outliereknek] veszik, így még jobban igaz a korlátosság, bár nyilván ténylegesen hibás értékeket nem lehet bevenni az elemzésbe. Mégis a legnagyobb korlátosság a kérdőívekre jellemző, ahol általában az adott kérdésekre adható válaszok korlátosak, hiszen általában 1-től 3-ig, vagy 1-től 5-ig adhatóak. Nehéz elképzelni olyan kérdőívet, ahol a skála nem kötött, vagy nagy tartományt [1-100] ölel fel.

A korlátosság kritériumból következik az a tétel, amely szerint, ha az adott módszer input paramétereinek és/vagy input értékeinek tartománya a számegyenesen meghatározott intervallumba esnek, továbbá ha ebből adódóan a kimenetei eredmények értékeinek numerikus tartománya is meghatározott, akkor az „összefüggés” és „nem összefüggés” ítéletek arányának adottnak kell lennie. Ez az arányszám változhat az egyes véletlen összefüggések vizsgálatakor. A későbbiekben látjuk, hogy van olyan eset, ahol ez az arányszám körülbelül 50%-nak kell, hogy legyen, de vizsgálunk olyan esetet is, ahol kisebb arányszám is elég erős a véletlenszerű kapcsolatok feltételezéséhez.

Egy adott analízisnél a következő lépéseket tesszük meg:

- (1) A módszer alapvető matematikájának ismertetése;
- (2) Véletlen összefüggés csoportjába történő osztályozás;
- (3) A módszer esetében a véletlen összefüggés jelenség pontos definiálása;
- (4) A paraméterek, különösen a tartomány paraméter definiálása;
- (5) Bizonyítás és validáció számítógépes szimuláció segítségével.

A további fejezetekben ezt a módszertant követjük és mutatjuk be a módszerek véletlen összefüggéseit. Két olyan módszert vizsgálunk, amelyek két külön osztályba tartoznak:

- (1) ANOVA próba, amely statisztikai módszer és a második osztályba tartozik;
- (2) Regresszió, amelynek több fajtáját is vizsgáljuk és az első osztályba tartozik.

Mielőtt rátérnénk az eredményekre, meg kell jegyeznünk, hogy semmilyen módon nem vagyunk a tényleges kapcsolatok léte ellen. Nem a valódi összefüggések létét tagadjuk, csupán azt mondjuk, hogy bizonyos körülmények között magas valószínűséggel jönnek létre véletlenszerű kapcsolatok.

4. ANOVA véletlen összefüggései

Ebben a fejezetben a varianciaanalízisre [ANOVA] koncentrálunk, valamint bemutatjuk az első módszert, amellyel kiszámítható az ANOVA érzékenysége.

A módszer alapötlete, hogy kiszámoljuk az összes lehetőséget, majd abból meghatározunk egy arányszámot, amely megmutatja a „összefügg” és a „nem függ össze” ítéletek arányát. Ez a technika minden módszer esetében alkalmazható és az arányszám megmutatja a véletlenszerűségekre való érzékenységet. Ha például van egy kérdőívünk, ahol 5 kérdést teszünk fel, és a kérdésekre egyenként 3 [1, 2, 3] válasz adható, valamint mondjuk 5 emberrel töltöttük ezt ki, akkor az összesen $3^{5 \cdot 5} = 8,472 \cdot 10^{11}$ esetet jelent. Előállítjuk az összes esetet, majd minden esetre alkalmazzuk az ANOVA-t és vizsgáljuk a keresett arányszámot. A módszer hátránya a nagy számítási idő, mert a későbbiekben látni fogjuk, hogy az összes eset előállítása nem mindig valósítható meg valós időben.

A következőkben az ANOVA-ra vonatkozóan végigvesszük a véletlen összefüggések elemzésének lépéseit. Először a módszer alapvető matematikáját ismertetjük, amelyet a következő táblázatban foglalunk össze:

Forrás	Összes eltérés	Szabadságfok	Átlagos eltérés	F
Csoportok között	SSB	$k-1$	MSB	$\frac{MSB}{MSW} = F$
Csoportokon belül	SSW	$k * (n - 1)$	MSW	
Teljes	$SST=SSB+SSW$			

1. táblázat. ANOVA számításának összefoglalása

$SSB = \sum_{j=1}^k n_j * (\bar{x}_j - \bar{x})^2$ és $SSW = \sum_{j=1}^k \sum_{i=1}^{n_j} (x_{i,j} - \bar{x}_j)^2$, tovább k az oszlopok száma és n az elemek száma. A t paramétert itt nem használjuk és egyébként is igaz, hogy a keretrendszer nem minden elemét szükséges minden módszer elemzésénél felhasználni. A következőkben definiáljuk, hogy értelmezzük a véletlen összefüggést az ANOVA-ra vonatkozóan és hogyan tudjuk kiszámolni a keresett arányszámot.

ANOVA esetében az egyes értékek terjedelmének megszabása az MSB [csoportok közötti eltérések] és az MSW [csoportokon belüli eltérések] értékeit is meghatározzák, és ezzel együtt az F próbastatisztika értéke is korlátos. Másként megfogalmazva adott terjedelmű mért értékek adott terjedelmű F értékeket eredményeznek. Mivel az ítélet szempontjából fontos lehet mind a H_0 , mind a H_1 hipotézis, ezért ugyanakkora valószínűséggel kell bekövetkeznie egyiknek, mind a másiknak. Ellenkező esetben az egyik oldal elsőbbséget élvez az ítélet meghozatalakor, ami véletlen ítélethez vezethet.

Meg kell határozni adott k , n , illetve terjedelem ismeretében az összes lehetőséget. Amennyiben az ANOVA bemenetét egy mátrixként fogjuk fel, akkor ki kell számolni a mátrixban előforduló összes kombinációt. Ebben a teljes eseményrendszerben vizsgáljuk meg az „összefügg”-„nem függ össze” esetek számát, amelyeknek aránya ANOVA esetében 50% körülnek kell, hogy legyen. Nevezzük ezt a valószínűséget $\Omega(F_{k,n,a,b})$ -nek. Nyilván ez a arány nem szigorúan kötött, de másik oldalról nagyon torz sem lehet. Amely kombinációk elfogadják a H_0 -t, azokat 1-sel jelöljük [„átlagok egyenlők”]. Értelme szerűen 0-t kapnak a H_1 -t elfogadó esetek [„nem egyenlők”]. A keresett arányszám az 1-sek és 0-k aránya. Természetesen az összes lehetőséget kézzel kiszámolni nagyon hosszadalmas lenne, ezért egy programmal kalkuláljuk az arányszámot. Ez előremutató abban a tekintetben is, mivel nemrég született meg az első számítógéppel történő bizonyítás Erdősi tételére vonatkozóan [8].

A következőkben változtattuk a k , n , és r paramétereket és az eredményeket a következő táblázatokban foglaltuk össze:

$r(1,3)$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$	$k = 8$	$k = 9$	$k = 10$
$n = 3$	93.8424	94.8517	94.2379	94.2995	94.5072	94.5324	94.5904	94.5307
$n = 4$	94.9195	94.5366	94.3690	94.3732	94.4812	94.6244	94.6660	94.7394
$n = 5$	94.6713	94.4210	94.5278	94.5478	94.7034	94.7130	94.7565	94.8103
$n = 6$	94.5788	94.4749	94.5947	94.7099	94.7352	94.7324	94.7677	94.7734
$n = 7$	94.6950	94.6260	94.6334	94.7715	94.7911	94.8055	94.8238	94.8500
$n = 8$	94.6579	94.7205	94.7438	94.8090	94.7997	94.8387	94.8280	94.8553

2. táblázat. ANOVA első eredményei

Látható, hogy nagyon magas százalékokat kaptunk, amelyek értelmében a H_0 elfogadásának valószínűsége nagyon magas. Vizsgálatunk folytatása a korábban meghatározott szisztematikus módon történik, növeljük a k és n értékeket. Az $r(1,5)$ és $r(1,10)$ tartományokat vizsgáltuk továbbá. Hasonlóan magas értékeket kaptunk ezen esetekben is, a H_0 mindig elsőbbséget élvezett a H_1 -gyel szemben.

Ugyanakkor az ANOVA alkalmazásának meghatározott feltételei vannak, amelyeket egy elemzés során be kell tartanunk. Ezek a feltételek a következők:

- (1) A mintákat véletlenszerűen választottuk ki;
- (2) Az egyes minták szórásnégyzetei statisztikailag megegyeznek;
- (3) A minták normál eloszlást követnek.

A pontos eredményekhez alkalmaznunk kell ezen feltételeket a teljes eseményrendszer kiszámításakor és a keresett arányokat ennek megfelelően kell kiszámolni. A feltételeket beépítettük abba a szimulációs programba, amely előállította az eredményeket. Normalitás vizsgálathoz a *D'Agostino-Pearson* tesztet [9] használtuk, míg a szórásnégyzetek egyenlőségét *Bartlett* próbával [10] ellenőriztük. Értelemszerűen azon kombinációkat, amelyek nem mentek át a teszteken kivettük a teljes eseményrendszerből, és nélkülük határoztuk meg az arányszámokat. Eredményeinket a következő táblázatokban foglaltuk össze.

r(1,3)	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9	k = 10
n = 4	99.1770	99.7866	99.7968	99.9116	99.9486	99.9648	99.9728	99.9838
n = 5	99.4000	99.3800	99.6800	99.7778	99.8117	99.8975	99.8791	99.9334
n = 6	99.5326	99.8606	99.9162	99.9511	99.9699	99.9821	99.9887	99.9929
n = 7	98.5536	98.8494	99.1591	99.3574	99.5113	99.6108	99.6796	99.7337
n = 8	99.4011	99.5661	99.7306	99.8324	99.8876	99.9227	99.9463	99.9623

3. táblázat. ANOVA elemzése feltételekkel $r(1,3)$ tartományban

r(1,5)	k = 3	k = 4	k = 5	k = 6
n = 3	91,8654	91,7281	91,5740	91,4789
n = 4	95,4156	95,5324	95,7285	95,9544
n = 5	95,6233	95,8425	96,0967	96,3240
n = 6	97,1799	97,4015	97,6664	97,9025

4. táblázat. ANOVA elemzése feltételekkel $r(1,5)$ tartományban

r(1,10)	k = 3	k = 4
n = 3	78,4756	74,0973
n = 4	88,2454	86,8794

5. táblázat. ANOVA elemzése feltételekkel $r(1,10)$ tartományban

Az eredményeken látszódik, hogy még mindig elsőbbséget élvez a H_0 , ugyanakkor, ahogy a tartományt növeljük, úgy ez az elsőbbség csökken. Ez mindenképpen iránymutató lehet az ANOVA véletlen összefüggésre való érzékenységének további vizsgálatához.

Az Ω kiszámítás módszerének hátránya, hogy nagyon időigényes az összes kombináció előállítása. Ha például $k = 3$ és $n = 4$, valamint a $r(1,5)$, akkor az összes lehetőség $5^{3*4} = 5^{12} = 2,441 * 10^8$. Ha növeljük az értékeket, akkor jóval nagyobb eseményhalmazt kapunk. Sajnos a teljes eseményrendszer kiszámítása nem mindig valósítható meg valós időben, ezért nem folytattuk a táblázatokat, ugyanakkor végeztünk kísérleteket kisebb $k = 3$, $k = 4$ és növekvő n esetében. A tartományt a számítási kapacitások csökkentése miatt szintén kisebbre [$r(1,3)$; $r(1,5)$] választottuk. Hasonlóan magas eredményeket kaptunk, mint a megfelelő táblázatok értékei. A továbbiakban egy másik módszert mutatunk be, amely képes a véletlen összefüggésre való érzékenység kimutatására.

5. Véletlen összefüggés szimulációval

Ebben a fejezetben egy másik eljárást ismertetünk, valamint a korrelációanalízis véletlen összefüggésekre való érzékenységet vizsgáljuk. Ez a módszer gyorsabb számítást tesz lehetővé, ugyanakkor csak közelítő értékeket ad. Ebben az esetben a keretrendszer összes eszközt felhasználjuk.

A statisztika axiómaként tekint a normál eloszlásra. Ha normál eloszlás szerint generálunk adatsorokat [feltételek], akkor megspórolhatnánk az összes kombináció előállítását, amely nagyon sok időt vesz igénybe. Egy ilyen esetben értelemszerűen szükségtelen a normalitásvizsgálat is. Megjegyezzük, hogy ez csaknem az összes statisztikai próbát érinti, hiszen mindig a normál eloszlásból indulunk ki. Ugyanakkor a normál eloszlás sűrűségfüggvénye zárt formában nem adható meg, így a vele történő számolás és szimuláció nehézkes lehet. Megoldásként szükséges egy hasonló alakú és funkciójú eloszlás, amelynek a sűrűségfüggvénye zárt formában előállítható. Ez a feltétel később a szimulációs programnál is fontos kritérium. A feltételnek a Cauchy-eloszlás felel meg, azonban ezután a végleges teszthalmaz előállításához még transzformációkra van szükség, amelyet a következőkben ismertetünk.

A számítógép mindig $[0, 1]$ intervallum között generál számokat egyenletes eloszlás alapján. Ezen (ál)véletlen számokat kell transzformálni, hogy majd az eredményként megjelenő értékek Cauchy-eloszlás alapján generálódjanak. A műveletet a következő integrál adja meg:

$$\int_{-\infty}^{+\infty} c * \frac{1}{1+x^2} = c * [\arctg x]_{-\infty}^{+\infty} = c * \left(\frac{\pi}{2} - \left(-\frac{\pi}{2} \right) \right), \quad (1)$$

ahol $c = \frac{1}{\pi}$. A Cauchy sűrűségfüggvényből egy eloszlásfüggvényt állítunk elő. Látható továbbá, hogy a Cauchy eloszlás primitív függvénye az arkusz tangens függvény. Ez az eredmény programozás technológiai szempontból is érdekes, hiszen nem csak a $[0, 1]$ intervallumban szeretnénk értékeket generálni, hanem a $(-\infty, +\infty)$ között. Szükséges egy leképezés a $[0, 1]$ zárt intervallumból a $(-\infty, +\infty)$ nyitott intervallumra. Ezt a leképezést is alakjából kifolyólag teljesíti az arkusz tangens függvény. Mivel a számítógép által generált értékek, mint függvényértékek jelennek meg, ezért az arkusz tangens inverz függvényével kell számolni, azaz végeredményként egy tangens függvényt kapunk, amelyet a következő formában használtunk fel:

$$x_i = mean + dev * \tan(\pi * (rand_i - 0.5)), \quad (2)$$

ahol x_i a keresett adatérték, $mean$ a feltételezett eloszlás átlaga, dev a szórása és a $rand_i$ pedig a számítógép által generált $[0, 1]$ közötti egyenletes eloszlású véletlen szám. Az előzőeket végrehajtva egyszerűen előállíthatóak azok a véletlenszerűen generált adatsorok, amelyek között vizsgáljuk az összefüggések karakterisztikáit. Mielőtt a konkrét vizsgálatra rátérnénk, megjegyezzük, hogy a normális eloszlást a Cauchy eloszlással csak közelítjük a korábban megfogalmazott ok miatt. Konkrétan nem adott paraméterű normális eloszlást követnek a teszthalmazok értékei, ugyanakkor közelítő eljárásokkal közel normál eloszlás szerint is előállíthatóak az adatok. Ez sokkal gyorsabb módszer, mint az egész eseményrendszer előállítása. Hátránya, hogy a keresett arányszámot csak közelítően tudjuk előállítani. Ezért megfelelően nagyszámú szimulációra van szükség, amely azonban még mindig sokkal kisebb időt vesz igénybe, mint Ω előállítása.

A véletlen kapcsolatok (1) csoportjába tartozik a regressziószámítás és az ezzel szorosan összekapcsolódó determinisztikus együttható $[r^2]$. Erre a problémakörre megfogalmazva azt állítjuk, hogy adott adatsorokat minél több regressziós módszerrel vizsgálva nagy valószínűséggel lesz legalább kettő, amely összefügg. Ebben az esetben az összefüggés

klasszikus mérőszáma az r^2 , amely mindig 0 és 1 között vehet fel értéket. Itt k jelenti az adatsorok számát, n az egyes adatsorokhoz tartozó értékek számát, r a koordináták lehetséges értékeit, t pedig a regressziós módszerek számát, amellyel a generált adatsorokat elemezzük. A valószínűség alapján, ha minél többször állítjuk elő az adatsorokat, akkor megfelelően nagy szám felett egyre inkább megközelítjük az elméleti valószínűséget. Kutatásunkban a lineáris és kvadratikus regressziót vizsgáltuk [$t = 2$]. A korábbiaknak megfelelően változtattuk paramétereinket és az eredményeket egy táblázatba foglaltuk össze. A táblázat értékei azt jelentik, hogy adott k , n , r és t esetében mennyi korrelációt találtunk, úgy hogy közben 1000 iterációt hajtottunk végre. Akkor jegyeztük fel az összefüggést, ha r^2 értéke legalább 0,7 volt.

r(1,5)	k = 5	k = 10	k = 15	k = 20
n = 10	102	490	1240	2206
n = 20	10	21	48	93
n = 30	0	2	1	10
n = 40	0	0	0	1
n = 50	0	0	1	0

6. táblázat. Regressziószámítás eredménye 1000 iterációval, $t = 2$

A k növelésével növekszik a talált korrelációk száma is. Ugyanakkor n növelésével ez csökkent. A tartomány változtatásával is ugyanez a tendencia figyelhető meg. A regressziós számok növelésétől a számok javulását várjuk, de ez még továbblépési lehetőség. Ugyanakkor erős az a gondolat, hogy „megfelelően” választott k , n , r és t paraméterek mellett létrejön összefüggés.

6. Konklúzió és továbblépési lehetőségek

Ebben a cikkben egy új elmélettel foglalkoztunk, amelyet véletlen összefüggéseknek nevezetünk el. Az elmélet szerint a matematikai eredményeken túlmenően véletlenszerűen is kaphatunk eredményeket, és ez a véletlenszerűség nagy valószínűséggel van jelen bizonyos esetekben. Bemutattuk a véletlen korreláció osztályait, majd egy olyan keretet definiáltunk, amelyben lehetőség nyílik az adott elemzések véletlenszerűségeire való érzékenységének mérésére. Az ANOVA statisztikai tesztet és a regressziószámítást elemeztük ebben az egységes keretben. Mindkét esetben tapasztaltunk érzékenységet, és az érzékenység a k , n , r , valamint t paraméterek mentén változik.

A teljes eseményrendszer kiszámításakor nehézkes a teljes tér kiszámítása, ezért elkezdtünk egy dimenziócsökkentő algoritmus fejlesztését, amellyel a számolás gyorsítható. A szimulált normáloszlású adatsorok technika is működőképes minden esetben és valós idejű eredményt ad. A számolásokat egy saját készítésű programban végeztük, a működést más implementációkkal validáltuk. Az ANOVA esetében magának a módszer helyességét Excellel ellenőriztük le. Az 1-sek és 0-k száma ki kell, hogy adja az összes lehetőség számát, amely egyértelműen ellenőrizhető a k és n paraméterekkel történő számítással. A regressziós módszerek helyességét szintén Excellel [lineáris eset], illetve az R szoftver segítségével ellenőriztük [kvadratikus eset].

A két számítási mód eredményeinek egy vagy több adott módszeren történő összevetésének tervezése egy triviális lépés továbblépési lépés. További statisztikai tesztek érzékenységét kezdtük el vizsgálni, különösen az ANOVA feltételeire koncentrálva. Az elemzésbe később adatbányászati algoritmusok is bevonhatóak. Kutatásunkkal a véletlenszerű eredményekre akarjuk felhívni a figyelmet, és a folytatásban szeretnénk, ha a tudományos módszertanok

közé bekerülne az az ellenőrzés is, amely csökkentené vagy kizárná a véletlen összefüggés lehetőségét. Ezáltal nem a tényleges összefüggések létét akarjuk kizárni, hanem ellenkezőleg, a pontos és valódi kapcsolatok valódiságát akarjuk erősíteni.

7. Köszönetnyilvánítás

Ez a tanulmány a TÁMOP-4.2.2.C–11/1/KONV-2012-0015 számú (Föld-rendszer) projekt keretében, az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

Irodalomjegyzék

- [1] Csermely Péter, Gergely Pál, Koltay Tibor, Tóth János: Kutatás és közlés a természettudományokban, *Osiris Kiadó* (1999) Budapest, ISBN: 9633795281.
- [2] Petter Lake, Haakon Breien Benestad, Bjorn Reino Olsen: Research Methodology in the Medical and Biological Sciences, *Academic Press* (2007), ISBN: 9780123738745.
- [3] J. A. Khan: Research methodology, *Aph Publishing Corporation* (2008), ISBN: 9788131301364.
- [4] Ananya Mohapatra, Pradyout Mohapatra: Research methodology, *Partridge Pub* (2014), ISBN: 9781482817904.
- [5] Lee Hooper, Christopher Bartlett, George Davey Smith, Shah Ebrahim: Systematic review of long term effects of advice to reduce dietary salt in adults, *British Medical Journal [BMJ]* **325** (2002), 628–632, ISSN: 0959-8138.
- [6] Steve Pljesa: The impact of Hypertension in Progression of Chronic Renal Failure, *Bantao Journal* **1** (2003), 71-75, ISSN: 1312-2517.
- [7] Geoff Cumming: Understanding the new statistics, *Routledge* (2011), ISBN: 978-0-415-87968-2.
- [8] Boris Konev, Alexei Lisitsa: A SAT Attack on the Erdos Discrepancy Conjecture, <<http://arxiv.org/pdf/1402.2184v1.pdf>>, utolsó megtekintés: 2014. június 26.
- [9] Ralph B. D'agostino, Albert Belanger & Ralph B. D'agostino Jr.: A Suggestion for Using Powerful and Informative Tests of Normality, *The American Statistician* **44** (1990), 316-321, ISSN: 0003-1305.
- [10] Neil J. Salkind: Encyclopedia of Research Design, *SAGE Publications* (2010), ISBN: 9781412961271

A mátrix faktorizáció alkalmazásának korlátai

The bounds of the applications of the matrix factorization

Buza Antal^a, B. Kis Piroska^b

^aDunaújvárosi Főiskola
buza.antal@yahoo.com

^bDunaújvárosi Főiskola
pbkism@yahoo.com

Absztrakt: A mátrixok szorzattá bontását több területen alkalmazzák. Az egyik, viszonylag gyakran hivatkozott alkalmazás az ajánló rendszerekben a hiányzó értékek pontos, vagy legalább közelítő értékének meghatározása. Sokan ezt minden különösebb vizsgálat, vagy megkötés nélkül javasolják és néhány alkalmazási eredménnyel „igazolva látják” a módszer használhatóságát. Elméletileg is belátható, és néhány példával szemléltetjük is, hogy a mátrix faktorizáció ilyen megalapozatlan alkalmazása egymással lényegesen ellentmondásban levő következtetések, javaslatok „tudományos” alátámasztására szolgálhat, amelyek nyilván nem lehetnek mind helyesek. Megmutatjuk, hogy milyen további megfontolásokat kell keresni ahhoz, hogy a mátrix faktorizációs módszerrel kapott eredmények megnyugtatóan javasolhatók legyenek valamilyen gyakorlati alkalmazásban. Ezzel kapcsolatban felmerül a mátrix faktorizációra vonatkozó stabilitás vizsgálatának igénye is. Adunk erre vonatkozó definíciókat, és megmutatjuk, hogy a mátrix faktorizáció nem stabilis.

Kulcsszavak: adatbányászat, mátrix faktorizáció, ajánló rendszerek.

Abstract: Matrix factorization is used in a numerous areas. One of them is the recommender systems which are relatively often referred applications for finding of the exact missing values or at least giving approximate values. In a number of cases, application of matrix factorization is suggested without any consideration and restriction moreover some application results are presented „to prove” the usability of the method. It can be theoretically proved and shown by few examples that the unfounded application of the matrix factorization may lead to „scientific supporting” of essential opposing conclusions. However, each of the contrary conclusions is not true, of course. We show a few necessary further considerations taking into account at the application of the matrix factorization in order to the results of the method become reliably usable. Besides finding of the missing values other problem is the stability of the matrix factorization. Definitions are given referring to the stability and we show that the matrix factorization is not stable.

Keywords: datamining, matrix factorization, recommender system.

1. Bevezetés

Az ajánló rendszerek abban adnak segítséget, hogy minél megbízhatóbban el tudjuk dönteni azt, hogy bizonyos árukat (például filmeket, könyveket) szolgáltatásokat (mint például az üdülést, az éttermet) kinek érdemes ajánlani és kinek nem. Ha ezt a kérdést meg tudjuk válaszolni, az többek között célzott reklámok alkalmazását teszi lehetővé, ezáltal a reklámköltség csökken, a hatékonyság pedig javul. A javaslatot a korábban gyűjtött tapasztalatok alapján igyekeznek kidolgozni. A korábbi tapasztalatok kérdőívek, vagy a vásárlók viselkedésének megfigyelése segítségével gyűjthetők, és táblázatos alakban is rögzíthetők. Ha például filmekkel kapcsolatosan gyűjtik a nézők véleményét, akkor azt, hogy

hány pontot ad egy-egy néző az általa megtekintett filmre, összegezzük az 1. táblázathoz hasonló táblázatban.

1. táblázat. A nézők által az általuk látott filmekre adott tetszési pontszámok.

	<i>film1</i>	<i>film2</i>	<i>film3</i>	<i>film4</i>
<i>néző1</i>	16			
<i>néző2</i>		17	16	
<i>néző3</i>		15	10	
<i>néző4</i>				18
<i>néző5</i>		7	6	
⋮	⋮	⋮	⋮	⋮

Ennek a táblázatnak a peremek nélküli tartalma tekinthető egy hiányosan kitöltött mátrixnak. Az 1. táblázatnak megfelelő hiányos mátrix:

$$\begin{pmatrix} 16 & \cdot & \cdot & \cdot \\ \cdot & 17 & 16 & \cdot \\ \cdot & 15 & 10 & \cdot \\ \cdot & \cdot & \cdot & 18 \\ \cdot & 7 & 6 & \cdot \end{pmatrix}$$

A mátrix ismeretlen értékeit ponttal jelöltük.

Ha most például az a kérdés, hogy a *néző2* figyelmét érdemes-e a *film4*-re felhívni, akkor meg kellene állapítani, hogy – legalább közelítőleg – mekkora lenne majd a *néző2* által a *film4*-re adott tetszési pontszám. Előfordulhat, hogy a vizsgált következtetés levonásához a mátrix több értékére van szükségünk, például ha azt keressük, hogy valamely érték a sorában minimális, vagy maximális, akkor az érintett sor összes értékét meg kell határozni. A mátrixos ábrázolásra „lefordítva” a kérdés tehát egy vagy több jelenleg ismeretlen, hiányzó érték minél jobb közelítéssel való meghatározása.

Erre a feladatra többféle megoldási módszert javasolnak. Egy megközelítés szerint az ismert helyeken hasonló értékeket tartalmazó sorok alapján lehet megállapítani a kérdéses értéket. Ez a „*k*-legközelebbi szomszéd” módszer e feladatra történő adaptálása.

Több szerző szerint ennél jobb módszer a mátrix faktorizáció alkalmazása [4], [2]. Ez alatt azt értik, hogy keressünk két olyan mátrixot, amelyek szorzata az ismert helyeken pontosan, vagy legalább jó közelítéssel, a kiindulási mátrix ismert értékeit adja. Ekkor azt javasolják, hogy a kiindulási mátrix hiányzó értékeinek tekintsük a szorzat mátrix megfelelő helyein előállt értékeket. E módszert javasló „alaplakk” [1] semmilyen kikötést, vagy vizsgálatot nem javasol a kapott eredmények megbízhatóságának ellenőrzésére. A módszer alkalmazhatóságát mások is [2],[3], [7] alátámasztják néhány példával.

2. A mátrix faktorizáció nem egyértelmű

Nevezük faktor-mátrix párok azon mátrixokat, amelyek olyanok, hogy a szorzatuk a kiindulási mátrix ismert értékeit pontosan, vagy jó közelítéssel előállítják. Ha a faktor-

mátrixokra nagyon erős kikötést nem teszünk, akkor egy kiindulási mátrixhoz végtelen sok különböző faktor-mátrix pár fordulhat elő. A kérdéses következtetés levonására a faktor-mátrixok szorzat mátrixát használjuk. Ezek között előállnak olyanok, amelyek egymásnak teljesen ellentmondó következtetéseket támasztanak alá. Ilyenekre az [5]-ben példákat is mutatunk.

3. A mátrix faktorizáció nem stabilis

A mátrix faktorizáció módszerét a fentiek szerint a gyakorlatban alkalmazva nyilvánvalóan csak véges sok faktor-mátrix pár előállítására van mód. E faktor-mátrix párok szorzat mátrixai különbözőek lehetnek. Az ajánló rendszerek szempontjából a különbözőség egy határon belül elfogadható. Ha a szorzatmátrixok – legalább a vizsgálat szempontjából fontos helyeken – valamiféle stabilitást mutatnak, akkor a kapott eredmény használható lehet.

A stabilitásra vonatkozóan többféle definíciót is használhatunk. A [6] cikkben igyekeztünk összefoglalni a gyakorlati szempontból használható stabilitási megközelítéseket.

Értelmezhető a stabilitás a faktor-mátrix párok szorzatmátrixának egy-egy elemére vonatkozóan. Ha a vizsgált elem értékei szórása és átlaguk hányadosa egy küszöb alatt marad a különböző faktor-mátrix párok szorzatmátrixaiban, akkor a vizsgált elemet stabilisnak tekinthetjük.

Ha a kiindulási mátrix összes ismeretlen értékeire vonatkozó küszöbértékekre is adható elég kis felső korlát, akkor a kiindulási mátrixot a mátrix faktorizáció szempontjából stabilisnak tekinthetjük.

Mivel a kiindulási mátrix ismert értékei sem biztos, hogy teljesen megbízhatóak, hiszen más-más vevői körre történő felmérések eredményei egymástól különbözhetnek, így fontos lehet vizsgálni e különbözőségek hatását az eredményre.

Vizsgálhatjuk tehát a stabilitás kérdését úgy is, hogy a kiindulási mátrix ismert elemeiben történő változás mekkora változást okoz a faktor-mátrix párok szorzatmátrixaiban az eredetileg ismeretlen értékekre nézve.

A [6] cikkben megmutattuk, hogy az ajánló rendszerekben alkalmazott mátrix faktorizációs módszer a fent vázolt egyik stabilitási megközelítés szerint sem tekinthető stabilisnak.

4. Amikor alkalmazható a mátrix faktorizáció módszere

Kijelenthető tehát, hogy az ajánló rendszerekben a mátrix faktorizáció módszere a konkrét alkalmazási területek értelmezése nélkül felelősséggel nem használható.

Amennyiben mind a faktor-mátrixokat, mind azok szorzatát értelmezni tudjuk, akkor a módszer használható lehet.

Első példaként megemlíthetjük, hogy ha tenyészállatok napi tápanyag igényét tartalmazza a kiindulási mátrix, akkor az faktorizálható oly módon, hogy az egyik faktor-mátrix a különböző táplálékok tápanyag összetételét adja meg, a másik faktor-mátrix pedig azt tartalmazza, hogy melyik állat milyen táplálékból mennyit fogyasztott. Ekkor az, hogy egy

nap alatt egy-egy állat melyik tápanyagból mennyit fogyasztott össze a két faktormátrix szorzataként áll elő.

Megjegyezzük, hogy ez a feltételezés is valamilyen szintű egyszerűsítés, mert nem biztos, hogy egy-egy tápanyag a különböző táplálékokban bevive ugyanúgy hasznosul. A pontos összefüggés megállapításához a szakterület szakértőjével kell konzultálni.

Másik példa már bizonytalanabb. A bevezetőben is említett, a filmekre vonatkozó tetszési pontszámok esetleg(?) előállíthatók oly módon, hogy feltételezzük, hogy a filmek nézői aszerint értékelik a filmeket, hogy mennyire szeretik a filmben szereplő színészeket, az író, a rendezőt, az operatőrt és így tovább. Ekkor az egyik faktormátrix jellemezheti azt, hogy egy-egy néző hogyan értékeli a film alkotóit, a másik faktormátrix leírhatja azt, hogy egy-egy filmben melyik alkotónak mekkora szerepe volt. Így az össz tetszési vélemény kialakulhat úgy is, hogy nézőnként és filmenként összegezzük az alkotókra vonatkozó nézői vélemények és az alkotóknak a konkrét filmre való hatása szorzatai értékeit.

Az valójában kérdéses, hogy helyes-e az a feltevésünk, hogy a filmről alkotott nézői vélemény tényleg ilyen összetevőkből és ilyen módszerrel számítható-e ki. Ha igen, akkor a faktorizáció módszere megbízható eredményeket szolgáltathat.

Természetesen egyes példákban – amire szerzők hivatkoznak – előfordulhat, hogy a faktor-mátrixok és a szorzás értelmezése nélkül is jónak, a gyakorlatban használhatónak bizonyuló faktor-mátrixokat is találhatunk. Ez azonban egyrészt szerencse kérdése (az, hogy a végtelen sok faktorizáció közül valamely módszerrel éppen a „jó”-t találtuk meg), másrészt nincs rá kritériumunk, hogy megállapítsuk, hogy amit találtunk az esetleg éppen a „jó”-e. Megítélésünk szerint, ha értelmezés nélkül ugyan, de jónak bizonyuló faktorizációt találunk, akkor érdemes azt tovább vizsgálni, mert feltehetően értelmezhető, és meglehet, hogy eddig nem ismert összefüggés felismerésére vezethet.

5. Konklúzió

A mátrix faktorizáció módszerének az ajánló rendszerekben való alkalmazása a faktor-mátrixok és azok szorzásának értelmezése nélkül egymással teljesen ellentmondó következtetésekre vezethet. Ezeknek legfeljebb egyike lehet helyes, és a módszer semmilyen kritériummal nem szolgál arra, hogy eldöntsük azt, hogy – a különben végtelen sok lehetőség közül – az általunk előállított faktorizáció a konkrét alkalmazásban helyesnek tekinthető-e vagy sem.

Irodalomjegyzék

- [1] Y. Koren, R. Bell and C. Volinsky: Matrix factorization techniques for recommender systems, *IEEE, Computer*, (August 2009), 42-49.
- [2] P. Forbes and M. Zhu: Content-boosted Matrix Factorization for Recommender Systems: Experiments with Recipe Recommendation, *Rec-Sys'11*, (2011)

- [3] R. Karimi, C. Freudenthaler, A. Nanopulos and L. Schmidt-Thieme: Comparing Prediction Models for Active Learning in Recommender Systems, <http://www.ismll.uni-hildesheim.de/pub/pdfs/karimi-gfkl.pdf> (2010)
- [4] G. Takács, I. Pilászy, B. Németh and D. Tikk: Investigation of Various Matrix Factorization Methods for Large Recommender Systems, *KDD netfix weekshop*, (2008).
- [5] A. Buza and P.B. Kis: Pitfall of the matrix factorization for recommender systems, (under review). (2014)
- [6] A. Buza and P.B. Kis: Instability of matrix factorization used in recommender systems, *Annales Univ. Sci.Sect. Comp.* 42 (2014), (under print)

Membrán számítások határokkal

Membrane computations with boundaries

Csajbók Zoltán Ernő^a, Mihálydeák Tamás^b

^aDebreceni Egyetem Egészségügyi Kar
csajbok.zoltan@foh.unideb.hu

^bDebreceni Egyetem Informatikai Kar
mihalydeak.tamas@inf.unideb.hu

Absztrakt: A membrán rendszer vagy P -rendszer az élő sejtek felépítése és működése, valamint a biokémiai folyamatok által inspirált újelvű számítási modell. Egy ilyen rendszerben a hierarchikusan elrendezett régiókat membránok határolják. A régiók objektumokat tartalmaznak, megengedve az objektumok multiplicitását (multihalmaz). Az evolúciós szabályok a régiók tartalmát módosítják, míg a kommunikációs szabályok az objektumok régiók közötti mozgását írják elő. A membrán számítás a szabályok maximális párhuzamos, nondeterminisztikus végrehajtásával történik. Akkor ér véget, ha nincs több végrehajtható szabály. A természeti folyamatokban a kommunikációs szabályok által reprezentált események a membránok közelében történnek. Ez szükségessé teszi a „közel lenni a membránhoz” absztrakt fogalmának, röviden az absztrakt határfogalom megadását membrán rendszerekben. A klasszikus Pawlak-féle életlen halmazelmélet (rough set theory) központi fogalma a határ. Ezért először az életlen halmazok elméletét általánosítottuk multihalmazokra, majd ennek segítségével a membrán rendszerekben megadtuk a membránok körüli határsáv absztrakt, nem térszerű leírását. Cikkünkben röviden bemutatjuk modellünket, ismertetjük a témakörben elért legújabb eredményeinket.

Kulcsszavak: membrán rendszerek, multihalmazok, multihalmaz közelítő terek

Abstract: Membrane systems or P systems are inspired by the structure and working of living cells as well as biochemical processes. Membranes delimit regions which are arranged in a cell-like hierarchical structure. A set of rules is given for every region. These rules model reactions inside regions and passing objects through membranes. They act in nondeterministic and maximally parallel manner. Active cell components involved in real biological processes have to be close enough to a membrane in order to be able to pass through it. Pawlak's classical rough set theory gives a plausible opportunity to model boundary zones around cell-like formations. However, this theory works within conventional set theory, and so to apply its ideas to membrane computing, firstly, we worked out an adequate approximation framework for multisets. Next, we proposed a two-component structure consisting of a P system and an approximation space for multisets. These two components mutually cooperate. Using the approximation technique, we specified the closeness around membranes, even from inside and outside, via boundaries of regions in the sense of multiset approximations. Then, communication rules within the P system were defined in such a way that they operated in the boundary zones solely. Restricting communication rules to these boundaries, the interactions along the membranes can be controlled locally during the membrane computations.

Keywords: membrane systems, multisets, multiset approximation spaces

1. Bevezetés

A membrán számítások (membrane computing) a természet-motivált számítástudomány (natural computing) részterülete. A membrán rendszer vagy más néven P -rendszer fogalmát Gheorghe Păun vezette be a 2000-es évek legelején [1], [2], [3]. Olyan absztrakt számítási modell, amelynek kialakítását az élő sejtek felépítése és működése, továbbá a biokémiai folyamatok lezajlása motiválták.

Az alapmodellnek számtalan variánsa van. Ezek legtöbbje univerzális, azaz számítási ereje megegyezik a Turing-gépével. Végső soron tehát egy újjelvű, erős, ugyanakkor szemléletes absztrakt számítási modelltől van szó. A membrán számításokat az ISI már 2003-ban (!) a számítástudomány új részterületeként említette [20]. Az ISI Essential Science Indicators [1] szerint a tudományág fejlődését elindító [1] Păun cikkekre történő hivatkozás 2002 szeptember-októberében a leggyorsabb növekedést produkálta a számítástudományban. Ezért 2003 februárjában megkapta a „gyorsan terjedő cikk” („fast breaking papers”) minősítést.¹

Az alapmodellben az *objektumoknak* nevezett alkotóelemek *régiókban* helyezkednek el. A régióban egy objektumnak több *példánya* is jelen lehet az oldatokban úszó molekulák mintájára. Az ennek megfelelő matematikai struktúra a *multihalmaz*. A régiókat *membránok* határolják, elkülönítve azok belső tartalmát a közvetlen külvilágtól. A membránokkal körülzárt egységek – a sejtyszerű felépítésre emlékeztető módon – hierarchikus struktúrában helyezkednek el.

A legfelső szintű régiót a teljes membránrendszer *környezete* veszi körül. Mindegyik régióhoz tartozik egy *szabályhalmaz*. Kétféle típusú szabály van. Az *evolúciós szabályok* a belső objektumok kölcsönhatásait szabályozzák a biokémiai reakciókhoz hasonló módon. A *kommunikációs szabályok* a membránok áteresztő képességét modellálják, és a biológiai folyamatokhoz hasonlóan az objektumok régiók közötti mozgását írják elő. Az objektumok bármely régióból csak az ő szülőjébe juthatnak ki (ún. *out típusú symport szabály*), illetve egy régióba csak az ő szülőjéből juthatnak be (ún. *in típusú symport szabály*). Lehetőség van arra is, hogy a gyermek-szülő régiók között az objektumok egyidejűleg ki/be mozogjanak (ún. *antiport típusú szabály*).

A szabályok végrehajtása után általában megváltozik a régiók tartalma, a rendszer új *konfigurációja* jön létre. A membrán rendszerekben a számítás nem más, mint egy kezdő konfigurációból kiinduló konfiguráció-sorozat. A számítás leállítása, illetve eredményének megállapítása sokféleképpen lehetséges.

A szabályok végrehajtása *maximális párhuzamos* módon történik. Ez azt jelenti, hogy egy számítási lépés során minden végrehajtható szabályt végre is kell hajtani, ráadásul mindaddig, amíg lehetséges. Megkötés, hogy egy objektum példány csak egyetlen szabály végrehajtásban vehet részt. Természetesen egyidejűleg több szabály is végrehajtható, illetve az előbbi megkötés miatt a szabályok „versenyeznek” az objektum példányokért. Ekkor az alkalmazandó szabályok és objektum példányok kiválasztása véletlenszerű, következképpen a számítás *nemdeterminisztikus*.

A valós biológiai folyamatokban az objektumok membránokon áthaladó mozgása a membránok közelében történik. Számos elmélet létezik a közelség modellezésre. A [6] általános topológiai eszközöket alkalmaz. A [12] fuzzy elméletre épülő megoldást, a [13] a membránokhoz rendelt szabályokat javasol. A [4], [5] cikkek az objektumok régió belüli térbeli helyzetére építenek. Általában a „közelség”, „közelében lenni” fogalma azonban nem feltétlen térszerű. Kifejezheti például azt is, hogy két vegyület között igen erős a reakcióra való lépés készsége.

A Pawlak-féle életlen halmazelmélet (rough set theory, RST) [10], [11] kiváló lehetőség a halmazok körüli határ, így az *absztrakt közelségfogalom* megragadására. Az elmélet azonban a klasszikus halmazelmélet keretei között működik, ezért először az RST fogalmi rendszerét általánosítottuk multihalmazokra [7], [9], neve: multihalmaz approximációs tér. Ezt követően két komponensből álló struktúrát javasoltunk [7], [8]: membrán rendszer és a hozzákapcsolódó multihalmaz approximációs tér. Ebben a kettős struktúrában minden egyes régióknak mint multihalmaznak képezhető a határa.

¹ <http://www.esi-topics.com/fbp/fbp-february2003.html>

2. Multihalmazok

Legyen U egy véges nemüres halmaz, \mathbb{N} jelölje a természetes számok halmazát. Az $M : U \rightarrow \mathbb{N} \cup \{\infty\}$ leképezést U feletti *multihalmaznak* nevezzük.

Jelölje $\mathcal{MS}(U)$ az U feletti multihalmazok halmazát. Az M multihalmaz *véges*, ha $M(a) < \infty$ ($a \in U$). M *üres* multihalmaz, jelölése \emptyset , ha $M(a) = 0$ ($a \in U$).

Ha $M \neq \emptyset$ véges, akkor a $w_M = a_{k_1}^{M(a_{k_1})} a_{k_2}^{M(a_{k_2})} \dots a_{k_l}^{M(a_{k_l})}$ karaktersorozat permutációival reprezentálható; $M = \emptyset$ esetén $w = \lambda$, ahol λ az üres sztring. Megállapodás szerint a w_M bármely permutációja reprezentálhatja az M -t.

Véges multihalmazok halmazának neve *makrohalmaz*. A következőkben az alábbi makrohalmazokra lesz szükség:

- $\mathcal{MS}^n(U) = \{M \in \mathcal{MS}(U) \mid M(a) \leq n, a \in U\}$ ($n \in \mathbb{N}$),
- $\mathcal{MS}^{<\infty}(U) = \bigcup_{n=0}^{\infty} \mathcal{MS}^n(U)$.

Legyen $M, M_1, M_2 \in \mathcal{MS}(U)$ és $\mathcal{M} \subseteq \mathcal{MS}(U)$. A multihalmazok közötti relációk és műveletek a következőképpen értelmezhetők.

Multihalmazok relációja:

- *egyenlőség* reláció: $M_1 = M_2$, ha $M_1(a) = M_2(a)$ ($a \in U$);
- *tartalmazási* reláció: $M_1 \sqsubseteq M_2$, ha $M_1(a) \leq M_2(a)$ ($a \in U$).

Multihalmazok közötti műveletek:

- $M_1 \sqcap M_2$ *metszet*: $(M_1 \sqcap M_2)(a) = \min\{M_1(a), M_2(a)\}$ ($a \in U$);
- $\bigcap \mathcal{M}$ *metszet*: $(\bigcap \mathcal{M})(a) = \min\{M(a) \mid M \in \mathcal{M}\}$ ($a \in U$);
- $M_1 \sqcup M_2$ *halmaztípusú unió*: $(M_1 \sqcup M_2)(a) = \max\{M_1(a), M_2(a)\}$ ($a \in U$);
- $\bigcup \mathcal{M}$ *halmaztípusú unió*: $(\bigcup \mathcal{M})(a) = \sup\{M(a) \mid M \in \mathcal{M}\}$ ($a \in U$);
definíció szerint $\bigcup \emptyset = \emptyset$;
- $M_1 \oplus M_2$ *összeadás*: $(M_1 \oplus M_2)(a) = M_1(a) + M_2(a)$ ($a \in U$);
- $M_1 \ominus M_2$ *kivonás*: $(M_1 \ominus M_2)(a) = \max\{M_1(a) - M_2(a), 0\}$ ($a \in U$);
- n -*szeres összeadás* ($\oplus_n, n \in \mathbb{N}$) a következő induktív definícióval adható meg:
 - $\oplus_0 M = \emptyset, \oplus_1 M = M,$
 - $\oplus_{n+1} M = \oplus_n M \oplus M.$

Az n -szeres összeadás segítségével n -szeres tartalmazási reláció ($\sqsubseteq^n, n \in \mathbb{N}$) definiálható. Ehhez legyen $M_1 (\neq \emptyset), M_2 \in \mathcal{MS}(U)$. Ekkor $M_1 \sqsubseteq^n M_2$, ha $\oplus_n M_1 \sqsubseteq M_2$, de $\oplus_{n+1} M_1 \not\sqsubseteq M_2$.

3. Multihalmaz közelítő tér

3.1 Általános multihalmaz közelítő tér

A multihalmaz közelítő térnek öt alapvető komponense van:

- *ábécé*: absztrakt objektumok véges halmaza;
- *tartomány*: az ábécé feletti multihalmazok nemüres halmaza;
- *alaprendszer*: a tartomány előzetesen elkülönített multihalmazainak halmaza, amely a közelítések bázisául szolgál; elemei az *alap-multihalmazok*;
- *definiálható multihalmazok*: az alap-multihalmazokból származtatott multihalmazok, az alsó- és felső közelítések, valamint a határ „jelöltjei”;
- *közelítő pár*: az alsó- és felső közelítéseket határozzák meg.

Formálisan, a $MAS(U) = \langle \mathcal{MS}^{<\infty}(U), \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, l, u \rangle$ 5-ös U feletti (általános) multihalmaz közelítő tér az $\mathcal{MS}^{<\infty}(U)$ tartománnyal, ha

1. $\mathfrak{B} \subseteq \mathcal{MS}^{<\infty}(U)$, és ha $B \in \mathfrak{B}$, akkor $B \neq \emptyset$.
 \mathfrak{B} az *alaprendszer*, elemei az *alap-multihalmazok*.
2. $\mathfrak{D}_{\mathfrak{B}} \subseteq \mathcal{MS}^{<\infty}(U)$ a \mathfrak{B} kiterjesztése a következő minimális feltétellel: ha $B \in \mathfrak{B}$, akkor $\oplus_n B \in \mathfrak{D}_{\mathfrak{B}}$ ($n \in \mathbb{N}$).
 $\mathfrak{D}_{\mathfrak{B}}$ elemei a *definiálható multihalmazok*.
3. Az $l, u : \mathcal{MS}^{<\infty}(U) \rightarrow \mathcal{MS}^{<\infty}(U)$ függvények $\langle l, u \rangle$ *gyenge közelítő párt* alkotnak, ha
(C0) l és u *definiálható*: $l(\mathcal{MS}^{<\infty}(U)), u(\mathcal{MS}^{<\infty}(U)) \subseteq \mathfrak{D}_{\mathfrak{B}}$;²
(C1) l és u *monoton*: ha $S_1 \subseteq S_2$ ($S_1, S_2 \in 2^U$), akkor $l(S_1) \subseteq l(S_2)$, $u(S_1) \subseteq u(S_2)$;
(C2) u *normalizált*: $u(\emptyset) = \emptyset$;
(C3) *teljesül a gyenge közelítő tulajdonság*: ha $M \in \mathcal{MS}^{<\infty}(U)$, akkor $l(M) \sqsubseteq u(M)$.

Az l az *alsó-*, az u a *felső közelítő függvény*.

A definíciók alapján azonnal belátható, hogy az l is *normalizált*: $l(\emptyset) = \emptyset$.

Az $MAS(U)$ multihalmaz közelítő tér *totális*, ha bármely $M \in \mathcal{MS}^{<\infty}(U)$ esetén, létezik olyan $D \in \mathfrak{D}_{\mathfrak{B}}$ definiálható multihalmaz, hogy $M \sqsubseteq D$; egyébként a tér *parciális*.

Ésszerű feltételezés, hogy az alap-multihalmazok és n -szeres összegük alulról pontosan közelíthetők. Bizonyos esetekben ez a kitétel elvárható a definiálható multihalmazoktól is. Ezeket a követelményeket fogalmazza meg formálisan is a következő két feltétel.

Az $\langle l, u \rangle$ gyenge approximációs pár

- (C4) *granulált*, ha minden $B \in \mathfrak{B}$ alap-multihalmaz esetén $l(\oplus_n B) = \oplus_n B$ ($n \in \mathbb{N}$);
- (C5) *standard*, ha minden $D \in \mathfrak{D}_{\mathfrak{B}}$ definiálható multihalmaz esetén $l(D) = D$.

Fontos kérdés, hogy az alsó/felső közelítések hogyan viszonyulnak magához a közelítendő halmazhoz. Három eset különböztetendő meg az alábbiak szerint.

Az $\langle l, u \rangle$ gyenge approximációs pár

- (C6) *alulról erős*, ha $l(M) \sqsubseteq M$ ($M \in \mathcal{MS}^{<\infty}(U)$), azaz az l függvény *kontraktív (összehúzó)*;
- (C7) *felülről erős*, ha $M \sqsubseteq u(M)$ ($M \in \mathcal{MS}^{<\infty}(U)$), azaz az u függvény *extenzív*;
- (C8) *erős*, ha egyidejűleg alulról és felülről is erős, azaz minden $M \in \mathcal{MS}^{<\infty}(U)$ multihalmazt alulról $l(M)$, felülről $u(M)$ korlátozza: $l(M) \sqsubseteq M \sqsubseteq u(M)$.

A (C3)–(C8) feltételek alapján az $MAS(U)$ multihalmaz approximációs tér *gyenge/granulált/standard/alulról erős/felülről erős/erős*, ha az $\langle l, u \rangle$ approximációs pár *gyenge/granulált/standard/alulról erős/felülről erős/erős*.

3.2 Pawlak-féle multihalmaz approximációs tér

Konkrét számításokhoz szükséges a definiálható multihalmazok valamint az alsó/felső közelítések közelebbi meghatározása. Kézenfekvő ezeket a fogalmakat a Pawlak-féle elmélet mentén általánosítani. A kiindulópont a definiálható multihalmazok fogalmának szűkítése.

A $MAS(U)$ multihalmaz közelítő tér *szigorúan halmazunió típusú*, ha a definiálható multihalmazok $\mathfrak{D}_{\mathfrak{B}}$ halmaza a következő induktív definícióval adott:

1. $\emptyset \in \mathfrak{D}_{\mathfrak{B}}$,
2. $\mathfrak{B} \subseteq \mathfrak{D}_{\mathfrak{B}}$;
3. ha $\mathfrak{B}^{\oplus} = \{\oplus_n B \mid B \in \mathfrak{B}, n = 1, 2, \dots\}$ és $\mathfrak{B}' \subseteq \mathfrak{B}^{\oplus}$, akkor $\bigsqcup \mathfrak{B}' \in \mathfrak{D}_{\mathfrak{B}}$.

² $l(\mathcal{MS}^{<\infty}(U))$ és $u(\mathcal{MS}^{<\infty}(U))$ az l , illetve az u függvények értékkészletét jelölik.

A következő lépés az közelítő pár pontosítása.

Legyen $MAS(U) = \langle \mathcal{MS}^{<\infty}(U), \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, l, u \rangle$ szigorúan halmazunió típusú multihalmaz közelítő tér. Az $l, u : \mathcal{MS}^{<\infty}(U) \rightarrow \mathcal{MS}^{<\infty}(U)$ függvények *Pawlak-féle multihalmaz közelítő párt* alkotnak, ha minden $M \in \mathcal{MS}^{<\infty}(U)$ esetén

1. $l(M) = \bigsqcup \{ \oplus_n B \mid n \in \mathbb{N}^+, B \in \mathfrak{B} \text{ és } B \sqsubseteq^n M \};$
2. $b(M) = \bigsqcup \{ \oplus_n B \mid B \in \mathfrak{B}, B \not\sqsubseteq M, B \cap M \neq \emptyset \text{ és } B \cap M \sqsubseteq^n M \};$
3. $u(M) = l(M) \sqcup b(M).$

$A b : \mathcal{MS}^{<\infty}(U) \rightarrow \mathcal{MS}^{<\infty}(U)$ függvény a *Pawlak-féle határ*.

A definíciók alapján közvetlenül belátható, hogy az előzőekben megadott definiálható halmazokkal és Pawlak-féle multihalmaz közelítő párral, az $MAS(U)$ eleget tesz a (C0)–(C3), (C4) és (C6) feltételeknek, más szavakkal alulról erős, granulált tér.

A szigorúan halmazunió típusú, Pawlak-féle közelítő párral rendelkező tér neve *Pawlak-féle multihalmaz közelítő tér*.

4. Kommunikációs membrán rendszerek

Egy m -edfokú μ membrán struktúra olyan m -edfokú ($m \geq 1$) gyökeres fa, amelynek csúcsait az $1, \dots, m$ egészek azonosítják. A μ membrán struktúra az $R_\mu \subseteq \{1, \dots, m\} \times \{1, \dots, m\}$ halmazzal reprezentálható. Az $\langle i, j \rangle \in R_\mu$ pár jelentése, hogy a μ gráfban az i -edik (szülő) csúcsból a j -edik (gyerek) csúcsba él vezet. Formálisan: $\text{parent}(j) = i$.

Legyen μ egy m -edfokú membrán struktúra, V pedig egy véges ábécé. A

$$\Pi = \langle V, \mu, w_1, w_2, \dots, w_m, R_1, R_2, \dots, R_m \rangle$$

sorozat *kommunikációs P-rendszer*, ha

1. $w_i \in \mathcal{MS}^{<\infty}(V)$ ($i = 1, 2, \dots, m$);
2. R_i ($i = 1, 2, \dots, m$) szabályok véges halmaza, ahol valamennyi $r \in R_i$ szabály a következő formájú lehet:
 - a. $\langle u, in \rangle$ vagy $\langle u, out \rangle$, ahol $u (\neq \lambda) \in \mathcal{MS}^{<\infty}(V)$ (*symport* típusú szabály);
 - b. $\langle u, in; v, out \rangle$, ahol $u (\neq \lambda), v (\neq \lambda) \in \mathcal{MS}^{<\infty}(V)$ (*antiport* típusú szabály).

Adott $\Pi = \langle V, \mu, w_1, w_2, \dots, w_m, R_1, R_2, \dots, R_m \rangle$ P-rendszer esetén, legyen

$$MAS(\Pi) = \langle \mathcal{MS}^{<\infty}(V), \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, l, u \rangle$$

egy szigorúan halmazunió típusú multihalmaz közelítő tér Pawlak-féle $\langle l, u \rangle$ közelítő párral. A $MAS(\Pi)$ neve *kapcsolt membrán közelítő tér*.

5. Kommunikációs szabályok korlátozása a membrán határokra

Miután adott a Π membrán rendszer és a kapcsolt $MAS(\Pi)$ közelítő tere, a $MAS(\Pi)$ térben meghatározhatók a w_1, w_2, \dots, w_m régiók mint multihalmazok Pawlak-féle határa. A $b(w_1)$ teljes egészében a membrán rendszer környezetébe illeszkedik. A $b(w_i)$ ($i = 2, \dots, m$) határok azonban nem feltétlen követik a membrán struktúrát, mert a w_2, \dots, w_m régiók határa „kilóghat” a szülő régióból. Ezért a Pawlak-féle határokat az alábbi módon hozzá kell igazítani a membrán struktúrához.

Tegyük fel, hogy adott a $\Pi = \langle V, \mu, w_1, w_2, \dots, w_m, R_1, R_2, \dots, R_m \rangle$ P -rendszer és a kapcsolt $\text{MAS}(\Pi) = \langle \text{MS}^{<\infty}(V), \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, l, u \rangle$ membrán közelítő tere. Ekkor tetszőleges $B \in \mathfrak{B}$ és $i = 1, 2, \dots, m$ esetén legyen:

$$N(B, i) = \begin{cases} 0, & \text{if } B \sqsubseteq w_i \text{ or } B \sqcap w_i = \emptyset; \\ n, & \text{if } i = 1 \text{ and } B \sqcap w_1 \sqsubseteq^n w_1; \\ \min\{k, n \mid B \sqcap w_i \sqsubseteq^k w_i, \text{ and } B \ominus w_i \sqsubseteq^n w_{\text{parent}(i)}\}, & \text{egyébként.} \end{cases}$$

Ezt követően a határok legyenek ($i = 1, \dots, m$):

$$\text{bnd}(w_i) = \bigsqcup \{ \oplus_{N(B, i)} B \mid B \in \mathfrak{B} \};$$

$$\text{bnd}^{\text{out}}(w_i) = \text{bnd}(w_i) \ominus w_i;$$

$$\text{bnd}^{\text{in}}(w_i) = \text{bnd}(w_i) \ominus \text{bnd}^{\text{out}}(w_i).$$

A bnd függvény megadja a régió határát. A $\text{bnd}(w_i)$ ($i = 1, \dots, m$) határok definiálható multihalmazok. Valamennyi $\text{bnd}(w_i)$ két részből áll, a $\text{bnd}^{\text{out}}(w_i)$ a régió *külső*-, a $\text{bnd}^{\text{in}}(w_i)$ a régió *belső határa*. Ezek nem feltételen definiálható multihalmazok.

A definíciókból azonnal következik, hogy $\text{b}(w_1) = \text{bnd}(w_1)$, de általában $\text{b}(w_i) \neq \text{bnd}(w_i)$ ($i = 2, \dots, m$).

A régió határok megadása után a kommunikációs szabályok végrehajtása a membránok közelére korlátozható. Formálisan, az i -edik ($i = 1, \dots, m$) régió $r \in R_i$ szabálya csak akkor hajtható végre, ha

- $u \sqsubseteq \text{bnd}^{\text{out}}(w_i)$, feltéve, hogy az r in típusú $\langle u, \text{in} \rangle$ symport szabály;
- $u \sqsubseteq \text{bnd}^{\text{in}}(w_i)$, feltéve, hogy az r out típusú $\langle u, \text{out} \rangle$ symport szabály;
- $u \sqsubseteq \text{bnd}^{\text{out}}(w_i), v \sqsubseteq \text{bnd}^{\text{in}}(w_i)$, feltéve, hogy az r antiport típusú $\langle u, \text{in}; v, \text{out} \rangle$ szabály.

Bizonyítható, hogy ezekkel a korlátozásokkal a membrán számítások ténylegesen végrehajthatók ([7], Theorem 1).

6. A maximális párhuzamosság kontrollálása

A membrán rendszerek maximális párhuzamosságának kontrollálására számos kísérlet történt:

- minimális vagy korlátozott párhuzamosság [14], [15];
- prioritások, preferenciák hozzárendelése a szabályokhoz [16];
- metabolikus P -rendszerek [17], [18];
- általános topológiai eszközök [6] stb.

A Π membrán rendszer és a kapcsolt $\text{MAS}(\Pi)$ membrán közelítő tér által alkotott kétkomponensű rendszerben a kommunikációs szabályok végrehajtása a membránok körüli határsávra korlátozható. Kérdés, hogy ez miképpen hat a maximális párhuzamosságra.

A [19] cikkben bemutattuk, hogy egyrészt ez a korlátozás nem szünteti meg a membrán rendszerek működésének két fontos alapelvét, a maximális párhuzamosságot és a nemdeterminisztikusságát. Ugyanakkor kimutattuk azt is, hogy ekkor a maximális párhuzamosság bizonyos mértékig kontrollálható.

7. Köszönetnyilvánítás

A publikációt támogatta a TÁMOP–4.2.2.C–11/1/KONV–2012–0001 projekt. A projektet támogatta az Európai Unió, társfinanszírozta az Európai Szociális Alap.

Irodalomjegyzék

- [1] Păun, Gh.: Computing with membranes. *Journal of Computer and System Sciences* **61**(1), 108–143 (2000)
- [2] Păun, Gh.: Membrane Computing. An Introduction. Springer, Berlin (2002)
- [3] Păun, Gh., Rozenberg, G., Salomaa, A. (eds.): The Oxford Handbook of Membrane Computing. Oxford Handbooks. Oxford University Press, Inc., New York (2011)
- [4] Barbuti, R., Maggiolo-Schettini, A., Milazzo, P., Pardini, G., Tesei, L.: Spatial P systems. *Natural Computing* **10**(1), 3–16 (2011)
- [5] Cardelli, L., Gardner, P.: Processes in Space. In: Ferreira, F., Löwe, B., Mayordomo, E., Mendes Gomes, L. (eds.) CiE 2010. LNCS, vol. 6158, pp. 78–87. Springer, Heidelberg (2010)
- [6] Csuhaj-Varjú, E., Gheorghe, M., Stannett, M.: P Systems Controlled by General Topologies. In: Durand-Lose, J., Jonoska, N. (eds.) UCNC 2012. LNCS, vol. 7445, pp. 70–81. Springer, Heidelberg (2012)
- [7] Mihálydeák, T., Csajbók, Z.E.: Membranes with boundaries. In: Csuhaj-Varjú, E., Gheorghe, M., Rozenberg, G., Salomaa, A., Vaszil, G. (eds.) CMC 2012. LNCS, vol. 7762, pp. 277–294. Springer, Heidelberg (2013)
- [8] Mihálydeák, T., Csajbók, Z.E.: Partial approximation of multisets and its applications in membrane computing. In: Lingras, P., Wolski, M., Cornelis, C., Mitra, S., Wasilewski, P. (eds.) RSKT 2013. LNCS, vol. 8171, pp. 99–108. Springer, Heidelberg (2013)
- [9] Mihálydeák, T., Csajbók, Z.E., Takács, P.: Communication rules controlled by generated membrane boundaries. In: Alhazov, A., Cojocaru, S., Gheorghe, M., Rogozhin, Y., Salomaa, A. (eds.) Membrane Computing, 14th International Conference, CMC 2013, Chişinău, Republic of Moldova, August 20-23, 2013, Revised Selected Papers. LNCS, vol. 8340, pp. 265–279. Springer, Berlin Heidelberg (2014)
- [10] Pawlak, Z.: Rough sets. *International Journal of Computer and Information Sciences* **11**(5), 341–356 (1982)
- [11] Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
- [12] Aguzzoli, S., Besozzi, D., Gerla, B., Manara, C.: P systems with vague boundaries: the t-norm approach. In: Proceedings of Brainstorming Workshop Uncertainty in Membrane Computing, 8-10 November 2004, Palma de Mallorca, Spain, pp. 97–105.
- [13] Bernardini, F., Manca, V.: P Systems with Boundary Rules. In: Păun, Gh., Rozenberg, G., Salomaa, A., Zandron, C. (eds.): Membrane Computing. International Workshop, WMC-CdeA 02, Curtea de Arges, Romania, August 19-23, 2002. Revised Papers, LNCS, vol. 2597, pp. 107–118. Springer-Verlag, Berlin, Heidelberg, New York (2003)
- [14] Ciobanu, G., Pan, L., Păun, Gh., Pérez-Jiménez, M.J.: P systems with minimal parallelism. *Theoretical Computer Science* **378**(1), 117–130 (2007)

- [15] Ibarra, O.H., Yen, H.C., Dang, Z.: On various notions of parallelism in P systems. *Int. J. Found. Comput. Sci.* **16**(4), 683–705 (2005)
- [16] Raman, A., Kim, H., Oh, T., Lee, J.W., August, D.I.: Parallelism orchestration using dope: the degree of parallelism executive. In: Hall, M.W., Padua, D.A. (eds.) Proceedings of the 32nd ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2011, San Jose, CA, USA, June 4-8, pp. 26–37. ACM (2011)
- [17] Manca, V.: Fundamentals of metabolic P systems. In: Păun et al., [3], pp. 475–498
- [18] Manca, V., Bianco, L., Fontana, F.: Evolution and oscillation in P systems: Applications to biological phenomena. In: Mauri, G., Păun, Gh., Pérez-Jiménez, J.M., Rozenberg, G., Salomaa, A. (eds.) WMC 2004. LNCS, vol. 3365, 63–84. Springer, Heidelberg (2005)
- [19] Csajbók, Z.E., Mihálydeák, T.: Maximal Parallelism in Membrane Systems with Generated Membrane Boundaries. In: A. Beckmann and E. Csuhaj-Varjú and K. Meer (Eds.): Language, Life, Limits. 10th Conference on Computability in Europe, CiE 2014, Budapest, Hungary, June 23-27, 2014. Proceedings, LNCS, vol. 8493, 103–112. Springer International Publishing, Switzerland (2014)
- [20] Păun, Gh., Pérez-Jiménez, M.J., Ștefănescu, Gh.: Membrane computing and programming. *The Journal of Logic and Algebraic Programming* **79**(2010) 289–290.

Néhány megjegyzés a polinomszerű Boole-függvényekhez

Some remarks to the polynomial-like Boolean functions

Gonda János

ELTE IK

andog@inf.elte.hu

Absztrakt: A logikai függvények egy véges halmazzal önmagába képező függvények, speciális esetként, ha a halmaz elemeinek száma 2, akkor ezek a függvények a Boole-függvények. Egy Boole-függvény több különböző alakban is megadható, közülük a leggyakoribbak a diszjunktív illetve a konjunktív normálalakok, ezek között is elsősorban a kanonikus alakok, hiszen ezek lényegében véve azonosak a függvénytáblával. Ezen formák mellett gyakori a függvény polinommal való reprezentálása, a Zsegalkin-polinommal való leírás. Mind a kanonikus alakok, mind a Zsegalkin-polinom egy 2^n -hosszúságú 0 - 1 sorozattal adható meg, amely tekinthető a kételemű test fölötti 2^n -dimenziós tér elemének. Ebben a leírásban az áttérés a kanonikus alakról a polinomosra, illetve vissza, megadható egy lineáris transzformációval. A transzformációnak van sajátértéke és a sajátértékhez tartozó sajátaltére. Az altérbe tartozó függvények az úgynevezett polinomszerű illetve konjunktívan polinomszerű Boole-függvények, amelyek generálják a Boole-függvények teljes halmazát.

Amellett, hogy függvények egy jól körülírható osztályának tanulmányozása, tulajdonságainak minél szélesebb körű feltárása önmagában is érdekes és fontos matematikai feladat, adott esetben az eredmények gyakorlati alkalmazása is fontos lehet. A polinomszerű Boole-függvények által megvalósított logikai hálózatok jellemzője, hogy az alapvető ÉS- és VAGY-kapukkal való megvalósítás topológiája azonos az ugyanezen függvényt realizáló, az ÉS- és KIZÁRÓ VAGY kapukkal való felépítés elrendezésével.

A transzformációt és annak tulajdonságait, valamint a polinomszerű és a konjunktívan polinomszerű Boole-függvényeket és számos jellemzőjüket több cikkben is leírták. A konferencián elhangzó előadásban az ilyen függvények néhány további tulajdonságáról lesz szó.

Kulcsszavak: Boole-függvény, Zsegalkin-polinom, polinomszerű Boole-függvény, konjunktívan polinomszerű Boole-függvény

Abstract: Boolean functions are the functions mapping a finite set into itself. As a special case, if the number of the elements of the set is equal to 2, then the functions are the Boolean-functions. A Boolean function can be defined in several ways. The most frequent forms are the disjunctive and conjunctive normal forms, especially the canonical ones, as these forms are essentially the same as the function tables. Beside these forms it is frequent to represent a Boolean function by a polynomial, to describe the function by the so-called Zhegalkin-polynomial. As the canonical forms, as the Zhegalkin polynomial can be given by a 2^n -long 0 - 1 sequence, which can be considered as an element of the 2^n -dimensional linear space over the finite field of 2 elements. In this forms of the description of the functions, the transition from the canonical form to the polynomial one and in the other direction, respectively, can be given by a linear transform. The transform has an eigenvalue and an eigensubspace belonging to the eigenvalue of the transform. The functions, lying in that subspace are the so-called polynomial-like and the conjunctively polynomial-like Boolean functions. The functions of the subspaces generate the whole set of the Boolean functions.

The study of a well-defined class of the functions, the widest possible range of the exploration of its properties in itself is an interesting and important mathematical task, but, in addition, the appropriate practical application of the results is also important. An important feature of the logical networks, realized by polynomial-like or conjunctively polynomial-like Boolean functions, respectively, is that the topology of the realisation by the basic AND-, OR- and NOT-gates is identical to the layout of the same function, built by AND- and XOR-gates.

The abovementioned transform and its properties, and the polynomial-like and conjunctively polynomial-like Boolean functions, as well as their traits can be found in several articles. In this paper some further properties of such functions are described.

Keywords: Boolean functions, Zhegalkin polynomial, polynomial-like Boolean function, conjunctively polynomial-like Boolean function

1. Bevezetés

A Boole-függvények mind az elmélet, mind a gyakorlat számára fontos matematikai objektumok. A számítástechnika egyik alapját, az irányítástechnika, a kriptográfia fontos kellékét jelentik ezek a függvények és informatikai, műszaki megvalósításuk, így tanulmányozásuk, tulajdonságaik vizsgálata, bizonyos tulajdonságokkal rendelkező osztályaik körülhatárolása és speciális vonásaik feltárása, alkalmazásuk körének meghatározása alapvető. Ezek a vizsgálatok már a 19. század első felében megkezdődtek, majd a 20. században az alkalmazások mind szélesebb tárházának létrejöttével erőteljesebben folytatódtak. Bár matematikailag egy elég kis területet ölel fel ez a téma, mégis, az önmagában való érdekessége, és még inkább felhasználásának rendkívüli fontossága miatt kutatásuk nem lankad, újabb és újabb tulajdonságai kerülnek felszínre, amelyek ismét újabb vizsgálati lehetőségeket indukálnak, sőt, kényszerítene ki.

A Boole-függvények véges halmazon, közelebbről egy kételemű halmazon értelmezett, néhány változós függvények (ez jelenthet egy-két változót, de akár több százat is), amelyek az adott véges halmazba képeznek, így az egyes függvények értelmezési tartománya véges, következésképpen a függvény megadható a rendezett párok felsorolásával, például egy táblázatban való felírásával. Egy más lehetőség bizonyos kifejezéseket megadni (az őket definiáló tulajdonságokkal együtt), majd kiértékeléssel ezekhez függvényeket rendelni. Műveletek egy halmazát funkcionálisan teljesnek mondunk, ha segítségükkel valamennyi függvény kifejezhető, és egy ilyen funkcionálisan teljes halmaz bázis, ha minimális ezzel a tulajdonságával. Boole-függvények esetén funkcionálisan teljes halmazt alkot az ÉS-, VAGY- és NEM-művelet, másként, az előbbi sorrendben, a konjunkció, a diszjunkció illetve a tagadás. Bázist alkot például az ÉS- és a NEM-művelet, vagy a VAGY- és a NEM-művelet, de önmagában funkcionálisan teljes a NEM-ÉS művelet (NAND, Sheffer-művelet, Sheffer-vonás), vagy a NEM-VAGY művelet (NOR, Peirce-művelet, Peirce-vonás), és ezek nyilvánvalóan egyben bázisai is a Boole-függvények halmazának. Az ÉS-, VAGY- és NEM-műveletek segítségével épül fel a függvény egy diszjunktív illetve egy konjunktív normálalakja, azaz olyan, az adott függvényt megvalósító kifejezés, amelyben a változók illetve ellentettjeik bizonyos ÉS-kapcsolatainak VAGY-kapcsolata, illetve fordítva, VAGY-kapcsolatainak ÉS-kapcsolata írja le a függvényt. Egy függvénynek több különböző diszjunktív illetve konjunktív normálalakja lehet, ám ezek között van egy-egy, a kanonikus diszjunktív normálalak és a kanonikus konjunktív normálalak, amely már egyértelműen meghatározott. Ebben a formában a diszjunkció illetve a konjunkció minden tagja valamennyi változót tartalmazza vagy eredeti, vagy tagadott alakjában.

A függvény változóinak sorrendjét rögzítve, és n változó esetén 0-tól $n-1$ -ig indexelve, továbbá egy tagadott változóhoz 0-t, egy nem tagadotthoz 1-et rendelve minden mintermhez, azaz minden olyan konjunkcióhoz, amely az összes változót tartalmazza vagy tagadott, vagy nem tagadott formában, tartozik egy egyértelműen meghatározott, 2^n -nél kisebb nemnegatív egész szám, és ezen számok megadása is meghatározza az adott függvényt. Ám ezeket a számokat tekinthetjük a kételemű test fölötti n -dimenziós térben egy vektor azon komponensei indexének, amely komponensek értéke 0. Ekkor a függvényt egy 2^n -hosszúságú 0 – 1 sorozattal adjuk meg. Ez a sorozat is egyértelműen jellemzi az adott függvényt. Magát a sorozatot a

függvény kanonikus diszjunktív normálalakjához tartozó spektrumának mondjuk. Ehhez teljesen hasonlóan adható meg a konjunktív normálalakhoz tartozó spektrum is.

Egy másik, funkcionálisan teljes halmaz, és egyben bázis, a XOR- és ÉS-művelet (az előbbi a KIZÁRÓ VAGY, másként az EXCLUSIVE OR), kiegészítve a konstans 1-művelettel. Ezekkel a műveletekkel is megadható egy normálalak, amely már egyértelmű. Ebben a normálalakban tagadás nélküli változók konjunkciójának, valamint esetleg a konstans 1-nek a kizáró vagy kapcsolata adja a függvényt. Ezt az alakot szokás a függvény Zsegalkin-polinomjának is nevezni. A változók rögzített sorrendje és ennek megfelelő indexelése mellett most is egy 2^n -hosszúságú $0 - 1$ sorozattal adható meg a függvény, a függvény Zsegalkin-polinomjához tartozó spektrumával.

Adott függvény kanonikus diszjunktív normálalakja és Zsegalkin-polinomja spektrumai között a kapcsolatot, azaz az áttérést egyik spektrumról a másikra egy, a kételemű test fölötti 2^n -dimenziós tér lineáris transzformációja adja meg. A transzformáció mátrixa igen egyszerű szerkezetű:

$$\mathbf{A}^{(0)} = \mathbf{1}$$

$$\mathbf{A}^{(n+1)} = \begin{pmatrix} \mathbf{A}^{(n)} & \mathbf{0}^{(n)} \\ \mathbf{A}^{(n)} & \mathbf{A}^{(n)} \end{pmatrix},$$

ahol n a függvény változóinak száma. A fentiekből következik, hogy $\mathbf{A}^{(n)}$ egy 2^n -edrendű kvadratikus mátrix, és könnyen adódik, hogy a mátrix inverze önmaga ([3], [5]).

2. Második fejezet

A Boole-függvények egy speciális csoportját képezik azok, amelyekre teljesül, hogy a kanonikus diszjunktív normálalakjához és Zsegalkin-polinomjához tartozó spektruma azonos, azaz $\underline{k} = \mathbf{A}^{(n)} \underline{\alpha}$, ahol $\underline{\alpha}$ és \underline{k} az n -változós Boole-függvény két spektruma a fentebbi sorrendben. Ezek a Boole-függvények a polinomszerű Boole-függvények. Az n -változós polinomszerű Boole-függvények egy 2^{n-1} -dimenziós alteret képeznek az összes n -változós Boole-függvény 2^n -dimenziós terében, feltéve, hogy $n > 0$.

Boole-függvények egyértelműen leírhatóak a kanonikus diszjunktív normálalakjukkal is, és ehhez az alakhoz ugyanaz a spektrum tartozik, mint a diszjunktív formához. A kanonikus normálalak

$$f = \prod_{i=0}^{2^n-1} (\alpha_i + M_i),$$

ahol $M_i = \sum_{j=0}^{n-1} (a_j^{(i)} \oplus x_j)$ az i -edik maxterm (az összeadás a logikai VAGY-ot, a szorzás a logikai ÉS-t, míg \oplus a KIZÁRÓ VAGY-ot jelöli). A kifejezésekben α_i a függvénynek a normálalakhoz tartozó spektruma, míg $a_j^{(i)}$ az i bináris felírásának j -edik jegye. Definiáltuk az úgynevezett módosított maxtermeket: $M_i' = \sum_{j=0}^{n-1} (\overline{a_j^{(i)}} \oplus x_j)$, ahol a felülhúzás a negálás, a ta-

adás jele. Könnyen belátható, hogy $M_i = M_{2^{n-1-i}}$. A függvény ebben a bázisban is egyértelműen megadható:

$$f = \prod_{i=0}^{2^n-1} (\beta_i + M_i),$$

és most $\alpha_i = \beta_{2^{n-1-i}}$. Ezt az alakot a függvény módosított konjunktív normálalakjának nevezük, és az f Boole-függvény konjunktívan polinomszerű, ha $\underline{\beta} = \underline{k}$, vagyis, ha a módosított normálalak és a Zsegalkin-polinom spektruma azonos [7]. Az előbb megadott hivatkozáshoz tartozó cikk alapján általában $\underline{k} = \mathbf{U}^{(n)} \underline{\beta} = (\mathbf{A}^{(n)} \mathbf{P}^{(n)}) \underline{\beta}$, így a függvény pontosan akkor konjunktívan polinomszerű, ha sajátvektora az $\mathbf{U}^{(n)}$ mátrix által meghatározott transzformációnak. Az előbbi kifejezésben $\mathbf{P}^{(n)}$ olyan mátrix, amelyben a mellékátló elemei 1-ek, míg a többi elem 0. Ismét [7]-ben található $\mathbf{U}^{(n)}$ minimál- és karakterisztikus polinomja, míg [8]-ban látható, hogy $\mathbf{A}^{(n)} \mathbf{P}^{(n)} = \mathbf{A}^{(n)T} \mathbf{A}^{(n)}$. A konjunktívan polinomszerű Boole-függvények is alteret képeznek a Boole-függvények terében, de ennek dimenziója általában kisebb, mint a polinomszerű Boole-függvényeké (lásd [7]-ben az $\mathbf{U}^{(n)}$ sajátértékének multiplicitását).

3. Harmadik fejezet

Új eredményként a polinomszerű Boole-függvények deriváltjával foglalkozunk.

1. Tétel: Legyen $n \in \mathbf{N}$, $n > i \in \mathbf{N}_0$, f n -változós Boole-függvény, és $\varphi(f)$ a hozzá tartozó Zsegalkin polinom. Ekkor $\varphi\left(\frac{d}{dx_i} f\right) = \frac{d}{dx_i}(\varphi(f))$, ahol

$$\frac{d}{dx_i} f = f(x_0, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_{n-1}) \oplus f(x_0, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_{n-1})$$

az f Boole-deriváltja. Hasonlóan, $\frac{d}{dx_i}(\varphi^{-1}(p)) = \varphi^{-1}\left(\frac{d}{dx_i} p\right)$.

Bizonyítás: A tétel igazsága nyilván nem függ attól, hogyan rendezzük a függvény változóit, ezért feltehetjük, hogy $i = n - 1$. Legyen $f = \bar{x}_{n-1} f^{(0)} + x_{n-1} f^{(1)}$, ahol $f^{(0)}$ és $f^{(1)}$ már nem függ x_{n-1} -től, továbbá $\varphi(f) = p$, $\varphi(f^{(0)}) = p^{(0)}$ és $\varphi(f^{(1)}) = p^{(1)}$. Most $\frac{df}{dx_{n-1}} = f^{(0)} \oplus f^{(1)}$ és

$\varphi\left(\frac{df}{dx_{n-1}}\right) = p^{(0)} \oplus p^{(1)}$, hiszen változók azonos halmazán értelmezett Boole-függvények KI-ZÁRÓ VAGY-gyal képzett lineáris kombinációjának transzformáltja a tagok transzformáltjainak hasonló módon képzett összege. De $\varphi(f) = p^{(0)} \oplus x_{n-1}(p^{(0)} \oplus p^{(1)})$, és ennek a deriváltja $\frac{d(\varphi(f))}{dx_{n-1}} = p^{(0)} \oplus p^{(1)}$, ami megegyezik az előző eredménnyel. A tétel második állítása közvetlenül adódik φ invertálhatóságából.

△

- 2. Tétel:** Legalább egyváltozós polinomszerű Boole-függvény i -edik változója szerinti Boole-deriváltja akkor és csak akkor polinomszerű Boole-függvény, ha $f = x_i f^{(1)}$ és $f^{(1)}$ $n-1$ -változós polinomszerű Boole-függvény, ahol $f = \bar{x}_i f^{(0)} + x_i f^{(1)}$.

Bizonyítás: Ismét elegendő azt az esetet nézni, amikor $i = n-1$, hiszen polinomszerű Boole-függvényben a változók indexeit permutálva ismét ugyanannyi változós polinomszerű Boole-függvényt kapunk. Írjuk az n -változós polinomszerű Boole-függvényt $f = \bar{x}_{n-1} f^{(0)} + x_{n-1} f^{(1)}$ alakban, ahol mind $f^{(0)}$, mind $f^{(1)}$ az x_0, \dots, x_{n-2} változók Boole-függvénye, és jelölje $\tau_f(f)$ azt a Boole-függvényt, amelynek a kanonikus diszjunktív normálformájához tartozó spektruma az f Zsegalkin-polinomjának spektrumával azonos. Mivel f polinomszerű Boole-függvény, ezért $f^{(0)} = \tau_f(f^{(1)}) \oplus f^{(1)}$. Ekkor

$$\frac{df}{dx_{n-1}} = f^{(0)} \oplus f^{(1)} = \tau_f(f^{(1)}) \oplus f^{(1)} \oplus f^{(1)} = \tau_f(f^{(1)}),$$

és így $\frac{df}{dx_{n-1}}$ akkor és csak akkor polinomszerű, ha

$$\tau_f(f^{(1)}) = \frac{df}{dx_{n-1}} = \tau_f\left(\frac{df}{dx_{n-1}}\right) = \tau_f(\tau_f(f^{(1)})) = f^{(1)},$$

azaz ha $f^{(1)}$ egy $n-1$ -változós polinomszerű Boole-függvény, és

$$f^{(0)} = \tau_f(f^{(1)}) \oplus f^{(1)} = f^{(1)} \oplus f^{(1)} = 0.$$

Ekkor viszont $f = \bar{x}_{n-1} f^{(0)} + x_{n-1} f^{(1)} = x_{n-1} f^{(1)}$.

△

- 3. Következmény:** az f n -változós nem nulla polinomszerű Boole-függvénynek akkor és csak akkor lesz az $x_{i_0}, \dots, x_{i_{l-1}}$ változók mindegyike szerinti Boole-deriváltja $n-1$ -változós polinomszerű Boole-függvény, ahol $l \leq n$, ha $f = f^{(n)} = f^{(n-l)} \prod_{j=0}^{l-1} x_{i_j}$, ahol $f^{(n-l)}$ az x_{i_j} -ktől különböző $n-l$ változó polinomszerű Boole-függvénye. Ez $l = n$ esetén azt jelenti, hogy $f = \prod_{j=0}^{n-1} x_{i_j} = m_{2^{n-1}}^{(n)} = f_{2^{n-1}}^{(n)}$, és csak ez a Boole-függvény olyan n -változós polinomszerű Boole-függvény, amelynek minden változója szerinti deriváltja $n-1$ -változós polinomszerű Boole-függvény.

Bizonyítás: Az állítás indukcióval könnyen adódik. Ha $l = 1$, akkor ez azonos az előbbi állítással. Most tegyük fel, hogy egy n -nél kisebb pozitív egész l esetén igaz az állítás, és legyen f egy olyan n -változós polinomszerű Boole-függvény, amelynek $l+1$ különböző változója szerinti Boole-deriváltja $n-1$ -változós polinomszerű Boole-függvény. Az általánosság csorbítása nélkül feltehetjük, hogy ez az $n-1$ -változós polinomszerű Boole-függvény, így az in-

dukciós feltevés szerint $f = \prod_{i=0}^{l-1} x_i \cdot \tilde{f}$, ahol \tilde{f} az x_l, \dots, x_{n-1} változók polinomszerű Boole-függvénye. \tilde{f} függ x_l -től, és felírható $\tilde{f} = \bar{x}_l \tilde{f}^{(0)} \oplus x_l \tilde{f}^{(1)}$ alakban az x_{l+1}, \dots, x_{n-1} változók $\tilde{f}^{(0)}$ és $\tilde{f}^{(1)}$ Boole-függvényével. Ha $f^{(0)} = \prod_{i=0}^{l-1} x_i \cdot \tilde{f}^{(0)}$, és hasonlóan, $f^{(1)} = \prod_{i=0}^{l-1} x_i \cdot \tilde{f}^{(1)}$, akkor $f = \bar{x}_l f^{(0)} \oplus x_l f^{(1)}$. Mivel f x_l szerinti Boole-deriváltja is polinomszerű, ezért $f = x_l f^{(1)}$, és $f^{(1)} = \prod_{i=0}^{l-1} x_i \cdot \tilde{f}^{(1)}$ az $x_0, \dots, x_{l-1}, x_{l+1}, \dots, x_{n-1}$ változók polinomszerű Boole-függvénye. $f^{(1)}$ akkor és csak akkor polinomszerű Boole-függvénye ezeknek a változóknak, ha $\prod_{i=0}^{l-1} x_i = m_{2^{l-1}}^{(l)} = f_{2^{2^{n-1}}}^{(l)}$ az x_0, \dots, x_{l-1} változók, és $\tilde{f}^{(1)}$ az x_{l+1}, \dots, x_{n-1} változók polinomszerű Boole-függvénye. $f_{2^{2^{n-1}}}^{(l)}$ -re teljesül ez a feltétel, így akkor és csak akkor lesz f -nek az x_0, \dots, x_l változók mindegyike szerinti Boole-deriváltja $n-1$ -változós polinomszerű Boole-függvény, ha $f = \prod_{i=0}^l x_i \cdot \tilde{f}^{(1)}$, és $\tilde{f}^{(1)}$ az x_{l+1}, \dots, x_{n-1} változók polinomszerű Boole-függvénye.

△

Irodalomjegyzék

- [1] S. B. J. Akers: On a theory of Boolean functions, *J. Soc. Ind. Appl. Math.*, **7** (1959), 487–498.
- [2] R. Beigel: The polynomial method in circuit complexity. In Proceedings of the Eighth Annual Structure in Complexity Theory Conference (San Diego, CA, 1993), Los Alamitos, CA, *IEEE Comput. Soc. Press.* (1993), 82-95.
- [3] P. Calingaert: Switching function canonical forms based on commutative and associative binary operations. In FOCS '61: Proceedings of the 2nd Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1961), Washington, DC, USA, 1961. *IEEE Computer Society* (1961), 217-224.
- [4] M. Davio, J.-P. Deschamps, and A. Thayse: Discrete and switching functions, *Georgi Publishing Co., St.*, (1978).
- [5] J. Gonda: Transformation of the canonical disjunctive normal form of a Boolean function to its Zhegalkin-polynomial and back, *Ann. Univ. Sci. Budapest. Sect. Comput.* **20** (2001), 147-156.
- [6] J. Gonda: Polynomial-like Boolean functions, *Ann. Univ. Sci. Budapest. Sect. Comput.* **25** (2005), 13–23.
- [7] J. Gonda: Conjunctively polynomial-like Boolean functions, *Acta Math. Acad. Paedagog. Nyházi. (N.S.)*, **23** (2) (2007), 89–103.
- [8] J. Gonda: Three remarks on the polynomial-like Boolean functions, *Ann. Univ. Sci. Budapest. Sect. Comput.* **32** (2010), 189-200

- [9] R. J. Lechner: Harmonic analysis of switching functions, In *Recent Developments in Switching Theory*, *Academic Press, New York* (1971), 121-228.
- [10] E. L. Post: Introduction to a general theory of elementary propositions, *Am. J. Math.*, **43** (1921), 163–185.

Automataelméleti alapú titkosítási rendszerek

Cryptosystems based on Automata Theory

Dömösi Pál^a, Horváth Géza^b

^aNyíregyházi Főiskola, Matematikai és Informatikai Intézet
domosi@nyf.hu

^bDebreceni Egyetem, Informatikai Kar
horvath.geza@inf.unideb.hu

Absztrakt: Magyarországon a 60-as évek közepétől folynak intenzív automataelméleti kutatások, elsősorban a szegedi automataelméleti iskolában. Ezen a területen számos fontos magyar eredmény született. Előadásunk célja olyan, a kriptográfiában közvetlenül hasznosítható automataelméleti kutatások áttekintése, melyek oktatása véleményünk szerint a magyarországi felsőoktatásban időszerűvé vált.

Kulcsszavak: titkosítás, automataelmélet

1. Bevezetés

Az alkalmazott titkosítási kulcsok szempontjából alapvetően kétféle titkosítási módszert ismerünk: A szimmetrikus titkosítási rendszer esetén a titkosítás kulcsának ismeretében a visszafejtés kulcsa, illetve a visszafejtés kulcsának ismeretében a titkosítás kulcsa könnyen meghatározható. Emiatt mindkettőnek titkosnak kell lennie, s ezen kulcsokat csupán a titkos üzenet küldője és az üzenet címzettje ismerheti. A nyilvános kulcsú rejtjelezési eljárás esetén a titkosítás kulcsa nyilvános, a visszafejtés kulcsa pedig csak az üzenet fogadója számára ismert [6].

Az automataelmélet [1, 3 – 5] természetes bázist szolgáltat a kriptográfiai rendszerek tervezésére, és számos automataelméleti alapú titkosítási rendszer született [7, 8]. Egy részük Mealy automatákon vagy azok általánosításán, más részük sejtautomatákon alapul. A Mealy automata alapú rendszereknél a nyílt szöveget egy Mealy automata kapja meg inputként, s a titkos szöveg a Mealy automatának ezen input hatására kiváltott outputja lesz. A visszafejtés ugyanígy történik az által, hogy visszafejtés során a titkos szöveg az input, s a visszanyert nyílt szöveg az output. A sejtautomata alapú rendszereknél a nyílt szöveg a sejtautomata egy állapota lesz, s meghatározott számú lépés után keletkező állapot fogja szolgáltatni a titkos szöveget. A visszafejtés ugyanígy történik azáltal, hogy a titkosított szövegből mint állapotból kiindulva meghatározzuk a kezdő állapotot mint visszafejtett szöveget. Mindkét változatnak léteznek szimmetrikus és nyilvános kulcsú változatai is.

Dolgozatunkban ezen rendszer típusok tárgyalása után két további automataelméleti alapú titkosítási rendszert ismertetünk.

2. Hagyományos automataelméleti alapú titkosító rendszerek

Mealy automatán egy $M=(Q,T,V,d,f)$ rendszert értünk, ahol

- Q , T és V nem üres halmazok, az *állapothalmaz*, a *bemenő jelek*, s a *kimenő jelek halmaza*;
- $d: Q \times T \rightarrow Q$ *átmeneti függvény*;
- $f: Q \times T \rightarrow V$ *kimeneti függvény*.

Amennyiben a fenti halmazok végesek, véges Mealy automatáról beszélünk. Ez esetben a fenti Q , T , V halmazokat (állapot, bemenő, kimenő) *ábécéknek* is hívjuk. A véges Mealy automata átmeneti és kimeneti függvényeit táblázatos alakban is szokásos megadni. A táblázat első sorában az állapotok, első oszlopában a bemenő jelek vannak felsorolva, s valamely x_j -vel jelölt sorban és q_i -vel jelölt oszlopban a $(d(q_i, x_j), f(q_i, x_j))$ pár található.

M	q_0	q_1	q_2
x_0	(q_1, y_2)	(q_0, y_1)	(q_2, y_0)
x_1	(q_0, y_1)	(q_2, y_2)	(q_1, y_0)
x_2	(q_1, y_0)	(q_2, y_2)	(q_0, y_1)
x_3	(q_2, y_2)	(q_0, y_0)	(q_1, y_1)

A Mealy automata alapú titkosítási rendszerek kulcsa egy ilyen táblázatból, továbbá az automata egy rögzített állapotából áll. A nyílt szöveg ábécéje a kulcsautomata bemenő jelhalmaza, a titkos szövegé pedig a kulcsautomata kimenő jelhalmaza. Ha például a kulcs a fenti táblázat, s mondjuk, a q_1 állapot, akkor a titkosítás úgy történik, hogy ellenőrizzük, hogy a nyílt szöveg mint input sztring hatására a kulcsautomata a q_1 állapotból kiindulva milyen kimenő sztringet szolgáltat. Mondjuk, ha a nyílt szöveg $x_0x_1x_2x_0x_0x_2$, akkor a titkos szöveg $y_1y_1y_0y_1y_2y_2$ lesz, hiszen $f(q_1, x_0)=y_1$, $d(q_1, x_0)=q_0$, $f(q_0, x_1)=y_1$, $d(q_0, x_1)=q_0$, $f(q_0, x_2)=y_0$, $d(q_0, x_2)=q_1$, $f(q_1, x_0)=y_1$, $d(q_1, x_0)=q_0$, $f(q_0, x_0)=y_2$, $d(q_0, x_0)=q_1$, $f(q_1, x_2)=y_2$, (és $d(q_1, x_2)=q_2$).

A sejtautomata azonos szomszédsági egy, vagy több dimenziós mintázat szerint összekapcsolt, szinkronizáltan működő *sejteknek* hívott elemi automaták olyan összessége, ahol az egyes sejtek állapota csak a saját állapotától és a szomszédainak az állapotától függ. Szinte a sejtautomata kutatások kezdete óta komoly erőfeszítések folynak azok kriptográfiai alkalmazására. Az ilyen típusú kriptográfiai rendszerek rendszerint a sejtautomaták kezdő állapotaként használják a titkosítandó üzenetet és a titkosító kulcs a sejtek átmeneti szabályaiból áll. A meghatározott számú lépés után keletkező állapot szolgáltatja a titkosított üzenetet. A visszafejtés ugyanígy történik azáltal, hogy a titkosított üzenetből mint állapotból kiindulva meghatározzuk a kezdő állapotot mint visszafejtett üzenetet.

A sejtautomata alapú titkosítás elvét egy egyszerű, egy dimenziós modellen keresztül mutatjuk be. A sejt egy kimenőjel nélküli automata, melynek átmeneti függvényét táblázatban adjuk meg. A táblázat első sorában az állapotok, első oszlopában a bemenő jelek vannak felsorolva, s a táblázat valamely x_j -vel jelölt sorhoz és q_i -vel jelölt oszlophoz a $d(q_i, x_j)$, azaz a (q_i, x_j) párhoz rendelt új állapot van hozzárendelve. A bemenő jelek valamely rögzített $n \geq 1$ -re az állapothalmaz elemeiből képzett n dimenziós (a_1, \dots, a_n) vektorok, ahol is valamely m -re a_m az adott sejtnak a sejtautomata topológia szerinti m -edik szomszédja állapotát jelöli. A

sejtautomata alapú titkosítási rendszerek kulcsa egy ilyen táblázatból, továbbá a szomszédsági topológia megadásából áll.

Tegyük fel, hogy a sejtautomata sejtjei olyan egyszerű lineáris topológia szerint helyezkednek el, ahol is minden egyes sejt átmenetét saját állapotán kívül csupán a baloldali szomszédja határozza meg. Ekkor egy két állapotú sejt átmeneti táblázata a következő alakú lesz:

S	0	1
0	0	1
1	1	0

Jelölje $d: \{0,1\} \times \{0,1\} \rightarrow \{0,1\}$ a sejt ezen táblázattal definiált átmeneti függvényét. Tegyük fel továbbá, hogy az első sejt úgy viselkedik, mintha olyan baloldali szomszédja lenne, melynek állapota fixen 1. Álljon a sejtautomata négy sejtéből, s a nyílt szöveg blokk legyen például a négy hosszúságú 1001 bináris sztring. Ekkor az első sejt a 0 állapotba megy át, hisz az 1 állapotban van és bal szomszédja rögzítetten 1-es, s emiatt $d(1,1)=0$. A második sejt az 1 állapotba megy át, hiszen 0 állapotban van és bal szomszédja 1-es, s emiatt $d(0,1)=1$. A harmadik sejt a 0 állapotba megy át, hiszen 0 állapotban van és bal szomszédja 0, s emiatt $d(0,0)=0$. Végül, a negyedik sejt az 1 állapotba megy át, hiszen 1 állapotban van és bal szomszédja 0, s emiatt $d(1,0)=1$. Esetünkben tehát az 1001 nyílt szöveghez tartozó titkos szöveg 0101 lesz. A táblázat és a topológia ismeretében a titkos szövegből a nyílt szöveg hasonló elemzéssel nyerhető vissza.

Mindkét modellnek számos változata ismeretes.

3. Egy folyamtitkosító

Most egy olyan folyamtitkosító rendszer működésének elvét ismertetjük, melyben a kulcs egy végállapotokkal ellátott iníciaális kimenő jel nélküli véges automata, s a titkosítás folyamán az automata segítségével a nyílt szöveg minden egyes karakteréhez egy adott hossztartományba eső, változtatható hosszúságú és véletlenszerűen választott karaktersorozat hozunk létre és ezen karaktersorozat összefűzéséből állítjuk elő a titkosított üzenetet [2].

Legyen a kulcs-automata átmeneti mátrixa - ahol 0 a kezdő állapot - az alábbi:

Az egyszerűség (bináris) kedvéért tegyük fel, hogy a nyílt szöveg karakterkészlete $\{0, 1\}$. Rendeljük a 0 karakterhez a 0 -t mint végállapotot, az 1 karakterhez pedig 1 -t mint végállapotot. Jelölje tetszőleges $\{0,1,2,3,4\}$ -beli i,j párra $L_{i,j}$ az összes olyan bemenő jelsorozat halmazát, melyek úgy viszik át az automatát az i állapotból a j állapotba, hogy közbülső állapotként egyetlen végállapot sem lép fel.

M	0	1	2	3	4
0	4	1	3	0	3
1	0	4	1	2	2

Titkosítás

Tekintsük az OK szónak a hexadecimális ASCII kódját: 4F4B. Ezt a 4F4B kódot binárisra konvertálva megkapjuk a 0100111101001011 sztringet. Tegyük fel, hogy ez a nyílt szövegünk. A rendszer a 0100111101001011 bitenkénti 0,1,0,0,1,1,1,1,0,1,0,0,1,0,1,1 értékeinek mindegyikéhez legalább egy bit és legfeljebb hat bit hosszú véletlenszerűen választott bináris jelsorozatot rendel.

Részletesebben, a rendszer az elsőként beolvasott bit hatására olyan véletlenszerű (legalább egy és legfeljebb hat bit hosszú) bitsorozatot generál, mely a kulcs-automatát a kezdő állapotból (azaz a 0 állapotból) az első beolvasott bit értékéhez (azaz a 0-hoz) rendelt végállapotba (azaz a 0 állapotba) mint elérendő végállapotba úgy tudja átvinni, hogy közben közbülső állapotként egyetlen végállapot (azaz sem a 0, sem az 1 állapot) sem lép fel. Konkrétabban, a 0 jel hatására az $L_{0,0}$ egy véletlenszerűen kiválasztott (legalább egy és legfeljebb hat bit hosszú) elemét fogja kiadni.

Ezt követően az aktuális állapot az előző elérendő állapot (0 állapot) lesz, s a rendszer a következő bit hatására olyan véletlenszerű (legalább egy és legfeljebb hat bit hosszú) bitsorozatot generál, mely a kulcs-automatát az aktuális állapotból (azaz a 0 állapotból) a második beolvasott bit értékéhez (azaz az 1-hez) rendelt végállapotba (azaz az 1 állapotba) mint elérendő végállapotba úgy tudja átvinni, hogy közbülső állapotként egyetlen végállapot (azaz sem a 0, sem az 1 állapot) nem léphet fel. Konkrétabban, a rendszer a másodikként beérkező 1 jel hatására az $L_{0,1}$ egy véletlenszerűen kiválasztott (legalább egy és legfeljebb hat bit hosszú) elemét fogja kiadni.

Hasonlóan működik a berendezés a beolvasott bitsorozat harmadik, negyedik, . . . , tizenhatodik bitjének hatására, melynek során az $L_{0,0}$ és az $L_{0,1}$ egy-egy véletlenszerűen kiválasztott (legalább egy és legfeljebb hat bit hosszú) elemének megjelenését követően sorra meg fog jelenni az $L_{1,0}$, $L_{0,0}$, $L_{0,1}$, $L_{1,1}$, $L_{1,1}$, $L_{1,1}$, $L_{1,0}$, $L_{0,1}$, $L_{1,0}$, $L_{0,0}$, $L_{0,1}$, $L_{1,0}$, $L_{0,1}$, $L_{1,1}$ halmazok (legalább egy és legfeljebb hat bit hosszúságú) egy-egy véletlenszerűen kiválasztott eleme.

Tegyük fel például, hogy ezek a véletlenszerűen kiválasztott (legalább egy és legfeljebb hat bit hosszú) bitsorozatok a következők:

1 (eleme $L_{0,0}$ –nak),
0011 (eleme $L_{0,1}$ –nek),
100 (eleme $L_{1,0}$ –nak),
1 (eleme $L_{0,0}$ –nak),
011 (eleme $L_{0,1}$ –nek),
0 (eleme $L_{1,1}$ –nek),
0 (eleme $L_{1,1}$ –nek),
1011 (eleme $L_{1,1}$ –nek),
100 (eleme $L_{1,0}$ –nak),
011 (eleme $L_{0,1}$ –nek),
1100 (eleme $L_{1,0}$ –nak),
1 (eleme $L_{0,0}$ –nak),

0011 (eleme $L_{0,1}$ –nek),
10100 (eleme $L_{1,0}$ –nak),
001011 (eleme $L_{0,1}$ –nek),
0 (eleme $L_{1,1}$ –nek).

Ez esetben az 1, 0011, 100, 1, 011, 0, 0, 1011, 100, 011, 1100, 1, 0011, 10100, 001011, 0 bitsorozatok összefüzéseként előáll egy 100111001011001011100011110010011101000010110 bitsorozat, mely a 0100111101001011 titkosítandó üzenet egy (véletlenszerűen választott) titkosított üzenete lesz.

Visszafejtés

A rendszer az 100111001011001011100011110010011101000010110 bitsorozatot kapja meg titkos üzenetként.

A visszafejtő algoritmus a 100111001011001011100011110010011101000010110 titkos üzenetet bitről-bitre beolvastva ellenőrzi, hogy ha az automatát a kezdő állapotából (azaz a 0 állapotból) elindítja, akkor a fellépő bitek mint bemenő jelek hatására milyen állapotsorozaton halad keresztül. Ezen ellenőrzés során a fellépő végállapotokat (0 vagy 1 állapotokat) kiírja. (A 2,3,4 állapotok fellépését tehát figyelmen kívül hagyjuk.) Ezen értékek rendre így adódnak: 0,1,0,0,1,1,1,1,0,1,0,0,1,0,1,1. Az eszközről érkezett ezen bináris értékek összefüzéseként megkapjuk a 0100111101001011 bitsorozatot, mely a visszafejtett titkosítandó üzenet lesz. Ezen bitsorozatot hexadecimális alakba felírva visszakapjuk a 4F4B hexadecimális karaktersorozatot, amit ASCII kódnak tekintve visszanyerhetjük az OK szót.

4. Egy blokktitkosító

Dolgozatunkat egy automaták Gluskov szorzatain alapuló blokktitkosító elvének ismertetésével zárjuk. Az automaták kompozícióit automaták egy olyan rendszere képezi, melyek egy kommunikációs hálózaton keresztül kapcsolatban vannak egymással. Minden automata diszkrét időpillanatokban képes állapotait változtatni az állapotok és egy globális input egy lokális átmeneti függvényével. A lokális állapot átmenetek egy globális állapot átmenetet eredményeznek a teljes hálózaton. Ez a fogalom a sejtautomata egy olyan irányú általánosítása, mikor is a globális átmenetet a sejtek (mint elemi automaták) állapotai mellett egy külső input is befolyásolja. Amennyiben a kompozíció minden egyes tagja egy és ugyanazon állapothalmazzal rendelkezik, állapot-homogén automata kompozícióról beszélünk. A globális input minden egyes eleme – melyeket bemenő jelnek is hívunk - tekinthető úgy is mint egy transzformáció, mely a globális állapothalmazt önmagába képezi le. Bemenő jelek egymás utáni alkalmazásával egy újabb transzformációhoz jutunk.

Megfelelően választott kompozíciós struktúrák segítségével lehetőség nyílik nagyobb méretű kulcs-automaták gazdaságos tárolására. Nevezetesen, elegendő tárolni a kompozíció struktúráját, a komponens automaták kommunikációs függvényeit, s a kompozícióban szereplő elemi automata típusokat. Ha a komponens automaták átmeneti táblázatainak és a kommunikációs függvény (táblázatos vagy másféle) megadásának tárigénye a kompozíció-automata átmeneti táblázatának tárigényéhez képest csekély, akkor ez a megoldás célszerű

minden olyan esetben, mikor a visszacsatolási függvény értékek meghatározása kis lépésszámú algoritmussal megoldható.

A titkosítás kulcsa egy kimenő jel nélküli automata igen nagy méretű Gluskov hatványa. Ezen Gluskov hatvány állapothalmazát a nyílt szöveg összes lehetséges blokkjai alkotják, bemenő jelhalmaza pedig egybeesik az állapothalmazzal. A titkosítás során egy pszeudo-véletlenszám generátor a nyílt szöveg minden egyes blokkjához generálja a szorzat-automata véletlenszerűen választott bemenő jeleiből álló, adott számú, egy vagy több blokk hosszúságú input sztringet, s kiszámítja, hogy ez az input sztring a nyílt szöveg soron következő (először ez első) blokkját mint állapotot melyik állapotba viszi át. A kapott új állapot lesz a tekintett nyílt szövegblokkhoz tartozó titkos szöveg blokk, a titkos szöveg pedig ezen titkos szöveg blokkok konkatenációja lesz.

Egy nagyon egyszerű példával illusztráljuk ezt a rendszert is. A legegyszerűbb Gluskov hatványok egyike, mikor is

- a hatványt alkotó (azonos struktúrájú) automaták egy gyűrű-szerű struktúrában helyezkednek el,
- a globális bemenő jel egy annyi dimenziós vektor, mint ahány automata a hatványt alkotja,
- a hatvány első tagját alkotó automata állapot átmeneteit a hatvány első, valamint az utolsó tagját alkotó automaták állapota, továbbá a globális bemenő jel (mint vektor) első komponense,
- s minden további $i > 1$ esetén a hatvány i -edik tagját alkotó automata állapot átmeneteit a hatványt alkotó i -edik és $(i-1)$ -edik automaták állapota, továbbá a globális bemenő jel (mint vektor) i -edik komponense határozza meg.

Egy ilyen Gluskov hatványt alkotó minden egyes automata visszacsatolási függvényértékét tehát egy olyan pár határozza meg (a legegyszerűbb esetben ez a pár alkotja), melynek

- első tagja a tekintett automatát ezen gyűrű-szerű struktúrában megelőző automata állapota,
- második tagja pedig a globális bemenő jel (mint vektor) azon komponense, melynek sorszáma a tekintett automata sorszámával megegyezik.

Természetesen ennél jóval bonyolultabb Gluskov hatványokat is lehet tekinteni, kutatás kérdése, hogy érdemes-e.

Legyen az A automata átmeneti mátrixa - ahol 0 a kezdő állapot - az alábbi:

A	0	1
00	0	1
01	1	0
10	1	0
11	0	1

Tekintsük az előbbi szerkezetű (gyűrűszerűen kapcsolódó elemekből felépülő) A^4 (Gluskov hatvány-) automatát.

Ha tehát az A^4 automata egy abcd állapotban van, s ez az A^4 automata a vwxy bemenő jelet kapja (ahol a,b,c,d,v,w,x,y mindegyike $\{0, 1\}$ -beli), akkor átmegy egy olyan a'b'c'd' állapotba, mely a fenti átmeneti táblázat által meghatározott $\{0, 1\}$ -beli $a' = \delta(a, (d,v))$, $b' = \delta(b, (a,w))$, $c' = \delta(c, (b,x))$, $d' = \delta(d, (c,y))$ elemekből áll.

Titkosítás

Tekintsük a hexadecimális 4F4B ASCII kódját az OK szónak.

Ezt a 4F4B kódot binárisra konvertálva megkapjuk a 0100111101001011 sztringet.

Tegyük fel, hogy ez a nyílt szövegünk.

- A titkosítás során először adjunk A^4 -nek egy véletlen bemenő jelet, például az $y_0 = 1001$ -et. Ennek hatására A^4 átmegy az $a_0 = 0000$ kezdő állapotból az $y_1 = 1001$ állapotba. A titkos szöveg első bitnégyese tehát $y_1 = 1001$ lesz.

Ezután adjuk A^4 -nek azt a bemenő jelet, mely ebből az $y_1 = 1001$ állapotból a nyílt szöveg első bitnégyesével megegyező sorszámú $s_1 = 0100$ állapotba viszi át. A megfelelő bemenő jel $x_1 = 0001$.

- Most meg kell határoznunk az A^4 azon y_2 állapotát, melybe az $y_0 = 1001$ állapotból az $y_1 = 1001$ bemenő jel hatására átmegy: $y_2 = 1100$.

Ezután adjuk A^4 -nek azt a bemenő jelet, mely ebből az $y_2 = 1100$ állapotból a nyílt szöveg második bitnégyesével megegyező sorszámú $s_2 = 1111$ állapotba viszi át. A megfelelő bemenő jel $x_2 = 0101$.

- Meg kell határoznunk az A^4 azon y_3 állapotát, melybe az $y_1 = 1001$ állapotból az $y_2 = 1100$ bemenő jel hatására átmegy: $y_3 = 1001$.
- A következő bemenő jel az A^4 -t az előző $y_3 = 1001$ állapotból a nyílt szöveg harmadik bitnégyesével megegyező sorszámú $s_3 = 0100$ állapotba viszi át az automatát. A megfelelő bemenő jel $x_3 = 0001$.
- Végül, határozzuk meg az A^4 azon y_4 állapotát, melybe az $y_2 = 1100$ állapotból az $y_3 = 1001$ bemenő jel hatására átmegy: $y_4 = 0011$.

Az utolsó bemenő jel az A^4 -t az előző $y_4 = 0011$ állapotból a nyílt szöveg negyedik bitnégyesével megegyező sorszámú $s_4 = 1011$ állapotba viszi át az automatát. A megfelelő bemenő jel $x_4 = 0001$.

A 0100111101001011 nyílt szöveghez tartozó egyik titkosított szöveg tehát az 1001, 0101, 1001, 0001, 0001 sztringek 100101010100010001 konkatenációja lesz.

Visszafejtés

Tekintsük az 10010001010100010001 bitsorozatot titkosított szöveggént. Ezt a szöveget bitről-bitre beolvasva ellenőrizzük, hogy ha az A^4 automatát a kezdő állapotából (azaz a 0000 állapotból) elindítjuk, akkor a fellépő bitek mint bemenő jelek hatására milyen állapotsorozaton halad keresztül.

- $a_0 = 0000$ állapotból $y_1 = 1001$ állapotba az (véletlenszerűen választott jel) $y_0 = 1001$ hatására megy át,
- $y_1 = 1001$ állapotból $x_1 = 0001$ hatására $s_1 = 0100$ -ba megy át, mely a nyílt szöveg első bitnégyese,
- $y_0 = 1001$ állapotból $y_1 = 1001$ hatására az A^4 automata $y_2 = 1100$ állapotba, onnan pedig $x_2 = 0101$ hatására $s_2 = 1111$ -be megy át, mely a nyílt szöveg második bitnégyese,
- $y_1 = 1001$ állapotból $y_2 = 1100$ hatására az A^4 automata $y_3 = 1001$ állapotba, onnan pedig $x_3 = 0001$ hatására $s_3 = 0100$ -ba megy át, mely a nyílt szöveg harmadik bitnégyese,
- $y_2 = 1100$ állapotból $y_3 = 1001$ hatására az A^4 automata $y_4 = 0011$ állapotba, onnan pedig $x_4 = 0001$ hatására $s_4 = 1011$ -be megy át, mely a nyílt szöveg negyedik bitnégyese.

5. Köszönetnyilvánítások

The first author was supported by grants of Economic Development Operational Programme of Hungary, GOP-1.1.1-11-2012-0156, "Automata Theory-based, efficient and secure encryption system for computer equipments".

The second author was supported by the TÁMOP-4.2.2.C-11/1/KONV-2012-0001 project. The project has been supported by the European Union, co-financed by the European Social Fund.

Irodalomjegyzék

- [1] Pál Dömösi & Chrystopher L. Nehaniv (2005): *Algebraic theory of automata networks: An introduction*. SIAM Monographs on Discrete Mathematics and Applications 11, Society for Industrial and Applied Mathematics, (SIAM), Philadelphia, PA, doi:10.1137/1.9780898718492.
- [2] Pál Dömösi (2010): A novel cryptosystem based on finite automata without outputs. *Automata, formal languages and algebraic systems*, 23–32, World Sci. Publ., Hackensack, NJ, 2010.
- [3] Ferenc Gécseg (1986): *Products of automata*. EATCS Monographs on Theoretical Computer Science 7, Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, doi:10.1007/978-3-642-61611-2.
- [4] Ferenc Gécseg & István Peák (1972): *Algebraic theory of automata*. Disquisitiones mathematicae Hungaricae 2, Akadémia Kiadó, Budapest.
- [5] John E. Hopcroft, Rajeev Motwani & Jeffrey D. Ullman (2001): *Introduction to Automata Theory, Languages and Computation*, second edition. Addison-Wesley series in computer science, Addison-Wesley.

- [6] Alfred J. Menezes, Paul C. van Oorschot & Scott A. Vanstone (1996): *Handbook of Applied Cryptography*. Discrete Mathematics and Its Applications, CRC Press. doi:10.1201/9781439821916.
- [7] Valtteri Niemi: Cryptology (1997): Language-Theoretic Aspects. In: Grzegorz Rozenberg és Arto Salomaa (szerk.): *Handbook of Formal Languages*, 2. kötet, Springer, Berlin, 1997, 507-524.
- [8] Renji Tao (2009): *Finite Automata and Application to Cryptography*. Springer-Verlag, Berlin. doi:10.1007/978-3-540-78257-5.

Hálózatok kutatása a Tudáskezelő rendszerek labor tantárgy keretein belül az ELTE-n

Network research in the Knowledge Management Systems course in ELTE

Kósa Balázs^a, Kiss Attila^b

^aEötvös Loránd Tudományegyetem, Információs Rendszerek Tanszék
balhal@inf.elte.hu

^bEötvös Loránd Tudományegyetem, Információs Rendszerek Tanszék
kiss@inf.elte.hu

Absztrakt: Az ELTE IK Információs Rendszerek Tanszékén 2011 őszén indítottuk el a négy szemeszteres Tudáskezelő rendszerek labor tantárgyat elsősorban MSc-s hallgatók részére. Cikkünkben két csoport elmúlt két féléves munkáját és elért eredményeit szeretnénk bemutatni. Mindkét esetben a hálózatok kutatása egy-egy témakörére fókuszáltunk. Az első csoportban a hatásmaximalizálási feladat megoldására javasolt algoritmusokat néztük át, majd álltunk elő egy saját változattal. Itt adott egy információ-terjedési modell, egy k küszöbszám és egy hálózat. A kérdés: melyik k emberrel osszuk meg az elterjeszteni kívánt információt az első lépésben, hogy végül a lehető legtöbb emberhez eljusson a hír. Az információ itt lehet például egy új termék a piacon, azaz nem garantált, hogy akihez eljut a hír, tovább is adja azt az ismerőseinek. Az általunk javasolt megoldás egy korábbi módszert módosít, lényegében az algoritmusban használt klaszterezési eljárást cseréli le egy másikra. Ennek köszönhetően a futási idő százezres élszámú gráfoknál tizedére csökken; igaz, az elért emberek száma is, viszont ez utóbbi csupán legfeljebb 4 százalékkal. A másik csoportban első feladatként két, a nagyméretű hálózatok feldolgozására használt elosztott rendszer – HADOOP és Giraph – teljesítményét hasonlítottuk össze egymással és a szekvenciális módszeren alapuló megoldások teljesítményével. Másodszor a köztesesség (betweenness) és Linerank centralitási mértékek kapcsolatát vizsgáltuk. Ez utóbbit a Google és IBM kutatói javasolták az előbbi egy lehetséges alternatívájaként, mivel a köztesesség költséges számítása miatt nagyobb hálózatokon már nem használható. Első lépésként a két mérték valós és véletlen gráfokon vett Pearson és Spearman korrelációs együtthatóit vizsgáltuk; majd két algoritmus teljesítményének változását elemeztük a köztesesség Linerankre történő cseréje után. A fentiekben bemutatott eredményeket három cikkben publikáltuk [4] [6] [13]. A kutatás során elkészített algoritmusokat egy Java könyvtárba gyűjtöttük össze, melyet modulként a RapidMiner adatelemző szoftverbe is beépítettünk. Ennek működését a RapidMiner World 2014 konferencia keretében ismertettük [8].

Kulcsszavak: szociális hálózatok, hatásmaximalizálási feladat, elosztott rendszerek, centralitási mértékek

Abstract: The four semester long Knowledge Management Systems course was launched in the autumn of 2011 in the Department of Information Systems in ELTE. In this paper we present the results of two groups, which were achieved in the last two semesters. In both groups we focused on topics of network research. In the first group after examining algorithms solving the influence maximization problem we developed a new variant of these methods. In the influence maximization problem for a given information diffusion model, threshold k and a network one is to find those k actors of the network with whom the information should be shared initially in order to reach the largest number of actors at the end of the process. The information can be a new product, in other words, it is not guaranteed that after obtaining the information one automatically shares it with their friends. Basically, our method modifies an earlier algorithm, we substituted the clustering algorithm used as a submethod to another one. As a result, in the case of graphs with hundred thousand edges the running time decreases by an order of magnitude, while the number of the reached actors declines by 4% only. In the second group first we compared the performance of two distributed systems HADOOP and Pregel with each other and with the performance of methods implemented sequentially. The aforementioned two systems are used to handle networks of larger size. Secondly, we analyzed the relationship between two centrality measures, betweenness and Linerank. The latter measure was developed by the researchers of Google and IBM as an alternative to the

former one, since betweenness owing to its costly computation time cannot be applied on larger networks. As a first step we calculated the Pearson and Spearman correlation coefficients on both random and real-world graphs. Afterwards, we analyzed the change of the performance of two algorithms after substituting betweenness with Linerank in their computation. We published our results in three papers [4] [6] [13]. The algorithms implemented during our research were collected in a Java library, which was added to the data analyzing software RapidMiner as a modul. The functionalities of this modul were presented in the RapidMiner World 2014 Conference [8].

Keywords: social networks, influence maximization, distributed systems, centrality measures

1. Bevezetés

Az ELTE IK Információs Rendszerek Tanszékén 2011 őszén indítottuk el a négy szemeszteres Tudáskezelő rendszerek labor tantárgyat elsősorban MSc-s hallgatók részére. A munka egy-egy oktató vezetésével 4-5 fős csoportokban folyik heti 4 órában, amely mellé a hallgatók heti rendszerességgel 4-8 órányi otthon elvégzendő feladatot kapnak. A félév első felében általában a választott témakörhöz kapcsolódó elméleti háttér feltérképezése folyik, a résztvevőknek cikkeket kell feldolgozniuk, amely alapján prezentációt kell készíteniük és a többieknek előadást kell tartaniuk. A cikkek és a hozzá tartozó előadások anyagai egy a laborhoz tartozó TWiki oldalra kerülnek fel. A félév második felében aztán a megszerzett ismeretanyag alapján egy vagy több gyakorlati feladatot kell a hallgatóknak megoldaniuk. A teljesség igénye nélkül idáig a következő témakörök kerültek terítékre: közlekedési adatok bányászata, szemantikus portál készítése, twitter adatok adattárházba történő gyűjtése és vizualizációja, kiterjesztett valóság és ehhez kapcsolódóan android alkalmazások implementálása, gráfadatbázis-kezelő rendszerek vizsgálata stb. A csoportok több alkalommal ipari és pályázati projekteken is dolgoztak már például a FuturICT, KIC EIT ICT Labs keretei között. Az indulás óta több, mint 12 TDK, 4 szakdolgozat, 8 diplomamunka és 8 konferenciacikk született. A résztvevők közül többen is úgy döntöttek, hogy a laborfoglalkozásokon kapott első eredményeket doktori munka keretében teszik teljesebbé. A gyakorlati feladatok megoldása során elkészült prototípusok az ipari és más egyetemek tanszékeivel való kapcsolatot is elősegítik. Immáron hagyománnyá vált, hogy az őszi és tavaszi foglalkozások mellett nyári szakmai gyakorlatok is indulnak.

Cikkünkben két csoport elmúlt két féléves – 2013-2014-es tanév első és második szemeszter – munkáját és elért eredményeit szeretnénk bemutatni. A csoportokhoz tartozó TWiki oldal egy részlete az 1. ábrán található.

(b) Zoli: a pontok hasonlóságát számoló algoritmus esetében megnézni, mi a gond.

2. hét

(a) Peti: a [Defining and identifying communities in networks](#) című cikk átnézése, bemutatása, a Girvan-Newman algoritmus megvalósítása addig a részig, ahol a különböző klaszterezések jóságát hasonlítanánk össze, valamint a cikkekben szereplő adatszetek összeszedése.

- Cikkfeldolgozás [vázlata](#)
- [Lista](#) a cikkekben szereplő adatszetekről méretükkel (csúcsszám). Amelyiket sikerült letölteni, ott egy fájlnev is szerepel.

A legtöbb megtalálható ezekben a gyűjteményekben:

<http://www-personal.umich.edu/~mejn/netdata/>

<http://deim.urv.cat/~aarenas/data/welcome.htm>

- A készült kódok felkerültek a közös repository-ba

Marci: a [Benchmark graphs for testing community detection algorithms](#) című cikk átnézése, bemutatása, az ott bevezetett vlen gráf konstrukciót megvalósító implementáció keresése. Ha nincs, saját implementáció megvalósításának előkészítése.

Eredmény: [benchmarks_first.pdf](#)

1. ábra. A Tudáskezelő rendszerek labor tantárgy TWiki oldalának egy részlete.

Mindkét esetben a hálózatkutatás egy-egy témakörére fókuszáltunk. Az egyes csoport esetében a hatásmaximalizálási feladat megoldására javasolt algoritmusokat néztük át, majd álltunk elő egy új változattal. Részletek a 2. fejezetben találhatóak. Az eredményekből egy publikáció készült [13].

A másik csoportban első feladatként két, nagyméretű adatok, hálózatok feldolgozására használt elosztott rendszer – HADOOP és Giraph – teljesítményét hasonlítottuk össze egymással és a szekvenciális módszeren alapuló megoldások teljesítményével. Második feladatként pedig a köztesség (betweenness) [12] és Linerank [5] centralitási mértékek kapcsolatát vizsgáltuk. Ez utóbbit a Google és IBM kutatói javasolták az előbbi egy lehetséges alternatívájaként, mivel a köztesség költséges számítása miatt nagyobb hálózatokon már nem használható. Első lépésként a két mérték valós és véletlen gráfokon vett Pearson és Spearman korrelációs együtthatóit vizsgáltuk; majd két algoritmus teljesítményének változását elemeztük a köztesség Linerankre történő cseréje után. A részletek a 3. és 4. fejezetben kerülnek ismertetésre. Az eredményeket a [4] [6] cikkekben publikáltuk.

Az előző bekezdésekben említett kutatások során elkészített algoritmusokat egy Java könyvtárba gyűjtöttük össze, melyet modulként a RapidMiner adatelemző szoftverbe is beépítettünk. Ennek működését a RapidMiner World 2014 konferencia keretében ismertettük [8]. Egy példa az 5. fejezetben olvasható.

2. A hatásmaximalizálási feladat egy lehetséges megoldása

A hatásmaximalizálási feladatnál adott egy – általában szociális – irányított vagy irányítatlan hálózat, egy információ-terjedési modell, valamint egy k küszöbszám, s ezek mellett keressük azt a k pontot a hálózatban, akikhez első lépésben eljuttatva az elterjeszteni kívánt információt a folyamat végén a lehető legtöbb ponthoz jut el a hír. Az alapvető információ-terjedési modellek a [6] munkában kerültek bevezetésre. Mi a szakirodalomban leginkább elemzett *Független továbbgyűrűzési modellre (Independent Cascade Model)* koncentráltunk. Itt az információ diszkrét lépésekben terjed. Kezdetben, azaz a 0. lépésben a pontoknak egy A halmaza *aktív*. Az i . lépésben minden $(i - 1)$. lépésben aktívvá vált pont megpróbálhatja aktiválni a még nem aktív szomszédait ($0 < i$). A modellben minden u, v szomszédos pontpár esetén – azaz u -ból v -be indul el – adott egy p_{uv} valószínűség, amellyel az u pont aktiválhatja v -t. Tetszőleges u pont a folyamat során legfeljebb egyszer próbálhatja meg aktiválni a szomszédait. Ha egy pont aktívvá vált, többé nem lesz inaktív.

Jelölje $\sigma(A)$ az A ponthalmaz hatását, azaz azon pontok számát, amelyek a folyamat végén aktívak lesznek, ha kezdetben A elemei lettek aktivizálva. Mivel a folyamat nemdeterminisztikus, $\sigma(A)$ értékét Monte-Carlo szimuláció segítségével közelítik. Megmutatható, hogy ez az érték tetszőleges δ hibahatárral megközelíthető kellő mennyiségű szimuláció alkalmazásával. A gyakorlatban ez 10-20 ezer szimulációt jelent.

A hatásmaximalizálási feladat során tehát keressük azt az A ponthalmazt, melynek elemszáma k és $\sigma(A)$ a lehető legnagyobb a k méretű halmazok hatásai közül. Kempe és munkatársai [6] megmutatták, hogy a feladat NP-nehéz. Emellett azt is bizonyították, hogy a mohó algoritmus legfeljebb 63 százalékkal adhat rosszabb eredményt, mint az optimális megoldás. Formálisan:

$$\sigma(S^{mohó}) \geq \left(1 - \frac{1}{e}\right) \sigma(S^{opt}),$$

ahol $S^{mohó}$ és S^{opt} rendre a mohó algoritmus adta és az optimális megoldást jelölik. A Monte-Carlo szimulációk végrehajtása azonban, különösen nagyméretű hálózatok esetében, időigényes, így több későbbi publikációban is a módszer felgyorsítására tettek javaslatokat. A [14] cikkben például egy jól megválasztott klaszterező¹ algoritmus segítségével a hálózatot először csoportokra bontják, majd az egyes pontok és ponthalmazok hatását csupán a nekik megfelelő csoportokban mérik. A klaszterező algoritmusban egy kezdeti partíció létrehozása után a második lépésben azokat az A, B klasztereket vonják össze, ahol az A klaszterben van olyan u pont, melynek van olyan B klaszterbeli v szomszédja, ahol v A klaszteren kívüli hatásának és u A klaszteren belüli hatásának hányadosa meghalad egy küszöböt, azaz u a v ponton keresztül még túlságosan sok A klaszteren kívüli pontot érhet el. Ily módon egy a fentiekhez hasonló hibahatár az optimális megoldáshoz képest továbbra is garantálható. Az algoritmusra a 2. ábrán CGA-ként (*Community Based Greedy Algorithm, Közösségalapú mohó algoritmus*) hivatkozunk.

[13] cikkünkben az iménti klaszterező eljárást helyettesítettük egy másikkal [15], melynek hatására az iménti hibahatár már nem biztosítható, ugyanakkor a teljes algoritmus futási ideje a kísérletek során legalább egy nagyságrenddel javult, míg a folyamat végén aktívvá vált pontok száma legrosszabb esetben 4%-kal lett kevesebb, mint az eredeti módszer esetében.

¹ A cikkben a klaszter és közösség, valamint a gráf és hálózat kifejezéseket ugyanabban az értelemben, felváltva használjuk.

Az algoritmusokat Java 1.7 nyelven implementáltuk, a mérésekhez egy darab 12 magos gépet használtunk 2,67 GHz Intel Xeon CPU processzorokkal és 24 Gb belső memóriával. A 37154 pontot és 231584 élet tartalmazó NetPHY hálózatra vonatkozó eredmények a 2. ábrán láthatók. A hálózatot Wei Chen és munkatársai készítették [2] az arXiv tudományos publikációkat tartalmazó adatbázis fizika szekciójának alapján. Két kutatót reprezentáló pont között akkor megy él, ha van közös cikkük. Az 1991 és 2003 között megjelent publikációkat vették figyelembe. A 2. ábra utolsó sorában szereplő NewGreedy algoritmus szintén az iménti cikkben került bevezetésre, s a Monte-Carlo szimulációknak egy gyorsabb számítási lehetőségét aknázza ki.

	Futási idő (sec)			Hatás
	klaszterezés	mohó	összes	
saját	7	373	380	890
CGA	21	1203	1224	915
NewGreedy		4021	4021	916

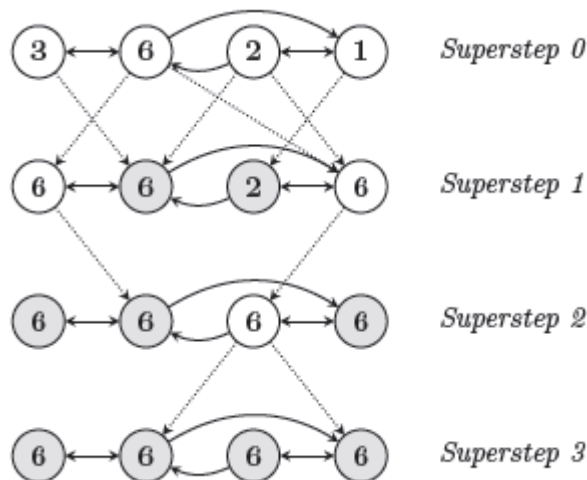
2. ábra. A NetPHY hálózatra vonatkozó adatok.

3. Elosztott rendszerek teljesítménye nagyméretű hálózatokon

A nagyméretű adatok feldolgozására használt, újonnan bevezetett elosztott rendszerek közül talán a MapReduce modellen [3] alapuló megvalósítások a legnépszerűbbek. Ennek egy lehetséges javításaként fejlesztették ki a Google kutatói a Pregel rendszert [10], amely kifejezetten hálózatok, gráfok hatékony kezelését tűzi ki célul. Mindkét esetben az adatok elosztása (*data distribution*), a másolatok létrehozása (*replication*), egy-egy számítási pont kiesésének kezelése (*fault tolerance*) valamint a terhelés kiegyensúlyozása (*load balancing*) automatikusan, a felhasználó elől elrejtve történik. Az ő feladata csupán azon függvények elkészítése, amelyek aztán párhuzamosan futnak az egyes számítási pontokon.

Mivel a MapReduce programozási modell széles körben elterjedt, emiatt csak a Pregel rendszert ismertetjük röviden. A Pregel programok bemenete minden esetben egy irányított gráf, ahol mind a pontokhoz, mind az élekhez egy felhasználó által módosítható érték tartozhat. Egy-egy számítási pont, a gráf egy-egy pontjának felel meg, így a bemenet szerkezete a számítási pontokban tükröződik. Ez jelentős javulást jelenthet az egyes algoritmusok futási idejében a MapReduce rendszerekhez képest, hiszen ott, ha a bemenet egy hálózat, akkor annak szerkezetét a legtöbb esetben minden Map-Reduce függvénypárnak valamiképpen továbbítania kell a soron következő Map-Reduce függvénypárhoz, ami jelentős mértékben lassíthatja a végrehajtási időt. Az élek a kezdőpontjukhoz tartozó számítási ponton tárolódnak. A számítás iterációk (*supersteps*) sorozataként fogható fel. Minden egyes iterációban a számítási pontok ugyanazt a felhasználó által definiált függvényt hajtják végre. Feldolgozzák az előző iterációban számukra küldött üzeneteket, megváltoztathatják saját és kimenő élek értékét, éleket törölhetnek és újakat definiálhatnak, üzenetet küldhetnek más pontoknak – nem feltétlenül csak a szomszédaiknak –, emellett szavazhatnak arra

vonatkozólag, hogy a teljes számítás a következő lépésben leálljon-e. A program akkor fejeződik be, ha minden pont kivétel nélkül a leállás mellett szavazott. Ha egy pont a leállás mellett teszi le a voksát, a következő iterációban inaktívvá válik, ugyanakkor egy másik pont által küldött üzenet ismét aktívvá teheti. Egy egyszerű, maximumkeresési algoritmus látható a 3. ábrán.



3. ábra. Maximumkeresés Pregel rendszerben (forrás: [10]).

Itt egy-egy iteráció során minden pont megnézi, hogy az aktuális értéke nagyobb-e szomszédjai aktuális értékénél, amelyet azok az előző lépésben küldtek el. Ha nem, az aktuális értékét a legnagyobb értékű szomszédjával teszi egyenlővé, s ezt az új értéket elküldi a szomszédainak. Ellenkező esetben nem változtatja meg az aktuális értékét, üzenetet sem küld, hanem ehelyett a leállásra szavaz.

[4] cikkünkben a HADOOP és Giraph rendszerek teljesítményét hasonlítottuk össze egymással és a szekvenciális megközelítésen alapuló megoldások teljesítményével. Ezek rendre a MapReduce és Pregel modellek Apache cég által megvalósított implementációi. Az összehasonlításhoz három különböző, gráfokat jellemző mérőszámot használtunk: az irányított háromszögek számát, az erősen összefüggő komponensek méretét, valamint a Lineranket, ami a Pagerankhez hasonlóan pontok fontosságát méri. A Linerankre a következő fejezetben térünk ki bővebben. A módszerek, melyek e három mérőszám kiszámítására szolgálnak, jellemzően három, egymástól eltérő algoritmuscsaládba sorolhatók. A háromszögek számának kiszámításakor minden egyes pont esetében csupán szűkebb környezetét kell figyelembe venni. Az összefüggő komponenseket megtaláló módszerek majd mindegyike a mélységi bejárást használja, amely elosztott rendszerek esetében elfogadhatatlanul hosszú ideig tarthat, míg a Linerank értékek mátrix-vektor szorzás iteratív alkalmazásával kaphatók meg.

Megvalósításainkat valós és véletlen hálózatokon is teszteltük. Ez utóbbiak generálásakor az erdőtüz modell (fire forest) [16] egy általunk bevezetett módosítását használtuk, mivel az eredeti modell szerint létrehozott gráfok nem tartalmaznak irányított háromszögeket. A Linerank kiszámítása során kapott futási időket a 4. ábrán található táblázat foglalja össze. A kísérletben minden munkaállomás 2Gb belső memóriával rendelkezett, s vagy két 3,2 GHz-es I5 processzorral, vagy egy darab 3 GHz-es I5 processzorral futott. Az időegységek másodpercben értendők. A Hadoop-20 oszlop 20 számítási ponttal működő HADOOP

rendszer reprezentál. A Hadoop-i és Giraph-j oszlopok jelentése az előbbihez hasonló módon értelmezhető. Érdekes módon a szekvenciális megvalósítás minden esetben jobban teljesít, mint a HADOOP, holott az utóbbi esetében köztudomású, hogy a mátrix-vektor szorzás és annak iterálása hatékonyan implementálható, s mi is egy hatékonyan elismert módszerrel dolgoztunk. A Giraph azonban 10 és 20 millió ponttal rendelkező gráfok esetén már jobban teljesít.

	Nodes	Edges	Seq	Hadoop-20	Hadoop-40	Giraph-10	Giraph-17
web-Google	875713	5105039	36,184	1755,529	2192,617	155,281	176,961
wiki-Talk	2394385	5021410	51,434	1463,529	915,685	145,097	155,601
soc-LiveJ	4847571	68993773	67,41	2764,598	2973,812	748,551	537,495
forest10M	10000000	24024725	315,21	1873,53	2064,873	250,466	235,531
forest20M	20000000	48097267	840,701	2618,853	2533,859	566,999	338,821

4. ábra. Futási idők a Linerank értékek számítása esetén.

4. Köztesség vs. Linerank

A köztesség centralitási mérték pontok fontosságát méri, azt hivatott kifejezni, hogy ha az információ a hálózat tetszőleges két pontja között mindig a legrövidebb úton vagy utakon terjed, akkor egy-egy ponton a hálózatban terjedő összinformáció hány százaléka fut át. Formálisan:

$$v_{\text{köztes}} = \sum_{u,w} \frac{b_{u,v,w}}{b_{u,w}},$$

ahol u, v, w a hálózat pontjai, ahol u és w biztosan különböznek is egymástól, $b_{u,w}$ és $b_{u,v,w}$ pedig rendre az u és w pontok között futó legrövidebb utak számát, valamint ezen utak közül azok számát adja meg, amelyek a v ponton is keresztül haladnak. Élek esetén a mérték az iméntihez hasonlóan definiálható. A köztesség több gyakorlati alkalmazás során is igen hasznosnak bizonyult, kiszámítása azonban időigényes, $O(nm)$ lépésszámú – ahol n a hálózat pontjainak, míg m az éleinek számát jelöli –, emiatt nagyobb, milliós pontszámú gráfok esetén már nem alkalmazható.

A Google és IBM kutatói épp emiatt javasolták a betweenness egy lehetséges alternatívájaként a Linerank mértéket [5]. Ez ugyanúgy a pontokon (vagy éleken) áthaladó információ mennyiségét hivatott megragadni, s amiként a neve is mutatja a Pagerankkel erős rokonságot mutat. A Linerank számításakor első lépésben az eredeti G gráf $L(G)$ élgráfját (*line graph*) kell megszerkeszteni. Az élgráfban minden pont a kiindulási gráf egy-egy élét reprezentálja. Az e_1 élét reprezentáló pontból akkor és csak akkor megy él az e_2 -t reprezentáló pontba, ha G -ben e_1 végpontja egybeesik e_2 kezdőpontjával. A következő lépésben $L(G)$ pontjainak Pagerank értékét kell kiszámítani. Mivel ezek a pontok G -ben egy-egy élnek felelnek meg, a kalkuláció eredményeként minden ilyen élhez egy mérőszámot kapunk, ami hasonlóképpen méri az él fontosságát, mint amiként a Pagerank érték egy-egy pontét. Az utolsó lépésben minden G -beli pontra összegezzük a belőle induló és bele érkező élek iménti értékeit. Érdemes észrevenni, hogy az élgráf mérete az eredeti gráf méretéhez képest sokkal nagyobb lehet. A [5] cikkben megmutatták, hogy $L(G)$ szomszédsági mátrixa miként bontható fel két ritka mátrix szorzatára, melynek eredményeként konstans szorzótól eltekintve a

Pagerank értékek ugyanolyan gyorsan kiszámíthatók, mint az eredeti G gráf esetében. A cikkben ugyanakkor nem vizsgálták a két mérték, köztesség és Linerank, kapcsolatát.

[7] munkánkban ezt a hiányt igyekeztünk pótolni. [5]-ben csupán pontok esetében definiálták a Linerank mértéket, a köztesség ugyanakkor élekre is megadható, így, hogy a két mérőszám élek esetén is összehasonlítható legyen, egy-egy él Linerank értékét az őt reprezentáló élgráfbeli pont Pagerank értékeként definiáltuk. Első lépésként e két mértékpár valós és véletlen gráfokon vett Pearson és Spearman korrelációs együtthatóit vizsgáltuk; majd két algoritmus teljesítményének változását elemeztük a köztesség Linerank-re történő cseréje után. A véletlen gráfok generálásához a [18] cikkben bevezetett modellt alkalmaztuk. Ez olyan hálózatokat generál, ahol a közösségek mérete β exponensű hatványeloszlás szerint változik. A pontok fokszáma szintén hatványeloszlású γ exponenssel. Az iménti paraméterek mellett megadható még egy μ keverési paraméter is. Minden u pont éleinek $1-\mu$ hányada kell, hogy olyan pontba vezessen, amely ugyanahhoz a közösséghez tartozik, mint u . Mind a véletlen, mind a valós gráfok esetén úgy találtuk, hogy a pontokra vonatkozó változatoknál mindkét típusú korreláció igen erősnek bizonyult 0.95 és annál magasabb értékekkel, míg az élekre definiált változatok esetén a korreláció jóval gyengébbnek bizonyult.

A Pearson és Spearman korrelációk meghatározása mellett az élekre vonatkozó változatok kapcsolatát a Girvan-Newman közösségfeltáró algoritmus [11] segítségével is vizsgáltuk. Az algoritmusban egymás után töröljük az éleket a köztességük csökkenő értéke szerint szerint. Egy-egy lépésben a legnagyobb köztességű él elhagyása után a maradék éleken újra kell számítani a köztességi értékeket, s a következő lépésben már az így kapott legnagyobb köztességű élet kell törölnünk. Egy idő múlva a gráf részgráfokra esik szét. Egy-egy ilyen részgráfot tekintünk egy-egy közösségnek. Természetesen részgráfok később szintén részgráfokra eshetnek szét, így közösségek egy hierarchiáját kapjuk. Az egy-egy lépéshez tartozó közösségek szerinti felbontások kapcsolatát dendrogrammal ábrázolhatjuk. Legjobb felbontásnak a legnagyobb *modularitással* rendelkezőt választjuk.

A Girvan-Newman algoritmus egy változataként a legnagyobb köztességű él helyett a legnagyobb Linerank értékkel rendelkezőt töröltük, egyébként minden más ugyanúgy működött, mint az eredeti módszer esetében. A két eljárás teljesítményét a *normalizált kölcsönös információ* (*normalized mutual information*) [17] szerint vizsgáltuk. Ahogy ez az alacsony korreláció alapján várható volt a két algoritmus eltérő közösségeket eredményezett. Emellett a köztességet használó módszer jóval jobban teljesített, mint a Linerank értékekre támaszkodó.

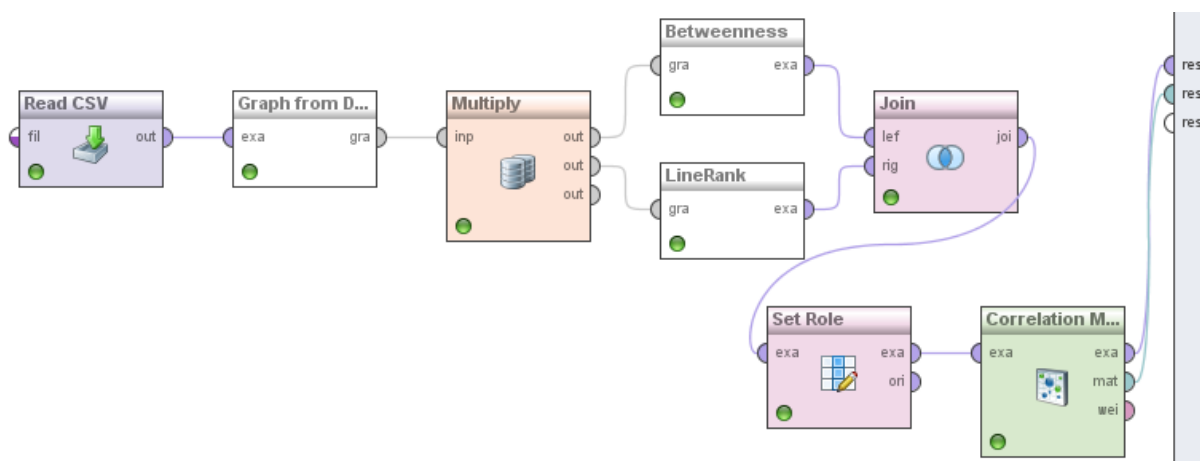
A pontokra vonatkozó változatok további összehasonlításához Boldi és munkatársainak egy kísérletét [1] végeztük el ismét, amiben azt vizsgálták, hogy pontok elhagyása miképpen befolyásolja a szociális és egyéb valós életből vett hálózatok szerkezetét. Minden egyes pont törlésekor természetesen a hozzá tartozó éleket is törölték. Előre rögzítettek \mathcal{G} számot ($0 \leq \mathcal{G} \leq 1$), s ha az elhagyott él száma $\mathcal{G}m$ fölé nő, nem töröltek több pontot – itt m a hálózat éleinek számát jelöli. A törlés sorrendjének megállapításához különféle mértékeket használtak, amelyeket egy pont és éleinek elhagyása után nem számítottak újra. A szerzők arra voltak kíváncsiak, hogy \mathcal{G} növekedésével miként csökken az egymásból él mentén elérhető pontpárok száma. Emellett vizsgálták még a régi és új gráf távolság eloszlásának divergenciáját. Úgy találták, hogy ezt leginkább a *relatív harmonikus átmérő változás* (*relative harmonic-diameter change*) ragadja meg. A kísérletek azt mutatták, hogy a pontok elhagyásának sorrendjét meghatározó mértékek közül a köztesség a legjobbak között

szerepelt, azaz köztesség szerinti csökkenő sorrendben vett törlés okozta változás a legjelentősebbek között volt.

Saját kutatásunk során úgy találtuk, hogy a korábban ismertetett modell által generált véletlen gráfok esetén (50000 pont) a kísérlet során a pontokra vonatkozó Linerank ugyanolyan jól teljesített, mint a köztesség. Valós hálózatok esetében azonban a köztesség adta sorrend általában felülmúlta a Linerank adta sorrendet, ugyanakkor ez sem „szerepelt rosszul”, a számítási időigény pedig jóval kevesebbnek bizonyult, mint a köztesség adta sorrendet használó törléseknél.

5. RapidMiner modul

Az előző fejezetekben bemutatott kutatások során implementált algoritmusok kódjait egy Java könyvtárba gyűjtöttük össze, amelyet aztán kiegészítő modulként a RapidMiner adatelemző szoftverbe is beillesztettünk. A fentiekén túl a modul két, hálózatok pontjainak hasonlóságát számoló módszert is tartalmaz. Az első, a *koszinusz hasonlóság*, a közös szomszédokon alapul [12]. A második definíciója rekurzív, két pont akkor tekinthető hasonlóknak, ha a szomszédjaik is hasonlóak. A részletek a [9] munkában olvashatók.



6. ábra. Köztesség és Linerank értékek szóródási ábráját elkészítő folyamat az Eopinions.com webhelyről származó hálózat esetében.

A modul eljárásainak alkalmazására egy példa a 6. ábrán látható. A folyamat egy valós életből vett hálózat esetén készíti el a köztesség és Linerank értékek szóródási ábráját. Az adathalmaz az Eopinions webhely felhasználóinak bizalmi hálózatát tartalmazza 75789 ponttal és 508837 éllel. A modulra vonatkozó további részletek a [8] cikkünkben olvashatók.

6. Köszönetnyilvánítás

Jelen kutatást részben a Magyar-vietnami TÉT 10-1-2011-0645 együttműködési pályázat, részben a FuturICT.hu nevű, TÁMOP-4.2.2.C-11/1/KONV-2012-0013 azonosítószámú projekt támogatta az Európai Unió és az Európai Szociális Alap társfinanszírozása mellett.

Irodalomjegyzék

- [1] Paolo Boldi, Marco Rosa és Sebastiano Vigna: Robustness of social and web graphs to node removal, *Social Netw. Analys. Mining* **3(4)** (2013), 829-842.
- [2] Wei Chen, Yajun Wang és Siyu Yang: Efficient influence maximization in social networks, *In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data mining, KDD* (2009), 199-208.
- [3] Jeffrey Dean és Sanjay Ghemawat: Mapreduce: Simplified data processing on large clusters. *In Proceedings of the 6th Conference on Symposium on Operating Systems Design & Implementation* **6** (2004), 10.
- [4] Péter Englert, Márton Balassi, Balázs Kósa és Attila Kiss: Efficiency issues of computing graph properties of social networks, *In Proceedings of the 9th International Conference on Applied Informatics* (2014), megjelenés alatt.
- [5] U Kang, Spiros Papadimitriou, Jimeng Sun és Hanghang Tong: Centralities in large networks: Algorithms and observations, *In SDM* (2011), 119-130.
- [6] David Kempe, Jon Kleinberg és Éva Tardos: Maximizing the spread of influence through a social network, *In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (2003), 137-146.
- [7] Balázs Kósa, Márton Balassi, Péter Englert és Attila Kiss: Betweenness versus linerank, *In Proceedings of the 6th International Conference on Computational Collective Intelligence Technologies and Applications* (2014), megjelenés alatt.
- [8] Balázs Kósa, Márton Balassi, Péter Englert, Gábor Rácz, Zoltán Pusztai és Attila Kiss: A basic network analytical package for RapidMiner, *RapidMiner World* (2014), megjelenés alatt.
- [9] E. A. Leicht, Petter Holme és M. E. J. Newman: Vertex similarity in networks, *arxiv:physics/0510143* (2005).
- [10] Grzegorz Malewicz, Matthew H. Austern, Aart J.C Bik, James C. Dehnert, Ilan Horn, Naty Leiser és Grzegorz Czajkowski: Pregel: a system for large-scale graph processing. *In Proceedings of the 2010 ACM SIGMOD International Conference on Management of data* (2010), 135-146.
- [11] M. E. J. Newman és M. Girvan: Evaluating community structure in networks, *Phys. Rev. E* **69(2)** (2004).
- [12] M. E. J. Newman: Networks: An Introduction, *Oxford University Press, Inc.* (2010).

- [13] Gábor Rácz, Zoltán Pusztai, Balázs Kósa és Attila Kiss: An Improved Community-based Greedy Algorithm for Solving the Influence Maximization Problem in Social Networks, *In Proceedings of the 9th International Conference on Applied Informatics* (2014), megjelenés alatt.
- [14] Yu Wang, Gao Cong, Guojie Song és Kunqing Xie: Community-based greedy algorithm for mining top-k influential nodes in mobile social networks, *In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (2010), 1039-1048.
- [15] Blondel, V. D., Guillaume, J., Lambiotte, R. és Lefebvre, E: Fast unfolding of communities in large networks, *In Journal of Statistical Mechanics: Theory and Experiment* **10** (2008).
- [16] Jure Leskovec, Jon Kleinberg és Christos Faloutsos: Graph evolution: Densification and shrinking diameters, *ACM Trans. Knowl. Discov. Data* **1(1)** (2007).
- [17] Santo Fortunato és Andrea Lancichinetti: Community detection algorithms: A comparative analysis, *In Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools* (2009).
- [18] Andrea Lancichinetti, Santo Fortunato és Filippo Radicchi: Benchmark graphs for testing community detection algorithms, *Phys. Rev. E* **78(4)** (2008).

Kísérlet a levelezős hallgatók költségmegtakarításának növelésére Sloodle alkalmazásával

An experiment to increase the cost-saving of part-time students by applying SLOODLE

Mucsics F. László

Károly Róbert Főiskola
mucsicsf@karolyrobert.hu

Absztrakt: E tanulmány célja, hogy a felnőttoktatásban vagy a felnőttképzésben tanulni vágyók számára idő és költséghatékony megoldást mutassak be, amely megkönnyítheti azoknak életét, akik távoktatás keretein belül szeretnének új ismeretekkel gazdagodni. A 21. században folyamatosan megújuló tudásra vagyunk kényszerítve, amely életre hívta az „élethosszig tartó tanulás”-t, mint életformát. Ennek beillesztése komoly kihívások elé állítja a tanulni vágyót és előfordulhat, hogy a megvalósításhoz kevésnek bizonyul a nap 24 órája. Ráadásul, ha a számukra megfelelő képzés lakhelyüinktől távol eső helyen történik, akkor jelentős költségekkel és idővesztéssel számolhatnak, valamint akár az is előfordulhat, hogy nem megoldható a tanórákon való részvétel. Az info-kommunikációs eszközök tárháza folyamatosan fejlődik, a magyarországi Internet előfizetések száma exponenciálisan növekszik, manapság már gyakorlatilag minden ember számára elérhető az Internet. A modern ember képzésében, akinek akár a telefonja segítségével a tenyerében elfér a fél világ, több ezer kilométeres távlatokat hidal át egyetlen hívással, miért pont az oktatás területén szabna gátat a fizikai távolság? Ennek a kérdésnek a megválaszolása motivál abban, hogy bemutassak olyan alternatív költséghatékony lehetőséget, mely egy lehetséges alternatív oktatási módszer irányába mutat. Hangsúlyozom, az általam végzett kísérlet egy alternatív lehetőség, nem kívánom vele helyettesíteni a hagyományos oktatási formát, sem a jól bevált tanítási módszereket. Kísérletemben egy olyan modern tanulási felületet érintek (Sloodle), ahol kombinálva használhatjuk a manapság legelterjedtebb internetes oktatási célra kifejlesztett rendszert (Moodle), valamint egy olyan virtuális világot (Second Life), ahol a tudást igazán felhasználóbarát és valóság-hű keretek között sajátíthatják el a nem nappali tagozatos hallgatók. Vizsgálom (a Gazdasági Matematika I. tárgyat) Sloodle-ben tanult hallgatók:

- az új rendszerhez való hozzáállását (életkor, informatikai telepítési nehézségek, elfogadás) és
- a hagyományos képzéshez viszonyított költségmegtakarítását.

Kulcsszavak: Moodle, Second life, Sloodle, költségmegtakarítás, valós idejű online oktatás

Abstract: The aim of this study is to present a relatively new educational method that makes time- and money-conscious solution more bearable for those who wish to continue their studies within the framework of distance education. The 21st century forces us to possess continuously renewable skills and expertise and that gave birth to the terminology “life-long learning”. To meet the requirements is really challenging and the 24 hours of the day does not always seem to be enough for the learner. In case of significant physical distance between the students’ home and place of education it is unavoidable to count with higher expenses or in more special cases it is simply insolvable to take part in consultations. The tools of info-communication technology have continuously been developing and the Internet subscriptions have been showing exponential growth in Hungary. This means that Internet is available for everyone in Hungary. The man of the 21st century can reach half of the world with his mobile phone – why would distance cause a problem in the field of education? This motivates me in finding a possible solution to show an alternative and cost-conscious way of education. I do emphasize that I am not intending to replace the traditional form of education. In my study I am presenting a learning environment with Sloodle, which is a combination of the virtual world of Second Life and of the learning management system of Moodle. I examine the students’ (involved in the experiment):

- attitude to the new educational system (age, installing difficulties, acceptance) and
- the cost-saving feature of the Sloodle education compared to traditional education.

Keywords: Moodle, Second life, Sloodle, cost-saving, real-time online education

1. Bevezetés

Az oktatásba belépő egyén azért tanul, hogy a képzés elvégzése során olyan ismeretekre tegyen szert, mely által a munkaerőpiacon jobban tud érvényesülni, így növelve a saját termelőképességét, illetve a munkájának piaci értékét. Az oktatásba való belépéskor akkori jövedelmének és szabad idejének egy részét feláldozza – a diplomaszerezés utáni – jövőbeli haszon reményében. Az oktatással és a képzéssel kapcsolatos költségek mintegy a tanuló saját magába investált befektetésnek tekinthetők. Az egyén hosszútávú elképzelése akkor valósul meg, ha a képzésének végeztével magasabb jövedelmet biztosító osztályba kerül, és a képzési költséget a képzés által megszerzett többletjövedelem meghaladja. Az oktatásba fektetett tőke-beruházás jövedelmezősége költség-haszon elemzéssel mérhető, melynek gyakori eszköze a belső megtérülési ráta kiszámításának alkalmazása (IRR). Ez a mutató a befektetés gazdaságosságra adja meg a választ [1], azaz hogy érdemes volt-e az egyénnek az oktatásba fektetni, vagy esetleg jobb lett volna azt a pénzüsszeget egy bankban elhelyezni.

Az elmúlt évtizedben a Károly Róbert Főiskola (innen: Főiskola) folyamatosan növekvő figyelmet fordított a nem nappali tagozaton tanuló hallgatók oktatásba fektetett tőkéjének minél hasznosabb megtérülésére. 2010. októberében megalakult az E-learning Módszertani Egység (innen: ELME), mely fő feladatának tekinti az informatika robbanásszerű fejlődésének köszönhető elektronikai eszközök és oktatási módszerek bevonását az oktatásba. Az ELME megalakulásakor létrehozott egy, azóta is folyamatosan fejlesztett Moodle oktatási keretrendszert (mentor.karolyrobert.hu), mely oktatási platformon a Főiskola hallgatói a tantárgyaikkal/kurzusaikkal kapcsolatos (1. ábra):

- információt oszthatnak meg (fórum),
- vizsgaidőpontokról tájékozódhatnak (adminisztráció, csoportok) és
- az oktatói lelkesedéstől függően a tankönyvön túlmutató tananyagokat szerezhettek (lecke, önellenőrző teszt, videó, fogalomtár).

The screenshot shows a Moodle course interface for 'Mátrixaritmetika'. The main content area lists various exercises and solutions for matrix operations. On the right, there are mathematical diagrams and equations, including a matrix equation $A \cdot B = C$ and a diagram illustrating 'Elementary Row Operations' with matrices $[A | I]$ and $[I | A^{-1}]$. Below this, there is a general matrix $A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$.

1. ábra. Gazdasági matematika II. kurzusának egy részlete ([2])

Az ELME folyamatos fejlődést követve további hallgatóbarát megoldásokat keres a hallgatói költségek csökkentésére. E tanulmány egy ilyen újabb megvalósításról, illetve annak hallgatói megítéléséről számol be.

2. A vizsgálat és céljai

A nem nappali tagozatos hallgatók konzultációs időpontjai a szorgalmi időszak heteinek a végére, péntekre, szombatra esetleg vasárnapra esnek. E tervezés oka az, hogy a nem nappali tagozaton tanuló diákok zöme családos, már munkahellyel rendelkező személyek, akik hétvégjüket tudják használni önmaguk szakmai gyarapításra. Hallgatóink különböző távolságokról utazva érkeznek a tömbösített, általában 8*45 perces tanórákra. A kísérlet keretében a 2013/14-es tanév őszi félévében az ELME lehetőséget biztosított egy tantárgy, a Gazdasági Matematika I. online hallgatására. A Second Life (innen: SL) környezetben egy virtuális tanteremet létrehozva a hallgatók közül a vállalkozóbb kedvűek otthonukból, munkahelyükről követhették az előadást. Ez az oktatási módszer nem új, több felsőoktatási intézet is kísérletezett a Sloodle (SL + Moodle) oktatás bevezetésével. Nemzetközi kitekintést végezve elsők között kell megemlíteni a Stanford Egyetemet, a Massachusetts Institute of Technology-t, valamint a Michigan Technological University-t. E három egyetem tekintélyes pénzüsségeket fektet be az online tanításba, jelen vannak a SL-ban is. Utóbbi, diákjai segítségével alkotta meg a „Campus of Hope” (=a Remény kampusza) virtuális egyetemvárost.



2. ábra. Campus of Hope – konferencia (saját szerkesztés)

A hazai viszonylatban az ELTE (Ollé János) és a Debreceni Egyetem (Kristóf Zsolt) intézményeknek voltak/vannak Sloodle oktatási kísérletei, melyek elsősorban, mint új oktatásmódszertani lehetőségként kerültek lebonyolításra.

A Főiskola oktatási kísérlet keretében 3 alkalommal (3 szombat) az elsőéves hallgatók a Főiskolára való utazás nélkül, virtuálisan vehettek részt a Gazdasági Matematika I. előadásokon. A beiratkozáskor történt egy tájékoztató, melyben a tárgyat hallgató diákok két csoportra lettek bontva aszerint, hogy a hagyományos vagy online módon vesznek részt a Gazdasági Matematika I. konzultációs órákon. A kísérlet lebonyolítása után egy a hallgatók által kitöltött kérdőív kiértékelésével arra kerestem a választ, hogy utólag milyen az online

tanításban részt vett hallgatók hozzáállása az új lehetőséghez, illetve mennyi költséget takarítottak meg a hagyományos tanuláshoz képest.

3. A vizsgálat menete és az eredmény

Az online résztvevő diákok számára létrehozásra került egy tanterem, melynek a kivetítőjére lett helyezve a Főiskola Moodle felülete. A hallgatói avatárok regisztrációját az avatárok pozicionálása követette, ezáltal a hallgatói belépéskor a diákok mindig erre a virtuális helyre kerültek. A hallgatói oldalnak a CD-n magkapott kliensprogram telepítése volt a feladata, majd a kapott felhasználónév-jelszó párossal állt módjukban belépni. Az adott napokon, a regisztrált diákok belépése után megvalósult a virtuális tanítás.



3. ábra. Az online tanítás kintről nézve (saját szerkesztés)

A Gazdasági Matematika I. tárgyat hallgató elsőéves tanulók összlétszáma 39 fő volt. Ebből 21 fő volt a kurzust online formában tanulók száma, mely hallgatói csoportbak az átlagéletkora 32,3 év. A csoport 95%-ának az első online órája volt ez, soha nem vett még részt hasonló programban. Szintén 95%-a otthonából követte az eseményeket, míg a maradék 5% a munkahelyéről. A saját gépen elvégzett telepítési feladattal 61%-uk gond nélkül megbirkózott, míg 24%-uknak segítséget kellett kérni, míg 5%-uk csak nagyon nehezen az utolsó pillanatban tudta megoldani a telepítési feladatot. Továbbá, egy 7 fokozatú Likert-skálán a hallgatók választ adtak mennyire passzol ez az oktatási forma a személyiségükhöz. Három változó felhasználásával (életkor, telepítési nehézségek, az új oktatási forma elfogadása) klasztereket hoztam létre, és az alábbi 6 klasztert kaptam:

1. *Rajongók az „érettebb” korosztályból* – e csoport átlagéletkora 51 év, az informatikában nem jártasak, kimondottan problémás volt számukra a telepítés, de az eredmény mindent kárpótolt, érdeklődésük töretlen.
2. *Mérsékeltlen érdeklődők* – életkoruk átlagosan 34 év, nincs ellenükre ez az oktatási forma, de a nem lelkesednek az új rendszerért.
3. *Elfogadók az „érettebb” korosztályból* – átlagéletkoruk 52 év, apróbb nehézségekbe ütköztek a telepítés során, elfogadják az új módszert, de nem lelkesednek érte (messze nem annyira, mint az 1. csoport)

4. *Lelkes fiatalok* – átlagéletkoruk 27 év, nem jellemzők a telepítési nehézségek és elfogadják az új rendszert, mint lehetséges tanulási formát.
5. *Y generáció* – éppen érettségizettek korosztálya, informatikát kedvelő csoport, nincsenek telepítési nehézségek, szerintük ez az oktatási forma a jövő.
6. *Technológiai megszállottak* – magas életkoruk (57 év) mellett, nincs telepítési nehézségük és lelkesednek az oktatási formáért. (Ez a csoport mindösszesen 1 főt tartalmaz, aki egy középiskolában rendszergazda, és rajong az új technológiai megoldásokért.)

A kísérlet végeztével a másik céloom az volt, hogy az egyéni költségmegtakarításokat is vizsgáljam. A kérdőívben forintosítva is rákérdeztem arra, hogy mennyivel került kevesebbe az online tanulásban részt vett hallgatóknak az, hogy nem kellett a Főiskolára utazniuk. Az átlagos, alkalmankénti megtakarítás 5.900 Ft volt, ami a 3 napos konzultáció esetén 17.700 Ft.

4. Összefoglalás

Megállapítható, hogy az új oktatási módszer alkalmazása jelentős mértékben csökkentheti a hallgatók oktatásba fordított befektetéseinek a mértékét. A 3 alkalommal átlagosan 17.700 Ft-ot spóroltak a hallgatók, ami nem kis összeg, különösen, ha figyelembe vesszük, hogy a legalacsonyabb tandíj mértéke 28.000 havonta. Ha nem csak egy tantárgy esetén lenne online konzultációra lehetőség az a hallgató indirekt költségeit jóval nagyobb mértékben csökkenthetné, ami így a belső megtérülési rátát növelhetné. Meg kell ugyanakkor említeni, hogy az idősebb hallgatók esetében kimondottan nehézkes volt az informatikai problémák leküzdése.

5. Köszönetnyilvánítás

Köszönetem fejezem ki a Főiskola Informatikai Osztályán dolgozó Fenyvesi György és Futó Zoltán kollégáimnak, akik informatikai szakértelme lehetővé tette a Sloodle oktatás megvalósítását. Szintén köszönetemet fejezem ki Németh Renáta végzős hallgatóknak, aki nagy akarással és még nagyobb türelemmel tartotta a kapcsolatot az online csoport tagjaival.

Irodalomjegyzék

[1] Júlia Varga: Oktatás-gazdaságtan, Közgazdasági Szemle Alapítvány, Budapest, 1998.

[2] mentor.karolyrobert.hu

Eredmények a több utas kommunikációs technológiák kutatásában

Research results on the multipath communication technologies

Almási Béla^a, Harman András^b

^aDebreceni Egyetem, Informatikai Kar
almasi.bela@inf.unideb.hu

^bIT Services Hungary/Debreceni Egyetem Informatikai Kar
Andras.Harman@t-systems.com

Absztrakt: Mintegy négy évvel ezelőtt kezdtünk egyeztetéseket arról, hogy az infokommunikációs hálózatok területén közös kutatást/fejlesztést alakítsunk ki. Több körös, hosszú egyeztetési folyamat eredményeként 2012-re sikerült egy ígéretes témát találni: A több utas kommunikációs rendszerek működésének vizsgálata IPv4/IPv6 környezetben.

A több utas kommunikációs technológia alapvetően abból a feltételezésből indul ki, hogy a ma elérhető hálózati eszközök (notebook, tablet, mobiltelefon) tipikusan több hálózati interfésszel rendelkeznek (WiFi, 3G, Bluetooth, ...). Ugyanakkor a klasszikus IP kommunikációs viszonyban csak egy interfész használható (az IP cím interfészhez kötött), így az interfészek kapacitását (pl. átbocsátóképesség növelés, vagy redundancia biztosításának céljára) nem tudjuk alkalmazni. A több utas kommunikációs technológiák ezen a területen kínálnak megoldásokat.

Az elmúlt másfél évben kutatók, PHD hallgatók és mérnök hallgatók bevonásával (nyolc fős kutatási csoportban) nagy erőforrásokat fordítottunk a terület kutatására. Az előadás bevezetőjében röviden áttekintjük a téma hátterét, a motivációs indíttatásokat s a területen már megtalálható eredmények alapvető koncepcióit. Az előadás legfőbb tartalmi elemét a kutatócsoport által létrehozott MPT több utas kommunikációs környezet (software library) ismertetése képezi. Speciális mérőlaborban és valós hálózati (business network) környezetben végzett mérési eredmények ismertetésével mutatjuk be milyen esetekben ad kifejezetten hatékony eredményt az MPT alkalmazása.

Kulcsszavak: több utas kommunikáció, multipath, MPT, MPTCP

Abstract: It was about four years ago that we began studies on research cooperation possibilities. After a long study process in 2012 we found an interesting research topic: investigating the multipath communication technologies in IPv4 and IPv6.

The basic assumption of the multipath communication is that many of the actually used networking nodes has got many network interfaces (WiFi, 3G, Bluetooth, etc.). The traditional IP communication technology uses only one interface (the Socket ID contains the IP address of the interface), so basically it is not possible to use multiple interfaces in one communication session. The multipath technologies search solution on this field.

In the past years we declared resources for research on the multipath communication technology. The most important result of the work is the MPT multipath communication software library. In this paper we introduce the working schema of the MPT environment, and we show some measurement results, which show the applicability of the MPT library.

Keywords: multipath communication, MPT, MPTCP

1. Bevezetés

Az IT Services Hungary és a Debreceni Egyetem Informatikai Karának kapcsolata évtizedes múltra tekint vissza. Az első kapcsolatok oktatási területen alakultak ki: az Informatikai Kar (az ITSH-val közösen kialakított) speciális képzési programok bevezetésével igyekezett a cég relatíve igen jelentősnek mondható IT szakemberigényét kielégíteni. A Train-IT képzések alapvetően operációs rendszerek és hálózatok területre fókuszáltak. A Train-IT képzések sikerességére alapozva a Debreceni Egyetem Informatikai Kara két speciális képzést alakított ki, melyet a Debreceni Egyetem alapvetően nem informatikai képzési területen tanuló hallgatói vehetnek fel (az eredeti szakjuk megtartása mellett), annak érdekében, hogy megszerezzék azokat az operációs rendszerhez és hálózatokhoz kötődő ismereteket, amelyek nélkülözhetetlenek ahhoz, hogy esetlegesen az ITSH alkalmazni tudja majd őket. A specializáció több éves bevezetése óta sikerrel működik.

Az oktatási kapcsolatoknál elért jó tapasztalatok kiváló alapot teremtettek arra, hogy az együttműködést gyakorlati kutatási területek felé is továbbvigyük/kiterjesszük. 2010-ben kezdődtek egyeztetések arról, hogy milyen területeken, milyen témában lehetne kifejezetten gyakorlati irányultságú kutatásokban együttműködni. A lehetséges kutatási területek kezdetektől a hálózatokra fókuszáltak: Multimédia kommunikáció, HD multimédia, IPv4/IPv6. Mintegy másfél-két éves szakmai egyeztetési munkálatok után kialakult egy olyan területspecifikáció, ami koncepcionálisan (a rendelkezésre álló erőforrás környezetet is figyelembe véve) reális alapokon kivitelezhetőnek tűnt: Olyan hálózati kommunikációs architektúrát kellene kialakítani/készíteni, melyben a kommunikációs viszony kevésbé érzékeny egy esetleges interfész leállására, illetve a különböző interfészek sávszélesség erőforrásait hatékonyan képes lenne aggregálni.

A cikk hátralévő részében az ezen a területen elért eredményeket tekintjük át, ill. foglaljuk össze röviden. A második fejezetben általánosan ismertetjük a több utas kommunikációs technológiákat. A harmadik fejezetben az MPT több utas szoftverkönyvtárral végzett tesztmérések eredményeit foglaljuk össze, s végül a konklúzióval zárul a cikk.

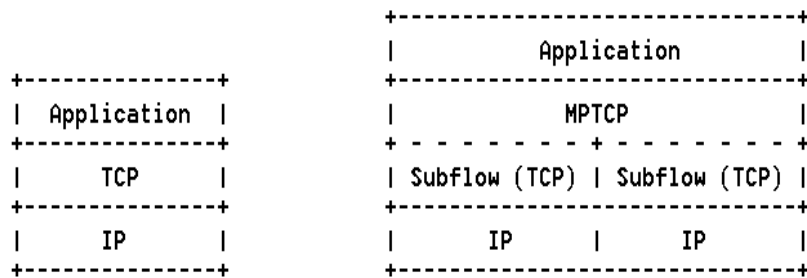
2. Több utas kommunikációs technológiák

A több utas kommunikációs technológiák tipikusan az OSI 3. rétege fölött működtethetők, bár meg kell jegyezni, hogy nagyon hasonló funkcionalitást találhatunk már L2-ben is: A link aggregációs technológiák (Etherchannel, LACP, IEEE 1905.1) alapvetően a link kapacitások összegzését szolgálják, hasonlóan a több utas kommunikációs technológiákhoz. Különösen figyelemre méltó az IEEE 1905.1 specifikáció (ld. [1]), ami akár különböző fizikai rétegbeli implementációk összegzését teszi lehetővé.

Mindemellett a több utas megoldás mind koncepcionálisan, mind technológiailag teljesen más környezetet jelent a magasabb rétegekben.

2013 januárjában jelent meg a transzport rétegben több utas TCP kommunikációt leíró MPTCP specifikáció [2]. Az MPTCP új koncepcionális hálózati architektúrát vezet be, melyben a korábbi transzport réteg funkcionalitását kettébontja: Az applikációk felé (a tradicionális TCP-vel kompatibilis) csatlakozást az MPTCP alréteg adja. Ez alatt kerülnek kialakításra a TCP alfolyamok („TCP Subflows”), melyek lehetőséget adnak a különböző interfészek és utak használatára (ld. 1. ábra). Az MPTCP kapcsolat („connection”) több út („multipath”) alkalmazásával lehetővé teszi a különböző interfészekhez kötődő erőforrások („throughput”) összevonását, aggregációját. Mindemellett redundanciát is biztosít: egy út

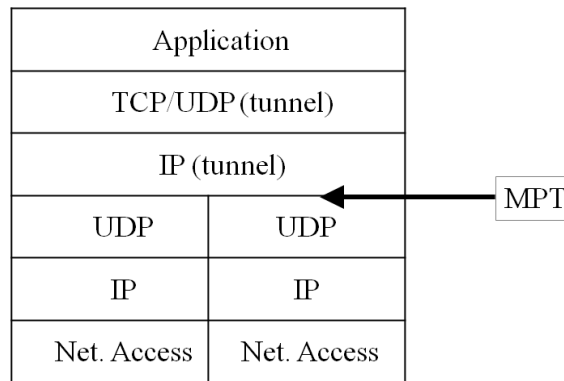
leállása nem jelenti az MPTCP kapcsolat végét, a rendelkezésre álló további útvonalakon a kapcsolat kommunikációja folytatható.



1. ábra. A hagyományos TCP/IP és az MPTCP/IP architektúra összehasonlítása

Több kísérleti mérés igazolja (pl. [3]), hogy az MPTCP számos esetben képes hatékonyan aggregálni a különböző utak kapacitását, átbecsátóképességét. Az MPTCP-hez kapcsolódóan azonban meg kell jegyezni, hogy a transzport rétegben működik, s kizárólag a TCP protokollra épülő applikációk tudják kihasználni a lehetőségeit, UDP protokollal nem működik. Továbbá jelenleg csak IPv4 implementáció érhető el Linux környezetben, IPv6 nem (ld. [4]).

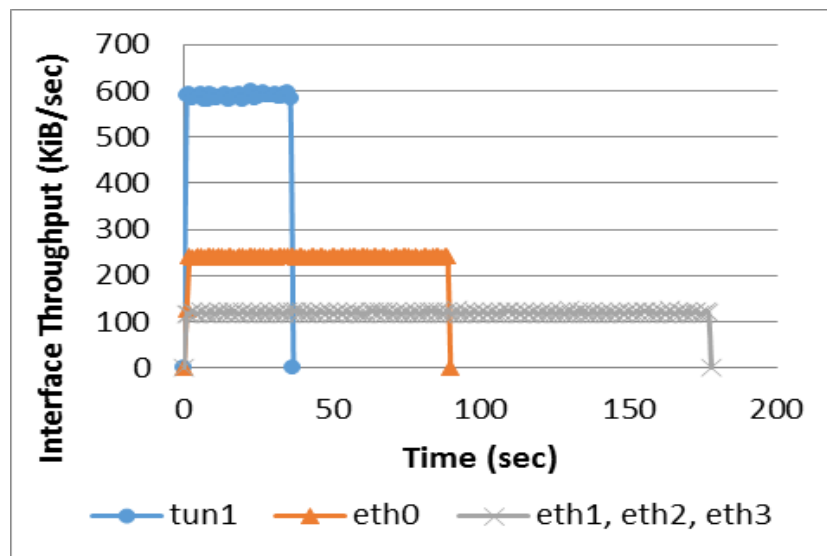
Kutatási együttműködés keretében 2012-ben indult el a Debreceni Egyetem Informatikai Karán a több utas kommunikációs technológiák hálózati rétegben való implementálásának vizsgálata. A kutatási munkák egyik legfontosabb eredményeként kialakult egy több utas Linux kommunikációs szoftverkönyvtár, mely lehetőséget biztosít a kommunikációs felek között bármilyen IP verzióval és bármilyen transzport protokollal a több utas kommunikációra. Az MPT környezet koncepcionális működési vázában (ld. 2. ábra) egy logikai interfész (tunnel interfész) kerül bevezetésre. Az applikációk a tunnel interfészt ugyan úgy használhatják, mint bármely fizikai interfészt. Azonban az applikáció által a tunnel interfészre küldött (IPv4 vagy IPv6 verziójú) IP csomagot az MPT szoftver kapja meg inputként, s a több utas környezetre felkonfigurált fizikai interfészek közül választva kerül továbbításra a csomag a fizikai interfészen. A továbbítás előtt egy új IP csomag (és UDP szegmens) készül, mely új IP csomag fejlécében már a fizikai interfészek adatai fognak szerepelni. A célállomásnál a fizikai interfészre beérkező IP csomag/UDP szegmens az UDP fejrészben szereplő MPT célra fenntartott portszáma alapján a vevő oldali MPT szoftverhez kerül. Az MPT szoftver „kicsomagolja” a feladó állomáson beágyazott tunnel interfészhez kötődő csomagot, s ezt továbbítja a vevő állomás tunnel interfészére, majd onnan a cél applikációhoz.



2. ábra. Az MPT több utas szoftverkörnyezet architektúrája

3. Mérési eredmények az MPT több utas szoftverrel

Az MPT környezettel számos mérést végeztünk. A mérések legfontosabb célja az útvonalak átbocsátóképességének (throughput) összegzési hatékonyságának vizsgálata volt. Az első mérések két út (szimmetrikus és aszimmetrikus sebességekkel) throughput aggregációját vizsgálták. Az eredmények (ld. [5]) azt mutatták, hogy a két út átbocsátóképességét 95%-os hatékonysággal összegezte az MPT környezet. Hasonlóan jó eredményeket láthatunk a [6] cikkben, de itt már négy út kapacitásának összegzésére. A 3. ábrán látható, hogy a throughput összegzés nemcsak átlagban tapasztalható, hanem időben egyenletes összegzést mutat.



3. ábra. 20MB fájltranszfer aggregáció négy úton (eth0: 2Mbps; eth1,eth2,eth3: 1Mbps)

A [7] cikkben az MPT szoftver aggregációs képességeinek határteljesítményéről olvashatunk: Az MPT képes 12 db 100Mbps sebességű útvonal aggregációjára. A legfrissebb mérési eredmények (ld. [8]) azt mutatják, hogy az MPT környezet a throughput aggregáció mellett lehetővé teszi, hogy egy kommunikációs viszonyon belül csomagvesztés-mentesen váltsunk WiFi-ről 3G-re, s vissza.

4. Konklúzió

A jelenleg széleskörűen használt „egy utas” kommunikáció egy lehetséges jövőbeli alternatívája lehet a több utas kommunikáció, ahol egy kommunikációs viszonyon belül több hálózati interfészen (Wifi, 3G, stb.) keresztül több útvonal kapacitását használhatjuk párhuzamosan. A cikkben áttekintettük a több utas kommunikációs technológiákat, különös tekintettel a kutatási együttműködés eredményeként létrejött MPT több utas kommunikációs szoftverkörnyezetre. A tesztlaborok mérései azt mutatták, hogy az MPT kiválóan összegzi az egyes utak kapacitásait, s jól alkalmazható arra is, hogy egy kommunikációs viszonyon belül backup lehetőséget biztosítson (pl. 3G interfészen keresztül).

5. Köszönetnyilvánítás

A cikk elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

Irodalomjegyzék

- [1] IEEE Standards Association: 1905.1-2013 - IEEE Standard for a Convergent Digital Home Network for Heterogeneous Technologies, 2013. Available <http://standards.ieee.org/findstds/standard/1905.1-2013.html> (Downloaded 04/04/2014)
- [2] A. Ford, C. Raiciu, M. Handley, O. Bonaventure: "TCP Extensions for Multipath Operation with Multiple Addresses"; IETF RFC-6824, 2013. Available: <http://tools.ietf.org/html/rfc6824> (Downloaded 04/04/2014)
- [3] C. Paasch, G. Detal, S. Barré, F. Duchêne, O. Bonaventure: "The fastest TCP connection with Multipath TCP"; ICTEAM, UCLouvain, Louvain-la-Neuve, Belgium; <http://multipath-tcp.org/pmwiki.php?n=Main.50Gbps> (Downloaded: 04/04/2014)
- [4] Multipath TCP Linux kernel implementation. Available at <http://www.multipath-tcp.org/> (Downloaded 06/06/2014)
- [5] B. Almási, Sz. Szilágyi, "Throughput Performance Analysis of the Multipath Communication Library MPT", Proceedings of the 36th International Conference on Telecommunications and Signal Processing (TSP 2013, ISBN:978-1-4799-0403-7), pp 86-90, Rome, Italy, 2013.
- [6] B. Almási, Sz. Szilágyi, "Multipath FTP and Stream Transmission Analysis using the MPT Software Environment", International Journal of Advanced Research in Computer and Communication Engineering. Vol. 2. Issue 11, pp 4267-4272, 2013
- [7] G. Lencse, Á.Kovács, "Testing the Channel Aggregation Capability of the MPT Multipath Communication Library", Proceedings of International Conference on Internet Technologies and Applications 2014 (ICITA 2014) (part of World Symposium on Computer Networks and Information Security 2014, WSCNIS 2014), Hammamet, Tunisia, Accepted to appear, 2014.

- [8] B. Almási, "A Solution for Changing the Communication Interfaces between WiFi and 3G without Packet Loss", Proceedings of the 37th International Conference on Telecommunications and Signal Processing (TSP 2014, ISBN: 978-80-214-4983-1), pp 73-77, Berlin, Germany, 2014.

Mobil hálózatok véges forrású modellezése spectrum renting és handover hívások használatával

A queueing model for Spectrum Renting and handover calls in Mobile Cellular Networks

Tamás Bérczes^a, János Sztrik^a, Jinting Wang^b, Xuelu Zhang^b, Fang Wang^b, Ádám Horváth^c

^aDebreceni Egyetem

{berczes.tamas, sztrik.janos}@inf.unideb.hu

^b Beijing Jiaotong University

jtwang@bjtu.edu.cn, zhang_xuelu@163.com, 12118402@bjtu.edu.cn

^cNyugat-magyarországi Egyetem

horvath@inf.nyme.hu

Absztrakt: Az elmúlt időkben a mobil hálózatok terheltségének növekedésével párhuzamosan egyre növekedett a téma kutatásának fontossága is. Jelen publikáció keretében mobil hálózatok egy véges forrású modelljét mutatjuk be spektrum bérlés használatával. A modell elkészítésénél véges forrású rendszert használtunk, mely a hálózatot használó populációt szemlélteti. A modellünk figyelembe veszi az úgynevezett handover hívásokat is, ami a cellák között mozgó, éppen hívást bonyolító felhasználókat jelenti. A numerikus eredményeket a MOSEL-2 programcsomag segítségével számítottuk ki. Külön vizsgáltuk spektrum renting esetében az hasznot a frekvenciák bérlési költségének figyelembevételével is.

Kulcsszavak: mobil hálózatok, véges forrású rendszerek, visszatérési rendszerek

Abstract: Following the common practice applied in the theory of retrial queues for performance evaluation, this paper proposes a finite-source retrial queueing model to consider spectrum renting and handover calls in mobile cellular networks, in which service providers may rent each other's unutilized frequency bands. The model incorporates switching servers on and off in groups, considers new and handover calls. We present a novel way to take into account the blocking probability of new and handover calls.

Keywords: mobile cellular networks; retrial queues; finite-source

1. Bevezetés

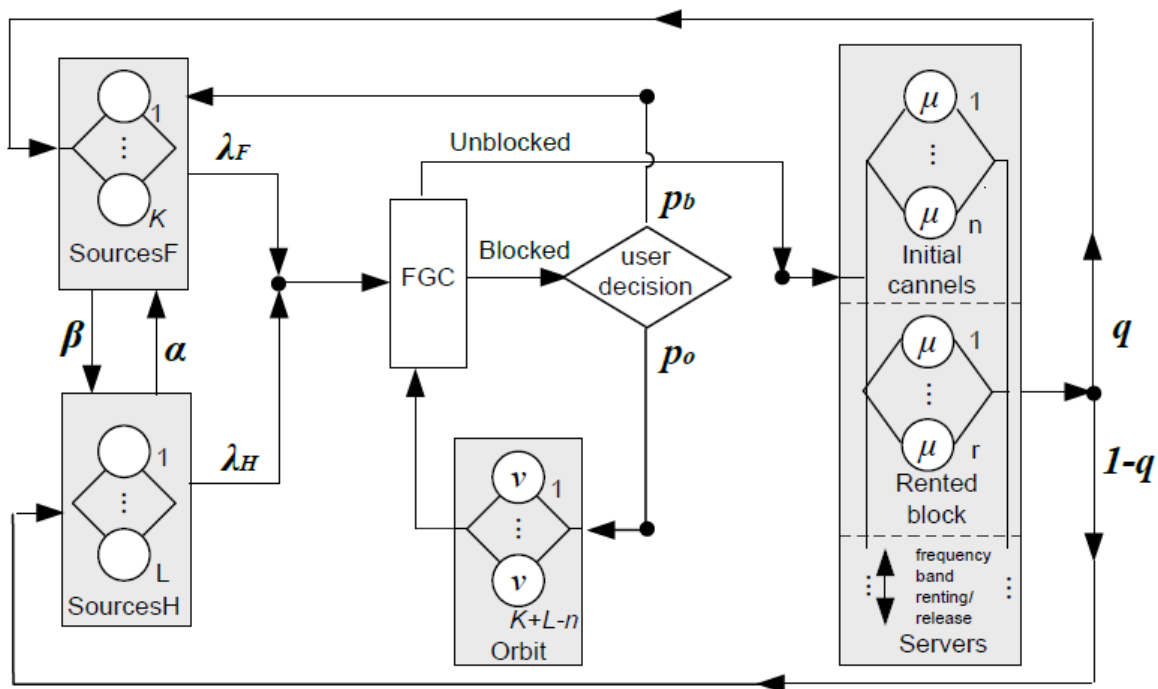
Az elmúlt időkben a mobil hálózatok terheltségének növekedésével párhuzamosan egyre növekedett a téma kutatásának fontossága is. Nagyon nehezen becsülhető előre egy adott mobil cella terheltségének alakulása. Újabb frekvenciák vásárlása nagyon költséges, így abban az esetben, ha ezek a frekvenciák kihasználatlanok, ezek felesleges extra költséget jelentenek plusz bevétel nélkül. Ellenkező esetben, ha nincs elegendő frekvencia a bejövő hívások egy része elutasításra kerül, ami a felhasználók elpártolását eredményezheti. Egy lehetséges megoldás a probléma kezelésére az úgynevezett spektrum bérlet használata.

A spektrum bérlet használatának alapötlete Mitola [1] cikkében található meg először. A spektrum bérlet lényege, hogy magas terheltség esetén, amikor a használatban lévő frekvenciák száma elér egy bizonyos előre definiált limitet, akkor a rendszer operátora megpróbál újabb frekvenciákat bérelni. Sikeres bérlet esetén, ha a rendszer terheltsége csökken és a szabad frekvenciák száma egy szinten előre definiált érték alá esik, a bérelt frekvenciát visszaadjuk. Így ideiglenesen megnövelhető a cellánk kapacitása [5].

Jelen kiadvány az ELEKTRO 2014 konferencián bemutatásra került és megjelent eredmények egy rövid kivonata. A részletes eredmények és elemzések a Bérczes és Horváth cikkben található [5].

2. Rendszermodell

Az 1. ábra a rendszer modellünket szemlélteti. A modell két véges forrást tartalmaz: a SourcesF az újonnan érkező hívások generálására alkalmas véges forrást tartalmazza, míg a SourcesH a handover hívások kezdeményezésére használt véges forrás.



1. ábra. Rendszermodell

Egy adott cella szempontjából a beérkező hívásokat két csoportba soroljuk:

- Új hívás: a hívás kezdeményezése a cellában történik. Az 1. ábrán a SourcesF jelöli az új hívások generálására alkalmas véges forrást. Minden felhasználó a SourcesF forrásból λ_r intenzitással generál új hívást.
- Handover hívás: Az eredeti hívás egy másik cellában indult, viszont a felhasználó mozgásának következtében a hívó cellát vált. Ilyenkor biztosítani kell a szolgáltatónak hívás folyamatoságát. Az 1. ábrán a SourcesH tartalmazza a handover hívások generálására alkalmas felhasználók véges forrását. Minden felhasználó a SourcesH forrásból λ_H intenzitással generál handover hívást.

A handover hívások védelmében az FGC (Fractional Guard Channel) szabályt alkalmazzuk. Ha egy bejövő hívás jóváhagyásra kerül automatikusan elfoglal egy szabad csatornát. A hívás időtartama exponenciális eloszlású μ paraméterrel. A hívás befejezése után a használt csatorna felszabadul és a felhasználó visszakerül a SourcesF forrásba.

Ha a bejövő hívást a rendszer elutasítja, a felhasználónak két lehetősége van: i) p_o valószínűséggel csatlakozik az orbithoz, ahonnan ν intenzitással újra próbálkozik szabad csatornát kapni; ii) feladja a próbálkozást, visszakerül a SourcesF forrásba.

Do és társai két küszöbértéket használt a spektrum bérlés implementálására [2]. Jelen modellünkben mi is ezt a módszert választottuk. Amikor a szabad frekvenciák száma egy adott határérték t_1 alá esik, akkor a rendszer operátor megpróbál új frekvenciákat bérelni. A medellben p_r -el jelöljük annak a valószínűségét, hogy a bérlési próbálkozás sikeres volt. Így $p_f = 1 - p_r$ annak a valószínűsége, hogy a bérlés sikertelen volt. Ilyenkor az operátor ν_r intenzitással egészen addig próbálkozik a bérléssel, amíg sikeres nem lesz illetve amíg a szabad frekvenciák száma t_1 határérték alatt van.

A bérelt frekvenciákat visszaadjuk abban az esetben, ha a szabad csatornák száma nagyobb lesz mint a $t_2 + r$ határérték. Ilyenkor a rendszer egy μ_r paraméterű exponenciális felszabadítási idő után visszaadja a bérelt frekvenciákat.

A handover hívások elutasításának minimalizálására a modellünkben az FGC szabályt alkalmaztuk [3],[4],[6]:

- LFGC (Limited average FGC)

$$\beta_{i,j} = \begin{cases} 1 & \text{if } 0 \leq i \leq N_j - \lfloor g \rfloor - 2, \\ 1 - g + \lfloor g \rfloor & \text{if } i = N_j - \lfloor g \rfloor - 2, \\ 0 & \text{else.} \end{cases}$$

- NPS (Non prioritization Scheme), ahol $\beta_{i,j} = 1$

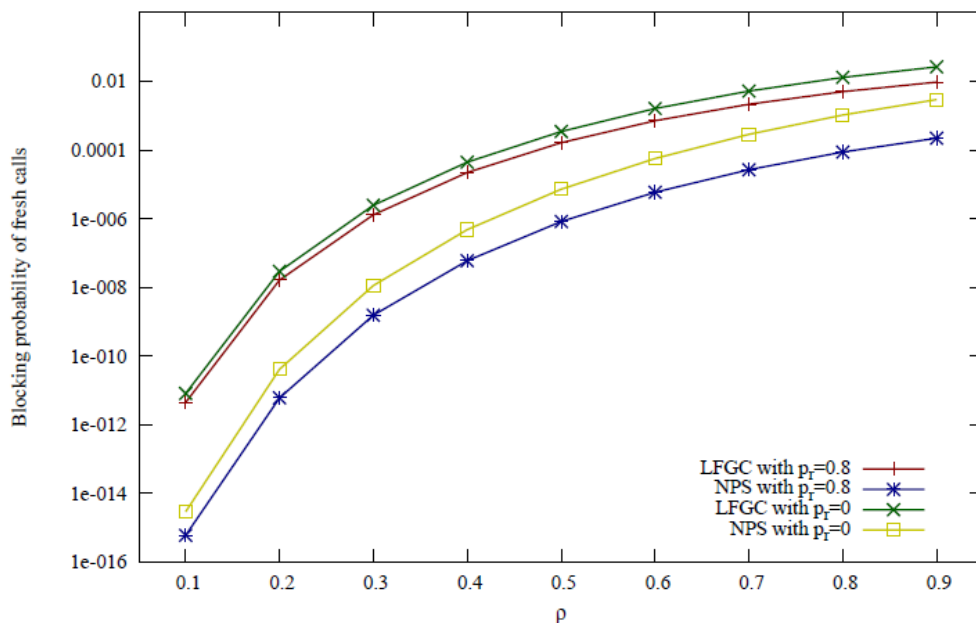
ahol a használt jelöléseket az 1. táblázat mutatja,

$I(t)$	A foglalt csatornák száma
$J(t)$	A bérelt frekvencia blokkok száma
$\beta_{i,j}$	Egy hívás elfogadása ha $I(t)=i$ és $J(t)=j$
n	A saját csatornák száma
N_r	A bérelt frekvencia blokkok száma
N_j	Az elérhető csatornák száma
g	A védett csatornák száma

1. Táblázat. Jelölések

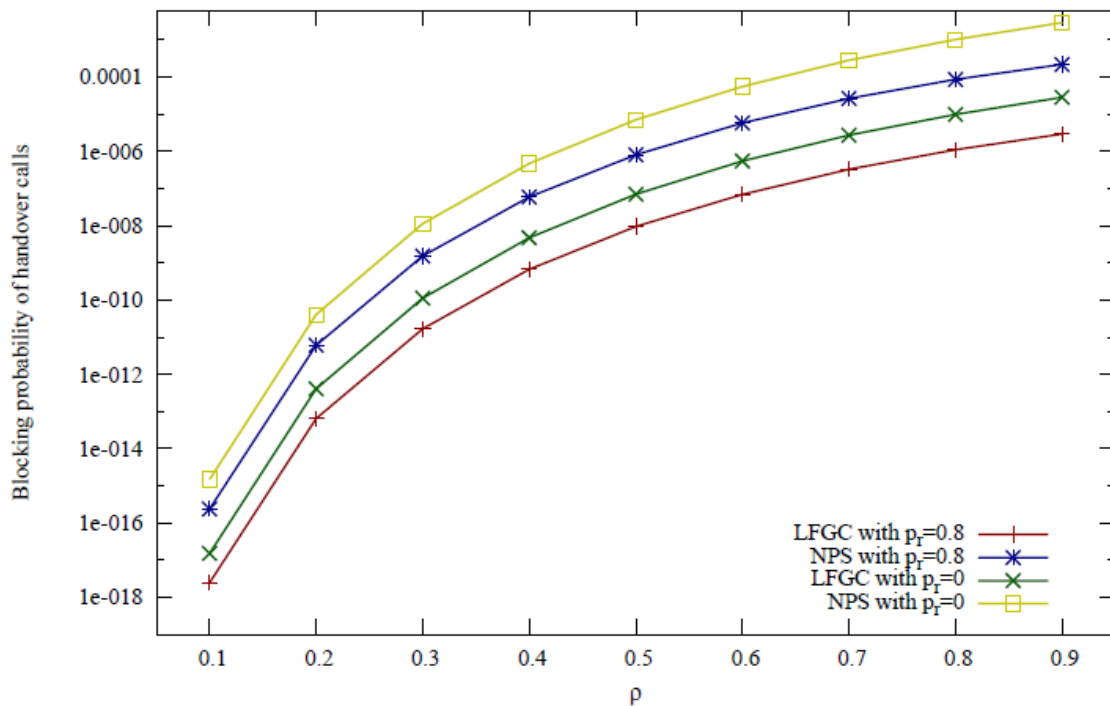
3. Numerikus eredmények

A modellt leíró folytonos idejű Markov lánc állapottere nagyon nagy, ezért az egyensúlyi egyenletek felírását és megoldását a MOSEL-2 program segítségével végeztük el [7]. A 2. táblázatban összefoglaltuk a használt paraméterek listáját. A következő grafikonokon láthatunk néhány numerikus eredményt.



2. ábra. Új hívások blokkolási valószínűsége a normalizált forgalmi intenzitás függvényeként

A 2. ábrán az új hívások blokkolásának valószínűségét látjuk a normalizált forgalmi intenzitás függvényeként. $p_r = 0$ azt az esetet mutatja, amikor nincs frekvencia bérlés. Látható, hogy az intenzitás növelésével a blokkolási valószínűségek nőnek, valamint szignifikáns eltérés figyelhető meg abban az esetben ha használunk frekvencia bérlést illetve ha nem. Szintén megfigyelhetjük, hogy az új hívások esetében nagyobb blokkolási valószínűséget kapunk, ha használunk védett csatornákat a handover hívások esetében.



3. ábra. Handover hívások blokkolási valószínűsége a normalizált forgalmi intenzitás függvényeként

A 3. ábrán a handover hívások blokkolásának valószínűségét látjuk a normalizált forgalmi intenzitás függvényeként. $p_r=0$ azt az esetet mutatja, amikor nincs frekvencia bérlés. Hasonlóan a 2. ábrához az intenzitás növelésével a blokkolási valószínűségek nőnek. Látható továbbá, hogy spektrum bérlés használatával a blokkolási valószínűségek csökkenthetőek. Szintén megfigyelhetjük, hogy a handover hívások esetében a blokkolási valószínűségek kisebbek, ha használunk védett csatornákat a handover hívások védelmében.

Normalizált forgalom intenzitás	ρ_0	[0.1..0.9]
Új hívások érkezési intenzitása	λ_F	
Handover hívások intenzitása	λ_H	
A SourceF mérete	K	100
A SourceH mérete	L	50
A védett csatornák száma	g	2.9
A visszatérés intenzitása	ν	1
A blokkonkénti csatornák száma	r	8
Saját csatornák száma	n	16
Átlagos hívási idő	$1/\mu$	1/53.22
Spektrum bérlési küszöbérték 1	t_1	3
Spektrum bérlési küszöbérték 2	t_2	6
Bérlés sikerességének valószínűsége	p_r	

2. Táblázat. Paraméterek

4. Köszönetnyilvánítás

Bérczes Tamás publikációt megalapozó kutatása a TÁMOP-4.2.4.A/2-11/1-2012-0001 azonosító számú Nemzeti Kiválóság Program – Hazai hallgatói, illetve kutatói személyi támogatást biztosító rendszer kidolgozása és működtetése konvergencia program című kiemelt projekt keretében zajlott. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Horváth Ádám publikációt megalapozó kutatása a TÁMOP-4.2.4.A/2-11/1-2012-0001 azonosító számú Nemzeti Kiválóság Program – Hazai hallgatói, illetve kutatói személyi támogatást biztosító rendszer kidolgozása és működtetése konvergencia program című kiemelt projekt keretében zajlott. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] J. Mitola III, “Cognitive radio for flexible mobile multimedia communications” in *Mobile Multimedia Communications, 1999. (MoMuC'99) 1999 IEEE International Workshop on*, pp. 3–10, IEEE, 1999.
- [2] T. V. Do, N. H. Do, and R. Chakka, “A new queueing model for spectrum renting in mobile cellular networks,” *Computer Communications*, vol. **35**, no. 10, pp. 1165–1171, 2012.
- [3] R. Guerin, “Queueing-blocking system with two arrival streams and guard channels,” *Communications*, IEEE Transactions on, vol. **36**, no. 2, pp. 153–163, 1988.
- [4] R. Ramjee, D. Towsley, and R. Nagarajan, “On optimal call admission control in cellular networks,” *Wirel. Netw.*, vol. **3**, pp. 29–41, Mar. 1997.
- [5] T. Bérczes and A. Horvath, “A Finite-Source Queuing Model for Spectrum Renting in Mobile Cellular Networks,” *Proceedings of ELEKTRO2014*, 2014
- [6] T. V. Do, “Solution for a retrial queueing problem in cellular networks with the fractional guard channel policy,” *Mathematical and Computer Modelling*, vol. **53**, no. 11–12, pp. 2058–2065, 2011.
- [7] P. Wüchner, H. de Meer, J. Barner, and G. Bolch, “A brief introduction to MOSEL-2,” in *Proc. of MMB 2006 Conference (R. German and A. Heindl, eds.)*, GI/ITG/MMB, University of Erlangen, VDE Verlag, 2006.

Gépek és gépelemek diagnosztikája, élettartam becslése gépi tanulás segítségével

Machine fault diagnosis and remaining useful lifetime estimation of machines with machine learning techniques

Deák Krisztián^a, Kocsis Imre PhD^b, Vámosi Attila^c

^aDebreceni Egyetem Műszaki Kar, Gépészmérnöki Tanszék
deak.krisztian@eng.unideb.hu

^bDebreceni Egyetem Műszaki Kar, Műszaki Alaptárgyi Tanszék
kocsisi@eng.unideb.hu

^cDebreceni Egyetem Műszaki Kar, Műszaki Alaptárgyi Tanszék
vamosi.attila@eng.unideb.hu

Absztrakt: A karbantartás jelentősége napjaink bonyolult gépei esetén egyre növekvő jelentőséggel bír. A karbantartás hatékonyságának fokozására számos informatikai eszközt alkalmaznak a mérnöki gyakorlatban, amely a diagnosztika és a hibaprognózis szerves részét képezi. Adatgyűjtés során a gép vagy gépelem monitorozása történik különböző szenzorokkal, mint rezgésmérőkkel, termoérzékelőkkel, akusztikus emisszió méréssel, zajméréssel. A jelek mintavételezése során fontos, hogy megfelelően hatékony technikával történjen, nagy jelentőséggel bírnak a wavelet transzformáción alapuló módszerek, mivel zajjal terhelt környezetben is képesek a hasznos jel eredményes szűrésére. A jel további feldolgozása időtartományban és frekvenciatartományban egyaránt lehetséges. Az időtartománybeli vizsgálatok során a jelek statisztikai paraméterei alapján történik a diagnosztizálás, mint a jelek csúcsértéke, átlagos értéke, ferdesége, lapultsága. A frekvenciatartománybeli vizsgálatok során leginkább a Fourier transzformáció használatos a jelfeldolgozásban. A mesterséges intelligencián alapuló módszerek a gépek és gépelemek diagnosztizálása, prognosztizálása során egyaránt hasznos. E célból a mesterséges neurális hálók, a fuzzy megközelítés, a neuro-fuzzy rendszerek és a tartóvektor gépek (SVM) meghatározó jelentőséggel bírnak. A kapcsolódó mérések validálásához olyan tesztpadok kerültek kifejlesztésre, amelyek segítségével szimulált körülmények között is lehetséges az elméletek alátámasztása.

Kulcsszavak: gép, karbantartás, diagnosztika, prognózis, gépi tanulás

Abstract: Maintenance is getting more important concerning especially sophisticated machines. To increase the efficiency of diagnosis several hardware and software are used in the engineering practice that are vital part of a complex system. General method is to monitor the physical parameters of the machines with sensors measuring acceleration, speed, temperature, noise etc. Appropriate sampling of the data is basically necessary so that they could make difference between useful signals from the background noise. After data acquisition, data processing can be executed both in time domain and frequency domain. In time domain, statistical parameters of the signals are frequently calculated, such as peak value, mean value, skewness, kurtosis. For frequency domain analysis, Fourier transform is widely applied. Machine learning methods help to decide whether the signal is normal or contains any harmful parts as the indicator of hidden machine faults. Artificial neural networks, fuzzy systems, hybrid neuro-fuzzy systems, ANFIS, support vector machines (SVM) have outstanding significance. To validate the theory special test-rigs are applied creating artificial conditions in order to measure and analyse the parameters one by one, separately that can ensure higher reliability and less faults.

Keywords: machine, maintenance, diagnostics, prognosis, machine learning

1. Bevezetés

A gépállapot becslés a maradék élettartambecslés, amely meghatározó jelentőségű a helyes működés szempontjából. A modern karbantartási szemlélet igényli, hogy csökkenjen a fenntartási költség, az üzemeltetés költsége, növekedjen a megbízhatóság, illetve, hogy a gép élethossza tervezhető legyen. Több modell került kidolgozásra az elmúlt években, az egyik a klasszikus megbízhatósági modellek, az állapotalapú modellek, és a kombinált modellek. A hibaprognózis szoros kapcsolatban áll a diagnosztikával, amely során a gép egy adott pillanatban lévő állapotát lehet megállapítani, forgó gépek esetén többnyire rezgésmérésen alapuló módszerek segítségével. Annak érdekében, hogy a diagnosztikai kiértékelés, illetve a hiba előrejelzés hatékonyságát fokozni lehessen, célszerű a gépi tanuláson alapuló módszerek alkalmazása, amely gyorsítja a kiértékelés, ugyanakkor minimalizálja az emberi tévedésből eredő hibákat. A neurális hálók, a fuzzy alapú megközelítés, a neuro-fuzzy rendszerek, a tartóvektor gépek (SVM) központi módszerek és a mérnöki gyakorlatban is fokozott jelentőséggel rendelkeznek. Jelen cikk a mesterséges neurális háló alapú megközelítésre fókuszál.

2. Diagnosztikai módszerek

A gépállapotvizsgálat egyik központi eleme a gördülőcsapágyak állapotának vizsgálata, mivel minden forgó gépben kiemelt fontossággal bírnak, A gördülőcsapágyak élettartama kifáradási jelenségekkel van összefüggésben, amely során a csapágygyűrűn, a gördüloelemeken a felületi repedések következtében kipattogzódás lép fel. A kísérletek vagy fárasztó vizsgálatokkal történnek, amely során a csapágyat tönkremenetelig járatják, vagy mesterségesen generált hibák okozta válaszjeleket elemzik, amely alapján mélyebb diagnosztikai elemzések végezhetők. Az utóbbi vizsgálatokhoz roncsolásmentesen bontható csapágyakat kell alkalmazni. Jelen vizsgálataink köréba tartozó csapágyak a mesterséges hiba előidézését követően ismételtelen működésképes állapotba hozhatók.

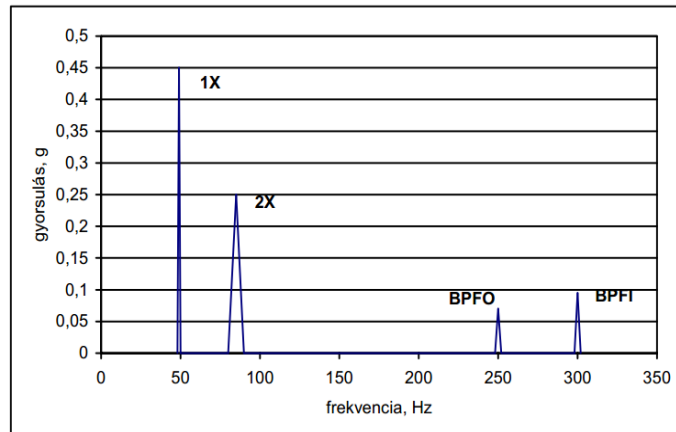
A fárasztóvizsgálatok során a rezgés értékek trendszerű változását mérik, illetve a hőmérsékletet, a kenőanyag összetétel változást. A folyamat hosszú ideig tart, a csapágy túlterhelésével, növelt fordulatszámmal, kenőanyag megvonásával a tönkremenetel gyorsítható, szimulált körülmények közötti mérések végezhetők. [7].

A legegyszerűbb csapágy rezgésdiagnosztikai módszer, amikor a rezgésparaméterek mérése során a mért értékek statisztikai jellemzői kerülnek meghatározásra. Ilyen értékek a négyzetes középérték (RMS), és a csúcstényező (Crest factor), a valószínűségi változók sűrűségfüggvénye, és továbbá a variancia, kurtózis, a ferdeség (skewness) és a lapultsági együttható. A hibátlan csapágyak rezgés gyorsulásának valószínűség sűrűségfüggvénye normál eloszlású (Gauss-eloszlás). A hibás csapágyé más valószínűség sűrűségfüggvénnyel rendelkezik [1]. Kutatók [6] arra a következtetésre jutottak, hogy a rezgésértékek további statisztikai momentumai is hatékonyan felhasználhatók állapotfelügyelet, diagnosztika és prognózis céljára. A hibátlan csapágyak mérések szerinti Kurtózis értéke hozzávetőlegesen 3. A 3-asnál nagyobb érték a lehetségesen előálló meghibásodás indikátora [2]. A módszer lokális hibáknál, mint a pontszerű kipattogzódás hatékonyan működik viszont hátránya, hogy nem lokális jellegű hibáknál, vagyis a gyűrűmenti kiterjedt meghibásodások esetében a Kurtózis értéke megmarad a 3 érték körül, amely alapján félrediaosztizálható a hibás csapágy állapota, a kezdeti meghibásodások sem minden esetben detektálhatók, ezért az ipari gyakorlatban a kurtózisfigyelés széles körben nem terjedt el. [3, 5, 4, 6]

A legáltalánosabban használt módszerként megmarad a gyors Fourier transzformáció (FFT) arra, hogy az időtartományból a frekvenciatartományba átlépve a csapágyak és egyéb

gépek, gépelemek hibafrekvenciái diagnosztizálhatók legyenek. (1. ábra) A csapágyak elemzésénél maradvány, ismerve a fordulatszámot, a csapágy típusát, így geometriai paramétereit, meghatározott összefüggések segítségével kiszámítható a belső gyűrű- külső gyűrű- gördülőelem- és kosárhibákból eredő hibafrekvenciák, amelyek a rezgésspektrumban jól körülhatárolható csúcsokként jelentkeznek.

További kutatások arra irányultak, hogy az idő-frekvenciatartománybeli vizsgálatok, mint az Short-Time Fourier Transform (STFT) és a wavelet transzformáció (WT) eredményesen használhatók zajjal terhelt környezetben is a gépek és gépelemek állapotának diagnosztizálására.



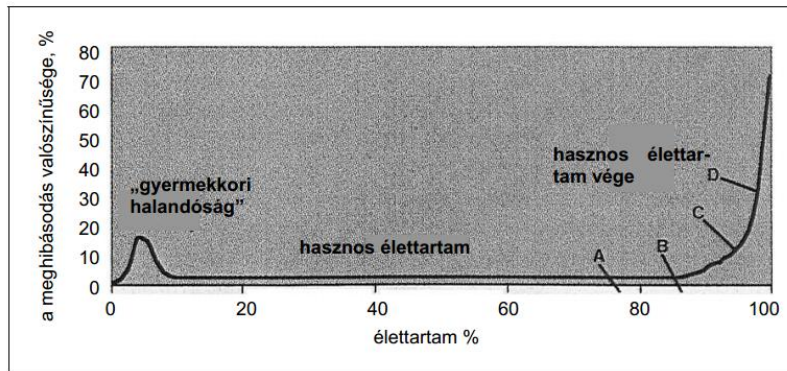
1. ábra. Fourier transzformáció után kapott hibaszpektrum [10]

3. Prognózis módszerek

A prognózishoz kapcsolódó karbantartási stratégiák a megelőző karbantartástól erednek, amelyek az állapotalapú felügyeleti rendszerekké alakultak (CBM). A CBM egyik fő eleme az adatok gyűjtése és tárolása, amelyek alapján a gép mindenkori állapota azonosítható és előrejelezhető. A prognózis csökkenti a géphibákból eredő váratlan leállások idejét, a karbantartási költségeket. Az alábbi fajtaikat különböztethetjük meg:

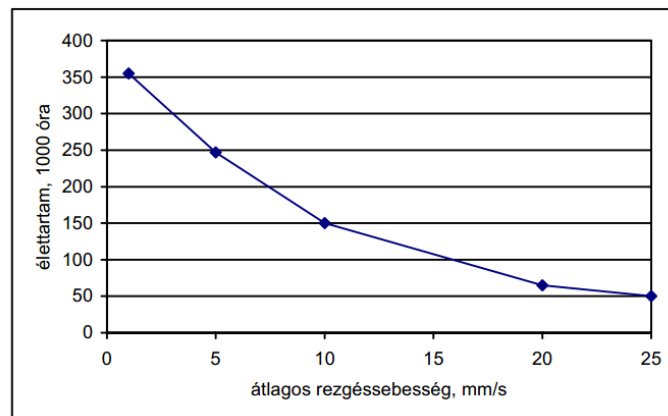
- Klasszikus megbízhatóság alapú prognózis
- Állapotfelügyeleten alapuló prognózis
- Integrált modellek

A klasszikus modell a mérési adatok matematikai eloszlását veszi alapul. Általában Poisson, Weibull eloszlás használatos, ez utóbbi alkalmas a gyakorlatban is előforduló „kádörbe” (2. ábra) modellezésére. A matematikai modellek alkalmasak továbbá repedésterjedés vagy hámlási folyamatok leírására, amelyek a gépek és gépelemek károsodása folyamán rendszeresen előfordulnak.



2. ábra. Élettartam görbe gépekre és gépelemekre [10]

A gépek károsodása közben előforduló repedésterjedés szimulálására alkalmas a Paris formula. Gyakran végeselem módszerrel alátámasztott eljárásokat alkalmaznak, amellyel a mechanikai folyamatok modellezhetők. A rezgésmérés általi adatokból meg lehet becsülni a tönkremeneteli folyamat sebességét, ezáltal a meghibásodás előrejelezhető. A rezgéssebesség és az élettartam közötti általános kapcsolat látható a 3. ábrán.



3. ábra. Gépelemek élettartam és a rezgéssebesség kapcsolata [10]

4. Mérések és gépi tanulás alkalmazása

A csapágyak mesterséges hiba okozta hibajelek diagnosztizálására, mérési céllal, kísérleti jelleggel a Debreceni Egyetem Műszaki Kar Gépészmérnöki Tanszékén csapágy vizsgáló tesztpad épült (4. ábra). A tesztpad 5 különböző típusú csapágy vizsgálatát teszi lehetővé változtatható fordulatszám mellett. A jelenlegi mérések során használt mérőeszközök és szoftverek listája az alábbi.

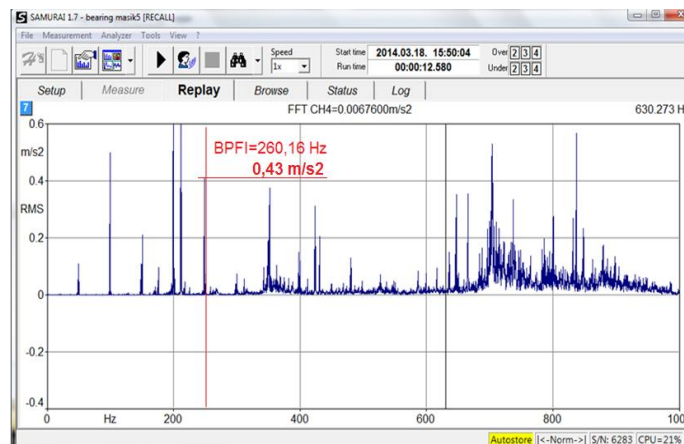
- PCB 603C01 rezgésyorsulás mérő szenzor, csavaros rögzítéssel
- National Instrument NI 9234 DAQ adatgyűjtő és NI CompactRIO mérőeszköz
- NI Labview szoftver, Sound and Vibration Measurement Suite
- TESTO 476 stroboszkóp fordulatszám méréshez
- Olympus BX61 mikroszkóp
- Soundbook zaj- és rezgésmérő rendszer
- SPSS statisztikai szoftver az időtartománybeli rezgésparaméterek meghatározásához
- Acer Aspire 5542G Notebook



4. ábra. A csapágyvizsgáló tesztpad

Jelen vizsgálat során 6206 típusú csapágy belső gyűrűjén lézeres megmunkáló géppel kis kiterjedésű mesterséges hiba került kialakításra. A hiba geometriai mérései szerint 658 μm átlagos átmérővel és 257 μm átlagos mélységgel rendelkezik, amelyen a gördülőelemek forgás közben áthaladnak és a rezgésspektrumban fellépő belső gyűrű hibát generálnak. A geometriai méréseket segítette a rendelkezésre álló Olympus BX61 típusú optikai mikroszkóp. A belső gyűrű hibafrekvencia (BPFI) 2880 1/ min névleges motorfordulatszám mellett került kiszámításra, amely 260,16 Hz-re adódott, ismelve a csapágy típusát, kapcsolódószöget, fordulatszámot, gördülőelemek számát és a vonatkozó csapágygyűrű átmérőt. Az 5. ábrán a rezgésmérés gyors Fourier transzformáltja látható, ahol 260,16 Hz környékén jól kivehető rezgéscsúcs tapasztalható, amely a belső gyűrű hibára utal.

A külső gyűrű hibafrekvencia (BPFO, ball passing frequency outer race), a kosár hibafrekvencia (FTF fundamental train frequency), a gördülőelem hibafrekvencia (BSF ball spin frequency) hasonló módon kiszámításra került a kapcsolószög, fordulatszám, és a gyűrűátmérők, gördülő elemek számának ismeretében. BPFO= 171,84 Hz, FTF= 19,2 Hz, BSF=112,32 Hz.



5. ábra. A méréssel kapott frekvenciakép 6206 tesztcsapágnál

A neurális háló (ANN) 5 bemeneti paraméterrel és öt kimenetivel rendelkezik, amelyek a lehetséges hibaállapot (lazaság, gyűrűhiba, gördülőelem hiba) indikátorai, ennek megfelelően a jó csapágy +1, a rossz csapágy -1 minősítést kap. Egy előre definiált topológiával és architektúrával rendelkező neurális hálózat került betanításra MATLAB Neural Network Toolbox segítségével. Felügyelt, ellenőrzött tanulást alkalmaztam, ahol ismertek a kimeneti targetek. Szakirodalmi kutatásaim szerint javasolt a Levenberg-Marquart backpropagation algoritmus, mint a Newton gradiens módszer alternatíva, mivel a leggyorsabb.

A neurális háló öt bemeneti paramétere: két időtartománybeli paraméter: átlagos rezgésyorsulás amplitúdó (A_{MEAN} , amplitude), kurtózis (K, kurtosis), három frekvenciatartománybeli paraméter: külsőgyűrű hibafrekvencia jel amplitúdó értéke (X_{BPFO}), belsőgyűrű hibafrekvencia jel amplitúdó értéke (X_{BPF0}), gördülőelem hibafrekvencia jel amplitúdó értéke (X_{BSF}).

A neurális háló három kimeneti paramétere: lazaság, gyűrűhiba (külső- vagy belsőgyűrű), gördülőelem hiba.

Habár hasonló méréseket korábban már végeztünk, ott a csúcstényező is figyelembe lett véve és az átlagos amplitúdó helyett a maximális amplitúdó szerepelt bemeneti paraméterként.

Az idő- és frekvenciatartománybeli paraméterekkel a neurális háló betanításra került és a validálást követően a diagnosztikai rendszer hatékonyságát az 1. táblázat összegzi, benne a tipikus hibákkal, amelyek a csapágy adott elemeinek meghibásodására jellemző.

Hiba jellege	Hiba-frekvencia	Neurális háló hatékonysága
Gördülő elem hiba	2*BSF, BPFI, BPFO	89,1 %
Gyűrű hiba	BPFI, BPFO	84,2%
Lazaság	Fs	91,8%

1.Táblázat. A csapágyhibákra utaló tünetek és a neurális háló hatékonysága

5. Összegzés

A gépi tanuláson, a mesterséges intelligencián alapuló diagnosztikai és prognózis rendszerek a gépek, gépelemek, csapágyak állapotvizsgálatát megkönnyíti. Jelen cikkben mesterséges neurális háló segítségével történt csapágyállapot diagnosztika, amely előrejelzésként is használható. A mesterséges neurális hálók működése a rendszer betanításán alapul, amely ennek eredményeképpen képes az automatikus hibafelismerésre. A jelenlegi mérési sorozatban a Matlab Neural Network Toolboxal épített neurális háló öt bemeneti paraméterrel és három kimenetivel rendelkezik, amelyek a lehetséges hibaállapot (lazaság, gyűrűhiba, gördülőelem hiba) indikátorai, ennek alapján képes diagnosztizálni. A validáláshoz próbapad épült, amely segítségével elfogadható hibafelismerési hatékonyság elérhető volt.

Irodalomjegyzék

- [1] Collacott Ra.: Mechanical fault diagnosis. *London: Chapman and Hall, 1977.*
- [2] Dyer, Stewart Rm.: Detection of rolling element bearing damage by statistical vibration analysis. *Trans ASME, J Mech Design 1978; 100(2):229–35.*
- [3] Gustafsson.: Detection of damage in assembled rolling element bearings. *ASLE Preprint 61-AM 3B-1. 16th ASLE, Philadelphia, PA, 1961. 39pp.*
- [4] Kim Py: A review of rolling element bearing health monitoring (II): preliminary test results on current technologies. In: *Proceedings of Machinery Vibration Monitoring and Analysis Meeting, Vibration Institute, New Orleans, LA, 26–28 June, 1984.*
- [5] Mathew J, Alfredson Rj: The condition monitoring of rolling element bearings using vibration analysis. *Trans ASME, J Vibr, Acoust, Stress Reliab Design 1984; 106:447–53.*
- [6] N. Tandon: A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings, *Tribology International, Vol. 32 (8) (1999), pages 469-480.*
- [7] S. Ericsson, N. Grip, E. Johansson, L.-E. Persson, R. S Jöberg, J.-O. S Strömberg: Towards automatic detection of local bearing defects in rotating machines. *Mechanical Systems and Signal Processing 19 (2005), pp. 509-535.*
- [8] Tóth L., Tranziens és kváziperiodikus folyamatok analízise az idő –frekvencia tartományban. Doktori értekezés.
- [9] Zhao, Chen, Guo, Li, Neuro-fuzzy Based Condition Prediction of Bearing Health *Journal of Vibration and Control, Vol. 15, No. 7, 1079-1091 (2009)*
- [10] Üzemfenntartási tevékenységek: forgógépek meghibásodásának korai előrejelzése. http://www.omikk.bme.hu/collections/mgi_fulltext/uzem/2005/06/0606.pdf

Gépjármű elektronika programozása RTOS rendszeren a fogyasztáscsökkentés jegyében

Programming of vehicle electronics in RTOS system to achieve better consumption

Háber István^a, Tamás László^b

^aPécsi Tudományegyetem, Pollack Mihály Műszaki és Informatikai Kar, Műszaki Informatika Tanszék
ihaber@gmail.com

^bPécsi Tudományegyetem, Pollack Mihály Műszaki és Informatikai Kar, Műszaki Informatika Tanszék
tamaslaszlo89@gmail.com

Absztrakt: A Pécsi Tudományegyetem Műszaki Karának Pollack Eco Team hallgatói műhelye 2011 óta fejleszt olyan gépjárművet amelynek fogyasztása 1400 km megtételére teszi alkalmassá 1 liter benzinnek megfelelő energiával. A jármű európai szintű versenyeken bizonyított, legutóbb a rotterdami Eco Marathonon ért el 7. helyezést.

A jármű egyszemélyes, 34kg önsúlyú, elektromos hajtású és az energiát a hajtáshoz egy hidrogéncella biztosítja. Azon kívül, hogy a tervezés és építés nagy körültekintéssel történt és a legmagasabb műszaki színvonalat képviseli (kiemelkedően alacsony légellenállás és gördülési ellenállás), az elektronikai rendszer és a szabályozó algoritmusok teszik lehetővé, hogy minden egység az elérhető legmagasabb határfokon működjön. A magas rendelkezésre állás érdekében autóiipari szabványok kerültek bevezetésre a rendszer tervezése során, így megfelel az AUTOSAR standardnak. Az elektronika elosztott rendszerű, STM 32 bites kontrollerek CAN buszon kommunikálnak egymással. Három fő komponens a vezérlőelektronika amely a hajtásért, az üzemanyagcella vezérléséért és a biztonsági feladatokért felel, a kormányelektronika ami a gombok kezelését és a kijelző működését vezérli, és egy külön adatgyűjtő rendszer amely a feszültség-, áramadatokat, a hőmérsékleteket, GPS pozíciót, stb. gyűjti egy SD kártyán, továbbá rádiós kapcsolaton keresztül valósít meg telemetriát.

Kulcsszavak: gépjármű elektronika, autosar, üzemanyagcella, határfok

Abstract: The University of Pécs, Faculty of Engineering, Pollack Eco Team student's workshop developed a vehicle which makes it suitable for a consumption of 1,400 km with one liter of petrol equivalent energy. The vehicle has proven Europe-wide competitions last time on the Eco Marathon Rotterdam it has achieved the 7th place.

The vehicle can carry a single person, its unladen weight is 34kg and the electric energy and for the propulsion is provided by a hydrogen cell. In addition to the design and construction which was made and of the highest technical standard is represented (very low aerodynamic drag and rolling resistance), the electrical system and the control algorithms allow each unit to operate on the highest efficiency. To ensure high availability automotive standards were carried into effect during the system design, so it satisfies the requirements of the AUTOSAR standard. The electronics works as a distributed system, STM 32-bit controllers communicate with each other CAN bus architecture. There are three main components: the control electronics (ECU) which takes care for propulsion, for the fuel cell control management and for security tasks, the steering electronics that manage the pushbuttons and the display operations, and a separate data acquisition system which collects the voltage, current data, temperature, GPS position, etc.. on an SD card, and implements telemetry link through radio.

Keywords: vehicle electronics, ecu, autosar, fuel cell, efficiency

1. Bevezetés

A Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Karán olyan ultraalacsony fogyasztású járműprototípusának fejlesztése folyik, amelynek elektromotoros meghajtásához saját fejlesztésű hidrogén-üzemanyagcella szolgáltatja az elektromos áramot.

A fejlesztések kipróbálásának, valamint a konkurenciával való közvetlen összehasonlítás és benchmarking terepül szolgálnak a különböző, fogyasztási járműversenyek. 2010-ben kezdődtek a fejlesztések egyetemi és kormányzati támogatással, és a fejlesztések teljesen a kar dolgozói és hallgatói közreműködésével lettek megvalósítva meg.

Ez idáig az alábbi versenyeken vett részt, és az eredményeink alapján folyamatosan fejlődni tudtunk:

- Shell Eco Marathon 2011 – Lausitz (GER),
- Shell Eco Marathon 2012 – Rotterdam (NED),
- Solar Race 2012 – Cartagena (ESP)
- Shell Eco Marathon 2013 – Rotterdam (NED)
- Shell Eco Marathon 2014 – Rotterdam (NED)

A legjobb elért eredmény hidrogéncellával a 2014-es versenyen

A versenyen a csapatoknak egy adott pályán kell tíz kört megtenniük legalább 25 km/h-s átlagsebességgel [1], legfeljebb 39 perc alatt ahhoz, hogy hivatalos, mért eredményt érhessenek el.



1. ábra Járműváltozatok

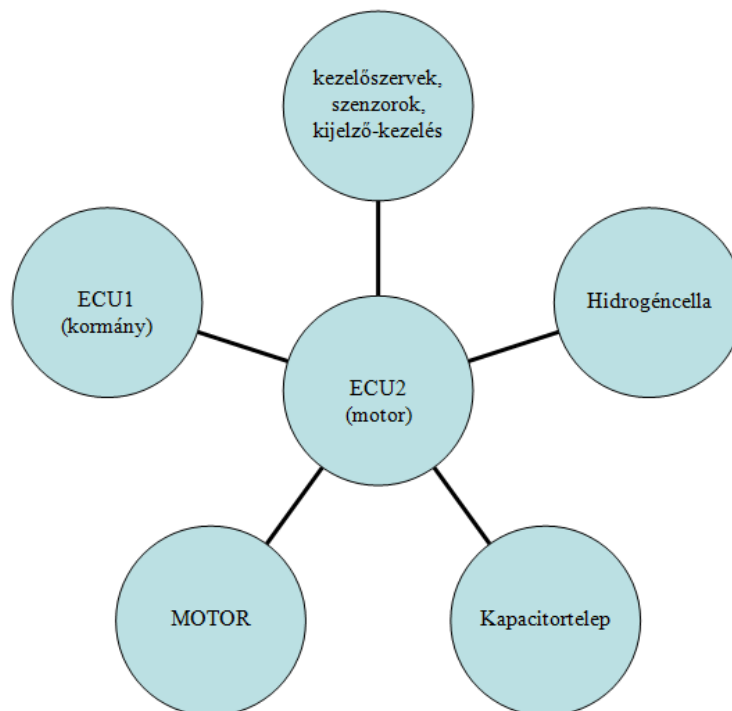
A jármű első változatai 2010-ben lettek tervezve, az alapvető cél az alacsony légellenállás megvalósítása volt. Alapvető ismeretként tudjuk, hogy a vízcsepp alakjának nagyon alacsony a légellenállása, tehát ebből indult ki a tervezés, majd váratlan fordulatként a gyilkos bálna (*Orcinus orca*) jelent meg a rajzokon mint formai elem. Ennek a vízi állatnak a felépítése modellezésnél mindvégig szem előtt lett tartva: az orr, a hát íve, de még a kerekek nyúlványai is a tőle kölcsönzött alakban nyilvánultak meg. A 2011-es Shell Eco Marathonon ezért TEAM ORCA-névvel állt rajthoz a PTE-csapata, amely a csapat-névben és az akkori járműfestésében is a gyilkos bálna szellemiségét hordozta.

Az ORCA 2012-ben továbbfejlesztett változata már az optimalizálás jegyében született. Numerikus szélcsatorna-vizsgálatokkal (CFD) lett továbbfejlesztve a forma, amely ezek után már kevésbé hasonlított a csapat névadójára, de a kiindulási alap miatt és a természet előtt tisztelegve meg lett tartva az ORCA név az jármű esetében, míg a csapat a Pollack Eco Team nevet kapta. A légellenállási tényezőt (C_d) sikerült majdnem 0,03-al 0,056-ra csökkenteni azzal, hogy változtatások lettek az íveken és közelebb lett helyezve a test a talajhoz. A hajtás a korábbi, belsőégésű motor helyett hidrogéncellára és elektromotorra lett cserélve, amely számos új kihívást jelentett. Az új járműkarosszéria a repülőgépiparban minősített karbonkompozit-anyagból lett elkészítve azért, hogy a jármű tömege oly módon legyen a lehető legkisebb szinten tartva, hogy a merevsége ne csökkenjen.

A karosszéria mechanikai tulajdonságai a FEM (Finite Element Method) módszerével lettek vizsgálva, így már a tervezési fázisban megadtuk a karbon-kompozit műanyagokra jellemző rétegrendet és így a jármű tömege valóban alacsony lehet. [2- Mechedu] Így történt hogy a korábbi 51 kg-os járműtömeget sikerült gondos tervezéssel 32 kg-ra csökkenteni.

A 2012-ben épített járműnél az aerodinamikai fejlesztések (CFD szimulációk) voltak a legfontosabbnak, 2013-ban ugyanennek a formának a tömege lett optimalizálva (FEM módszerek [2]), míg a 2014-es versenyre az elektronikai fejlesztések kaptak a legnagyobb hangsúlyt.

Természetesen a korábbi versenyeken is szükséges volt vezérlőelektronika alkalmazása, de az csak a legszükségesebb feladatokat látta el az akkor elérhető legalapvetőbb módon.



2. ábra Az elektronika blokkvázlata

A működési elv nagyvonalakban tehát a következő: A hidrogéncella termeli az elektromos áramot melyhez palackos hidrogén szolgáltatja az üzemanyagot. A hidrogéncella teljesítménye a terhelés függvénye (névleges: 500W). Az ultrakapacitor telep (70 Farad) a gyorsításokkor jelentkező csúcsterhelést szolgálja ki. Egy 200 W-os keféss DC motor gyorsítja a járművet, amit a kormányon levő gomb segítségével tud a pilóta kezelni. Az ECU feladata tehát a teljes rendszer összehangolása, amit az új elektronika elosztottan végez, mivel a gomb- és kijelző-kezelés a kormányban levő kontroller feladata, a hajtás és az üzemanyagcella összehangolása a hátsó (fő-)elektronika feladata és ezek CAN-buszon keresztül kommunikálnak, mivel vannak olyan adatok/értékek, melyeket meg kell osztaniuk egymással.

2. A szoftverkörnyezet

Míg régebben nem is vagy csak alig volt jellemző a beágyazott szoftvert futtató elektronikus vezérlőegységek megléte a járművekben, addig manapság szinte minden egyes modern ECU azzal működik. Az ezredfordulóra az autóelektronika oly mértékű térhódítása volt tapasztalható, hogy szükségessé vált ezen egységek szoftvereinek az egységesítése. Ugyanis egyre nagyobb problémát jelentett azok specifikálása, implementálása és tesztelése. Ezért 2002-ben számos neves autógyártó összefogásával megalapították az AUTOSAR konzorciumot és elkezdték kidolgozni a szabványt, melyet a mai napig is fejlesztenek. Az egyesülés célja az autókban használt szoftver architektúra, illetve a szoftver modellezési és integrációs technológia egységesítése volt.

Az AUTOSAR egy mozaikszó, mely az AUTomotive Open System ARchitecture szavakból tevődik össze. Ez egy bárki számára elérhető, az autóipar szoftverarchitektúráját specifikáló szabvány.

Főbb jellemzői [12]:

- szabványokon alapuló szoftverfunkciók az ECU-k számára,
- skálázhatóság a különböző járművek számára,
- hordozható szoftver,
- jól definiált nyílt architektúra,
- nagy megbízhatóságú fejlesztési irányelvek,
- nemzetközi autóipari szabványok támogatása.

A szabvány fő irányelve az, hogy az egyes ECU-k funkcionalitását egymástól és a hardvertől függetlenül, atomi szoftverkomponensek (SWC – SoftWare Component) valósítják meg. A komponensek egymással szabványos AUTOSAR interfészekon keresztül kommunikálnak, amely VFB-n (Virtual Function Bus) zajlik. Ez egy absztrakt buszrendszer, melynek a feladata az, hogy elrejtse az egyes ECU-k határait a szoftverkomponensek elől. Ezáltal megvalósul az, hogy hardvertől függetlenül lehet fejleszteni az egyes szoftvermodulokat. Az architektúrának két nézőpontja van. Az egyik, ami a teljes rendszert egy egészként kezeli, az autóban lévő összes szoftverkomponenst összefoglalja, melyek csak a VFB-n keresztül vannak összekötve, mely így egy logikai rendszertervet ad. A másik lehetőség, pedig mikor már az egyes szoftvermodulok a valóságnak megfelelően, az azt futtató ECU-kban szerepelnek.

Mindez egy RTE-n (Runtime Environment) fut, mely az architektúra talán legfontosabb része. Teljes mértékű hardverfüggetlenséget szolgáltat az alkalmazási réteg felé, inter- és intrakommunikációt tesz lehetővé az ECU-k számára, interfészt biztosít az egyes modulok közötti adatcserére, illetve futtatja az alkalmazásrétegben található szoftverkomponenseket.

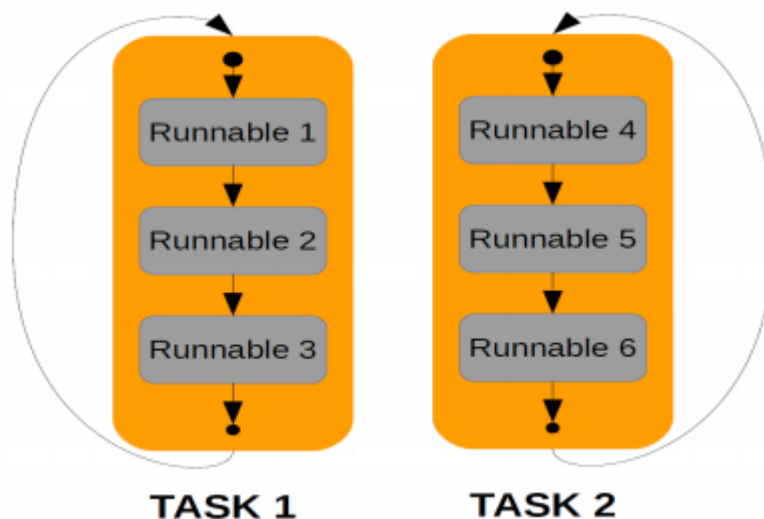
Az ORCA jármű vezérlőelektronikája az Autosar alapelvek alapján lett megtervezve azon funkciók bevonásával, amik a jármű számára használhatók. Természetesen a teljes rendszert nem lenne értelme implementálni, illetve túlmutatna egy egyetemi projekt méretein. A rendszernek egyébként része a teljes szoftveres megvalósítás is, de az nem hozzáférhető, így saját megoldásokat kellett alkalmazni, amely első lépésben a STM32F4 lapkán futó RTOS (Real time Operating System) operációs rendszert jelenti.

Az RTOS-sel szemben támasztott követelmények:

- hozzáférhető legyen a forráskódja,
- támogassa az elektronikában használt mikrovezérlőt,
- rendelkezzen a perifériák kezeléséhez szükséges programkönyvtárakkal,
- platformfüggetlenül lehessen használni a fejlesztőkörnyezetét,
- támogassa a párhuzamos feladatok végrehajtását,
- ingyenes legyen.

A ChibiOS/RT megfelel a követelményeknek, továbbá kiváló technikai támogatottsággal rendelkezik (fórumok, weblapok, stb), a forráskódja rendkívül jól dokumentált és végül, ami nagyban segíti a fejlesztés menetét, az a rendkívül egyszerű alkalmazásfejlesztési lehetősége.

Az AUTOSAR alapú alkalmazások komponensekből épülnek fel. Ez egy önálló, jól definiált funkciót végző szoftver elem, mely környezetével csak előre meghatározott interfészekon keresztül kommunikál. Ezeknek a futtatásához nyújt lehetőséget az RTE. Az egyes komponensek legfontosabb eleme a végrehajtható elem (futtatható függvény vagy ahogy a szabványban szerepel: runnable), melyek tartalmazzák a feladatuknak megfelelő utasításokat, például egy szenzor jelének a feldolgozása, egy beavatkozó eszköz vezérlése vagy egy az ezeket felhasználó szabályozás megvalósítása. Az egyes komponensek szájakba (task) vannak rendezve, melyeken belül a futtatható függvények szekvenciálisan és ciklikusan futnak le, azonban több szál is definiálható, melyek viszont párhuzamosan hajtódnak végre, ezt ábrázolja a 3. ábra.

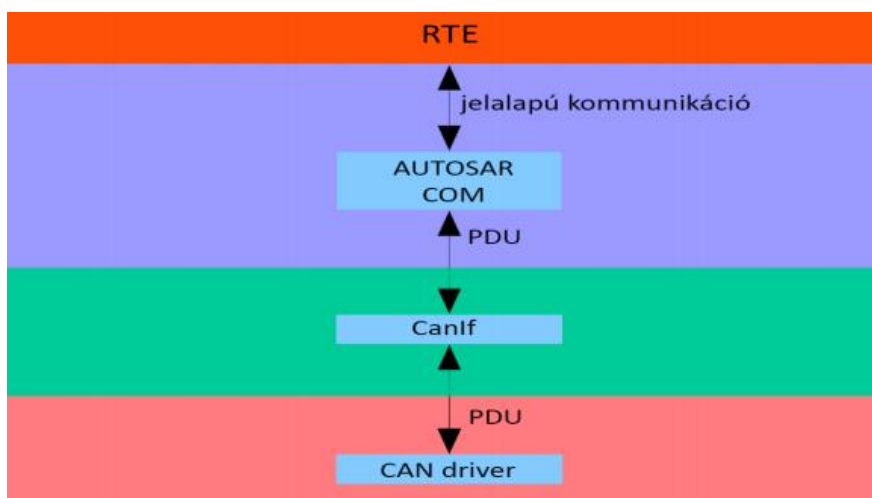


3. ábra Futtatható függvények a szájakon belül

Az egyes komponensek közötti kommunikációhoz szükség van egy szabványos interfészre. Az AUTOSAR-ban ezek külső függvényhívásokkal vannak megvalósítva, melyek globális változók írásával és olvasásával vannak leimplementálva, mellyel így megvalósul az ECU-n belüli kommunikáció. Az egyes komponensek több ilyen interfésszel is rendelkezhetnek,

amik adatáramlási irányuknak megfelelően, küldhetnek (Rte_Write függvényhívás) vagy fogadhatnak (Rte_Read függvényhívás) adatokat. Azonban még hátra van az ECU-k közötti kapcsolattartás. Ez úgy lett kialakítva, hogy az interfészeket megvalósító külső függvényhívások további függvényhívást intézhetnek. Ez a COM modul, ugyanis ennek a komponensnek a függvényeit (Com_SendSignal és Com_ReceiveSignal függvényhívás) meghívva lehet kezdeményezni az ECU-k közötti kommunikációt.

Az RTE és CanIf között elhelyezkedő COM modul (4. ábra) a kommunikáció lebonyolításának központi egysége, amely jel (signal, adat) és PDU (Protocol Data Unit, csomag) alapú információcserét tesz lehetővé. Adatküldéskor az RTE felől érkező jeleket PDU-kba rendezi és továbbküldi a CanIf modul felé, fogadáskor a beérkező PDU-k alapján pedig jeleket generál. Jeleket küldeni a COM modulnak a Com_SendSignal függvénnyel, míg fogadni a Com_ReceiveSignal függvénnyel lehetséges.



4. ábra A kommunikációs szoftver-réteg

A CanIf modul feladata azonban nem csak üzenetek küldéséből és fogadásából áll. Küldéskor a beérkező PDU-kat ki kell egészítenie a CAN protokollnak megfelelő adatelemekkel, fogadáskor pedig az érkező CAN csomagokból kell előállítani a PDU-kat. A CAN meghajtót (CAN driver), a ChibiOS-ban megvalósított programkönyvtár biztosította.

Ahhoz, hogy el lehessen készíteni az autót irányító szoftvert, elsőnek fel kellett térképezni, hogy milyen bemenetek lesznek az autóban, majd azokat felhasználva tervezhetők meg a vezérlő komponenseket, amelyek a kimeneteket hajtják meg. Ezen az absztrakciós szinten nem kellett azzal törődni, hogy az adott modul melyik ECU-n is fut, ugyanis az AUTOSAR-nak köszönhetően ez teljesen el van rejtve, azok csak szabványos interfészeket látnak, így az egész jármű szoftverét egy egységként lehet kezelni.

3. Digitális bemenetek

Ide tartoznak azok a komponensek, amelyek egy-egy digitális bemenetet olvasnak be. Feladatuk általában ennyiben ki is merül, azonban néhány modul, a biztonság érdekében alive counterrel lett kiegészítve, ugyanis ezeknek a jelei nincsenek helyben (ECU1-ben) felhasználva, csupán csak továbbítva vannak, így azok információ tartalmát biztosítani kell, mert jelentős mértékben befolyásolják az autó működését. A bemeneteket a 1. táblázat foglalja össze.

1. Táblázat: A bemenetek összegzése

	Érzékelés helye	Feldolgozás helye	Alive counter
Gáz gomb (ThrottleSensor)	ECU1	ECU2	igen
Fék (BrakeSensor)	ECU1	ECU2	igen
Első vészleállító gomb (FrontEmergencyButton)	ECU1	ECU2	igen
Hátsó vészleállító gomb (RearEmergencyButton)	ECU2	ECU2	nem
Rádió gomb (RadioButton)	ECU1	ECU2	nem
Funkció gomb (FunctionButton)	ECU1	ECU1	nem
Hidrogén szenzor (HydrogenSensor)	ECU2	ECU2	nem
Hidrogéncella ki- és bekapcsolás (FcOnOffButton)	ECU1	ECU2	nem

4. Analóg és impulzus bemenetek

Azok a bemenetek, melyek analóg információt hordoztak. Ezeket kétféle alcsoportba lehet osztani: közvetlen és közvetett.

A közvetlen analóg bemeneteket jelentett az a bemenet, melyeket a mikrovezérlőbe épített ADC (Analog to Digital Converter) konverterrel lett mérve. Ilyen bemenet volt az ultrakondenzátor feszültségét mérő komponens (UltraCapacitorVoltageSensor) is, mely az ADC-től érkező nyers információt valós feszültségértékké alakítja át. A közvetlen mérés azonban nem itt valósul meg, hanem egy párhuzamos szálban, ahol az analóg konverter folyamatosan működik, a komponens csak lekérdezi az ott lévő adatokat. Itt található továbbá, a digitális zajsűrés is. Mivel ez a bemenet rendelkezik hardveres szűréssel, ezért csak a konverter zajait kell kiküszöbölni, így erre a célra egy könnyen kivitelezhető, négyes középérték számolós algoritmust lett kidolgozva.

A másik csoport a közvetett mérés, ami azt jelenti, hogy az analóg jel nem közvetlenül a mikrokontrollerrel volt mérve, hanem egy külső eszköz segítségével, melyhez az valamilyen buszon keresztül csatlakozott, ami az esetünkben SPI volt. Ilyen közvetett mérést kellett végeznie az akkumulátor feszültségét mérő komponensnek (AuxBatteryVoltageSensor). A buszról érkező adatok azonban elég nagyfokú eltérést mutattak a valóságban mért értékektől, így azt mindenképpen pontosítani kellett, ezért lineáris interpoláció lett végezve a releváns értéktartományon belül. Ez a tartomány az autóban lévő lítium polimer akkumulátornál nagyjából 13 és 8 Volt közé tehető, ez lett felosztva tizenkét egységre, majd ki lett számolva minden egyes részhez tartozó korrekciós érték.

Két darab olyan bemenet van, melyen impulzust kellett számolni. Az egyik a sebességszenzortól érkezik, a másik pedig a hidrogéncella ventilátorától. Ezek detektálásához

az ICU (Input Capture Unit) meghajtót használja, ami az egységidő alatt beérkező lefutóélek számát adta meg. Ezekből a számértékekből a sebességszenzor komponens (SpeedSensor) meghatározta a hátsó kerék fordulatszámát (RPM-jét), a másik komponens (FuelCellFanSignalSensor) pedig eldöntötte, hogy működik-e a ventilátor vagy sem.

5. Vezérelt egységek (kimenetek)

A bemenetek alapján a megfelelő kimeneti értékeket kellett beállítani.

(A legalapvetőbb ilyen vezérlés a jármű mozgatása, a gyorsítás, de ezzel a következő fejezetben külön foglalkozunk.)

Ultrakondenzátor töltő vezérlés (PowerPathControl) – Az egyedüli bemenete ennek a komponensnek az ultrakondenzátor feszültség értéke volt. A kondenzátor töltés vezérléséhez két darab kimenet tartozik, melyeket annak feszültség szintje alapján kell logikai egybe vagy nullába állítani.

Hidrogénpumpa vezérlés – Ez esetben egyetlen egy bemenetet kell feldolgozni, ami a kondenzátor feszültsége, és egyetlen egy kimenetet, ami ebben az esetben egy PWM jel kitöltési tényező állítás volt.

Hidrogéncella vezérlés – Két bemenetet, egy kimenetet és egy állapotváltozót kellett logikailag kezelni. Az első bemenet a hidrogéncella be- és kikapcsoló (FcOnOffButton), melynek logikai értéke egy az egyben továbbítva van a kimenetre (FuelCellEnable).

A másik bemenet pedig hidrogéncella ventilátorának a logikai jele, ennek értéke szintén közvetlenül beírásra került az állapotváltozóba (Rte_TransmitBuffer_FuelCell_state).

Rádiókapcsolat vezérlés – A rádiókapcsolat ki- és bekapcsolásához, egy darab ki- és bemenetet kell felhasználni. A bemeneten (RadioButton) érkező logikai szintet módosítás nélkül továbbítva van a komponens a kimenete felé (RadioEnable).

Állapotjelző LED vezérlés – Ennek a komponensnek az a feladata, hogy kormányon lévő állapotjelző ledet vezérelje. Hat darab bemenetet és az öt darab kimenetet lehet definiálni.

2. Táblázat: A státuszlednek bementfüggései

Logikai bemenet	Kimenet	Logikai állapot
BrakeStatus == 0	BrakeLed	0
BrakeStatus == 1	BrakeLed	1
ThrottleStatus == 0	MotorLed	0
ThrottleStatus == 1	MotorLed	1
FuellCellState == 0	StackLed	0
FuellCellState == 1	StackLed	1
EmergencyState == 0 && EmergencyButtonState == 0	EmergencyLed	0
EmergencyState == 1 && EmergencyButtonState == 0	EmergencyLed	felváltva 1 és nulla
EmergencyState == 1 && EmergencyButtonState == 1	EmergencyLed	1

Ezekon kívül az akkufeszültség szintjéről is visszajelző Led ad tájékoztatást.

Vészleállítás – Ez egy kiemelten fontos vezérlő komponens, ugyanis hiba esetén ez fogja leállítani az autót. Bármely bemenetén érkező logikai 1 értékű állapot, azonnali vészleállítást okoz. Bemeneti interfészei:

- első vészleállító gomb,
- hátsó vészleállító gomb,
- hidrogénszivárgás érzékelő,
- belső vészleállítás állapotjelző,
- alive counter.

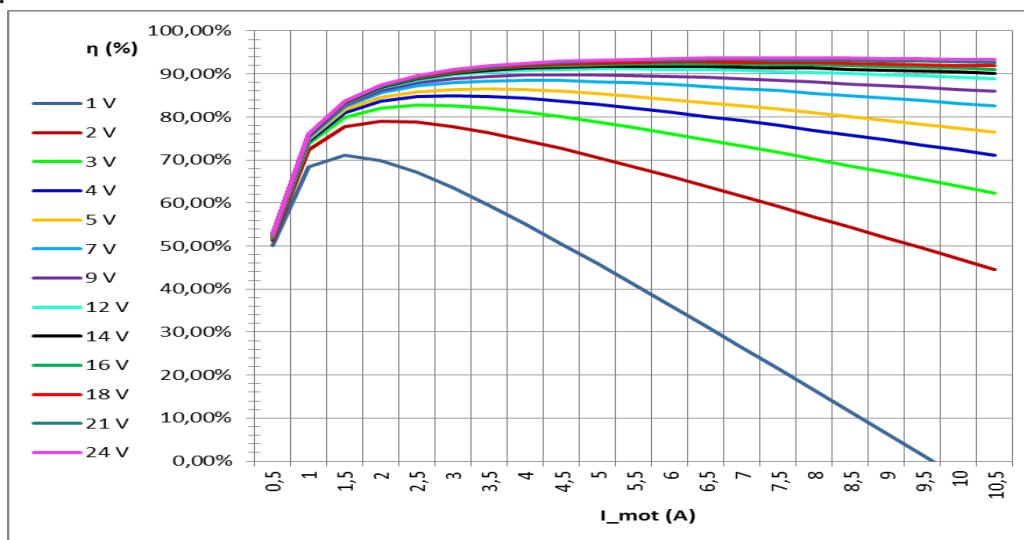
Vészleállítás bekövetkeztekor a modul nemcsak a kimenetét (Emergency), hanem a belső állapotjelző változóját is egybe állítja. A hiba megszűnésekor ez a változó továbbra is „1” értékű lesz, így az autó újraindítása csak külső beavatkozással történhet. Ezáltal megakadályozható az, hogy a sofőr a helyzetet felülbírálva újraindítsa az autót, azért, hogy tovább haladhasson a pályán a saját testi épségét kockáztatva.

6. A motor tulajdonságai, vezérlése

Nyilvánvalóan nagyon fontos dolog a jármű hajtásához alkalmazott motor magas hatásfoka, de ezt nem csak a megfelelő elektromotor megválasztásával lehet elérni, hanem a mindig optimális tartományban való működtetéssel.

Nem célszerű a vezető számára teret adni a gyorsulási karakterisztika megválasztásával, vagyis nem a hagyományos személygépjárművek metodikáját szabad követni ahol a „minél jobban nyomom a gázt, annál jobban gyorsulok” elv érvényesül. Az ORCA járműben egy nyomógombbal lehet gyorsítani, amit ha lenyomunk akkor az előre beprogramozott karakterisztika alapján „ad gázt” a program, a vezető csak a gyorsítás során elérni kívánt sebességet határozza meg.

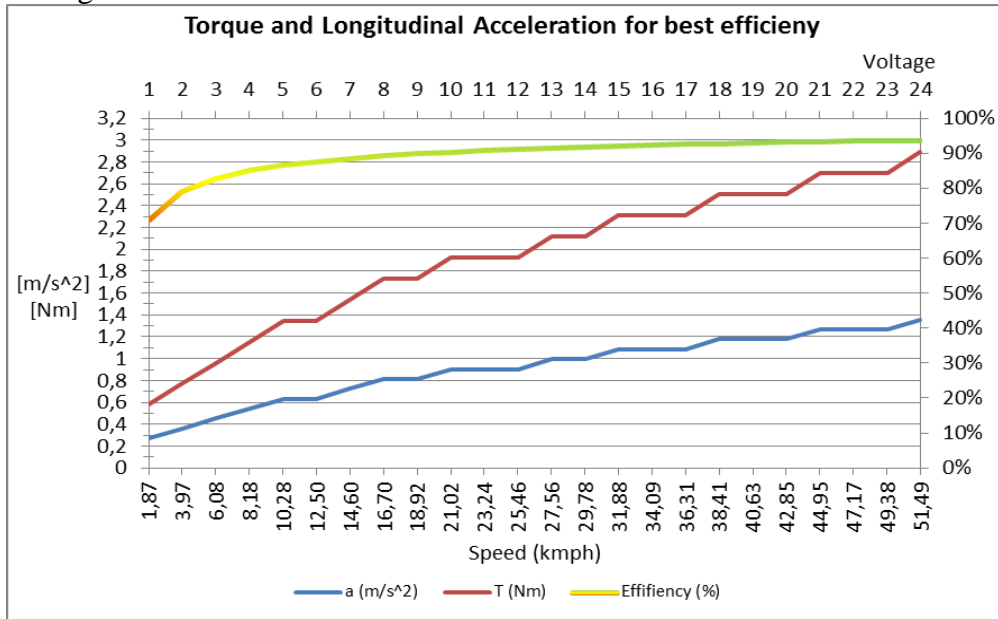
A karakterisztika meghatározásához ismerni kell a motor hatásfokát a feszültség és áramerősség függvényében, ez az adott 200 W-os motor esetében a következő módon alakul:



5. ábra Motorhatásfok az áramerősség-feszültség függvényében

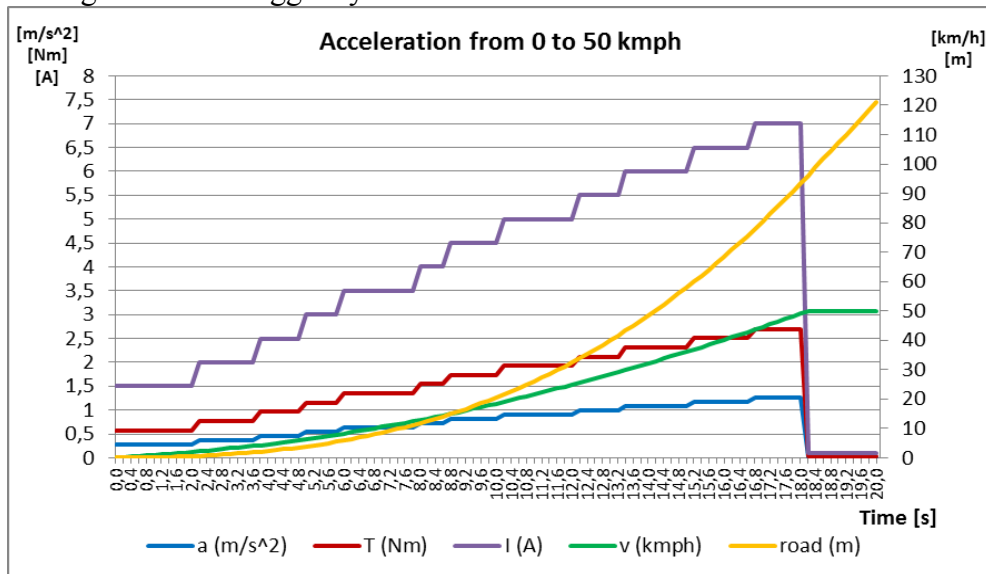
Alacsony áramfelvételnél az összes feszültségtartományban egyaránt alacsony a hatásfok, 50-55% között mozog. Viszont ahogy növeli a terhelést, alacsony fordulatszámon (alacsony feszültségen) hirtelen esik át a holtpontra a görbe, magasabb fordulaton azonban csak keveset csökken, vagy megközelítőleg állandóra vált.

Ezek alapján a gyorsítás és nyomaték görbét is meg lehet adni, törekedve arra, hogy a gyorsítási folyamat során az alacsony hatásfokú szegmensek csak kis időhányadban jelenjenek meg:



6. ábra Hatásfok változása a gyorsításhoz mérten

Az így megismert adatokból már meg lehet határozni, hogy egyenes úton hogyan kellene a gyorsítást elvégezni az idő függvényében:



7. ábra Gyorsulás 0-ról 50 km/h-ra az idő függvényében, és a megtett út

Ezt a gyorsítási görbét regressziós analízissel lekövetve a következő gyorsítási-egyenletet kapjuk, amelyet már meg lehet adni a controllernek:

$$y = 67.26 + 28.75 * e^{-2/x} \tag{1}$$

A második megoldás is feszültség-modifikáción alapszik, azonban itt egy statikus feszültség-értéket adunk a motorra, ami megfelelhet a kívánt végsebesség eléréséhez szükséges feszültségnek is akár. Ezt az állandó feszültséget nem folyamatosan adjuk a motorra, hanem egy úgynevezett PWM (Pulse Width Modification) - vagy más néven impulzus-szélesség

modifikációval. Egy bizonyos frekvencián kell adni négyszögjel-kommunikációval digitális jeleket a motorra, a motor fordulatszáma pedig attól függ, milyen széles kitöltéssel van logikai 1-be állítva a négyszögjel. Minél nagyobb a kitöltési tényező, annál magasabb lesz a motor fordulatszáma.

Így aztán kimenő értéknek vagy egy szorzó-tényezőt kell adni a ciklusonkénti feszültségnöveléshez, vagy egy "PWM" kitöltési tényezőt milliszekundum értéként.

Ha x a pillanatnyi motor-fordulatszám, akkor a szabályozás fix paraméterei pedig a következők:

1. $x < 209.42 \text{ rpm} \rightarrow 0.4406 \text{ V/s}, 1.5 \text{ A}, 0.5775 \text{ Nm}$
2. $209.42 \text{ rpm} < x < 44.56 \text{ rpm} \rightarrow 0.5875 \text{ V/s}, 2 \text{ A}, 0.77 \text{ Nm}$
3. $444.56 \text{ rpm} < x < 679.71 \text{ rpm} \rightarrow 0.7344 \text{ V/s}, 2.5 \text{ A}, 0.9625 \text{ Nm}$
4. $679.71 \text{ rpm} < x < 914.85 \text{ rpm} \rightarrow 0.8813 \text{ V/s}, 3 \text{ A}, 1.155 \text{ Nm}$
5. $914.85 \text{ rpm} < x < 1398 \text{ rpm} \rightarrow 1.0282 \text{ V/s}, 3.5 \text{ A}, 1.3475 \text{ Nm}$
6. $1398 \text{ rpm} < x < 1633.1 \text{ rpm} \rightarrow 1.1751 \text{ V/s}, 4 \text{ A}, 1.54 \text{ Nm}$
7. $1633.1 \text{ rpm} < x < 2116.3 \text{ rpm} \rightarrow 1.3219 \text{ V/s}, 4.5 \text{ A}, 1.7325 \text{ Nm}$
8. $2116.3 \text{ rpm} < x < 2847.4 \text{ rpm} \rightarrow 1.4688 \text{ V/s}, 5 \text{ A}, 1.925 \text{ Nm}$
9. $2847.4 \text{ rpm} < x < 3330.6 \text{ rpm} \rightarrow 1.6157 \text{ V/s}, 5.5 \text{ A}, 2.1175 \text{ Nm}$
10. $3330.6 \text{ rpm} < x < 4061.7 \text{ rpm} \rightarrow 1.7626 \text{ V/s}, 6 \text{ A}, 2.31 \text{ Nm}$
11. $4061.7 \text{ rpm} < x < 4792.8 \text{ rpm} \rightarrow 1.9095 \text{ V/s}, 6.5 \text{ A}, 2.5025 \text{ Nm}$
12. $4792.8 \text{ rpm} < x < 5524 \text{ rpm} \rightarrow 2.0563 \text{ V/s}, 7 \text{ A}, 2.695 \text{ Nm}$
13. $5524 \text{ rpm} < x < 5759.1 \text{ rpm} \rightarrow 2.2032 \text{ V/s}, 7.5 \text{ A}, 2.8875 \text{ Nm}$

A szabályozás referenciapontjaihoz szükséges még a felvett áramot is figyelni, ebből határozhatjuk meg, hogy a valós körülmények között megvalósul-e a papírforma, amennyiben eltér, kell feltételeket szabni a szabályozásba, hogy milyen irányú és nagyságú áramerősség-különbségek esetén hogyan reagáljon a rendszer a gyorsítás kontrollálásában.

7. Köszönetnyilvánítás

A cikk elkészülte a TÁMOP-4.2.2.C-11/1/KONV-2012-0005, Jól-lét az információs társadalomban pályázat finanszírozásában valósult meg.

Köszönet Csizmadia Tamásnak a motorkarakterisztikák számításáért és a Pollack Eco Team jelenlegi és korábbi tagjainak a megvalósításért.

Irodalomjegyzék

- [1] Shell: Shell Eco-marathon 2014 Rules, <http://s01.static-shell.com/content/dam/shell-new/local/corporate/ecomarathon/downloads/pdf/sem-global-official-rules-chapter-1-2014.pdf> [2014.06.05.]
- [2] I. Haber,N. Novak: Composite alternative vehicle with solar equipment, Proceedings of the 1st Regional Conference - Mechatronics in Practice and Education, 08-10. 12. 2011, Subotica, Serbia, pp. 192-200.

NI ELVIS & QUANSER modellek a számítógépvezérelt-irányítások oktatásában

NI ELVIS & QUANSER trainers for teaching control systems

Jancskárné Anweiler Ildikó, Várady Géza, Schiffer Ádám, Sári Zoltán

Pécsi Tudományegyetem, PMMIK / Műszaki Informatika Tanszék

jancskarne.ildiko@pmmik.pte.hu

Absztrakt: A Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Karán a Mérnök Informatikus szakon 1998-tól alkalmazzuk a NI LabVIEW grafikus fejlesztőrendszert oktatási anyagok, demonstrációs programok fejlesztésére a mérés-adatgyűjtés, szabályozások, számítógépes folyamatirányítás, kép/hangfeldolgozás, stb. témakörökben. A grafikus programozó szoftver mellett a National Instruments egyes hardver eszközei is megtalálhatók oktató laboratóriumainkban. Az ún. „hardware-in-the-loop” szimulációs rendszerek segítségével az egyszerű kétpontos szabályozásoktól a digitális PID-algoritmus vizsgálatán át eljuthatunk a fuzzy szabályozások szemléltetéséig. Többek közt a Quanser és NI együttműködésével kifejlesztett egyszabadsági fokú „helikopter” modelljét felhasználva elkészítettünk egy összetett: fuzzy, adaptív, kaszkád-jellegű irányítást bemutató szimulációs rendszert.

Kulcsszavak: hardware-in-the-loop szimuláció, digitális szabályozási modell, fuzzy szabályozás demonstráció

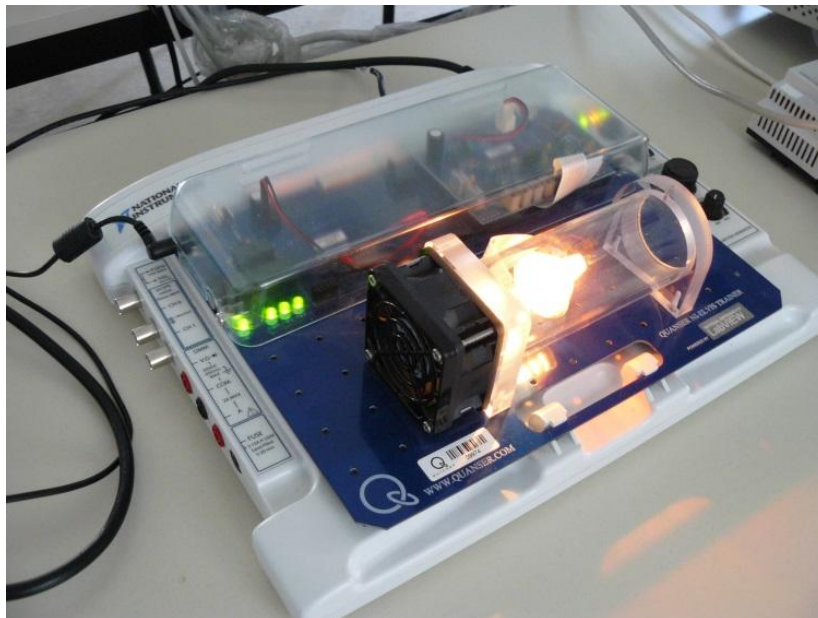
Abstract: At the Department of Information Technology at University of Pécs Hungary PMMIK, the LabVIEW graphical programming system and Quanser QNET Trainers are applied for support the teaching of automation and control theory. With these „hardware-in-the-loop” arrangements we can demonstrate control strategies from the two-point control to the fuzzy control. We have developed LabVIEW programs for visualizing fuzzy control systems operations. In this paper these programs and the control strategy of an adaptive fuzzy control of a 1 DoF flight is presented.

Keywords: hardware-in-the-loop simulation, digital control model, fuzzy control demonstration

1. Bevezetés

Az alkalmazott informatika egyik igen fontos, gyors ütemben fejlődő műszaki vonatkozású területe az autonóm rendszerek információtechnológiája: a holonikus rendszerek, a kibernetikai rendszerek informatikai vonatkozású komponenseinek megértéséhez elengedhetetlenek a megfelelő, alapozó jellegű irányítástechnikai ismeretek. A LabVIEW grafikus fejlesztőrendszerrel [1] a 90-es évek közepén ismerkedtünk meg. A fejlesztő rendszer rugalmassága, a programtervezés koncepciója (input–feldolgozás–output) a széleskörű eszközellésztés támogatás, a könnyedén látványos megjelenítő felület-létrehozás kínálta lehetőségek miatt több területen is bevontuk oktatásunkba. Többek között az elmúlt évek során számtalan demonstrációs anyagot dolgoztunk ki az irányításeleméleti anyagok szemléltetésére illetve a hallgatók gyakorlati oktatásának támogatására [2]. Néhány évvel ezelőtt az akkor még létező szakképzési támogatásokból be tudtunk szerezni néhány NI ELVIS gyakorlópadot és hozzájuk illeszthető oktatómodelleket: Quanser NI Engineering Trainers (QNET) [3], amelyeket a számítógépes irányítások, szabályozások szemléltetésére terveztünk felhasználni. Mivel minden eszközből csak egy darab áll rendelkezésre, úgy terveztük, hogy –miután megismerkedtünk a panelek adta lehetőségekkel–, oktatási anyagokat

dolgozunk ki, és egy hallgatói labor méretűre bővítjük az eszközállományt. Sajnos a szakképzési támogatások megszüntetésével ez a folyamat leállt. A QNET oktatómodellekhez a gyártó saját LabVIEW programokat is ad, melyek szakasz-identifikációra és a szabályozó algoritmus paraméterezésére, optimalizálására kidolgozott kész programok. Ezek a programok a szabályozástechnika oktatásában kiválóan alkalmazhatók. A számítógép-vezérelt irányítások oktatásában azonban a hangsúly már a saját programozású szabályozó algoritmusokon van, így az informatikus hallgatók gyakorlati oktatásában az eszközöket, mint irányítandó fizikai rendszert tekintjük, amelyek programjainknak inputokat szolgáltatnak, outputjaink pedig végrehajtó-beavatkozó jelként funkcionálnak. Így tehát ún. „hardware-in-the-loop” szimulációs rendszereket készítünk. Jelenleg három fajta QNET oktató modellel rendelkezünk: az első az ún. HVAC (Heating, Ventilation, Air Conditioning) modell, melyet a gyártó az állásos illetve a PI- szabályozás szemléltetésére szánt (1. ábra), a második egy DC-motor modell, mellyel fordulatszám illetve pozíciószabályozásokat tervezhetünk, a harmadik egy ún. egyszabadsági fokú helikopter modell (Vertical Take-Off and Landing (VTOL)), mellyel a kaszkádszabályozásra mutathatunk példát.



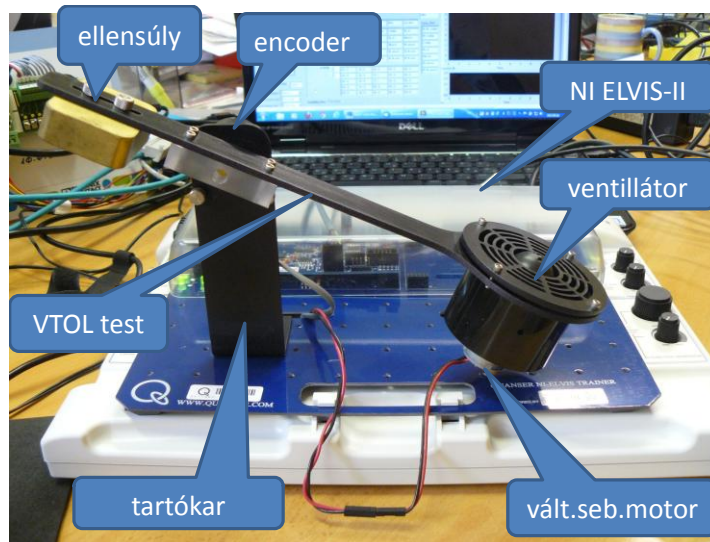
1. ábra NI ELVIS-II gyakorlópadba illesztett HVAC modell

2. A VTOL modell

Az NI ELVIS-II gyakorlópadba illesztett egyszabadsági fokú helikopter modell (VTOL) egy függőleges síkban elfordulni képes, egy ponton tengelyhez rögzített kar, melynek az egyik végén egy változtatható fordulatszámú motorral hajtott ventilátor helyezkedik el, a másik végén egy változtatható pozíciójú ellensúly található (lásd 2. ábra). A kar alaphelyzete az ábrán látható: a nehezebb ventilátortest lehúzza a kar végét a talajig, amely a vízszinteshez képest kb. -25° kitérést jelent. A ventilátorra feszültséget kapcsolva az armatúra áram által meghatározott nyomaték hatására a kar felemelkedik, a kar dinamikai tulajdonságát az ellensúly pozíciójának a változtatásával befolyásolhatjuk. A kar szabadtest ábrája és modell egyenletei a modell kézikönyvében megtalálhatók [4]. Az esettanulmányban az ellensúlyt úgy pozícionáltuk, hogy a vízszintes pozíciót 0° -kitérésnek tekintve, a kar erősen nemlineáris és

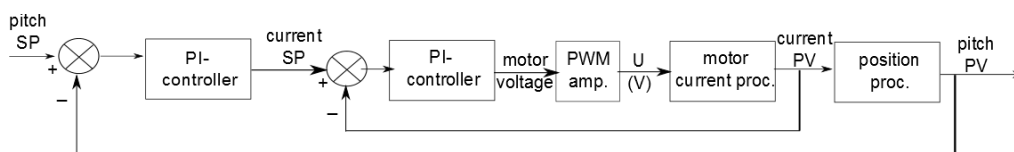
nem szimmetrikus viselkedést mutasson: a vízszinteshez képest pozitív szöget bezáró pozíció és negatív szögű (vízszintes alatti) pozíció eltérő dinamikai tulajdonságú.

A QNET modellek esetében a NI ELVIS gyakorlópadok a folyamatcsatoló szerepét töltik be: az ELVIS gyakorlópadba rejtett A/D kártyára csatlakoznak a modell mért folyamatváltozói és a számítógép által küldött digitális beavatkozó jelek analóg jellé alakítása is itt történik. Az ELVIS gyakorlópad USB csatolón kapcsolódik a számítógéphez, a LabVIEW eszközközkezelőjében konfigurálható. Az analóg és digitális ki/bemeneti jelek mint eszközhöz rendelt adatsatornák kezelhetők a LabVIEW programban. A QNET modellek elektronikája zárt, nem módosítható, a folyamatjelek adottak. A VTOL modell esetében az armatúra áram és feszültség analóg bemeneti jelek, a tengelynél rögzített digitális encoder digitális bemeneti jel, a motorra kapcsolandó feszültség analóg kimeneti jelcsatornánaként kezelendő. Az encoder számláló bemenetként funkcionáló digitális bemeneti jeléből a kar aktuális pozíciója (szögelfordulás) számítható.



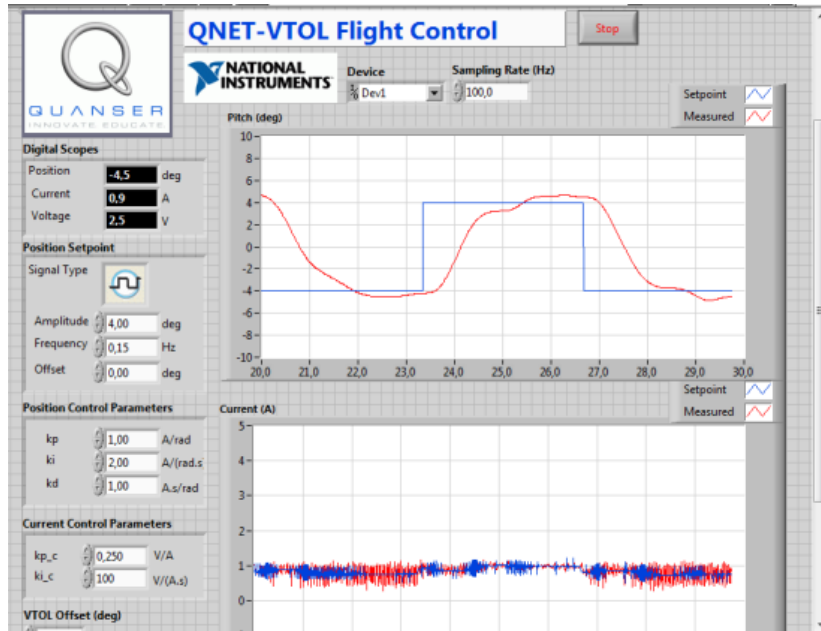
2. ábra NI ELVIS-II gyakorlópadba illesztett egyszabadsági fokú helikopter modell

A gyártó a modellt összetett (kaszád) szabályozás demonstrálására szánta. A VTOL test két alrendszerre bontható: a motorfeszültség – áram és az áram – pozíció dinamikára. A két dinamika két külön irányítási stratégiát kíván, az ún. belső vagy követő szabályozót a feszültség – áram dinamikához célszerű hangolni, a külső, vezető szabályozót pedig az áram – pozíció dinamikára. A gyártó által szállított LabVIEW programok és munkafüzetek segítségével ez a feladat programozás nélkül megoldható, a hangsúly itt tehát a szabályozó hangolásán van. A 3. ábra mutatja a kaszkád szabályozás jól ismert hatásvázlatát, a 4. ábrán a gyári program frontpanel képét láthatjuk.



3. ábra A VTOL tréner kaszkád szabályozás hatásvázlata

Ez a szabályozási rendszer szolgál összehasonlítási alapul az általunk kidolgozott összetett, fuzzy szabályozó rendszerhez. A feladat legyen a következő: a vízszintes pozícióhoz (0°) képeset $\pm 4^\circ$ periodikus négyszögjel alapjel változás követése minél gyorsabban, maradó hiba nélkül, minél kevesebb lengéssel. A 4. ábrán látható, hogy a klasszikus kaszkádszabályozás gyári beállítás szerint nem képes biztosítani adott periódus időn belül az állandósult állapotot, a jel nem kerül be az 5%-os dinamikus pontosság sávjába.



4. ábra A „gyári” kaszkád szabályozás LabVIEW frontpanel képe. A felső trendgörbe a vezető kör pozíció kívánt (kék) és tényleges (piros) értékeit mutatja. Az alsó trendgörbén a követő körben a motoráram kívánt és tényleges értékeit látjuk.

3. Fuzzy-PI szabályozás

Ma már gyakorlatilag minden szabályozásra is alkalmas irányítóberendezés, PLC gyártója felajánlja a digitális PID algoritmus mellett opcióként a fuzzy szabályozó algoritmust is. Ez az algoritmus az esetek döntő többségében a fuzzy PI- algoritmust jelenti, amelynek bemenő jele a hibajel és annak időbeli megváltozása, kimenő jele pedig a végrehajtójel-változás.

A fuzzy logika, a fuzzy szabályozások a 90-es évek elejétől szerepel oktatási palettánkon [5]. A fuzzy szabályozó tervezése a hallgatóknak első látásra túl bonyolult dolognak tűnik, különösen a terminológia újdonsága miatt. Később, ahogy begyakorolják a fogalmakat, és végigvezetjük őket egy fuzzy PI-szabályozó tervezésének lépésein, úgy gondolják, hogy egy nagyon egyszerű és nagyszerű dologgal ismerkedhettek meg: nincs differenciálegyenlet, nincs Laplace-transzformáció, rajzolgatunk néhány trapéz, háromszöget és kész a szabályozás. De honnan tudjuk, hogy megfelelően fog működni? A stabilitásvizsgálatra nincsenek általános formulák. A behangolásra nincsenek olyan ökölszabályok, mint a PID- szabályozók esetében [6][7]. A szabályozó tervezéséhez pedig nem rendelkeznek olyan mértékű szakértői tapasztalattal, amely egyébként a fuzzy rendszerek tervezésénél elvárható.

Az általunk első körben kidolgozott LabVIEW programok a legegyszerűbb és legáltalánosabban alkalmazott fuzzy PI- szabályozási algoritmusnak a tervezéséhez, szabályozás közbeni viselkedésének tanulmányozásához nyújtanak gyakorlati segítséget. A

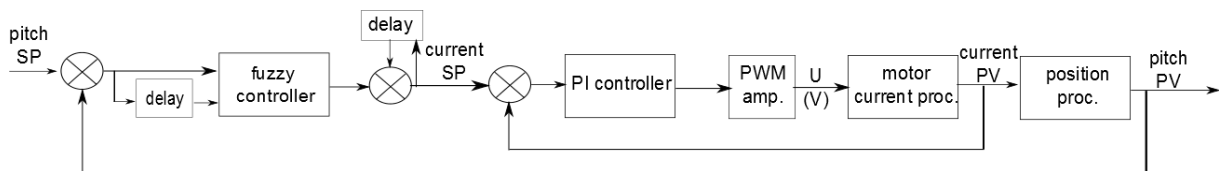
fázistere, amelyben az egyensúlyi pont a középső elemnek megfelelő szabály: ha a hiba zérus és a hiba változása zérus, ne változtassuk a beavatkozó jelet. Alapjel-változásra a szabályozás aktív szabályai folyamatosan változnak, szaturációban pl. a peremek aktivizálódnak, stabil szabályozás esetében vissza kell kerülni a középső szabályhoz.

A fázistérbeli viselkedés megfigyelésével tapasztalatot szerezhetnek a hallgatók a szabályozó viselkedéséről. A fázistérbeli viselkedés ötleteket adhat a fuzzy szabályozó módosítására [9]. A rosszul megtervezett szabályozó hibája (akár azért, mert nem fedi le az inputok alaphalmazai által kifeszített teret: hiányos, ritka a szabálybázis [10], akár azért mert rosszul vannak elosztva a tagsági függvények és ez pl. túl nagy lengéseket eredményez, stb.) könnyebben észrevehető, a szabályozás folyamata kiválóan szemléltethető.

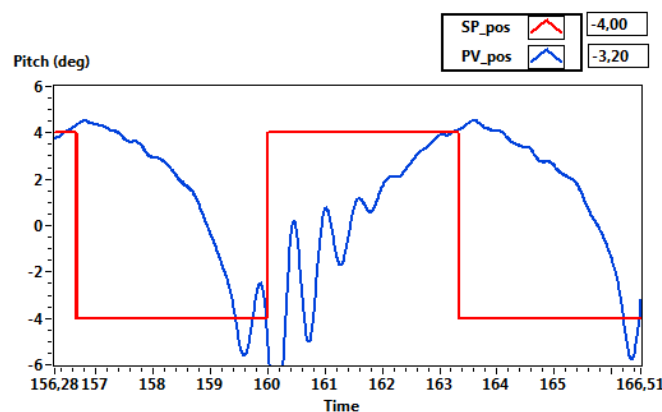
Elkészítettük a QNET oktatómodellekhez kapcsolódóan azok fuzzy szabályozását is. A program megjelenítő felülete nem sokban tér el az előzőtől, viszont ott a szabályozott berendezés egy valóságos eszköz, amelyhez a csatlakozást az NI ELVIS oktatópad analóg be-/kimeneti perifériái biztosítják.

4. VTOL modell fuzzy szabályozása

A QNET VTOL modell segítségével a fuzzy szabályozás lehetséges alkalmazásának egy összetettebb modelljét készítettük el. A kaskád szabályozás struktúráját megtartva, első lépésben a vezető szabályozót cseréltük le egy fuzzy PI- szabályozóra a 6. ábra hatásvázlata szerint. A VTOL modell nemlineáris dinamikája miatt a szabályozás nem elégíti ki az előírt feltételeket: a 7. ábra mutatja, hogy négyszögjel alapjel-változás követése mennyivel labilisabb a vízszinteshez képest negatív szögpozícióban, mint a pozitív szögpozíció esetén. A szabályozás pedig túl lassú a kívánt érték változási frekvenciájához mérten.



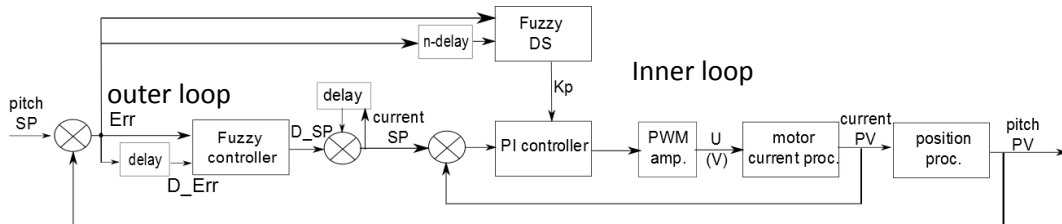
6. ábra kaskád szabályozás hatásvázlata, a vezető szabályozó fuzzy PI- szabályozó



7. ábra Négyszög alapjel követése, a szabályozó módosítása szükséges a gyorsítás és a lengések csillapítása miatt

A továbbiakban megpróbáljuk javítani a szabályozás minőségi paramétereit. Hangolható a szabályozás a belső kör (követő), illetve a külső kör (vezető) szabályozójának módosításával. Mivel a fuzzy szabályozó optimális beállítására nincsenek olyan „ökölszabályok”, optimalizálási technikák, mint a PID- szabályozóra, ezért a hangolás iteratív folyamat, mely során a hangolást végző személy is egyre több tapasztalatot szerez.

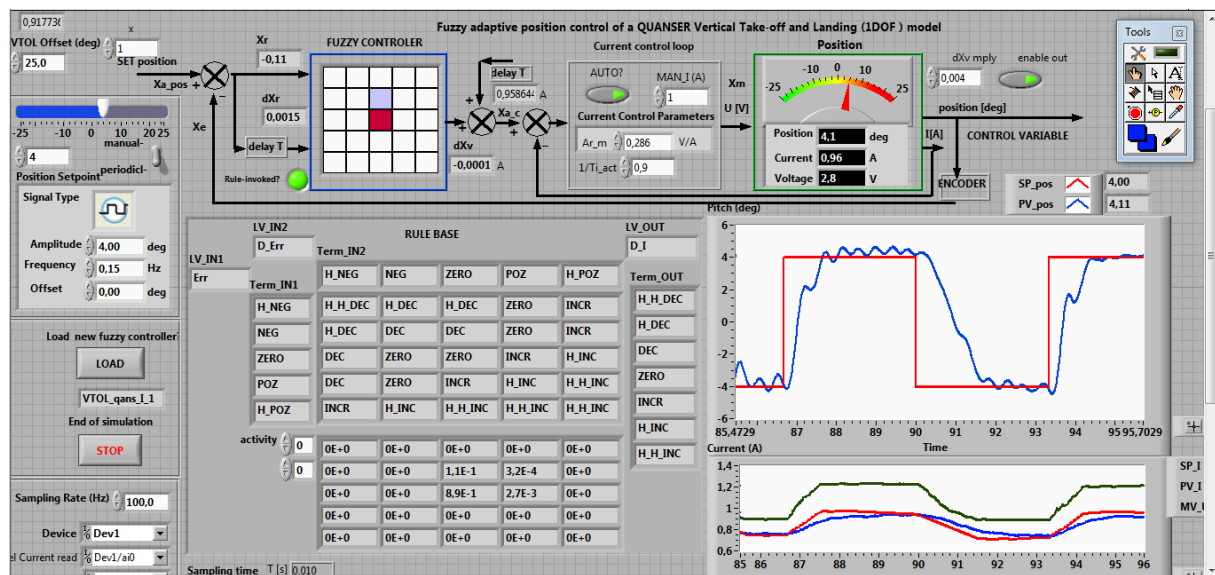
Esetünkben a szabályozást fel kell felgyorsítani, a lengések amplitúdóját pedig csökkentenünk kell az alsó szögpozícióban. Több lépésben eljuthatunk végül egy fuzzy döntési rendszerrel (Fuzzy DS) kibővített kaszkádszabályozási rendszerhez, melynek hatásvázlata a 8. ábrán látható.



8. ábra kaszkádszabályozás hatásvázlat, a vezető szabályozó fuzzy PI- szabályozó, a követő szabályozó erősítése adaptív

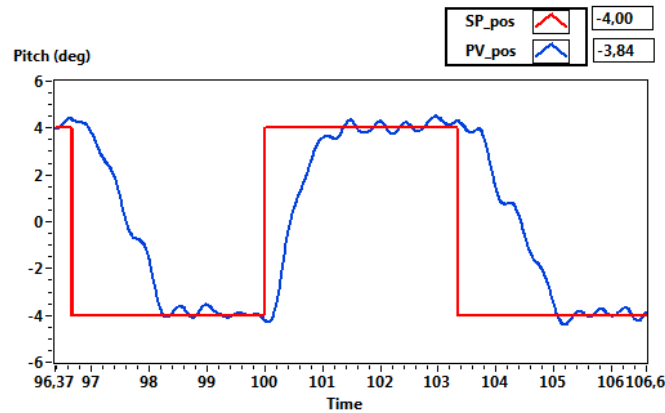
A gyorsításhoz a belső kör erősítési tényezőjét növeljük, de csak a tranziensek idejére, amíg a tényleges pozíció távol van a kívánt pozíciótól. Ahogy közelít a pozíció a kívánt értékéhez, a követő kör erősítési tényezőjét csökkenteni kell a túllendülés és a labilis viselkedés elkerülésére. Tehát megpróbáljuk a követő kör erősítési tényezőjét adaptálni a hibajelhez. Az adaptálódási feltételeket szintén fuzzy IF..THEN.. szabályok segítségével fogalmazzuk meg.

A nagy amplitúdójú lengések megakadályozására pedig a külső, vezető kör fuzzy szabályozó szimmetriáját „eltorzítjuk” (a szimmetrikus fuzzy PI-szabálybázist és a bemenő nyelvi változók tagsági függvényeit is), így a szabályozó eltérő mértékben fog beavatkozni negatív illetve pozitív hibajelek esetén.

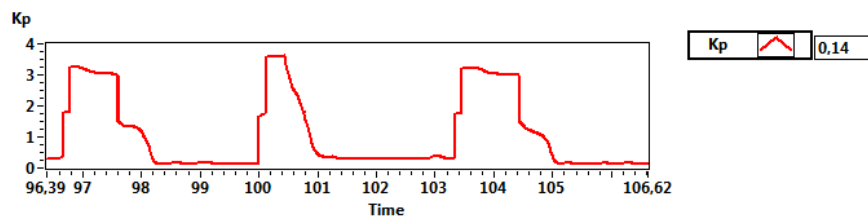


9. ábra QNET VTOL modell adaptív fuzzy kaszkád szabályozás frontpanel képe. Az ábra felső részén, a hatásvázlatban láthatók: a vezető fuzzy szabályozó fázistere: az ábrán a pillanatnyi szabályozás az egyensúlyi pontban van; a követő kör PI- szabályozójának pillanatnyi paramétereit; A VTOL tréner ELVIS panelen keresztül mért folyamatváltozói. Középen: az eltorzított szabálybázis.

A követő kör gyorsabb és pontosabb lett, a pozíciószabályozás gyorsabb lett, a lengések amplitúdója csökkent. A kívánt és tényleges pozíció trendgörbéjének részlete és a követő kör erősítési tényezőjének alakulása a 10. és 11. ábrákon láthatók. A szabályozás jellemző paraméterei javultak a klasszikus kaszkádszabályozáshoz képest.



10. ábra Alapjel-követés: a szabályozás gyorsabb: a $T_{90\%}$ és a T_{sz} szabályozási idő is meghatározható, mivel a lengések bent maradnak az 5%-os dinamikus pontosság sávjában



11. ábra A követő kör erősítési tényezőjének változása, melyet a hibajel és a jelváltozás tendenciájából egy fuzzy következtető rendszer határozott meg.

5. Összegzés

A nagy létszámú tanulócsoportokkal az eszközigényes gyakorlati oktatás egyre kevésbé lehetséges. Meg kell ragadnunk minden olyan lehetőséget, amellyel mégis a gyakorlati tapasztalatok megszerzésének az érzetét keltjük a hallgatókban. Az általunk bemutatott „hardware-in-the-loop” szimulációs rendszerekkel, a kidolgozott programok megjelenítő felületeivel a gyakorlatorientáltságra törekedtünk. A QNER trénerrel szerzett tapasztalataikat a hallgatóink a robotprogramozási projektjeikben hasznosíthatják.

6. Köszönetnyilvánítás

A fejlesztés a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú pályázat III./6. munkacsoport támogatásával történt.

Irodalomjegyzék

- [1] LabVIEW is a trademark of National Instruments Corporation
- [2] Jancskárné Anweiler I.: Számítógépvezérelt Irányítások, Interneten elérhető tananyag, készült a Phare ERF-ERFP-DD2002-HU-B-01 project 9.modul keretében, http://e-oktat.pmmf.hu/szamitogep_vezerelt_iranyitasok (2004)
- [3] QNET HVAC, VTOL are trademarks of Quanser Inc.
- [4] Quanser Inc.: QNET Practical Control Guide (2009)
- [5] Jancskárné A. I., Szakonyi L.: FuzzyTECH 3.1 Fejlesztőrendszer Alkalmazása a Műszaki Informatika Képzésben Folyamatok Fuzzy-elvű Szabályozásánál, Konferencia Kiadvány, Informatika a Felsőoktatásban '96 és Networkshop, szerk.: Bakonyi P., Herdon M., Debrecen, (1996). augusztus 27-30. 841-848.
- [6] C. Volosencu: Tuning Fuzzy PID Controllers, Introduction to PID Controllers - Theory, Tuning and Application to Frontier Areas, Prof. Rames C. Panda (Ed.), InTech, (2012).
- [7] J. Jantzen: Tuning Of Fuzzy PID Controllers, Technical University of Denmark, report 98-H 871, (1998)
- [8] fuzzyTECH is a registered trademark of INFORM GmbH.
- [9] D. Harikrishna, N. V. Srikanth: Phase Plane Method Stability Analysis of Fuzzy LogicPSS, Journal of Electrical Engineering, Vol.11. Ed.: 4, (2011), 153-159
- [10] T. L. Kóczy, D. Tikk: Fuzzy Rendszerek, Typotex, Budapest. (2000)

Hőtechnikai folyamatszimuláció lakótérben

Heat energetic process simulation in an accommodation

Szakonyi Lajos, Sári Zoltán

PTE Pollack Mihály Műszaki és Informatikai Kar

szakonyi.lajos@pmmik.pte.hu; sari.zoltan@pmmik.pte.hu

Absztrakt: Vizsgálataink olyan általános érvényű modellek felállítását célozták, melyek egy lakótér leggyakoribb „üzemállapotainál” a fűtési rendszerben történt technológiai, ill. üzemviteli módosítások, a hőenergia-szolgáltatás zavarai, s a környezeti hőmérséklet ingadozó jellege esetén is az előbbieket következményeként a céljellemező (léghőmérséklet) változását egymástól független hatásokként (információ-átviteli kategóriákként) képesek jellemezni. A lakótér energetikai szempontból eltérést mutató üzemállapot-alternatívái közül a konvekciós és a sugárzásos hőhasznosítást megvalósító radiátoros fűtést modelleztük a légtér és falazat közötti hőtranszport figyelembevételével. A munkaponti linearizálás alkalmazásával nyert működési egyenletekből transzformálással racionális törtként adódó átviteli függvények igen előnyösen, egymástól elkülöníthető módon, külön-külön jellemezhetik az energiaellátásban jelentkező zavarok hatását az üzemvitelre. Így a lakótér egyes működési módjainak, üzemállapotainak matematikai modellekkel történő leképezése, ezek segítségével az irányítási stratégiák felállítása, az irányított, intelligens rendszer szimulációja alapján az optimális üzemvitel meghatározása jól követhetően végezhető el. Természetesen az előbbi tevékenységek indokolják az egyes üzemviteli modellek szimulációs futtatásait, ezek kiértékelését és hasznosítását. A fejlesztés a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú „Jól-lét az információs társadalomban” című pályázat PMMIK Műszaki Informatika Tanszék III./6. „Smart Building” munkacsoportjában valósul meg. k

Kulcsszavak: matematikai modellezés, folyamatdinamika, folyamatirányítás,

Abstract: The main goal of our investigations is to establish a general model of an accommodation, which is capable of representing the various noises and inputs of a system as separate effects. From the possible and important operating conditions of an accommodation we have modeled the convective and radiative heating, where the heat transfer between the walls and the air had been taken into account. The transfer functions resulting from the transformed linearized system-equations represent the effect of the inputs of the system separately. Based on this approach the modeling and simulation of any particular operating mode can be done easily, and the control strategy for the optimal operation of the accommodation can be developed on the bases of the simulation results.

Keywords: mathematical modeling, process dynamics, control

1. Bevezetés

A kapcsolódó - e közlemény közvetlen előzményeként tekinthető „Intelligens lakótér folyamatdinamikája” c. előadásanyagban [1] rögzítésre kerültek egy hőszigeteléssel ellátott, gázkazánnal, helyiség-hőmérséklet és külső hőmérséklet alapján történő hőmérséklet-

szabályozással üzemelő társasházi lakótér hőtechnikai viszonyait jellemző működési egyenletek. A definiált folyamatdinamikai modellek (radiátoros fűtés a konvekciós és sugárzásos hőhasznosítás, a légtér és falazat közötti hőtranszport, a filtráció stb. figyelembevételével) struktúrájának meghatározása a deduktív modellezési technikát követve indult, de a modellparaméterek tisztázásához szükséges volt a működő objektumon kísérleteket, méréseket elvégezni, s a mérési eredmények a modell-paraméterek finomításánál a modell-szimuláció során megbízható támpontokat szolgáltatottak. A modellstruktúra és a főbb paraméterek előzetes meghatározása megtörtént, előbbieket ismeretében jelen előadás a modell-szimulációval kapcsolatos ismeretekkel kíván foglalkozni.

2. A folyamatdinamikai modell struktúrája és paraméterei

A mellékelt 1. ábra a lakótérre értelmezhető transzportfolyamatok egyes összetevőit szemlélteti. Lakótéri objektumra – kapcsolódva a megelőző publikációhoz – a hőenergia átszármaztatásának mikro-jelenségeit is formalizáló mérlegegyenleteknek [2,3] a munkaponti linearizálással nyert alakjai az alábbiak (az egyes időfüggő változók értelmezése: felülvonással jelzett munkaponti értékük, Δ -val jelzett eltérésük összege) a fűtőközegre:

$$\begin{aligned} & (\overline{q_{vf}} + \Delta q_{vf}) \rho_f c_{pf} (\overline{\mathcal{G}_{fb}} + \Delta \mathcal{G}_{fb}) - (\overline{q_{vf}} + \Delta q_{vf}) \rho_f c_{pf} (\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk}) - \alpha A (\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk}) + \alpha A (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) = \\ & = V_f \rho_f c_{pf} \frac{d(\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk})}{dt}, \end{aligned} \quad (1)$$

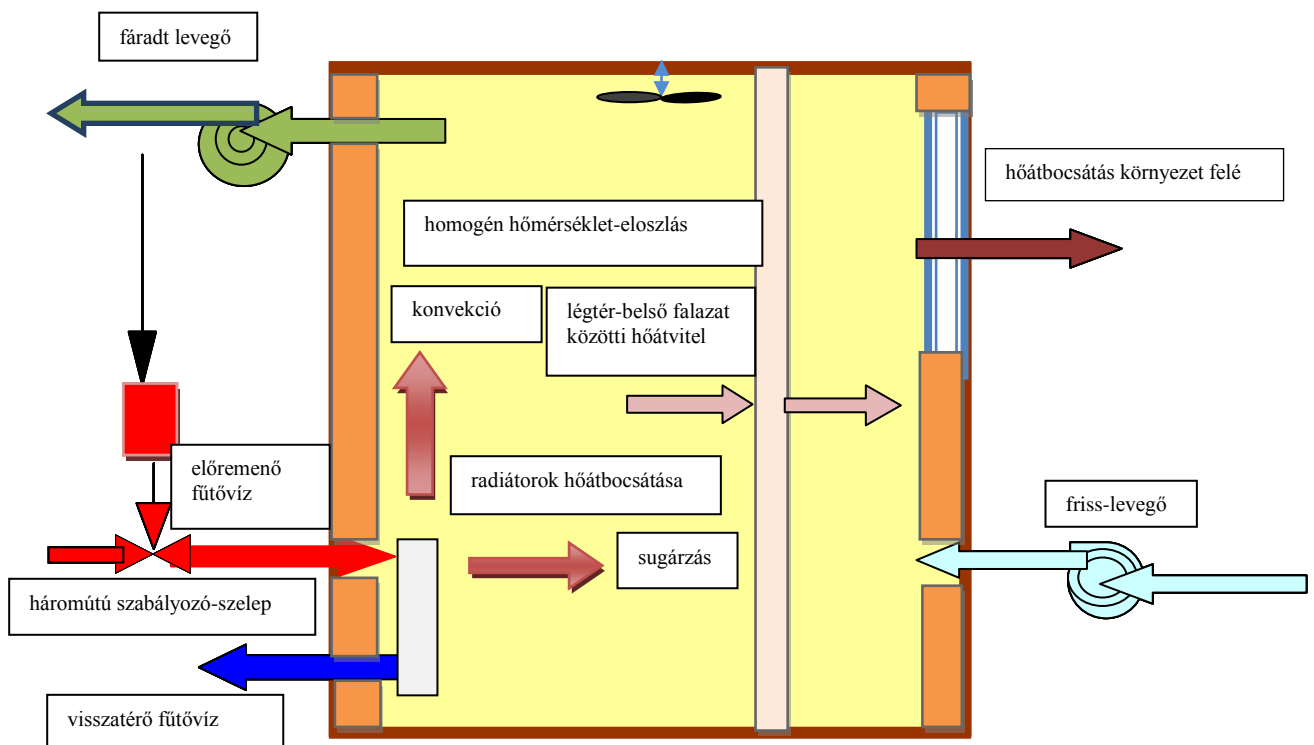
a légtérre:

$$\begin{aligned} & (\overline{q_{vl}} + \Delta q_{vl}) \rho_l c_{pl} (\overline{\mathcal{G}_{lb}} + \Delta \mathcal{G}_{lb}) - (\overline{q_{vl}} + \Delta q_{vl}) \rho_l c_{pl} (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) + \alpha A (\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk}) - \\ & - \alpha A (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) - \alpha_t A_t (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) + \alpha_t A_t (\overline{\mathcal{G}_t} + \Delta \mathcal{G}_t) - \alpha_k A_k (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) + \alpha_k A_k (\overline{\mathcal{G}_k} + \Delta \mathcal{G}_k) = \\ & = V_l \rho_l c_{pl} \frac{d(\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk})}{dt}, \end{aligned} \quad (2)$$

s végül a légtér és a falazat közötti hőátszármaztatásra:

$$\alpha_t A_t (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) - \alpha_t A_t (\overline{\mathcal{G}_t} + \Delta \mathcal{G}_t) = V_t \rho_t c_t \frac{d(\overline{\mathcal{G}_t} + \Delta \mathcal{G}_t)}{dt} \quad (3)$$

(ahol q_{vf} - fűtőközeg térfogatsebesség; ρ_f - fűtőközeg sűrűség; c_{pf} - fűtőközeg fajhő; g_{fb}, g_{fk} - a fűtőközeg belépési, kilépési hőmérséklete; g_{lk} - légtér hőmérséklet; α, A - a radiátorok hőátadási tényezője, hőátadásra hasznosítható felülete; V_f - folyadékterfogat; α_k, A_k - hőátadási tényező, hőátadó felület környezet felé; g_k - környezeti hőmérséklet; ρ_l, c_{pl} - levegő sűrűség, állandó nyomáson mért fajhő; V_l - légtér térfogat; α_t - a belső válaszfalak hőátbocsátási tényezője; g_t - a beltéri falak hőmérséklete; A_t, V_t, ρ_t, c_t - a válaszfalak felülete, térfogata, sűrűsége, fajhője; t - időkoordináta).



1.ábra. A hőtranszport összetevői.

A modellépítésnél felhasznált - működtetéssel, konstrukcióval, hőtranszporttal kapcsolatos - főbb jellemzők az alábbiakban kerültek rögzítésre:

ρ_f - fűtőközeg sűrűség : **989,82 kg/m³**; c_{pf} - fűtőközeg fajhő : **4176 J/kg K**; α - hőátbocsátási tényező (a fűtővíz-légtér között) : **10,5 W/m² K**; A - hőleadók hasznosítható felülete : **16,741 m²**; V_f - fűtőközeg folyadékterfogat : **0,076455 m³**; α_k - hőátbocsátási tényező (lakótér-környezet) : **0,586 W/m² K**; A_k - hőátadó felület környezet felé : **116,58 m²**; ρ_l - levegő- sűrűség : **1,293 kg/m³**; c_{pl} - levegő állandó nyomáson mért fajhő : **1012 J/kg K**; V_l - légtér térfogat: **174,1m³**; α_t - hőátadási tényező (a légtér és a belső

falazat között) : **2,72 W/m² K**; V_t - a beltéri válaszfalak térfogata : **5,48 m³**; A_t – válaszfalak felülete : **45,63 m²**; $c_t, \text{ill. } \rho_t$ - a téglafajhője : **880 J/kg K**, ill. sűrűsége : **2500 kg/m³**; $\overline{q_{vf}}$, ill. $\overline{q_{vl}}$ - a szellőztetés légáramának, ill. a fűtővíz térfogatáramának munkaponti értéke: **0,000016 m³/s**, , ill. **0,171 m³/s** .

A munkaponti értékek alapján az állandósult állapotra vonatkozó hőmérleg a fűtőközeg:

$$\overline{q_{vf}} \rho_f c_{pf} \overline{\mathcal{G}_{fb}} - \overline{q_{vf}} \rho_f c_{pf} \overline{\mathcal{G}_{fk}} - \alpha A \overline{\mathcal{G}_{fk}} + \alpha A \overline{\mathcal{G}_{lk}} = 0, \quad (4)$$

a légtér:

$$\overline{q_{vl}} \rho_l c_{pl} \overline{\mathcal{G}_{lb}} - \overline{q_{vl}} \rho_l c_{pl} \overline{\mathcal{G}_{lk}} + \alpha A \overline{\mathcal{G}_{fk}} - \alpha A \overline{\mathcal{G}_{lk}} - \alpha_t A_t \overline{\mathcal{G}_{lk}} + \alpha_t A_t \overline{\mathcal{G}_t} - \alpha_k A_k \overline{\mathcal{G}_{lk}} + \alpha_k A_k \overline{\mathcal{G}_k} = 0, \quad (5)$$

s a beltéri falazat esetén:

$$\alpha_t A_t \overline{\mathcal{G}_{lk}} - \alpha_t A_t \overline{\mathcal{G}_t} = 0. \quad (6)$$

Az előbbi differenciál-egyenletekben képezve a kéttagú kifejezések szorzatát, s eltekintve a másodrendűen kicsiny mennyiségektől ($\Delta * \Delta$), majd levonva előbbiekből az állandósult állapotokra vonatkozó egyenleteket, lineáris differenciál-egyenleteket nyertünk, melyekre alkalmaztuk a *Laplace*-transzformációt (Δ jelölést elhagyásával). Bevezetésre kerültek a *Stanton*-számként értelmezhető konstansok (a_i), s az áramló fűtővíz és a friss-levegő átlagos tartózkodási idejét jelentő időállandók:

$$a_1 = \frac{\alpha A}{\rho_f c_{pf} \overline{q_{vf}}}, \quad a_2 = \frac{\alpha A}{\rho_l c_{pl} \overline{q_{vl}}}, \quad a_4 = \frac{\alpha_k A_k}{\rho_l c_{pl} \overline{q_{vl}}}, \quad a_6 = \frac{\alpha_t A_t}{\rho_l c_{pl} \overline{q_{vl}}}, \quad a_7 = \frac{\alpha_t A_t}{\rho_t c_t V_t}; \quad \overline{t_f} = \frac{V_f}{\overline{q_{vf}}} \text{ ill.} \\ \overline{t_l} = \frac{V_l}{\overline{q_{vl}}} \text{ (az áramló fűtőközeg, ill. levegő átlagos tartózkodási ideje).} \quad (7)$$

A légtérbe belépő frisslevegő hőmérsékletét azonosnak tekintettük a környezeti hőmérséklettel, tehát: $\overline{\mathcal{G}_{lb}} = \overline{\mathcal{G}_k} = 0$. (A belépésnél és a kilépésnél értelmezett fűtőközeg-hőmérséklet munkaponti értékeinek különbségét képezve adódó, a következő összefüggésben alkalmazott jelölés: $\Delta \overline{\mathcal{G}_f} = \overline{\mathcal{G}_{fb}} - \overline{\mathcal{G}_{fk}}$.) A fűtőközeg kilépési hőmérsékletére (\mathcal{G}_{fk}) rendezve a transzformált algebrai egyenleteket, összevetésükkel a légtér-hőmérséklet *Laplace*-transzformáltjára rendezett alakban nyertük az alábbi eredő átviteli függvényt, mely az egyes bemenetek transzformáltjait egymástól elkülönítetten tartalmazza, tehát $\mathcal{G}_{lk}(s) =$

$$\begin{aligned}
 &= \frac{(a_2 s + a_2 a_7)(1 + a_1 + s \overline{t_f}) \left[\mathcal{G}_{lb}(s) - \frac{\overline{\mathcal{G}_{lk}}}{q_{vl}} q_{vl}(s) + a_4 \mathcal{G}_k(s) \right]}{(1 + a_2 + a_4 + a_6 + s \overline{t_l})(a_2 s + a_2 a_7)(1 + a_1 + s \overline{t_f}) - a_2 a_6 a_7 (1 + a_1 + s \overline{t_f}) - a_1 a_2 (a_2 s + a_2 a_7)} + \\
 &+ \frac{a_2 (a_2 s + a_2 a_7) \left[\mathcal{G}_{fb}(s) + \frac{\Delta \overline{\mathcal{G}_f}}{q_{vf}} q_{vf}(s) \right]}{(1 + a_2 + a_4 + a_6 + s \overline{t_l})(a_2 s + a_2 a_7)(1 + a_1 + s \overline{t_f}) - a_2 a_6 a_7 (1 + a_1 + s \overline{t_f}) - a_1 a_2 (a_2 s + a_2 a_7)}.
 \end{aligned} \tag{8}$$

Az előbbi kifejezésből leszarmaztatjuk az alapjel-követésre vonatkozó és az egyes zavar-átviteli függvényeket.

A *Stanton*-számok bevezetésével, s a munkaponti értékek figyelembevételével az állandósult állapotra vonatkozó hőmérleg a fűtőközeg:

$$\overline{\mathcal{G}_{fb}} - \overline{\mathcal{G}_{fk}} - a_1 \overline{\mathcal{G}_{fk}} + a_1 \overline{\mathcal{G}_{lk}} = 0, \tag{9}$$

a légtér:

$$\overline{\mathcal{G}_{lb}} - \overline{\mathcal{G}_{lk}} + a_2 \overline{\mathcal{G}_{fk}} - a_2 \overline{\mathcal{G}_{lk}} - a_6 \overline{\mathcal{G}_{lk}} + a_6 \overline{\mathcal{G}_t} - a_4 \overline{\mathcal{G}_{lk}} + a_4 \overline{\mathcal{G}_k} = 0, \tag{10}$$

s a beltéri falazat esetén:

$$a_7 \overline{\mathcal{G}_{lk}} - a_7 \overline{\mathcal{G}_t} = 0. \tag{11}$$

Az előbbi munkaponti egyenletek értelmezése alapján a következőket állapíthatjuk meg. A (11) egyenletből következően állandósult állapotban a légtér hőmérséklete megegyezik a beltéri falazat hőmérsékletével:

$$\overline{\mathcal{G}_{lk}} = \overline{\mathcal{G}_t}. \tag{12}$$

A (9) egyenlet alapján a fűtővíz-hőmérséklet és a céljellemezőnek tekinthető légtér-hőmérséklet munkaponti értékei közötti összefüggés az alábbiak szerint számítható:

$$\overline{\mathcal{G}_{fk}} = \frac{\overline{\mathcal{G}_{fb}} + a_1 \overline{\mathcal{G}_{lk}}}{1 + a_1} \tag{13}$$

A (12) összefüggést, valamint a $\overline{g_{lb}} = \overline{g_k} = 0$ azonosságot figyelembe véve a (10) egyenlet alapján a fűtővíz kilépő hőmérséklete és a légtér-hőmérséklet közötti kapcsolat pedig a következő:

$$\overline{g_{lk}} = \frac{a_2}{1 + a_2 + a_4} \overline{g_{fk}}. \quad (14)$$

A (13) és a (14) egyenletek összevetéséből adódóan állandósult állapotban a belépő folyadék hőmérsékletének hatását a légtér hőmérsékletére az alábbi összefüggéssel jellemezhetjük:

$$\overline{g_{lk}} = \frac{a_2}{1 + a_1 + a_2 + a_4 + a_1 a_4} \overline{g_{fb}}. \quad (15)$$

A (8) összefüggéssel jelzett modellstruktúrába, mint operátor-tartománybeli eredő átviteli függvénybe helyettesítsük be a (7) jelű kapcsolatokkal jellemzett, konstansként tekintett modell-paramétereket. Így a tört számlálójában elkülönítetten szereplő öt bemeneti jellemzőre külön-külön is meghatározhatók a zavarátviteli, illetve az alapjel-követésre vonatkozó átviteli függvények (eközben a másik négy bemenet időbeli állandóságát feltételezzük, tehát transzformáltjaikat zérussal helyettesítjük). Az alábbiak szerint számszerűsített zavarátviteli függvények kifejezik a légtér hőmérsékletére kifejtett hatását

- a belépő levegő hőmérséklet-változásának:

$$Y_1(s) = \left[\frac{0,000982194 (s + 0,0000103042) \cdot (s + 0,000765495)}{s^3 + 0,00337456 s^2 + 1,58921 \cdot 10^{-6} s + 1,17788 \cdot 10^{-11}} \right], \quad (16)$$

- a fűtővíz belépési (előremenő) hőmérséklet-változásának:

$$Y_2(s) = \left[\frac{1,61476 \cdot 10^{-7} \cdot s + 1,66387 \cdot 10^{-12}}{s^3 + 0,00337456 s^2 + 1,58921 \cdot 10^{-6} s + 1,17788 \cdot 10^{-11}} \right], \quad (17)$$

- a környezet hőmérséklet-ingadozásának:

$$Y_3(s) = \left[\frac{0,000300159 (s + 0,0000103042) \cdot (s + 0,000765495)}{s^3 + 0,00337456 s^2 + 1,58921 \cdot 10^{-6} s + 1,17788 \cdot 10^{-11}} \right], s \quad (18)$$

- a friss-levegő áram bevezetés, a szellőztetés igénybevételének:

$$Y_4(s) = \left[\frac{0,124639 \cdot (s + 0,0000103042) \cdot (s + 0,000765495)}{s^3 + 0,00337456 s^2 + 1,58921 \cdot 10^{-6} s + 1,17788 \cdot 10^{-11}} \right]. \quad (19)$$

A fűtővíz árama - mint módosított jellemző – változtatásának a céljellemezőre (szabályozott jellemzőre) gyakorolt hatása pedig a következő, alapjel-követésre vonatkozó átviteli függvény alapján értelmezhető:

$$Y_5(s) = \left[\frac{0,236865 \cdot s + 2,44069 \cdot 10^{-6}}{s^3 + 0,00337456 s^2 + 1,58921 \cdot 10^{-6} s + 1,17788 \cdot 10^{-11}} \right]. \quad (20)$$

Az előbbi átviteli függvényekben a számítási műveletek elvégzése az s -operátorra nézve harmadrendű egyenletek megoldását, a zérus-helyek és pólusok kiszámítását tette szükségessé. Így nyertük az átviteli függvényeknek az alábbi, 3 részlettörré bontott, gyöktényezős alakját:

$$Y_1(s) = \left[\frac{0,0000906427}{s + 0,000556407} + \frac{0,00089021}{s + 0,00281062} + \frac{1,34142 \cdot 10^{-6}}{s + 7,53193 \cdot 10^{-6}} \right] \quad (21)$$

$$Y_2(s) = \left[\frac{0,000071271}{s + 0,000556407} - \frac{0,0000715619}{s + 0,00281062} + \frac{2,90955 \cdot 10^{-7}}{s + 7,53193 \cdot 10^{-6}} \right] \quad (22)$$

$$Y_3(s) = \left[\frac{0,0000277005}{s + 0,000556407} + \frac{0,000272049}{s + 0,00281062} + \frac{4,09939 \cdot 10^{-7}}{s + 7,53193 \cdot 10^{-6}} \right] \quad (23)$$

$$Y_4(s) = \left[-\frac{0,0115025}{s + 0,000556407} - \frac{0,112967}{s + 0,00281062} - \frac{0,000170225}{s + 7.53193 \cdot 10^{-6}} \right] \quad (24)$$

$$Y_5(s) = \left[\frac{104,546}{s + 0,000556407} - \frac{104,972}{s + 0,00281062} + \frac{0,426795}{s + 7.53193 \cdot 10^{-6}} \right] \quad (25)$$

Az átviteli függvények átalakításával adódó időállandós formák a következők:

$$Y_1(s) = \left[\frac{0,162907}{1797,25 \cdot s + 1} + \frac{0,178098}{132768 \cdot s + 1} + \frac{0,316731}{355,793 \cdot s + 1} \right] \quad (26)$$

$$Y_2(s) = \left[\frac{0,128091}{1797,25 \cdot s + 1} + \frac{0,0386296}{132768 \cdot s + 1} - \frac{0,0254613}{355,793 \cdot s + 1} \right] \quad (27)$$

$$Y_3(s) = \left[\frac{0,0497846}{1797,25 \cdot s + 1} + \frac{0,0544268}{132768 \cdot s + 1} + \frac{0,0967932}{355,793 \cdot s + 1} \right] \quad (28)$$

$$Y_4(s) = \left[-\frac{20,6727}{1797,25 \cdot s + 1} - \frac{22,6004}{132768 \cdot s + 1} - \frac{40,1928}{355,793 \cdot s + 1} \right] \quad (29)$$

$$Y_5(s) = \left[\frac{187894}{1797,25 \cdot s + 1} + \frac{566647}{132768 \cdot s + 1} - \frac{373485}{355,793 \cdot s + 1} \right] \quad (30)$$

Az előbbi átviteli függvények az operátor-tartományban három, konstansokkal szorzott PT_1 jelátviteli tag összegeként tekinthetők, melyek általános alakban megadva:

$$Y_1(s) = \left[\frac{k_{11}}{T_1 s + 1} + \frac{k_{12}}{T_2 s + 1} + \frac{k_{13}}{T_3 s + 1} \right]; \quad Y_2(s) = \left[\frac{k_{21}}{T_1 s + 1} + \frac{k_{22}}{T_2 s + 1} + \frac{k_{23}}{T_3 s + 1} \right];$$

$$Y_3(s) = \left[\frac{k_{31}}{T_1 s + 1} + \frac{k_{32}}{T_2 s + 1} + \frac{k_{33}}{T_3 s + 1} \right]; \quad Y_4(s) = \left[\frac{k_{41}}{T_1 s + 1} + \frac{k_{42}}{T_2 s + 1} + \frac{k_{43}}{T_3 s + 1} \right]$$

$$Y_5(s) = \left[\frac{k_{51}}{T_1 s + 1} + \frac{k_{52}}{T_2 s + 1} + \frac{k_{53}}{T_3 s + 1} \right] \quad (31)$$

(ahol T_1, T_2, T_3 - az elsőrendű tagok időállandói).

Digitális (mintavételes) irányítási rendszer megtervezéséhez, intelligens rendszerek szimulálásához, létrehozásához, működtetéséhez – a folyamatos elemet, esetünkben az átviteli függvényével jellemzett lakótéri objektumot – szükséges kiegészíteni tartó-, s mintavételező szervvel, majd meghatározni az így nyert tagcsoport impulzus-átviteli függvényét [4]. Ez teszi lehetővé az irányítási algoritmussal működtetett objektum vizsgálatát, modellezését, számítógépes szimulációját, az optimális üzemeltetés meghatározását.

Az operátor-tartományban az összegzés elemek párhuzamos kapcsolódására utal. Így ebben az esetben egyenértékű kapcsolásként megadhatunk három párhuzamosan kapcsolt ágat, s az egyes ágakban zérusrendű tartót, mintavételezőket, s konstanssal szorzott PT_I - tagokat tartalmazó tagcsoportot. Az előbbi általános alakban megadott átviteli függvények esetében a leszarmaztatott impulzus-átviteli függvények általános alakja az alábbi:

$$Y_i(z) = \left[k_{i1} \frac{1 - e^{-\frac{T}{T_1}}}{z - e^{-\frac{T}{T_1}}} + k_{i2} \frac{1 - e^{-\frac{T}{T_2}}}{z - e^{-\frac{T}{T_2}}} + k_{i3} \frac{1 - e^{-\frac{T}{T_3}}}{z - e^{-\frac{T}{T_3}}} \right] \quad (32)$$

(ahol T - mintavételi idő; $i=1,2,3,4,5$),

s melyekbe a konstansokat behelyettesítve nyerjük a lakótéri dinamikákat is tartalmazó impulzus-átviteli függvényeket:

$$Y_1(z) = \left[0,316731 \frac{1-e^{-0,00281062T}}{z-e^{-0,00281062T}} + 0,162907 \frac{1-e^{-0,000556407T}}{z-e^{-0,000556407T}} + 0,178098 \frac{1-e^{-7,5319310^{-6} \cdot T}}{z-e^{-7,5319310^{-6} \cdot T}} \right] \quad (33)$$

$$Y_2(z) = \left[-0,0254613 \frac{1-e^{-0,00281062T}}{z-e^{-0,00281062T}} + 0,128091 \frac{1-e^{-0,000556407T}}{z-e^{-0,000556407T}} + 0,03863 \frac{1-e^{-7,5319310^{-6} \cdot T}}{z-e^{-7,5319310^{-6} \cdot T}} \right] \quad (34)$$

$$Y_3(z) = \left[0,0967932 \frac{1-e^{-0,00281062T}}{z-e^{-0,00281062T}} + 0,049785 \frac{1-e^{-0,000556407T}}{z-e^{-0,000556407T}} + 0,05443 \frac{1-e^{-7,5319310^{-6} \cdot T}}{z-e^{-7,5319310^{-6} \cdot T}} \right] \quad (35)$$

$$Y_4(z) = \left[-40,1928 \frac{1-e^{-0,00281062T}}{z-e^{-0,00281062T}} - 20,6727 \frac{1-e^{-0,000556407T}}{z-e^{-0,000556407T}} - 22,6004 \frac{1-e^{-7,5319310^{-6} \cdot T}}{z-e^{-7,5319310^{-6} \cdot T}} \right] \quad (36)$$

$$Y_5(z) = \left[-37348,5 \frac{1-e^{-0,00281062T}}{z-e^{-0,00281062T}} + 187894 \frac{1-e^{-0,000556407T}}{z-e^{-0,000556407T}} + 566647 \frac{1-e^{-7,5319310^{-6} \cdot T}}{z-e^{-7,5319310^{-6} \cdot T}} \right] \quad (37)$$

Az előbbi, az irányítandó objektumra (lakótér), mint szakaszra jellemző impulzus-átviteli függvényeket a közös nevezőre-hozás során polinom alakban felírva, a nyert impulzus-átviteli függvények alakja az alábbi:

$$Y_i(z) = \frac{\left[k_{i1} \left(1 - e^{-\frac{-T}{T_1}} \right) + k_{i2} \left(1 - e^{-\frac{-T}{T_2}} \right) + k_{i3} \left(1 - e^{-\frac{-T}{T_3}} \right) \right] z^2}{z^3 - \left(e^{-\frac{-T}{T_1}} + e^{-\frac{-T}{T_2}} + e^{-\frac{-T}{T_3}} \right) z^2 + \left(e^{-\frac{-T}{T_1}} e^{-\frac{-T}{T_2}} + e^{-\frac{-T}{T_1}} e^{-\frac{-T}{T_3}} + e^{-\frac{-T}{T_2}} e^{-\frac{-T}{T_3}} \right) z + e^{-\frac{-T}{T_1}} e^{-\frac{-T}{T_2}} e^{-\frac{-T}{T_3}}} +$$

$$+ \frac{\left[k_{i1} \left(e^{-\frac{-T}{T_2}} + e^{-\frac{-T}{T_3}} \right) \left(1 - e^{-\frac{-T}{T_1}} \right) + k_{i2} \left(e^{-\frac{-T}{T_1}} + e^{-\frac{-T}{T_3}} \right) \left(1 - e^{-\frac{-T}{T_2}} \right) + k_{i3} \left(e^{-\frac{-T}{T_1}} + e^{-\frac{-T}{T_2}} \right) \left(1 - e^{-\frac{-T}{T_3}} \right) \right] z}{z^3 - \left(e^{-\frac{-T}{T_1}} + e^{-\frac{-T}{T_2}} + e^{-\frac{-T}{T_3}} \right) z^2 + \left(e^{-\frac{-T}{T_1}} e^{-\frac{-T}{T_2}} + e^{-\frac{-T}{T_1}} e^{-\frac{-T}{T_3}} + e^{-\frac{-T}{T_2}} e^{-\frac{-T}{T_3}} \right) z + e^{-\frac{-T}{T_1}} e^{-\frac{-T}{T_2}} e^{-\frac{-T}{T_3}}}$$

$$\begin{aligned}
 & k_{i1} e^{\frac{-T}{T_2}} e^{\frac{-T}{T_3}} \left(1 - e^{\frac{-T}{T_1}}\right) + k_{i2} e^{\frac{-T}{T_1}} e^{\frac{-T}{T_3}} \left(1 - e^{\frac{-T}{T_2}}\right) + k_{i3} e^{\frac{-T}{T_1}} e^{\frac{-T}{T_2}} \left(1 - e^{\frac{-T}{T_3}}\right) \\
 & + \frac{\left(e^{\frac{-T}{T_1}} + e^{\frac{-T}{T_2}} + e^{\frac{-T}{T_3}} \right) z^3 - \left(e^{\frac{-T}{T_1}} e^{\frac{-T}{T_2}} + e^{\frac{-T}{T_1}} e^{\frac{-T}{T_3}} + e^{\frac{-T}{T_2}} e^{\frac{-T}{T_3}} \right) z^2 + e^{\frac{-T}{T_1}} e^{\frac{-T}{T_2}} e^{\frac{-T}{T_3}}}{z^3 - \left(e^{\frac{-T}{T_1}} + e^{\frac{-T}{T_2}} + e^{\frac{-T}{T_3}} \right) z^2 + \left(e^{\frac{-T}{T_1}} e^{\frac{-T}{T_2}} + e^{\frac{-T}{T_1}} e^{\frac{-T}{T_3}} + e^{\frac{-T}{T_2}} e^{\frac{-T}{T_3}} \right) z - e^{\frac{-T}{T_1}} e^{\frac{-T}{T_2}} e^{\frac{-T}{T_3}}}
 \end{aligned} \tag{38}$$

(ahol T_1, T_2, T_3 - az elsőrendű tagok időállandói ; T - mintavételi idő; $i = 1, 2, 3, 4, 5$).

Az előbbieken vázolt, a lakóteret jellemző „szakasz-dinamikák” alapján jellemeztük a végbemenő hőtechnikai változásokat, továbbá ezekhez illesztettük az irányítási rendszer algoritmusát, azok egyes alternatíváit.

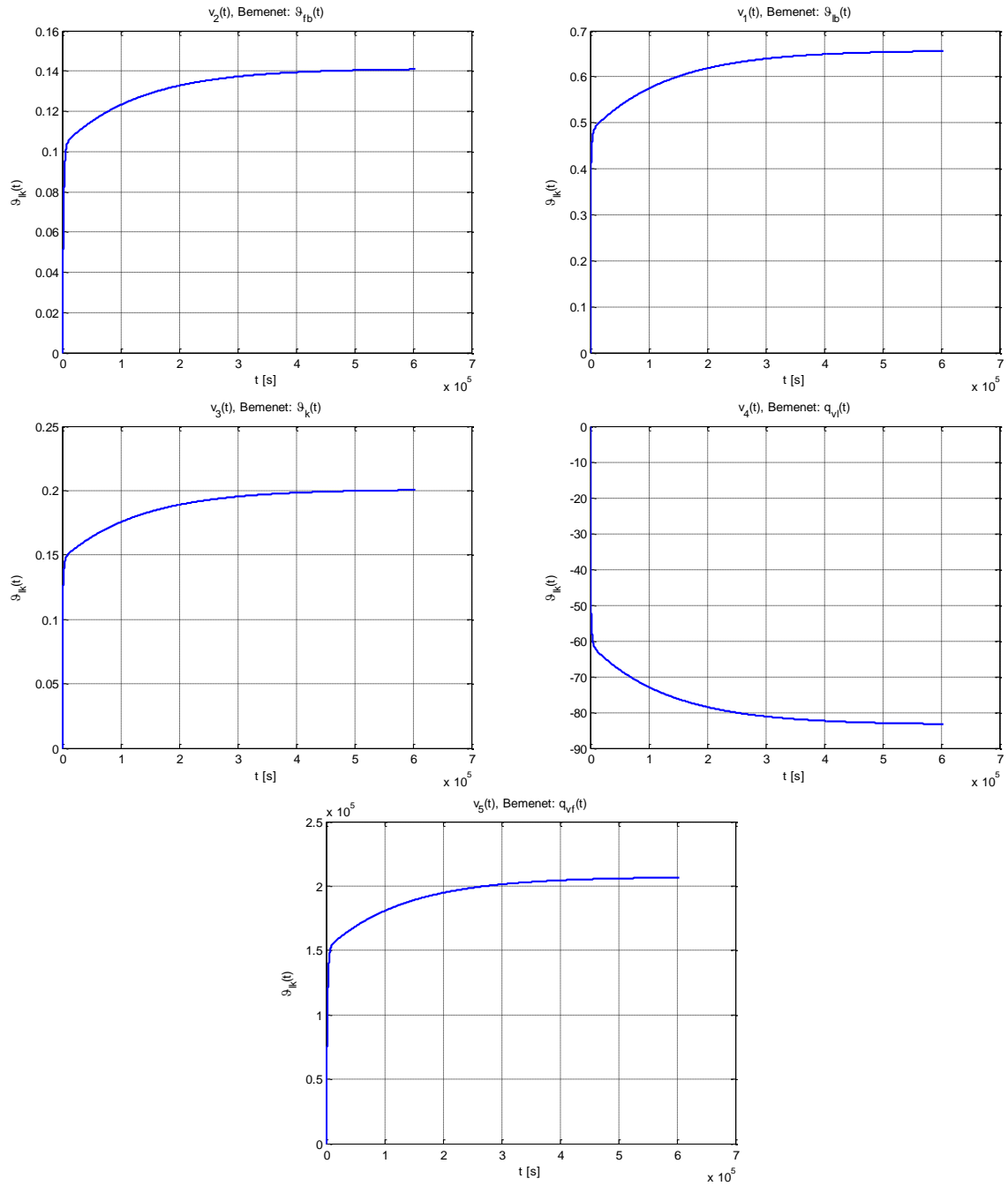
3. A hőtechnikai folyamatok számítógépes szimulációja

A struktúrájával és modell-paramétereivel rögzített folyamatdinamikai modellek vizsgálatára a -szimulációs feladatokra kiválóan alkalmas, grafikus felhasználói felülettel rendelkező - MATLAB fejlesztő-környezetben került sor. A modell matematikai leírását jelentő eredeti, nemlineáris differenciál-egyenletek (nemlineáris modell), illetve a megelőző fejezetben leírt munkaponti linearizálással linearizált egyenletek (lineáris modell) adott feltételekre történő megoldásán, a különböző üzemállapotok, beavatkozások, zavarások szemléltetésén túlmenően a modell-megoldások összevetésére, a tranziens lefutások közös koordináta-rendszerben történő ábrázolására is sor került. Utóbbi, lineáris modell ugyan nem számol az időfüggő változók szorzatával (ezeket „másodrendűen kicsiny” mennyiségként tekinti), de a lakótér fűtési időszakára definiált paraméter-beállítások, dinamikák, kisfrekvenciás időbeli változások esetén, s ezek szimulációja során nem mutatott számottevő eltérést a nemlineáris modelltől, az állapotteres leírással [5] nyert megoldásoktól.

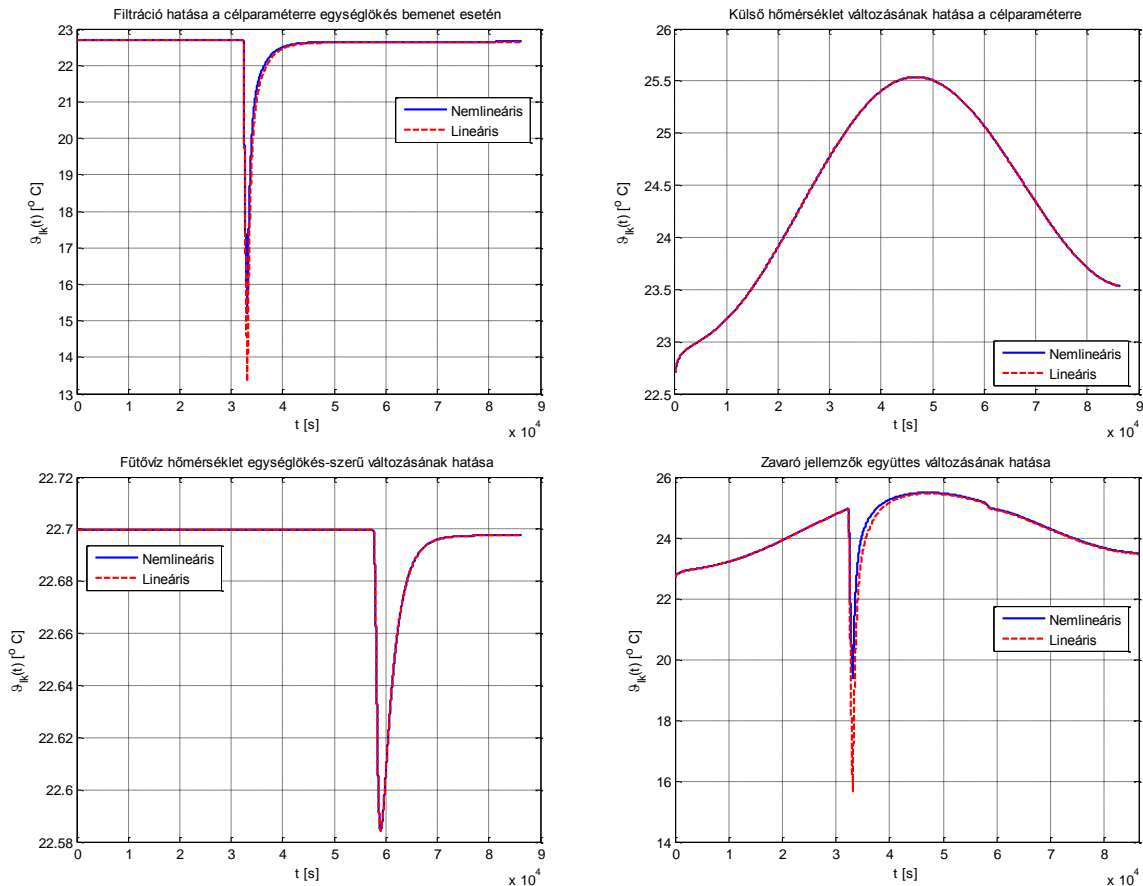
Az azonos lakótéri objektumra felállított, lineáris és a nemlineáris modellek szimulálásával nyert, a működtetés során jelentkező zavaroknak, irányított beavatkozásoknak a céljellemezőre kifejtett hatását bemutató tranziensek a mellékelt futtatásokon követhetők. Ezeken megfigyelhető a zavarok hatása külön-külön és együttesen is, valamint az irányítandó objektum önálló, ill. különböző szabályozó algoritmussal történő működtetése. A vizsgált objektumra a kétféle (nemlineáris, lineáris) modell-leírás és a - tranziensek lefutásaként szemléltethető - megoldásuk közötti csekély különbözőség indokolja, hogy a további épület-energetikai vizsgálatoknál, a több, egymástól részleges függetlenséggel működtetett lakótéri egységből felépülő épületek folyamatdinamikai modellezésénél kövessük e modellezési technikát, a tárgyalt munkaponti linearizálást.

A 2. ábrán a lakótéri klímát befolyásoló bemeneti jellemzők (időbeli ingadozásuk következtében zavarást jelentő, ill. az előbbieik kompenzálására beavatkozási lehetőséget biztosító, időfüggő változók) ugrásszerű vizsgáló-jelként való megváltoztatására adott, a légtéri hőmérséklet alakulásában jelentkező válasz-függvények – átmeneti függvényekként – egymástól elkülönített dinamikákat szemléltetnek.

A 3. ábrán látható, hogy az előbbi, a lakótéri klímát alapvetően meghatározó bemenetek időbeli jellegét, alakulását a működtetett rendszerben tapasztalható tényleges változtatásokhoz igazodva rögzítettük. Így a környezeti hőmérséklet napi, periodikus jelleggel történő ingadozását $\pm 4 \cdot C^\circ$ amplitúdójú szinusz-függvénnyel közelítettük, a véges időtartamig tartó szellőztetés és a lakás egyéb melegvíz-fogyasztásának belépésével jelentkező fűtővíz-hőmérséklet csökkenés ($3 \cdot C^\circ$) jellegét négyyszöglökésként definiáltuk. Az előbbiekre adott, a légtéri hőmérsékletben jelentkező válaszok egyedi és együttes hatása követhető a 3. ábra egyes diagramjain.

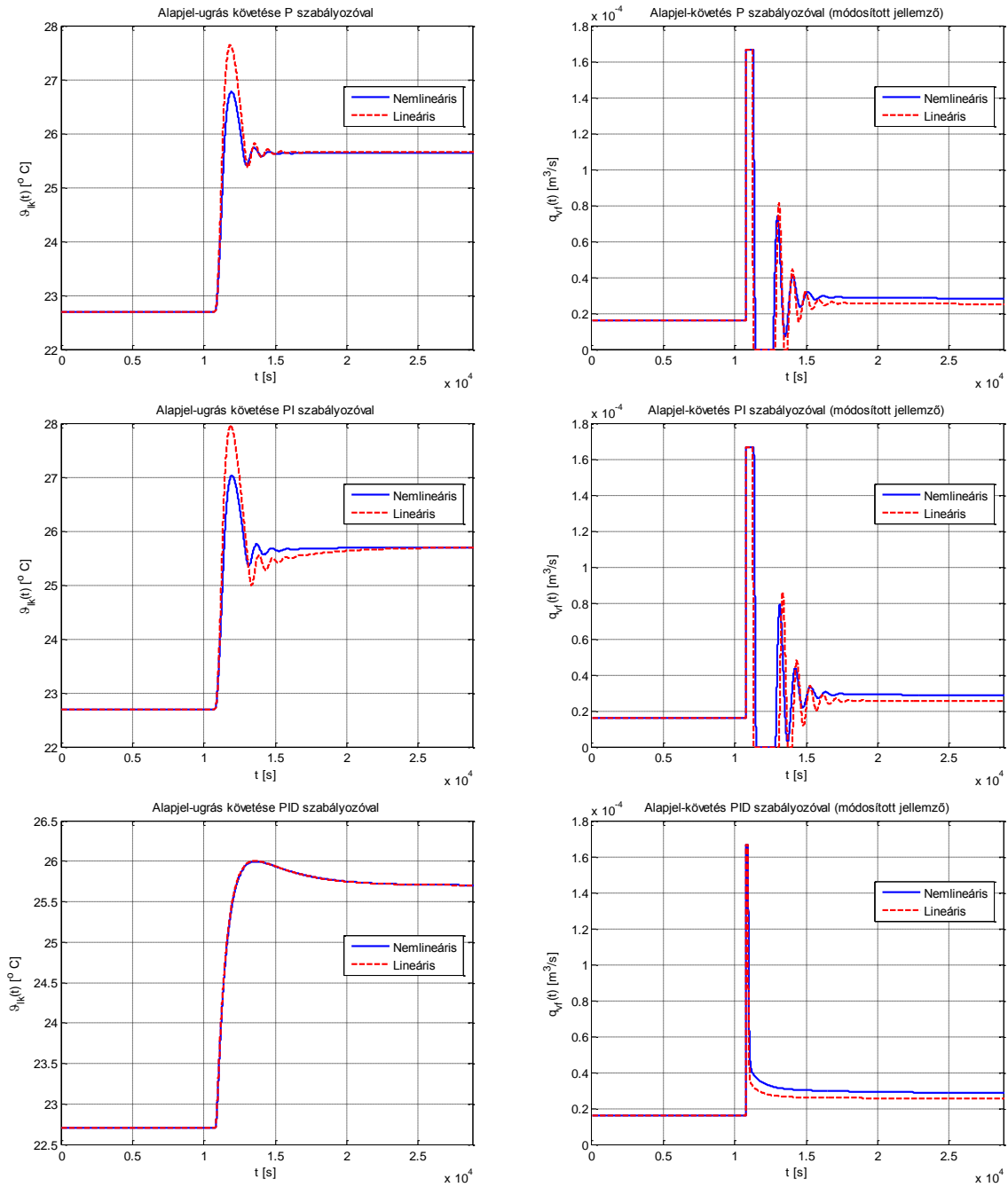


2. ábra. Zavarás-ugrás ill. módosított jellemző-ugrás bemenetre adott válasz-függvények: $v_1(t)$, $v_2(t)$, $v_3(t)$, $v_4(t)$, ill. $v_5(t)$.



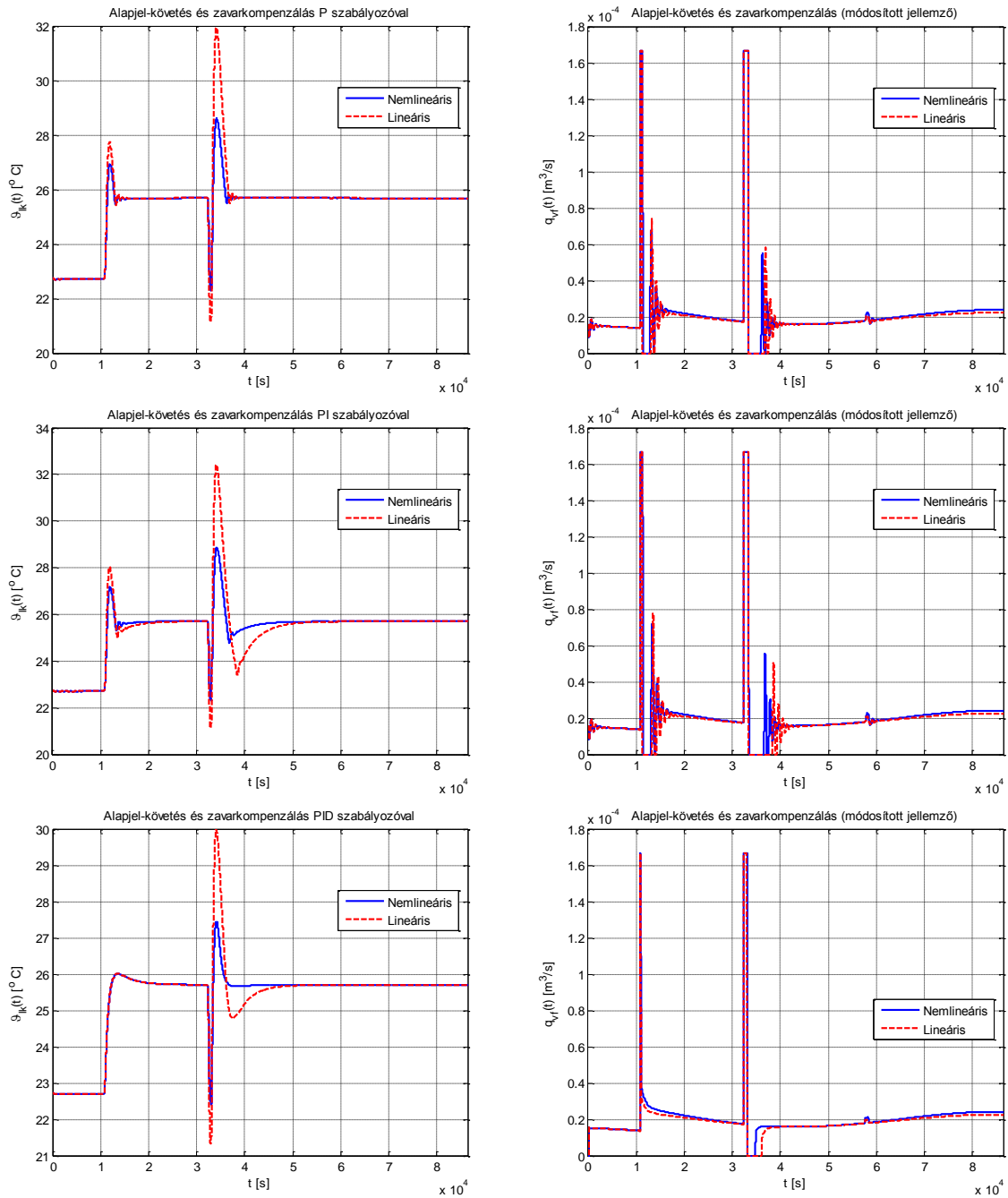
3. ábra. A különböző zavarások tipikus jelzerű változásának hatása a légtér hőmérsékletére a lakótér szabályozás-nélküli üzemeltetése során (a környezeti hőmérséklet napi periodikus változása szinuszfüggvénnyel közelítve).

A 4. ábrán zavarás-mentes állapotban (bemeneti jellemzők munkaponti értékükön rögzítve), a különböző szabályozó algoritmussal (P -, PI - PID -) működtetett lakótéri objektum alapjelkövetésre (a beállítandó lakótér-hőmérséklet $3 \cdot C^\circ$ -os ugrásszerű növelésére) vonatkozó válasz-függvényei szerepelnek a tényleges lakótéri hőmérséklet és a szabályozó-szelep nyitását követő módosított jellemző (fűtővízáram térfogatsebessége) tranzienst lefutásának szemléltetésével. A szabályozó algoritmus beállított paramétereit (A_R – a P -szabályozó arányossági átviteli tényezője; A_I – az integráló szabályozó integrálási átviteli tényezője; A_D , T^* – a valóságos differenciáló elem (DT_I jelátviteli tag) differenciálási tényezője, időállandója) nem az optimális beállítást (pl. minimális túllendülés, ill. beállási idő) jelentették, hogy jobban láthatók legyenek a lineáris, ill. a nemlineáris modell-megoldások között mutatkozó esetleges eltérések.



4. ábra. Alapjel-ugrás követése szabályozott lakótérben különböző szabályozó algoritmusok alkalmazása esetén (szabályozó paraméterek: $A_R=2 \cdot 10^{-4} \text{ m}^3/\text{sK}$; $A_I=5,4 \cdot 10^{-8} \text{ s}^{-1}$; $A_D=0,117 \text{ s}$; $T^*=0,3145 \text{ s}$)

Az 5. ábrán az előzőekben közölt szabályozó-paraméter beállítással és szabályozó algoritmussal (P-, PI- PID-) működtetett lakótéri objektum - ugyancsak a korábbiakban értelmezett alapjel-ugrásra és zavarójel-bemenetre vonatkozó - együttes válasz-függvényei szerepelnek a tényleges lakótéri hőmérséklet és a módosított jellemző transzienseinek bemutatásával.



5. ábra. Az alapjel-követés (légtér-hőmérséklet alapértékének ugrásszerű megváltoztatására adott válasz) és az egymástól függetlenül fellépő zavarok kompenzálása különböző szabályozó algoritmusokkal (a környezeti hőmérséklet napi periodikus változása szinuszfüggvénnyel közelítve; szabályozó paraméterek: $A_R=2 \cdot 10^{-4} \text{ m}^3/\text{sK}$; $A_I=5,4 \cdot 10^{-8} \text{ s}^{-1}$; $A_D=0,117 \text{ s}$; $T^*=0,3145 \text{ s}$)

4. Konklúzió

A szimulációs futtatások igazolták a lineáris és a nemlineáris modell-megoldások közötti hasonlóságot. Így a lakótér egyes működési módjainak, üzemállapotainak a - bemutatott és alkalmazott munkaponti linearizálás eredményeként - nyert matematikai modellekkel történő leképezése, ezek segítségével az irányítási stratégiák felállítása, s az irányított rendszer szimulációja viszonylag egyszerű matematikai módszerekkel és számítógépes apparátussal volt elvégezhető. Célunk továbbra is olyan, intelligens épületirányítási rendszer megtervezése, mely a lakótéri egyedi, sajátos dinamikák alapján építi fel a többlakásos, többszintes lakóépület topológiáját, könnyen módosítható, s képes alkalmazkodni a helyi, lakótéri (lakásonkénti) igényekhez. A kidolgozott identifikációs módszerek és modellezési technikák remélhetően további feladatainkhoz megfelelő alapot jelentenek.

5. Köszönetnyilvánítás

A projekt a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú "Jól-lét az információs társadalomban" című pályázat keretében valósul meg.

Irodalomjegyzék

- [1] Szakonyi L.: Intelligens lakótér folyamatdinamikája, Informatika a felsőoktatásban Konferencia, Debrecen, 2014.
- [2] L. Szakonyi: Energetic model of an elementary pipe-segment of a steam-water network, *Pollack Periodica, An International Journal for Engineering and Information Sciences*. HU ISSN 1788. 1994. Akadémiai Kiadó, Budapest 2007. Vol. 2, No. 1, pp. 63-78.
- [3] L. Szakonyi - I. A. Jancsár - Z. Sari: Energetic model for an elementary unit of a steamnetwork, *Pollack Periodica, An International Journal for Engineering and Information Sciences*. HU ISSN 1788. 1994. Akadémiai Kiadó, Budapest 2006. Vol. 1, No. 3, pp. 91-102.
- [4] Szakonyi L. - Jancskárné A. I.: Folyamatirányítás (Mintavételes szabályozási rendszerek), *PTE PMMK jegyzet*, Pécs, 2002
- [5] A. V. Oppenheim, A. S. Willsky: Signals and systems, *Prentice-Hall*, 1982

Intelligens lakótér folyamatdinamikája

Process dynamics of an intelligent accommodation

Szakonyi Lajos

PTE Pollack Mihály Műszaki és Informatikai Kar

szakonyi.lajos@pmmik.pte.hu

Absztrakt: Intelligens lakótér megtervezéséhez, ellenőrző és irányítási rendszerének kifejlesztéséhez szükséges ismernünk az irányított beavatkozásoknak és az esetleges nemkívánatos zavaroknak a céljellemzők (a beállítandó termodinamikai állapotjellemzők) tényleges értékeire gyakorolt hatását. Az *a priori* modellezési technikát követve, megmaradási törvényeket felállítva - élve a munkaponti linearizálással - határozhatjuk meg az egyes működtetési alternatíváknál az operátor-tartománybeli átviteli függvényeket. A folyamatdinamikai modellek paraméterezéséhez szükség volt kísérleti identifikációra is, mely hőszigeteléssel, hőmérsékletszabályozással ellátott lakótérben lefolytatott méréseket jelentett. A definiált folyamatdinamikai modellek (radiátoros fűtés a konvekciós és sugárzásos hőhasznosítás, a légtér és falazat közötti hőtranszport, stb. figyelembevételével) harmadrendű időkéleltetéses jelátviteli tagokat eredményeztek, melyek tartó- és mintavételező szervekkel kiegészítve impulzus-átviteli függvényekkel jellemezhető tagcsoportot szolgáltatottak. Az egyes átviteli függvények az adott üzemállapotnál egyértelműen jellemzik a folyamatdinamikát, melyhez kiválasztható a megfelelő irányítási rendszer, megtervezhető az intelligens irányítási stratégia. A lakótéri egyedi, sajátos dinamikák alapján - s így e műveleti egységekből felépítve többalakos, többszintes lakóépület topológiáját képviselő energia-elosztó hálózatot - lehetővé válhat a helyi, lakótéri (lakásonkénti) igényekhez is alkalmazkodó, esetenként könnyen módosítható épületirányítási rendszer megtervezése, beüzemelése. A fejlesztés a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú "Jól-lét az információs társadalomban" című pályázat PMMIK Műszaki Informatika Tanszék III./6. „Smart Building” munkacsoportjában valósul meg.

Kulcsszavak: matematikai modellezés, folyamatirányítás, intelligens rendszerek.

Abstract: In order to develop a control system for an intelligent accommodation, one has to know the impact of the control signals and other undesired noises on the goal parameters (thermodynamic attributes). Following the *a priori* modeling technique the transfer functions corresponding to various inputs, noises and operating modes can be given on the bases of a linearized system of conservation equations. The parameters of the dynamic model had been estimated by measurement. The dynamic representation of the accommodation taking into account the heating, heat losses, filtration, and the heat capacity of walls, resulted in third order transfer functions, which can be directly applied for simulations. Each of the transfer functions describes the effect of one particular input, and based of this kind of separation, the linear approximation of the behavior of the accommodation can be modeled accurately around any operating point. This representation can enable the construction of the energetic model of a whole building, following the topology of the building, and allows the establishment of an intelligent control strategy.

Keywords: mathematical modeling, process control, intelligent systems

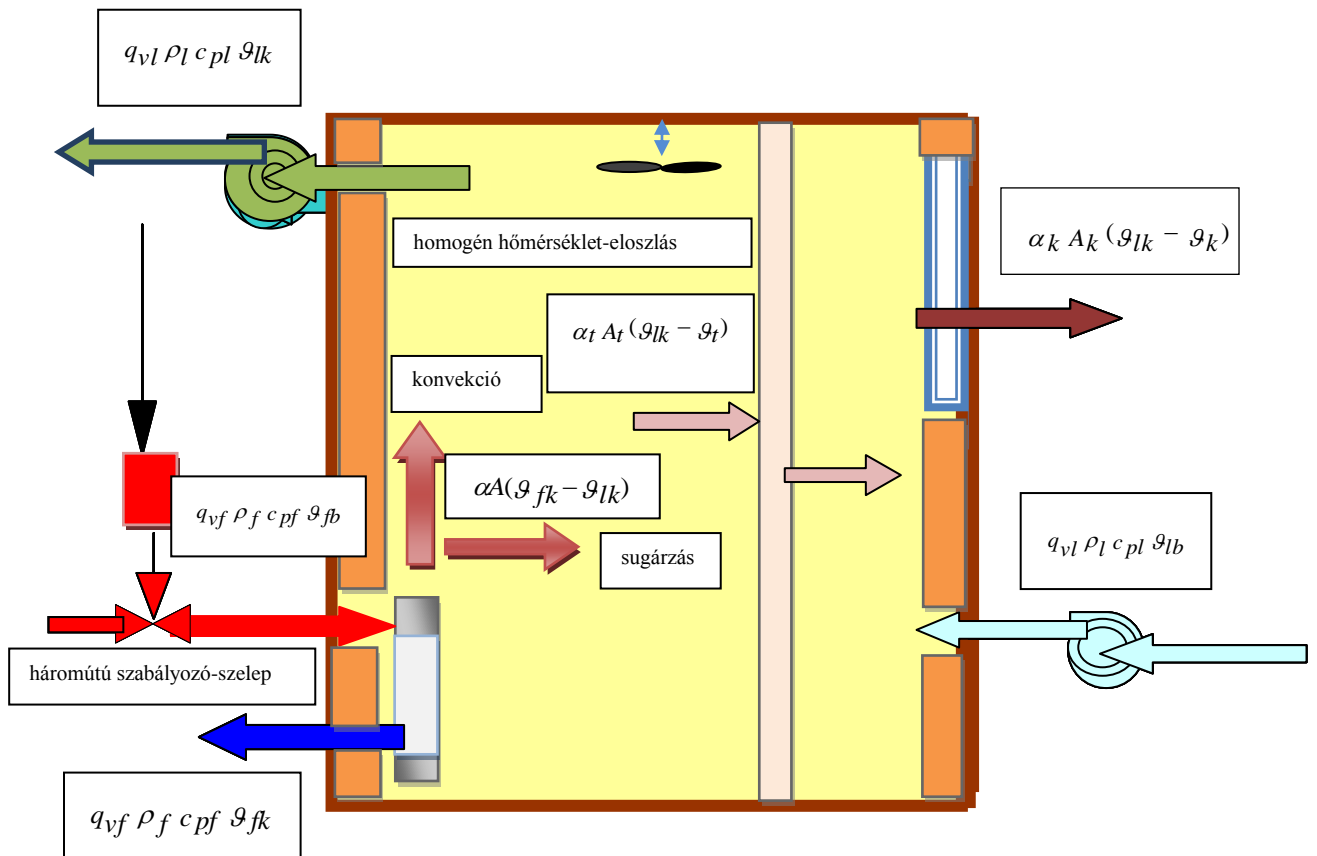
1. Bevezetés

Célunk olyan általános érvényű modell felállítása volt, melyek egy lakótér leggyakoribb „üzemállapotainál” (a fűtési rendszerben történt technológiai, ill. üzemviteli módosítások, a hőenergia-szolgáltatás zavarai, s a környezeti hőmérséklet ingadozó jellege esetén) a zavarok céljellemezőre (léghőmérsékletre) gyakorolt hatását, s ezeket egymástól függetlenül is képes jellemezni. Épületek energetikai modellezésével foglalkoznak többek között az [1] és a [2] munkák. Irányítási rendszert megtervezni az instacionárius (átmeneti, tranziens) működés ismeretében lehetséges, ezért is követelmény az átmeneti jelenségeket, az állandósult állapotok közötti időszakot leíró modellek létrehozása, a folyamatdinamika ismerete [3].

A következőkben az *a priori* modellezési technikát követve [4], műszaki-szolgáltató rendszerekre megmaradási törvényeket felállítva [5,6,7,8] - élve a munkaponti linearizálással - határozhatjuk meg az egyes üzemviteli alternatíváknál az operátor-tartománybeli átviteli függvényeket [9]. Utóbbiak inverz *Laplace*-transzformálásával a tranziensek lefutása szemléltethető. Az egyes átviteli függvények az adott üzemállapotnál egyértelműen jellemzik a folyamatdinamikát, melyhez kiválasztható a megfelelő irányítási rendszer, megtervezhető az irányítási stratégia [10,11]. E közlemény egy kiválasztott modellstruktúra esetén a modellparaméterek meghatározásának menetét kívánja bemutatni kísérleti identifikáció alkalmazásával, publikációs adatok hasznosításával, s erre épül egy kapcsolódó közlemény is a vizsgált lakótér hőtechnikai modelljének szimulációját bemutató.

A mellékelt 1. ábrához kapcsolódva - radiátoros fűtés esetén - a következő üzemállapot-alternatíváknál vizsgáltuk a lakótér dinamikáját, figyelembe véve:

- a környezeti veszteségeket;
- a szellőztetés hatását (friss levegő beszívást);
- a szellőztetés és a környezeti veszteségek együttes hatását;
- az előbbieket mellett a falazat hőkapacitásának szerepét.



1. ábra. A lakótéri transzportfolyamatok összetevői.

(q_{vf} - fűtőközeg térfogatsebesség; ρ_f - fűtőközeg sűrűség; c_{pf} - fűtőközeg fajhő; ϑ_{fb} , ϑ_{fk} - a fűtőközeg belépési, kilépési hőmérséklete; ϑ_{lk} - légtér hőmérséklet; α , A - a radiátorok hőátbocsátási tényezője, hőátadásra hasznosítható felülete; V_f fűtővíz térfogat; $\alpha_k A_k$ - a környezet felé értelmezett hőátbocsátási tényező, hőátadó felület; ϑ_k , ϑ_{lb} - a környezeti hőmérséklet, a levegő belépési hőmérséklete; ρ_l, c_{pl} - levegő sűrűség, állandó nyomáson mért fajhő; V_l - légtér térfogat; α_t - a belső válaszfalak hőátbocsátási tényezője; ϑ_t - a beltéri falak hőmérséklete; A_t, V_t - a válaszfalak felülete, térfogata; t - időkoordináta;)

2. A modell-struktúra rögzítése

A fűtőközegrre a fűtött tér belépési és kilépési csőszelvényénél értelmezhető konvekciót, a fűtött légtér felé a közvetett hőátadást, s a fűtőközeg hőáramának időbeli változását tekintetbe vevő modell a következő energia-megmaradási egyenlettel jellemezhető:

$$q_{vf} \rho_f c_{pf} \vartheta_{fb} - q_{vf} \rho_f c_{pf} \vartheta_{fk} - \alpha A (\vartheta_{fk} - \vartheta_{lk}) = V_f \rho_f c_{pf} \frac{d \vartheta_{fk}}{dt} \quad (1)$$

(ahol q_{vf} - fűtőközeg térfogatsebesség; ρ_f - fűtőközeg sűrűség; c_{pf} - fűtőközeg fajhő; $\mathcal{G}_{fb}, \mathcal{G}_{fk}$ - a fűtőközeg belépési, kilépési hőmérséklete; \mathcal{G}_{lk} - légtér hőmérséklet; α, A - a radiátorok hőátadási tényezője, hőátadásra hasznosítható felülete; V_f - folyadékterfogat; t - időkoordináta).

A lakótér légtérének hőmérsékletét írjuk le a nyílászárókon, illetve a falazaton át a környezet felé irányuló, továbbá a légtér és a belső válaszfalak közötti hőtranszportot jellemző, ugyancsak átadási árammal:

$$q_{vl} \rho_l c_{pl} \mathcal{G}_{lb} - q_{vl} \rho_l c_{pl} \mathcal{G}_{lk} + \alpha A (\mathcal{G}_{fk} - \mathcal{G}_{lk}) - \alpha_t A_t (\mathcal{G}_{lk} - \mathcal{G}_t) - \alpha_k A_k (\mathcal{G}_{lk} - \mathcal{G}_k) = V_l \rho_l c_{pl} \frac{d \mathcal{G}_{lk}}{dt} \quad (2)$$

(ahol α_k, A_k - hőátadási tényező, hőátadó felület környezet felé; \mathcal{G}_k - környezeti hőmérséklet; ρ_l, c_{pl} - levegő sűrűség, állandó nyomáson mért fajhő; V_l - légtér térfogat; α_t - a belső válaszfalak hőátadási tényezője; \mathcal{G}_t - a beltéri falak hőmérséklete; A_t - a válaszfalak felülete).

A belső falazat és a légtér közötti hőátadási folyamatokra vonatkozó mérleg átmeneti állapotban az alábbi:

$$\alpha_t A_t (\mathcal{G}_{lk} - \mathcal{G}_t) = V_t \rho_t c_t \frac{d \mathcal{G}_t}{dt} \quad (3)$$

(ahol V_t, ρ_t, c_t - a válaszfalak térfogata, sűrűsége, fajhője).

Az előbbi nemlineáris differenciál-egyenletek a munkaponti linearizálás alkalmazásával átalakíthatók lineáris egyenletekké, ha az egyes időfüggő változókat úgy értelmezzük, mint munkaponti értékük (felülvonással jelezve), s ettől való eltérésük (Δ jelölést alkalmazva) összege. Így nyerjük a következő egyenleteket a fűtőközetre:

$$\begin{aligned} & (\overline{q_{vf}} + \Delta q_{vf}) \rho_f c_{pf} (\overline{\mathcal{G}_{fb}} + \Delta \mathcal{G}_{fb}) - (\overline{q_{vf}} + \Delta q_{vf}) \rho_f c_{pf} (\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk}) - \alpha A (\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk}) + \alpha A (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) = \\ & = V_f \rho_f c_{pf} \frac{d(\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk})}{dt}, \end{aligned} \quad (4)$$

a légtérre:

$$\begin{aligned} & (\overline{q_{vl}} + \Delta q_{vl}) \rho_l c_{pl} (\overline{\mathcal{G}_{lb}} + \Delta \mathcal{G}_{lb}) - (\overline{q_{vl}} + \Delta q_{vl}) \rho_l c_{pl} (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) + \alpha A (\overline{\mathcal{G}_{fk}} + \Delta \mathcal{G}_{fk}) - \\ & - \alpha A (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) - \alpha_t A_t (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) + \alpha_t A_t (\overline{\mathcal{G}_t} + \Delta \mathcal{G}_t) - \alpha_k A_k (\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk}) + \alpha_k A_k (\overline{\mathcal{G}_k} + \Delta \mathcal{G}_k) = \\ & = V_l \rho_l c_{pl} \frac{d(\overline{\mathcal{G}_{lk}} + \Delta \mathcal{G}_{lk})}{dt}, \end{aligned} \quad (5)$$

s végül a légtér és a falazat közötti hőátzármasztásra:

$$\alpha_t A_t (\overline{\mathcal{G}}_{lk} + \Delta \mathcal{G}_{lk}) - \alpha_t A_t (\overline{\mathcal{G}}_t + \Delta \mathcal{G}_t) = V_t \rho_t c_t \frac{d(\overline{\mathcal{G}}_t + \Delta \mathcal{G}_t)}{dt}. \quad (6)$$

A munkaponti értékek alapján az állandósult állapotra vonatkozó hőmérleg a fűtőközeg:

$$\overline{q}_{vf} \rho_f c_{pf} \overline{\mathcal{G}}_{fb} - \overline{q}_{vf} \rho_f c_{pf} \overline{\mathcal{G}}_{fk} - \alpha A \overline{\mathcal{G}}_{fk} + \alpha A \overline{\mathcal{G}}_{lk} = 0, \quad (7)$$

a légtér:

$$\overline{q}_{vl} \rho_l c_{pl} \overline{\mathcal{G}}_{lb} - \overline{q}_{vl} \rho_l c_{pl} \overline{\mathcal{G}}_{lk} + \alpha A \overline{\mathcal{G}}_{fk} - \alpha A \overline{\mathcal{G}}_{lk} - \alpha_t A_t \overline{\mathcal{G}}_{lk} + \alpha_t A_t \overline{\mathcal{G}}_t - \alpha_k A_k \overline{\mathcal{G}}_{lk} + \alpha_k A_k \overline{\mathcal{G}}_k = 0, \quad (8)$$

s a beltéri falazat esetén:

$$\alpha_t A_t \overline{\mathcal{G}}_{lk} - \alpha_t A_t \overline{\mathcal{G}}_t = 0. \quad (9)$$

Képezve a kéttagú kifejezések szorzatát, eltekintve a másodrendűen kicsiny mennyiségektől ($\Delta * \Delta$), s levonva előbbiekből az állandósult állapotokra vonatkozó egyenleteket, lineáris differenciál-egyenleteket nyertünk, melyekre alkalmaztuk a *Laplace*-transzformációt. A transzformált algebrai egyenletek az alábbiak a fűtőközegre:

$$\overline{q}_{vf} \rho_f c_{pf} \mathcal{G}_{fb}(s) + \overline{\mathcal{G}}_{fb} \rho_f c_{pf} \overline{q}_{vf}(s) - \overline{q}_{vf} \rho_f c_{pf} \mathcal{G}_{fk}(s) - \overline{\mathcal{G}}_{fk} \rho_f c_{pf} \overline{q}_{vf}(s) - \alpha A \mathcal{G}_{fk}(s) + \alpha A \mathcal{G}_{lk}(s) = V_f \rho_f c_{pf} s \mathcal{G}_{fk}(s), \quad (10)$$

a légtérre ($\overline{\mathcal{G}}_{lb} = \overline{\mathcal{G}}_k = 0$):

$$\overline{q}_{vl} \rho_l c_{pl} \mathcal{G}_{lb}(s) + \overline{\mathcal{G}}_{lb} \rho_l c_{pl} \overline{q}_{vl}(s) - \overline{q}_{vl} \rho_l c_{pl} \mathcal{G}_{lk}(s) - \overline{\mathcal{G}}_{lk} \rho_l c_{pl} \overline{q}_{vl}(s) + \alpha A \mathcal{G}_{fk}(s) - \alpha A \mathcal{G}_{lk}(s) - \alpha_t A_t \mathcal{G}_{lk}(s) + \alpha_t A_t \mathcal{G}_t(s) - \alpha_k A_k \mathcal{G}_{lk}(s) + \alpha_k A_k \mathcal{G}_k(s) = s V_l \rho_l c_{pl} \mathcal{G}_{lk}(s), \quad (11)$$

valamint a belső falazatra:

$$\alpha_t A_t \mathcal{G}_{lk}(s) - \alpha_t A_t \mathcal{G}_t(s) = s V_t \rho_t c_t \mathcal{G}_t(s). \quad (12)$$

Bevezetve a *Stanton*-számként értelmezhető konstansokat és időállandókat:

$$a_1 = \frac{\alpha A}{\rho_f c_{pf} q_{vf}}, \quad a_2 = \frac{\alpha A}{\rho_l c_{pl} q_{vl}}, \quad a_4 = \frac{\alpha_k A_k}{\rho_l c_{pl} q_{vl}}, \quad a_6 = \frac{\alpha_t A_t}{\rho_l c_{pl} q_{vl}}, \quad a_7 = \frac{\alpha_t A_t}{\rho_t c_t V_t}; \quad \overline{t}_f = \frac{V_f}{q_{vf}} \text{ ill.}$$

$$\overline{t}_l = \frac{V_l}{q_{vl}} \quad (\text{az áramló fűtőközeg, ill. levegő átlagos tartózkodási ideje}), \text{ nyerjük az előbbi}$$

egyenletekből a fűtőközeg kilépési hőmérsékletére (\mathcal{G}_{fk}) rendezett formák alapján az alábbi kapcsolatokat (bevezetett jelölés: $\overline{\Delta\mathcal{G}_f} = \overline{\mathcal{G}_{fb}} - \overline{\mathcal{G}_{fk}}$):

$$\begin{aligned} & \frac{a_1}{1+a_1+st_f} \mathcal{G}_{lk}(s) + \frac{1}{1+a_1+st_f} \mathcal{G}_{fb}(s) + \frac{\overline{\Delta\mathcal{G}_f}}{q_{vf}} \frac{1}{1+a_1+st_f} q_{vf}(s) + \frac{1}{1+a_1+st_f} \overline{\mathcal{G}_{fb}} = \\ & = \frac{1+a_2+a_4+a_6+st_l}{a_2} \mathcal{G}_{lk}(s) - \frac{1}{a_2} \mathcal{G}_{lb}(s) + \frac{1}{a_2} \frac{\overline{\mathcal{G}_{lk}}}{q_{vl}} q_{vl}(s) - \frac{a_6}{a_2} \mathcal{G}_t - \frac{a_4}{a_2} \mathcal{G}_k(s) - \frac{1}{a_2} \overline{t_l} \mathcal{G}_{lk}. \quad (13) \end{aligned}$$

Az előbbi egyenletről kiemelve a légtér-hőmérséklet *Laplace*-transzformáltját, s elkülönítetten szerepeltetve az irányítandó objektum (lakótér) bemeneti-, valamint állapot-jellemzőit, az alapjel-követésre vonatkozó és az egyes zavar-átviteli függvényeket leszámaztatására alkalmas, alábbi formát nyertünk, tehát $\mathcal{G}_{lk}(s) =$

$$\begin{aligned} & \frac{(a_2s + a_2a_7)(1 + a_1 + st_f) \left[\mathcal{G}_{lb}(s) - \frac{\overline{\mathcal{G}_{lk}}}{q_{vl}} q_{vl}(s) + a_4 \mathcal{G}_k(s) \right]}{(1 + a_2 + a_4 + a_6 + st_l)(a_2s + a_2a_7)(1 + a_1 + st_f) - a_2a_6a_7(1 + a_1 + st_f) - a_1a_2(a_2s + a_2a_7)} + \\ & + \frac{a_2(a_2s + a_2a_7) \left[\mathcal{G}_{fb}(s) + \frac{\overline{\Delta\mathcal{G}_f}}{q_{vf}} q_{vf}(s) \right]}{(1 + a_2 + a_4 + a_6 + st_l)(a_2s + a_2a_7)(1 + a_1 + st_f) - a_2a_6a_7(1 + a_1 + st_f) - a_1a_2(a_2s + a_2a_7)} \quad (14) \end{aligned}$$

3. A modell-paraméterek meghatározása

A kidolgozott folyamatdinamikai modellek egyes modell-paramétereinek (üzemviteli-, konstrukciós-, ill. anyagjellemzők), az állapotjellemzők munkaponti értékeinek meghatározására kísérleti identifikációra került sor. Az előbbieket a fűtési időszakban egy hőszigeteléssel ellátott, gázkazánnal (zárt égésterű, mesterséges huzattal működő, kondenzációs készülék fűtésre és átfolyós rendszerű melegvíz készítésre; 24,9 kW hőteljesítménnyel), kombinált szabályozást biztosító időjárásfüggő szabályozóval felszerelt, társasházi, $\sim 75 \text{ m}^2$ –es lakótér esetén valósultak meg. A méréssel meghatározott jellemzők az alábbiak voltak:

- környezeti-, helyiség-, melegvíz-, előremenő, ill. visszatérő fűtővíz-hőmérséklet;
- az áramoltatott fűtővíz térfogat-sebessége, 3-útú motoros szelep nyitása;

- a gázfogyasztás pillanatnyi, ill. integrált értéke;
- a szellőztetést biztosító frisslevegő térfogatsebessége;
- a légtér, a külső és a beltéri falazat, a fűtőrendszer folyadékterének jellemzői.

A modellépítésnél felhasznált - működtetéssel, konstrukcióval, hőszállítással kapcsolatos - főbb jellemzők az alábbiak szerint kerültek rögzítésre:

ρ_f - fűtőközeg sűrűség :**989,82 kg/m³**; c_{pf} - fűtőközeg fajhő :**4176 J/kg**; α - hőátbocsátási tényező (a fűtővíz-légtér között):**10,5 W/m² K**; A – hőleadók hatásos felülete : **16,741 m²**; V_f – fűtőközeg folyadékterfogat :**0,076455 m³**; α_k - hőátbocsátási tényező (lakóter-környezet) :**0,586 W/m² K**; A_k - hőátadó felület környezet felé :**116,58 m²**; ρ_l - levegő-sűrűség :**1,293 kg/m³**; c_{pl} - levegő állandó nyomáson mért fajhő :**1012 J/kg K**; V_l – légtér térfogat:**174,1m³**; α_t - hőátadási tényező (légtér-belső falazat) :**2,72 W/m² K**; V_t - a beltéri válaszfalak térfogata :**5,48 m³**; A_t – válaszfalak felülete :**45,63 m²**; c_t , ill. ρ_t - a téglafajhője :**880 J/kg K**, ill. sűrűsége :**2500 kg/m³**.

A következőkben a (14) eredő átviteli függvényben szereplő, konstansként tekintett, hőátadással kapcsolatos jellemzők meghatározásának (becslésének) menetét tárgyaljuk.

Radiátorok hőátbocsátása (hőátvitel a fűtővíz, ill. légtér között)

Az a_1 Stanton-szám meghatározása

Az 5 darab, különböző osztás-számú, azonos (0,6 m) magasságú, két konvektor-lemezzel szerelt, *DUNAFERR-LUX-uNI-DK 22* típusú acél lapradiátoroknak összesített hőteljesítménye katalógusból vett adatok összesítése alapján : *10,116 kW*. Az előbbi hőteljesítmény-érték *90/70/20 °C* hőfoklépcső esetén érvényes, névleges adat. Az identifikációs mérések során átlagosan a fűtővíz előremenő hőmérsékletére $\overline{\vartheta_{fb}} = 55 °C$, visszatérő hőmérsékletére $\overline{\vartheta_{fk}} = 37 °C$ adódott. Az előbbieket számtani átlagát tekinthetjük a

hőátadó felületek közepes hőmérsékletének : $\mathcal{G}_{rad} = 46\text{C}^\circ$. A légtérben mért $\overline{\mathcal{G}_{lk}}$ hőmérséklet 21C° átlagértékkel került rögzítésre (a hőszugárzással kapcsolatos számításoknál ez az érték lesz a térhatároló felületek eredő közepes hőmérséklete is). A radiátorok tényleges hőleadásának átszámítására szolgáló összefüggés az alábbi [12] :

$$\dot{Q} = \dot{Q}_{60} \left[\frac{\Delta \mathcal{G}_{köz}}{60} \right]^n \quad (15)$$

(ahol \dot{Q} - a tényleges üzemeltetési körülményekre átszámított hőteljesítmény; \dot{Q}_{60} - a 60C° névleges közepes hőmérséklet-különbséghez tartozó névleges hőteljesítmény (katalógus-adat); n - a radiátor típusától, magasságától függő állandó, kettős konvektorlemezzel ellátott (DK-típusú) lapradiátor esetén javasolt érték : 1,33).

A tényleges közepes hőmérséklet-különbség a következőképpen került meghatározásra:

$$\Delta \mathcal{G}_{köz} = \frac{\overline{\mathcal{G}_{fb}} + \overline{\mathcal{G}_{fk}}}{2} - \overline{\mathcal{G}_{lk}} = \frac{55\text{C}^\circ + 37\text{C}^\circ}{2} - 21\text{C}^\circ = 25\text{C}^\circ. \quad (16)$$

A (15) összefüggésbe történő behelyettesítéssel a tényleges hőteljesítmény:

$$\dot{Q} = 10116 \left[\frac{25}{60} \right]^{1,33} \text{W} = 3157,3874\text{W}. \quad (17)$$

A nevezett radiátor-típusnál a 90/70/20 hőfoklépcső esetén katalógusból vett, hosszegységre vonatkozó névleges hőteljesítmény : $\dot{q}_{60} = 2351 \frac{\text{W}}{\text{m}}$. Az előbbi érték figyelembevételével a tényleges üzemállapotra átszámított, hosszegységre vonatkoztatott hőteljesítmény:

$$\dot{q} = 2351 \left[\frac{25}{60} \right]^{1,33} \frac{\text{W}}{\text{m}} = 2351 \cdot 0,312118 = 733,79 \frac{\text{W}}{\text{m}}. \quad (18)$$

Az indokol $l_{össz}$ radiátor-hossz (melyből a tagszám, osztás-szám helyiségenként meghatározható) az alábbiak szerint számítható:

$$l_{össz} = \frac{\dot{Q}}{\dot{q}} = \frac{3157,3874\text{W}}{733,79 \frac{\text{W}}{\text{m}}} = 4,3\text{m}, \quad (19)$$

s ez viszonylag jó egyezőséget mutat az alkalmazott 5 db. radiátor teljes, tényleges hosszával. Vizsgált rendszerünkben a hőátvitel három alapvető fajtája (hővezetés, konvekció, hőszugárzás) egyidejűleg fordult elő. A radiátoros fűtésnél a hőt a fémfal mellett turbulens

áramlásban lévő folyadékból a légtérbe a radiátorlemezen át kell eljuttatni. Az áramló fűtőközeg és a fal érintkezési felületénél létrejövő lamináris jellegű filmrétegben alakul ki a belső folyadéktömeg és falfelület közötti hőmérsékletesés döntő hányada. A radiátorlemez külső felülete és az ezzel érintkező levegőréteg határánál ugyancsak vékony filmréteg képződik. A légtérbe irányuló hőátvitel hőmérséklet-különbségből eredő sűrűségkülönbséget, utóbbi a légnemű közeg elmozdulásával járó természetes konvekciót (szabadáramlást) eredményez. A turbulens, ill. lamináris fluidumok filmvastagsága, a filmek határfelületén a hőmérséklet azonban nem, ill. nehezen mérhető. Ezért a határrétegekben megvalósuló hőátvitel számszerű jellemzésére

- kísérletileg meghatározott α_k külső-, ill. α_b belső-oldali hőátadási tényezőt,
- a csőfalon történő hővezetést meghatározó λ_a hővezetési tényezőt, ill.
- a δ_a falvastagságot

figyelembevevő k hőátbocsátási (hőátviteli) tényezőt alkalmaznak az alábbiak szerint:

$$k = \frac{1}{\frac{1}{\alpha_b} + \frac{\lambda_a}{\delta_a} + \frac{1}{\alpha_k}} \quad (20)$$

(ahol λ_a - acél-fal hővezetési tényező; δ_a - radiátor falvastagság; α_k - külső-, ill. α_b - belső-oldali hőátadási tényező).

Esetünkben a radiátor acéllemezből készült falának vastagsága $1,25 \text{ mm}$, a konvektor-lemezek vastagsága $0,4 \text{ mm}$. A DK - jelű radiátorok szélessége csak 97 mm . A függőleges vízcsatornák osztása 50 mm , így $l_{\text{össz}}=4,55 \text{ m}$ összes hosszúság esetén ez $2 \times 91 \text{ db}$ függőleges vízcsatornát jelent. Radiátorok működtetése esetén a hőátbocsátási tényezőben a domináló, az előbbi összefüggés nevezőjében harmadikként szereplő, külső-oldali hőátadási tényezőt tartalmazó tag. A hőleadók hőteljesítményének meghatározására az alábbi összefüggés szolgál:

$$\dot{Q} = kA\Delta\vartheta_{köz} \quad (21)$$

(ahol \dot{Q} - hőteljesítmény; k - hőátbocsátási tényező; A - hatásos hőátadó felület; $\Delta\vartheta_{köz}$ - a hőleadó közepes felületi hőmérséklete és a felvevő közeg (levegő) között értelmezett hőmérséklet-különbség, továbbiakban közepes hőmérséklet-különbség).

A radiátorok hőleadása konvekcióval és sugárzással valósul meg. A szakirodalomban fellelhetők a konvektív és a sugárzásos hőleadás meghatározására, megoszlási arányuk

becslésére, számítására vonatkozó összefüggések. A külső-oldali hőátadási tényező megadható konvekciós és sugárzásos összetevők összegeként:

$$\alpha_k = \alpha_c + \alpha_s \quad (22)$$

(ahol α_k , α_c , α_s - külső- (levegő)-oldali eredő, konvekciós, sugárzásos hőátadási tényező).

A konvekciós hőátadás mértékét számos jellemző (v - sebesség, c_p - fajhő, βg - köbös hőtágulási tényező · gravitációs gyorsulás), l - hossz, $\Delta\vartheta$ - hőmérséklet-különbség, η - viszkozitás, ρ - sűrűség, λ - hővezetési tényező) aktuális értéke befolyásolja. A dimenzióanalízis alkalmazásával felderíthetők a változók közötti összefüggések, s minimalizálható a változók száma. Tehát az alábbi egyenletben a φ hőáramsűrűség $QL^{-2}\tau^{-1}$ (alapmennyiségek: Q - hőmennyiség; L - hosszúság; τ - idő), s az előbbi független változók behelyettesítésével nyert hatványszorzat dimenziójának egyezőségét feltételezve:

$$\varphi = (\textit{konst.})v^a c_p^b (\beta \cdot g)^c l^d \Delta\vartheta^e \eta^f \lambda^g \rho^h, \quad (23)$$

az egyenlet két oldalán szereplő alapmennyiségek kitevőit összevetve, s rendezve nyerjük a hőáramlás Nusselt-típusú függvényeit:

$$\varphi = (\textit{konst.}) \frac{\lambda \Delta\vartheta}{l} \left(\frac{lv\rho}{\eta} \right)^a \left(\frac{c_p \eta}{\lambda} \right)^b \left(\frac{\beta g l^3 \rho^2 \Delta\vartheta}{\eta^2} \right)^c = \alpha \Delta\vartheta. \quad (24)$$

Az előbbi függvénykapcsolatot kissé átrendezve, s a hasonlóságelmélet alapján bevezetett dimenziómentes mennyiségek szokásos jelölését alkalmazva:

$$Nu = \frac{l\alpha}{\lambda} = (\textit{konst.}) Re^a Pr^b Gr^c \quad (25)$$

(ahol Nu - az átadásos és a vezetési hőáram hányadosaként értelmezett *Nusselt-szám*; Re - a tehetetlenségi erő és a belső súrlódási erő hányadosaként ismert *Reynolds-szám*; Pr - a *Pecllet-szám* (a konvektív és vezetési hőáram viszonya) és a *Reynolds-szám* arányaként adódó *Prandtl-szám*; Gr - a térfogategységre eső felhajtóerő és a belső súrlódási erő viszonyaként megadható *Grashof-szám*).

A (25) összefüggésben szereplő hatványkitevők és a konstans értékei függenek az áramlási viszonyoktól, a vizsgált objektum szerkezeti kialakításától, az anyagjellemzőktől, az üzemviteli paraméterektől, tehát a hőátvitel körülményeitől. A Nusselt-típusú függvényeknek a szakirodalomban [13], [14] számos változata megtalálható. Lapradiátorok esetén a levegő-

oldali konvekciós hőátadási tényező meghatározására – függőleges, sík falak mentén kialakuló, szabadkonvekciós, határtalan térben kialakuló hőátadást feltételezve – a (25) összefüggésnek megfelelő Nusselt-típusú egyenlet alábbi változata:

$$Nu = \frac{l\alpha}{\lambda} = (\text{konst.})(GrPr)^m = (\text{konst.}) \left(\frac{\beta g l^3 \rho^2 \Delta \vartheta c_p \eta}{\eta^2 \lambda} \right)^m \quad (26)$$

került alkalmazásra, s $10^9 \leq Gr Pr \leq 10^{12}$ esetén:

$$Nu = 0,13 (Gr Pr)^{0,33}. \quad (27)$$

A légtérben mért és a szakirodalomban található gázjellemzőket behelyettesítve a dimenziómentes számok kifejezésébe:

$$Gr Pr = \frac{27,49 \cdot 10^{-3} \cdot 9,81 \cdot 0,6^3 \cdot 25 \cdot 1008 \cdot 1,87 \cdot 10^{-5}}{(1,68 \cdot 10^{-5})^2 \cdot 0,024} = 4,0524 \cdot 10^9, \quad (28)$$

a Nusselt-számra adódó érték:

$$Nu = 0,13 \cdot (4,0524 \cdot 10^9)^{0,33} = 192,526. \quad (29)$$

Az előbbiből, s a (26) összefüggésből adódóan a konvekciós hőátadási tényező, mint összetevője a teljes külső-oldali hőátadási tényezőnek:

$$\alpha_c = \frac{\lambda \cdot Nu}{l} = \frac{0,024 \frac{W}{mK} \cdot 192,526}{0,6m} = 7,7 \frac{W}{m^2 K}. \quad (30)$$

A konvekciós hőátadás értelmezése után tekintsük a sugárzásos (radiációs) hőátadás alaptörvénye (Stefan-Boltzmann törvény) alapján a testek által kibocsátott, ill. elnyelt hőenergia számítására vonatkozó összefüggést:

$$Q = \sigma \cdot A \cdot T^4 \quad (31)$$

(ahol Q - hőenergia [J]; A – a sugárzásnál értelmezhető felület [m^2]; σ - Stefan-Boltzmann állandó: $5,67 \cdot 10^{-8} \left[\frac{J}{m^2 K^4} \right]$; T - hőmérséklet [K]).

Az előbbi összefüggésben értelmezett energiát az ún. abszolút fekete (tökéletesen sugárzó) test lenne képes kisugározni, a valóságban létező „szürke” testek - ε feketeségi fokuknak megfelelően - az előbbi hőenergia-értéknek csak bizonyos hányadát emittálják:

$$C = \varepsilon C_f \quad (32)$$

(ahol C - az ε feketeségi fokú ($0 \leq \varepsilon \leq 1$) test sugárzási együtthatója; C_f - az abszolút fekete test sugárzási együtthatója: $5,67 \cdot \frac{W}{m^2 K^4}$).

Két különböző hőmérsékletű felület közötti hőcsere az alábbi összefüggéssel jellemezhető:

$$Q = \frac{C_k}{\left(\frac{T_1}{100}\right)^4 - \left(\frac{T_2}{100}\right)^4} \quad (33)$$

(ahol C_k - az emittáló (1) és az elnyelő (2) felületre vonatkozó, kölcsönös sugárzási együttható; T_1 - a hőt sugárzó felület abszolút hőmérséklete; T_2 - a hőt elnyelő felület abszolút hőmérséklete).

A C_k kölcsönös sugárzási együttható számítására szolgáló összefüggés a következő:

$$\frac{1}{C_k} = \frac{1}{C_1} + \frac{A_1}{A_2 \left(\frac{1}{C_2} - \frac{1}{C_f} \right)} \quad (34)$$

(ahol C_1 , ill. C_2 - a hőt sugárzó, ill. elnyelő felület sugárzási együtthatója, A_1 , ill. A_2 - a hőt sugárzó, ill. elnyelő felület).

Feherre festett radiátorok felületének feketeségi foka $40-90^\circ C$ hőmérséklet-tartományban:

$0,92$, s így a számítható sugárzási együttható: $C_1 = 5,22 \cdot \frac{W}{m^2 K^4}$. Esetünkben $A_1 \ll A_2$,

ugyanis a hőt sugárzó radiátorfelület elhanyagolható a hőt felvevő, a környező légtérben értelmezhető térhatároló felületekhez képest, ezért $C_k = C_1$. A térhatároló felületek hőmérsékletét azonosnak tekintve a légtér hőmérsékletével ($T_2 = T_{lk}$), a sugárzással leadott hőáram számítására szolgáló összefüggés:

$$\dot{Q} = C_k A \left[\left(\frac{T_{rad}}{100} \right)^4 - \left(\frac{T_{lk}}{100} \right)^4 \right] \quad (35)$$

(ahol A - a radiátor hatásos felülete; T_{rad} , ill. T_{lk} - a hőleadó felület közepes abszolút hőmérséklete, ill. a légtér átlagos abszolút hőmérséklete; \dot{Q} - hőteljesítmény).

Az előbbieken alapján az α_s sugárzásos, levegő-oldali hőátadási tényező meghatározására az alábbi összefüggés szolgál:

$$\alpha_s = C_k \frac{\left(\frac{T_{rad}}{100} \right)^4 - \left(\frac{T_{lk}}{100} \right)^4}{T_{rad} - T_{lk}}. \quad (36)$$

Méréseink és a szakirodalomból vett adatok alapján elvégezve a behelyettesítést:

$$\alpha_s = 5,22 \cdot \frac{W}{m^2 K^4} \frac{\left(\frac{319K}{100}\right)^4 - \left(\frac{294K}{100}\right)^4}{319K - 294K} = 6,022 \frac{W}{m^2 K}. \quad (37)$$

A (30) és a (37) képletek alapján számított eredményeket figyelembe véve, s ezeket a (22) összefüggés alapján összegezve a teljes külső (levegő)-oldali hőátadási tényezőre adódó érték:

$$\alpha_k = 7,7 \frac{W}{m^2 K} + 6,022 \frac{W}{m^2 K} = 13,722 \frac{W}{m^2 K}. \quad (38)$$

A hőteljesítmény és hőmérséklet-eloszlás, a konvekciós és a sugárzásos hőleadás elkülönítésére, a különválasztott hőátadási tényezők hely- és hőmérsékletfüggésére szolgáló összefüggéseket közöl a [15] publikáció. A [16] szakirodalom a konvekciós és a sugárzásos hőleadás arányát vizsgálja a teljes hőleadásban az alábbiak szerint:

$$\dot{Q} = \left[k \left(\frac{\Delta \mathcal{G}_{köz}}{\Delta \mathcal{G}_{névl}} \right)^n + (1-k) \left(\frac{\Delta \mathcal{G}_{köz}}{\Delta \mathcal{G}_{névl}} \right)^m \right] \dot{Q}_{névl}, \quad (39)$$

s konvekciónál $n = 1,25 \dots 1,33$, ill. sugárzásos hőleadásnál $m = 1,2 \dots 1,26$ kitevő tartományban a kitevők változtatása esetén elemzi a $\dot{Q}_{névl}$ névleges hőteljesítményből számított tényleges \dot{Q} teljesítményadatok közötti eltéréseket. A hőleadás e bontása esetünkben is alkalmazást nyert.

A továbbiakban elemezzük a belső-oldali (fűtővíz-oldali) hőátadási tényező meghatározását, melyet befolyásol a fűtőközeg áramoltatását biztosító, tehát kényszerített konvekciót eredményező, változtatható fordulatszámú keringtető szivattyú aktuális üzemállapota. Ebben az esetben a Nusselt-típusú függvényeket alkalmazva elsősorban a *Reynolds*- és a *Prandtl-szám* a domináló. Az előbbi Nusselt függvények egyik alkalmazható alakja csőbeni (esetünkben a lapradiátorok függőleges vízcsatornáiban megvalósuló) folyadékáramlás esetére az alábbi:

$$Nu = \frac{\alpha_b D_e}{\lambda_f} = 1,86 \cdot \left(\frac{\eta_f}{\eta_{fal}} \right)^{0,14} \left(\frac{D_e G}{\eta_f} \cdot \frac{c_{pf} \eta_f}{\lambda_f} \cdot \frac{D_e}{l} \right)^{0,33} \quad (40)$$

(ahol α_b - belső-oldali hőátadási tényező; D_e - a függőleges vízcsatorna-keresztmetszet egyenértékű átmérője; λ_f - fűtővíz hővezetési tényező; η_f , ill. η_{fal} - fűtővíz viszkozitása átlaghőmérsékleten, ill. falhőmérsékleten; G - tömegáram-sűrűség egy függőleges vízcsatornában; c_{pf} - fűtővíz fajhő; l - zavartalan úthossz; egy vízcsatorna magassága).

Az egy vízcsatornára számított tömegáram-sűrűség és keresztmetszet alapján bevezethető a q_{mf} tömegáram (például a fűtővíz szabályozószelep 20%-os átlagos nyitása az identifikációs mérések idején - munkaponti értéként - $0,0027\text{m}^3/\text{min}$ összes térfogatáramot jelentett, melynek egy vízcsatornára eső hányadát a víz sűrűségével beszorozva nyerhető a $0,00024474\text{ kg/s}$ érték). Az előbbi összefüggést átalakítva dimenziós alakra, miután elvégeztük az összevonásokat:

$$\alpha_b = 2,02 \cdot c_{pf}^{\frac{1}{3}} \lambda_f^{\frac{2}{3}} \left(\frac{\eta_f}{\eta_{fal}} \right)^{0,14} \frac{q_{mf}^{\frac{1}{3}}}{D_e l^{\frac{1}{3}}}. \quad (41)$$

Behelyettesítéssel a folyadék-oldali hőátadási tényezőre adódó érték:

$$\alpha_b = 2,02 \cdot 4176^{\frac{1}{3}} \left[\frac{J}{kgK} \right]^{\frac{1}{3}} 0,58^{\frac{2}{3}} \left[\frac{W}{mK} \right]^{\frac{2}{3}} \left(\frac{0,65 \cdot 10^{-3} \frac{kg}{sm}}{0,75 \cdot 10^{-3} \frac{kg}{sm}} \right)^{0,14} \frac{[2,4474 \cdot 10^{-4}]^{\frac{1}{3}} \left[\frac{kg}{s} \right]^{\frac{1}{3}}}{1,375 \cdot 10^{-2} m \cdot 0,6^{\frac{1}{3}} m^{\frac{1}{3}}} = 119,56 \frac{W}{m^2 K} \quad (42)$$

A (20) összefüggésbe behelyettesítve a Nusselt függvényekben szereplő dimenziómentes számok (komplexek) alkalmazásával nyert hőátviteli jellemzőket, az eredő hőátbocsátási (hőátviteli) tényezőre adódó érték:

$$k_{lr} = \frac{1}{\frac{1}{119,56 \frac{W}{m^2 K}} + \frac{0,00125m}{50 \frac{W}{mK}} + \frac{1}{13,722 \frac{W}{m^2 K}}} = 12,302 \frac{W}{m^2 K}. \quad (43)$$

Mérési eredményeink felhasználásával tehát sor került a konvekciós és a sugárzásos hőhasznosítás elkülönítésére, számszerűsítésére. A korábbiakban megnevezett publikáció (39) összefüggését követve, s a sugárzásos hőátadásnál $m = 1,25$ kitevőt rögzítve meghatározásra került a (15) és a (39) összefüggésekben szereplő n hatványkitevő.

A fűtési célokat szolgáló 5 darab lapradiátorokon túlmenően számolnunk kellett egy fürdőszobai csőradiátor és a csatlakozó csővezetékek hőleadásával is, s így a rendelkezésre

álló, névleges összes hőteljesítmény a következőképpen alakult. A lapradiátorok hosszegységre vonatkoztatott névleges hőteljesítménye ($\dot{q}_{60} = 2351 \text{ W/m}$) és a $4,55 \text{ m}$ tényleges hossz alapján:

$$\dot{Q}_{60} = 2351 \frac{\text{W}}{\text{m}} \cdot 4,55 \text{ m} = 10697,05 \text{ W} , \quad (44)$$

s így a lapradiátorok tényleges hőteljesítménye a (15) összefüggés alkalmazásával ($n = 1,3$ esetén):

$$\dot{Q} = 10697,05 \text{ W} \cdot \left(\frac{25 \text{ K}}{60 \text{ K}} \right)^{1,3} = 3427,5943 \text{ W}. \quad (45)$$

Az előbbi értékből a sugárzásos hőteljesítmény a (35) összefüggés alapján számítva, s ismerve a lapradiátorok hősugárzásnál hasznosuló, $A_{rad} = 7,027 \text{ m}^2$ felületét:

$$\dot{Q}_s = 5,22 \frac{\text{W}}{\text{m}^2 \text{ K}^4} \cdot 7,027 \text{ m}^2 \cdot 28,841 \text{ K}^4 = 1057,915 \text{ W}. \quad (46)$$

A különbözetként számított konvekciós hőteljesítmény:

$$\dot{Q}_c = \dot{Q} - \dot{Q}_s = (3427,5943 - 1057,915) \text{ W} = 2369,6793 \text{ W}. \quad (47)$$

Az előbbieket figyelembevételével a „dupla konvektoros” radiátorok vizsgálataink idején $69,13 \%$ -ban konvektív, $30,87 \%$ -ban sugárzásos hőátadással üzemeltek, ami jó egyezőséget mutat a $DK 22$ típusú *Dunaferr* radiátorokra publikált adatokkal. Az előbbi arányokat elfogadva, s a (39) összefüggést alkalmazva meghatározhatók a konvekciós és a sugárzásos hőleadást leíró tagokban szereplő kitevők értékei, s az így érvényes egyenlet az alábbi:

$$3428 \text{ W} = \left[0,691 \cdot \left(\frac{25 \text{ K}}{60 \text{ K}} \right)^{1,32} + 0,308 \cdot \left(\frac{25}{60} \right)^{1,25} \right] \cdot 10697 \text{ W} . \quad (48)$$

A fürdőszobai csőradiátor, valamint a gázkazánt és a lapradiátorokat összekötő fűtővízhálózat esetén ugyancsak számoltunk a természetes konvekcióval, valamint a hősugárzással. A következőkben határozzuk meg a lapradiátorok és a csőradiátor, csővezetékek alkotta fűtőrendszer együttes, légtér felé történő hőátbocsátását, s az előbbiekből következően a folyamatdinamikai modell egyik munkaponti értékéül szolgáló hőátbocsátási tényezőt. A (21)

összefüggés alkalmazása a lapradiátorokra és az egyéb hőforrásokra az alábbi értékeket szolgáltatja:

$$\dot{Q}_{lr} = k_{lr} A_{lr} \Delta g_{köz} = 12,302 \frac{W}{m^2 K} \cdot 11,14436 m^2 \cdot 25 K = 3427,5943 W, \quad (49)$$

illetve

$$\dot{Q}_{cs} = k_{cs} A_{cs} \Delta g_{köz} = 6,91212 \frac{W}{m^2 K} \cdot 5,5961 m^2 \cdot 25 K = 967,022 W. \quad (50)$$

A hőátadásra szolgáló összes felület $16,741 m^2$, melyből a lapradiátorok felülete (A_{lr}) 66,572 % -ot, az egyéb hőforrásoké (A_{cs}) 33,428 % -ot jelent. Az eredő hőátbocsátási tényezőt a lapradiátorok és az egyéb hőátadók felületarányokkal súlyozott tényezőiből számítva:

$$\alpha = 0,66572 \cdot 12,302 \frac{W}{m^2 K} + 0,33428 \cdot 6,91212 \frac{W}{m^2 K} = 10,5 \frac{W}{m^2 K}. \quad (51)$$

Az üzemviteli jellemzők behelyettesítésével nyert dimenziómentes szám, mely kapcsolatos a fűtőközeg és a légtér közötti hőátvitellel:

$$a_1 = \frac{\alpha A}{\rho_f c_{pf} q_{vf}} = \frac{10,5 \frac{W}{m^2 K} \cdot 16,741 m^2}{989,82 \frac{kg}{m^3} \cdot 4176 \frac{J}{kgK} \cdot 4,5 \cdot 10^{-5} \frac{m^3}{s}} = 0,945021. \quad (52)$$

Az a_2 Stanton-szám meghatározása

Az alábbi képletbe történő behelyettesítés szolgáltatja az újabb dimenziómentes számot:

$$a_2 = \frac{\alpha A}{\rho_l c_{pl} q_{vl}} = \frac{10,5 \frac{W}{m^2 K} \cdot 16,741 m^2}{1,293 \frac{kg}{m^3} \cdot 1012 \frac{J}{kgK} \cdot 0,171 \frac{m^3}{s}} = 0,785589. \quad (53)$$

A szellőztetéssel (a filtráció figyelembevételével) belépő $\overline{q_{vl}}$ hideglevegő-áram munkaponti (az identifikációs mérések idején rögzített) értékének meghatározása hőmérsékletméréssel, ugrásszerű zavarással (nyílászárók nyitása $4C^\circ$ -os hideglevegő beáramoltatásával), s az alábbi közelítő hőmérleg alkalmazásával:

$$\overline{q_{vl}} \rho_l c_{pl} (g_{lb} - g_{lk}) = V_l \rho_l c_{pl} \frac{d g_{lk}}{dt} \quad (54)$$

valósult meg. Ugyanis a $174,1 \text{ m}^3$ -es lakótérben a vizsgálat megkezdése előtti időpontban mért 21 C° légtérhőmérséklet 5 C° - kal csökkent 5 min elteltével, tehát az előbbi, a légtérre felírt, konvektív tagokat és a lokális változást tartalmazó hőmérleg egyszerűsített alakjába való behelyettesítés:

$$\overline{q_{vl}}(4 \text{ C}^\circ - 21 \text{ C}^\circ) = 174,1 \text{ m}^3 \frac{-5 \text{ C}^\circ}{300 \text{ s}} \quad (55)$$

alapján $\overline{q_{vl}} = 0,171 \frac{\text{m}^3}{\text{s}}$.

A légtér és a környezet, valamint a légtér és a falazat közötti hőátvitel

Az α_4 Stanton-szám meghatározása

Az α_k hőátbocsátási tényező egyik összetevőjeként az α_{k1} a (20) képlet mintájára az alábbi összefüggés segítségével került rögzítésre:

$$\alpha_{k1} = \frac{1}{\frac{1}{\alpha_{bel}} + \sum \frac{\delta}{\lambda} + \frac{1}{\alpha_{k\u00fcl}}} = \frac{1}{\frac{1}{23 \frac{\text{W}}{\text{m}^2 \text{K}}} + \frac{0,015 \text{ m}}{0,7 \frac{\text{W}}{\text{mK}}} + \frac{0,24 \text{ m}}{0,5 \frac{\text{W}}{\text{mK}}} + \frac{0,1 \text{ m}}{0,04 \frac{\text{W}}{\text{mK}}} + \frac{0,01 \text{ m}}{0,7 \frac{\text{W}}{\text{mK}}} + \frac{1}{8 \frac{\text{W}}{\text{m}^2 \text{K}}}} = 0,314 \frac{\text{W}}{\text{m}^2 \text{K}} \quad (56)$$

(ahol $\alpha_{bel}, \alpha_{k\u00fcl}$ - a légtéri, ill. a kültéri levegőre vonatkozó belső-, ill. külső-oldali hőátadási tényező; λ, δ - hővezetési tényező, ill. rétegvastagság a belső vakolat, a téglafalazat, a hőszigetelés és a külső műanyag-vakolat esetén).

Publikációs adatok szerint *B30-as* falazótéglából készült falazat hőátbocsátási tényezője hőszigetelés nélkül $1,5 \text{ W/m}^2 \text{K}$, s ez 5 cm -es hőszigeteléssel $0,52 \text{ W/m}^2 \text{K}$ -re, 16 cm -es szigeteléssel $0,22 \text{ W/m}^2 \text{K}$ -re csökkenthető. Előbbiek esetünkben egybeesést mutatnak a 10 cm -es hőszigeteléssel ellátott külső falazatra elvégzett számításokkal. Homlokzati, üvegezett, PVC keretszerkezettel készült nyílászárók esetén a 7/2006. (V. 24.) TNM rendelet szerinti követelményérték: $1,6 \text{ W/m}^2 \text{K}$, ezt tekinthetjük az eredő α_k hőátbocsátási tényező másik, α_{k2} jelű összetevőjeként. Fűtetlen helyiségek feletti alsó zárófödém esetén pedig a harmadik

összetevője az eredő hőátbocsátási tényezőnek publikációs adatok alapján az $\alpha_{k3} = 0,5$ W/m^2K érték. A felállított folyamatdinamikai modellekben - a környezet felé történő, s veszteségként tekinthető hőátvitelt jelentő átadásos tagban szereplő - eredő hőátbocsátási tényező a három eltérő értékű összetevő felületarányok szerinti súlyozásával került meghatározásra az alábbiak szerint:

$$\alpha_k = 0,23 \cdot 0,314 \frac{W}{m^2 K} + 0,118 \cdot 1,6 \frac{W}{m^2 K} + 0,643 \cdot 0,5 \frac{W}{m^2 K} = 0,586 \frac{W}{m^2 K} . \quad (57)$$

Ugyanis a vizsgált lakótér teljes külső síkfelülete $116,58 \text{ m}^2$, melynek összetevői a hőszigetelt falazat, az üvegezett nyílászárók és a lakótér alatti szigetetlen alsó zárófödém felülete teszi ki. A kalorikus, üzemviteli és anyag-jellemzők behelyettesítésével nyertük a külső környezet felé irányuló hőátvitellel kapcsolatos dimenziómentes számot:

$$a_4 = \frac{0,586 \frac{W}{m^2 K} \cdot 116,58 \text{ m}^2}{1,293 \frac{\text{kg}}{\text{m}^3} \cdot 1012 \frac{\text{J}}{\text{kgK}} \cdot 0,171 \frac{\text{m}^3}{\text{s}}} = 0,3053. \quad (58)$$

Az a_6 Stanton-szám meghatározása

Az α_t hőátbocsátási tényezőt 12 cm -es, kisméretű téglára publikált, $2,72 \text{ W/m}^2K$ értéként rögzítettük. A beltéri válaszfalak felülete : $A_t = 45,63 \text{ m}^2$, térfogata : $V_t = 5,48 \text{ m}^3$, az ismert paraméterek behelyettesítésével nyert újabb Stanton-szám:

$$a_6 = \frac{\alpha_t A_t}{\rho_l c_{pl} q_{vl}} = \frac{2,72 \frac{W}{m^2 K} \cdot 45,63 \text{ m}^2}{1,293 \frac{\text{kg}}{\text{m}^3} \cdot 1012 \frac{\text{J}}{\text{kgK}} \cdot 0,171 \frac{\text{m}^3}{\text{s}}} = 0,55468 \quad (59)$$

Az a_7 konstans, illetve az átlagos tartózkodási idők meghatározása

A beltéri falazat jellemzőinek ismeretében az a_7 konstans meghatározására szolgáló kifejezés a következő értéket szolgáltatja:

$$a_7 = \frac{\alpha_t A_t}{\rho_t c_t V_t} = \frac{2,72 \frac{W}{m^2 K} \cdot 45,63 \text{ m}^2}{2500 \frac{\text{kg}}{\text{m}^3} \cdot 880 \frac{\text{J}}{\text{kgK}} \cdot 5,48 \text{ m}^3} = 1,02948 \cdot 10^{-5} \text{ s}^{-1} \quad (60)$$

(ahol ρ_t , ill. c_t - a téglá sűrűsége, ill. fajhője),

melynek reciproka 27 óra. Az 5 perces időtartamot jelentő szellőztetésnél értelmezhető (\bar{t}_l), illetve a fűtővízre meghatározható (\bar{t}_f) átlagos tartózkodási idő az alábbi törtekbe történő behelyettesítéssel nyerhető:

$$\bar{t}_l = \frac{V_l}{q_{vl}} = \frac{174,1 m^3}{0,171 \frac{m^3}{s}} = 1018,129s = 0,2828h, \text{ ill.} \quad (61)$$

$$\bar{t}_f = \frac{V_f}{q_{vf}} = \frac{0,076455 m^3}{4,5 \cdot 10^{-5} \frac{m^3}{s}} = 1699s = 0,4719h. \quad (62)$$

4. Konklúzió

A modellparaméterek ismeretében a működési egyenletekből munkaponti linearizálással nyert transzformált algebrai egyenletek, s az ezekből racionális törteként adódó átviteli függvények igen előnyösen, egymástól elkülöníthető módon, külön-külön jellemezhetik az energiaellátásban jelentkező zavarok hatását az üzemvitelre. A lakótéri egyedi, sajátos dinamikák alapján, s így e műveleti egységekből felépítve többlakásos, többszintes lakóépület topológiáját képviselő energia-elosztó hálózatot a létrehozott modellek felhasználásával a későbbiekben lehetővé válhat a helyi, lakótéri (lakásonkénti) igényekhez is alkalmazkodó, esetenként könnyen módosítható, intelligens épületirányítási rendszer megtervezése.

5. Köszönetnyilvánítás

A projekt a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú "Jól-lét az információs társadalomban" című pályázat keretében valósul meg.

Irodalomjegyzék

- [1] Siddharth Goyal, Prabir Barooah: A method for model-reduction of non-linear thermal dynamics of multi-zone buildings; *Energy and buildings; Elsevier* (2011).
- [2] Emily M. Ryan, Thomas F. Sanquist: Validation of building energy modeling tools under dealized and realistic conditions; *Energy and buildings; Elsevier* (2011).
- [3] R. Mohilla - B. Ferencz: Chemical process dynamics, *Akadémiai Kiadó, Budapest, 1982.*

- [4] Árva P. - Nagy D. - Szeifert F.: Vegyipari rendszerek , *VVE jegyzet*, Veszprém, 1979.
- [5] L. Szakonyi: Energetic model of an elementary pipe-segment of a steam-water network, *Pollack Periodica, An International Journal for Engineering and Information Sciences*. HU ISSN 1788. 1994. Akadémiai Kiadó, Budapest 2007. Vol. 2, No. 1. pp. 63-78.
- [6] L. Szakonyi - I. A. Jancsár - Z. Sari: Energetic model for an elementary unit of a steamnetwork, *Pollack Periodica, An International Journal for Engineering and Information Sciences*. HU ISSN 1788. 1994. Akadémiai Kiadó, Budapest 2006. Vol. 1, No. 3. pp. 91-102.
- [7] Szakonyi L.: Infokommunikációs technológia kidolgozása és regionális hasznosítása az energiacelosztás területén, *Informatika a felsőoktatásban Konferencia*, Debrecen, 2005. Konferenciakiadvány, pp. 139. (ISBN 963 472 9009 6.)
- [8] Szakonyi L.:_Városi vízgőzhálózat modellezése és identifikációja. *Doktori (PhD) értekezés*. Pannon Egyetem, Veszprém, 155 p. 2009.
- [9] Szakonyi L.: Jelek és rendszerek, *PTE PMMK jegyzet*, Pécs 2002.
- [10] Szakonyi L. – Jancskárné A. I.: Szabályozások, *PTE PMMK jegyzet*, Pécs, 2002.
- [11] Szakonyi L.-Jancskárné A. I.-Sári Z.: Regionális anyagáram-hálózat modellezése és identifikálása infokommunikációs rendszer kiépítésével, *INFORMATIKA KORSZERŰ TECHNIKAI KONFERENCIA*, 2010. Dunaújváros, pp.268-275. (ISBN:978-963-9915-38-1)
- [12] Csoknyai I.:Fűtéstechnika I.*példatár*, BME Épületgépészeti és Gépészeti Eljárás Technika Tsz, Budapest, 2008.
- [13] Perry J. H.: Vegyészmérnökök kézikönyve, *Műszaki Könyvkiadó*, Bp. 1968.
- [14] Hodúr C. – Sárosi H.: Hőtani Műveletek, *Jegyzet*, Szegedi Tudományegyetem, Szeged, 2007.
- [15] Garbai L. - Nagy L.: A hőleadók elméleti hőtechnikai kérdései, *Magyar Épületgépészet*, 2009/9. sz., p. 26-29.)
- [16] Garbai L. - Radnai N.: A szekunder fűtési rendszerekben alkalmazott hőleadók hőtechnikai számításai, 2. rész, *Magyar Épületgépészet*, 2011/12. sz., p. 26-29.2011/12. sz., p. 26-29.

Intelligens épületfelügyelet autonóm drónnal

Intelligent building surveillance by autonomous drone

Géza Várady^a, Zoltán Sári^a, Ádám Schiffer^a, Ildikó Jancskár^a

^aPécsi Tudományegyetem, Pollack Mihály Műszaki és Informatikai Kar, Műszaki Informatika Tanszék
varady.geza@pmmik.pte.hu

Absztrakt: Az intelligens épületek koncepciója, habár még számos kérdés nyitott, manapság már széles körben ismert és egyre gyakrabban felhasznált. A kitelepített szenzorok, kamerák, aktuátorok monitorozásra és beavatkozásra adnak lehetőséget. A szenzorok, kamerák kitelepítése azonban nem minden esetben oldható meg. A már meglévő, esetleg műemlék épületek adatainak gyűjtése, felmérése, monitorozása, a nehezen hozzáférhető épületrészek megfigyelése mobil megoldásokat igényel. A kvadrokopterek széles választéka elérhető. Ezen eszközök stabil, egyszerű irányítása alkalmassá teszi őket, hogy beltérben és kültérben egyaránt alkalmazhatóak legyenek a fenti feladatok elvégzésére. A távirányítás helyett azonban sokszor előre programozott illetve autonóm manőverekre van szükség. A beltéri navigációt különböző szenzorokkal kiegészítve, főleg a képi adatokra támaszkodva lehet megoldani. Egy ilyen eszköz kifejlesztése a célja a Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Karának Műszaki Informatika Tanszékén működő kutatócsoportunk munkatársainak.

Kulcsszavak: autonóm drón, beltéri navigáció, kutatás

Abstract: However many questions are still unanswered; the concept of intelligent buildings is widely known and increasingly used. The installed sensors, cameras and actuators make it possible to monitor and intervene. Nevertheless, fixed installation of sensors is not possible in every case. Measuring older buildings, possibly monuments or hard to reach areas need mobile solutions. For that purpose, broad variety of quadcopters is available. These devices are easy to control and are rather stable and as such, they are well applicable for indoor and outdoor tasks. However, in many cases there is a need to do autonomous tasks rather than manual control. The goal of our research group at the University of Pécs, Pollack Mihály Faculty of Engineering and Information Technology, Department of Information Technology is to create such a device and platform.

Keywords: autonomous drone, indoor navigation, research

1. Bevezetés

Kutatócsoportunk a Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Karának Műszaki Informatika Tanszékén tevékenykedik. Tanszékünk automatizálással, robotizált rendszerekkel, vezérléstechnikával, mérésadatgyűjtéssel, kép- és jel feldolgozással és az ehhez köthető szoftveres és modellezési technológiákkal foglalkozik. A profilunk jól illeszkedik az okos ház koncepcióhoz, így 2013 elejétől sikeresen indítottunk kutatást „Smart Building” témakörben. Célunk olyan mobil mérőrendszer kidolgozása volt, amely elsősorban beltéri, de kültéri helyszíneken is képes adatgyűjtésre. A kvadrokopterek jó eszköznek ígérkeztek e célok megvalósítására, az egyes részproblémák jól illeszkednek a kutatási profilunk mellett az oktatáshoz is. A cikk a következőkben a kutatás céljairól, a felmerült újabb irányokról és eddigi eredményeinkről számol be.

2. Informatika és információ a társadalomban

Napjaink társadalmi igénye az információhoz jutás. Az információ mérése, feldolgozása egyre kifinomultabb rendszereken és módszerekkel zajlik. A mai fejlett társadalmakban mindennaposá váltak az okos eszközök, melyek közül a mobiltelefonok terjedtek el a legjobban. Az újabb és újabb technológiai megoldásokkal az épített környezetünket, az ebben létrehozott infrastruktúrát élhetőbbé, kényelmesebbé, de legfőképp hatékonyá tehetjük okos eszközök felhasználásával. Az alapötlet az, hogy a környezetünkből származó információk alapján hozzuk meg döntéseinket, amik akár interakciót is megalapozhatnak a környezettel. Ebben a folyamatban a növekvő információmennyiség okán egy természetes igény a mért adatok és információk előfeldolgozása, illetve az egyre bonyolultabb szabályrendszer szerinti automatikus beavatkozás.

3. Információs infrastruktúra

Az információ felhasználásához alapvető lépés annak begyűjtése, amit valamilyen információs infrastruktúra segítségével tehetünk meg. Az épített környezetben ma már egyre gyakoribb, hogy az informatikai eszközöket a tervezésben figyelembe veszik. A régebbi épületekbe azonban ez az infrastruktúra nem került betervezésre. Itt esetenként utólagos felszerelésre van lehetőség, ami sokszor esztétikai problémát jelent, illetve technikai akadályokba ütközik. Az esetleg műemléki, történelmi értéket képviselő épületek esetében pedig egy ilyen beavatkozás nem lehetséges. Ez utóbbi nem feltétlenül probléma, nem feltétlenül kell mindent szenzorokkal és beavatkozókkel felszerelni. De még műemléki épületek esetében is előfordulhat, hogy időszakosan szükség van ilyen infrastruktúra kiépítésre, illetve olyan felmérési munkák adódhatnak, ahol különböző szenzorok kihelyezésével megvalósítható, vagy olcsóbban, gyorsabban, kisebb beavatkozással valósítható meg egy felmérési munka (páratartalom mérés, hőmérsékletmérés, stb..). A kitelepíthető hagyományos mérőrendszerek az adott épület paramétereikhez igazodva kerülhetnek elhelyezésre. Sokszor nem lehet tetszőleges pozíciókban rögzíteni, fűzni. Ez adta az ötletet arra, hogy térben jól pozícionálható, viszonylag stabil, kisméretű repülő eszközöket, un. UAV (Unmanned Aerial Vehicle) eszközt használjunk, amely képes a lebegésre is, illetve kis sebességű, finom mozgásokra.

4. AR. Drone 2.0

Az elképzelésünket nem teljesen saját fejlesztésre akartuk először alapozni. Ki szeretnénk volna próbálni, hogy mire képes egy kvadkopter, illetve mit tudunk elérni vele. Ehhez több lehetőséget áttekintve a Parrot cég AR. Drone 2.0 eszközét találtuk alkalmasnak. Az AR. Drone 2.0 eszköz egy kvadkopterbe épített WiFi-router, amire a kliens távirányító kapcsolódhat.

A távirányításhoz a cég csak a szoftvert teszi elérhetővé, az okos telefont vagy tabletet külön kell beszerezni. A kvadkopterbe egy HD-Ready kamera is be van építve, a távirányító szoftveren pedig élő képet kapunk. Az eszközhöz létezik API is, aminek a segítségével a drónra játékokat lehet fejleszteni.

Ezek többnyire *augmented reality* (kiterjesztett valóság) játékok, ahol az élő képbe 3D-s objektumokat lehet elhelyezni, amelyek reagálnak az élő kép változásaira. Az eszköz ára 100 000 Ft alatt van, így jó alapja lehet a kísérletezésnek.



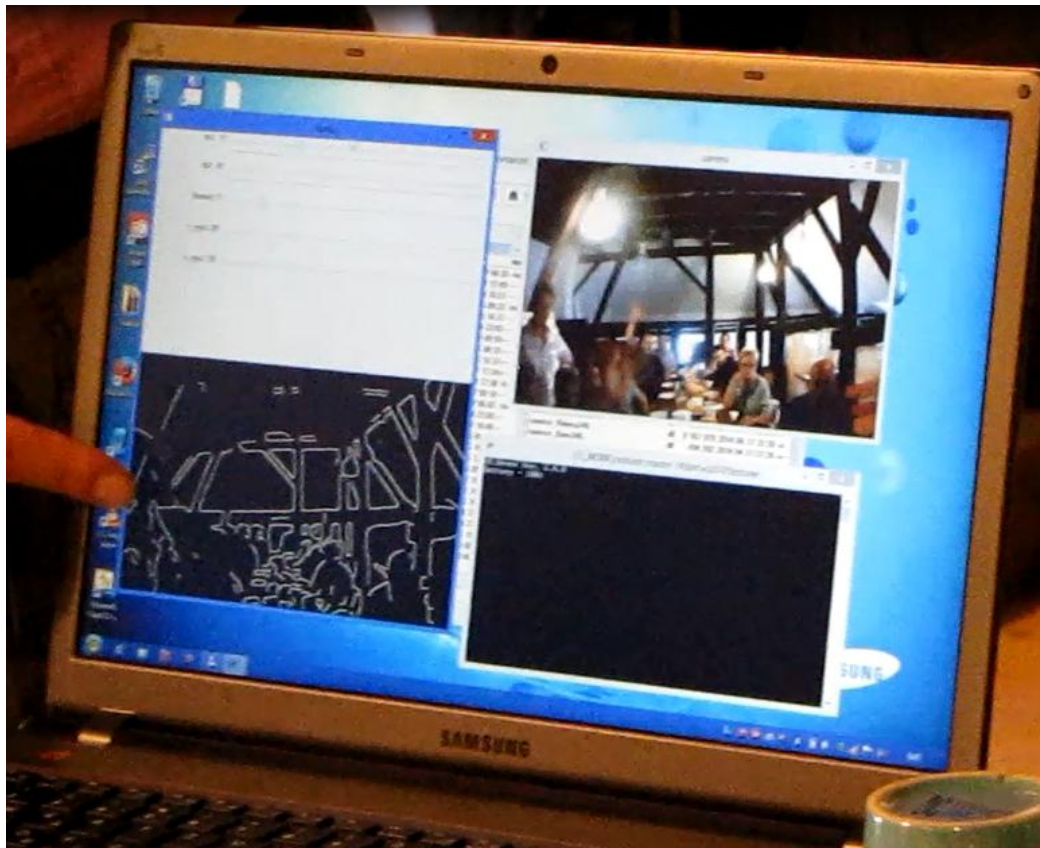
ábra 1 AR.Drone 2.0 a pécsi Zsolnay Kulturális Negyedben

5. Tapasztalatok

A drónnal több beltéri és kültéri repülést, képfelvételezést valósítottunk meg. Bár a drónba épített kamera felbontása alapvetően sok mindenre elegendő (720p 30fps), a fix, nagy látószögű optika miatt az egyik első igény a jobb minőségű kisméretű digitális kamerák ill. fényképezők felszerelése a drónra. Mivel a drón a gyári specifikáció alapján 40 dkg terhet tud vinni, kisebb kompakt eszközöket rá lehet szerelni. Tapasztalataink szerint a 40 dkg-os határ nem a zökkenőmentes repülés határa, hanem az a határsúly, amikor a drón már nem tud emelkedni, folyamatosan süllyed. Egy kisméretű GoPro 3 kamerával már érezhetően máshogy mozog a drón. Egy további tapasztalatunk, hogy a beltéri védőburok kültéren is fontos lenne, de ebbe a szél folyton belekap, illetve könnyen leugrik a helyéről, gátolva ezzel a rotorokat. A drón protokollja szerint ilyenkor leállás, lezuhanás következik be. A drón távirányítása élő kép alapján nagy gyakorlatot igényel, nehéz a torzított képen a mélységeket felismerni. A drón pozicionálását egy alsó VGA kamera segíti. Ha a látott kép homogén, pl. egy szürke folyosó minták nélkül, a drón oldalirányba elcsúszhat. Ha van alul mintázat, egész pontosan lebeg és mozog. A fenti tapasztalatok alapján egyrészt kiegészítő szenzorokat terveztünk beszerelésre a drónba, illetve saját tervezésű és rotorvezérlésű drón létrehozását tűztük ki.

6. Jelen állás - kép alapú vezérlés

A drón élő képét felhasználva lehetőség van objektumok felismerésére, a drónt körülvevő tér feltérképezésére. Alapból képi adataink vannak, amit ki lehet egészíteni a fedélzeti műszerek magasság adataival. Ha ennél többet szeretnénk, extra szenzorokat kell illeszteni a géphez. Ezt direktben, alapból nem tehetjük meg, a gép saját alaplapjához nem férünk hozzá. Ezt a problémát egyébként le tudnánk küzdeni, de kérdéses, hogy a vezérlést végző processzort terheljük-e extra feladatokkal. Az egyetlen gyári kiegészítő egy GPS modul, amivel kültéren naplózhatjuk a mozgásunkat, illetve adott helyekre vihetjük a drónt. Jelenleg az egyetlen szenzor, a kamera segítségével próbálunk dolgozni. Csapatunk a drónt a saját C++ környezetben írt programból vezérli, a streamelt képi adatokon végzett képfeldolgozási eljárások segítségével. A drón most egy speciális mintát, *tag*-et keres a kameraképeken, amit különböző fény- és kontrasztviszonyok között nagy biztonsággal megtalál. Ha pedig látja a célt, az előre rögzített állapotba hozza magát a célhoz képest. Ha a cél mozog, korrigál, így a drón a célpontot folyamatosan követi, ráfordul, magasságban rááll, közelíti, adott esetben, ha a tag-et közelebb visszük, eltolat tőle.



ábra 2 A drón vezérlőszoftvere - tanszéki fejlesztés

Ha a tag kimegy a látómezőjéből, akkor az adott pozícióban lebeg és vár, illetve folyamatosan keresi a tag-et a látómezőjében. Itt a szabályrendszerek átírásával egyre szofisztikáltabb működést érhetünk el és a tag alapú követést igénylő feladatokra már részben most alkalmas a rendszer.

7. Jövőbeli fejlesztési irányok

A következő fejlesztési lépések több párhuzamos vonalon futnak. Az első vonal a tag alapú követés továbbfejlesztése, ahol a tag bonyolult ábra, arc, gesztúra lehet. Egy másik vonal a tag alapú rendszer tag nélkülivé tétele. Ez azt jelenti, hogy a gép a látott élő képen magának keres kapaszkodókat, un. *feature*-öket. Ezek segítségével „felismeri” hol van, ha mozog, folyamatosan viszonyítani tud az előző pozíciójához. Ennek a módszernek a fejlesztésével ismeretlen terepen tudunk haladni, a terepet feltérképezni. A képi alapú érzékelést további szenzorokkal egészíthetjük ki, amely további lehetőségeket ad a felhasználásra. Egy további fejlesztési vonal pedig a saját drón építéséhez kötődik. Itt részben oktatási példaként, részben a mostani gép hiányosságai miatt saját tervezésű és vezérlésű kvadkoptert szeretnénk építeni. Tanszékünkön van már tapasztalat áramlási modellek vizsgálatában, szénszálas szerkezetek tervezésében és építésében, vezérlőalgoritmusok kidolgozásában és paraméterezésében, így jól haladunk ezen a vonalon is.

8. Konklúzió

Kutatási témánk több területen is ígéretes megoldásokkal kecsegtet. Az alkalmazhatóság már most látszik, az eddigi fejlesztésekkel kiegészítve a szórakoztató eszközt már munkára is lehet fogni. Az eszköz fejlesztésébe a diákokat is be lehet vonni, ahol vezérlési, szabályozási, képfeldolgozási, programozási, mérési és műszerillesztési feladatokat végezhetnek el.

A kutatást a TÁMOP-4.2.2.C-11/1/KONV-2012-0005, Jól-lét az információs társadalomban pályázat támogatja.

Irodalomjegyzék

- [1] Z Sári, I Jancskár, Á Schiffer, G Várady, Quadrotor copter programming and control
In: Architectural, Engineering and Information Sciences - 9th International PhD & DLA Symposium: Abstracts Book. Konferencia helye, ideje: Pécs, Magyarország, 2013.10.21-2013.10.22. Pécs: University of Pécs Pollack Mihály Faculty of Engineering, 2013. p. 142.
- [2] Z Sári, I Jancskár, Á Schiffer, G Várady, Autonomous quadcopter landing system design using onboard and inertial visual sensing
In: Architectural, Engineering and Information Sciences - 9th International PhD & DLA Symposium: Abstracts Book. Konferencia helye, ideje: Pécs, Magyarország, 2013.10.21-2013.10.22. Pécs: University of Pécs Pollack Mihály Faculty of Engineering, 2013. p. 143. (ISBN:ISBN 978-963-7298-54-7)
- [3] I Jancskár, Z Sári, Á Schiffer, G Várady
Soft control of a 1 degree-of-freedom helicopter
Konferencia helye, ideje: Pécs, Magyarország, 2013.10.21-2013.10.22.
Pécs: Pécsi Tudományegyetem, Pollack Mihály Műszaki és Informatikai Kar, 2013. (ISBN:978-963-7298-54-7)

- [4] G Várady, I Jancskár, Z Sári, Á Schiffer, Remote controlled quadrocopter aims and results
In: Architectural, Engineering and Information Sciences - 9th International PhD & DLA Symposium: Abstracts Book. Konferencia helye, ideje: Pécs, Magyarország, 2013.10.21-2013.10.22. Pécs: University of Pécs Pollack Mihály Faculty of Engineering, 2013. p. 164. (ISBN:ISBN 978-963-7298-54-7)

Hardverközeli programozás oktatása a DIY Calculator segítségével

Teaching hardware programming with DIY Calculator

Varga Imre^a

^aDebreceni Egyetem, Informatikai Rendszerek és Hálózatok Tanszék
varga.imre@inf.unideb.hu

Absztrakt: A Debreceni Egyetemen folyó Mérnök-informatikus BSc képzés egyik alappillére a programozás oktatás, mely két részre osztható. Egyrészt magas szintű programnyelvekről tanulnak a hallgatók, másrészt megismerkednek a hardverközeli programozás alapjaival. Az utóbbi témakörbe tartozó tantárgyak célja az, hogy bemutassa a hallgatóknak a számítógép részletes felépítését és működési elvét valamint megismertesse őket az assembly szintű programozás alapjaival. Közben lehetőség nyílik a tanulmányok során szerzett különböző ismeretek szintetizálására. Például megérthetik a hallgatók, hogy egy C nyelven megírt forráskódból hogyan lesz, gépi kód majd futó folyamat és eközben milyen változások játszódnak le a processzor elektronikai áramkörökben.

Ezen célok eléréséhez kiváló eszköz a didaktikai célokkal kifejlesztett úgynevezett Do It Yourself Calculator. Ez nem más, mint egy 8-bites CISC processzor alapú számológép szimulátor program assembler-rel és debugger-rel kiegészítve. Egyszerű felépítése és működési elve könnyebben megérthető, mint egy modern, komplex x86-os architektúráé, viszont segítségével megvalósítható és tanulmányozható például akár a tömbök kezelése, rekurzív alprogramhívás, megszakításkezelés, különböző perifériák meghajtása, stb. Arra szeretnék rávilágítani, hogy a hallgatók által is közkedvelt DIY Calculator milyen hasznos eszköz lehet a képzés során és milyen könnyen érhetőek el vele sikerek. Erre alapozva a valós számítógépes architektúrák felépítése és működése könnyebben megérthető.

Kulcsszavak: hardver, assembly, programozás

Abstract: One of the basic pillars of the education of Software Engineering at the University of Debrecen is the teaching of programming, which has two sections. On the one hand students study high level programming languages. On the other hand they can get to know the basics of hardware-near programming. The aim of the subjects in the latter topic is that to show how the computer build up, what are the principles of its operation, how is it working and introduce the students into the basics of assembly level programming. During studies students have possibility to synthesize their knowledge. They can get experience how a C source code became machine code and then turn into a process and during this time what kind of changes happen in the circuits of the processor.

To reach these goals an excellent tool is the didactically developed so called Do It Yourself Calculator. This is an 8-bit CISC processor based calculator simulator program integrated by assembler and debugger. Its simple structure and operation is easier understandable than the modern complex x86 architecture. Nevertheless with the help of this tool students can investigate for example how the arrays are handled, how the recursive subroutine call and interrupt handling works, and how some peripherals are driven. I would like to highlight how useful can be the popular DIY Calculator during the education. Based on the knowledge related to the virtual Calculator real architectures can be easily understandable.

Keywords: hardware, assembly, programming

1. Bevezetés

A Debreceni Egyetemen folyó Mérnök informatikus BSc képzés során a hallgatók különböző kategóriákba sorolható tantárgyakat tanulnak. Egyik ilyen kategória a mérnöki szempontból fontos, alapvető ismereteket tartalmazó tantárgyakat (pl. Fizika, Elektronika, Digitális technika, stb.) gyűjti egy csokorba. Egy másik tárgykört alkotnak a logikailag magasabb szintű informatikai rendszerekhez kapcsolódó tantárgyak, mint például a Mesterséges intelligencia, Adatbázisrendszerek, Vállalatirányítási rendszerek, stb. A kettő között egyfajta összeköttetést jelenthet a programozási tantárgykör, mely két további alcsoportra osztható. A felső szinten található a Magasszintű programozás 1-2 illetve a Bevezetés a LabView programozásba nevű tárgyakat az alsón pedig a Hardverközeli programozás 1-2 tantárgyakat. Utóbbiaknak több szempontból is fontos szerepe van. Egyrészt az aktuális mintatanterv 2009-es bevezetése óta ez a fő színtere annak, hogy a hallgatók egyáltalán megismerkedjenek a számítógép felépítésével és működésével (ugyanis a korábbi Számítógép architektúrák tárgy kikerült a mintatantervből). Másrészt segítségével egységgé szintetizálható a korábban szerzett ismeretanyag, így a hallgatók átláthatják például, hogy egy C nyelven írt program futása során történő értékadás hogyan is valósul meg az egyszerű elektronikai elemekből felépülő regiszterekben lejátszódó folyamatok révén. Tehát a mérnöki képzésben kiemelkedő fontosságú hardveres ismeretek és alacsony, assembly szintű programozási képesség megszerzése a Hardverközeli programozás tantárgy hatáskörében van.

Ennek oktatása során a Neumann elvű számítógépekkel történő megismerkedés válaszúthoz vezet. Vagy az idejétmúlt, de egyszerű rendszerekkel kezdjük az ismerkedést vagy egyből az aktuális technológia nagyon összetett világába ugrunk fejest. Mindkét módszernek vannak előnyei és hátrányai is. Szerintem a mai architektúrákat egy kezdő számára nagyon nehéz megérteni. Nem elég átlátni a nagyméretű általános és speciális regiszterkészletet, olyan fogalmak is hamar előtérbe kerülnek, mint a lebegőpontos egység, vektorizálás, pipelining, több szintű cache és több szintű megszakításkezelés, DMA, HyperThreading, többmagos processzorok, stb.

Ha egyszerű architektúrával akarjuk először megismertetni a hallgatókat, akkor azzal a problémával találkozhatunk, hogy az oktatásban is használható, didaktikailag megfelelő szimulátor nagyon kevés van és azok is licenszhez kötöttek. Ezen problémák orvoslására az ún. Do It Yourself Calculator kiváló gyógyír lehet.

2. A DIY Calculator

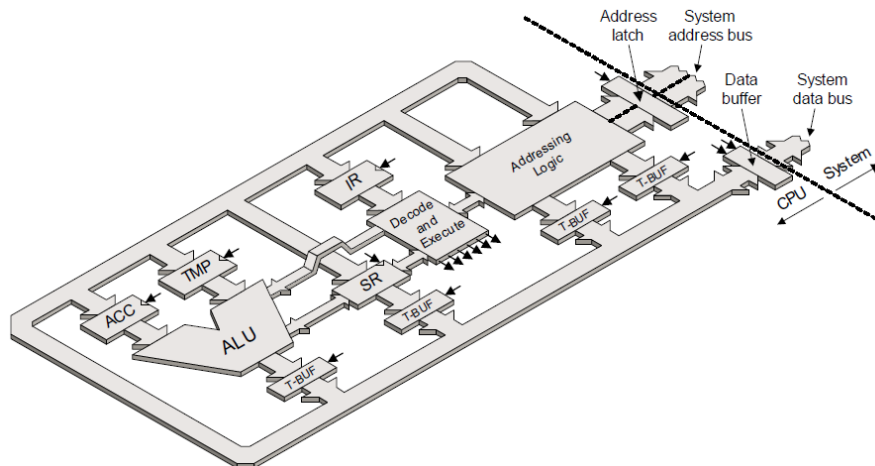
Mi is ez az eszköz? Ez nem más, mint egy egyszerű felépítésű hardver szimulátor és az ennek a programozását lehetővé tevő fejlesztői környezet. A szoftver és egyéb kiegészítők az alkotók honlapjáról [1] ingyenesen letölthető és Windows rendszeren futtatható. Fejlesztése során didaktikai szempontokat is figyelembe vettek. Felépítése és működése egyszerű, jól dokumentált. A működés során lehetősége van a programozóknak az egyes regiszterek vagy memóriarekeszek tartalmának nyomon követésére. A program az utasítások végrehajtásának a menetét, technikai részleteit részletesen, szöveges formában képes a felhasználó elé tárni. A szoftverhez létezik egy könyv [2], amely az alapoktól kezdve bevezeti a programozót abba, hogyan kell egy olyan összetettebb szoftvert megírni erre az architektúrára, amely egy egyszerű számológép funkcionalitását valósítja meg. A program dokumentációjához tartozik még egy ún. adatkönyv [3], amely az architektúra részletes elektronikai felépítésén túl a nyelv utasításkészletének részletes leírását is tartalmazza. Nézzük most meg először a szimulált hardver felépítést, majd a programozásra szolgáló saját nyelv jellemzőit.

2.1. Az architektúra

Megszokott módon a mikroszámítógépünk agya a CPU, amely a műveletek elvégzéséért és döntések meghozataláért felelős. Az operatív memória tartalmazza az adatainkat és a programunkat. Az Input/Output portok lehetővé teszik, hogy a CPU kommunikáljon a külvilággal. A rendszernek imént említett komponensei a buszrendszer segítségével cserélnek információt. Nézzük hogyan is épülnek fel ezek a konkrét rendszerben!

CPU

A szimulált hardver szíve egy 8-bites CISC processzor, melynek három alrendszere az aritmetikai logikai egység (ALU), a vezérlő egység (CU) és a címző egység. Ezeket egy belső buszrendszer köti össze. Mindegyik egység működését regiszterek segítik. A CPU sematikus felépítését az 1. ábra mutatja.



1. ábra A DIY Calculator CPU-jának logikai szerkezete [3]

A regiszterek funkciójukat tekintve hasonlóak az Intel 8086-os processzor egyes regisztereihez, de elnevezésüket, méretüket és néhány apróbb jellemzőt tekintve eltérőek lehetnek attól. A virtuális hardver főbb regiszterei a következők:

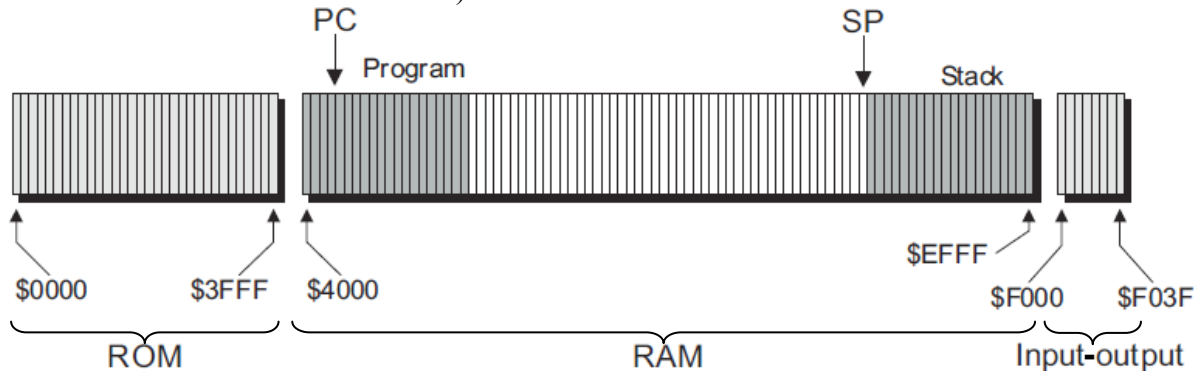
- *Programszámoló regiszter (PC):* 16 bites memóriacím tárolására szolgál, amely címen található a legközelebb végrehajtandó utasítás.
- *Utasítás regiszter (IR):* a végrehajtás alatt álló gépi utasítás 1 byte-os opkódját tartalmazza.
- *Akkumulátor (ACC):* általános munkaregiszter, mely 1 byte méretű adatot képes tárolni és a legtöbb művelet egyik operandusa és eredménye is ide kerül.
- *Státusz regiszter (SR):* a végrehajtást befolyásoló 5 jelzőbitet tartalmazó (logikailag 1 byte-osnak tekintett) regiszter. Bitjeinek jelentése a következő. I (interrupt): be van-e állítva megszakítás figyelés, O (overflow): a legutóbbi művelet során történt-e túlsordulás, N (negative): az ACC tartalma előjeles egész számbábrázolást tekintve negatív-e, Z (zero): az ACC tartalma vagy a legutóbb végrehajtott művelet eredménye nulla-e illetve összehasonlító művelet egyenlőséget adott-e, C (carry): a legutóbb végrehajtott művelet során keletkezett-e átvitelbit illetve összehasonlító művelet eredménye az e hogy az ACC tartalma nagyobb, mint az összehasonlítás operandusa.
- *Index regiszter (X):* indexelt címzés esetén a címetolás 2 byte-os értékét tároló regiszter.
- *Verem mutató (SP):* a memória azon címét tartalmazó 2 byte-os regiszter ahová a rendszer verembe betenni kívánt elem kerül illetve veremből történő kivétel esetén az SP által tárolt címnél egyel nagyobb memóriacímen lévő érték (verem teteje) kerül kivételre.
- *Megszakítási vektor (IV):* a megszakítási rutin RAM-beli címét tároló 16 bites regiszter.
- *Átmenti programszámoló A és B regiszter (TPCA, TPCB):* abszolút, indexelt és indirekt címzéshez szükséges 2 byte-os ideiglenes tároló.

- *Átmeneti regiszter (TMP)*: kétoperandusú művelet esetén a második operandust átmenetileg tartalmazó 8 bites regiszter.
- *Address latch (MAR)*: az itt tárolt cím kerül a címbuszra, azaz olvasás esetén erről a memóriacímről olvasunk, írás esetén erre a címre kerül a tárolandó adat.
- *Data buffer (MDR)*: olvasás esetén az adatbusz tartalma ide kerül, írás esetén az MDR tartalma kerül az adatbuszra.

A hivatalos adatkönyv [3] CPU egyes részeinek felépítését pontosan leírja, néha egészen logikai kapu szintig lemenve. Ha nem csak a szerkezeti felépítés érdekel bennünket, akkor a logikai működésen túl az egyes utasítások végrehajtása során különböző síneken mérhető feszültségek idődiagramjait is megtekinthetjük órajeltől órajelre. Ez hihetetlenül fontos lehet mérnöki szempontból és így a képzés nélkülözhetetlen részét alapozhatjuk erre az eszközre.

Memória

A hardver mások fontos eleme a memória, mely a Neumann elveknek eleget téve azt adatainak és a programunk tárolására szolgál. A DIY Calculator csak olvasható (ROM) és írható-olvasható (RAM) memóriát is tartalmaz. A memóriarekeszek byte-onként címezhetőek, 16 bites címekkel. Így egy 64kB méretű címtérünk van, aminek az elején (16kB) ROM található, majd a (44kB méretű) RAM helyezkedik el (2. ábra). Az utolsó 4kB méretű tartományban 'memory mapped I/O' technikát alkalmazva input/output címeket találhatunk (bár nincs kihasználva az összes cím).



2. ábra: A 64kB-os címtér felosztása

I/O protok

Ahogy már említettük 'memory mapped I/O' révén ugyanaz a címbusz használható memóriarekesz és I/O port hivatkozásra. 32 input (\$F000-\$F01F) és 32 output (\$F020-\$F03F) port érhető el. Ezekre a hamarosan tárgyalandó perifériák kapcsolódnak.

Buszrendszer

Itt is, mint általában 3 busz található: cím-, adat- és vezérlő busz. A kétirányú adatbusz 8 bit széles illeszkedve az ACC méretéhez. A 64kB-os címtérnek megfelelően a címbusz 16 sít tartalmaz. A vezérlő busz mindössze néhány vezetékből áll. Ilyen az írási (*write*) illetve olvasási (*read*) engedélyező vonal, a hagyományos *reset* vonal, a szinkronizálásra szolgáló órajelet biztosító *clock* vonal továbbá a megszakítási kérelem (*IRQ*) és megszakítási nyugta (*IACK*) vezeték.

Perifériák

A CPU az I/O portokon és a buszrendszeren keresztül különböző perifériákat ér el. Az alapértelmezett periféria az ún. Front panel, mely egy LCD kijelzöt, egy 6 elemű LED sort, egy (átcímkezhető) számológép billentyűzetet (keypad) valamint néhány vezérlő gombot (On/Off, Reset, Run, Step) tartalmaz. Ezen kívül használhatunk egy szöveges terminált, egy teljes qwerty billentyűzetet, és az ún. Workbench nevű eszközt. Utóbbin helyet kapott 2 darab 8 bites kapcsolósor, egy 8 bites LED sor, egy-egy dekódolt szimpla és dupla valamint egy dekódolatlan 7 szegmenses kijelző. A Calculator segítségével tehát a hallgatók az ilyen egyszerűbb hardver elemek meghajtását is elsajátíthatják.

2.2.A DIY Calculator assembly nyelve

Az előbbieken áttekintett hardver programozására assembly szinten van lehetőség. A program ASCII karakterekből épül fel. A nyelv betűnek tekinti az angol abc karaktereit és az aláhúzás () jelet, a kis és a nagy betűket nem különbözteti meg. A numerikus konstansok megadhatóak decimális (pl.: 90), bináris (pl.: \$5A) vagy hexadecimális (pl.: %1011010) alakban. Nevesített konstans fordítási direktívával hozható létre. Egy forráskód maximum egy darab '1 címés' utasítást tartalmazó sorokból áll, ahol egy utasítás 4 elemből épül fel: címke, mnemonik, operandus, megjegyzés.

- *Címke*: az utasításokra történő hivatkozáshoz szükséges. Betűvel kezdődik és betű vagy számjegy karakterrel folytatódó maximum 8 karakter hosszúságú azonosító kettősponttal (:) lezárva. Mindig egy RAM címet jelent, azt helyettesíti. Nem kötelező elem.
- *Mnemonik*: az utasítás/operátor néhány kereketeres kulcsszava (pl. LDA: Load Accumulator).
- *Operandus*: a kétoperandusú operátor második operandusa (az első mindig az ACC-ben van). Lehet adat vagy cím is. Nem minden utasításban szükséges.
- *Megjegyzés*: kettős kereszt (#) karaktertől a sor végéig található programrészt az assembler figyelmen kívül hagyja.

A nyelv 47 végrehajtható és 7 fordítónak szóló utasítást (direktíva) ismer. Az előbbieket besorolhatóak néhány kategóriába: mozgató utasítás, bitenkénti műveletek, aritmetikai műveletek, vezérlésátadó és egyéb utasítások. Utóbbiak mnemonikja pont (.) karakterrel kezdődik. Memóriaterület foglalás (például statikus változó részére) direktívával történik. A betöltés és a címfordítás kezdőcímének a megadása szintén direktíván keresztül valósul meg.

A CPU 7 különféle címzési módot tud megvalósítani, hogy egy adatra hivatkozzon:

- *rejtett (implied)*: a műveletnek nincs szüksége operandusra (vagy csak egyre, de az az ACC-ben van), így ez nem is 'valódi' címzés. (pl.: INCA)
- *alap/nagy kódba épített (standard/big immediate)*: memóriában az operandus kód után közvetlenül találjuk az 1 vagy 2 byte méretű operandust. (pl.: ADD \$10, BLDSP \$EFFF)
- *alap/nagy közvetlen (standard/big absolute)*: RAM-ban az utasítás opkódját egy 2 byte-os memóriacím követi, amely megmondja hol található a memóriában az 1 vagy 2 byte méretű operandus. (pl.: OR [\$4100])
- *indexelt (indexed)*: a műveleti kód után egy címet találunk, ehhez hozzáadva az index regiszter (X) tartalmát megkapjuk azt a címet, ahonnan az operandust vennünk kell. (pl.: SUB [\$4100,X])
- *közvetett (indirect)*: az opkód utáni két byte-on egy cím található, amely megmutatja, hol van az a másik cím, ami meghatározza az operandus értékét. (pl.: JMP [[\$4100]])
- *előzetesen indexelt közvetett (pre-indexed indirect)*: a műveleti kód utáni 2 byte-os címhez hozzáadva az index regiszter értékét megkapjuk azt a címet, ahol megtalálható az operandus címe. (pl.: LDA [[\$4100,X]])
- *utólagosan indexelt közvetett (indirect post-indexed)*: az utasítás gépi kódját a memóriában egy cím kövezi, mely egy másik 2 byte-os cím helyét határozza meg. Az ezen a címen található értékhez hozzáadva az index regiszter értékét megkapjuk azt a címet, ahol megtalálható az operandus értéke. (pl.: STA [[\$4100],X])

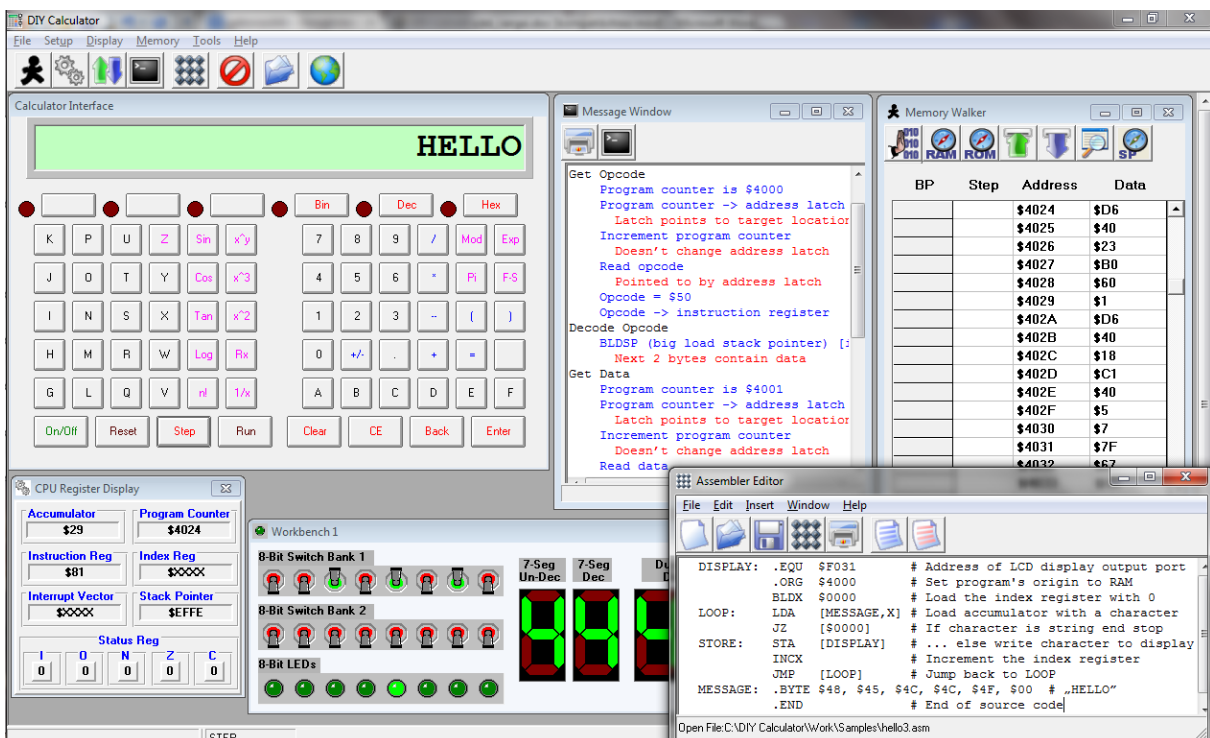
Amint látható ez az egyszerű kis eszköz is egészen összetett viselkedést tud megvalósítani. Bár a neve alapján csak egy egyszerű számológép hardvernek gondolhatnánk, azért képességei lehetőséget nyújtanak ahhoz, hogy akár bonyolultabb programozási eszközöket és adatszerkezeteket is megvalósítsunk vele. Készült már például BASIC interpreter is ezen a nyelven.

2.3.A fejlesztői környezet

A program forráskódja a fejlesztői környezet editorában (is) megírható. .asm kiterjesztésű file-ként jelenik meg a PC-n. Ebből képez az assembler (szöveges formátumú) gépi kódot .ram kiterjesztéssel. Ez már közvetlenül betölthető a DIY Calculator RAM-jába, majd a 'Run' gomb megnyomásával futtatható is a program. Lehetőség van továbbá a futtatásra utasításonként is. Hogy jobban megérthető és következő legyen a fordítás menete az assembler a fordítás részleteiről (forrás és gépi kód összerendelés, címkéknek megfelelő címek listája, stb.) további log file-okat is készít .lst és .rad kiterjesztéssel.

Bár az assembler szintaktikai hiba esetén nem túl bőbeszédű, a fejlesztői és szimulációs környezet nyomkövető eszközök egész garmadáját biztosítja a programfejlesztő hallgatók rendelkezésére (lásd 3. ábra). Az úgynevezett Memory Walker-rel byte-ról byte-ra megtekinthetjük a teljes memória tartalmát, töréspontokat helyezhetünk el benne. A 'CPU regiszter display' valós időben mutatja a fontosabb regiszterek (ACC, IR, PC, SR, X, IV) aktuális tartalmát a futás során. Az 'I/O port display' megmutatja néhány ki és bemeneti puffer pillanatnyi tartalmát. A 'Message window' nevű eszköz a futás közben történő eseményeket írja le szövegesen szinte órajel szintre lebontva az időt. Ha a futás túl gyors a virtuális hardver órajelét is tudjuk állítani szoftveresen. Mindezek az eszközök nagyon hasznosak ahhoz, hogy a hallgatók megértsék a háttérben folyó mérnöki szemmel fontos eseményeket. Segítségükkel könnyebben rájöhetnek, hogyan kell optimálisabb a programot írni akár magas nyelven is. Mindezekre valós hardver esetén egyáltalán nincs lehetőség. Vegyük még ehhez hozzá, hogy a DIY Calculator felépítéséhez és működéséhez részletes dokumentáció érhető el [3], ami nem minden mai valós architektúra esetén mondható el. A dokumentáció annyira részletes, hogy két csapat is megvalósította a hardvert FPGA segítségével.

A 3. ábra a számítógép fő egységét (a front panelt), a debug valamint I/O eszközök egy részét mutatja be. Az ábra jobb alsó sarkában pedig az assembler editora található. Ebben egy példát láthatunk a DIY Calculator programra, mely a „HELLO” sztringet írja ki a front panel LCD kijelzőjére.



3. ábra: A DIY Calculator fejlesztői- és futatókörnyezetének egy része

3. Összegzés

A mérnökinformatikus képzés fontos feladata, hogy megismertessük a hallgatókkal, hogyan is épül fel a számítógép mérnöki szemmel illetve, hogy a programozással kapcsolatos és az elektronikai ismeretek közötti rést áthidaljuk. Ennek a célnak az eléréséhez nagyszerű eszköz lehet a DIY Calculator nevű virtuális hardver. Felépítése egyszerű, a dokumentációk alapján könnyen megérthető. Programozása saját assembly-szintű nyelven történik. A használat közben könnyebben megérthetik a hallgatók hogyan is működik hardverközeli szinten az, amit például egy C program esetén akár nap, mint nap használnak. Gondoljunk itt a következőkre: mutatók, tömbök elemeire hivatkozás, sztringek tárolás, eljáráshívás, érték visszaadás függvény esetén, paraméterátadás, lokális változók dinamikus élettartam kezelése, rekurzió, veremkezelés, megszakításkezelés, stb. Emellett – mellékhatásként – néhány külső elektronikai eszköz (pl.: kódolt/kódolatlan 7 szegmenses kijelző, LED sor, kapcsolósor) meghajtását is elsajátíthatják. A hallgatók $\frac{3}{4}$ része vallja azt, hogy a Calculator segített neki a magas szintű programozási ismereteinek az elmélyítésében is.

Természetesen a DIY Calculator nem helyettesíti a valós processzorok (x86, ARM, stb.) bemutatását és ezek programozásának az elsajátítását. Azonban kezdők számára első lépésnek kiváló, mivel a processzorműködés, programfuttatás valós időben monitorozható a fejlesztői környezet és a hardver szimuláció nyomkövető eszközrendszerével, könnyebbé téve a megértést és az elsajátítást. Néhány éves tapasztalat alapján azt mondhatom, hogy használata az képzés határfokát javítja.

Irodalomjegyzék

- [1] Online: <http://www.clivemaxfield.com/diycalculator> (legutóbbi megtekintés: 2014. június 14.)
- [2] Clive „Max” Maxfield és Alvin Brown: The definitive guide to How Computers Do Math, Wiley (2005).
- [3] Clive „Max” Maxfield és Alvin Brown: The official DIY Calculator Data Book, (2005)
- [4] Clive „Max” Maxfield és Alvin Brown: Introducing the Virtual DIY Calculator, *Everyday Practical Electronics* **34**, No. 10 (2005), 694-696.

Humor a programozás oktatásában

Humour in teaching programming

Achs Ágnes

Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Kara
achs@pmmik.pte.hu

Absztrakt: Az olyan barátságok, amelyek nevetéssel kezdődnek, erősek szoktak lenni. – írja Stephen King. Kérdés, hogy ez mennyire igaz a programozással való barátkozásra.

Bár informatikusokat képzünk, hallgatóink nem kis részének nehézséget okoz a programozás megtanulása. Különösen igaz ez az objektum orientált paradigma esetén. Fontos-e, hogy érzelmileg is motiváljuk ezeket a hallgatókat? Tudunk-e X-, Y- vagy akár Baby-boom generációként hatni a Z-generációs diákokra? Ez bizony nehéz kérdés, koránt sem biztos, hogy mindig sikerrel birkózunk meg vele.

Előadásomban szeretném bemutatni az általunk tanított tematikát (C#, Java), illetve azt, hogy milyen feladatok kitérésével igyekszünk felkelteni a hallgatók érdeklődését. Egyúttal örülnék, ha előadásom reflexiójaként a tisztelt kollégák véleményeznék ezt a munkát, illetve tanácsokkal, javaslatokkal látnának el a folytatást illetően.

Kulcsszavak: programozás oktatása, humor, kreativitás

Abstract: The friendships beginning with laughter used to be strong. – says Stephen King. Whether is it true for befriending the programming?

Despite our students are going to be IT professionals, they have difficulties to learn and understand programming, mainly the object oriented paradigm. Is it important to emotionally motivate these students? Can affect the X, Y or Baby-boom generation to the Z-generation students? This is certainly a difficult question, by no means certain that we can solve it.

In my presentation I will introduce our teaching syllabus (C #, Java) and the tasks that are designed to attract students' interest. At the same time I would be happy if I could hear the reflections of the Honourable colleagues and suggestions to the future works.

Keywords: teaching programming, humour, creativity

1. Bevezetés

„Az iskola arra való, hogy az ember megtanuljon tanulni, hogy felébredjen tudásvágya, megismerje a jól végzett munka örömét, megízlelje az alkotás izgalmát, megtanulja szeretni, amit csinál, és megtalálja azt a munkát, amit szeretni fog.”

Szent-Györgyi Albert

Szerencsés ember vagyok, mert megadott, hogy számomra az iskola azt jelenti, mint Szent-Györgyi Albert számára. Ezt jelentette a kisvárosi általános- és középiskolám, ezt a debreceni egyetem, és ugyancsak ezt jelenti a pécsi egyetem műszaki kara, ahol a katedra másik oldalán szerethetem azt, amit csinálok és azokat, akiket taníthatok.

Azt, hogy szép emlékeket őrzök egykori iskoláimról, egyértelműen annak köszönhetem, hogy jó tanáraim voltak. Szeretettel emlékszem rájuk. Példájuk, illetve felmenőim, testvérem, barátaim, kollégáim példája arra ösztökél, és azért tesz felelőssé, hogy a katedra túloldalán

állva is megpróbáljam ugyanazt képviselni, amit tőlük kaptam, és amit Szent-Györgyi ilyen szépen megfogalmazott.

De vajon sikerül-e? Erre nem tudok válaszolni, ezt a diákjaim tudják eldönteni. Remélem, van közöttük olyan, akit kicsit én is hozzásegíthettem ahhoz, hogy megtanulja szeretni, amit csinál.

Cikkemben elsősorban arról szeretnék írni, hogy milyennek látom az oktatás helyzetét, mi segíthetné a diákok tanulását, a végén pedig néhány szót ejtek az általam tanított témákról. A téma körüljárásakor véleményemet az irodalomjegyzékben említettek tudásával, tapasztalatával igyekszem alátámasztani.

2. C-generációs diákok

Felsőoktatásban tanítóként nem lenne tisztünk a közoktatással foglalkozni, de egyrészt nem hagyhatjuk figyelmen kívül azt a közeget, ahonnan diákjaink érkeznek, másrészt az oktatáshoz „mindenki ért”, így mindenkinek van véleménye róla. Tekintheznék az egész kérdésre úgy, mint valamilyen, általunk nem befolyásolható peremfeltételre, de mégis azt javaslom, hogy időzzünk el egy kicsit a témánál, járjuk körbe, hátha mégsem teljesen mozdíthatatlan a rendszer, hátha mégis találunk legalább egy-két apró rést, amelyet tágitva tágul a mi mozgásterünk is.

Kezdjük avval, hogy kiket is tanítunk. Csupa olyan fiatalt, akik fiatalabbak az internetnél, akik számára természetes az állandó online jelenlét, akik számtalan forrásból kapják és várják az információt, és akik roppant sérülékenyek ebben az információzuhatagban. Szokás őket születési dátum szerint Y vagy Z generációba sorolni, de szellemes a C-generáció kifejezés is (constant connectivity, collaboration, stb.) [12]. Azokat a fiatalokat sorolják ide, akik szinte lényük meghosszabbításának tekintik az internetet, és nem élhetnek mobil kütyük nélkül. Lényeges tulajdonságuk az is, hogy az internetet a kapcsolattartás eszközének tekintik, közösen alakítanak véleményt, fogadnak el igaznak vélt állításokat – mindezt sokszor kritika nélkül. Hogy ez jó-e vagy sem, veszélyes-e rájuk nézve, vagy sem, most nem taglalom, de azt gondolom, hogy magát a lehetőséget ki kellene használnunk, és akár ilyen módon is felkeltethetnénk az érdeklődésüket, kíváncsiságukat. Evvel nyilván nem mondtam semmi újdonságot, egészen biztos, hogy sok kolléga él is a közösségi média nyújtotta lehetőséggel, azonban ez csupán egy egyszerű eszköz, amely talán besorolható a korábban említett rések közé, de a lényege egyáltalán nem érinti.

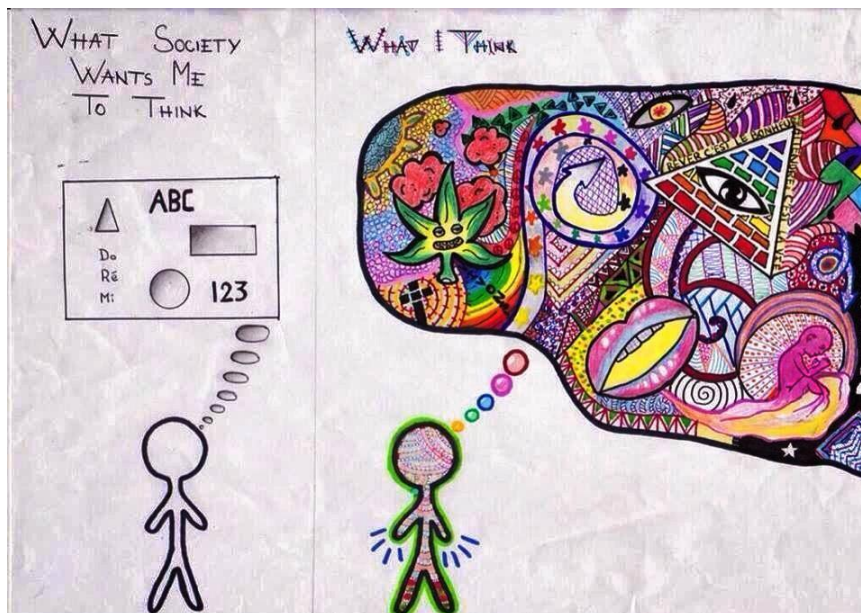
A lényeg a 85 éves Ilonka tanító néni talán már közhelyként hangzó, de mélyen igaz megjegyzésében rejlik: „régén a gyereket tanítottuk, ma a tananyagot”. Valóban: figyel-e a mai iskola a gyerekre, vagy általában a diákra? Vagy ő is csak egy „ipari” termék, aki hatéves korában bekerül az iskola-gyárba, és kikerül onnan a futószalag valamelyik outputján?

Mélyen egyetértek Ilonka néni szavaival, de mégis azt gondolom, hogy bármilyen kiváló is volt a hajdani oktatás, bárhány Nobel-díjas és bármennyi nagy tudású ember is nőtt ki a régi almamátterek falai közül, nem lehet teljes mértékben megtartani a hagyományos kereteket. A C-generáció mást igényel.



Pink Floyd: A fal

Ken Robinson szavai szerint „A probléma az, hogy úgy próbálunk elébe menni a jövőnek, hogy azt csináljuk, amit a múltban csináltunk. És közben fiatalok milliói nem értik, mi értelme van az iskolába járásnak. Amikor mi mentünk iskolába, akkor azt mondták nekünk, hogy ha keményen dolgozunk, akkor bekerülünk az egyetemre, és utána lesz munkánk. A mi gyerekeink már nem hiszik ezt el, és igazuk van. A diploma ma már nem garancia. És főleg nem az, ha a hozzá vezető út során éppen azokat a dolgokat marginalizáljuk, amiket ők a maguk számára fontosnak tartanak.” [9]



Másutt ezt mondja: „Állítom, hogy a kreativitás ugyanolyan fontos az oktatásban, mint az írástudás, és azzal megegyező szinten kellene bánjunk is vele.” [8]

Hogy ezt miképpen oldották meg 50-100-200 évvel ezelőtt, nem tartozik jelenlegi témánkhoz, de hogy a sokak által kért, igényelt, követelt, javasolt egyik megoldás a tarsolyukban volt, abban biztos vagyok. Ez a megoldás a művészeti nevelés. Bár nagyon fontosnak tartom a kérdést, nem időzünk el túlzottan mellette, hiszen – hangsúlyozva, hogy az értelmiségivé válás létfontosságú elemének tartom a művészetekkel való kapcsolatot – ez elsődlegesen mégsem az informatikai felsőoktatás feladata. (Ugyanakkor nem egyszer kiderül egy-egy jó képességű diákról, hogy színjátzó körbe járt, táncol, énekel, zenél, stb..)

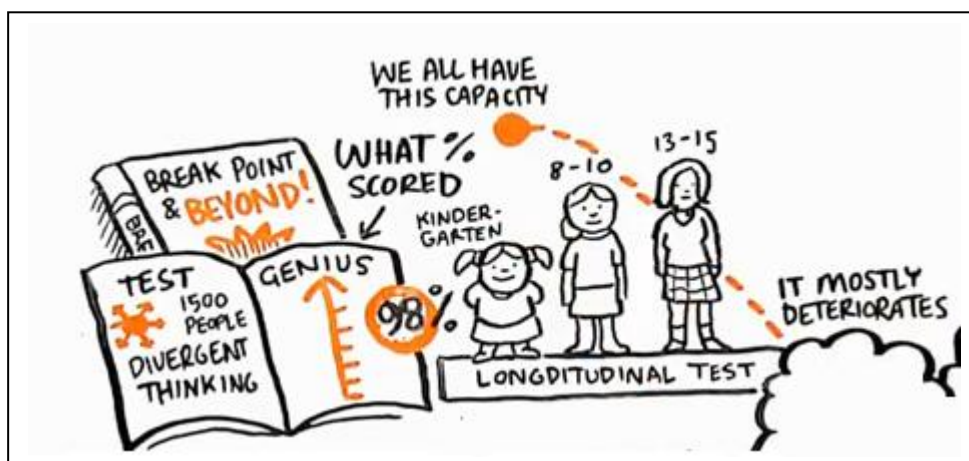
Engedtessek meg azonban, hogy néhány mondat erejéig kiemeljem a művészetek szerepét. Ken Robinson szerint: „A művészet elsősorban az esztétikai élményt célozza meg. És ez akkor jön létre, amikor az érzékszerveinket csúcsra járattuk. Amikor jelen vagyunk a pillanatban, és rezonálunk arra, amit átélünk, ilyenkor teljes mértékben élünk.” [9] Ezt különösen fontos minél többször átéreztetni a diákokkal, hogy ne csak az online közösségek felszíniességét érezzék.

Freund Tamás az agykéreg működésére vonatkozó vizsgálataira alapján állítja ugyanezt: „Ezek az eredmények és következtetések azt igazolják, hogy az érzelemvilág gazdagsága jelentősen befolyásolja tanulási képességünket, kreativitásunkat. Ezért kell oktatási rendszerünkben, elsősorban középiskolában, jóval nagyobb hangsúlyt kapnia az érzelemvilág gazdagítását szolgáló művészeti és erkölcsi nevelésnek. ... A művészeti élmények révén tárjuk szélesre a fiatalokban a befogadás folyosóját az agy és a külvilág között, hiszen ugyanezen a folyosón

közlekedik az alkotóképesség is, csak ellenkező irányban. Ezáltal nem csak kreativitásunk nő, hanem emberségesebb emberekké válhatunk...” [5]

A kreativitás egyik előfeltétele a divergens gondolkodás, vagyis az, hogy egy probléma megoldásának többféle módját is lássuk, arra való képesség, hogy egy kérdésre sokféle választ találjunk. Erre ragyogó példa a Niels Bohrról szóló közismert anekdota, amely szerint fizika vizsgáján az „unalmas” tanult válasz helyett 5-6 szellemes megoldást is adott a vizsgakérdésre. [13]

Ismét Ken Robinsonra hivatkozom, aki az egyik előadásában bemutatta egy, a divergens gondolkodás mérésére készült teszt eredményét. Idézem: „1500 emberrel megíratk egy tesztet, aminek az volt a címe, hogy A törésponton túl. Aki ezen elér egy bizonyos szintet, az zseni a divergens gondolkodásban. A kérdés az, hogy az 1500 embernek hány százaléka érte el ezt a szintet? Még valamit el kell árulnom, óvodásokról van szó. 98%! Ez egy longitudinális vizsgálat volt, és megismételték a gyerekekkel a vizsgálatot öt évvel később, 8-10 éves korukban. Mit gondolnak, mi volt az eredmény? 50%. Újra megvizsgálták őket öt évvel később 13-15 éves korukban. Látjuk a trendet ugye? ... Ez az egész két dolgot mutat. Egyrészt azt, hogy mindannyian képesek vagyunk a divergens gondolkodásra. A másik, hogy ez a képességünk az idővel romlik. Mi történik a gyerekekkel, amíg felnőnek? Sok minden. Meggyőződésem, hogy ezek közül a legfontosabb az, hogy iskolázzuk őket” [9]



Sajnos saját diákjaim körében is tapasztalom a divergens gondolkodás hiányát. Sokan „szenvednek”, ha többféle megoldást mutatok, akkor meg pláne, ha ezek között hibás is szerepel (szándékosan). Ugyanakkor részben vitatkoznék is Robinson megállapításával. Tény, hogy ebben a romló tendenciában komoly felelőssége van az oktatásnak, de nem hanyagolhatjuk el a média agymosó szerepét sem, illetve azt, hogy a gyorsan változó információ-özönben „nyerőnek” tűnő játékstratégia az elsőként adódó válasz elfogadása. Más kérdés, hogy az oktatásnak komoly figyelmet kellene szentelnie erre a jelenségre is.

Azt gondolom, sokan egyetértünk abban, hogy az oktatás alapos változtatásra szorul, azonban ez a sokat szidott, áldott és átkozott iskolarendszer ebben a düledező állapotában is működik úgy ahogy, és az a feladatunk, hogy ilyen keretek között is nyújtsunk valami biztatót.

3. Érzelmi motiváltság

„Az UNESCO szerint a következő 30 évben több ember szerez majd diplomát az oktatásban, mint a történelem kezdete óta. ... Hirtelen a diploma nem ér semmit. Amikor én tanultam, ha diplomád volt, munkád is volt. ... De most a diplomás gyerekek gyakran hazamennek videojátékokat játszani, mert ahová eddig elég volt a BA, most már MA kell, MA helyett meg PhD. Ez az akadémiai infláció folyamata. És azt mutatja, hogy az oktatás teljes struktúrája mozog a lábunk alatt.” Ezt 2006-ban mondta Robinson [8]. Benne vagyunk a jelzett folyamat közepében, és látjuk, hogy mennyit ér a diploma. Informatikus végzettséggel kicsit szerencsésebb a helyzet, de csak akkor van értéke, ha a diák jól megtanulja a szakmát.

A diákok zöme tisztában van evvel, mégis jó részük tehetetlenül várja, hogy sült galambként ölébe pottyanjon a semmit sem érő diploma. Csak találgatni tudok, hogy ennek mi lehet az oka, de csaknem biztos, hogy összefügg a divergens gondolkozás beszűkülésével, és véleményem szerint nagyon erős kapcsolatban lehet az úgynevezett tanult tehetetlenséggel, amelyről többek között Mérő László könyvében olvashatunk [6]. Ebben bemutatja a Martin Seligman nevéhez fűződő kutyakísérletet. A kísérlet lényege, hogy a kutyák egy csoportja megtanulta, hogyan javíthat a sorsán. A kontrollesoport sorsa a másik csoporttól függött, és nem a saját cselekedeteitől. Később mindkét csoportot olyan körülmények közé helyezték, ahol minden egyes állat tudott volna változtatni a helyzetén. Azt tapasztalták, hogy azok a kutyák, amelyek korábban megtanulták, hogy cselekedeteiknek hatása van, az új helyzetben is addig próbálkoztak, amíg megoldást találtak. Azok viszont, amelyeknek korábban nem volt lehetőségük befolyásolni a sorsukat, csak szűkölve lapítottak, meg sem próbáltak kimenekülni a kényelmetlen helyzetükből, lesunytt fejjel várták, hogy elmúljon.

Később a kísérletet emberekkel is megvalósították, és ugyanezre a felfedezésre jutottak, vagyis az ember, sőt, embercsoportok is képesek megtanulni a tehetetlenséget. Ha a diák kiskorától kezdve azt tapasztalja, hogy nem sok beleszólása van saját sorsának alakulásába, nem sok választási lehetősége van, akkor bizony könnyen tehetetlenné válhat.

Szerencsére Seligman és társai megsajnálják a tehetetlenségre megtanított kutyákat, és megpróbálták visszafordítani a folyamatot. Ezért a második kísérleti szituációban (amelyben lehetőség volt a változtatásra) igyekeztek megtanítani a kutyákat arra, hogy ugorjanak, mert ettől javul meg a helyzetük. „Rángatták szerencsétlen kutyákat, de azok egyre csak vonakodtak. Némelyikük azonban mégis megtanulta, és egy idő után magától is ugrott. Ezek a kutyák ezzel végleg kigyógyultak a tanult tehetetlenségből, a későbbiekben már minden helyzetben aktívan keresték a menekülés útját.” [6 (79. o.)] A tanult tehetetlenség tehát az esetek egy részében gyógyíthatónak bizonyult. Ezt a folyamatot illetik a tanult optimizmus kifejezéssel.

Vajon ki tudjuk-e ráncigálni diákjainkat a tehetetlenség állapotából? Azt gondolom, hogy bár nem kis munkával, de legalább egy részüket igen. Ennek azonban nagy ára van: teljes tanári jelenlétet igényel. Azt a jelenlétet, amelyet már a művészetek kapcsán is említettem (amikor jelen vagyunk a pillanatban, és rezonálunk arra, amit átélünk), azt a jelenlétet, amely szükséges az élet rezdüléseinek megtapasztalásához.

Véleményemet Suplicz Sándor kutatása is alátámasztja. Ezt írja: „Kutatásaink tanulsága szerint a tanárok érzelmi kompetenciái közvetlen hatással vannak a diákok tanulására. Az érzelmi hatások a személyiségen keresztül érvényesülnek. Akkor lehet eredményes a tanár, ha viselkedése és érzelmei harmonizálnak egymással és a helyzettel, ha hiteles, kongruens

személyként élük meg diákjai, ha elfogadhatóan, empátiával, tisztelettel és méltósággal kezeli diákjait.” [11 (10. o.)]

Ugyanezt állítja Csorba F. László is: „a tanítás mindig igényli a személyességet, a bizalomnak, a beleérző képességnek valamilyen szintjét. Ezért nem helyettesítheti a tanárt egy eszköz, például egy könyv vagy egy számítógépes program. Másodsorban: ez a személyesség eredendően nem közvetlen, hanem a 'tárgyon' (diszciplínán, tudományon) keresztül érvényesül.” [4] Bár nem tartozik a jelen cikk témájához, de roppant fontosnak tartom Csorba másik megjegyzését is, ezért ezt is idézem: „És semmi nem pótolhatja a tanár és tanár közti személyes kapcsolatot sem A diák ebből ugyan közvetlenül nem sokat érez, közvetve azonban annál többet.” [4]

Csermely Péter még ennél is továbbmegy, amikor ezt állítja: „A jó tanár nyitott és együttműködő. A jó tanár kifogyhatatlan annak a szeretethálózatnak az építésében, amely a diákjainak nyit új és új utakat. ... Egy ilyen, önfogadáson, önzetlenségen, bizalmon és értékkeresésen alapuló alapállás olyan energiákat mozgósít a diákok fejlődése érdekében, amelyek létéről addig senki még csak nem is álmodott.” [3]

A fentieket tudjuk, érezzük, tapasztaljuk, de van-e „racionális” alapjuk? Érzelmek nélkül nincs értelem. Mérő László azt mondja, hogy „a gondolkodás egyik fontos inputját mindig az érzelmeink jelentik, még a legracionálisabb gondolkodás esetében is.” [6 (110. o.)] Forgács József szociálpszichológus megfigyeléseire támaszkodva azt is írja, hogy „jó néhány társas helyzetben az érzelmeik erősen befolyásolják nemcsak az emberi kapcsolatok alakulását, hanem a gondolkodás módunkat is” [6 (129. o.)] Kísérletekre hivatkozva állítja, hogy a jókedv segíti a kreativitást. Érdekes módon ugyanakkor a rosszkedvnek is vannak előnyei. „Rossz hangulatban kevésbé hatnak döntéseinkre az előítéleteink, mint jó hangulatban.” [6 (130. o.)] Ahogy szellemesen megfogalmazza: „Nincs olyan, hogy pozitív vagy negatív gondolkodás, csak okos és buta gondolkodás van (és közöttük sokféle közbűlő színvonal). Az okos gondolkodás pontosan annyi pozitív, illetve negatív elemet tartalmaz, amennyit az adott téma épp indokol.” [6 (117. o.)]

A továbbiakban azonban mégis inkább a pozitívak közé sorolt érzelmekről írok majd néhány szót. Ismét Ken Robinsont idézem. Szerinte „három fő alapelve van az emberi élet virágzásának. ... Az első az, hogy az emberi lények természetüknél fogva különfélék és sokrétűek. ... Második alapja a kíváncsiság. ... A harmadik alap pedig ez: az emberiség természetéből adódóan kreatív. ... Ezért ennyire érdekes az emberi kultúra, ezért ennyire változatos és dinamikus.” [10]

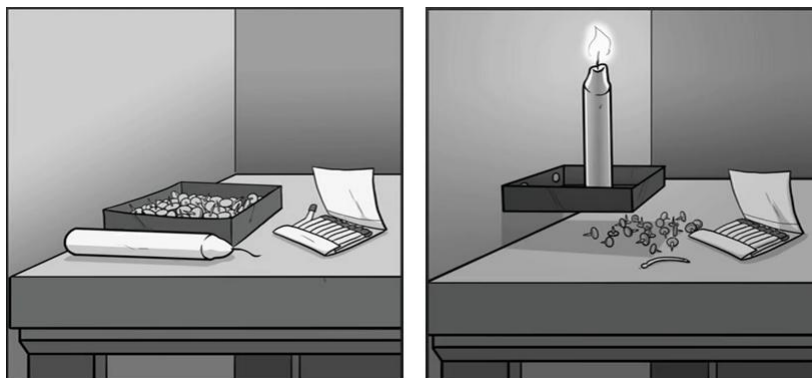
Ugyanebben az előadásában ezt is kijelenti: „Ha fellobbantjuk a kíváncsiság lángját a gyerekekben, nagyon gyakran minden további segítség nélkül maguktól tanulnak. A gyerekek született tanulók. Nagy teljesítmény elrontani ezt a képességet, vagy elnyomni azt. ... A tanítás egy kreatív hivatás. ... A jól felfogott tanítás nem egy átadó rendszer. A jó tanárok ezt teszik, de a jó tanárok ezen kívül még mentorálnak is, ösztönöznek, provokálnak, bevonnak. Végére is az oktatás a tanulás maga. Ha nincs tanulás, nincs oktatás sem. ... A tanár szerepe a tanulás előmozdítása, ez minden.” [10]

Ha szerencsénk van, akkor „csak” az a dolgunk, hogy ne oltuk ki a diák kíváncsiságát. Ha nincs ilyen szerencsénk, akkor pedig az, hogy fellobbantsuk azt. Ennek egyik – de szeretném aláhúzni, hogy korántsem egyetlen – eszköze a humor. A jó, nem erőltetett, szívből jövő humor. A rossz, destruktív humor sokat árthat. Suplicz szerint „Ha egy tanár alkatához nem áll közel a humor, attól még lehet jó tanár, de ha valaki a humorral, mint idegen tollal szeretne ékeskedni, könnyen visszajára fordulhat a népszerűség keresése.” [11 (51. o.)]

Ezt is hozzáteszi: „A diákok számára a stressz mentes, oldott légkör a pedagógiai hatékonyság és kreativitás szempontjából kívánatos. A humorérzék háttérében divergens gondolkozási képesség és kreatív elemek állnak, amelyeket az iskolákban gyakran hiányolunk.” [11 (52. o.)]

Azt, hogy a humor és a divergens gondolkozás között kapcsolat van, Blága Emese tudományos diákköri dolgozata is taglalja, illetve kísérletekkel támasztja alá. Dolgozatában idézi Robert D. Whisonant *The Effects of Humor on Cognitive Learning in a Computer-Based Environment* című 1998-as tanulmányának egy részletét, amely szerint „fiziológiai vizsgálatok kimutatták, hogy a humor megítélése és létrehozása az agy egészét mozgósítja. McGhee 1983-as kutatása során megállapította, hogy a humor megítélésekor mindkét agyféltekénk működésbe lép: a bal agyféltekénk, amely a nyelvért és a logikus gondolkodásért felelős, azonosítja és felismeri a viccben a részleteket, a csattanót, míg a jobb agyféltekénk segít a humor megítélésében és megértésében. Egy egyetemistákon végzett kísérletben divergens gondolkozást igénylő feladatokat kellett megoldaniuk a kísérleti személyeknek, és arra figyeltek fel a kutatók, hogy az egyetemisták a feladatok megoldásakor mindkét agyféltekéjüket használták. Ebből arra lehet következtetni, hogy a divergens gondolkodás szoros összefüggésben áll a humor megítélésével és létrehozásával, mivel mindkét folyamat során az emberi agy azonos régiói lépnek működésbe.” [2]

Talán ez lehet a fiziológiai háttere annak, hogy a jókedv segíti a kreativitást, vagy legalábbis a kreativitás bizonyos fajtáit. A kreativitással kapcsolatban gyakran emlegetik az úgynevezett gyertya-kísérletet. Eredetileg azt a kérdést vizsgálták vele, hogy vajon mennyire tudunk elrugaszkodni a tárgyak eredeti használati módjától, azóta azonban „több bőrt is lehúztak” a kísérletről. A megoldandó feladat: a kísérleti alanyoknak adnak egy gyertyát, egy levél gyufát és egy doboz rajzszöveget, és úgy kell a gyertyát a falhoz rögzíteni, hogy a meggyújtott gyertya ne csöpögjön le az asztalra. Később Barbara Fredrikson és Sam Glucksberg is megismételte a kísérletet más-más peremfeltételekkel. Fredrikson a kísérleti alanyok egy részét jókedvre hangolta a feladat kiosztása előtt. Ezt tapasztalta: „akik jó hangulatban voltak, lényegesen nagyobb százalékban és gyorsabban oldották meg a feladatot, mint azok, akik semleges hangulatban voltak. A legkevésbé sikeresek azok voltak, akiket korábban rossz hangulatba hoztak.” [6 (128. o.)] Ugyanakkor ez a különbség nem minden feladat esetén érvényes. Ha ugyanis úgy adták a feladatot, hogy a dobozból ki voltak öntve a rajzszövegek, és a doboz üresen hevert, akkor az emberek többsége sikeresen és gyorsan megoldotta a problémát. Ez esetben nem volt érdemi különbség a három csoport teljesítménye között. [6 (128.o.)]

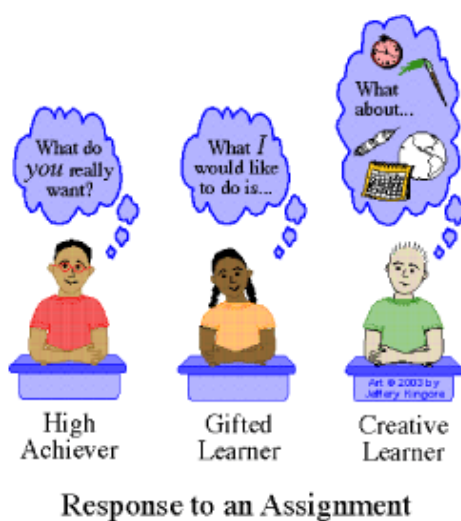


Glucksberg másképp csavarta meg a kísérletet: pénzjutalmat kínált azért, ha valaki minél gyorsabban oldja meg. Erről a kísérletről Dan Pink számolt be, és így folytatta: „A kérdés:

Mennyivel gyorsabban oldotta meg ez a csoport a problémát? A válasz: Átlagosan három és fél perccel hosszabb idő kellett a feladathoz. Három és fél perccel több.” [7]

Ő is kipróbálta a másik változatot is. Ez esetben az ösztönzés remekül működött. Dan Pink ezt így kommentálja: „A 'ha-akkor' típusú jutalmazás remekül működik azoknál a feladatoknál, ahol pár egyszerű szabály van, és a cél világosan látható. A jutalmak, természetüknél fogva, csőlátást okoznak, rájuk koncentrálnak az agyunk. Pont ezért működnek jól sok esetben. És az ilyen feladatoknál, mint ez, a csőlátás, ahol csak a célt látod, egyenesen rá fókuszálsz, nagyon jól működik. De az igazi gyertya problémánál nem így akarsz nekilátni. A megoldás nem itt van. A megoldás nem kézenfekvő. Tágabb körben akarsz vizsgálgódní. A jutalom tulajdonképpen csőlátást okoz, és korlátozza a lehetőségeinket.” [7]

Amikor feladatokat adunk a diákjainknak, lehetőség szerint érdemes figyelembe venni ezeknek a kísérleteknek az eredményeit.



4. Programozási feladatok

Egy korabeli szóbeszéd szerint Sir Christopher Wren, a londoni Szent Pál katedrális építészé egy alkalommal a kőfaragók között járva arról faggatta őket, hogy mit csinálnak. „Követ faragok” – jött az első, gondterhelt, kissé ingerült válasz. „Keményen dolgozom, hogy elstartsam a családomat” – felelte a második munkás. A harmadik kőfaragó viszont ragyogó arccal ezt válaszolta: „Katedrális építék!”

Végül szeretnék néhány mondatot mondani arról, hogy hogyan próbálok alkalmazni az eddig elmondottakat a programozás oktatása során.

Mint a legtöbb felsőoktatási intézményben, nekünk is elég szigorúan definiált keretek között kell dolgoznunk, még akkor is, ha a keretek egy részét mi magunk írjuk elő. Azok a diákok, akik eleve programozónak készülnek, vagy már évek óta programozgatnak, maguktól is tudják, hogy csak nagyon sok önálló munkával lehet megtanulni programozni. Viszont sok olyan hallgató van, akiket rá kell szoktatni arra, hogy önállóan is dolgozzanak. Természetesen erre megvannak a szigorú „motivációs” feltételek: zárthelyik, vizsgák, kötelező feladatok, de azt szeretném, ha nem csak ügyes „kőfaragók” lennének, hanem lehetőleg minél több „katedrális-építő” kerüljön ki közülük. Ezért szükség van olyan motivációs eszközökre is, amelyek hatására egy idő után belső igénnyé válik a jól végzett munka. Mérő ezt mondja a gyertya-

probléma kapcsán: „Kiderült, hogy az igazi motiváció kulcsa valahol magában a munkában van, azt kell bizonyos értelemben gazdagabbá tenni.” [6 (273. o.)]. József Attila szebben ír: „Az én vezérem bensőmből vezérel!”.

Egyik diák pedig így fogalmazta meg ugyanezt szakdolgozata bevezetőjében: „Eleinte csak a tárgy nehézségétől félve töltöttem el rengeteg időt programozással. Végül azt vettem észre magamon, hogy egyre több, megoldáskereséssel eltöltött idő után kezdem megszeretni a programozást. Maga a sikerélmény, ahogy „életre kel” az általunk megírt programkód, további lelkesedést vált ki az emberből, újabb és újabb feladatok megoldásával fokozva ezt.”

Azt gondolom, hogy az egyik, idővel belsőbe áthajló motiváció lehet az, ha igyekszünk minél érdekesebb, esetenként humoros, ötletes feladatokat kitalálni és kiadni. Akár zárthelyi példaként is. Egy szellemes feladat esetenként feszültségoldó is lehet. Talán még akkor is, ha maga a megoldandó probléma nem is olyan könnyű. Persze, korántsem biztos, hogy ez mindig sikerül, mint ahogy az sem, hogy amit az egyik ember szellemesnek talál, azt a másik is. De akkor is próbálkozni kell vele. Néhány érdekesebbnek szánt feladat olvasható [1] elején.

A másik nagyon jó motivációs tényező, ha bizonyos mértékig szabad kezet adunk a diákoknak, és engedjük, hogy használhassák a fantáziájukat, kreativitásukat.

Nagyon szeretem a programozás (Java) vizsgát. És ami sokkal fontosabb: a diákok jó része is szereti. A legutóbbi vizsgaidőszakban a hallgatók 49.3%-a választotta a vizsgájukra vonatkozó állításokat tartalmazó anonim felmérés során a „jó buli volt” opciót, és csaknem 75%-uk azt, hogy „sokat tanultam belőle” (73 vizsgázó volt).

Mi a titka ennek a „sikernek”? Meggyőződésem, hogy az, hogy kreatívak lehetnek. Mégpedig azért, mert a vizsga ebből áll: önállóan ki kell találni egy megoldandó feladatot, programot írni rá, majd az aznap vizsgázók körében bemutatni. A vizsga tétje látszólag az, hogy a zárthelyik alapján elért jegyet eggyel javítani vagy rontani lehet (vagy persze, szinten tartani), de az igazi tét messze nem ez, hanem az, hogy megízleljék a jól végzett munka örömét. És sokan meg is ízlelik. Persze, vannak, akik épp csak a minimumszintet érik el. Valószínűleg olyanok is vannak, akik nem is maguk írják meg a programot, de az igazán fontos az, hogy sok olyan diák is van, aki valóban élvezzi a feladatot. Nem egy már a félév elején elkezdni csinálni, van olyan is, aki már a tárgy felvétele előtt azon gondolkodik, hogy mit is készítsen majd. Úgy látom, hogy az, aki össze tudja kapcsolni a feladatot a hobbjával, vagy eszébe jut, hogy milyen programnak örülne a barátnője, különösen jó dolgot tud készíteni. Messze jobbat, mintha megkötném a kezét egy sablonosan megfogalmazott feladattal. Néhány – ad hoc módon választott – vizsgafeladat képernyőképe látható [1] 18. oldalától.

Írásom zárásaként két dolgot szeretnék megemlíteni. Az egyik az, hogy bár mindegyik tehetséges tanítványomnak örülök, sokat tanulok tőlük, de legbüszkébb azokra a 3/3-as¹ diákokra vagyok, akik vagy Java fejlesztők lettek később, vagy legalább 3-as, 4-es szinten teljesíteni tudták a tárgyat, és főleg: örültek az elkészült projektjüknek. Egy ilyen diáktól kaptam a napokban ezt a levelet:

„Csak jelezni szeretném, hogy nem tudom, hogy a Tanárnő készítette-e a java államvizsga feladatokat, de tetszettek. És aránylag még jól is sikerült szerintem. Ha lett volna még egy kis időm, akkor szerintem meg tudtam volna oldani. Viszont abban a két órában olyan érzésem volt, amit nem gondoltam, hogy valaha lehetséges lesz. Sosem gondoltam volna, hogy élvezni fogom a java programozást :)”

¹ Viccesen így nevezik azokat a diákokat, akik kimerítették a lehetséges keretüket, és harmadszorra is felvették a tárgyat.

Épülnek majd katedrálisok? Szeretném azt hinni, hogy igen. És a mi diákjaink közül is lesznek építők. Erre utal az alábbi anonim vélemény is:

„A következő hallgatóknak én is csak annyit tudok mondani, hogy készüljenek, gyakoroljanak sokat, mert szép dolog létrehozni valamit, több ez mint egy technikai eszköz, és ha így állnak hozzá, sokkal könnyebb lesz teljesíteni a tárgyat is. Köszönöm, hogy itt lehettem.”

(Néhány további vélemény olvasható [1] 25. oldalán.)

5. Összegzés

Cikkemben tapasztalataimat, véleményemet igyekeztem alátámasztani pszichológusok, agykutatók, oktatáskutatók véleményével, kutatási eredményeivel. Ha a leírtak arra ösztönzik a kedves olvasót, hogy elolvassa az irodalomjegyzékben ajánlott könyvet, cikkeket, és megnézzé a javasolt videókat, akkor írásom elérte a célját.

6. Köszönetnyilvánítás

Köszönetet mondok volt tanárainknak, családomnak, barátainknak, kollégáimnak, hajdani és mostani diákjainknak.

Irodalomjegyzék

- [1] Achs Ágnes: Programozási feladatok.
<http://pmmik.pte.hu/data/2014/0626/085/feladatlapok.pdf>
- [2] Blága Emese: Humor, intelligencia és zárt gondolkodás – XIII. Erdélyi Tudományos Diákköri Konferencia, 2010. május 14-16.
http://etdk.adatbank.transindex.ro/pdf/pszi_blaga.pdf
- [3] Csermely Péter: Milyen a jó tanár? (2010.) <http://csermelyblog.tehetsegpont.hu/node/28>
- [4] Csorba F. László: A tanár személyisége a természettudományok oktatásában - A barlanghasonlat – Új Pedagógiai Szemle, 1998. június
<http://www.ofi.hu/tudastar/tanar-szemelyisege>
- [5] Freund Tamás: Hullámtörés - a marihuána hatása az agyhullámokra és a memóriára, Mindentudás Egyeteme, V. szemeszter, 12. előadás, 2004, november 22.
<http://mindentudas.hu/elodasok-cikkek/item/91-hullam%20tore%20-%20a%20marihuana-hatasa-az-agyhullamokra-es-a-memoria-ra.html>
- [6] Mérő László: Az érzelmek logikája, Tercium kiadó, Budapest, 2010.
- [7] Dan Pink: The puzzle of motivation (2009.)
https://www.ted.com/talks/dan_pink_on_motivation

- [8] Ken Robinson: How schools kill creativity (2006.)
http://www.ted.com/talks/ken_robinson_says_schools_kill_creativity
- [9] Ken Robinson: Changing education paradigms (2010.)
http://www.ted.com/talks/ken_robinson_changing_education_paradigms
- [10] Ken Robinson: How to escape education's death valley (2013.)
http://www.ted.com/talks/ken_robinson_how_to_escape_education_s_death_valley
- [11] Suplicz Sándor Tanárok pszichológiai jellemzői diákszemmel, PhD értekezés, Debreceni Egyetem BTK, 2012. <http://ganymedes.lib.unideb.hu:8080/dea/handle/2437/161946>
- [12] Nincs szerző: Generáció C: a digitális kor véleményvezérei, ELTE Társadalomtudományi Karának hírlevele, 2010. 11. 08.
http://www.sociomater.hu/hu/110074/Generacio_C_a_digitalis_kor_velemenyezerei.page
- [13] Niels Bohrról szóló legenda: <http://epa.oszk.hu/00800/00804/00428/60114.html>



Mesterséges intelligencia oktatása Python segítségével

Teaching Artificial Intelligence by using Python

Aszalós László

Debreceni Egyetem, Informatikai Kar

laszalos@unideb.hu

Absztrakt: A Debreceni Egyetemen a Mesterséges intelligencia tantárgy oktatásának évtizedek óta része a különféle algoritmusok megismertetése a hallgatókkal. Ehhez az algoritmusokat több módon is megfogalmazzuk, a pszeudokód mellett szépen sorra kerültek az épp divatos, akkoriban oktatott programozási nyelvek, így a Pascal, C és újabban a Java is. A Programozó matematikus majd Programtervező informatikus hallgatóknál ez a megközelítés hatékony, és egyéb kompetenciákat is fejleszt. Viszont a programozás iránt kevésbé motivált Mérnök és Gazdaságinformatikus diákok esetén más megközelítést kell választani. A képzés részleges megújítása során, a Russel-Norvig féle tankönyvre áttéréssel felmerült annak lehetősége, hogy a korábban használt programcsomagok helyett másokat alkalmazzunk az előadások és az otthoni felkészülés során. A tankönyv elterjedtsége miatt már kész alapokat kaphatunk meg különböző programnyelveken. Viszont ezeket az aktuális képzéshez át kell alakítani, illetve ki kell egészíteni. A cikkben leírt nagyobb témakörök közel felében ez már megtörtént, a többi esetben pedig folyamatban van.

A lehetséges programnyelvek közül hosszas meggondolás után a Python programnyelvet választottuk, melynek értelmező (interpreter) megvalósítása miatt az egyes feladatok implementálása, futtatása leegyszerűsödik, és emiatt felgyorsul. A Python több éve tartó verzióváltása miatt a korábbi verzióban elkészült kód újraszervezése és leegyszerűsítése szükséges, hogy hallgató minél egyszerűbb és könnyebben emészthető forrással találja szembe magát. Ez természetesen kicsit hátráltatja a céljaink elérését, de reméljük, hogy összességében a diákjaink nyerne a vele. Habár a fő hangsúly az algoritmusokon van, fontos azok hatékonyságának mérése, illetve a módszerek vizualizációja. Az előbbi részben már implementálva van az elérhető programokban, ami segíthet megcáfolni pár előzetes elképzelést, viszont az utóbbira ezekben még senki nem fordított figyelmet. A mesterséges intelligencia oktatásában használt más rendszereknél ezek jól működnek, ezért mi is célul tűztük ki a megoldás, illetve az ahhoz vezető út vizualizációjának implementálását. Reményeink szerint ezzel felkelthetjük a diákok figyelmét, és ezzel segíthetjük az ismeretek elsajátítását.

Kulcsszavak: mesterséges intelligencia, Python programozási nyelv, oktatóprogram

Abstract: At University of Debrecen at the course of Artificial Intelligence, various algorithms are taught. These algorithms are formulated in a number of ways, in addition to the pseudo code, they are addressed in favored programming languages, such as Pascal, C and Java. The students of the curriculum Programming Mathematics mathematician are able to profit from this approach. However, our student of IT Engineering, and Business IT who are less motivated towards programming, need a different approach. During a partial renewal of the curriculum, the Russell-Norvig AIMA textbook provided us with the possibility to replace the software packages used in the presentations and in the students' self-preparation. The popularity of this textbook meant the availability of many software in different programming languages. However, these need be converted to the actual program, or need to be supplemented. We have updated more than half of the software, and the others are in progress.

After a lengthy consideration, the Python programming language was chosen. The source code written in a previous version of Python needs a complete refactoring to become more simple and readable. This has a bit hampered the achievement of our goals, but we hope that overall our students benefit from it. Although the main focus is on the algorithms, it is important to measure their effectiveness, and visualize the running of the algorithms. The former has been partially implemented in the programs so it could help to disprove a few preliminary ideas; however no one paid attention the latter yet. Other systems used in the teaching of artificial intelligence do well, therefore we plan to implement in our system. We hope to raise the attention of the students, and help them learn the skills.

Keywords: artificial intelligence, Python programming language, tutorial software

1. Bevezetés

Az 1972-ben elindított *Programozó matematikus* képzés közel két évtized alatt jutott el addig, hogy szükségessé vált a folytatásának (ezt ma mesterképzésnek hívjuk) a bevezetése. Szükség esetén ehhez hazai és külföldi oktatók segítségét is igénybe vettük. Sajnos a Mesterséges intelligencia tantárgy oktatása esetén ez külső okok miatt nem jött össze, így Dragálin Albert helyi logikaprofesszor vállalta el a tantárgy oktatását, melyet haláláig folytatott.

Akkor a *Mesterséges intelligencia* elnevezés négyszemeszternyi tananyagot takart, ahol az egyes féléveket egy-egy sorszám azonosította. Az első félév az állapottér reprezentációt, valamint a hagyományos keresési módszereket (mélyégi, szélességi, A*, alfa-béta) tárgyalta, beleértve a kétszemélyes játékokat. A második félév az automatikus tételbizonyítás témakörét járta körül. A harmadik félév a Prolog nyelvet, míg a negyedik a szakértői rendszereket ismertette meg a hallgatókkal. Az évek folyamán Várterész Magda valamint Bognár Katalin átvette egyes félévek oktatását, illetve én is szerepet kaptam Dragálin Albert halála után.

A Bolognai rendszer bevezetése során a négyszemeszternyi tananyag megmaradt. Az első félév minden alapképzésben részt vevő Programtervező informatikus számára kötelező, míg a rákövetkező három félévet a Mesterséges intelligencia szakirányt választó, mesterképzésben részt vevő diákoknak szánjuk, bár más szakirányú diákok is felvehetik. Az új képzéssel a tárgyak a tartalomhoz illő nevet kaptak. Természetesen az átnevezés mellett hangsúlyeltolódások is történt, a *Mesterséges intelligencia 3* elnevezésű tárgy a *Mesterséges intelligencia nyelvei* nevet kapta, ahol a Prolog helyett időnként más nyelvek is feltűnnek.

Ahogy beindul a Bolognai rendszerben a *Mérnök informatikus és Gazdaságinformatikus* alapszintű képzés, ott is kötelező tárgy lett az első féléves Mesterséges intelligenciái kurzusból, ugyanazzal a tematikával, mint a Programtervező informatikus diákok esetén. A létszámok megnövekedése miatt már nem volt tartható, hogy egy oktató tanítsa mind a három társaságnak az előadást, így mindegyik szak külön oktatót kapott. Ennek eredményeképp lett a Gazdaságinformatikus képzés oktatója Mihálydeák Tamás. Ő új elképzeléssel vágott bele az oktatásba. A közel húszéves tematikát jelentősen átalakította, aminek persze megvoltak az indokai. A Mérnök és Gazdaságinformatikus képzések esetén még a mesterképzés során sincs ráépülő tananyag, tehát eme diákok soha többé szervezett képzés részeként nem fognak többet ezzel a témakörrel foglalkozni. Így nem biztos, hogy egy az egyben a PTI képzésre kitalált tematikát kell követni.

Miután Russel és Norvig átfogó könyve [1] – mely első kézből ad betekintést az aktuális alkalmazási, valamint kutatási irányokba – elkészült, és a Panem kiadó és a Tankönyvtár jóvoltából ingyen elérhető magyar nyelven [2], másrészt miután az elmúlt évtizedben több mint ezer egyetemen ez alapján oktatják a mesterséges intelligenciát – ezért igen bőséges oktatási segédlet érhető el –, így érdemes volt számunkra is alkalmazkodni ehhez a könyvhöz. Természetesen az előző képzés tematikája nem volt ezzel ellentétes – a korábban illetve újabban oktatott témakörök 60-70 százalékban fedik egymást –, ám teljesen más a megközelítés módja, ezért az a váltás a segédanyagok újbóli elkészítését tette szükségessé. Ez megtörtént már az előző évben az előadásokon használt fóliák esetén, és folyamatban van az

algoritmusok illusztrálására, kipróbálására, begyakorlására használt rövidebb-hosszabb programkódok esetében is. Ebben a cikkben ez utóbbiról írunk részletesebben.

A következő fejezetben ismertetjük a hallgatók előképzettségét, amit majd a Russel-Norvig kötet mellé jelenleg elérhető segédanyagok ismertetése követ. Ezután a tananyag nagyobb témaköreit bejárva, kitérünk rá, hogy milyen fejlesztések történtek már meg, illetve mit tervezünk a következőekben. Végül összefoglaljuk a cikket.

2. Hallgatói előképzettség

A *Bevezetés a mesterséges intelligenciába* tantárgynak két előfeltétele van, egyik *Az informatika logikai alapjai*, míg a másik a *Magasszintű programozási nyelvek*. Mindkét tárgy valóban szükséges kompetenciákat oktat: az állapottér reprezentáció gyakran logikai formulákat használ, és a tudásalapú ágens logikai következtetések alapján dönt a soron következő lépésről; míg az egyes keresési algoritmusok megértéséhez, vagy esetleges implementációjához szükséges a programozói gyakorlat.

A *Magasszintű programozási nyelvek* tantárgy sokak számára vízvázasztó, mert igen nehezen teljesítik a követelményeket. Pedig e tantárgy keretein belül csak a végletesen leegyszerűsített programkódokkal találkozhatnak a hallgatók. A több mint 100 soros, nem megszokott, elcsépelet tankönyvi mintapéldán túllépő, valódi problémát megoldó forráskóddal rendszerint a Mesterséges intelligencia keretében találkozhatnak a hallgatók először. Ez persze megint sok embernek meg nem ugorható akadályt jelent, és ezért itt is nagy a lemorzsolódás aránya.

Ennek valószínűleg az is az oka, hogy a diákok úgy jelentkeznek informatikus szakra, hogy nincsenek programozási előismereteik. Ez jellemző mindhárom informatikus szak esetén. Míg a programtervező és mérnök informatikus hallgatók jellemzően egy programnyelvet tanulnak (Java illetve C++ egy halvány C alapozás után), a gazdaságinformatikus hallgató képzése igen szerteágazó: C alapozás, Visual Basic, C elmélyítése, esetlegesen C++ valamint Java. Miután sok diák a közgazdász-képzés ellehetetlenítése miatt választja a gazdaságinformatikus szakot, így itt a legkisebb a diákok motivációja a programozás iránt. Éppen ezért próbáljuk visszaszorítani ezt a programnyelvi sokszínűséget, hátha egy nyelv alapos megismerése segít a programozás elmélyítésében, mely után már akár önképzéssel is megtanulható bármely másik nyelv.

Az elmúlt években tapasztalható tendenciákat figyelve úgy látom, hogy egyre több tankönyv használja a Python nyelvet; cikkekben többsoros algoritmusok helyett elég 1-2 sor Python kód valamely elterjedt programkönyvtár rutinjaira alapozva; a nagyobb egyetemeken egyre több és több bevezető kurzusa a Python-t használja; sokan a Raspberry Pi miniszámítógépre és a Python programozási nyelvre alapozva [3, 4, 5] próbálják felkelteni a tíz év körüli tanulók figyelmét a programozás iránt. Éppen ezért megfontolandó ennek a nyelvnek előtérbe helyezése, és első nyelvként tanulása az egyetemen a programozás iránt nem igazán elhivatott diákok számára.

3. Elérhető segédanyagok

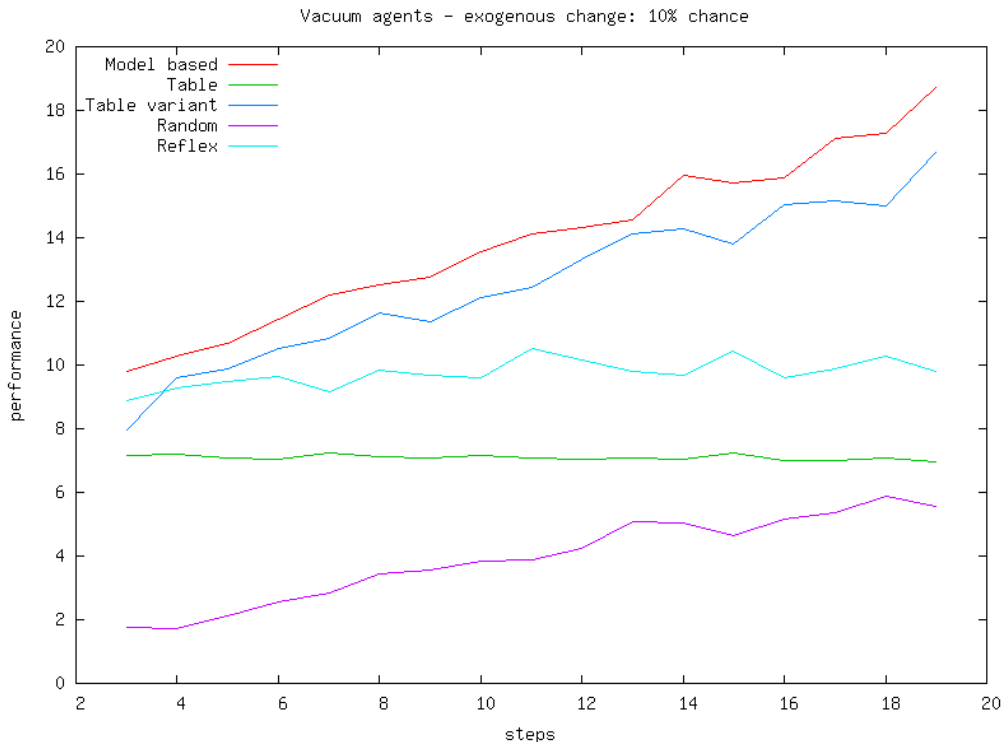
A Russel-Norvig könyv honlapján [6] keresztül elérhetjük a könyvben szereplő pszeudokódokat, de ugyanitt elérhetőek Java, Lisp és Python nyelvű programok is. A Lisp nyelvű kódok közelebb vannak a húsz évhez, mint a tízhez, bár ez a minőségükben nem jelenik meg. Viszont alkalmazásának korlátot szab, hogy egyetlen diákunk sem tanulja ezt a nyelvet szervezett keretek között. A Java programok esetén az életkor így nem állapítható meg, viszont a harmadik kiadáshoz kapcsolódóan egy frissítésen estek át. A Java nyelvű programok előnye, hogy többet közülük telepítés nélkül is ki lehet próbálni egy böngészőben. Jellemzően az egyszerű ágenseket bemutató, illetve hagyományos keresési algoritmusokat ismertető kódokat érhetjük el így. A Debreceni Egyetemen saját fejlesztésben elkészült egy programcsomag, mely a keresési módszerek implementálásában nyújt segítséget [7]. Az általam oktatott Gazdaságinformatikus hallgatók a Mesterséges intelligencia kurzus időpontjában még nem ismerik ezt a nyelvet, így számunkra ez sem tekinthető jó megoldásnak. A honlapról elérhető Python nyelvű kód 2007-es keltezésű. Azóta ez egy-két ráncfelvarráson átesett, ám gondot okoz a Python verzióváltása. Ha valaki a korszerűbb, kicsit olvashatóbb Python3 kódot szeretné, akkor sokat kell keresgetnie utána. Van még a könyvhöz kapcsolódó, ám már nem karbantartott Prolog, C és C++ forráskód is.

Felmerülhet, hogy mit kívánunk elérni ezekkel a kódokkal, miért nem jók a könyvben szereplő pszeudokódok? Az egyes szakokon tapasztalható különböző szintű motivációk miatt a válaszok is különbözőek lehetnek. A Programtervező informatikusok esetén fontos, hogy konkrét példát lássanak egy összetettebb rendszerre, azt saját igények alapján kiegészíthessék, és például a szakdolgozatukban felhasználhassák. Erre az előbb említett programcsomag [7] tökéletesen megfelel, főleg hogy az oktatásban leginkább használt programnyelven íródott. Az általam oktatott gazdaságinformatikus diákoknál a készen kapott programcsomaghoz fejlesztés igen kevés diáknál jön szóba. Itt inkább a figyelemfelkeltés, az ábrázolás, számítógéppel segített tanulás és bizonyos tévképzetek megcáfolása lehet a cél. A programozással szembeni ellenérzéseket részben a minél rövidebb programokkal lehet eloszlatni. Mindezek a vezettek oda, hogy az oktatás során bemutatott algoritmusok, rutinok esetén a Python nyelvet használtam. A Python adatszerkezeteinek és kapcsolódó műveleteinek bőséges választéka jelentősen lerövidíti az implementációt, gyakran még a pszeudokódnál is tömörebben fogalmazhatunk. Az angol nyelvet ismerők könnyedén olvashatják a Python forrást, még jelentősebb programozói ismeret nélkül is. Hosszan el lehetne vitatkozni azon, hogy milyen programozási nyelvet érdemes tanítani a diákjainknak, ám egyre több, különféle rangsorok elején álló egyetemen kezdik a programozás oktatását a Python nyelvvel.

4. Tematika és segédletek

A Russel-Norvig tankönyv [1] szellemében a kurzusunkban az ágens megközelítés kerül a középpontba. Ennek megfelelően a szemeszter a mesterséges intelligencia fogalmának, feladatának és lehetőségeinek tisztázása után az első pár előadásnak az ágens és környezete kapcsolatának bemutatása a feladata. A jellemző példa ebben az esetben a robotporszívó, mely több helyiség takarításáért felelős. A tankönyvhöz elérhető példaprogramok több fajta ágenst is implementálnak, kihasználják az objektumorientáltság adta lehetőségeket, és ehhez bőséges megjegyzés mellett közel ötszáz sort használnak fel, melyből majd száz sort egy ki nem használt GUI kezdeményezés foglal el. Ezeket a definíciókat alkalmazva egy reflex-, egy

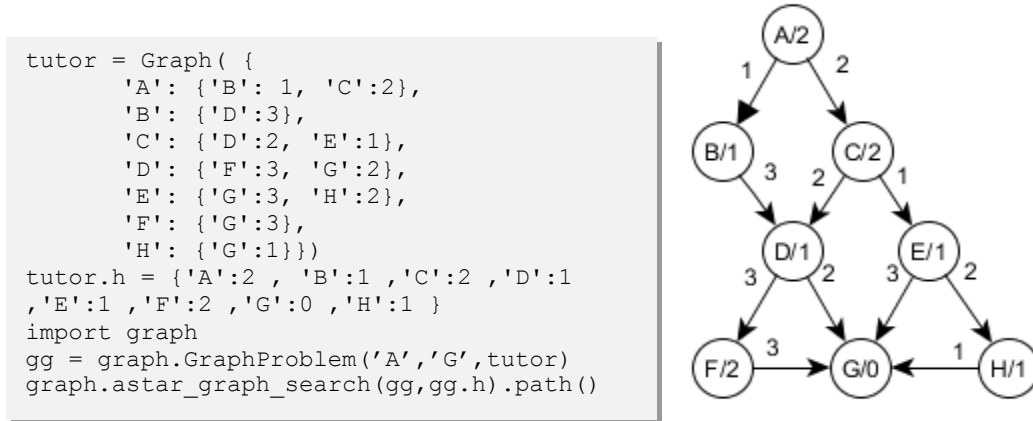
véletlen, egy táblázat-vezérelt, és egy modell-alapú ágenszt definiáltam, melyeknek működését követhetjük nyomon, illetve takarítási hatékonyságukat (statisztika) a környezet tulajdonságaitól függően (lásd 1. ábra). Ehhez természetesen pár sor forrás elegendő volt. Ahhoz, hogy egy célorientált ágenszt megvalósítsunk – pl. a robotporszívónak ki kell takarítania az egész lakást úgy, hogy nem ismeri a geometriáját; vagy a Wumpus világban be kell gyűjteni az aranyat az ágensnek, miközben elkerüli a csapdákat és a szörnyet – már diákokot próbáló feladat, akár helyi verseny feladata is lehet, hasonlóan egy-két amerikai egyetem gyakorlatához. Sajnos a diákjaink erre nem motiváltak, pedig sokat lendítene a tananyag elsajátításában.



1. ábra. Különböző típusú ágensek hatékonysága kis mértékben változó környezet esetén.

Bizonyos feladatok megoldásához az ágensek kereséseket alkalmaznak. A keresések ugyancsak sok megjegyzéssel ellátott kódja elfér Pythonban kétszáz sorban, noha implementálva van a mélységi, szélességi, és optimális fa- és gráfkeresés, korlátos és iteratív mélységi keresés, best-first és A* algoritmus. Igaz, a perem kezelésével kapcsolatban közel 80 soros implementációja van a sornak, veremnek és a kupacnak. Ezt – mint programkönyvtárat – használva egy hagyományos probléma – mint a kancsós öntögetési feladat, vagy a hanoi tornyok problémája – húsz-harminc sorban megfogalmazható, így akár egy probléma megfogalmazása, vagy akár annak az implementálása kiadható feladatként. Tetszőleges gráf megadása asszociatív tömbök asszociatív tömbjeként nagyon egyszerű (2. ábra), így közel száz sor hosszúságú, gráfokhoz kapcsolódó kódot felhasználva, mindenféle gráffal leírható feladat lépésről lépésre követhetővé válik. Mivel a gyakorlatokon a különféle algoritmusok elsajátítása konkrét gráfokon történik, a megoldások ellenőrzése, illetve újabb és újabb konkrét gráfokon lefuttatása segítheti a begyakorlást, illetve a megértést. Másrészt igen gyakran van a sorrendiségről téves elképzelései a diákoknak. Például arról, hogy az informált kereséseknél a minimális heurisztikájú csúcs választása

hogyan történik, ha több ilyen csúcs is létezik. A program által megadott megoldások átvizsgálása talán sikerült megszüntetni ezeket az elképzeléseket. A diákoknak kiadott állapottér-reprezentációhoz statisztikák elkészítésére alkalmazható kódot is mellékelünk, és bemutattuk, hogy ugyanazon feladat megoldásához vezető különféle algoritmusok mennyi munkát igényelnek a számítógéptől.



2. ábra. Gyakorlógráf megadása és rajta A* keresés futtatása.

A keresési algoritmusok egyik alkalmazása a kényszerkielégítési problémák megoldása. Ezeket rendszerint rejtvényeken keresztül próbáljuk oktatni. Egyik eklatáns feladat a Sudoku rejtvény, amit mindenki ismer, így ezek után nem nehéz megfogalmazni kényszerekkel. Ebbe a csoportba tartozik a térkép kiszínezése három vagy négy színnel, n királynő elhelyezése egy $n \times n$ -es sakktablán vagy az Einstein-féle Zebra rejtvény. Ez utóbbiak mindegyike implementálva lett Python nyelven, de a kód revíziója, egyszerűsítése még várat magára. Addig ezt a részt, mint egy fekete dobozt használhatják a diákok. Miután kényszerkielégítési feladatok és megoldásaik megtalálhatóak egyik feladatgyűjteményünkben [8], így egyelőre még ezt használtuk példának, de az egységesítés érdekében a közeljövőben át szeretnénk térni a Python megvalósításra. A mai diáknak – főleg ha nem ő küszködik meg a program implementációjával – nem igazán köti le a figyelmét az, hogy hosszabb számolás után a program eredményt szolgáltat, és azt kiírja valamilyen formátumban (3. ábra). Persze lehet játszani a visszalépéses keresésnél, hogy melyik heurisztikát használjuk, és ezzel mennyire gyorsul fel a megoldás megtalálása, de ez nem mutatja meg a lényegét, a megoldás születését. Ezért érdemes lenne a program végrehajtása során a megoldás állapotát megjeleníteni. Talán a leglátványosabb a térképszínezési feladat közbenső állapotainak ábrázolása lehet – ha a térképen valóban színeződnek különböző színekkel az egyes megyék –, és ekkor nyilvánvalóvá válna, hogy melyik megoldási módszernek melyek a hátulütői, illetve mi emberek milyen heurisztikákat használunk egy feladatmegoldás során.

```
hu = csp.MapColoringCSP(list('RGBY'),
    """BAZ: SzSzB HB JNSz H N; SzSzB: HB; HB: JNSz B; JNSz: H P BK Cs; B: Cs;
    H: N P; N: P; Cs: BK; BK: P B T F; P: F KE; KE: F GyMS Ve; F: T S Ve;
    T: B S; B: S; S: Ve Z; Ve: GyMS Va Z; GyMS: Va; Va: Z""")
csp.backtracking_search(hu, mcv=True, lcv=True, fc=True, mac=True)
csp.min_conflicts(hu)
```

3. ábra. Magyar megyék megadása, és kiszínezése négy színnel két különböző módszert használva.

A keresés ellenséges környezetben – azaz a hagyományos kétszemélyes játékok – egy másik alkalmazása, illetve inkább továbbfejlesztése a keresési algoritmusoknak. A Python nyelvű kódrészletek még itt is csak az alapot jelentik, bár igen stabil alapot biztosítanak. Az alfa-béta vágással akár a felhasználó ellen is jól játszó játékot lehet percek alatt összedobni, de ez túllép a tervezett tananyagot. Fejlesztést igényel a nyerő stratégia megadásához szükséges címkézése a játéknak, ami viszonylag egyszerű feladat. Szükség lenne még a játékok leírására használt kód tisztítására, és felhasználó elleni játéknál az input jobb kezelésére. Bár tartalmi változást nem jelent, de a figyelem felkeltését itt is növelhetné a játékok köré felhúzott grafikus felület, illetve a játék teljes gráfjának/fájának ábrázolása.

A tudásalapú ágensek esetén a fontos eszköz a következtetés, illetve az azt megalapozó logika. A végeláthatatlan nulladrendű logikai leírást gyakran lerövidítheti az elsőrendű logika alkalmazása. Viszont amit a nyerünk a réven, ki kell fizetni a vámon. Annak érdekében, hogy egyszerűnek tűnő kérdésekre a programunk válaszolhasson, az *Automatikus tételbizonyítás* című MSc tárgy tananyagának teljes implementációjára szükség van. Ezt meg is kapjuk a tankönyv honlapjáról, de a szűkösre mért előadások közül egyet felemésztene csupán ennek az implementációnak a bemutatása. Az önálló tanulást elősegíthetné, ha az előre- illetve a hátrafele láncolást tetszőleges példára lehetne alkalmazni, így ehhez kapcsolódóan ezt kívánjuk implementálni nulladrendben.

A tudásalapú ágensek nagyszerűen alkalmazzák a logikát, de a valós életben igen ritkán vagy fekete, vagy fehér valami; tehát szükségünk van további árnyalatokra is. Ezt a valószínűségek hozhatják be a modelljeinkbe. A valószínűség alkalmazása az egyik oka a mesterséges intelligencia rejtett szárnyalásának. A képzés során ennek csak egy picike szeletét tudjuk az időhiány miatt bemutatni, és ez nem más, mint a Bayes-hálózatok. Ez a feltételes valószínűségek értékein, illetve ezek összefüggésein alapul. Noha vannak kisebb-nagyobb, ingyenes illetve fizetős programok is ennek a területnek az oktatására, az egységesítés érdekében érdemes lenne itt is Python nyelvű programokkal kezelni a számításokat. A tankönyv mellé kapott kódok ebben az esetben szegényes alapot jelentek. Szerencsére ezzel a problémával már mások is szembesültek, így más programkezdmények is elérhetőek [9], de a teljes megoldás még várat magára.

5. Összefoglalás

A Bolognai képzés bevezetésével, és új szakok indításával új kihívásokkal kerültünk szembe. Az eltelt idő alatt nyert tapasztalat elegendő muníciót ad ahhoz, hogy óvatos lépésekkel megreformáljuk a tananyagot, illetve a képzés során felhasznált – akár órán demonstrációra, akár otthon önálló felkészülésre szánt – segédletet. Az ágensek, valamint a keresési algoritmusok bemutatására szolgáló programokat már elkészítettük (átírtuk illetve

kibővítettük) Python nyelven, de a többi témakör programjai még nincsenek diákok számára bemutatható formában. Gyakran csak a már kész kód felülvizsgálatára, átírására van szükség, illetve bővítésére az általunk leggyakrabban használt példákkal, hogy az előadáson vagy a gyakorlaton bemutatott feladatot újra és újra le lehessen futtatni. A figyelemfelkeltés érdekében a játékprogramok készítésénél felhasznált módszereket, eszközöket kívánjuk bevetni. Az implementáció tervezett határideje 2014 vége, hogy a következő Mesterséges intelligencia kurzus esetén a teljes tananyaghoz egységes minőségben, és egységes formátumban adhassunk segédeszközöket a diákok, illetve igény esetén az oktatóik kezébe. Az elsődleges célcsoport természetesen a gazdaságinformatikus diákság, de miután az elkészült programok semmilyen módon nem épülnek ennek a szaknak a specialitásaira, más informatikus, vagy mesterséges intelligencia – vagy akár a programozás – iránt érdeklődő személy is eredményesen használhatja az elkészült és a közeljövőben elkészülő programokat.

6. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] S. Russel és P. Norvig, *Artificial Intelligence: A Modern Approach 3/E*, Cloth: Prentice Hall, 2010.
- [2] R. Stuart és P. Norvig, „Mesterséges Intelligencia,” 2013.
http://www.tankonyvtar.hu/hu/tartalom/tamop425/0026_mi_4_4.
- [3] A. Bradbury és B. Everard, *Learning Python with Raspberry Pi*, John Wiley & Sons, 2014.
- [4] S. McManus, A. Robinson és M. Cook, *Raspberry Pi Projects*, John Wiley & Sons, 2013.
- [5] C. Richardson, *Minecraft Pi Book*, CC BY-NC-SA 3.0, 2013.
- [6] „Artificial Intelligence: A Modern Approach,” Berkeley University, 2013.
<http://aima.cs.berkeley.edu>.
- [7] M. Kósa és Pánovics János, „Keresőalgoritmusok objektumorientált megközelítése a Mesterséges intelligencia tárgy bevezető kurzusán,” in *XVII. SzámOkt - VIII. ENELKO Nemzetközi Számítástechnika és Energetika-Elektrotechnika Konferencia*, Nagyvárad, 2007.
- [8] L. Aszalós és P. Battyányi, „Prolog feladatgyűjtemény,” 2011.
http://www.tankonyvtar.hu/hu/tartalom/tamop425/0046_prolog_feladatgyujtemeny/ch01s09.html.
- [9] K. Hauser, „Algorithms for Optimization and Learning,” Indiana University, 2012.
<http://homes.soic.indiana.edu/classes/spring2012/csci/b553-hauserk/bayesnet.py>.

A projekteszközök oktatásának didaktikai kérdései¹

Didactic questions of teaching project tools

Baráth Áron, Cséri Tamás, Horváth Gábor, Pataki Norbert, Porkoláb Zoltán, Simon Melinda, Szügyi Zalán

Eötvös Loránd Tudományegyetem, Informatika Kar,
Programozási Nyelvek és Fordítóprogramok Tanszék

{baratharon, cseri, xazax, patakino, gsd, melinda, lupin}@caesar.elte.hu

Absztrakt: Manapság egy jó szoftverfejlesztő szakembernek nem elég programozási nyelveket ismernie, hanem fontos a hatékony csapatmunkát elősegítő projekteszközök használatát is elsajátítania. A szoftverfejlesztés mindennapjaiban használunk verziókövető rendszereket (pl. SVN, Git), build eszközöket (CMake, Autotools, Maven stb.), fejlesztői környezeteket (pl. Visual Studio, Eclipse), dokumentáció generáló programokat (pl. Doxygen, Javadoc) és continuous integration rendszereket (pl. Jenkins). A felsőoktatás gyakran elmulasztja ezek tanítását, részben mert nehezen besorolhatóak a hagyományos tárgyak keretei közé, részben a kurrens ipari tapasztalatok hiányában. 5 évvel ezelőtt kezdtük el meghirdetni ezen eszközök oktatására az ELTE programtervező informatikus BSc szakon a Projekteszközök című tárgyat. A tárgyat jellemzően harmadéves hallgatók veszik fel. Jelen előadásban arra keressük a választ, hogy milyen nehézségekkel néztünk szembe ez alatt az 5 év alatt. Bemutatjuk, hogy mik a tapasztalataink a projekteszközök oktatásának tekintetében. Olyan kérdések merülnek fel, hogy mennyire legyen nyelvfüggő egy ilyen kurzus? Hány különböző eszközt mutassunk be egy-egy eszköztípus kapcsán? Milyen interfészt válasszunk, amikor több lehetőség is van? Sokszor szintaktikusan és szemantikusan is jelentős különbség van a különböző eszközök között. Hogyan lehet szoftverek használatát tanítani a programfejlesztéshez? Ezek egyszerű, könnyen megérthető eszközök vagy bonyolultan konfigurálható, nagy rendszerek? Tárgyaljuk, hogy a hallgatók mennyire veszik komolyan a tárgy témáját, milyen korábbi tapasztalatokkal érkeznek, mennyi újdonságot nyújtunk nekik és milyen mély ismereteket tudunk átadni egy-egy téma kapcsán?

Kulcsszavak: projekt eszközök, felsőoktatás, ELTE

Abstract: Nowadays the knowledge of programming languages is not enough for software developers, but they need to know the usage of project tools that are important from the view of efficient team work. Version control systems (for instance SVN, Git), build tools (Cmake, Autotools, Maven, etc.), Integrated Development Environments (for example Visual Studio, Eclipse), documentation generators (such as Doxygen or Javadoc) and continuous integration systems (like Jenkins) are widely-used in the everyday software development. In higher education the teaching of these tools is ignored very often. One of the possible reasons of this phenomenon is hard to classify it among the traditional subjects. Another important reason is due to the lack of current industrial experiences. We have started to teach Project tools course on BSc in computer science 5 years ago. The course is attended in the 6th semester. We present our experiences according to the teaching of this course, what kind of problems occurred. The following questions are raised. How language-specific this course should be? How many tools from a specific type should be introduced? Which interface of tools should be chosen, if more than one is available? There can be significant differences between these tools. How we may teach usage of software tools for software development? On the other hand, these tools are simple, easily understandable, on the other hand, the same tools require quite complex configuration with hundreds of plugins. We present the attitude of students and their former experiences. We analyze how deep knowledge may be passed to the students.

Keywords: project tools, higher education, ELTE

¹ Készült az Emberi Erőforrások Minisztériuma támogatásával a Kiemelt támogatási szerződés (18370-8/2013/TUDPOL) keretében.

Bevezetés

Egy jó szoftverfejlesztő embernek az általa használt programozási nyelven kívül számos ismeretre van szüksége, ahhoz hogy boldogulni tudjon a mai szoftverfejlesztő cégeknél. A forráskód egy verziókövető rendszerben van tárolva nyelvtől független módon. A fordítási folyamatok, tesztelesek automatizálva vannak. A dokumentációt a forráskódban lévő speciális kommentek alapján generálhatjuk, különböző kimeneti formátumban. Egy megfelelő szoftverrendszer az összes fejlesztő munkáját folyamatosan egybegyűjtve értékelheti ki, jelezheti az újonnan felmerülő problémákat, különböző lávalámpák mutatják a fejlesztett szoftver állapotát. Ezen eszközök általános használatát a felsőoktatásban is kellene tanítani, hogy a frissen kikerült hallgatók ne az első munkahelyén kelljen ezt megtanulnia, valamint ne kerüljön a felvételi interjúján hátrányos helyzetbe.

Sajnos ezek az eszközök projektenként, nyelvenként, cégenként rendkívül nagy változatosságot képesek bemutatni, így sokszor egy-egy témakör kapcsán nem elég egy eszközt ismertetni, például a build eszközök rendkívül különböző megközelítéssel tudnak dolgozni.

Többek között azért nehéz egy ilyen tárgynak bekerülnie a felsőoktatásba, mert eszköz használatot tanít, ráadásul sokszor nem is igazán bonyolult eszközökét. Egy dokumentáció generáló program, ami a forráskódban elhelyezett kommentek alapján elkészíti a dokumentációt, nem olyan bonyolult, mint egy programozási nyelv vagy a formális nyelvek elmélete. Ez sokszor oktatóként is nehéz feladat: hogyan tartsunk fent a hallgatóság érdeklődését, egyáltalán hogyan lehet egy 90 perces órát megtartani az adott témakörben. A hallgatóság is könnyen elveszíti a motivációját ilyen esetben. Ezt rendszeresen vissza is csatolják a hallgatók a félév végén. Ennek ellentmond, hogy ezen szoftver némelyike viszont nagyon bonyolultan paraméterezhető, használható, például a Maven-hez több száz plugin használható, amit a szoftverfejlesztők sem tartanak fejben, hanem internet segítségével nyomozzák ki a létezését és használatát. A Jenkins rendszerhez is több, mint 600 plugin használható, ami szintén nem megtanítható tananyag.

A tanított eszközök egy része programozási nyelvtől függetlenül tanítható (például a verziókövetés), másik részük viszont nagyon is nyelvfüggő (pl. tesztelési keretrendszerek). A tárgyat felvett hallgatók általában a Java-t preferálják, a C++ tudásuk általában gyengébb. Szükségszerűen változtatjuk a Java-hoz, illetve C++-hoz kapcsolható eszközök bemutatását, de ez nehezíti a hallgatók dolgát, mert ők nehezebben váltanak nyelvet. A nyelv mellett felmerül az operációs rendszerek kérdése is, mi általában Linux-ot választjuk, így a Linux-on is elérhető eszközöket mutatjuk be. A félév elején a hallgatók betekintést nyernek a shell környezetbe és a reguláris kifejezésekbe.

Sajnos ezek a projekteszközök gyorsan változnak, nagyságrendekkel gyorsabban, mint a programozási nyelvek. Például a Java-hoz használt build rendszerek rendszeresen cserélődnek: az Apache Ant imperatív leírásáról áttértek az Apache Maven deklaratív megközelítésére, amit ma már sok helyen váltanak a Groovy-alapú Gradle-re. Könnyen lehet elavult tudást átadni a hallgatóknak, ezért folyamatosan figyelni kell az ipari trendeket.

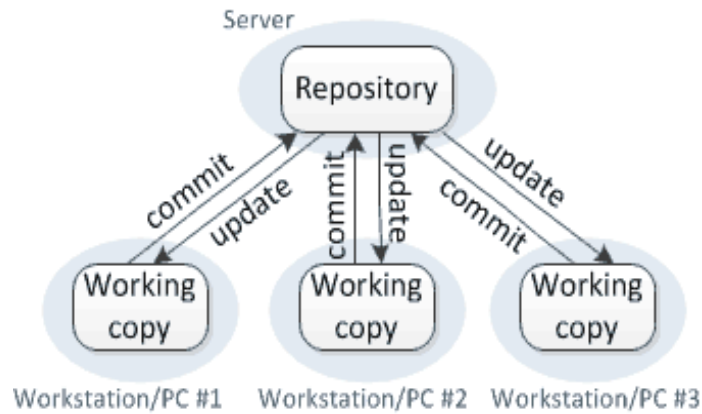
Ebben a cikkben végignézzük azokat az eszköztípusokat, amelyeket a tárgyunk keretein belül bemutatásra kerülnek. Átnézzük a konkrét eszközöket és azok oktatási vonatkozásait. Tárgyaljuk az oktatási problémákat, bemutatjuk a mi megközelítésünket.

Az oktatott eszközök

Ebben a fejezetben átnézzük az eszköztípusonként a tárgyunkhoz kapcsolódó problémákat, illetve az általunk javasolt megoldásokat.

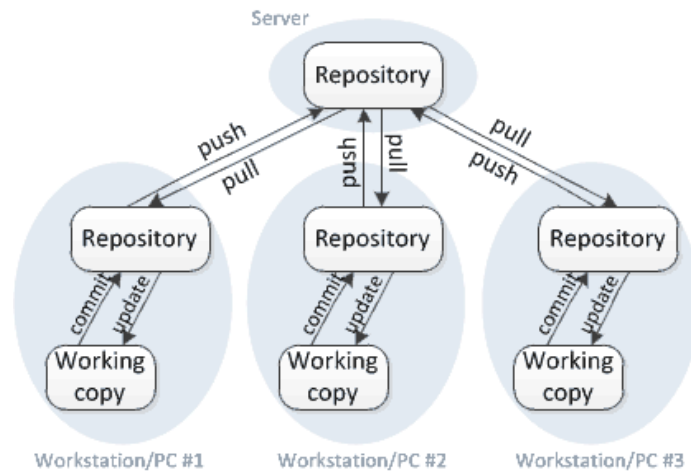
A verziókövetést el lehet magyarázni programozási nyelvtől független módon. Általában az *SVN* és *Git* rendszereket tanítjuk, mert ezek a legelterjedtebbek, valamint ezek érdemben különböznek [4,8]. Az *SVN* esetén egy központi repository létezik, minden felhasználó ez alapján frissíti a lokális másolatát, valamint ide küldi be a változtatásait (lásd 1. ábra). A *Git* ezzel szemben elosztott repository-val rendelkezik, minden fejlesztőnek van saját, önálló repository-ja, ahova beküldi a változtatásait (lásd 2. ábra). A repository-kat push és pull request-ekkel lehet szinkronizálni.

Centralized version control



1. ábra. Az SVN központi repository modellje

Distributed version control



2. ábra. A Git elosztott repository modellje

A verziókövető rendszerek esetén fontos kérdés, hogy milyen interfészen keresztül használjuk, mert több különböző megközelítés lehetséges. Használhatjuk parancssori eszközökkel, grafikus felületű önálló szoftverekkel (pl. *TortoiseSVN*, *RapidSVN*), valamint fejlesztői környezetekben plugin-okon keresztül. Ezek közül mindegyiket szokták használni, viszonylag gyorsan lehet váltani az interfészek között, ha a programozó tisztában van a verziókövető fogalmaival. Mi technikai okokból a parancssori klienst ismertetjük, de mivel ez az alapvető megközelítés, később nem okozhat gondot a grafikus interfész alkalmazása sem.

Az automatizált fordításhoz használt build eszközökben vannak a legnagyobb különbségek. Ezek az eszközök már szorosabban kapcsolódnak programozási nyelvekhez, mint a verziókövető rendszerek. A jellemzően C++-hoz közelebb álló rendszerek közül általában a *make-et* (és modernebb utódait) szoktuk tanítani, de volt már példa a Linux-alapú rendszerek *Autotools* eszközkészletének oktatására is. A Java nyelv vonalán kezdetben *Apache Ant* eszközt ismertettük. Ezután bekerült az *Apache Maven* is a tárgy oktatásába, jellemzően koncepcionális oldalról megközelítve. A Maven script-ek karbantarthatósága kérdéses, nehezen írható file-ok, aminek az elméleti alapjait fontos átlátni. A gyakorlati megközelítése a Maven script-eknek nem tanulható igazán a használható plugin-ok számossága miatt. Az alábbi minimális példa bemutatja, hogy miért nehéz a terjedős, deklaratív xml gyakorlati oktatása:

```
<?xml version="1.0"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/maven-v4_0_0.xsd">
<modelVersion>4.0.0</modelVersion>
<groupId>org.foo</groupId>
<artifactId>my-app</artifactId>
<packaging>jar</packaging>
<version>1.0-SNAPSHOT</version>
<name>my-app</name>
<url>http://maven.apache.org</url>
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-eclipse-plugin</artifactId>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <configuration>
        <archive>
          <manifest>
            <addClasspath>>true</addClasspath>
            <mainClass>org.foo.App</mainClass>
          </manifest>
        </archive>
      </configuration>
    </plugin>
  </plugins>
</build>
```

```
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-surefire-plugin</artifactId>
    </plugin>
</plugins>
</build>
<dependencies>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>3.8.1</version>
        <scope>test</scope>
    </dependency>
</dependencies>
</project>
```

Az Ant eszköz kikerült az oktatásból mivel ipari alkalmazása háttérbe szorult. Jelenleg feltörekvőben van a Gradle rendszer is, jelenleg nem tanítjuk, de elképzelhető, hogy hamarosan szükség lesz ennek az eszköznek az oktatására is [1].

Az egység- és integrációs tesztelés eszközeit könnyebben lehet bemutatni, hiszen a tesztelési keretrendszerek inkább szintaktikusan különböznek: az assertion-öket más formátumban kell leírni. Ebben a témakörben jellemzően a tesztelés elméleti alapjait lehet lefektetni: például a különböző tesztelési megközelítéseket: egység-, integrációs, regressziós, rendszer-, teljesítmény, stb. teszteléseket megkülönböztetni, a kód lefedettségének vizsgálatát bemutatni és a tesztelés gyakorlati tudnivalóit ismertetni [5,7]. A Java esetében ipari de facto szabvány keretrendszer a *JUnit*, ezt ismertetjük [8]. Azonban a C++ nyelv esetében nincs ilyen általánosan elfogadott eszköz. Számos lehetőség közül lehet választani: *GoogleTest*, *Boost Test*, *CxxTest*, *UnitTest++*, *CppUnit* stb. A mi választásunk a *GoogleTest*-re esett.

Fejlesztői környezetek közül az *Eclipse-et* szoktuk ismertetni, bár a hallgatóknak korábbi tanulmányaikból már van tapasztalatuk ezzel az IDE-vel. Ebben a témakörben az a bonyolult kérdés, hogy ki mit szokott kihasználni az Eclipse funkcionalitásából, kinek mire lesz szüksége, milyen háttérismeret szükséges a környezet használatához [3]. Ezekre a kérdésekre a válasz sokszor személyiség- és projektfüggő. Probléma továbbá, hogy a menürendszerben megtalálható funkciókat mutatjuk be, ami sokszor csak átlag alatti képességű hallgatóknak jelent újdonságot. A hibakeresés és debug-golás általában az Eclipse témakörben hangzik el.

A dokumentációgenerálás működését két eszközön mutatjuk be: a *Doxygen* eszközt, amely több különböző programozási nyelvvel képest együttműködni (pl. C++, C#, Fortran), valamint a Java fejlesztési környezetben használt de facto szabvány *Javadoc-ot*. A hallgatóknak ezzel kapcsolatban már vannak tapasztalataik, valamint a téma mélysége is sekély, de a tárgy keretein belül szükséges beszélni erről a témakörrel is.

Jelenleg nem a tananyag része a continuous integration eszközök bemutatása [2]. A probléma az, hogy a continuous integration eszközök használata is erősen feladat-, projekt- és csapatfüggő. Kis csapatoknál nem is feltétlenül tud érvényesülni. Itt is csak koncepcionális oktatást lehetne alkalmazni, mert egy ilyen rendszer mindig konfigurációs paraméterek kérdése.

Nem képezi a törzsanyag részét a profiler-ek, statikus elemzők, memóriaszivárgás detektorok használata, de amikor belefér a félév tananyagába, akkor ezek is szóba kerülnek.

Összefoglalás

Az iparban használt projekt eszközök tanítása fontos feladata lenne a felsőoktatásnak, de ez sokszor nem teljesül különböző okokból kifolyólag. Mi elindítottunk egy ilyen kurzust az ELTE Informatikai Karán. A tárgy keretein belül tanítunk verziókövető rendszereket, build eszközöket, tesztelési keretrendszereket, dokumentáció generáló szoftvereket. A cikkben bemutattuk, hogy milyen konkrét eszközök oktatását végezzük és mi ennek az oka. Részleteztük, hogy milyen problémák merültek fel az elmúlt években a tárgy tanításának során, valamint példákat mutattunk a hallgatói visszajelzésekre.

Köszönetnyilvánítás

Ezúton szeretnénk megköszönni azoknak a hallgatóknak a visszajelzését, akik a félév végén korrekten kitöltötték az anonim kérdőíveket a tanulmányi rendszerben, ezzel elősegítve, hogy a tárgyunk jobbra válhasson, és mi is tanulhassunk az elkövetett hibáinkból.

Irodalomjegyzék

- [1] Tim Berglund, Matthew McCollough: Building and Testing with Gradle, First Edition, O'Reilly Media, 2011.
- [2] Jon Bowyer, Janet Hughes: Assessing Undergraduate Experience of Continuous Integration and Test-Driven Development, Proceedings of the 28th International Conference on Software Engineering (ICSE 2006), pp. 691-694.
- [3] Zhixiong Chen, Delia Marx: Experiences with Eclipse IDE in Programming Courses, Journal of Computing Sciences in Colleges 21(2), pp. 104-112.
- [4] Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato: Version Control with Subversion, O'Reilly Media, 2004.
- [5] Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black: A szoftvertesztelés alapjai, ALVICOM Kft., 2010.
- [6] John Loelinger, Matthew McCollough: Version Control with Git, Second Edition, O'Reilly Media, 2013.
- [7] Joan C. Miller, Clifford J. Maloney: Systematic mistake analysis of digital computer programs, Communications of the ACM 6(2), pp. 58-63.
- [8] Petar Tahchiev, Felipe Leme, Vincent Massol, Gary Gregory: JUnit in Action, Second Edition, Manning Publications Co., 2010.

Agilis Szoftverfejlesztés Agilis Oktatása

Agile Education of Agile Software Development

Bujáki Attila^a, Dr. Balla Katalin^b,

^{a, b} Budapesti Műszaki és Gazdaságtudományi Egyetem
Irányítástechnika és Informatika Tanszék

^a bujaki.attila@sqi.hu

^b balla@iit.bme.hu

Absztrakt: Agilisnak lenni manapság jóformán kötelező az élet szinte minden területén. Aki nem eléggé mozgékony, gyors, élelmes, alkalmazkodó – az valószínűleg kevésbé is lesz sikeres a mai világban. Az „agilitás” fogalmának definícióját keresve az informatikusok büszkék lehetnek: ismét az ő szakterületük vált iránymutatóvá, hiszen agilis munkavégzés iránti igény a szoftverfejlesztéssel kapcsolatban fogalmazódott meg! Szoftverfejlesztési projekteket próbáltak meg legelőször agilisán végrehajtani, s a sikert követően kiderült, hogy más diszciplínákban is alkalmazható az agilis szoftverfejlesztésben szerzett tapasztalat. Mára az agilis fejlesztés, tervezés, kódolás és tesztelés mellett önálló tudományterülete alakult ki az agilis projektmenedzsmentnek, az agilis csapatoknak, az agilis környezetnek.

Cikkünk első részében az agilis szoftverfejlesztés rövid bemutatására kerül sor; megfogalmazzuk, hogy milyen jellemzők különböztetnek meg egy agilis szoftverfejlesztési életciklust, szervezetet és projektet a hagyományosaktól. Kiemeljük, hogy szerintünk milyen képességekre, tulajdonságokra van fokozott mértékben szükségük az ilyen csapatban dolgozó kollégáknak.

A második részben az agilis elemek oktatásban való felhasználásáról lesz szó; ha ugyanis mi szoftverfejlesztőket oktatunk agilis munkára, akkor kézenfekvő, hogy oktatásunk is agilis legyen! Az agilis oktatás jegyeire és hatékonyságára vonatkozóan gyakorlati oktatási tapasztalatainkat szeretnénk megosztani. Bemutatunk néhány eszközt és technikát, amelyeket a BME IIT-n, valamint a Simonyi Károly Szakkollégiumban és a TTNY oktatólabor tanfolyamain jól hasznosíthatónak találtunk az újfajta szoftverfejlesztési megközelítés elsajátításának támogatására. A cikk végén a BME Pszichológia Intézetével tervezett kutatásunkról lesz szó, amelynek célja az agilis szoftverfejlesztéshez szükséges emberi készségek, képességek és személyiségjegyek vizsgálata.

Kulcsszavak: agilis szoftverfejlesztés, agilis projektmenedzsment, agilis munkavégzés, agilis oktatás

Abstract: Being agile these days is getting more and more a necessity on every different field of life. Those who are not quick enough, not able to adapt fast – those are more likely to fail in the business world. Looking for the real definition of „Agility” – professionals of the IT sector can be really proud of their results, since their area is the flagship, the leader of the agile revolution. They were the first ones able to define, use and profit from agility in practice. Different software development projects were the first ones to be done in an agile manner, and after their success it became clear: agility and the connected, vast amount of gathered experience could be adopted to, and used in other disciplines as well.

These days agile development, design, implementation, coding and testing are becoming separate art and discipline on their own, next to agile project management, agile teams and agile environments which are already widely accepted.

In the first part of our article we share some ideas about agile software development in general; we define what differentiates an agile software development lifecycle, organization and project from a traditional one. We would like to highlight what skills and characteristics are needed for effective work in such team, in our opinion.

In the second part, we present our ideas about using agility in education. This is needed, because if we would like to prepare agile developers and teach students about agile working, then it is really essential to do this in an

agile manner, using agile education practices. We would like to share our practical experiences about the topic, and present some special properties agile education might have, highlighting its efficiency. Some of our previously used techniques and tools are also going to be presented; they have been found to be extremely useful during workshops held at the BME IIT department, at the Simonyi Károly College for Advanced Studies, and at the TTNy training lab, where we used them to present this new, agile approach in Software Development.

In the final part of our article, we present our planned research in collaboration with the BME's Institute of Psychology, which aims to investigate human personal characteristics that are – according to our expectations – enabling the most effective working in agile teams.

Keywords: agile software development, agile project management, agile working, agile education

1. Bevezetés

A szoftverfejlesztésben, ahogy egyre komplexebb rendszerek fejlesztése iránti igények merültek fel, óhatatlanul szükségessé vált különféle módszertanok kidolgozása, amelyek tulajdonképpen bevált jó gyakorlatok felismeréséből indultak ki, ezek széles körben való alkalmazását propagálták, a csoportos munkavégzés megkönnyítésére. Az egyre összetettebb feladatok megvalósítására a korábbiaknál nagyobb csapatokra és bonyolultabb szervező tevékenységre volt szükség, így a szoftverfejlesztési módszertanok is egyre összetettebbek lettek (kiegészültek, például, projektirányítási elemekkel is).

Ám hiába volt minden igyekezet, a nagyméretű, komplex szoftverprojektek fogalmával összefonódott a sikertelenség, késedelem és a kudarc élménye: ha el is készült a termék, gyakran nem nyújtotta a várt funkcionalitást vagy egyszerűen soha nem is került gyakorlati alkalmazásra. A korábban alkalmazott módszerek nem működtek jól az új környezetben. [1] Kezdetét vette a szoftverkrízis. Dijkstra szavait idézve: „A szoftverkrízis fő oka, hogy a gépek jó néhány nagyságrenddel nagyobb teljesítményűek lettek. Egyszerűen szólva, amíg nem voltak gépeink, a programozás nem okozott problémát, aztán amikor gyenge számítógépekkel rendelkezünk, a programozásuk csak egy kellemes, még kezelhető problémát okozott, most pedig, hogy gigantikus szuperszámítógépeink vannak, a programozásuk is hasonlóan gigászi problémává nőtte ki magát” [2].

Véleményünk szerint, a fenti idézet mára, az informatika és a számítógépek soha nem látott méretű térnyerésével hatványozottan igaz. Később ugyan megtanultunk együtt élni a szoftverkrízis jelenségével, problémáival; annyi történt, hogy a különféle módszertanok, praktikák, eszközök elviselhetőbbé tették a bizonytalanságot, ami a nagyméretű projektekkel járt, de valódi, végső megoldást nem kaptunk.

Egyre inkább el kellett fogadnunk, hogy a szoftverfejlesztés problémáit nem tudjuk megoldani, hanem meg kell tanulnunk együtt élni a jelenséggel. Vagyis: nem arra kell törekednünk, hogy problémamentes – mondjuk, tökéletesen, „egyszer és mindenkorra” specifikált, (sosem változó specifikációjú!) – rendszert fejlesszünk; be kell látnunk, hogy ez nem valós elvárás. Ehelyett, a szoftverfejlesztőknek tudomásul kell venni, hogy a specifikáció változhat, és meg kell tanulni - ezzel együtt - szoftvert fejleszteni, használható rendszereket átadni. Mi sem támasztja ezt jobban alá, mint a The Standish Group által nem régiben publikált „The Chaos Manifesto” kiadványa [3], amely az IT és szoftverprojektek sikerességét, eredményességét vizsgálta – illetve az ezekkel párhuzamba állítható, legfontosabb tényezők felismerését tűzte ki célul.

A tanulmány szerint a modern szoftverprojekteknek is csak a 39%-a mondható sikeresnek, a fennmaradó 61%-ból pedig: 48% valamilyen szempontból problémás volt; késett vagy pedig jelentősen túllépte a tervezett költségkeretet. A projektek 18%-a pedig egyenesen csőd volt; vagy sohasem készült el, vagy ha el is készült a projekt és a tervezett termék, akkor az eredmény soha nem került használatba.

Érdekes eredmény, hogy a tanulmány által kiemelt, egyik fontos, a vizsgált projektek alapján meghatározott sikertényezőt – az agilis folyamatok jelentették. Ennek alapján is kijelenthetjük, hogy a rohanó világunkban a szoftver és IT projektek területén is rendkívül fontos, hogy lépést tudjunk tartani a világgal, a versenytársainkkal; ehhez pedig mindenképp szükséges feltétel, hogy kellőképpen agilisak legyünk.

A következő fejezetben röviden bemutatásra kerül az agilis szoftverfejlesztés, valamint, hogy mi tesz egy szervezetet agilissá.

2. Az agilis szoftverfejlesztés jellemzőinek rövid bemutatása

Agilisnak lenni, annyit tesz, hogy mozgékonyak, ügyesnek, gyorsnak, életrevalónak lenni.[4] De vajon ezt hogyan értelmezhetjük a szoftverfejlesztésre és a különféle szervezetekre nézve? Rendkívül divatos kifejezésről és irányzatról van szó, azt viszont nagyon fontosnak tartjuk, hogy minden esetben valóban tisztába legyünk azzal, hogy mi az irányzat valós jelentősége és célja. Ne engedjük, hogy a divat diktáljon, így fontos, hogy valóban megismerkedjünk a különféle eszközökkel, módszertanokkal – azok lényegével és céljaival – hogy amikor a döntésre kerül a sor, valóban a megfelelőt alkalmazhassuk, az adott helyzetben leginkább hasznosat választhassuk ki a lehetőségek garmadájából. Csak a különféle agilis szoftverfejlesztési módszertanokat, technikákat tekintve is számos lehetőségünk van.

Ennek megfelelően, az „agilis szoftverfejlesztés” valójában számos különféle módszertan, eszköz, technika és elv közös, átfogó elnevezése. Ez számos szempontból eltér a hagyományos szoftverfejlesztési módszertanoktól, technikáktól. Nem minden esetben alkalmazható jól, és nem való mindenkinek. Más jellegű munkavégzést vár el a csapattagoktól, amelybe nem feltétlen illeszkedik be minden munkatárs könnyedén. Ezzel együtt, sok esetben alapmegoldásnak, mintegy „kötelezőnek” tekintik az agilis szoftverfejlesztést; tapasztalatunk szerint ennek az az oka, hogy az agilitást csak felületesen ismerők összekeverik az agilis munkát a módszertan nélküli, „ad-hoc” szoftverfejlesztéssel.

Mára komoly mennyiségű tapasztalat áll rendelkezésre a területen, a különféle módszerek és technikák alkalmazását tekintve számos esettanulmány olvasható, és széles kínálata érhető el a különféle „agilis” tanfolyamoknak, képzéseknek is a piacon.

2.1 Az ügyfél szerepe

Jelentős különbség a hagyományos módszertanokkal szemben, hogy az agilis szoftverfejlesztés sokkal inkább ügyfélközpontú. A fejlesztés minden fázisában ez kell, hogy a szemünk előtt lebegjen: „Hogyan tegyük elégedetté, boldoggá az ügyfelet?”.

Vajon minek örülne jobban? Ha több hónap munkavégzés után előállnánk egy több száz oldalas követelményspecifikációval, vagy ha már röviddel a kezdeti egyeztetést követően működő, tesztelt szoftvert mutatnánk be, vagy akár szállítanánk is le? Képzeljük magunkat csak az ügyfél helyébe. Mikor lennénk igazán meggyőzve arról, hogy a megfelelő csapat kezébe adtuk a projektet?

Az agilis szoftverfejlesztésben az elsődleges prioritásunk az ügyfél igényeinek a kielégítése, úgy hogy az életciklus korai szakaszától kezdve, folytonosan szállítjuk a számára már értékes, és a gyakorlatban is alkalmazható üzleti funkcionalitást megvalósító szoftvert.

Ezt úgy tehetjük meg, hogy valamilyen iteratív és inkrementális megközelítést alkalmazunk. Ehhez a rendszer elkészítéséhez elvégzendő feladatokat megfelelően kicsi részekre kell felbontani, mivel az idő tipikusan nagyon rövid; 1-2 hetes iterációkkal dolgozva csak a kisméretű feladatok elvégzésére van időnk. Lényeges az is, hogy hogyan történik a felbontás.

A felbontást „Just In Time” („éppen időben” - a szükségesnél se nem korábban, se nem később!) végezzük. A megoldandó feladatok pontos részleteinek kidolgozására csak akkor kerül sor, amikor már valóban szükség lesz rájuk, mivel az elkövetkező iterációban megvalósításra kerülnek. Ha ezt az elgondolást követjük, akkor így elkerülhetjük a szoftverfejlesztés egyik legnagyobb pazarlását (waste), a túl korai részletes tervezést, így jelentős erőforrásokat takaríthatunk meg. Miért is végeznénk el az egész termék részletes tervezését, ha esetleg az adott funkciók végül nem is kerülnek bele az elkészült végtermékbe? Lehetséges, hogy egészen más irányba halad majd a termék fejlesztése, amely irányt mindenkor a megrendelő igényei és preferenciái diktálják.

2.2 Agilis tervezés

Az agilis szoftverfejlesztésnél arra koncentrálnunk, ami „igazán fontos”. (Mi lesz az ügyfél számára a legértékesebb?) Ténylegesen azt szállítsuk, ami értéket képvisel a megrendelőnek, valóban hasznos üzleti funkcionalitást adjunk át, és azon dolgozzunk mindenkor, ami számára épp a legértékesebb. Ennek megfelelően lehetőséget biztosítunk a számára, hogy a preferenciáit módosítsa iterációról iterációra, és így befolyásolja, hogy a fejlesztés milyen irányba zajlik majd a projekt folyamán.

Tipikus tévhit, hogy az agilis szoftverfejlesztés során nem készül dokumentáció. Dokumentációra igenis szükség van, de csak azért, hogy szoftvert fejleszthessünk a segítségével. Az öncélú dokumentációkészítés viszont pazarlás. Nem szabad szem elől tévesztenünk az üzleti célkitűzésünket.

Ugyanígy, miért is dolgoznánk azokon, és miért szállítanánk le a megrendelőnek olyan funkcionalitást, ami a jelen helyzetben már nem képvisel számára értéket? Elképzelhető, hogy a projekt elején még fontosnak tartotta az adott funkcionalitást, de útközben kiderült, hogy arra még sincs akkora szüksége, viszont valami új, más dolog nagyon fontos lenne. Hagyományos esetben a követelmények befagyasztásával ez a játszma eldőlt és a szerződés aláírását követően módosításra nem, vagy csak csillagászati összegekért van lehetőség. Az agilis esetben viszont ezt megengedjük, sőt, bátorítjuk az ügyfelet, hogy a mindenkori preferenciának megfelelően dolgozhassunk, támogatjuk a változtatásokat, amennyiben így számára értékesebb funkcionalitást, több értéket szállíthatunk. Elsődleges célunk az ügyfél elégedettségének biztosítása, így ennek érdekében megteszünk minden tőlünk telhetőt.

Látható, hogy komoly szemléletváltást igényel az agilis szoftverfejlesztés bevezetése. A hagyományos projektháromszög megváltozik. A fix funkcionalitás helyett, a minőség és az időkeret kerül rögzítésre. Agilis esetben fix időkereten belül dolgozunk, például a projektet fél év hosszúra tervezzük, ezen belül a fejlesztés során a megrendelő mindenkori igényeinek megfelelő funkcionalitást szállítjuk, az elvárt minőségben. Amennyiben esetlegesen túlvállaltuk magunkat, akkor sem szállítunk a vártnál alacsonyabb minőségben, hanem az alacsonyabb prioritású funkcionalitások közül esetleg kivesszünk néhányat az átadandók közül.

2.3 Agilis fejlesztés és tesztelés

Nagyon fontos, hogy a minőségből nem engedünk. Nem megfelelően tesztelt, ki nem próbált, félkész funkciót nem adunk át a megrendelőnek. Ehhez rendkívül sokat kell tesztelni. Agilis szoftverfejlesztésnél sok tesztet írunk, ennek megfelelően ezeket automatizáljuk is, hogy le tudjuk őket futtatni szükség esetén újra és újra. A tesztek készítésének az elsődleges célja nem a 100%-os, vagy 80%-os, vagy hasonló konkrét értékű lefedettség kell, hogy legyen, hanem a tesztek fognak minket támogatni abban, hogy magabiztosan állíthassuk: igen, a rendszer valóban úgy működik, ahogy terveztük és ígértük a megrendelőnek. Mindig biztosítjuk azt is,

hogy a korábban átadott funkcionalitások közül is minden hibátlanul működik (regressziós tesztelés).

A fejlesztési folyamat során rendszeresen visszajelzést kérünk. Az ügyfelet szorosán bevonjuk a fejlesztési folyamatba is, nem csak a kezdeti követelmény meghatározásba, de a teljes ciklus során. A megrendelő lesz a segítségünkre, hogy biztosak lehessünk: jó az irány, amerre a projekt halad.

2.4 Agilis alapelvek, agilis elemek összefoglalása

Fontos ismét kiemelni, hogy ehhez az ügyfél magas fokú közreműködése is szükségeltetik, azaz megfelelő elkötelezettségre van szükség a megrendelő oldalán is. Az agilis szoftverfejlesztésnél alapvető feltételezés, hogy az ügyfél valamilyen formában elérhető a fejlesztési folyamat során. Legalább online, vagy telefonon naponta adott órákban, de lehetőleg személyesen válaszoljon a felmerülő kérdésekre, hogy valóban azt a terméket készíthessük el, ami leginkább szolgálni fogja az ő üzleti igényeit. Egyfajta órakulcs szerepét tölti be a fejlesztési folyamat során, a felmerülő, követelményekkel, elvárásokkal, preferenciákkal kapcsolatos kérdéseinkre ad választ. Szeretnénk az ügyfelet bevonni a folyamatba, ahelyett, hogy egy előre rögzített szerződés alapján dolgoznánk. A megrendelő magas fokú bevonását favorizáljuk a szerződéses egyeztetéssel szemben.

Az agilis szoftverfejlesztésben is tervezünk (tipikus tévhit, hogy nem készülnek tervek). Készülnek, de ezeket is azért készítjük, hogy segítsenek minket a szoftverfejlesztési munkánkban. Rugalmasabban kezeljük a terveket; amennyiben valami változott, úgy merjük a terven is változtatni. Ne hagyjuk, hogy egy nem aktuális, irreális terv akadályozzon a munkában. A változások elfogadását favorizáljuk a tervek szolgai követése helyett.

Fontos, hogy reális terveket készítsünk. Az agilis szoftverfejlesztéstől se várjunk csodát, még az legideálisabb esetben se. Ez is csak egy hasznos eszközkészlet, ami segítségünkre lehet, hogy sikeres termékfejlesztést végezhessünk, amely bizonyos esetekben már bebizonyította, hogy rendkívül hasznos lehet, ha megfelelően alkalmazzuk.

2001. februárjában került sor az Egyesült Államokbeli Utah államban az Agilis Kiáltvány, az Agilis Manifesztó megfogalmazására[5], amelyben összefoglalásra kerültek a legfontosabb alapelvek és preferenciák, amik az új módszereket leginkább áthatották és meghatározták.



1. ábra - Az Agilis Kiáltvány

A Kiáltványon kívül még 12 alapelv[6] került meghatározásra, amelyek az új technikák és gyakorlatok közös filozófiáját összefoglalták.

A Kiáltvány és az alapelvek egyszerű iránymutatásként szolgálnak, ha szeretnénk megérteni az agilis szoftverfejlesztés filozófiáját.

Az agilis szoftverfejlesztés már sok különféle környezetben jól bizonyított. Számos jól bevált technika és módszertan épül ezekre az alapelvekre, de ismét kiemelnénk: fontos, hogy ne engedjünk a kísértésnek. Ne a divat diktáljon, mindig azt a munkamódszert válasszuk, ami épp az adott helyzetben, adott környezetben leginkább előnyösnek és hasznosnak mutatkozik.

3. Mi szükséges az agilis munkához?

3.1 Környezet, csapat

Véleményünk szerint, ez a fajta – agilis – munkavégzés nem való mindenkinek. Jellemzően egy más jellegű környezetet és munkavégzést tételez fel, mint amilyenek a „hagyományos” szoftverfejlesztő csapat működését elképzeljük. Tipikusan kreatívabb, kommunikatívabb munkavégzést jelent a csapattagok számára, akik kereszt-funkcionális csapatokban végzik a munkájukat, amely ennek köszönhetően jellemzően kevésbé monoton.

Ezzel együtt viszont több felelősség jár. Tipikusan (többet) kell gondolkodni, több kreatív megoldást várnak a csapattagoktól, kevesebb dolgot kapnak készen, és aktívan részt vesznek számos jelentős döntés meghozatalában és kimunkálásában. Fontos tényező a megfelelő motiváció megléte is, mivel a csapattagok maguk határozzák meg a csapat számára a vállalásokat az iterációkra, így fontos, hogy valóban dolgozni akarjanak, és valóban

igyekezzenek az ügyfél elégedettségét biztosítani. Emiatt magasabb fokú önfegyelem és valódi csapatmunka szükséges az agilis projektekben.

A felelősség kérdésköre is egészen másképp alakul az agilis csapatoknál, mivel tipikusan nincs *egy* projektmenedzser, aki felelős lenne a projektért, hanem a csapat egésze felel a projekt sikeréért. Ennek megfelelően a csapatot a megfelelő hatáskörrel kell felruházni és valóban meg kell bízni bennük. Hinni kell bennük, hogy valóban végrehajtják a rájuk bízott feladatot. Itt ismételt felmerül a motiváció és a bizalom jelentősége.

Bevált gyakorlat, hogy az iterációk végén valóban maga „a csapat” mutassa be az újonnan elkészült funkcionalitást a megrendelőnek. Ennek számos előnye van. Talán a legfontosabb, hogy a sikertelenség esetén, a bőrükön érezhetik a csapattagok a kudarcot. Ha egy csapattal előfordult már, hogy az iteráció végén, a tervezettel ellentétben semmi konkrét eredményt nem tudott bemutatni, az egy meglehetősen mély nyomot hagy a csapattagokban, és vélhetőleg nem szeretnék ezt újra átélni. (Hasonlóan, a sikeres munkavégzés esetén erős a pozitív hatás, hiszen valóban a projektcsapat kerül a dicsfénybe, és nem más viszi el a „pálmát”. Ez rendkívül motiváló lehet a csapat számára, és egy egészen újfajta sikerélményt ad a fejlesztő csapat számára, ami megsokszorozhatja a hatékonyságukat.)

További komoly hozománya ennek a megoldásnak, hogy a csapattagok valóban megtapasztalhatják, hogy a megrendelői oldalon is „hús-vér” emberek vannak, akik számítanak rájuk, és valódi üzleti igényeik vannak, amelyeket a készülő termékkel szeretnének megoldani.

Ez a személyes találkozás sokkal közelebb hozza a két felet, és valóban érezteti a csapattal, hogy valós emberek számítanak rájuk, és bíznak a munkájukban.

3.2 Agilis szerződések

A szerződéseket tekintve is egész más megközelítést kíván az agilis munkavégzés. A hagyományos esetben tipikusan a követelményeket meghatározó dokumentum szolgált a szerződés alapjául. Az abban meghatározott funkciók szállítása a meghatározott határidőre volt a kitűzött feltétel. Az agilis esetben ez máshogyan alakul. Ennél rugalmasabb megoldásra van szükség, mivel a projekt során szállított funkciók pontos meghatározására csak a projekt során kerül sor. A megoldást speciális szerződések kötésével biztosíthatjuk. Az agilis szerződések világába az „Agile Contracts Primer” dokumentum egy rendkívül átfogó, a problémakört több oldalról is bemutató leírás.[7] Fontos kiemelni, hogy egy sikeres szerződéshez, ami nagyban hozzájárulhat egy agilis projekt sikeréhez, nagyon fontos, hogy a szerződést előkészítő jogi szakértők is megfelelően megértsék az agilis szoftverfejlesztés alapelveit, annak célkitűzéseit. Nem szabad figyelmen kívül hagynunk a nagyméretű szoftverprojekteket jellemző nagyfokú bizonytalanságot, márpedig pontosan ez a jellemző a hagyományos szerződésekre és projektekre, hiszen megpróbálnak lehetőleg mindent rögzíteni előre.

Egy megfelelő szerződés nagyban támogatja a sikeres kollaborációt a megrendelő és a szállító között, biztosítja az átláthatóságot, és megalapozza a megfelelő bizalom kialakulásának lehetőségét. Egy jó szerződés még kevés a sikerhez, de egy jó kiindulási alapot adhat a projekt sikerének, illetve a későbbi sikeres együttműködésnek. Fontos, hogy biztosítsa az ügyfél megfelelő bevonását, részvételét és az elkötelezettségét a projekt sikerét illetően. Amennyiben az ügyfél nem kellőképpen elkötelezett a projekt sikere iránt, úgy a csapat sem fog tudni megfelelően dolgozni, hiszen hiányozni fog a szükséges segítség a projekt céljainak pontos meghatározásában.

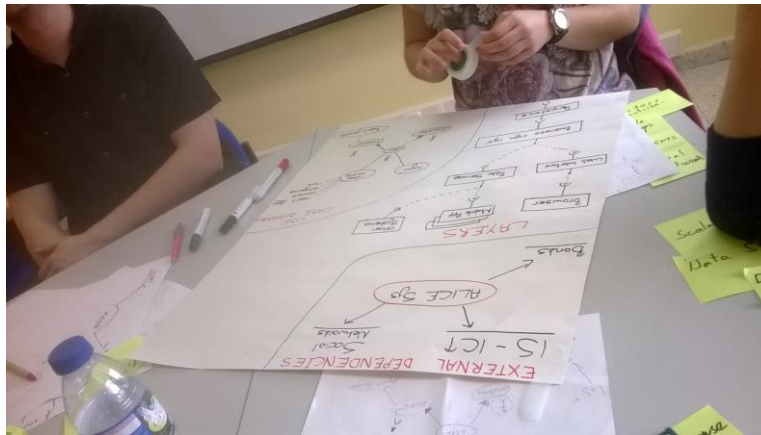
Hagyományosan a nagy szoftverprojekteket (a hagyományos szerződésekben), úgy kezelték, mint a nagy építkezési projekteket. Hosszú ciklusidők, és a késői visszajelzés voltak a jellemzők. Azonban fel kellett ismerni, hogy a szoftverprojektek egészen más jellegűek is

lehetnek. Korai szállítással, folytonos visszajelzéssel, és rövid ciklusidőkkel sokat nyerhetünk. Ezek lesznek a jellemzői az agilis projekteknek.

A korai szállításnak köszönhetően a projekt megtérülése jóval kedvezőbben alakul, így a befektetés jóval korábban térülhet meg, és jóval kisebb az esély a kudarcra is. Ha a fejlesztés korai fázisától szorosan bevonásra kerül az ügyfél, és már korán szállítunk, nagymértékben csökken annak a valószínűsége, hogy a projekt egy egészen rossz irányba menne el, és az ügyfél is elégedettebb lesz, mivel már jóval korábban birtokba veheti a kívánt funkcionalitás egy részét. Ráadásul az általa meghatározott preferenciák, prioritások szerint a legértékesebb funkciók kerülnek átadásra az első kiadásban, így a lehető legnagyobb értékű megtérülést tapasztalhatja, már a kezdetektől fogva.

3.3 Agilis technikák, eszközök

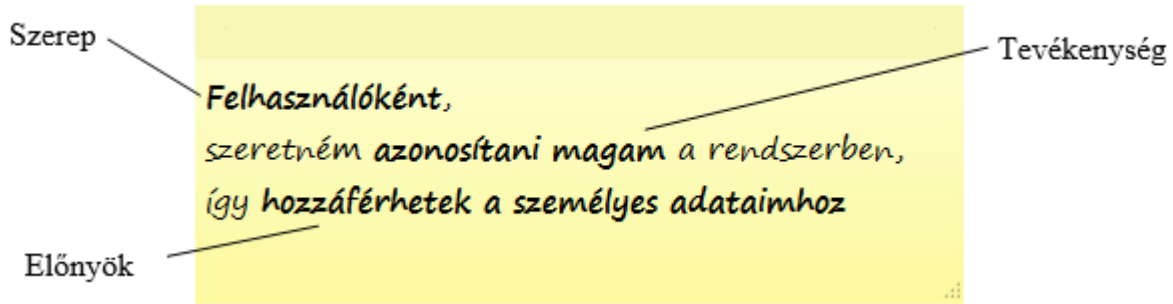
Az agilis tervezésre is az a jellemző, hogy egyszerűbb, de praktikusabb, valóban az igazi célkitűzés elérését támogató eszközöket alkalmaznak. Bonyolult, hosszadalmas projektterv dokumentumok helyett egyszerű diagramokat (pl.: „burn-down chart”), a követelmények rögzítésére tipikusan „User Story”-kat alkalmaznak [8].



2. ábra – Agilis, kollaboratív architektúrális tervezés

A „burn-down chart” egyszerű formában teszi lehetővé a projekt előrehaladásának a követését. Egyszerű kialakításának köszönhetően egy pillantásra áttekinthető a projekt jelen állapota, és a haladás tempója a várthoz képest. Egy a témában kevésbé jártas személy is könnyen megértheti a mondanivalóját. Ez viszont nem feltétlen mondható el a hagyományosan alkalmazott bonyolult eszközökről, és összetett diagramokról.

Az agilis követelménymenedzsmentben hasonlóan egyszerű megoldást alkalmaznak. Igen elterjedt a „User Story”-k alkalmazása, amely egy felhasználó központú megközelítése a követelmények rögzítésének. A hagyományos értelemben vett módon nem is rögzítjük rajtuk részletesen a követelményeket, csak emlékeztetőként szolgál, hogy később még ilyen funkciókra is szükség lehet. A részletek kidolgozása várható, egészen addig, amíg már az adott funkciók megvalósítása következik.

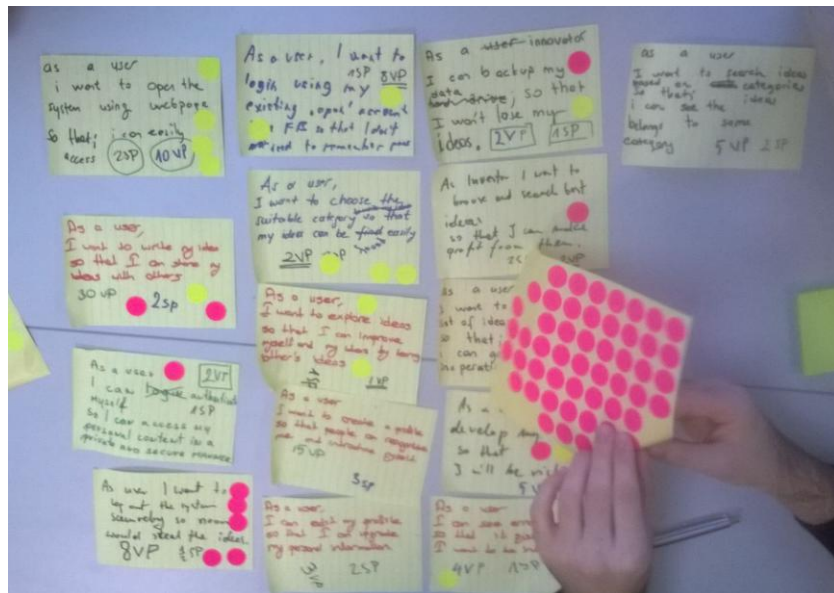


3. ábra - Egy User Story felépítése

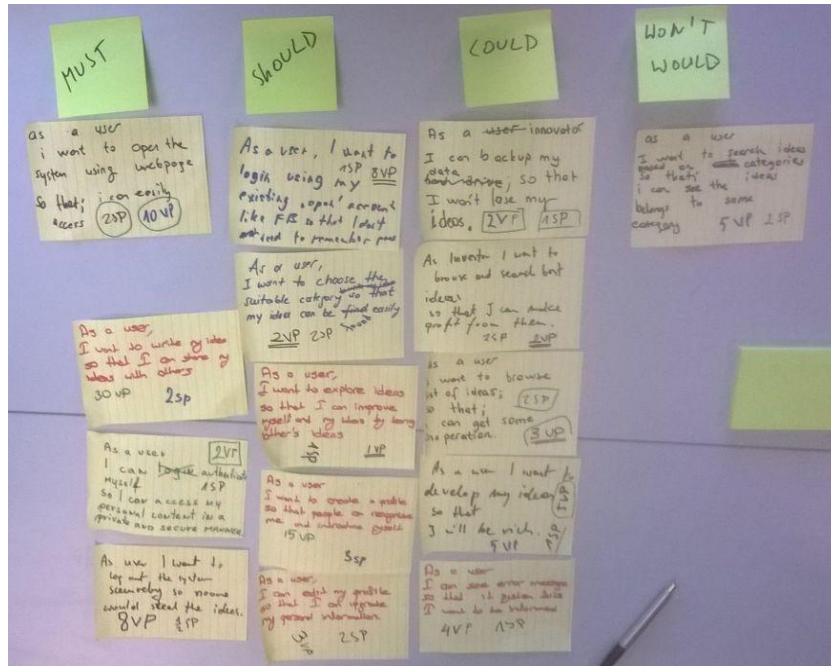
Érdeemes a „User Story”-kat a megrendelő által is jól érthető (szakterület specifikus) nyelven megfogalmazni, így a kommunikációban jól használható segédeszközzé válnak. Első megközelítésben egy rögzített struktúrájú kijelentést fogalmazzunk meg, a következő mondatszerkezet szerint: „szerep-ként, szeretnék tevékenység, így: fő előnyök”. Például magas szinten, egy könyvtári rendszer fejlesztésénél hasonló elvárások jelenthetnek meg: „Könyvtárosként, szeretném tudni meghosszabbítani a kölcsönzési határidőket, így az ügyfelek hosszabbítási igényeit a rendszerben is kezelhetem.”

A jól használható „User Story”-k megfogalmazása és használata némi körültekintést igényel. Érdeemes betartanunk az INVEST elveket. Azaz egy jó „User Story”: független (independent), az ügyféllel tárgyalható (negotiable), értékes (valuable), felbecsülhető (estimable), megfelelően kicsi (small) és tesztelhető (testable). [9]

Ismételten fontosnak tartjuk kiemelni, hogy az agilis szoftverfejlesztésben is tervezünk. Vannak rövidebb és hosszú távú terveink, de ezek egészen más formát öltenek, mint a hagyományos projektmenedzsmentben.



4. ábra – „Pont szavazás”: egy agilis becslési és tervezési technika



5. ábra – MoSCOW: egy agilis becslési és tervezési technika

A rövid távú terveket az iterációk tervezése jelenti, ezek tipikusan 1-2 hét hosszúak, így ez az adott iterációban megvalósításra kiválasztott elemekben ölt testet. Magasabb szinten, a középtávú tervezésre a kiadások tervezése szolgál, erre egy rendkívül jól használható eszköz a „User Story Mapping” (Jeff Patton [10]). Segítségével a csapat (az ügyféllel közösen), együtt kidolgozhatja, hogy milyen kiadásokat (release) tervez majd átadni a megrendelőnek használatra a projekt során. Ez tipikusan úgy épül fel, hogy az első átadásnál, valamilyen minimális, de már üzleti értékkel, a gyakorlatban is hasznosítható funkcionalitáshalmazt megvalósító rendszer kerül átadásra, akár már a projektkezdétől számított 1-2 hónapon belül. Ezt követően a további kiadásokat tipikusan valamilyen magasabb szintű, logikai funkcionalitás köré csoportosítva választják ki, hogy az új kiadás valamilyen jól elkülöníthető, jól meghatározott többletet képviselhesen.

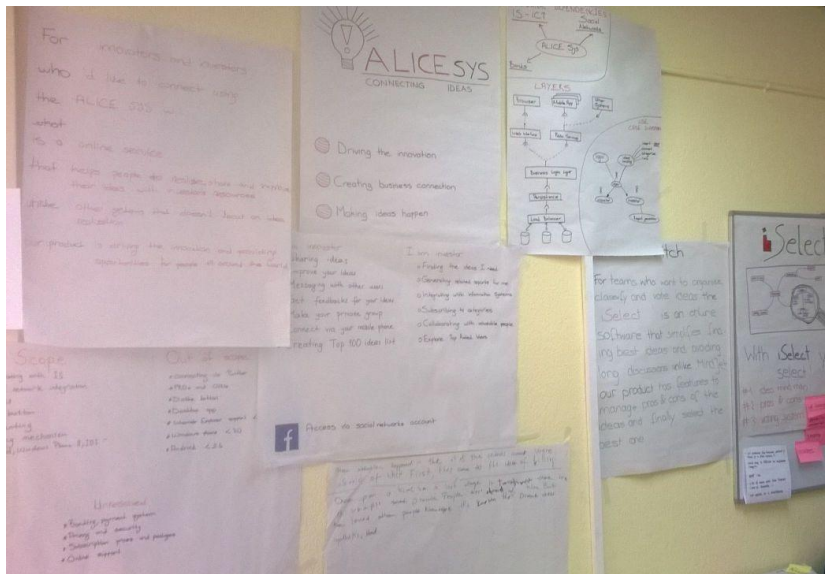


6. ábra - User Story Mapping: Agilis középtávú tervezés

Az agilis projektmenedzsment kevésbé hangsúlyozott része az agilis projektindítás. Hogyan érhetjük el, hogy a projektcsoport teljes egésze tisztán láthassa a várt termék vízióját, és hogy mindezt a megrendelővel közösen, együttműködve, kreatív módon dolgozhatták ki. Ezt a célt szolgálja az Inception Deck[11] létrehozása, amely az agilis projektindítás központi eleme lehet. Segítségével a csapat teljes egésze jól megértheti a termék lényegét, és a projekt pontos célkitűzését.

Lényegében az Inception Deck végrehajtása a termék és a projekt körülményeit jelenti, 10 különböző szempontból vizsgáljuk meg a terméket, amely lépések jól megfeleltethetők a hagyományos projektmenedzsment különböző praktikáinak: kockázatelemzés, környezettanulmány, követelmények meghatározása, és így tovább. A lényegi különbséget a technika csoportos jellege jelenti. A hagyományos projekttervezéssel az agilis projektindítást a csapat együtt, közösen, együttműködve készíti el. Így a folyamat végére biztosak lehetünk abban, hogy minden csapattag „képben van”, és ugyanazt érti a termék alatt.

Segítségével néhány nap munkával kialakítható a következő 6 hónapra a csapat célkitűzése. Ha úgy érezzük, hogy a projekt végrehajtása során, a csapat célt tévesztett, vagy pedig elavulttá vált a terv, akkor az indítási folyamatot újra végrehajthatjuk, így biztosak lehetünk benne, hogy mindenkor a megfelelő célkitűzéseknek megfelelően dolgozunk, maximálisan kielégítve a megrendelő igényeit.



7. ábra - Inception Deck: A termékvízió kompakt összefoglalója

A hosszabbtávú tervezéshez és az agilis szoftverfejlesztés szervezeti keretének biztosítására egy lehetséges megoldást kínál a Scaled Agile Framework™[12]. Ez egy Lean Agile keretrendszert biztosít, amelyek segítségével akár a nagyobb szervezetek is könnyebben boldogulnak az agilis szoftverfejlesztéssel. A megközelítés különböző szintek megkülönböztetését teszi lehetővé, így a hosszú távú tervezés támogatására megjelennek az „epik-ek”, amelyek egyfajta nagyon magas szintű követelményspecifikációnak tekinthetők; valamilyen (egyelőre) ködös elképzelést tartalmaznak a jövőbeli termékekről, projektekről. Ezek részletes kidolgozása később történik meg („just in time”), amikor már valóban szükség van a részletekre is, a valós munkavégzés megkezdéséhez.

Ez a keretrendszer egy rendkívül előremutató irányzatot testesít meg, ugyanis láthatóvá teszi az „áramlás”-t (a flow-t), amely segítségével a Lean Management gyakorlatai és elvei könnyebben átértelmezhetővé válnak a szoftverfejlesztés területére is.

A nagyobb szervezetek az agilis szoftverfejlesztést nehezebben tudják bevezetni, mivel az agilis technikákat tipikusan kisebb, egy munkahelyen, együtt dolgozó csapatokra dolgozták ki, melyek mérete a 3-20 fő közötti tartományba esik.

Ennél kisebb csapatok esetén az értelmét veszti, egyszerűen túl nagy többletmunkát jelent agilisan dolgozni. Nagyobb csapatok esetén pedig a kommunikációs többlet nő túl nagyra. A megbeszélések túl hosszúak lesznek, és az egész folyamat nehézkessé válik. Nagyobb csapatok esetén érdemes a csapatokat felbontani, hogy még továbbra is tudjanak önállóan dolgozni, viszont a létszámuk már az optimális intervallumba essen.

Erre több lehetőség van, amelyek közül talán a „Scrum of Scrums” a legelterjedtebb, amely lehetővé teszi több Scrum csapat közös munkáját.

3.4 A Scrum

És ha már szóba került, említsük meg, hogy az agilis projektek végrehajtására több lehetőségünk is van, de a legelterjedtebb a „Scrum” keretrendszer alkalmazása, amely egy iteratív és inkrementális folyamatot definiál a termékfejlesztés megvalósítására.

Számos speciális szerepet definiál, mint a „Scrum Master” vagy a „Product Owner”.

A „Scrum Master” egy fajta speciális projektmenedzser, aki a Scrum elveinek a betartásáért felel, biztosítja, hogy a folyamat a megfelelően folyjon, és hogy a csapattagok valóban megértsék és megfelelően alkalmazzák a Scrum és az agilitás elveit. A különböző meetingek és események lebonyolításáért is ő a felelős.

A „Product Owner” a megrendelő képviselője. Amennyiben lehetőség adódik rá, az a legjobb, ha valóban a megrendelőtől érkezik valaki, aki a termékkel kapcsolatos kérdésekre válasszal szolgálhat. Ő prioritizálja a tervezett termék funkciót, illetve ő határozza meg a termékkel kapcsolatos egyéb elvárásokat is. Amennyiben a megrendelői oldalról nem érhető el egy személy, egyfajta „proxy”, közvetítő „Product Owner” is betöltheti a szerepet, de ez már kevésbé szerencsés eset.

Az ügyfél rendelkezésre állása, és megfelelő elkötelezettsége mellett még nagyon fontos szerepet tölt be a csapat közös munkaterülete. Lehetőleg egy nagy nyitott irodában kapjon helyet a Scrum csapat, amely tagjai között megvan minden szükséges kompetencia, ami szükséges lehet a projekt végrehajtásához. Amennyiben egy rövidebb szakaszban, valamilyen speciális plusz szakértelemre van szükség a munkavégzés során, akkor specializált szakértőket is igénybe vehet a csapat időről-időre, de a fontosabb feladatok elvégzéséhez szükséges szakértelemnek a csapatban is rendelkezésre kell állnia, hogy megfelelően folyhasson a munkavégzés a projekt során.

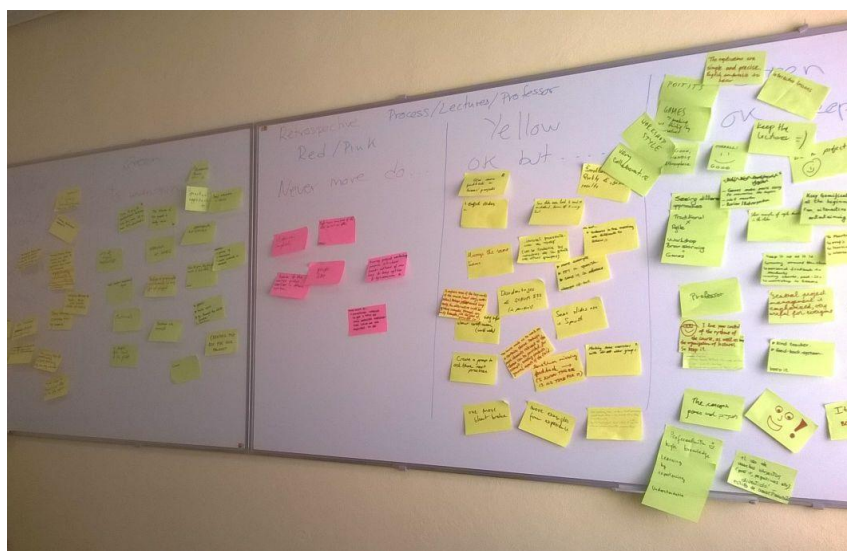
A projektindítást követően a projekttel kapcsolatos elvárások, magas szintű követelmények már rendelkezésre állnak. A valós munkavégzést megelőzően, az első iteráció előtt kerül sor a sprint tervező találkozóra. A Scrum keretrendszerben az iterációkat sprinteknek nevezzük. A sprinttervező meetingen meghatározásra kerül, hogy mely User Story-k, mely funkciók kerülnek megvalósításra majd a sprint során. Ezek megfelelő mértékben meghatározásra, kibontásra kerülnek, hogy ez alapján már megkezdődhessen a munka rajtuk. A legapróbb részletekig azonban felesleges belemenni, hiszen a megrendelő úgymint rendelkezésre fog állni a sprint során is. A termékkel kapcsolatos összes elvárás halmazát a Scrum keretrendszerben Product Backlog-nak nevezzük, míg a Sprint során megvalósításra tervezett, kiválasztott követelmények halmazát Sprint Backlog-nak nevezzük. Így a Sprint tervező meeting során tulajdonképpen a Product Backlog elemeiből kerülnek kiválasztásra, az aktuális, a Product Owner által végzett prioritizálásnak megfelelően a Sprint Backlog elemei, amelyek a következő sprint során elkészülnek. Elkészülés alatt itt valóban teljes mértékben kész funkcionalitásról beszélhetünk, mivel ezek megvalósításra és tesztelésre kerülnek, olyan szinten, hogy a sprint végére elkészülő inkrementum valóban olyan minőségű legyen, hogy az

akár a megrendelőnél átadásra is kerülhessen. (Az angol terminológia szerint itt „Shippable Product Increment”-ről szokás beszélni, mivel tulajdonképpen a sprint során elkészült inkrementum valóban szállítható is lenne az ügyfélhez.)

Tipikus azonban, hogy nem kerül átadásra az ügyfél számára minden sprint után az elkészült inkrementum, csak ha a korábban említett, egy adott kiadáshoz tartozó, logikailag összetartozó funkcionalitás teljes egészében valóban elkészült. Az újonnan elkészült változat telepítése nyilván erőfeszítéseket kíván a megrendelőtől is, így nem feltétlen praktikus minden sprint után újabb és újabb változat telepítése a célkörnyezetben. (Bár ennek minden feltétele adott.)

A sprintek során naponta megrendezésre kerül a napi Scrum meeting, ahol csapattagok beszámolnak az előző találkozó óta elért eredményekről, és az esetleges problémákról. Ezt a találkozót szokás állva megtartani, így a hossza minimálisra csökkenthető, hiszen a csapattagok nem szeretnék hosszasan állni, így mindenki minimálisra fogja a mondanivalóját. (Itt nagyon jól látszódik, hogy túl nagy csapatok esetén ez könnyen elhúzódhat. 20-25 fő esetén, már ha mindenki csak néhány percet beszél, akkor is kezelhetetlenül hosszúra nyúlhat ez a napi találkozó, míg 5-15 fő esetén ez még könnyen megvalósítható.)

A sprint végén két speciális találkozó kerül megrendezésre, ezek a sprint review és a sprint retrospekció. A review (áttekintés) az elkészült inkrementumra és a termékre fókuszál. Bemutatásra kerül az elkészült inkrementum, demót tart a csapat a megrendelőnek, és visszajelzést kap közvetlenül az ügyfélről.



8. ábra - Agilis visszajelzés

A retrospekció (visszatekintés) a folyamatra fókuszál. Azt igyekszik meghatározni a csapat, hogy mi volt, ami jól ment és mi volt a problémás a sprint során. Így a folyamatfejlesztésre is lehetőség nyílik. Kiemelten fontos, hogy az itt elhangzó javaslatok, problémák valóban rögzítésre kerüljenek írásban is, hiszen rögzítés hiányában könnyen lehet, hogy a felmerülő ötletek, javaslatok hamarosan elvesznek, és így nem kerülhetnek a gyakorlatban is alkalmazásra. (Ahogy ez külön kiemelésre került a CHAOS Manifesztóban is. [3])

Láthattuk, hogy a Scrum semmi konkrétat nem mond a projekt végrehajtása során arról, hogy pontosan hogyan folyjék a munkavégzés, így látható, hogy valóban egy keretrendszer. Csak keretet biztosít a munkavégzésnek, azon belül, hogy konkrétan hogyan kerülnek megvalósításra a funkciók, arról nem mond semmit.

Különböző megoldásokról és javaslatokról olvashatunk, mint hogy Kanban-t alkalmazhatunk a Scrum kereten belül a sprintek során, de még a Team Software Process™, vagy a Personal

Software Process™ alkalmazása is lehetséges, ahogy számos esettanulmányban olvashatunk erről. [14][15]

A fentiekben egy rövid áttekintését láthattuk az agilis szoftverfejlesztés különböző, leginkább elterjedt eszközeinek. A lista korántsem teljes, újabb és újabb eszközök és technikák tűnnek fel.

Egy részletesebb bevezető áttekintést találhatunk Jonathan Rasmusson, rendkívül könnyen olvasható *The Agile Samurai* című művében. [11]

4. Agilis oktatás

Az agilis szoftverfejlesztésben alkalmazott technikák és módszerek számos szempontból jelentősen eltérnek a korábban használtaktól. Ebben az esetben a fejlesztő csapat tagjainak egészen más jellegű feladatok is jutnak, így egy kreatívabb, rugalmasabb munkakörben találhatják magukat.

Joggal merül fel a kérdés, hogy egy ilyen, eredendően más jellegű munkavégzést mennyiben érdemes másképpen oktatni? Azokat a szakembereket, akik egy ilyen kreatívabb munkakörben kapnak majd helyet, mennyire érdemes felkészíteni erre a feladatra? Hogyan érhetjük el, hogy valóban képesek legyenek ilyen jellegű munkavégzésre is? Elsősorban fontosnak tartjuk, hogy egyáltalán megismerhessék az agilis munka jellemzőit, a várható feladatok jellegét, így esetleg átértékelhessék korábbi, szoftverfejlesztésre vonatkozó elképzeléseiket és eldönthessék, hogy valóban szeretnének-e ilyen környezetben dolgozni, vagy inkább a hagyományos, dokumentumvezérelt fejlesztés az, ami nekik testhezálló.

A piacon elérhető különféle agilis tanfolyamok, azt hiszem, hogy ezt a kérdést részben meg is válaszolják, hiszen tipikusan fokozott kreativitást és nagymértékű részvételt, kollaborációt várnak el a résztvevőktől, és így készítik fel őket arra, hogy egy ilyen jellegű kreatívabb környezetben is jobban helyt tudjanak állni.

A korábban általam tartott (B.A.) agilis szoftverfejlesztéssel kapcsolatos bevezető tanfolyamaimon közös vonás volt, hogy rendkívül meglepetten fogadták az amúgy egyetemi közeghez szokott résztvevők, hogy nagyfokú aktivitást várok el tőlük, illetve, hogy kreatív munkavégzésre lesz szükségük a tanfolyam során. Hasonló tapasztalatokat olvashatunk az eindhoveni informatikus hallgatókkal kapcsolatban is. [16]



9. ábra - Bevezetés az agilis szoftverfejlesztésbe tanfolyam a Schönherz Zoltán Kollégiumban

Jellemzően nagy élvezettel vettek részt a tanfolyamon, miután túltették magukat az első megrázkódtatáson, hogy itt most tényleg nekik is „osztottak lapokat”, és valójában kipróbálhatják milyen érzés lehet egy ilyen csapatban dolgozni.

Ez pedig mindenképpen nagyon fontos; mivel itt munkavégzés történik, a hallgatóság csak úgy tudja megérteni, hogy hogyan is folyik az ilyen jellegű projektmunka, ha valóban a saját bőrén érezheti, hogy milyen egy önszervező csapat részeként részt venni fontos, komoly döntésekben is. Így az oktatásban részt vevők megtapasztalhatják, hogy milyen is valóban felszabadítani a kreatív gondolataikat, ezáltal belekóstolhatnak ebbe a jellegű munkavégzésbe is. Ehhez pedig nagyon fontosak a megfelelő társas és kommunikációs készségek, ami pedig nem feltétlen volt elengedhetetlen a hagyományos dokumentumvezérelt munkavégzésben.

Számos különböző kezdeményezésről olvashatunk, amelyek ezt a célt tűzték ki maguk elé. Ezek közül kiemelnénk Alexey Krivitsky által kidolgozott, több csapatos Scrum szimulációt, amelynek során a résztvevők LEGO kockákkal való építkezéssel helyettesítik a termékfejlesztési folyamatot, amely segítségével játszva tanulhatják meg a szervezeti méretű Scrum folyamatok lényegét, illetve adnak nagyszerű lehetőséget a tanácsadók, „edzők” (coach) számára, hogy az esetleges kerülendő gyakorlatokra felhívják a figyelmet. [17][18][19]



10. ábra - Több csapatos Scrum szimuláció LEGO kockákkal [17]

Egy szervezet, és egy szakember is rendkívül sokat nyerhet abból, ha az esetleges kerülendő gyakorlatokat egy játékos szimulációban sikerül fülön csípni és nem egy nagy költségvetésű projekt sikere múlik azon, hogy vajon időben ráismerjenek a felmerülő problémákra.

Egy külső szemlélő gyakran könnyebben felismeri a csapatban megjelenő problémákat, mint egy csapattag, aki a teljes folyamat alatt a termékfejlesztésre koncentrálna, és nem feltétlen van lehetősége és ideje a nagy stressz mellett még a csapat munkafolyamataira is kellő figyelmet fordítani. Ráadásul ezek a szimulációk úgy kerülnek megkonstruálásra, hogy lehetőleg minél stresszesebbek legyenek, hogy a résztvevők stresszhelyzetbeli, kritikus fontosságú viselkedését a felszínre hozzák a gyakorlat alatt. Így ezek a fenti célra rendkívül jól használhatóak.

Érdeemes megvizsgálni, hogy a hagyományos egyetemi oktatásban, ezen belül a mérnök-informatikus képzésben hogyan alkalmazhatók az agilis elemek. Ha cikkünk előző fejezeteit

végiggondoljuk, akkor a csapatmunka, kreativitás, rövid iterációk, felelős „megrendelők” gyakori visszajelzések jutnak eszünkbe. Mindezek az egyetemi előadásokon és laborokon is alkalmazhatók – bár kétségtelen, hogy a passzív hallgatóságot kezdetben frusztrálni fogja, ha aktivitást várunk el tőlük.

Ugyanakkor, a gyors, figyelmet megosztó elektronikai eszközök korában a hallgatóság igen hamar alkalmazkodni fog a több „ingerületet” közvetítő módszerhez. Határozottan élvezték a hallgatók a „Planning Poker” kártyákat (amelyekkel licitálni lehetett egy-egy szoftverfejlesztési folyamat időtartamára vonatkozóan).

Tapasztalatunk az, hogy „agilis” oktatás során elsősorban az oktatónak kell agilisnak lennie, és mesterien kell az agilis alapfogalmakkal, eszközökkel és technikákkal bánnia. Jó hatást érhetünk el, ha valós helyzeteket – és nem „iskolapéldákat” – mutatunk be a hallgatóknak.

Órák végén piros, sárga és zöld „cetliket” oszthatunk ki, amelyek közül a pirosakra „Ezt ne tedd máskor!”, a sárgákra „Ez OK volt.”, a zöldekre pedig „Ez nagyon jó volt!” – típusú megjegyzéseket írhatnak. (Írni is fognak – így könnyen, gyorsan értékes visszajelzéseket kaphatunk, melyeket „agilisan” felhasználva előadásainkat jobbá- az „ügyfelek”: a hallgatók számára értékesebbé tehetjük.

5. Tervezett kutatás

A későbbiekben a BME Pszichológia Intézetének munkatársaival tervezünk egy kutatást indítani, amelynek keretein belül az agilis szoftverfejlesztés, és munkavégzés lélektani jellemvonásait szeretnénk részletesebben megvizsgálni. Milyen jellemző személyiségjegyek segítenek egy ilyen csapatban való megfelelő helyállásban? Milyen viselkedésminták figyelhetők meg a csapattagok viselkedésében?

Szeretnénk megvizsgálni, hogy vajon milyen tulajdonságok szükségesek ahhoz, hogy egy csapattag, szoftverfejlesztő valóban sikerrel helytállhasson egy ilyen környezetben is, ami tipikusan különbözik a hagyományos környezettől, amelyben a hasonló munkát végezték a munkatársak.

Irodalomjegyzék

- [1] Ian Sommerville – Software Engineering - 9. kiadás, ISBN-13: 978-0-13-703515-1, 5. oldal
- [2] Edsger W. Dijkstra: The Humble Programmer, Communications of the ACM, 15. kötet, 10. szám, 1972 október
- [3] The Standish Group: The Chaos Manifesto 2013 – Think Big, Act Small, <http://www.versionone.com/assets/img/files/CHAOSManifesto2013.pdf>, (2014.06.21.)
- [4] Idegen szavak gyűjteménye – idegenszavak.hu, <http://idegen-szavak.hu/agilis>, (2014.06.22.)
- [5] Az Agilis Kiáltvány, <http://agilemanifesto.org/>, (2014.06.22.)
- [6] Az Agilis Alapelvek, <http://agilemanifesto.org/principles.html>, (2014.16.22.)

- [7] Tom Arbogast, Craig Larman, és Bas Vodde: Agile Contracts Primer, http://www.agilecontracts.org/agile_contracts_primer.pdf, (2014.06.21.)
- [8] Janeve George: Writing User Stories Effectively, <http://www.janeve.me/articles/writing-user-stories-effectively> , (2014.06.22.)
- [9] Agile Alliance – Agile Guide: INVEST, <http://guide.agilealliance.org/guide/invest.html> (2014.06.22.)
- [10] Jeff Patton: User Story Mapping, http://www.agileproductdesign.com/presentations/user_story_mapping/ , (2014.06.21.)
- [11] Jonathan Rasmusson: The Agile Samurai: How Agile Masters Deliver Great Software, ISBN-13: 978-1934356586, (47-71. oldal)
- [12] Scaled Agile Framework, <http://scaledagileframework.com/> , (2014.06.22.)
- [13] Scrum Primer, <http://www.scrumprimer.org/> , (2014.06.22.)
- [14] Medical Device Summit: Medical Device Software: Leveraging Agile and the Team Software Process, <http://www.medicaldevicesummit.com/Main/Features1/Medical-Device-Software-Leveraging-Agile-and-the-T-1365.aspx> , (2014.06.22.)
- [15] Guoping Rong, Dong Shao, He Zhang: SCRUM-PSP: Embracing Process Agility and Discipline, <http://www.nicta.com.au/pub?doc=4164> , (2014.06.22.)
- [16] Lego Scrum Workshop – Eindhoven, <http://softwaredevelopmentisfun.blogspot.hu/2013/02/lego-scrum-workshop.html> , (2014.06.22.)
- [17] Lego4Scrum – Weboldal, <http://www.lego4scrum.com/> , (2014.06.23)
- [18] Alexey Krivitsky: A Multi-Team, Full-Cycle, Product-Oriented Scrum Simulation with LEGO Bricks the Small & Medium Business Edition, <http://2013.agileee.org/wp-content/uploads/2011/12/Scrum-Simulation-with-LEGO-Bricks-v2.0.pdf> , (2014.06.23.)
- [19] Alexey Krivitsky: Egy több csapatos, Teljes ciklusú, Termék-orientált Scrum Szimuláció LEGO kockákkal Kis és Közepes méretű szervezetek számára, <https://www.dropbox.com/s/zoy78zt2mn1qzyx/ScrumsimulationwithLEGOv2.0-Hungarian.pdf> , (2014.06.23.)

A szimuláció, mint oktatási eszköz¹

Simulation as an education tool

Daiki Tennó^a, Dr Szlávi Péter^b, Dr Zsakó László^c

^{a,b,c} Eötvös Loránd Tudományegyetem Informatikai Kar

Média- és Oktatásinformatika Tanszék

^a dt@inf.elte.hu

^b szlavip@elte.hu

^c zsako@caesar.elte.hu

Absztrakt: A „nyugati típusú” tudományos módszerek alapvető hangsúlyt fektetnek a megismerésre, a jelenségek elméleti magyarázatára, azaz a rendszerek működését szabályozó tényezők pontos modellezésére. Sok olyan jelenség van, amelynek közvetlen megfigyelése nehézségbe ütközik még az eszközökkel jól felszerelt specialisták számára is. Előadásunkban kitérünk egyrészt az informatika szerepére e területeken, másrészt egy lehetséges módszertani felépítést mutatunk, ahogy a témát akár az informatika (programozás) oktatásában fel lehet használni, akár a kifejlesztett rendszerek használatára sor kerülhet egy szaktantárgy keretein belül.

Nézetünk szerint a programozás oktatásban a szimulációs célokat szolgáló programok fejlesztése nagyszerű motiváció a fiatalságnak és minden nyelvi eszközt a megfelelő mélységig be lehet vezetni a tanulók számára úgy, hogy a „célként” fejlesztett programot használni is tudják, nem „csak” a programozás kedvéért hozzuk őket létre. Az előadáson megmutatjuk, hogy néhány soros program segítségével hogyan lehet például gázcsepecskéket tartalmazó „edényben” zajló jelenségeket modellezni, esetleg különböző hatásokat figyelembe venni (pl. gyorsulás, erőtér jelenléte).

Mutatunk arra is példát, hogy ha a szimulációt, mint kísérleti eszközt szeretnénk használni, akkor milyen kihívásoknak kell megfelelni és azokat hogyan lehet a kor szellemének maximálisan megfelelő rendszerekkel megvalósítani. Bemutatunk egy rendszert, amely gázok és folyadékok bizonyos jelenségeinek szimulációjára készült és alkalmas a megfigyelt komplex, kísérleti eszközökkel nehezen megvalósítható, valóságbeli jelenségek modulszerű bemutatására.

A témával lassan évtizedek óta foglalkozunk, jelenleg folyamatban van egy modern szimulációs programcsomag előállítására. Ebből adunk ízelítőt.

Kulcsszavak: szimuláció, modellezés, fizika, biológia, kémia, programozás, objektum, firemonkey

Abstract: Western-type scientific methods put significant emphasis on the comprehension and theoretical explanation of phenomena, that is, on the accurate modelling of factors that govern system operation. There are a great number of phenomena which are difficult even for well-equipped specialists to observe directly. Our paper, on the one hand, will present the role of informatics in these fields; on the other hand, we will offer a possible methodological structure that can be used both in the classes of informatics (programming) and in the education of the specific field.

In our view, developing programs for the purposes of simulation is an excellent task in the education of programming, because, for one, it is motivating for the students and, for two, it is possible to introduce each linguistic tool to the extent that the students can not only create but also use the program. In our talk we will demonstrate how a few-line-long program can be used to model phenomena taking place inside a container of gas particles and how specific effects like acceleration or force field can be considered.

¹ A TÁMOP-4.1.2.B2-13/1 „Pedagógusképzést segítő szolgáltató és kutatóhálózatok továbbfejlesztése és kiszélesítése, 2. Országos módszertani és képzésfejlesztési komponens” pályázat támogatásával.

We will also address the challenges of using simulation as an experimentation tool, while offering suggestions how to execute in the most modern ways. As an example, we will display a system made to simulate certain phenomena related to liquids and gas particles, whose main merit is that it can visualize real-life events that are hard to observe through complex experimentation tools.

We have been researching the topic for decades; in fact, currently we are working on a modern simulation program set. Our paper will give a taste of this too.

Keywords: simulation, modelling, physics, biology, chemistry, programing, objects, firemonkey

Bevezetés

Elsőként szeretnénk rámutatni arra, hogy a szimulációnak a tanulási folyamatban és a tudományos megismerési folyamatban is nélkülözhetetlen szerepe van. Az informatika fejlődésével a szimulációs modellek olyan területeken is megoldással kecsegtetnek, ahol más eszköz használatára és/vagy közvetlen megfigyelésre nincs mód. Hosszan sorolhatnánk a konkrétumokat, most példaként vegyük csak az anyagszerkezet változási folyamatait, amelyekről tudjuk, hogy elemi részecskék (atomok, molekulák) sokaságainak egymásra hatásából, azok együttes viselkedéséből alakulnak ki a számunkra megfigyelhető struktúrák. Nyilván nincs módunk atomi szinten figyelni a történéseket egy páralecsapódást figyelve, de azt tudjuk, hogy a megfigyelhető világunkban az elemi részecskék közötti kapcsolatokból lesznek a páracseppek, az eső, a dér, minden.

A szimulációs eszközök arra koncentrálnak, hogy az elemi egyedek/részecskék között definiálnak elemi szabályokat, és e szabályrendszert addig egyszerűsítik és/vagy finomítják, amíg a modellt „működtetve” olyan eredményt nem kapnak, amely közelíti a valóságban megfigyelhető helyzeteket.

Az informatika segítségével a folyamatok lelassíthatók (pl. maghasadás), felgyorsíthatók (pl. társadalom szimulációk), felnagyíthatók (gáztér molekulái), lekicsinyíthetők (galaktikus mozgások). [1]

Az igazán profi szimulációs rendszerek hatalmas fejlesztői munka eredményeképpen jönnek létre, ennek következtében igen költségesek is. [2,3] Előadásunkban kísérletet teszünk arra, hogy motiváljuk a hallgatóságot: készítsen saját kísérletező eszközt, vagy bátran használjon olyan meglevő rendszert/rendszereket, amelyek szükséges mélységben mutatják meg a vizsgált tudományterület összefüggéseit.

Általános bevezetőnk végén összegzésül újra hangsúlyozzuk, hogy a szimuláció univerzális eszközként számtalan helyen egyedülként használható. [4]

Még egy fontos nézőpontból szeretnénk a szimulációra terelni a figyelmet. Tapasztalataink szerint a szimulációs rendszerek használata nagyon érdeklí a programozással ismerkedő korosztályt. Olykor túlságosan is nagyon érdeklí! Gondoljuk csak meg, hogy az informatikai játékipar legvonzóbb célközönsége ez a réteg, és a játékok zöme profi külsőségekkel rendelkező interaktív szimulációs rendszer!

Ha elfogadjuk e tényeket, akkor kézenfekvő lesz, hogy ha a tanulóinkkal a programozás tanítása során, játék- (szimulációs-) programok létrehozását tűzzük ki célul, akkor a megoldáshoz adott motivációs szintet nagyon hatékonyan megemeljük. Ha praktikus sorrendben bonyolítjuk az egyes szimulációs modelljeinket, akkor egyre bővülő fejlesztéseken keresztül, minden részfázisban sikerélményt adóan a tanulók „húzzák magukat előre” a szebb, a többet tudó „játékok” (szimulációs programok) fejlesztése közben. [4]

Cikkünkben, a cikkhez tartozó előadásunkban, és a jelenleg futó szimulációs eszközrendszer modernizálásával foglalkozó kutatás/fejlesztési munkánkban (*SziMOT-Projekt*) arra is muta-

tunk tematikát, hogy hogyan tartjuk elképzelhetőnek a programozás tanításába bevonni az egyébként szimulációs feladatokat is megoldó programok elkészítését.

A továbbiakban egy konkrét példán keresztül boncolgatjuk e kettősséget.

Feladat kitűzés

Sokszor lehet különböző rendszerek mögött meghúzódó hasonló modelleket megfigyelni. Azt gondoljuk: egyértelmű, hogy ha például tudjuk egy nyúl populáció életét modellezni, akkor hasonlóan tudunk mezei egér populációt is kezelni.

Talán még az is könnyen elfogadható, hogy zárt térben elhelyezett gázmolekulák és szűk helyre betett muslicák eloszlását vizsgálva is felfedezzük a hasonlóságot, pedig ez a két egyed csoport elég távol áll egymástól. (Hosszútávon igaz lesz, hogy a röpködő muslicák is, és a gázmolekulák is statisztikusan egyenletesen töltik ki a rendelkezésre álló teret.)

Végül, gondolnánk, hogy az elektromosan töltött részecskék, a változó elektrosztatikus tér és a társadalmi folyamatok között is lehet párhuzamot találni?

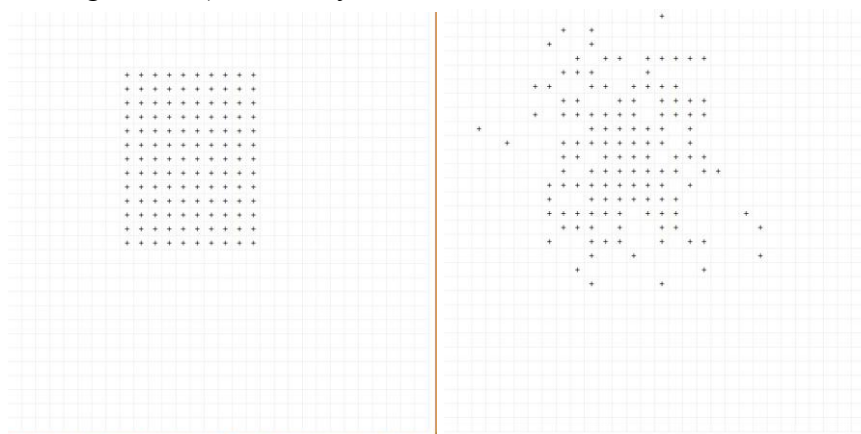
Sok minden csak tálalás kérdése! Ha a diákjainknak érdeklődése fizikai modellek felé irányul, akkor adjunk nekik fizikához közelálló játék (szimuláció) feladatot, ha inkább biológusok, vegyészek, akkor menjünk abba az irányba, ha humán beállítottságúak, akkor adjunk nekik valódi témát! Mindegyikben rengeteg egymásra épülő modell hozható létre egyszerű szabályok egymás melletti alkalmazásával.

Bevezető példánk:

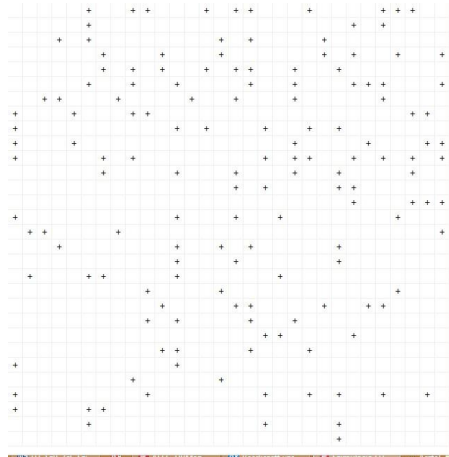
Vegyünk egy életteret, tegyünk bele sok egyforma egyedet! (Később konkretizáljuk az egyed fogalmát!)

Az élettér egyszerűség kedvéért legyen egy téglalap alakú terület, amely olyan méretű (négyzetes) cellákra van osztva, amelyekben vagy van egyetlen egyed vagy üres! Az egyedek között egyelőre semmilyen kölcsönhatást ne tételezzünk fel, de mindegyik véletlenszerűen egy szomszéd üres cellába lépjen, ha épp úgy adódik! [5]

Ennek a feladatnak egy lehetséges megoldása (Fizika_01.dpr) a következő képernyőket hozza (indulás után, sokáig futtatva) eredményül:



1. ábra. Induláskor téglalap alakzatban elhelyezett egyedek (baloldal), majd 4000 lépés után a „kezdődő” eloszlás (jobboldal).



2. ábra. 108 ezer lépés megtétele után a „homogén” eloszlás már egészen jól körvonalazódik.

Megemlítjük, hogy az ábrákat létrehozó program nem több mint 30 (igen, harminc!) program-sor beírásával létrehozható! (Fejlesztéseinket RAD Studio XE5 Delphi környezetben készítettük.)

A programban található deklarációt, értékadást, feltételes utasítást, ciklus szerkezeteket és input/output elemeket is. Azaz mindent, ami a kezdő programozási lépésekhez szükséges.

Amennyiben az *egyed* szó használatával nem vagyunk kibékülve (reméljük ez így lesz!), akkor mondhatjuk, hogy gázmolekulák, s a szimuláció egy edény kitöltését modellezi. Ha ezt értik, akkor azonnal tovább lehet lépni, egyrészt a rendszer viselkedését mérő/vizsgáló/bemutató irányba:

- eloszlásvizsgálat a tér nagyobb egységeiben (Fizika_02.dpr),
- ütközésvizsgálat a falakon (Fizika_03.dpr).

Ha a tanulóink ide eljutnak, akkor azonnal keveslik majd a program tudását. Mit tehetünk értük? Adjunk nekik bővítési lehetőségeket:

- „Melegítsük” az edény egy részét, és nézzük meg, hogy hogyan változik a rendszerünk! (Fizika_04.dpr)
- Legyen az egyedek között valamiféle kölcsönhatás (folyadék cseppek). (Fizika_05.dpr)
- Legyen irányított az egyedek mozgása (állandó erőhatás, gyorsulás, helyfüggő gyorsulás, ...). (Fizika_06.dpr)
- Legyen ki- és belépés a rendszerben (tudjuk folyamatossá tenni az áramló részecskék mozgását).
- Legyen többféle anyag a rendszerben.
- Legyenek „akadályok” a térben.
- Vizsgáljuk az akadályokon mért ütközésszámot.

Talán látható, hogy az egyszerű kis modellünket az iménti bővítésekkel olyan szintre fejlesztettük, amellyel akár forgó folyadékba helyezett tárgyak áramlástani viselkedésének alapjait is lehet szimulálni.

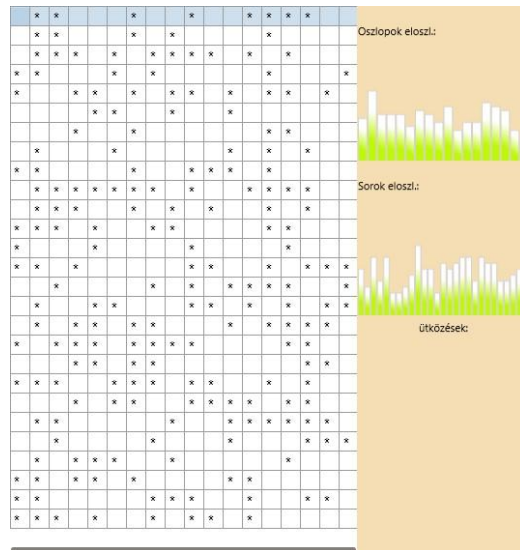
Amennyiben a játékos kedvű tanulóinkat még ez sem elégíti ki, akkor a rendszerbe tegyünk interaktív elemeket, amelyekkel például

- tereljük a részecskéket egy adott térrészbe,

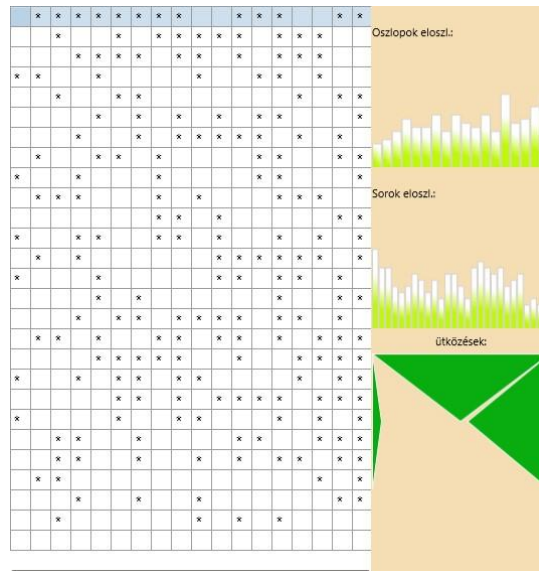
- minimalizáljuk a térben levő „tereptárgyak” ellenállását.

A fenti részfeladatok programozási igénye nem haladja meg az alap programozási képességet. Egymásra épülnek, és az elkészült program használata minden fázisban sikerélményt is ad! [6]

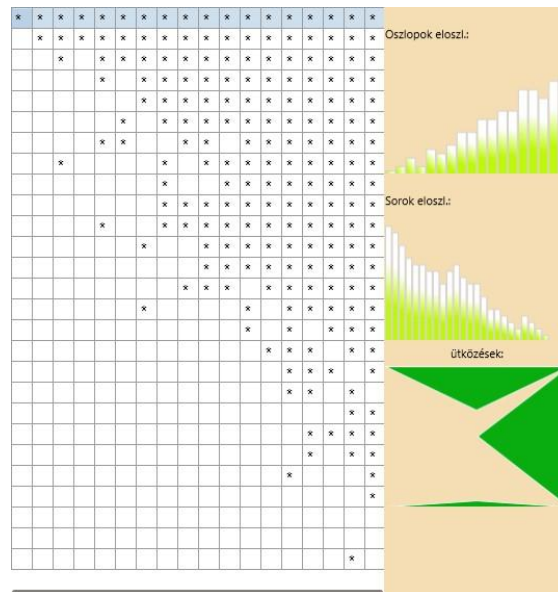
A Fizika_06 rendszerben a maximális lépés nagyságát 5-re állítottuk, cseppek kialakulását nem figyeltük, vízszintes erőhatásokat a következők szerint állítottuk. Balra: 15%; helyben marad: 28%; jobbra lép 57%; a függőleges erőhatásokat fel: 59%; marad: 28%; le: 57% arányban rögzítettük, majd hosszan futtattuk a rendszert (pár perc, kb.: 17 ezer lépés). A folyamat képekben:



3. ábra. Kezdő eloszlás (szép egyenletes eloszlás).



4. ábra. 3500 lépés után (az egyenletes eloszlás már éppen kezd megszűnni, de az ütközésszámon látszik az eredő mozgások irányultsága).



5. ábra. 17500 lépés után. (Nagyon jól látszik a jobbra felfele történő súlypont eltolódás. Ha hosszabb távon figyeljük a rendszert a balra lefele történő mozgások is látszanak, de gyakoriságuk az erőhatásoknak megfelelően alakul.)

Program verziók leírása

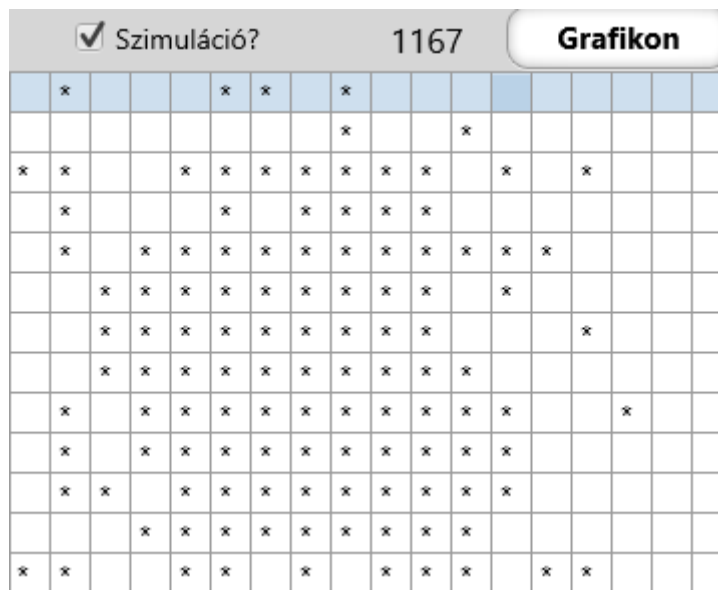
A cikk e fejezetében bemutatjuk, hogy a fejlődő szimulációs rendszerünket, amellyel programozásra is készítjük tanulóinkat, milyen lépések során juttatjuk egyre komplexebb állapotba.

Fizika_01.dpr

Első működő modellünk, mondhatni alap verzió. A legegyszerűbb eszközök használatára törekedtünk, hogy látszódjon, a szimulációs alap algoritmusok nagyon egyszerűek, kevés erőforrással már megoldható feladatok. Kis kiegészítésként becsempésztük a szimulációs tér állapotának a grafikus mentését, hogy utólag, filmszerűen is levetíthető legyen a szimulációs folyamat.

Lássuk, miket használtunk a programban?

- A szimulációs tér megjelenítésére egy **StringGrid**-et (táblázat). Ebben '*' karakter jeleníti meg az egyedeket. Ha a táblázatban az egérrel egy téglalapot rajzolunk, a téglalapba eső cellákba egy-egy egyed kerül a szimuláció során bármikor.
- Az egymást követő szimulációs lépések végrehajtására egy **Timer** komponenst, amely 1 ezred másodpercenként elindítja a folytonos keretmodellre épülő [1] szimulációs algoritmust.
- Egy **CheckBox**-ot (logikai jelölő négyzetet), amellyel szabályozni tudjuk a szimuláció futtatását, leállítását.



6. ábra. A szimulációs tér és a szimulációt vezérlő jelölő négyzet a grafikus eredmény megjelenítő gombbal.

A program megírása közben megismerkedhetünk a programozás alapfogalmaival, de **nem szükségszerűen** magyarázunk el minden egyes sort a tanulóknak! Kipróbált módszer szerint a tanulók is képesek (sokkal jobban, mint a felnőttek!), hogy bizonyos dolgokat egyszerűen elfogadjanak, igazodjanak a környezet adta kényelemhez! (Pl. az objektum orientáltság, vagy az eseményvezérlés,... fogalmát kimerítően nem definiáljuk.)

A tanulók által használni szükséges programozási szerkezetek:

- deklaráció, értékadás, feltételes utasítás, ciklusok

Fizika_02.dpr

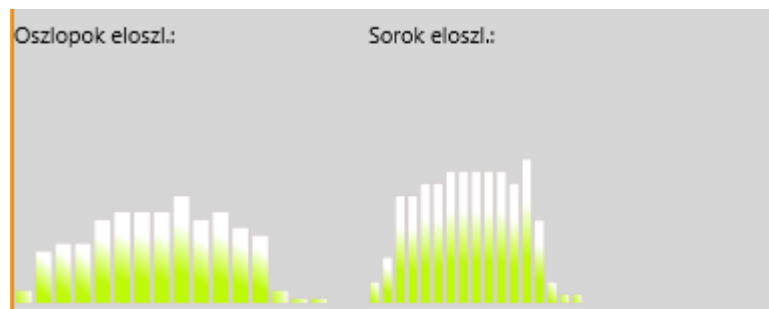
Az első program bővítése a szimulációs térben elhelyezett egyedek eloszlás vizsgálatának megjelenítésével.

Viszonylag nagyot ugrottunk, tudatosan, hogy mielőbb grafikus elemekhez érjünk. Amennyiben ezt valaki merésznék gondolja, a megjelenítéshez használjon numerikus értékek grafikus megjelenítése helyett egyszerű sor/oszlop eloszlási gyakoriságok numerikus megjelenítését.

Az első verzióban is fellelhető mentést kiegészítettük a grafikonok mentésével.

Új elemek:

- **Új ablak** az eredmény megjelenítéshez (ablakból ablak-megnyitás).
- **Gomb** (button) az ablak megnyitásához, visszalépéshez.
- Az ablakban **panel(szerű) komponens**, amelyre téglalapokat rajzoltunk az oszlop, illetve a sorok egyed gyakoriságának megfelelően. (Ezek „kisimulása” mutatja a „hosszú távú” egyenletes eloszlást.)



7. ábra. Egy pillanatnyi eloszlás helyzet. Látszik, hogy a széleken kevés a molekula, valamint, hogy az alsó sorokba még nem jutott el a kezdeti helyzetből egyetlen molekula sem.

Új programozási szerkezetek:

- **unit** (függvénykönyvtár) fogalma,
- **komponens létrehozása/megszüntetése**.

Fizika_03.dpr

Az előző verziót kiegészítettük adott „időegységen” belüli (500 lépés) ütközésszámok megjelenítésével. Oly módon oldottuk meg e feladatot (ismét a grafikus újdonságokra helyezve a hangsúlyt!), hogy egy téglalap megfelelő oldalaira befelé állítunk egyenlőszárú háromszögeket, és a háromszög magasságai lépés-csoportonként jellemzi az adott falra esett ütközések számát. Ily módon ugyan időben mindig csak az aktuális ütközés számot látjuk, de a mentett képkockák levetítése beszédesen mutatja be az eloszlás következtében egyenletessé váló falak terhelését.



8. ábra. Az egyes falakra ható „nyomás”, ütközés szám ábrázolása. (A két oldalsó falra kisebb ütközésszám volt megfigyelhető, mint az aljára, tetejére.)

Új elem:

- szabadon használható „**poligon**” megjelenítő komponens (TPath).

Új programozási szerkezetek:

- **eljárás** létrehozása,
- kis **matematika**, az egyes háromszögek számolásához.

Fizika_04.dpr

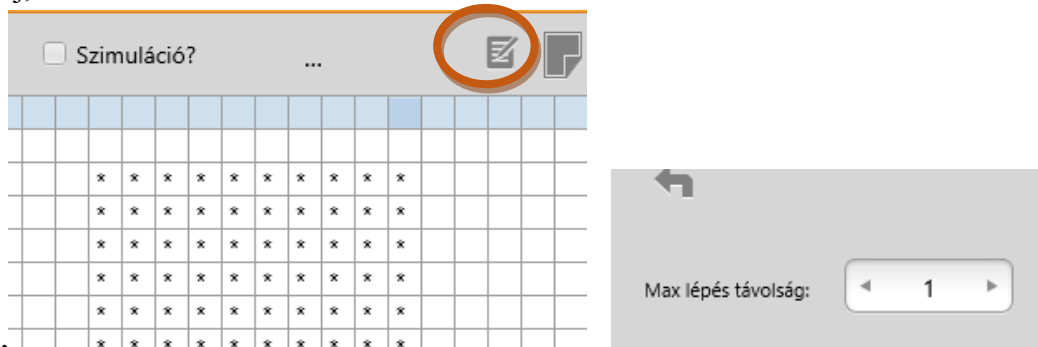
Ebben a változatban jelenik meg először olyan eszköz, amellyel a szimuláció során folyamatosan beavatkozhatunk a rendszer működésébe. A tér egy részében „megemeltük a hőmérsékletet”, azaz lehetővé tettük, hogy e térrészben tartózkodó egyedek nagyobbat is léphessenek, mint eddig, amikor csak a szomszéd cellába mozdulhattak. Bevezettünk egy „Max. lépés távolság” nevű paramétert, és hozzá létrehoztunk egy egyszerű felületi eszközt,

amelynek segítségével állíthatóvá vált, hogy az edény alsó felében levő „melegített” molekulák mekkorát képesek lépni.

Változtattunk a falak ütközés számolásán, figyelembe véve a molekula sebesség változás miatti impulzus változást is.

Új elem:

- Új, **Paraméter** beállításra előhívható **ablak**.



9. ábra. Paraméter beállító ablakot előhívó gomb (bal oldal) és a beállító vezérlő kontrol (jobb oldal).

- A szimuláció folyamán bármikor állítható „hőmérséklet érték” (**TSpinBox**).

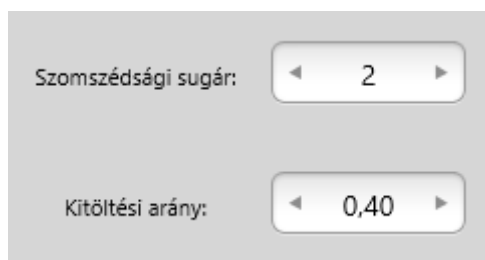
Fizika_05.dpr

Az előző verziókhoz képes a modellünk abban bővült, hogy az eddig önállóan változó egyedek egymással is kölcsönhatásba szeretnénk hozni. Be akartuk mutatni, hogy ha bizonyos sugárban a „látható” egyedek vonzzák egymást, akkor ez a térben speciális csoportokat hoznak létre. (A molekulák cseppeket kezdenek formálni.)

Két fontos változtatást vezetünk be. Az egyik, hogy a teret véletlenszerűen töltjük föl induláskor, a másik, hogy az egyedek egymásra hatásának hatósugara (ezzel együtt a kialakuló mintázat) a szimuláció során változtatható.

Új elem:

- A paraméter beállításra előhívható ablakba bekerült egy „szomszédsági sugár” és egy „kitöltési arány” paraméter.



10. ábra. Újabb elemek a paraméter állító ablakban.

- A beállító ablakból való visszalépéskor helyez el az algoritmus véletlenszerűen egyedeket a térben a megadott aránynak megfelelő sűrűségben.

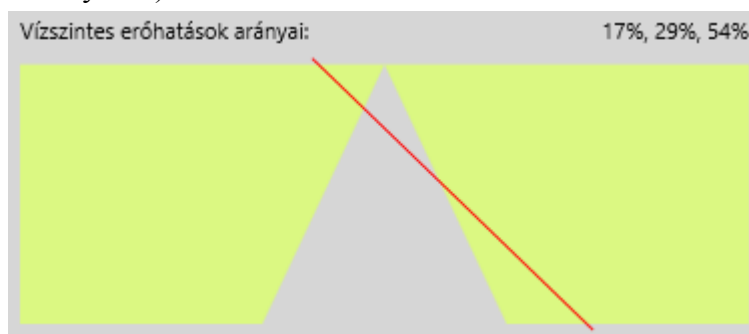
Fizika_06.dpr

Játékunkat tovább fejlesztettük oly módon, hogy a felhasználó képes legyen a következő beállításra: egy kiválasztott egyed a választható szomszéd cellák közül (irány, távolság)

melyeket részesítse előnyben a mozgása során. Modellezhetővé vált önállóan és/vagy együtt a vízszintes és a függőleges erőhatás is.

Hogy ne legyen unalmas a beviteli eszközöknek az alkalmazása, ebben a verzióban szemléltetjük, hogy hogyan lehet egészen speciális vezérlő eszközöket létrehozni. Ne ijedjen meg a kedves olvasó, ha esetleg kevésbé jártas a programozásban, egy önálló objektum osztályt definiáltunk, amely az ábráknak megfelelően teszi lehetővé egy háromállapotú eseményrendszerben az események bekövetkeztének beállítását. Szerettünk volna a mobil (kis képernyőméretű) felületek kezelésére is egy lehetséges példát nyújtani.

A két csúcsával összeérő trapéz és a közöttük kialakított háromszög az ábrán látható szakaszt 3 részre vágják. A 3 rész aránya tetszőlegesen állítható és csak egyetlen ujj elegendő a pontos beállításhoz is! (Természetesen tudjuk, hogy van egyszerűbb módja, de sok helyzetben ez a hozzáállás a leghatékonyabb!)



11. ábra. Vízszintes erőhatás arányai: egy egyed 17% valószínűséggel lép balra, 29% az egyhelyben maradás és 54% a jobbra lépés valószínűsége.

Új elem:

- Saját objektum osztály létrehozása, amely gondoskodik az érintőképernyős hármas arány megadásáról.

Összegzés

Egy rövid sorozattal bemutattuk, hogy a programozási készség fejlesztéséhez milyen lépéseken át vezethet az út, ha szimulációs feladatokat tűzünk ki a diákok számára.

Tudjuk, hogy a kezdeti egyszerűbb lépések után nagyot emeltünk a mércén, de a forrás programok letöltése és áttanulmányozása után hisszük, hogy érthetővé válnak a cikkben felvetett gondolatok.

Nem véletlen, hogy sok programozási versenyen (Nemes Tihamér OITV [7], Izsák Imre Gyula matematika-fizika-informatika verseny [8]), illetve programtermék versenyen (Neumann János Nemzetközi Programtermék Verseny [9]) jelennek meg szimulációs feladatok, szimulációs programok.

A professzionális szimulációs programok mellett megjelentek középiskolásoknak szóló szimulációs célú nyelvek, környezetek, mint pl. a NetLogo [10]

Irodalomjegyzék

- [1] Horváth László, Szlávi Péter, Zsakó László: Modellezés és szimuláció, *Mikrológia sorozat 1.* (2005)
- [2] Interactive Science Simulations. University of Colorado at Boulder, PhET project (2014)
<http://phet.colorado.edu/>
- [3] Virtual Amrita Laboratories Universalizing Education. Amrita Vishwa Vidyapeetham University (2014)
<http://amrita.vlab.co.in/>
- [4] Szlávi Péter, Zsakó László: Az informatika alkalmazási típusai a közoktatásban. *Informatika a Felsőoktatásban '96 – Networkshop '96*, Debrecen, 1996. augusztus 27–30, konferencia CD, 534–543
- [5] Manfred Eigen, Ruthild Winkler: A játék. Gondolat Könyvkiadó (1981)
- [6] Szlávi Péter, Zsakó László: Informatikai kompetenciák: A valós világ modellezése. *INFODIDACT 2013*, Zamárdi, 2013. november 21–22, konferencia CD, 1–17
- [7] Nemes Tihamér Országos Informatikai Tanulmányi Verseny – Programozás kategória. (2014)
<http://tehetseg.inf.elte.hu/nemes/index.html>
- [8] Izsák Imre Gyula komplex természettudományi verseny. Zrínyi Miklós Gimnázium, Zalaegerszeg. (2014)
<http://www.zmgzeg.sulinet.hu/izsak/>
- [9] Neumann János Nemzetközi Tehetségkutató Programtermék Verseny. I. Béla Gimnázium, Szekszárd (2014)
<http://www.ibela.hu/neumann/>
- [10] Bernát Péter: Modelling and simulation in education and the NetLogo simulation environment. *Teaching Mathematics and Computer Science* (2014) megjelenés alatt.

A matematikai szoftverek szerepe a mérnöki problémamegoldás átalakulásában

About the role of mathematical software in the alteration of problem solving in engineering

Kocsis Imre^a, Tóth László^b, Vámosi Attila^c, Deák Krisztián^d

^aDebreceni Egyetem Műszaki Kar
kocsisi@eng.unideb.hu

^bDebreceni Egyetem Műszaki Kar
laszlo.toth@bayzoltan.org

^cDebreceni Egyetem Műszaki Kar
vamosi.attila@eng.unideb.hu

^dDebreceni Egyetem Műszaki Kar
deak.krisztian@eng.unideb.hu

Absztrakt: A napi mérnöki munka döntően olyan műszaki rendszerek és folyamatok kezelését jelenti, melyek mögött kidolgozott és igazolt matematikai modellek és elméletek állnak. Az elméleti eredmények megeremtik az elvi lehetőségét a műszaki döntéseket megalapozó számításoknak, de ezek a számítási eljárások a nagyteljesítményű személyi számítógépek és a megfelelő szoftverek széleskörű elterjedése előtt megvalósíthatatlanok voltak. A döntéseket a tapasztalatokra, és a mérnöki becslésekre, az aktuálisan érvényben levő szabványokra alapozták, ami – a biztonsági követelmények elsődlegessége miatt – természetesen egyrészt a rendszerek túlméretezéséhez, másrészt fölösleges beavatkozásokhoz vezettek. Az informatikai eszközöknek az utóbbi évtizedekben megvalósult fejlődése teljes mértékben átalakította a mérnöki munkát. A legnagyobb hatása bizonyára a (gépészeti, statikai, villamos, stb.) tervező rendszerek elterjedésének volt. Ezekben a rendszerekben a háttérben futnak azok az algoritmusok, melyek az elméletben levezetett számításokat megvalósítják, a matematikai háttér a felhasználó számára rejtve marad, ezek ismerete nem feltétele a szoftverek szakszerű alkalmazásának. Ezek az alkalmazások a szerteágazó mérnöki tevékenységnek azonban csak egy részét fedik le, vannak olyan feladatok, melyek bonyolult matematikai számítások közvetlen alkalmazását igénylik. Az 1990-es évek előtt diplomát szerzett mérnök generációk a tanulmányaik során még nem találkoztak a nagy tudású matematikai szoftverekkel, és ezek hiányában úgy tekintenek a számítási módszerekben szereplő formulákra, mint szép elméleti eredményekre, de fel sem merül bennük, hogy a számításokat ténylegesen végrehajtsák. Az aktív számítógép-felhasználóként felnőtt generációk kezében azonban ott van a lehetőség, hogy munkájukban támaszkodjanak a matematikai szoftverekre, és ezzel hatékonyabban végezzék munkájukat anélkül, hogy a matematika „szerelmesei” lennének. E szemlélet alapjaiban átalakítja a mérnöki gondolkodásmódot és messzemenően előtérbe helyezi a „*mi van akkor, ha ...*” megközelítést mind a szerkezetek, rendszerek tervezésében, mind pedig az üzemeltetésben. Ebben a cikkben a Debreceni Egyetem Műszaki Karán a „Törésmechanika alapjai” tárgyban szerzett tapasztalatokon keresztül mutatjuk be a szemléletváltás lehetőségét, és azt, hogy a korszerű informatikai eszközökre alapozott feladatmegoldás miként változtathatja meg a műszaki döntések megszületését. A tapasztalatok felhívják a figyelmet arra, hogy a műszaki képzésekben a matematikai oktatás nem nélkülözheti az eszközrendszerek (általános matematikai szoftverrendszerek) használatának megkövetelését, és a műszaki tárgyakban a számításiigényes „*mi van akkor, ha ...?*” típusú oktatási szemlélet bevezetését.

Kulcsszavak: matematikai szoftverek, mérnöki döntések, törésmechanika

Abstract: A dominant part in the everyday work of engineers is the management of technical systems and processes. Although the mathematical background (models and theory) of the engineering decision making was

given, before the spread of high performance personal computers and software, mathematical knowledge was more theoretical than practical. Decisions were based on engineering experience, estimates and standards which – for safety requirements – led to oversized structures and unnecessary steps in many cases. During the previous decades engineers' work was reformed due to a development in IT. Probably, the effect of CAD systems (mechanical, civil, and electrical engineering applications) was the most significant. In these systems the algorithms are mostly hidden, users do not have to be aware of the details of the mathematical methods running in the background. However, there are fields of engineering where complicated mathematical calculations have to be realized directly. Generations of engineers, graduated before 1990s, who could not encounter developed mathematical software which are widely used today but was not available during their studies. Consequently these engineers consider the complicated formulas and methods as nice theoretical results without any practical benefit, and they ignore them in their everyday work. Members of the new generation have grown up as active computer users and have had the opportunity to use up-to-date mathematical software to enhance the efficiency of their work. The range of professional tools available nowadays has altered the mindset of engineers and support the "What, if ..." approach in the design and operation of structures and processes. This paper, based on our wide-ranging teaching experience of "Fracture mechanics" courses at the Faculty of Engineering, University of Debrecen, presents the possibility of bringing changes in the engineering thinking and problem solving by using mathematical software which can alter the engineering decision making process. Sharing our experience might draw attention to the role of mathematical software and to the importance of the introduction of the "What, if ..." approach in engineering education.

Keywords: mathematical software, decisions in engineering, fracture mechanics

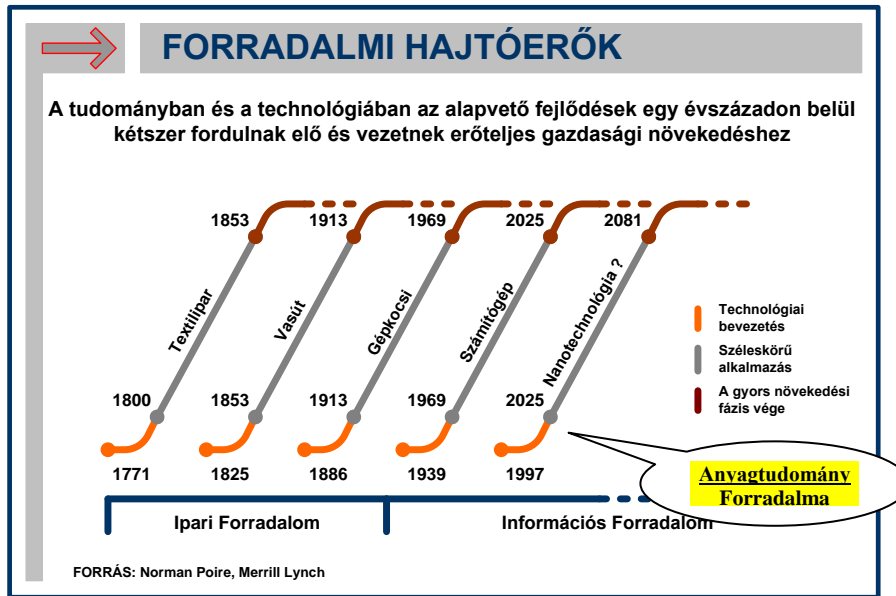
1. Bevezetés

Az idősebb generáció tagjaival folytatott beszélgetések során gyakran elhangzik a „bezzeg a mi időnkben” kifejezés, valamiféle elégedetlenséget kifejezve a világ aktuális dolgaival kapcsolatban. Ez az ellentét, amit szokás „generációs problémaként” emlegetni, törvényszerűen jelen van a társadalmakban, a gazdaságban, a mindennapi élet számos területén. Az egymást követő generációk gondolkodásmódjának változása jelentős mértékben a műszaki fejlődés következménye, hiszen az új műszaki eredmények nagymértékben, és sokszor robbanásszerűen megváltoztatják a gazdasági és társadalmi viszonyokat. Az elmúlt öt-hat generáció életét gyökeresen megváltoztató ún. „forradalmi hajtóerőket” az 1. ábra szemlélteti.

Az átalakulás, a fejlődés forradalmi hajtóerői rendre a következőket eredményezték:

- *textilipar* ⇒ megjelent a társadalmakban egy új réteg, a munkásosztály;
- *vasút* ⇒ megvalósult az egyén, az áruk nagy tömegének szervezett, menetrendszerű mozgatása;
- *gépkocsi* ⇒ megvalósult az egyének, az áruk önálló elhatározáson nyugvó mozgatása;
- *számítógép* ⇒ az információs technológia megjelenése új távlatokat és lehetőségeket nyitott a globalizációban;
- *anyagtudomány* ⇒ megvalósult az anyagtudomány forradalma, az élő- és élettelen anyagok egységes rendszerben történő kezelésének lehetősége.

A fentiek számos következménye közül ki kell hangsúlyozni, hogy forradalmi változások mennek végbe a termékekkel, a piaccal kapcsolatos gondolkodásmódban, a termelési, gyártási folyamatok jellegében, az iparvállalatok struktúrájában, stratégiai céljaiban és – ezekkel összefüggésben – a társadalomban is; átalakulnak a társadalmi értékek, kapcsolatok, az munka értékrendszere. Minden generációnak szembesülnie kell életében legalább egy ilyen mértékű változással, ehhez szükségszerűen alkalmazkodnia kell, fel kell készülnie az új struktúrában betöltendő szerepére, pozíciójára.



1. ábra. A gazdaságot, társadalmat átformáló műszaki alapú hajtóerők.

Napjaink egyik ilyen generációs problémáját a *számítástechnika* robbanásszerű fejlődése idézte elő. Az új eszközök és szoftverek az élet minden területén új lehetőségeket teremtettek, de ezekkel a lehetőségekkel nem egyformán képes élni mindenki. Általánosságban el lehet mondani, hogy minél fiatalabb korban ismerkedik meg valaki az új technológiával, annak inkább képes annak teljes értékű használatára. Az idősebb generáció részéről gyakran tapasztalható akár teljes mértékű elutasítás is a fiatalok számára már természetes újdonságokkal kapcsolatban, és a jól bevált módszerekhez való görcsös ragaszkodás.

Az informatika fejlődése nyomán bekövetkező egyik lényeges változás, hogy a matematika a *hozzáértők privilégiumából az alkalmazók eszközévé* vált az informatikai eszközök mindennapi használói számára. Persze az átlagos felhasználóban ez általában nem tudatosodik, hiszen a matematikai tudás „be van építve”, nem kell tudni róla, mint ahogyan a járművezetőnek sem kell ismernie az autója elektromos rendszerét ahhoz, hogy az ez által biztosított lehetőségeket megfelelően használja.

A matematikai szoftverek elterjedése egy különleges vonulata az IT térnyerésének, mivel a számítási lehetőségek kibővülése teljesen átalakította a gondolkodásmódot, lehetővé vált az elméleti eredmények „apró pénzre” váltása, a gyakorlati életben fellépő problémák megoldása azáltal, hogy a munka „nehezt” a számítógép végzi a betáplált algoritmusok futtatásával.

A műszaki életben igen nagy gazdasági jelentőséggel bír, hogy célszoftverek segítségével elfogadható költséggel meg lehet válaszolni „*mi van akkor, ha ...?*” típusú kérdéseket, hatékonyan lehet elemezni alternatív műszaki megoldásokat, és kiválasztani a lehető legmegfelelőbbet.

A szoftverek többségénél rejtve marad a matematikai háttér, a probléma megfogalmazásakor, és a szoftver által prezentált eredményben is a műszaki tartalom az elsődleges (ilyenek például a CAD vagy a VEM rendszerek). Vannak azonban olyan problémák, melyek a matematikai eszközök közvetlen alkalmazását igénylik. Ilyen esetekben a matematikai szoftverek jelentik a megoldást, de ehhez szükséges, hogy a mérnök tisztában legyen a lehetőségekkel, és rendelkeznie kell a minimális gyakorlattal, amire építhet a munkája során.

Ezért a műszaki matematika oktatása során be kell vezetni a matematikai szoftverek használatát, és az elméleti alapok mellett meg kell követelni a szoftverek alapszintű ismerete. Ha egy mérnök tisztában van a szoftverek által kínált lehetőségekkel, és kellő gyakorlatot

szerez ezek használatában, akkor a munkája során nem „fél” a számítógép által elvégzett számításokra hagyatkozni, a jelenségek leírása számára nem pusztán egyenletekként jelennek meg, hanem a megoldások keresésének természetes módjává válik.

Felismerve annak lehetőségét, hogy a számítógépekkel együtt felnőtt fiatalok számára a számítógép használata a feladatmegoldásban természetes, a Debreceni Egyetem Műszaki Karán a „Törésmechanika alapjai” című tárgy keretein belül azt a rendszert alakítottuk ki, hogy a gyakorlaton a hallgatók megismerik a gyakorlati törésmechanikai feladatok megoldása során használható matematikai szoftvereket, majd a tudásuk igazolása után kapnak egyéni feladatot, melyet projektmunka formájában lépésenként ki kell dolgozniuk a szoftver segítségével. A munkájukat végül prezentáció formájában kell bemutatniuk.

A tárgyat, a fent leírt új szemléletű megközelítés bevezetése óta, egy alkalmazott mechanika ágazaton végzett gépészmérnök és egy matematikus együtt oktatja. Az alábbiakban bemutatunk egy tipikus projekt feladatot a Maple szoftver segítségével, de a kidolgozás során a hallgatók bármely matematikai szoftvert használhatják.

2. Törésmechanika

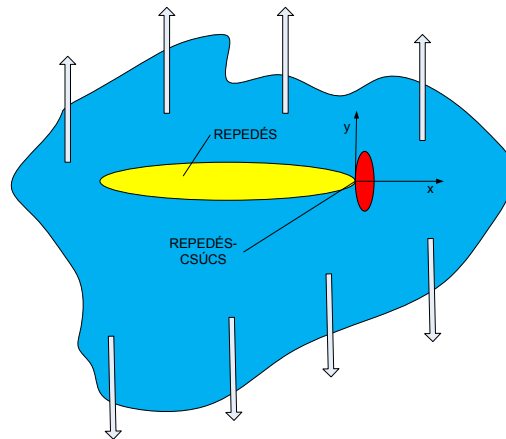
A mérnöki szerkezetek „teherbírásának” és biztonságának megítélése szempontjából fontos szerepe van az anyagban jelenlévő repedéseknek. A repedések terjedésével foglalkozó törésmechanika első nagy fejlődési időszaka a repedés-diagnosztikai módszerek elterjedéséhez köthető. Repülőgép-szerkezetek, hajószerkezetek, egyéb ismétlődő terhelésnek kitett szerkezeti elemek, járműalkatrészek tönkremenetelét vizsgálva derült fény az ún. ridegtörés jelenségre. Nagyméretű acélszerkezetek (hidak, hajók, tartályok) rideg törése sok súlyos katasztrófa kiváltó oka volt főleg a huszadik század második harmadában. A jelenség lényege, hogy a szerkezeti elem látszólag átmenet (előzmények) nélkül eltörik el, mert az anyagában minden esetben jelenlévő kezdeti repedések – a statikus teherbíráshoz képest – kismértékű, de sokszor ismétlődő terhelés hatására lassan, észrevétlenül növekednek a kritikus repedéshossz eléréséig, aztán a repedéshossz növekedése felgyorsul, és akár másodpercek alatt bekövetkezik a szerkezet törése.

A rideg törés jelenségének felismerése óta igen sok tapasztalat és elméleti eredmény született, melyek a törésmechanika tudományában összegződnek. De mit is takar pontosan a „törésmechanika” fogalma? *A törésmechanika a kontinuummechanika része, amely a repedésszerű hibákat tartalmazó szilárd testek külső hatására bekövetkező várható viselkedését tárgyalja.* Ha ez a definíció elfogadható, akkor a törésmechanikai tárgyalásmód mindenképpen két mozzanatot foglal magába:

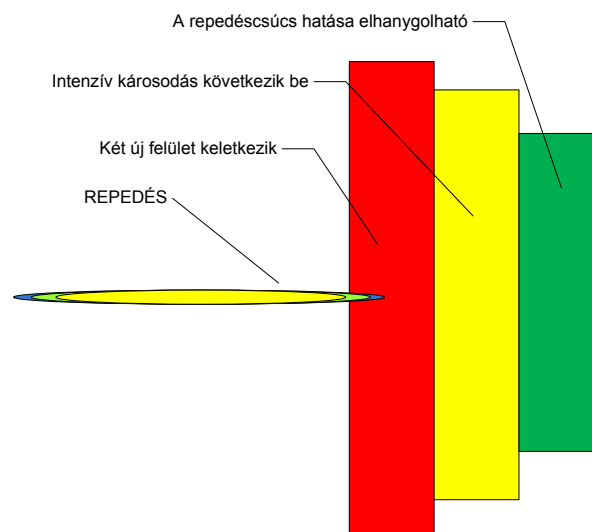
- a repedés csúcspont környezetében *kialakuló mezők* (alakváltozásai, feszültségi, energetikai) leírásának modelljeit;
- a repedés *stabilitási kritériumait* attól függően, hogy milyen modellt alkalmaztunk a mező leírására (e kritériumok feszültségi, alakváltozási vagy energetikai jellegűek lehetnek, amelyek közül az első kettő irányfüggő, az energetikai pedig skalármennyiség).

Miért kellett mindezt előre bocsátani? Azért, hogy egyértelműen megkülönböztethető legyen a törés

- matematikai elmélete, és
- mérnöki szemlélete.



2. ábra. A repedéscsúcs környezete.



3. ábra. A repedéscsúcs környezetének károsodása.

Mindkét esetben *modellt kell alkotni* a törés folyamatának leírására. A modell megalkotásához tekintsük az 2. és 3. ábrákon látható vázlatokat. Ha egy szilárd testben repedésszerű hiba van és ezt a testet külső hatás éri, akkor a repedés csúcsa előtt keletkezik egy olyan tartomány („az élet és halál mezsgyéje”), amelyben a következő folyamatok játszódnak le:

- nagyon intenzív károsodás, amelynek jellemzője a kialakuló mezők nagy gradiense (ezt szemlélteti a 2. ábra);
- az anyagra és állapotára jellemző kritikus értékű károsodásnál a lokális törés következik be, amelynek jellemzője a repedésfelület növekedése, azaz új repedésfelületek keletkezése, a korábban egyszeresen összefüggő anyagi tartományban kétszeresen összefüggővé válik.

A legegyszerűbb és egyben legkonzervatívabb elemzés az, amikor feltételezzük, hogy a repedéscsúcs közvetlen környezetében rugalmas alakváltozási, feszültségi mező alakul ki. Ezt teljesen korrekt módon *Ian Naismith Sneddon* az 1969-ben, társszerzővel kiadott, „*Crack problems in the classical theory of elasticity*” című könyvében tette meg. Ebben a modellben a repedés csúcsának környezetében a klasszikus rugalmasságtani elvek alapján számított feszültségmező minden esetben a következő alakú kifejezéssel számítható:

$$\sigma_{ij} = \frac{K}{\sqrt{2\pi r}} f_{ij}(\Theta),$$

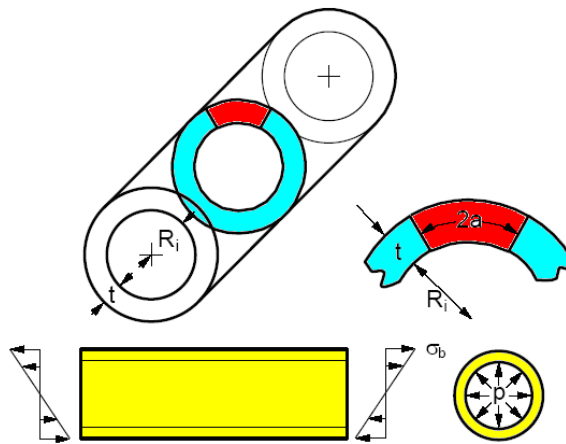
ahol

K - a feszültségintenzitási tényező, olyan invariáns mennyiség, amely függ a repedés hosszától, a repedésfelületre ható terhelés nagyságától és típusától (húzó, nyíró, csavaró), amelyet egyezményesen a terhelés típusára utalva K_I , K_{II} vagy K_{III} -mal jelöl a szakirodalom;

r - a repedéscsúcstól mért távolság;

$f_{ij}(\Phi)$ - a kiválasztott pontba mutató vektor és a repedés síkjával bezárt szögtől függő geometriai tényező.

A K feszültségintenzitási tényező számítására alkalmas összefüggések a gyakorlatban előforduló szinte minden szerkezeti elemre, minden lehetséges terhelési módra rendelkezésre állnak kézikönyvekben, és újabban adatbázisokban. Ezek többnyire meglehetősen összetett, bonyolult kifejezések. Például egy hengeres héjban (cső) levő átmenő repedés (4. ábra) esetén a viszonyokat az 1. táblázatban foglaltuk össze.



4. ábra. A belső nyomással és hajlítással terhelt csőben levő átmenő repedés.

Geometriai adatok:

R_i – sugár

$2a$ – repedéshossz

t – henger falvastagság

Terhelési adatok:

σ_b – hajlítófeszültség

p – belső nyomás

σ_m – a belső nyomásból származó membránfeszültség

Feszültségintenzitási tényező

$$K(a) = (F_0 \sigma_m + F_1 \sigma_b) \cdot \sqrt{\pi a}$$

$$\sigma_m = \frac{p}{\left(\frac{R_0}{R_i}\right)^2 - 1}$$

$$\xi = 0,3$$

$$R_0 = R_i + \frac{t}{2}$$

$$\tau = \frac{1}{\sqrt[4]{12 \cdot (1 - \xi^2)}} \cdot \sqrt{\frac{t}{R_0}}$$

$$\mu = \frac{\sqrt{2}}{\frac{1}{\operatorname{tg}\left(\frac{\pi - \alpha}{\sqrt{2}}\right)} + \frac{\sqrt{2}}{\operatorname{tg} \alpha}}$$

$$F_{k0} = 1 + \mu \cdot \frac{1 - \frac{\alpha}{\operatorname{tg} \alpha}}{2\alpha}$$

$$I_0 = \left(\sqrt{8} \cdot (F_{k0}^2 - 1) + \pi \cdot \frac{\beta^2}{\delta} \right) \cdot \frac{\alpha^2}{\tau}$$

$$\alpha = \frac{a}{R_0}$$

$$\delta = \frac{\alpha}{2\tau}$$

$$\beta = 1 + \frac{\pi}{16} \cdot \delta^2 - 0,0293 \cdot \delta^3, \text{ ha } \delta \leq 1$$

$$\beta = \left(\sqrt{8} \cdot \frac{\delta}{\pi} \right)^{0,5} + \left(\frac{0,179}{\delta} \right)^{0,885}, \text{ ha } \delta > 1$$

$$F_0 = \sqrt{\frac{I_0}{2\pi\alpha}}$$

$$F_{k1} = 1 + \frac{\mu}{4} \cdot \left(\alpha + \frac{\alpha}{\operatorname{tg}^2 \alpha} - \frac{1}{\operatorname{tg} \alpha} \right) \cdot \frac{\sin \alpha}{\alpha}$$

$$I_1 = \left(\sqrt{8} \cdot (F_{k1}^2 - 1) + \pi \cdot \frac{\beta^2}{\delta} \right) \cdot \frac{\alpha^2}{\tau}$$

$$F_0 = \sqrt{\frac{I_1}{2\pi\alpha}}$$

1. táblázat. A K feszültségintenzitási tényező számítása belső nyomással és hajlítással terhelt csőben levő átmenő repedés esetén.

Az összefüggésekre pillantva látható, hogy ha bármit is szeretnénk mondani egy olyan cső várható viselkedéséről, amelyik egy picit kilyukadt, akkor rendkívül sokat kellene számolni. Azt, hogy a gyakorlatban milyen tipikus kérdések tehetők föl, az alábbi példákkal szemléltetjük:

- Mennyire veszélyes ez a repedés?
- Mennyire érzékeny ez a repedés a repedés detektálása során alkalmazott módszerek megbízhatóságára?
- Milyen lesz a kritikus, a törést okozó repedés mérete?
- Hogyan változik ez a kritikus repedéshossz a repedésterjedéssel szembeni ellenállás, mint anyagjellemző megbízhatóságával?
- A számítási eredmények mennyire érzékenyek a geometriai, a terhelési és anyagi paraméterekre, azaz milyen lehetőségek vannak a biztonság növelésére?
- Ha a terhelés ciklikus, akkor mennyi a maradék élettartam, a berendezés üzemeltethető-e tovább, avagy nem? Ha igen milyen paraméterekkel, ill. milyen lehetőség van a biztonság növelésére?

A kérdések sorozatát lehetne feltenni, amelyeknek mindig és mindenkor igen jelentős gazdasági következményei vannak. Például tartályokban levő repedések kapcsán hazánk jelentős PB gáztároló kapacitásával kapcsolatban az *üzemeltethető / nem üzemeltethető* döntés meghozatala, vagy egy nagyvállalat – napi több százmillió forint termelési értékű – kompresszoraiban levő repedés várható viselkedése kapcsán a *leállítandó / feltételesen üzemeltethető T ideig* döntés.

A gyakorlati példák sorozatát lehetne sorolni, melyekkel összefüggés mindig igaz, hogy

- A roncsolásmentes, hibafeltáró vizsgálati módszerek folyamatosan és robbanásszerűen fejlődnek a mikroelektronikai, informatikai eszközrendszer fejlődése miatt.
- A korábbi felülvizsgálatok során hibamentesnek vélt szerkezetekben egyre nagyobb valószínűséggel anyagfolytonossági hibákat észlelhetünk, amelyek vagy az üzemeltetés során keletkeztek, vagy az észlelési technika fejlődésével váltak érzékelhetővé.
- Ha egy hibát észleltek, akkor nyilatkozni kell annak veszélyességéről, és meg kell válaszolni azt a kérdést, hogy „üzemeltethető-e tovább a szerkezet vagy nem, ha igen akkor változatlan paraméterekkel, vagy milyen feltételekkel?”

A műszakilag megalapozott döntések csak igen sok számítással alapozhatók meg, bonyolult függvények integrálására, deriválására, ábrázolására, zérushelyek keresésére van szükség. Ebben a folyamatban a számítástechnika olyan eszköz, amely nélkül meg sem kísérelhető a műszakilag megalapozott válasz megfogalmazása.

3. A matematikai szoftverek helye a műszaki matematika oktatásában

A műszaki felsőoktatásnak reagálnia kell a közoktatásban és általában a társadalomban az bekövetkezett változásokra, az oktatás eredményességét módszertani eszközökkel kell javítani, mely figyelembe veszik a felsőoktatásba kerülő hallgatók felkészültségét és szemléletmódját. A képzési folyamatban, így például a matematika oktatásában is a képzés meghatározott kimeneti követelményeinek elérését kell megcélozni. Az eredményesség pedig a későbbi mérnöki munka sikerességében mérhető.

A probléma alapú tanítás/tanulás módszerei jelentősen hozzájárulhatnak az eredményesség növeléséhez azáltal, hogy a motiváció a hallgató által választott szakterülethez kötődik.

A tanulóknál állandóan ott vannak a kisebb-nagyobb hordozható számítógépek (okos telefonok, tabletek), melyekkel – telepített szoftverek, vagy az internet segítségével – „mindent meg lehet csinálni” (például az iskolai matematikai feladatok jelentős része megoldható ezekkel), és ez meghatározza a fiatalok szemléletét, a problémamegoldáshoz való viszonyukat. Ezt a tanítási módszerek kialakításakor figyelembe kell venni, és építeni kell rá.

Mióta széles körben elterjedtek a zsebszámológépek kérdés, hogy azokat milyen módon célszerű felhasználni a matematika tanításában. A felsőbb matematika oktatásában a matematikai szoftverek alkalmazása kapcsán hasonló kérdések vetődnek fel. A válasz nem nyilvánvaló: míg az elméleti ismeretek elsajátítását támogató gyakorlás során nem célszerű a szoftverhasználat, mert a számolási lépések szerepének megértése a tanulás része, maga a számolás a tanulás, nem csak az a lényeg, hogy jöjjön ki az eredmény, addig a műszaki példák megoldása során kifejezetten hasznos a szoftverek bevonása, hogy a számolástechnikai nehézségek ne akadályozzák a műszaki problémák megoldásának folyamatát.

A számítógépek fejlődésével jelentősen megváltozott a műszaki elmélet és a gyakorlat kapcsolata. Míg a megfelelő számítási kapacitás hiányában az elméleti eredmények közvetlen gyakorlati alkalmazására nem volt lehetőség, és így a mérnökök végeredményben tapasztalati alapon oldottak meg problémákat, addig ma már a mérnöki számításokra kifejlesztett szoftverek képesek rövid idő alatt pontos számítások elvégzésére. A szerkezetek tervezésekor például a véges elemes szoftverek alkalmazásával elkerülhető a rendkívül költséges túlméretezés, amit korábban a „biztonság kedvéért” meg kellett tenni. A gyakorlati felhasználó számára ezek az alkalmazások fekete dobozként működnek abban az értelemben, hogy nem szükséges tisztában lenni a számolás részleteivel, „csak” helyesen kell megadni a bemenő adatokat, és helyesen kell értékelni az eredményt.

Vannak azonban olyan problémakörök is, melyek bonyolult analitikus számítások végrehajtását igénylik a mindennapi mérnöki munkában. Az itt bemutatásra kerülő példa a törésmechanika témakörből származik, ami tipikus esete annak, hogy a bonyolult analitikus számításokra képes matematikai szoftverek elérhetősége alapvetően változtatta meg a gyakorlati munkát. Korábban a szakemberek hiába ismerték a mechanikai modellekből levezetett formulákat, az azokkal való számolás reménytelen volt, nem is törődtek vele. A nagyteljesítményű számítógépekkel együtt felnövő generációk azonban másképpen állnak a kérdéshez, hiszen megszokják, hogy mindenre van szoftver, így számukra teljesen mást jelent az, hogy mi a „nehéz”.

Az ilyen fajta alkalmazási igény azt követeli meg, hogy a matematikai szoftverek használatára fel kell készíteni a mérnökhallgatókat a képzés során. A Debreceni Egyetem Műszaki Karon az alapozó tárgyak (többek között a matematika, fizika, informatika) oktatásáért felelős Műszaki Alaptárgyi Tanszéken a matematikai tárgyak keretében (elsősorban a matematika utolsó félévében) műszaki tartalmú projekt feladatokat adunk ki, amit a hallgatók szoftverek segítségével oldhatnak meg. A projekt feladatok úgy vannak meghatározva, hogy a megoldás során sok számolási lépést kelljen végrehajtani, és azok több matematika témakört érintsenek.

A hallgatók a saját szakjukhoz közel álló példák megoldása során szívesebben és mindenképpen eredményesebben tudják áttekinteni a matematikai ismereteket, ami a szigorlatra való felkészülést is jelentősen megkönnyíti. A szoftverek ismerete megteremtí a matematikai módszerek alkalmazásának lehetőségét is a később sorra kerülő műszaki tárgyokban.

A matematikai szoftverek alkalmazására épülő projektfeladatok ötlete a gépészmérnöki szak törésmechanika című tárgya keretében kialakult gyakorlatból ered. A törésmechanika tárgy nagy létszámú (50-100 fős) kurzusain az utóbbi félévek során igen kedvező tapasztalatokra tettünk szert az alábbi eljárást illetően: a kurzus első néhány gyakorlata a matematikai szoftverek elérhetőségéről, alkalmazási lehetőségéről szól (néhány kiválasztott szoftver esetén részletekbe menően tárgyaljuk a szükséges számolási eljárások megvalósításának módját), ez után a kurzus hallgatói levizsgáznak az általuk választott szoftverből egy olyan teszt megírásával, mely a későbbiekben szükséges számolási technikákat tartalmazta, így „licenct” szereznek a továbblépésre. Ez után mindenki egyéni feladatot kap, melynek kidolgozása után prezentáció formájában beszámol a műszaki probléma megoldásáról, a következtetésekről, és az alkalmazott matematikai eszközökről.

4. A projektfeladat bemutatása

A törésmechanika fejlődése felgyorsult, amikor a számítógépek számolási kapacitása lehetővé tette a mechanikai modellekben felírt bonyolult egyenletek megoldását. A gyakorlatban előforduló szerkezetekre, repedéstípusokra meghatározták a feszültséget a repedéscsúc közelében leíró feszültségintenzitási tényezőt, valamint az anyagra jellemző törési szívósságot (lásd pl. [1-4]), melyek ismeretében számolható, hogy adott terhelés esetén mekkora a kritikus repedéshossz (a stabil és az instabil repedésterjedés határa), így a szerkezet adott állapota biztonsági szempontból minősíthető annak alapján, hogy az adott a repedéshosszhoz tartozó K feszültségintenzitási tényező hogyan viszonyul a törési szívóssághoz (K_{Ic}). Ennek áttekintéséhez a K - a függvény ábrázolása szükséges.

A feszültségintenzitási tényező érzékenysége a repedéshossz változására szintén fontos jellemző [5], mivel a repedéshossz változása, illetve a leolvasás hibája ennek függvényében befolyásolja a biztonsági tényező (a K és a K_{Ic} értékek viszonyának) értékét. A kvázistatikus repedésérzékenységi index értékét adott repedéshossz esetén a K - a függvény deriváltja adja.

A törési szívósság és az ehhez tartozó kritikus repedéshossz statikus terhelés esetén jellemzi a szerkezetet. Mivel a repedés hossza kisebb terhelés esetén is növekszik (stabilan mert a repedés felületének növekedése, valamint a repedéscsúc közelében kialakuló képlékeny alakváltozás felemésztí az energiát), a gyakorlat szempontjából fontos kérdés, hogy adott szerkezet adott repedése ciklikus terhelés mellett mikor (hány terhelési ciklus után) éri el kritikus repedéshosszat, mert a megengedett üzemeltetési idő ez alapján határozható meg [5-7].

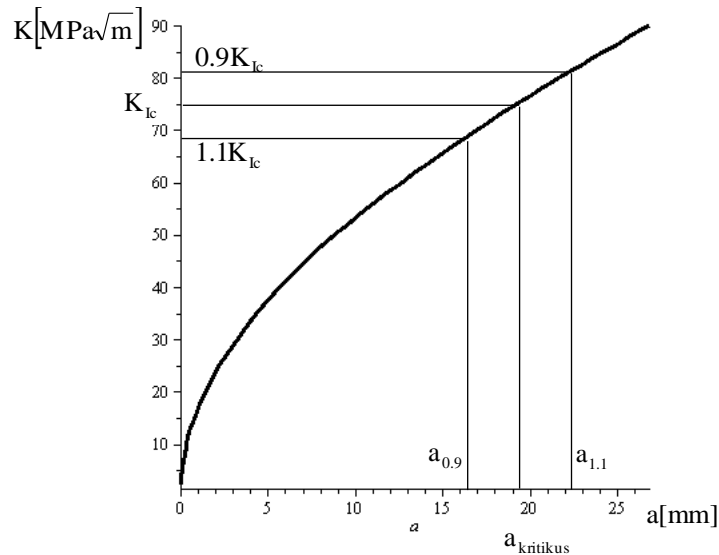
A szerkezet geometriájától, terhelésétől és a repedés elhelyezkedésétől, méretétől függően ez lehet néhány nap, de több év is. Ez adja a pontos számítás jelentőségét: ha a repedés veszélyes, akkor a gyors beavatkozással súlyos károkat lehet megelőzni, ha viszont nem az, akkor a tovább üzemeltetéssel lehet a fölösleges veszteségeket elkerülni. A ciklusokban kifejezett élettartam egy, a K - a függvényt is tartalmazó függvény integrálásával számítható.

A 2. pontban leírt törésmechanikai feladat (belső nyomással és hajlítással terhelt cső átmenő repedéssel) megoldásának lépéseit mutatjuk be.

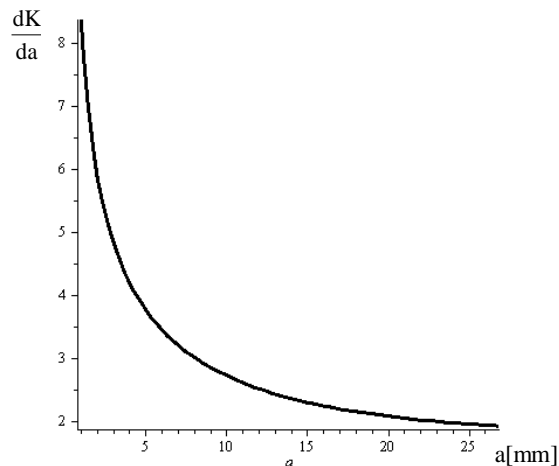
Az 1. táblázatban megadott formulák alapján képzett K - a függvénnyel kell a számításokat elvégezni. Az elsődleges feladat a kritikus repedéshossz kiszámítása, és ennek összevetése feltárt repedés hosszával. Ez a $K(a)=K_{Ic}$ egyenlet megoldását jelenti. A vizsgált esetben a

megoldás: $a_{kritikus}=19,31mm$. Az értékek áttekintéséhez nagy segítséget jelent a K függvény grafikonja (5. ábra).

Mivel a törési szívósságot kísérleti úton állapítják meg, a használt érték a tényleges érték becslése. Ezért az aktuális repedéshossz megítélésékor célszerű figyelembe venni a törési szívósságra rendelkezésre álló érték bizonytalanságát például úgy, hogy meghatározzuk a kritikus repedéshosszat a $0,9 \cdot K_{Ic}$ és $1,1 \cdot K_{Ic}$ törési szívósság értékek mellett. A példánkban ezek $a_{0,9}=15,84mm$, illetve $a_{1,1}=22,99mm$. A kapott eredmények úgy fogalmazhatók meg, hogy a törési szívósság értékének $\pm 10\%$ -os bizonytalanságát feltételezve a kritikus repedéshossz értéke a $15,84mm - 22,99mm$ tartományban van (5. ábra).



5. ábra. Feszültségintenzitási tényező – repedéshossz függvény (K - a)



6. ábra. Repedésérzékenységi index (a K - a függvény deriváltja)

Az állapot megítélésének alapja a repedés hosszának mérése. A mérés pontossága függ a mérés körülményeitől, az emberi tényezőktől és a mérőeszköz pontosságától. Az, hogy a vizsgálat során milyen mérési pontosságot kell biztosítani, a kvázistatikus repedésérzékenységi indextől függ [5-7], ami a K - a függvény differenciálhányadosa („meredeksége”). Ha a repedésérzékenységi index értéke nagy, akkor kis mérési pontatlanság

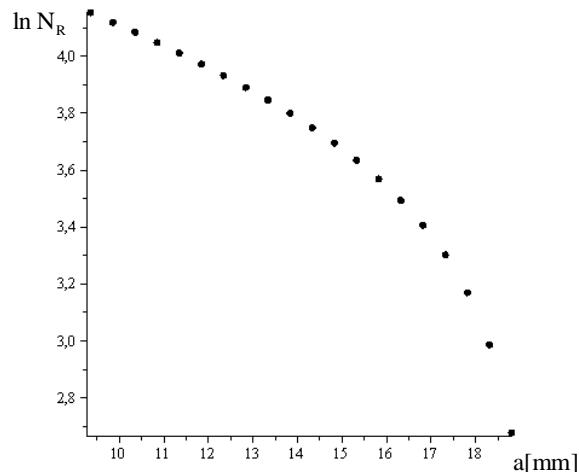
is jelentősen befolyásolja a feszültségintenzitási tényező számított értékét, így a megállapítás megbízhatósága jelentősen csökken, míg kis repedésérzékenységi index érték esetén a mérési pontosságnak kisebb a jelentősége. A – repedéshossztól függő – repedésérzékenységi indexet a dK/da derivált függvény adja (6. ábra).

A szerkezetek terhelése általában nem statikus, például egy olaj- vagy gázvezeték esetén az üzemi nyomás változik a szállított mennyiség (tömegáram) függvényében. A repedések viselkedése a dinamikus (ciklikus) terhelés esetén különbözik a kvázistatikus esettől. Míg a kvázistatikus esetben azt feltételezzük, hogy a repedés csak akkor „indul el”, ha fennáll a $K > K_{Ic}$ feltétel, és akkor rögtön instabil módon terjed, addig a ciklikus terhelésnél (egy bizonyos K érték, illetve az ennek megfelelő a_{th} repedéshossz felett) minden egyes ciklusban történik kis mértékű, stabil repedésterjedés. Ez a folyamat addig tart, míg a repedéshossz el nem éri a kritikus értéket, amikor elindul az instabil repedésterjedés. A szerkezet ciklikus terhelés esetén a – ciklusszában kifejezett – maradék élettartammal jellemezhető (N_R). Lükettető-húzó terhelés esetén

$$N_R(a_I) = \int_{a_I}^{a_{kritikus}} \frac{I}{C \cdot K(a)^n} da$$

ahol C és n az anyagtól és a geometriától függő jellemzők.

A 7. ábra a számított maradék élettartamokat mutatja logaritmikus skálán.



7. ábra. A ciklusszámban kifejezett élettartam (N_R) logaritmus a repedéshossz (a) függvényében

A projekt feladat része annak elemzése is, hogy csökkentett terhelés mellett hogyan változik a maradék élettartam, ezért a számolásokat újra el kell végezni az új (csökkentett) értékekkel is. Amennyiben a maradék élettartam értéke indokolja, ezen eredmények alapján lehet javaslatot tenni az üzemeltetési körülmények megváltoztatására.

5. Összefoglalás

A törésmechanikai példával illusztrált fejtegetés kapcsán a következő megállapítások tehetők:

1. A matematikai ismeretek mindennapi használatát természetes generációs problémák korlátozzák, gátolva ezzel a jelenségek, műszaki problémák megalapozott elemzését.
2. A műszaki-természettudományos területeken az oktatást úgy kell átalakítani, hogy a matematika a hozzáértők privilégiumából az alkalmazók eszköze legyen.

3. A matematika oktatásába az elvek ismertetése mellett be kell vezetni a szoftverrendszerek készségszintű használatát és ezek elsajátításáról az oktatás, vizsga keretében meg kell győződni (a „licenc” megszerzése kötelező, a vizsga előfeltétele).
4. A műszaki tárgyak széles körében az „alkalmazásszintű matematikára alapozott, egyéni feladat orientált” képzési tematikát célszerű kidolgozni és bevezetni.

Irodalomjegyzék

- [1] R. G. Forman, J. C. Hickman és V. Shivakumar: Stress-Intensity Factors for Circumferential Through Cracks in Hollow Cylinders Subjected to Combined Tension and Bending Loads, *Engineering Fracture Mechanics* **21**, No.3 (1985), 563-571
- [2] H. Tada és P.C. Paris és G.R. Irwin G.R., The Stress Analysis of Cracks Handbook. *Del Research Corp.*, Hellertown, Pa., U.S.A. 1973.
- [3] Rooke, D. P., Cartwright, D. J., Compendium of Stress Intensity Factors. *Her Majesty's Stationery Office*, London. 1976.
- [4] Stress Intensity Factors Handbook. Edited by Y. Murakami, *Pergamon Press*, 1987.
- [5] Szűcs, N., Tóth, L., Szávai, Sz.: Repedés terjedési érzékenység index, *Anyagvizsgálók lapja*, 2008/2, 66-71.
- [6] Tóth, L.: A computer aided assessment system of reliability cyclic loaded construction elements having flaws. *Proc. 1st Int. Conf. on Computer-Aided Assessment and control of Localized Damage*. Portsmouth, UK., Springer-Verlag, 1990, Vol.1. 39-53.
- [7] Tóth, L.: Reliability Assessment of Cracked Structural Elements under Cyclic Loading. in *Handbook of Fatigue Crack Propagation in Metallic Structures*. Edited By A. Carpinteri. Elsevier. 1994. 1643-1683.

OPNET® eszközök a hálózatmodellezés oktatásában

OPNET® Tools in Education of Network Modeling

Kuki Attila^a

^aDebreceni Egyetem, Informatikai Kar

kuki@inf.unideb.hu

Absztrakt: A tervezett előadásban bemutatnánk a Debreceni Egyetem Informatikai Karán a sztochasztikus szimulációval kapcsolatos kurzusok gyakorlati támogatásaként létrehozott hálózatmodellezési tárgyak oktatásában használt eszközöket. Számítógépes hálózati környezetek modellezése akkor hasznos igazán, amikor a drága eszközök beszerzése előtt fel lehet mérni a tervezett hálózat működési jellemzőit, és az esetleges szűk keresztmetszeteket időben lokalizálhatók. Egy ilyen eszköz egyetemi verzióját mutatnánk, be, az IT Gurut. Forgatókönyv-kezelésével alkalmas arra, hogy nagyobb méretű projekteket alrendszerre bonthassunk vagy „what if” típusú elemzéseket végezhessünk. Modellcsaládjai révén lehetőség van adott gyártók eszközeinek modellezésére is. Az oktatásban is hatékonyan alkalmazható, segítségével az OSI ISO modell minden rétegéhez szemléletes modelleket lehet bemutatni, illetve tanulmányozhatóak az ismert protokollok működési elvei is (pl. RIP, OSPF, TCP, RSVP). Példaként bemutatjuk, hogy az útválasztók a beérkező csomagoknál alkalmazott sorkezelési elveik (FIFO, PQ, WFQ) milyen hatással vannak a legfontosabb teljesítménymutatókra.

Kulcsszavak: modellezés, hálózatok, szimuláció

Abstract: This paper focuses on a set of modeling tools concerning real network communication problems. OPNET offers two free license tools for academic use. The IT Guru can be used with an individual license while the Modeler with an organizational license. In this paper the use of IT Guru will be presented in education and in research, as well. This tool can be used for protocols and problems in each OSI ISO layers. Equipments from different manufacturers can be investigated. The range and the topology of the network under consideration have also a lots of options to choose. In the modeling process numerous individual and global parameters and statistics can be chosen, the results provided by a discrete event simulation can be displayed as a scalar or a vector panel, there is a lot of output options, and the graphs of different scenarios can be compared easily. An example is provided on the effect of different queueing disciplines of routers.

Keywords: modeling, networks, simulation

1. Bevezetés

Az információtechnológiai szakemberek számára kiemelt jelentőséggel bír a valós rendszerek, például a számítógépes hálózatok viselkedésének és teljesítménymutatóinak modellezéssel történő vizsgálata. Egy számítógép-architektúra tervezése során fontos annak ismerete, hogy eleget tesz-e a rendszer a követelményeknek, illetve megfelelő teljesítménnyel működik-e. A költséges eszközök üzembe állítása előtt hasznosak lehetnek az előzetes információk a rendszer működésének optimális paramétereiről. Egy kiegyensúlyozott szimulációs környezet kialakítása azonban nem triviális feladat. A terület alapos ismerete és a modellezési és szimulációs fogalmak mély ismerete szükségeltetik hozzá.

Az OPNET® Technologies Inc. céget 1986-ban hozták létre, gyorsan a piacvezetők közé került, majd a tőzsdén is megjelent. 2012-ben felvásárolta az egyik piaci konkurenciája, a Riverbed. Az OPNET® többféle megoldást kínál a számítógépes hálózatok modellezésére, elemzésére. A legnépszerűbb eszközök között lehet megemlíteni a Modelert és annak egyszerűsített változatát, az IT Guru Academic Editiont. Széles körben alkalmazzák oktatási,

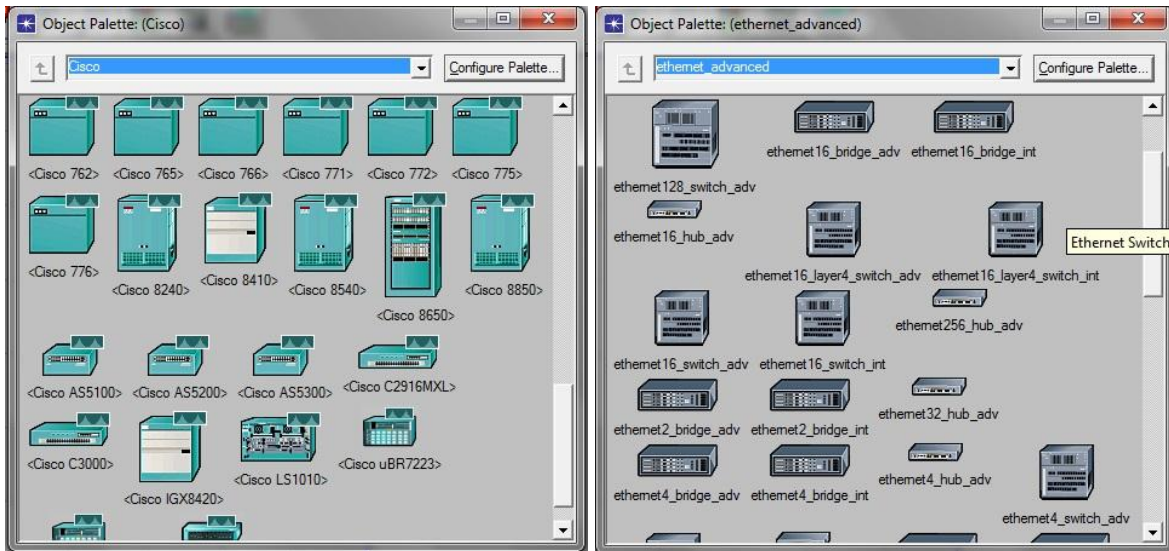
kormányzati és ipari szereplők a különféle valós hálózati architektúrák modellezésére, vizsgálatára, teljesítményjellemzőik becslésére, valamint a felsőoktatásban a számítógépes hálózatok és az infokommunikáció alap és haladó területeinek oktatására, kutatására. Oktatási környezetben az oktatók és a hallgatók az IT Guru Academic Edition eszközre ingyenes licencet igényelhetnek.



1. ábra. OPNET® IT Guru Academic Edition

Az OPNET® IT Guru Academic Edition egyszerűen használható felhasználói felületet biztosít. A szoftver eszközeinek segítségével könnyedén fel lehet építeni különböző hálózati topológiákat és konfigurációkat, és vizsgálni lehet azok rendszerjellemzőit, teljesítményét, valamint hatékonyságát. Az IT Guru számos modellcsaládot tartalmaz, amelyek segítségével szimulálhatóak a különféle valós hardverkörnyezetek és az OSI ISO modell protokolljai. Ezek részletes leírásai megtalálhatók például itt: [1], [2], [3]. Ilyen modell- és technológiacsaládok többek között a

- 3COM,
- Ascend,
- Atm,
- CISCO,
- Ethernet,
- Hewlett Packard,
- Internet Toolbox
- Juniper,
- NEC,
- Nortel,
- xDSL.



2. ábra. Különböző modellcsaládok

2. OSI ISO rétegek

A felsőoktatásban a hálózati kurzusok oktatói keresik és kedvelik azokat az eszközöket, amelyek segítségével hatékonyabban bevonhatják a hallgatókat a közös labormunkába, illetve jobban felkelthetik az érdeklődésüket a téma iránt. Ezek az eszközök az oktatott ismeretanyag mélyebb elsajátítását is lehetővé tehetik

Különösen hasznos, hogy az IT Guru eszközhöz számos előre elkészített laboratóriumi feladatsor és modell érhető el, melyeket az eszközt használó intézmények oktatói, esetleg hallgatói tesznek közzé. Ezen modellek alapján álljon itt néhány tipikus példa a hibrid hálózati modell rétegeinek megfelelően.

Fizikai és Adatkapcsolati réteg Ezekhez a rétegekhez jó néhány komplex feladat található. Az Ethernet szabványokat a jól ismert busz topológia szerint vizsgálják. A feladat a különböző intenzitások mellett generált csomagok és a fogadott csomagok számának ábrázolása skalár panel segítségével. Ez rögtön bevezet minket a haladó eredménykiértékelés világába. További forráskönyvek (lásd lentebb) segítségével vizsgálhatjuk az egyéb paraméterek hatásait (csomópontok száma, keretek mérete stb.). Hasonló feladatokat lehet megoldani a token ringekre is. Itt például a THT (Token Holding Time) paraméter kerülhet a figyelem középpontjába, illetve lehet vizsgálni a kiegyensúlyozott és a kiegyensúlyozatlan forgalomgenerálás eltéréseit. A vezeték nélküli világ is modellezhető ezen eszközök segítségével.

Hálózati réteg A legtöbb lehetőség itt ezen réteg tanulmányozására adódik. Többek között modellek építhetők, illetve található az IP-címek kezelésére, az IP protokoll haladó jellemzőire, a NAT (Network Address Translation) bemutatására, az IPv6 kérdéseire, a Quality of Service metrikáira, a különböző útválasztási protokollok működésének elemzésére (RIP, OSPF).

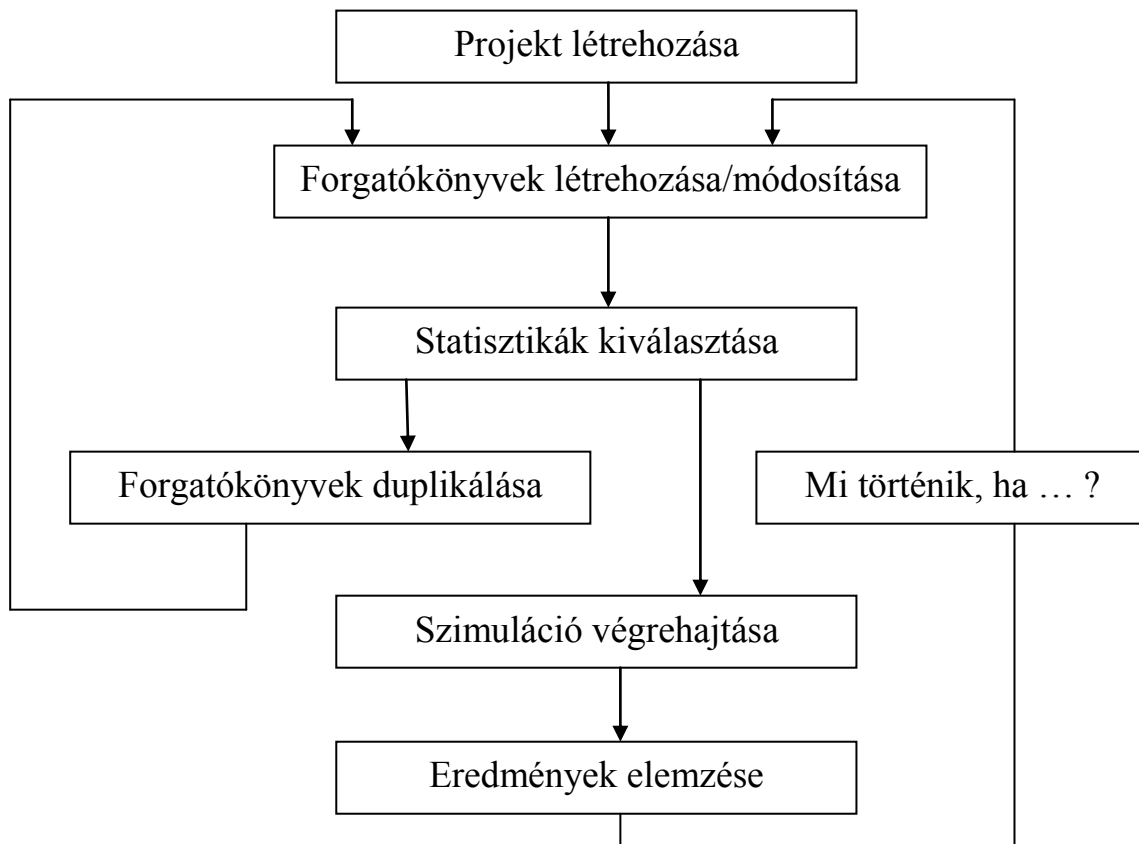
Szállítási réteg Kompakt modelleken keresztül vizsgálhatók a TCP és az UDP szállítási protokollok jellemzői. Például a TCP esetében igen szemléletesen mutatható be a torlódáskezelés, illetve a különféle algoritmusok (Reno, Fast Retransmit) működése.

Alkalmazási réteg Az OPNET® lehetővé teszi, hogy alkalmazásokat is szerepeltessünk a szimulációs modellekben. Egy különleges csomópont, az *Applications* írja le a használni kívánt alkalmazásokat, azok működési jellemzőit (explicit forgalom, háttérforgalom, hibrid forgalom). A szokásos alkalmazások (FTP, E-mail, Telnet, Web-böngészés (HTTP), videokonferencia, nyomtatás stb.) mellett saját alkalmazások definiálására is nyílik lehetőség. Az alkalmazások szimuláció alatti működését felhasználói profilok segítségével vezérelhetjük (a *Profiles* csomóponton).

3. Harmadik fejezet

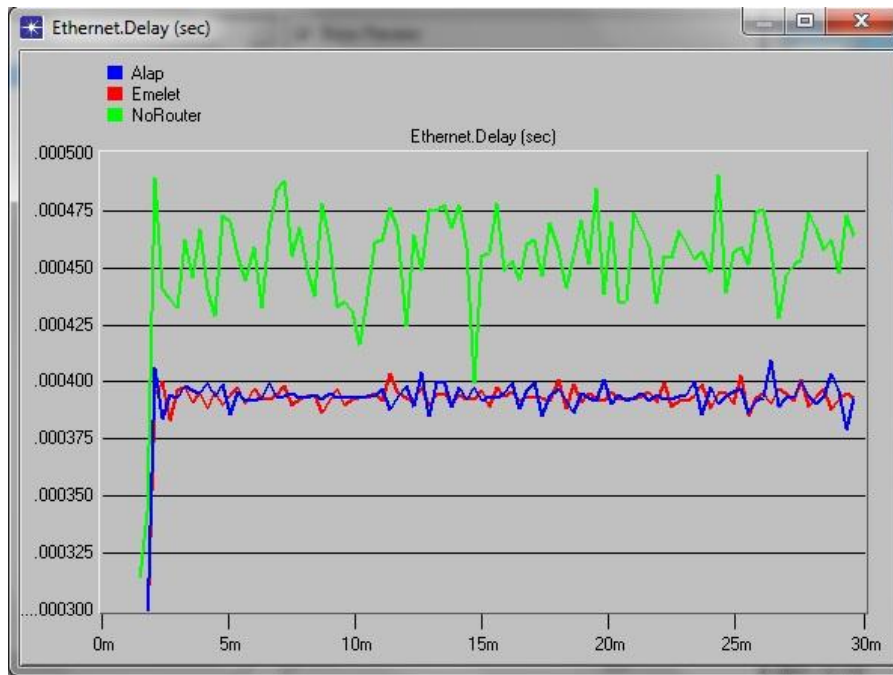
Az OPNET® környezetben különféle modellezési tevékenység végezhető. Általában vannak tipikus feladatok, amelyeket szinte minden esetben el kell végezni. Ilyenek például a modell felépítése, forgatókönyvek definiálása, a modell futtatása, az eredmények megtekintése és elemzése. Egy általános modellezési folyamat a következők szerint képzelhető el:

1. lépés: A hálózat topológiájának létrehozása,
2. lépés: Forgatókönyvek létrehozása,
3. lépés: Statisztikák kiválasztása,
4. lépés: A szimuláció végrehajtása,
5. lépés: Az eredmények elemzése,
6. lépés: Szükség esetén visszatérés a 2. lépéshez.



3. ábra. Modellezési folyamat

A forgatókönyvek segítségével egy nagyméretű szimulációs modell kisebb részekre darabolható, ahol lehetőség nyílik a tekintett terület adott szempont vagy konfiguráció szerinti vizsgálatára. Abban az esetben is nagyon hatékonyak a forgatókönyvek, ha a szimulációt kis módosításokkal többször végre szeretnénk hajtani, majd az eredményeket össze kívánjuk hasonlítani. Ekkor másolatot készíthetünk a tekintett forgatókönyvről, az új példányban végrehajtjuk a kívánt módosításokat (például csomaggenerálási gyakoriság egy csomóponton), majd futtatjuk a szimulációt (a már lefuttatott részeket nem szükséges ismételtén végrehajtani). Az eredmények különböző formában összevethetők.



4. ábra. Átfedő ábra

Az OPNET® a szimuláció során összegyűjthető statisztikák széles körét kínálja fel. Ezek a statisztikák azonban a szimuláció végrehajtásakor nem lesznek automatikusan kiválasztva. A szimuláció statisztikák kiválasztása nélkül is végrehajtható, tehát a felhasználó feladata, hogy a vizsgálni kívánt paramétereket (statisztikákat) a megfelelő módon kijelölje. A rendelkezésre álló statisztikák két csoportot képeznek:

Objektum statisztikák Ezek írják le a hálózat egy összetevőjének (csomópont, link, adatforgalom-igény stb.) viselkedését.

Globális statisztikák A teljes hálózatra vonatkoznak, általános hálózati paraméterek tartoznak ide. Ilyenek lehetnek például az Ethernet késleltetés vagy egy adott protokoll viselkedése. A szimuláció során a globális statisztikák a hálózat minden komponensére vonatkoznak.

4. Sorbaállítási elvek modellezése - példa

Ez a példa a forgalomirányítók sorkezelésére fókuszál (további részletekért lásd [4]). A beérkező csomagok a kiszolgálási értékeik alapján különböző sorokba (pufferekbe) irányíthatók. A forgalomirányítók többféle sorkezelési elv alapján képesek kezelni ezeket a

sorokat. Ebben a példában a FIFO (First In First Out), a PQ (Priority Queueing) és a WFQ (Weighted Fair Queueing) elveket fogjuk tekinteni.

FIFO – Ez a legegyszerűbb sorbanállási kiszolgálási elv. A beérkező csomagok egy sorban kerülnek elhelyezésre, és a feldolgozásuk a beérkezésük sorrendjében történik. Gyors, könnyen előre jelezhető kiszolgálási elv, jellemzőinek kiszámítására jól ismert módszerek vannak. Viszont hátrányi is vannak ennek a módszernek. Ez a kiszolgálási elv nem tudja megkülönböztetni a különböző típusú forrásból érkező csomagokat, mindegyiket ugyanolyan módon kezeli. Némely, az időre érzékeny csomagok (pl. interaktív hang) jelentős késleltetést szenvedhetnek.

PQ – Ez az elv egy természetes megoldást kínál a megkülönböztetett szolgáltatásra. A beérkező csomagok a saját prioritás értékük alapján különböző sorban kerülnek elhelyezésre. Adott időpillanatban a legmagasabb prioritással rendelkező csomag lesz kiszolgálva. A magas prioritású csomagok tömeges beérkezése azonban az alacsonyabb prioritású adatok jelentős késleltetését okozhatja ebben az esetben is.

WFQ – Ez a módszer egy áramlás-alapú algoritmus, mely a válaszidő csökkentésének érdekében a késleltetésre érzékeny adatokat a puffer elejére helyezi, a megmaradt sávszélességet meg igazságosan osztja meg a nagy sávszélesség-igényű adatfolyamok között.

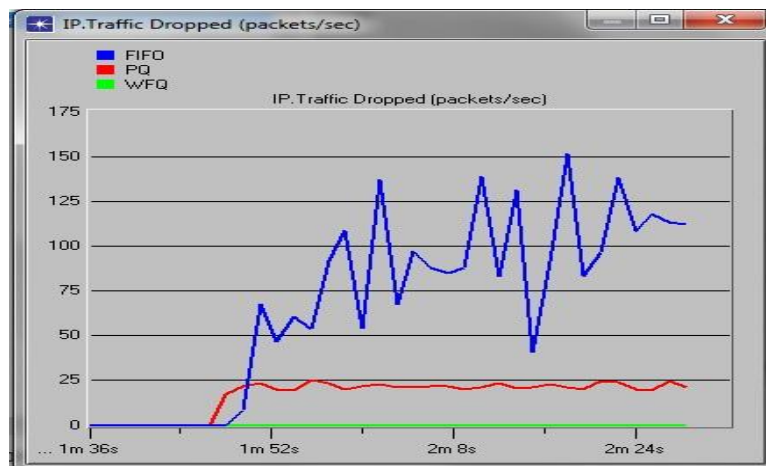
Az OPNET® az IP-fejrész következő ToS (Type of Service) bitjeit használja a prioritás értékeinek (a ToS alapú módszer helyett protokoll-alapú és portszám-alapú eljárás is alkalmazható):

- FTP: Best Effort (0),
- http: Standard (2),
- Video: Streaming Multimedia (4),
- VoIP: Interactive Voice (6).

A szimulációhoz a következő paraméterek (statisztikák) kerültek kiválasztásra:

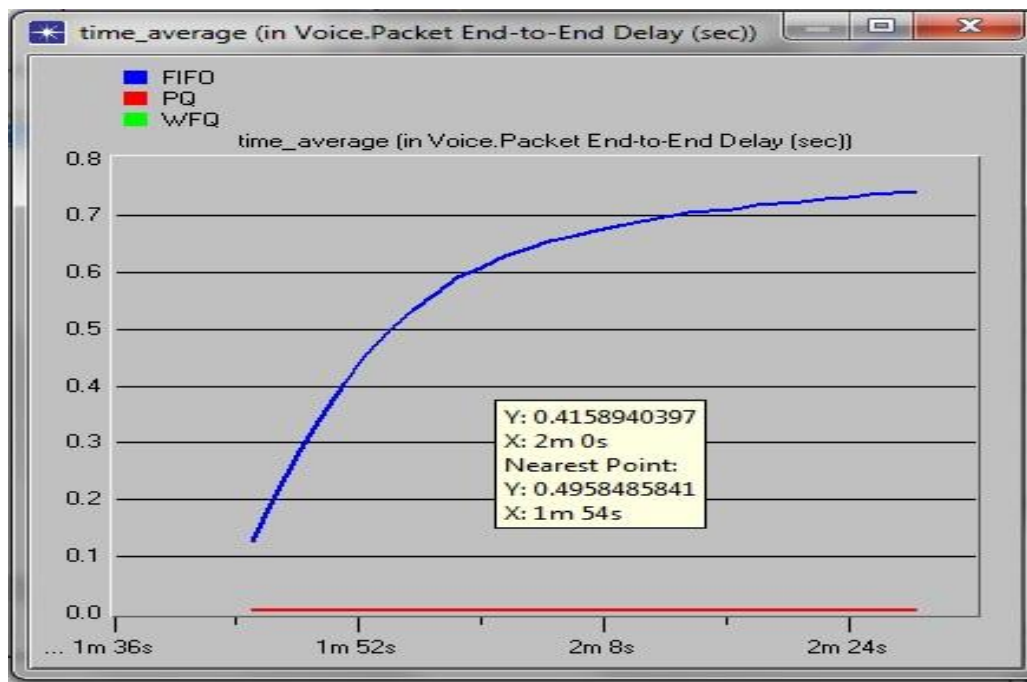
- Traffic Dropped (packets/sec),
- Traffic Received (bytes/sec),
- Packet Delay Variation,
- Packet End-to-End Delay (sec),
- Page Response Time (sec).

150 másodpercig tartó diszkrét esemény szimuláció (DES – Discrete Event Simulation) többek között a következőket eredményezte:



5. ábra. Az eldobott csomagok száma

Az 5. ábrán megfigyelhető konstans zérus szakasz annak az eredménye, hogy a szimuláció kezdete 100 másodpercre lett beállítva. Látható, hogy a vizsgált sorbaállítási módszerek közül a WFQ a leghatékonyabb. A 6. ábra megerősíti ezt az észrevételt.



6. ábra. End-to-End Delay a Voice csomagoknál

A zöld (világos színű) WFQ görbe egybeesik a vízszintes, kis értékű PQ görbével. Ebben az esetben ez a két sorbanállítási módszer ugyanazt az eredményt szolgáltatja.

5. Köszönetnyilvánítás

A publikációt a TÁMOP 4.2.2. C-11/KONV-2012-0001 projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Sethi, A., S., Hnatyshin, V., Y., The Practical OPNET® User Guide for Computer Networks, CRC Press, (2013).
- [2] Lu, Z., Yang, H., Unlocking the Power of OPNET Modeler, Cambridge University Press, (2012).
- [3] Adil, A., Malik, N., Modeling and Simulation of Computer Network Using OPNET, Lambert Academis Publishing, (2012).
- [4] Szilágyi, Sz., Almási, B., A Review of Congestion Management Algorithms on Cisco Routers, *Journal of Computer Science and Control Systems*, (University of Oradea Publisher, Oradea), **5**(1) (2012), 103-107, ISSN 1844 - 6043, University of Oradea Publisher, Oradea, (2012).

Logikai programozás oktatásának szemléltetése flash animációk segítségével

The demonstration of Logical Programming Education with the help of flash animations

Dr. Pántya Róbert

Károly Róbert Főiskola
rpantya@karolyrobert.hu

Absztrakt: Az oktatási tevékenységünk során nagyon fontos cél tanulóink problémamegoldó képességének fejlesztése. A logikai programozás és így a logikai programozási nyelvek oktatása nagyon hasznos eszköz ezen cél elérésében. A logikai nyelvek magas szintű nyelvek, az emberi gondolkodáshoz nagyon közel állnak. Az oktatásuk azonban nagyon sok kihívás elé állítja mind az oktatót, mind pedig a tanulót. Ebben az előadásban bemutatásra kerül egy kifejlesztett és kísérleti csoportokon tesztelt oktatási segédanyag, mely a Prolog logikai programozási nyelv oktatását tartalmazza. A tananyaghoz számos Flash animáció készült az egyes tananyagrészek szemléltetésére. Interaktív animáció ismerteti meg a logikai műveletekkel (és, vagy, kizáró vagy, tagadás, implikáció, ekvivalencia), a Prologban alkalmazott feladatmegoldási metódussal: a mintaillesztéssel és visszalépéssel, a rekurzió-szervezés sajátjaival, az összetett kifejezések, listák használatával, stb. Ezek az animációk lépésről-lépésre vezetik végig a tanulót az adott feladat megoldásán, nagy előnyük, hogy bármikor megállíthatóak, valamint többször is lejátszhatóak. Ezekre túl animációk készültek az ellenőrző tesztekhez is. A tananyag segédanyagai, az elkészült animációk, tesztek a Károly Róbert Főiskola e-learning rendszerében kerültek elhelyezésre, melynek során egy modern, korszerű oktatási tananyag vált elérhetővé tanulóink számára.

Kulcsszavak: Prolog, logikai programozás, Flash animáció, problémamegoldó gondolkodás

Abstract: During educational activity it is a very important aim to develop the students' problem solving ability. The logical programming and logical programme languages are very useful tools to achieve this aim. Logical languages are of high level they are close to human thinking. However, its education challenges both the lecturers and the students. This presentation introduces a self-developed educational material (based on Prolog logical programme language) and this course material has already been tested on experimental groups. Flash animations were also produced for certain educational parts. Interactive animations are used to introduce logical operations (and, or, exclusive or, negation, implication, equivalence) by the exercise-solving solutions used also in Prolog: sample-fitting and backtracking, recursion and its specialities, complex expressions, lists etc. These animations provide a step-by-step procedure throughout the method of solving an example. Their great advantage is that can be stopped any time and they can also be played several times. Furthermore, self-monitoring tests were also created and these tests are available in the Moodle framework that belongs to Karoly Robert College thus creating a modern, up-to-date course material for our students.

Keywords: Prolog, logical programming, Flash animation, problem solving way of thinking

1. Bevezetés

A logikai programozásnak, és így a Prolog-nak, az oktatása jótékony hatással van a tanulók probléma-megoldó képességének a javítására, melyet nagyon jól igazolt az az összetett kétcsoportos pedagógiai kísérlet, mely három gyöngyösi oktatási intézményben (Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola, József Attila Szakközépiskola, Szakiskola és Kollégium, Károly Róbert Főiskola) került megvalósításra 2011 őszén. [1]

A Károly Róbert Főiskolán a Logikai programozás tárgyat szabadon választható C tárgyként hallgathatják gazdaságinformatikus, valamint informatikus és szakigazgatási agrármérnök szakos tanulóink. A kurzushoz elektronikus tananyag készült, melyet a főiskola e-learning rendszerében érhetnek el hallgatóink. [2]

A Logikai programozás tárgy az alábbi témakörökkel ismerteti meg a hallgatót:

- Logikai műveletek,
- Egyszerű programok készítése Prologban,
- Mintaillesztés és visszalépés,
- Rekurzív függvény-hívások Prologban,
- Vágás (cut) művelet,
- Tagadás,
- Ciklusszervezés rekurzióval,
- Listák és összetett kifejezések kezelése Prologban,
- Adatbáziskezelés és Prolog.

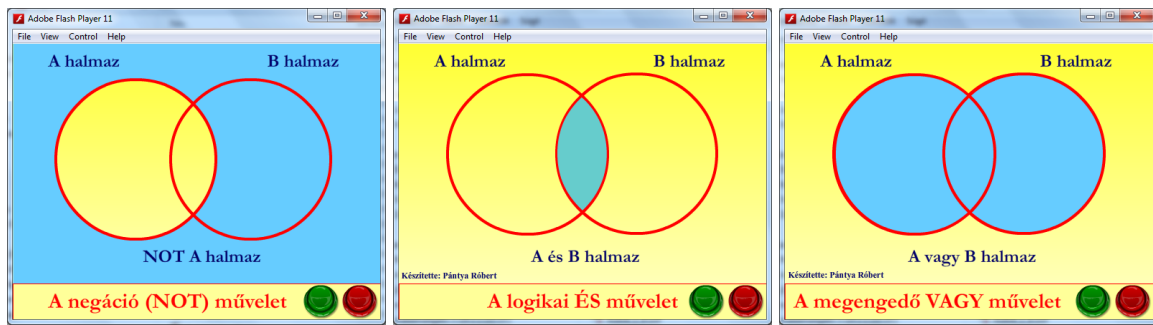
A kurzus a Prolog nyelv elméleti tananyagaként Márkus Zs. [3] munkájára épít, valamint felhasználja Makány Gy. [4] és Máday Á. [5] munkáit a gyakorlati feladatok forrásaként. Ebben az előadásban a Logikai programozás e-learning oktatási tananyaghoz készült közel negyven darab szemléltető animáció közül szeretnék néhányat részletesen bemutatni.

2. Animációk használata a tananyag szemléltetésére

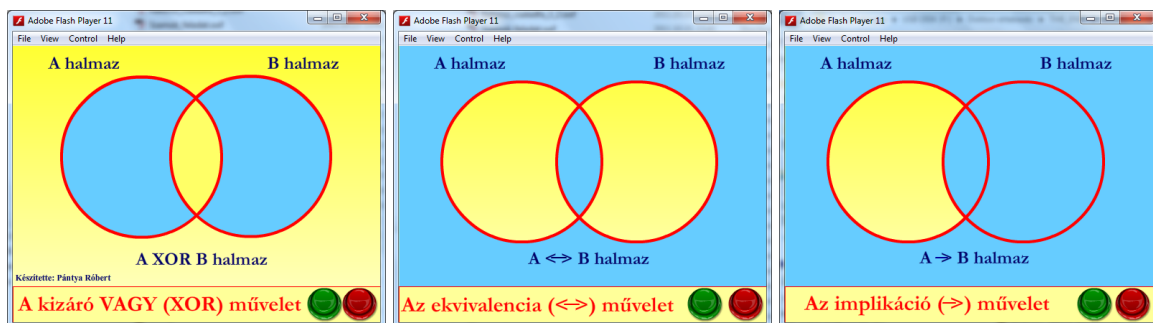
Az animációk Adobe Flash, valamint Adobe Captivate programok segítségével készültek. Minden egyes animáció interaktív, tehát a hallgató aktív közreműködését igényli. Egy piros gombbal bármikor leállítható, valamint egy zöld gombbal tovább folytatható a lefutása az animációnak, ezzel minden tanuló saját magának, saját befogadó-képességének megfelelően tudja ütemezni a szemléltető eszköz magyarázatának a sebességét. Az animációk elérhetőek a Károly Róbert Főiskola e-learning rendszerében [2]. A lejátszásukhoz ajánlott Adobe Flash Player megléte, de ennek hiányában sincs semmi gond, bármelyik böngészőben megtekinthető, futtatható. A következő alfejezetekben bemutatok bizonyos témakörökhöz kapcsolódóan ilyen animációkat.

Logikai műveletek

A Logikai programozás tantárgy bevezető témaköre a legfontosabb logikai műveletekkel (NOT, ÉS, VAGY, KIZÁRÓ VAGY, IMPLIKÁCIÓ, EKVIVALENCIA) ismerteti meg a hallgatót. Ebben a részben nemcsak a három alapvető logikai művelettel (NOT, ÉS, VAGY), hanem további három nagyon fontos művelettel foglalkozunk részletesen. Oktatási tapasztalatom szerint érdemes már a logikai programozás kurzus elején felhívni a hallgatók figyelmét a VAGY (megengedő), valamint a KIZÁRÓ VAGY közötti különbségre. Az IMPLIKÁCIÓ pontos ismerete pedig a Prolog feladatmegoldó mechanizmusa miatt elengedhetetlen. A következő ábrákon (1. és 2. ábra) az egyes animációk halmazműveletek segítségével mutatják be ezeket a logikai műveleteket, megfelelően vizualizálva az alapvető ismereteket.



1. ábra. A NEGÁCIÓ, az ÉS, valamint a VAGY (megengedő) logikai műveletek



2. ábra. A KIZÁRÓ VAGY, az EKVIVALENCIA, valamint az IMPLIKÁCIÓ logikai műveletek

Mintaillesztés és visszalépés

A kurzus ezen fejezetében a Prolog lényegi működési mechanizmusával és vezérlési stratégiájával, a mintaillesztéssel és visszalépéssel, ismerkedhetnek meg tanulóim.

Ez is egy különösen fontos rész, ezért nagyon sok gyakorló feladat, s ezekhez szemléltető animáció segíti az ismeretek befogadását.

Az Autók feladat bemutatásához készült animációt mutatom be részletesen. Ebben a feladatban a megvásárolandó autó kiválasztása a cél, melynek eldöntésénél figyelembe kell venni, hogy mindkét személynek (Jutka, Robi) tetszen az autó, valamint legyen rá pénzük is.

A feladat Prologban megfogalmazva a következő tényekkel és szabállyal valósítható meg:

```

robinak_tetszik(ford_t).
robinak_tetszik(ferrari).
robinak_tetszik(skoda_fabia).
robinak_tetszik(nissan_note).

jutkanak_tetszik(ferrari).
jutkanak_tetszik(nissan_note).
jutkanak_tetszik(toyota_hibrid).
jutkanak_tetszik(smart).
    
```

```
van_rá_péNZ(nissan_note).
van_rá_péNZ(skoda_fabia).
van_rá_péNZ(smart).
van_rá_péNZ(trabant).
van_rá_péNZ(wartburg).
```

```
vesznek_autót(X):- robinak_tetszik(X), jutkának_tetszik(X), van_rá_péNZ(X).
```

Fontos megjegyezni, hogy a Prolog kifejezések argumentumaiban a változókat nagybetűvel kell kezdeni, míg a konstansokat kisbetűvel. A konstansok megadhatóak egyszeres aposztrófok között is.

A feladathoz készült animáció halmazokba sorolja az adott szempontoknak megfelelő autókat, mellyel nagyon egyszerűen szemlélteti a logikai ÉS műveletet. A 3.ábrán található animáció emellett a `vesznek_autót` szabályt is szemlélteti, valamint az ezzel kapcsolatos célállítást megoldásának mechanizmusát.



3. ábra. A mintaillesztés és visszalépés szemléltetése

Rekurzió a Prologban

A rekurzió is egy kiemelt jelentőségű témakör a logikai programozás terén, ugyanis a ciklusszervezés alapvető eszköze.

Egy nagyon népszerű feladatot és szemléltetését mutatok be a következőkben.

A Rákóczi-családfa programban Prolog tények formájában tároljuk a személyek adatait (név, születési év, elhalálozási év – személy tényállítást), valamint az anya-gyermek (anya tényállítást) és a férj-feleség (házastárs tényállítást) kapcsolatokat a személyek között.

A következő programrészlet mutatja be ezeket a tényállításokat.

```
személy('Zrínyi Ilona', 1643, 1703).  
személy('Zrínyi Péter', 1621, 1671).  
személy('Frangepán Katalin', 1625, 1673).  
személy('Rákóczi Julianna', 1672, 1717).  
személy('II.Rákóczi Ferenc', 1676, 1735).  
személy('I.Rákóczi Ferenc', 1645, 1676).  
személy('II.Rákóczi György', 1621, 1660).  
személy('Báthory Zsófia', 1629, 1680).  
személy('Sarolta Amália', 1679, 1722).  
személy('Rákóczi Lipót', 1696, 1699).  
személy('Rákóczi József', 1700, 1738).  
személy('Rákóczi György', 1701, 1756).
```

```
anya('Zrínyi Ilona', 'Rákóczi Julianna').  
anya('Zrínyi Ilona', 'II.Rákóczi Ferenc').  
anya('Sarolta Amália', 'Rákóczi Lipót').  
anya('Sarolta Amália', 'Rákóczi József').  
anya('Sarolta Amália', 'Rákóczi György').  
anya('Báthory Zsófia', 'I.Rákóczi Ferenc').  
anya('Frangepán Katalin', 'Zrínyi Ilona').
```

```
házastárs('Zrínyi Péter', 'Frangepán Katalin').  
házastárs('I.Rákóczi Ferenc', 'Zrínyi Ilona').  
házastárs('II.Rákóczi György', 'Báthory Zsófia').  
házastárs('II.Rákóczi Ferenc', 'Sarolta Amália').
```

Ebben a feladatban szabályok segítenek megfogalmazni a különféle rokonsági viszonyokat (apa, anyós, após, utód - leszármazott, ős).

```
apa(Apu, Gyerek) :- házastárs(Apu, Anyu), anya(Anyu, Gyerek).
```

```
anyós(Meny, Anyós) :- házastárs(X, Meny), anya(Anyós, X).
```

```
anyós(Vő, Anyós) :- házastárs(Vő, X), anya(Anyós, X).
```

```
após(Meny, Após) :- házastárs(X, Meny), apa(Após, X).
```

```
após(Vő, Após) :- házastárs(Vő, X), apa(Após, X).
```

```
utód(X, Ős) :- anya(Ős, X).
```

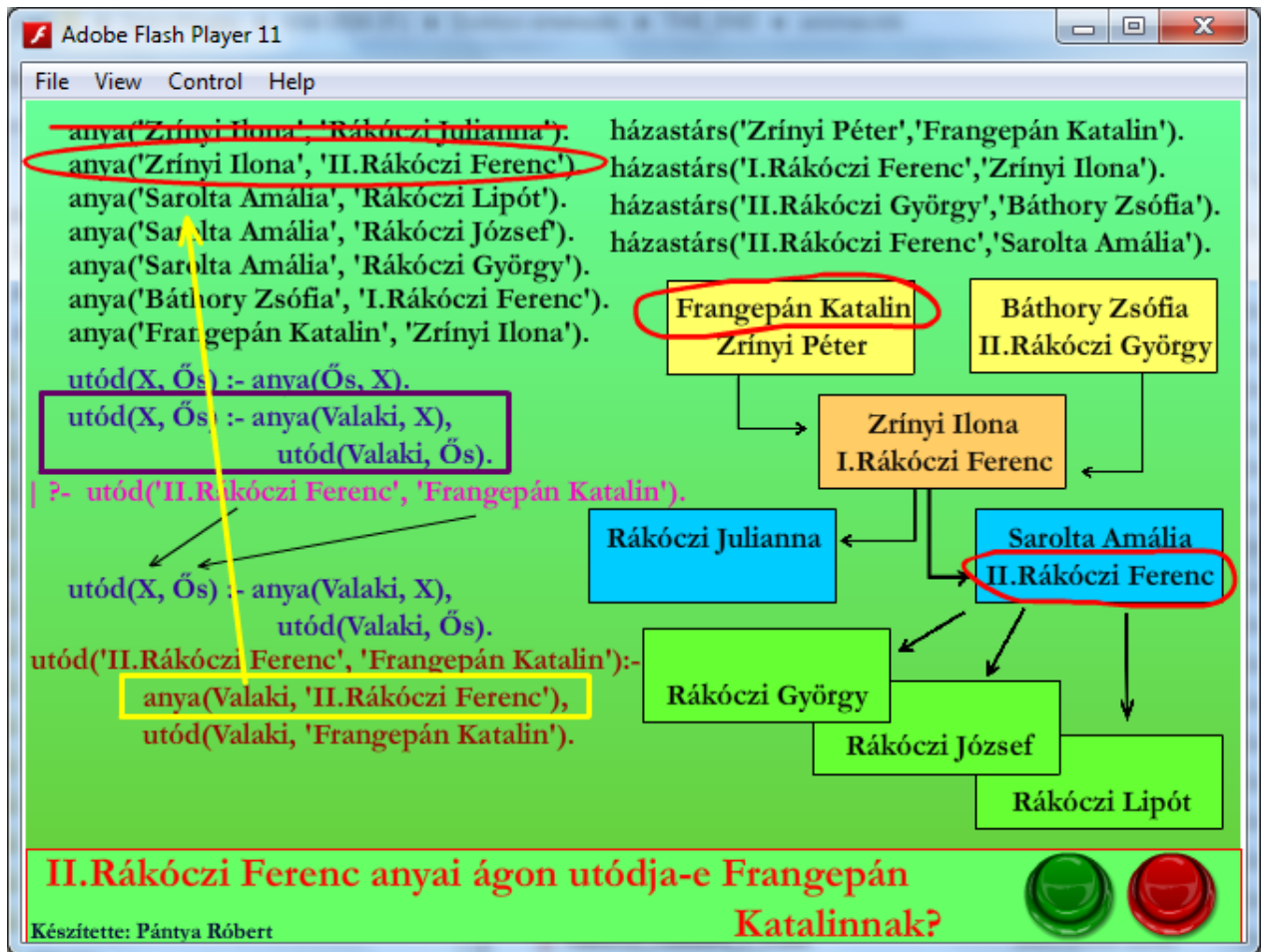
```
utód(X, Ős) :- anya(Valaki, X), utód(Valaki, Ős).
```

```
utód2(X, Ős) :- apa(Ős, X).
```

```
utód2(X, Ős) :- apa(Valaki, X), utód2(Valaki, Ős).
```

Az `utód` és `utód2` szabályok tartalmazznak rekurzív hívásokat, mellyel az apai, valamint az anyai ági leszármazást vizsgálhatjuk.

A 4.ábrán láthatjuk ennek a feladatnak a szemléltetését, melynek során részletesen bemutatásra kerül a feladatmegoldó algoritmus.



4. ábra. A rekurzió szemléltetése

Listák használata

Az összetett kifejezések közül az egyik legfontosabb a lista a Prologban, ezért ennek a kezelésével mindenképpen meg kell ismertetni a tanulókat.

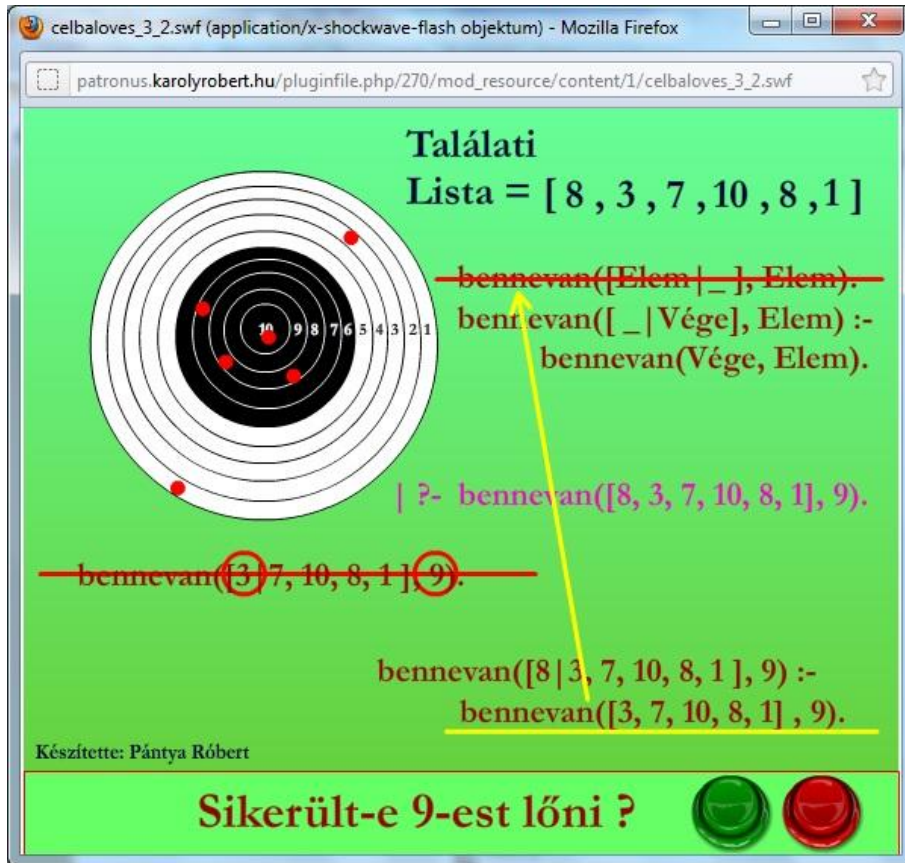
A listakezelés szabályainak bemutatására különösen alkalmasnak bizonyult a Célbalövés feladat, amelynek során a találatokat egy listába gyűjtjük össze.

Az 5.ábrán bemutatott feladat során azt kell eldönteni, hogy sikerült-e 9-est lőni (eldöntés programozási tétel használata).

Ezt a következő kódsor segítségével adhatjuk meg:

```

bennevan([Elem|_], Elem).
bennevan([_|Vége], Elem) :- bennevan(Vége, Elem).
    
```



5. ábra. Listák kezelésének szemléltetése

Adatbáziskezelés és Prolog

A kurzus ezen része az adatbáziskezelés és a Prolog kapcsolatáról szól. A Prolog deklaratív nyelv, csakúgy, mint a relációs adatbázisok nyelve, az SQL. Egy minta-adatbázis segítségével megmutatom, hogyan kell létrehozni Prolog-ban ezt az adatbázist, illetve a lekérdezéseket hogyan valósíthatjuk meg Prolog-ban.

A minta-adatbázisban településekről tárolunk adatokat (településkód, településnév, terület km²-ben, népesség, kistérségek kód, oktatáskód). Minden települést egy kistérségbe sorolunk be, s minden településhez megadjuk az adott településen megvalósuló legmagasabb oktatási formát. Tehát összesen három táblában tároljuk az adatokat úgy, hogy a település táblában szereplő kistérségek kód, valamint az oktatáskód idegen kulcsként funkcionál (6.ábra).

Település tábla

településkód	településnév	terület	népesség	kistérségkód	oktatáskód
1	Eger	104,77	58976	k1	o4
2	Lőrinci	33,29	5968	k4	o3
3	Heves	99,21	11450	k5	o3
4	Bélapátfalva	36,17	3391	k1	o2
5	Mónosbél	14,24	399	k6	o1
6	Gyöngyös				
7	Abasár				
8	Füzesabony				
9	Adebrő				

Kistérség tábla

kistérségkód	kistérségnév
k1	egri
k2	füzesabonyi
k3	gyöngyösi
k4	hatvani
k5	hevesi
k6	pétervásárai

Oktatásfajta tábla (legmagasabb oktatásfajta az adott településen)

oktatáskód	oktatásfajta_név
o1	Óvoda
o2	Általános iskola
o3	Középiskola
o4	Felsőoktatási intézmény

Kapcsolatok

```

    graph LR
      k1[kistérség] --- település[település]
      település --- o1[oktatásfajta]
  
```

Adatbáziskezelés és Prolog

Készítette: Pántya Róbert

6. ábra. Adatbáziskezelés és Prolog

Ezt a minta-adatbázist a Prologban úgy lehet megadni, hogy a három tábla minden egyes sorának (rekordjának) megfeleltetünk egy-egy Prolog ténytet, mely tényeket adatbázis predikátumoknak is szoktak nevezni a logikai programozás terén. A három tábla rekordjainak megfelelő Prolog tények sémája így adható meg:

település(településkód, településnév, terület, népesség, kistérségkód, oktatáskód).
kistérség(kistérségkód, kistérségnév).
oktatás(oktatáskód, oktatásfajta_név).

A Prolog tények pedig a következők:

```

település(1,'Eger',104.77,58976,k1,o4).
település(2,'Lőrinci',33.29,5968,k4,o3).
település(3,'Heves',99.21,11450,k5,o3).
település(4,'Bélapátfalva',36.17,3391,k1,o2).
település(5,'Mónosbél',14.24,399,k6,o1).
település(6,'Gyöngyös',54.1,33807,k3,o4).
település(7,'Abasár',26.65,3115,k3,o2).
település(8,'Füzesabony',46.34,7986,k2,o3).
település(9,'Aldebrő',21.78,791,k2,o2).
kistérség(k1,egri).
kistérség(k2,füzesabonyi).
kistérség(k3,gyöngyösi).
kistérség(k4,hatvani).
kistérség(k5,hevesi).
kistérség(k6,pétervásárai).
oktatás(o1,'Óvoda').
oktatás(o2,'Általános iskola').
oktatás(o3,'Középiskola').
oktatás(o4,'Felsőoktatási intézmény').
  
```

Amennyiben az adatbázisunk segítségével információt szeretnénk kinyerni, a lekérdezést a következőképpen fogalmazhatjuk meg SQL-ben, illetve Prolog-ban.

Melyek azok a települések, amelyek népessége 10 000 főnél nagyobb?

SQL	<pre>SELECT településkód, településnév FROM település WHERE népesség > 10000;</pre>
Prolog	<pre>népes_település(Településkód, Településnév) :- település(Településkód, Településnév, _, Népesség, _, _), Népesség>10000.</pre>

Mennyi az adott település népsűrűsége fő/km²-ben (népesség/terület)?

SQL	<pre>SELECT településkód, településnév, népesség/terület AS népsűrűség FROM település;</pre>
Prolog	<pre>népsűrűség(Településkód, Településnév, Népsűrűség) :- település(Településkód, Településnév, Terület, Népesség, _, _), Népsűrűség is Népesség/Terület.</pre>

Melyek azok a települések, amelyek népessége 5000 főnél nagyobb és az egri kistérséghez tartoznak?

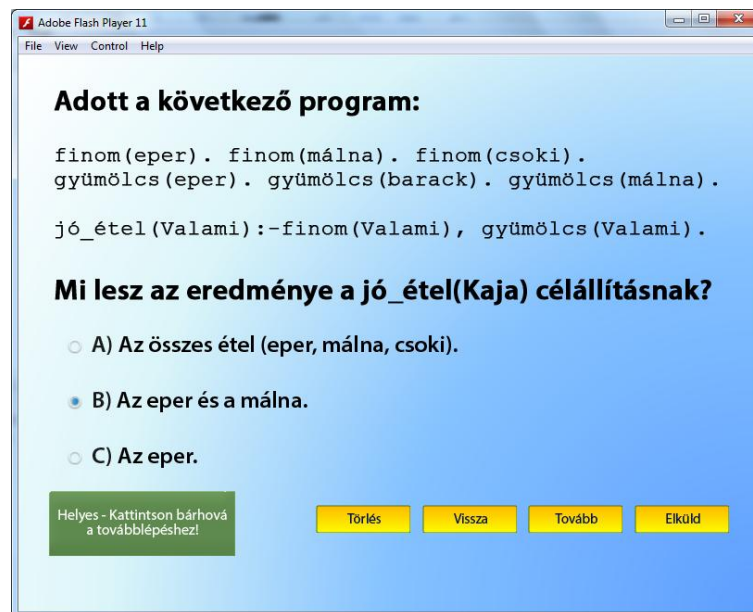
SQL	<pre>SELECT település.településkód, település.településnév FROM település, kistérség WHERE (település.kistérségkód = kistérség.kistérségkód) AND ((település.népesség > 5000) AND (kistérség.kistérségnév = "egri")) ;</pre>
Prolog	<pre>egri_kist_népes(Településkód, Településnév) :- település(Településkód, Településnév, _, Népesség, Kistérségkód, _), Népesség >5000, kistérség(Kistérségkód, egri).</pre>

Tudásellenőrző tesztek

Az elektronikus tananyagok használatakor rendkívül fontos a tudásellenőrzés, vagyis a megfelelő visszajelzések megadása a tanulás folyamatáról mind a tanulónak, mind pedig az oktatónak.

A tudásellenőrző tesztek minden nagyobb témakör végén rendelkezésre állnak, nagy előnyük, hogy korlátlan számban kitölthetőek, valamint a kellően nagyszámú kérdés-adatbázis (300-500 kérdés), valamint a kérdések véletlenszerű kiválasztása, a válaszok megkeverése, lehetővé teszi a megfelelő objektivitás elérését.

A teszteknel finoman hangolhatjuk a megoldási időt, az eredményesség elérését, valamint az e-learning rendszerben a továbbhaladást egy bizonyos teszt-eredmény elérésének a feltételéhez is köthetjük. A 7. ábra egy ilyen tudásellenőrző tesztet tartalmazó animációt mutat.



7. ábra. Tudásellenőrző teszt

3. Összefoglalás

A problémamegoldó gondolkodás fejlesztése nagyon fontos feladat minden oktatási szinten. A problémamegoldás fejlesztésére nagyon hasznos módszerek bizonyul a logikai programozás oktatása. A logikai programozás oktatása, tanulása nem egyszerű feladat, ezért ezen a területen is szükség van új módszerek kifejlesztésére. Úgy gondolom, hogy az interaktív animációk, megfelelően beágyazva egy e-learning rendszerbe, hatékonyan segítik a logikai programozás megértését, megtanulását, ezáltal a tanulók problémamegoldó képességének fejlesztésére is jótékony hatással vannak.

Irodalomjegyzék

- [1] Pántya R. (2011): Mesterséges intelligencia elemekkel támogatott programozás oktatása. Doktori értekezés, ELTE Informatika Doktori Iskola
- [2] <http://mentor.karolyrobert.hu>
- [3] Márkus Zs. (1988): Prologban programozni könnyű. Novotrade, Budapest.
- [4] Makány Gy. (1995): Programozási nyelvek: Prologika Mikrológia 7, ELTE Általános Számítástudományi Tanszék, Budapest
- [5] Máday Á. (2000): Prolog feladatgyűjtemény (szakdolgozat), ELTE

Térszemlélet fejlesztése a műszaki képzésben

Developing the spatial ability in the technical education

Papp Ildikó

Debreceni Egyetem, Informatikai Kar
papp.ildiko@inf.unideb.hu

Absztrakt: A műszaki-mérnöki képzésben igen fontos szerepet kap a térszemlélet, térlátás fejlesztése. A hagyományok szerint erre az Ábrázoló geometria, Műszaki ábrázolás tárgyak hivatottak, melyek a térbeli alakzatok szemléltetésén túl az azokkal megfogalmazott geometriai feladatok megoldásával a térbeli absztrakciós képességet teszik próbára. Maga a tárgy az aprólékos szerkesztő munkának köszönhetően a mérnöki szemlélet és pontos munkavégzés kialakításának is eszköze. Ideális esetben ezek a kurzusok kis létszámú csoportokban, az oktató folyamatos visszajelzéseivel, főként hallgatói ötletek megvalósítását jelentenék. De tudjuk, hogy ideális eset nincs, sőt még ideális-közeli eset sincs. Jelenleg évfolyam szintű előadások, és igen nagy létszámú gyakorlatok valósulnak meg. Ezeken kívül a nagyméretű hagyományos táblákat és az ahhoz használatos eszközöket felváltják a kisebb méretű filctollas- vagy digitális táblák.

A megváltozott körülményekhez alkalmazkodó oktató igyekszik ebből előnyt kovácsolni, és az infokommunikációs eszközöket és az élményközpontú oktatás elemeit bevonva változatos, érdeklődést felkeltő- és fenntartó órákat tartani. Az otthoni felkészülést és feladatmegoldást oly módon kell támogatni, hogy az elérhető segédlet visszaidézzé az órán hallottakat, folyamatosan elérhető, visszakereshető legyen, de az is fontos tényező, hogy a hallgatók igen jelentős része a kurzus teljesítésének idején nem rendelkezik saját használatú számítógéppel, a szoftverhasználatban kevésbé járatos, többnyire az okostelefonja jelenthet biztos pontot.

Az előadásomban azokat az IKT eszközöket szeretném bemutatni, melyeket az utóbbi években a Műszaki Karon alkalmaztam. A választott eszköztár hasznossága visszajelzésekkel és hallgatói teljesítményekkel többnyire mérhető. Minden órátípushoz, témakörhöz, tantermi adottságokhoz és vegyes összetételű, véletlen szerveződésű tanulócsoportokhoz egyaránt kiválóan alkalmas módszer nem létezik. Így magam részéről kötelezőnek tekintem a módszereim folyamatos felülvizsgálatát és megújítását.

Kulcsszavak: oktatás, élményközpontú oktatás, térszemlélet, modellezés, infokommunikációs technológia

Abstract: The development of spatial ability is very important in the technical education. The courses Descriptive Geometry and Technical Drawing make an important role in the representation of three-dimensional objects in the plane and solving the geometric problems formulated with them. Spatial abstraction ability of the students is put to test. Thanks to the painstaking construction the courses are the tools for accurate working and evolving engineering vision. Ideally, the education takes place in small groups with ongoing feedback of the instructor especially the common realization of students' ideas. But we know that there is no ideal case, or even close to that. Currently, grade-level lectures are implemented with very large groups on practices. The traditional tools are replaced with smaller marker whiteboards or digital boards, so new methods of teaching are necessary. The aim of my talk is to present my infocommunication technology tools which have been applied at the Faculty of Engineering in recent years. Considering the students' feedback, my methods are continuously changing, and I would like to find the optimal ones for my courses.

Keywords: education, experience-focused education, spatial ability, modeling, infocommunication technology

1. Bevezetés

A műszaki képzés első tanévében a természettudományos alapismeretek problémaorientált oktatása mellett a térszemlélet fejlesztésére is hangsúlyt fektetnek. A többszöri tantervdolgozásoknak köszönhetően az óraszámok minden szakon csökkentek és a

gazdasági- és épületkapacitási szempontokat is szem előtt tartva nagy létszámú csoportokkal kell dolgoznunk. A Debreceni Egyetem Műszaki Karán a gépészmérnök, műszaki menedzser, építőmérnök és építészmérnök Bs.C. szakos hallgatók kötelező tárgyai között szerepel a térbeli absztrakciós képesség fejlesztése az Ábrázoló geometria vagy Műszaki ábrázolás tárgyak keretein belül.

Pályám kezdetén, közel húsz éve, az oktatók a táblai szerkesztéseken kívül a körülményesen szállítható régi írásvetítőket használhatták. A kivetítéseken folyamatában lehetett bemutatni a hosszadalmas szerkesztést. A módszer előnye, hogy nem igényelt hosszabb előkészítést, csak a kiinduló ábrákat kellett előre megrajzolni, a szerkesztés az óra alatt került a fóliára. Mivel az oktató félig a hallgatóság felé fordult, így rögtön láthatta a visszajelzéseket. Ha ugyanazt a feladatot több csoportban is bemutattuk, akkor minden alkalommal újra kellett szerkeszteni, és nem volt lehetőség a visszalépésekre.

A visszalépésekre egy idő után egyre nagyobb igény mutatkozott, de a használható fizikai eszközök a régi maradtak, így olyan tananyagot, segédletet készítettem [1], amely a bemutatott feladatokat apró lépésekre bontva mutatja be, egymásra legfeljebb 2-3 fóliát helyezve. Ennek a technikának pozitív visszajelzése volt mind a hallgatók, mind a kollégák körében.

Napjainkban a Műszaki Kar szinte minden tanteremében lehetőség van a projektoros kivetítésre. A hallgatói létszámok nemcsak az előadásokon emelkedtek, hanem közel 40 fős gyakorlati csoportokkal is találkozhattam. Ez igen megnehezíti a hallgatói munka felügyeletét, így mindenképpen törekednem kellett arra, hogy a tábla közeléből minél inkább eltávolodhassak. Az otthoni felkészülés támogatására a Miskolci Egyetemen olyan elektronikus könyvet [2] készítettek, amelyben a szemléltető ábrák dinamikusak, egérrel mozgathatók, forgathatók, zoom-olhatók.

Több kolléga foglalkozott dinamikus geometriai szoftverek (pl. GeoGebra) oktatásban történő alkalmazásával, [3], [4]. Az általuk készített segédanyagok [5], [6] böngészőben futó Java appletként tekinthetők meg, nem igénylik a szoftver ismeretét, használatát.

2. Motiváció és az elkészült oktatási segédanyag

Az elsőéves hallgatók viszonylag kis része műszaki végzettséget adó középiskolából érkezik. Ők azok a hallgatók, akik már tanultak ábrázoló geometriát, és a többségük dolgozott már CAD rendszerrel. A többiek gimnáziumi tanulmányok után kezdik a műszaki tárgyakkal való ismerkedést, sem ábrázoló geometriai, sem tervezőrendszeri előképzettségük sincs. Így minden évben ilyen vegyes évfolyamokkal kell dolgoznunk. Néhány évvel ezelőtt, amikor felkérést kaptam előadások és gyakorlatok tartására, figyelniem kellett erre a helyzetre. A hallgatókon kívül a gyakorlatvezetők munkájához célszerű volt segítséget nyújtanom, így egy nagyon egyszerű szerkezetű weboldal létrehozása mellett döntöttem. (lásd 1. ábra) A tantárgy oldalának jelentős része egy táblázat, amely tulajdonképpen egy kibővített syllabus. Heti bontásban megnevezi a témakört, tartalmazza az előadáson használt diasorra, a feladatlapokra, házi feladatokra és egyéb segédanyagokra mutató linkeket. Ezen kívül állandó elem volt az ajánlott irodalomra és a gyakorló feladatokra mutató linkek listája, a házi feladat elkészítésének módja, illetve az aktualitások (zh időpontok, eredmények).

Ábrázoló geometria (MFABR31X04) Építőmérnök szakosoknak, 2012/2013. I. félév

Tananyag heti bontásban:

Oktatási hét	Témakör	Diasor	Feladatlap az előadáshoz	Egyéb segédanyag	Házi feladat
1. hét	Bevezetés, térelemek ábrázolása	01eax.pdf	-	vetületek	
2. hét	Illeszkedési feladatok	02eax.pdf	-	gyakorlatra	HF2
3. hét	Metszési feladatok	03eax.pdf	03eax_feladatlap.pdf	egy feladat megoldása	HF3
4. hét	A képsíkrendszer transzformációja	04eax.pdf	04eax_feladatlap.pdf	-	HF4
5. hét	Sík leforgatása	05eax.pdf	05eax_feladatlap.pdf	-	HF5
6. hét	Metrikus feladatok, mérőlegesség	06eax.pdf	06eax_feladatlap.pdf	Távolság és szög értelmezése	HF6
7. hét	Rajzhét, 1 ZH	-	-	-	

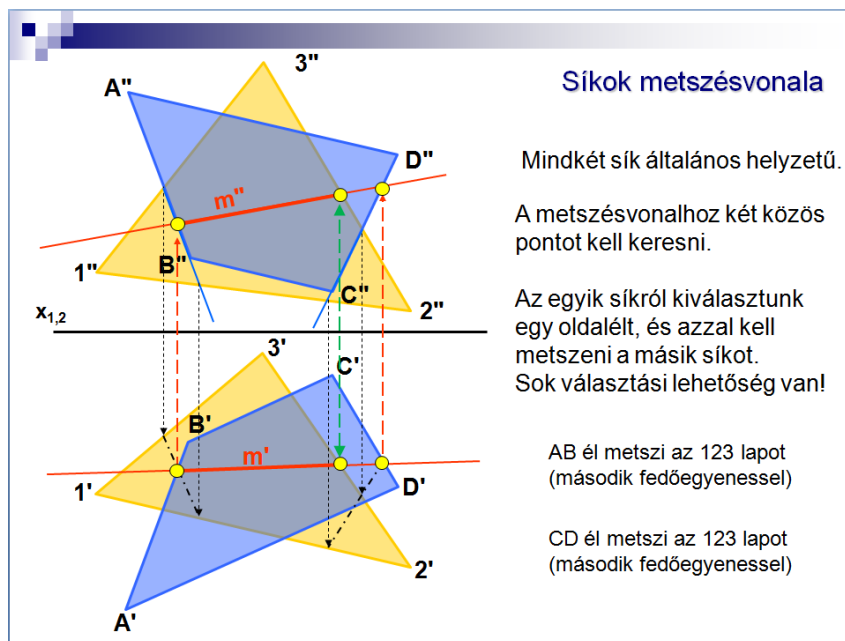
1. ábra. Az Építőmérnök szakosok oldalának egy részlete, 2012/2013. tanév

Az oktatási segédanyag kidolgozásakor az alábbi szempontokat követtem:

1. Letölthető legyen, de a letöltés után internetkapcsolat nélkül is használható maradjon.
2. Dinamikus tananyag készüljön és ne csak képfájlokat tartalmazó diasort használjak. A sok képet tartalmazó diasor a mérete miatt nehezebben játszható le, célszerű lenne az anyagot pdf verzióban is közzétenni. Ennek előnye, hogy akár mobiltelefonon is megnyitható.
3. A hallgatót segíteni szeretném az ábrázoló geometria tanulásában, így nem várhatom el bármilyen geometriai- vagy modellező rendszer kezelését. Az egyetemi tanulmányok legelső szemeszterében vagyunk, csak ajánlom ezt, és alkalomadtán az előadás színesítéséként megmutatok néhány egyszerű lépést egy tervező rendszerben (pl. SolidWorks).
4. Hosszabb távon vonzóbb, korszerűbb keretrendszert szeretnék a weboldal helyett, regisztrációt nem igénylő, bárki számára elérhető tartalomközvetítést valósítanék meg. Ebben szempont az is, hogy platformfüggetlen legyen úgy, hogy nekem ez viszonylag kevés háttér munkát jelentsen.

Az első két szempontnak megfelelő tananyagot egy békéscsabai középiskolában [7], illetve a Zágrábi Egyetem megfelelő témájú kurzusához már készítettek [8]. Mindkét munka hasonló technikát alkalmaz: PowerPointban teljesen animált diasorok, amelyek megengedik a tananyagban a lineáristól eltérő ugrásokat, előremutatásokat, visszacsatolásokat. Az ábrák a PowerPoint beépített alakzataiból (vonal, törött vonal, kör/ellipszis, körív/ellipszisív, téglalap, szabálytalan alakzatok, stb.) készülnek, a pozicionáláshoz a dia felületét koordinátasík 19 cm×25,5 cm-es részletének tekintjük. A megfelelő méretű, színű és vonaltípusú alakzatok, magyarázatok a lejátszás során filmszerűen jelennek meg, kiemelődnek, magukra vonják a néző figyelmét. A lejátszást módját tetszőlegesen lehet megadni, de célszerű a felhasználóra bízni egyszerű előre-hátra történő léptetéssel.

Mindkét munkát alaposan áttanulmányoztam, és a hasonlóságok mellett a különbségekre is odafigyeltem. Nagyon szigorú tematikát kell követnem, a legtöbb témát egy vagy legfeljebb két hét alatt dolgozzuk fel. Úgy terveztem, hogy a saját diasorom nem egymáshoz tartalomjegyzékkel kapcsolódó fájlok gyűjteménye lesz, hanem különálló, önállóan működőképes bemutatókból fog állni. Az említett munkákból az ötleteken kívül több ábrát is áttemeltem, de minden esetben változtattam az animáción, vagy az alakzatok helyzetén, vagy több magyarázó szöveget helyeztem el rajta. Igyekeztem letisztult oldalakat létrehozni rövid tájékoztató szövegekkel, szerkesztési menetek leírásával, az animációkat pedig úgy összeállítani, hogy ne legyenek túl dinamikusak. (lásd 2. ábra). A saját hallgatóim visszajelzései alapján elég hamar kialakult egy számukra elfogadható eszközhasználat, melyet a későbbiekben a kiszámíthatóság érdekében végig alkalmaztam.



2. ábra. Saját diasorom egy mintaoldala

A következő típusú oldalakat alkalmaztam:

- Összefoglaló dia: Általában a témakörök elején az a esetek áttekintésére használtam, szöveges információt tartalmaznak.
- Térbeli helyzetet magyarázó dia: A feladatok megoldása előtt a térbeli konstrukció bemutatására használtam. A helyzetek bemutatására a beépített alakzatokból felépített ábrákon túl gyakran fényképeket, nyomtatott jegyzetektől szkennelt ábrákat, 3D-s modelltől készült renderelt képeket használtam, melyeken utólag emeltem ki részleteket.
- Feladat megoldását bemutató dia: A rövidebb, egyszerűbb feladatok esetében egy oldalon mutattam be, a hosszabb feladatok esetében több diára bontottam, hogy a pdf verzióban is követhető legyen. Ez tulajdonképpen az [1]-ben alkalmazott technika újabb eszközzel kidolgozva.

A szempontrendszerem harmadik pontjának oly módon igyekeztem eleget tenni, hogy ahol lehetőség volt rá, az előadáson alkalmaztam 3D-s modellezést. Ez megtörte az előadás

monotonitását, aktivitás történt, és akár arra is lehetőség nyílt, hogy a jelenlévők mondják meg a készülő modell valamely paraméterét. Ezekről a modellekről készült képek utólag bekerültek a diasorba. Nem minden előadáson volt modellezésre idő vagy lehetőség, így a ritkasága, váratlansága, ismeretlensége miatt egy nagyon várt elemmé vált.

3. Az *Ábrázoló geometriát tanítok* blog

A kurzusok segédanyagai az első néhány félévben folyamatosan készültek, akkor a vágyott keretrendszerrel nem volt időm foglalkozni. Amikor a tananyagom már csak utógondozást igényelt, akkor a fennmaradó időt a keretrendszerre szántam. Egy magáncélú blog vezetése adta az ötletet, hogy a hallgatókkal való kapcsolattartás felülete egy blog legyen. Úgy gondoltam, hogy a fiatalok egy része az érdeklődési területeinek megfelelően ismeri a formát, (esetleg saját blogja is van), a kezdő blogolvasókat pedig igyekeztem több szempont szerinti rendezéssel segíteni. A 2013/2014-es tanév tavaszi félévben kisebb hallgatói létszámmal dolgoztam, így a régi weboldal mellett elindítottam az *Ábrázoló geometriát tanítok* blogot [10]. (lásd 3. ábra)



3. ábra. Az *Ábrázoló geometriát tanítok* (<http://abrazolottanitik.blogspot.hu/>) blog főoldala

A Google Blogger alkalmazásban minden sablonhoz tartozik a mobileszközön való megjelenés sablonja is, így a felhasználónak nem erre külön figyelnie, de természetesen lehetőség van a megjelenítés testre szabására. A megjelenésben egy egyszerű sablont és megszokottabb, kéthasábos elrendezést választottam, amelyben a baloldali szélesebb hasáb a posztok megjelenésének a helye, míg a jobboldali keskenyebb hasáb a következő kiegészítő információkat tartalmazza:

- *Köszöntő*, amelyből az olvasó a blog témájáról tájékozódhat. A Blogger a saját főoldalán felkínálhatja a blogot, illetve bárki véletlenül is idetévedhet.

- *Címkék*: A posztok tematikájához kapcsolódóan címkék alapján is elérhetők. A gyakoriságot a betűméretek és a zárójelben lévő számok is mutatják.
- *Ajánlott irodalom*, amely letölthető vagy böngészőben használható segédletekre mutató linkek listája.
- *Oldalmegjelenítés számlálója*, amely a saját megjelenítéseimet természetesen nem számolja.
- *Blogarchívum*, amely a megjelent posztokat a megjelenésük időpontjai alapján rendezi.
- *Népszerű bejegyzések*, ahol az utolsó hét nap öt leglátogatottabb posztját láthatjuk indexképpel és a bejegyzés első pár mondatával.

A főoldalon kívül további aloldalakat hoztam létre:

- *Házi feladatok*: Ezen az oldal összegyűjtve tartalmazza házi feladatok elkészítésének módját, a leadási határidőket és a házi feladatok linkjét.
- *Szorgalmi feladatok*: A féléves témákhoz kapcsolódóan a kötelezőkön túl szabadon választható feladatokat is kitűztem. Ezeket leadási határidejét és a feladatok linkjét, illetve a leadási határidő után a megoldást is ezen az oldalon lehet megtalálni.
- *A 2013/2014. II. félév oktatási posztjai* ezen az oldalon időben visszafelé érhetők el.
- *Értékelés*: Itt érhetők el a beadott feladatok, zárthelyi dolgozatok eredményei, a félév végi aláíráshoz szükséges minden információ.
- *Gyakorló feladatok*: Egyéni gyakorlásra szánt feladatok linkjeit tartalmazza.
- *Érdekességek*: Eredetileg nem terveztem ezt az oldalt, viszont a félév során találtam olyan geometriához, ábrázoló geometriához kapcsolódó érdekességeket, melyek külön ezen az oldalon is elérhetők.

Oktatással kapcsolatos bejegyzések hetente jelentek meg, igyekeztem az előző hét végén elérhetővé tenni azokat. A bejegyzések ugyanolyan szerkezetűek, bár témánként eltérő hosszúságúak lettek.

- *A heti tananyag rövid összefoglalása*: Alapvetően a heti témakör egy bekezdésnyire tervezett, a valóságban a témától függően ennél hosszabb leírása.
- *Előadás*: Az előadás diasorának pdf verziójának és az előadás követését segítő, a bemutatott feladatok kiinduló ábráira mutató linkeket adtam meg.
- *Házi feladat*: A kiadott feladat linkjét és beadási határidejét tartalmazza.
- *A gyakorlat anyaga*: Ebbe a pontba a feldolgozott feladat linkjét, illetve a gyakorlat után a megoldás menetét, táblaképet, vázlatot gyűjtöttem.
- *További segédanyag*: Kifejezetten a heti témakör megértéséhez ajánlott videókat, posztot, leírásokat, gyakorló feladatokat foglaltam egy pontba.

A bejegyzéseket úgy terveztem, hogy azok kommentelhetők legyenek, és ehhez ne kelljen senkinek sem regisztrálnia a felületen. Úgy gondoltam, hogy elindulhat egy egymást segítő, támogató beszélgetés. Néhány esettől eltekintve a hallgatók az e-mailben, vagy Neptun-üzenetben közvetlenül tőlem kérdeztek, vagy a közösségi oldalakon egymástól informálódtak. Tulajdonképpen ez volt a blog egyetlen olyan funkciója, amelyben rejlő lehetőséget a

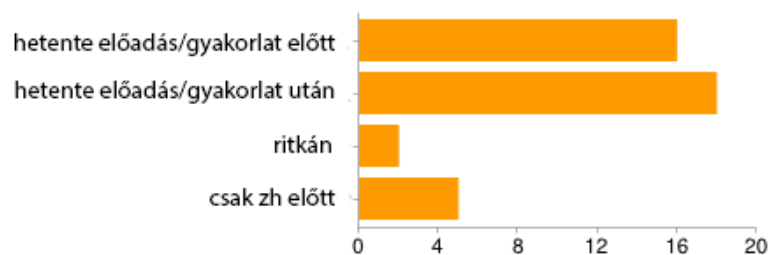
hallgatók nem használták ki, ennek ellenére a kommentelhetőség lehetőségét továbbra is fenntartom.

4. Hallgató vélemények

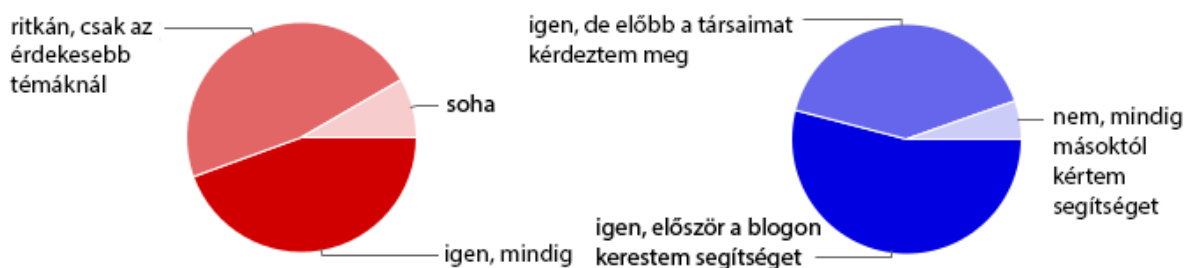
Miután nagyon sok időt szántam a tananyag elkészítésére és annak a publikálására, így fontosnak éreztem, hogy a célközönség, vagyis a hallgatóim hogyan vélekedtek róla. Természetesen menetközben több kötetlen beszélgetést folytattunk, amelyben megfogalmaztak kéréseket, de a félév végén egy online kérdőívet készítettem, melyet anonim módon tölthettek ki [11]. A kérdéseket a következő szempontok szerint csoportosítottam:

- *Kurzus, segédanyagok*
- *Követelmények*
- *A blogra (honlapra) vonatkozó kérdések*
- *Informatikai háttérre, elektronikus segédletekre vonatkozó kérdések*

Az összesítés során főként az utóbbi két kérdéskörre adott válaszokat emelem ki. [12]. Az online kérdőívet az Ábrázoló geometria II. kurzuson részt vett hallgatók 80 %-a töltötte ki. (A 2013/2014-es tanév tavaszi félévében 46 hallgató vette fel a tárgyat, a kérdőívet 37 fő töltötte ki.) A blog látogatottsága meghaladta az 5500 megjelenítést (2014. 06. 21.) a vele párhuzamosan működő, folyamatosan frissülő, régi típusú weboldal mellett, amely mindenképpen azt mutatja, hogy a hallgatóim rendszeresen használták az oldalt. A válaszadó hallgatók 83%-a heti rendszerességgel kereste fel a blogot (4. ábra), 92 %-a nemcsak a linkeket kereste, hanem a bejegyzéseket is olvasta (5- ábra). A heti témákat szinte mindenki könnyen megtalálta. A nyilatkozók 54 %-ának biztonságérzetet adott, hogy az oldalon minden szükséges információ elérhető, 41 %-uk előbb a társait kérdezte meg (5. ábra).

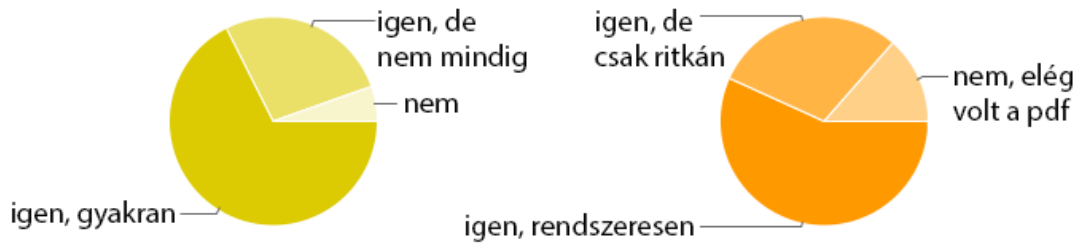


4. ábra. A blog látogatottsága [12]



5. ábra. A posztok olvasottsága, és a blog, mint elsődleges információforrás [12]

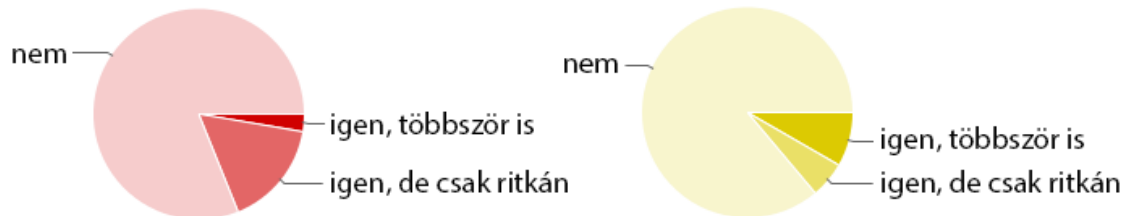
A hallgatók 68 %-a gyakran használta az előadás pdf verzióját, de nem tartották fontosnak ezen oldalak nyomtatását. A 87 %-uk használta az animált diasort (6. ábra).



6 ábra. Az előadások pdf verziójának és az animált diasor szükségessége [12]

Az előadások követéséhez készített feladatlapokat a nyilatkozók 57 %-a mindenképpen fontosnak tartotta, viszont ezek összeállításánál figyelembe kell vennem, hogy csak az igazán szükséges feladatokat dolgozzuk fel ily módon közösen.

A műszaki oktatásban a számítógéphasználat igen fontos szerepet kap, ezért fontosnak tartom, hogy csoportjaimat arra motiváljam, hogy használjanak 2D-s, 3D-s tervezőrendszereket, modellező programokat. A képzésben az első évben ők maguk még nem érzik ezt fontosnak, többnyire csak azon hallgatók próbálkoznak a szoftverhasználattal, akik szakközépiskolából érkeztek. A tavaszi félévben a hallgatók 81 %-a nem próbálkozott az ábrázoló geometriai feladatok 2D-s szerkesztéssel való megoldásával, és 86 %-uk nem próbálkozott térbeli formák 3D-s modellezésével (7. ábra).



7. ábra. Ábrázoló feladatok megoldása 2D-s szerkesztőprogram segítségével, a térbeli alakzatok 3D-s modellezésével [12]

A kurzus folyamán igyekeztem rámutatni a más tárgyakkal való kapcsolatra, érdekes modellezési és utólagos képmanipulációs példákat mutatva. Reményeim szerint belső igényük lesz arra, hogy minél hamarabb megismerjék, és gyakorlott felhasználói legyenek a megfelelő CAD rendszereknek, képszerkesztő programoknak.

Végezetül a szabadon megfogalmazott vélemények alapján azt tapasztaltam, hogy a hallgatók jól tudtak élni a felkínált tananyaggal, annak újszerű elemeivel, a rendszerezést segítő bloggal, és egészében egy hallgatóbarát, a felkészülést támogató, nem időrabló és inspiráló rendszernek tartották. A hallgatóság teljes kiszolgálása továbbra sem lesz célom, de a részleteit igyekszem mindig az adott évfolyam tudásához, képességeihez igazítani.

5. Tapasztalataim és további terveim

A 2. és 3. fejezetekben bemutatásra kerültek az Ábrázoló geometria kurzuson alkalmazott infokommunikációs eszközök. Ezek mellett természetesen a hagyományos technikákat is használtam, azaz táblai körzős-vonalzós szerkesztéseket és a szemléltetés során kézbe vehető modelleket. Az alkalmazott technikák együttesen egy változatos témafeldolgozást tettek lehetővé. Az előadások és a gyakorlatok nagy része az élményközpontú oktatás elveit követte, amely előbb a problémák tevékenység során történő megismerését, és ezen alapulva történő feladatmegoldást jelenti. A feladatmegoldás során többször visszatértünk a szemléltetéshez, bizonyos részek éppen ekkor kerültek kiemelésre. Ez a módszer egy ilyen gyakorlatias kurzus esetén mindenképpen javasolható.

A hallgatókkal való kapcsolattartás formájaként továbbra is a [10] blogot fogom működtetni. A hallgatók bizalommal keresték fel az oldalakat, áttekinthetőnek, informatívnak tartották, a felkészülés során időt takarítottak meg. A továbbiakban is hetente jelennek meg az oktatással kapcsolatos posztok, időnként pedig a geometriai érdekességeket fogok publikálni. Hiánypótlásként mindenképpen létre kívánok hozni egy újabb aloldalt, ahol az alapvető, elemi geometriai szerkesztések leírását gyűjtöm össze.

Az animált diasorokat újabb feladatok leírásával fogom kiegészíteni, az előadások követéséhez készített feladatlapokat átdolgozva egy munkafüzetet összeállítását tervezem. Ez nemcsak az oktatói munkát segítené, hanem a hallgatók számára is kényelmesebb lenne, mivel nem kellene hetente a nyomtatásra (fénymásolásra) időt szánniuk. (A hallgatóknak csak 28 %-a élhetett bármikor a nyomtatás lehetőségével, míg 58 %-uk hetente nyomtatási szolgáltatást vett igénybe.)

Az ábrázoló geometria oktatása során több esetben is felmerült, hogy akár kísérleti jelleggel is, de a hagyományos oktatási formát (előadás-gyakorlat) ki kellene egészíteni géptermi, szoftverrel támogatott gyakorlattal. Ennek során a feladatokat a tárgy témakörei szolgáltatnák, és először síkbeli szerkesztőeszközök használatát, majd a 3D-s modellezést ismernék meg a hallgatók. Sajnos eddig az alapozó Műszaki informatika tárggyal ilyenfajta kooperáció nem jött létre, de szabadon választható tárgyként egy új tárgy bevezetésére már kaptam ígéretet. A hallgatóim 54 %-ának lenne igénye egy ilyen szoftverrel támogatott gyakorlatra a hagyományos gyakorlat mellett.

Az ajánlott segédanyagok között a SZIE Ybl Miklós Építéstudományi Karán, illetve a BME-n készült videókat is ajánlottam. A hallgatóim egyötöde többet is megnézett, hasznosnak tartották, csak a filmek hosszúságát kifogásolták. A többség viszont hasznosnak véli a szerkesztések leírását bemutató, tutorial-szerű rövidebb videókat, így a távlati terveim között egy videó gyűjtemény összeállítása is szerepel. [12]

6. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Papp Ildikó: Műszaki ábrázolás MFK-s hallgatóknak, (2002), elektronikusan elérhető: <https://docs.google.com/file/d/0B4b8DTKHyn6PQVF2TElxcEZwejA/edit>
- [2] Bancsik Zsolt, Juhász Imre, Lajos Sándor: Ábrázoló geometria szemléletesen, elektronikusan elérhető: http://193.6.8.43/segedlet/dokumentumok/Abrazolo_geometria_szemleletesen.php
- [3] Nagy-Kondor, R.: Spatial Ability, Descriptive Geometry and Dynamic Geometry Systems, *Annales Mathematicae et Informaticae*, **37** (2010), 199-210.
- [4] Fazekas, S., Molnár, A.: Dinamikus geometriai program az ábrázoló geometria oktatásában, *Matematika és Informatika Didaktika Konferencia*, Szatmárnémeti, (2011) 01.28-30.
- [5] Árvainé Molnár Adrien: DE_műszaki ábrázolás_tavaszi félév, elektronikusan elérhető: <http://www.geogebraTube.org/student/bRphLqLfM#>
- [6] Németh László: GeoGebra feladatok, elektronikusan elérhető: <http://www.emk.nyme.hu/index.php?id=18121&L=1>
- [7] Gubis Katalin: Ábrázoló geometria (belső használatra készült aminált diasor), Vásárhelyi Pál Szakközépiskola, Békéscsaba
- [8] Vlasta Szirovicza: Descriptive geometry, Self-published, Zagreb, Croatia (2007) ISBN 978-953-95667-0-6
- [9] A GeomTeh3D projekt (Fund for Development of University of Zagreb) keretében készült oktatási segédanyagok. (2012) Elektronikusan elérhető: http://www.grad.hr/geomteh3d/radne_eng.html
- [10] Papp Ildikó: Ábrázoló geometriát tanítok, Építőmérnök-, építészmérnök- és műszaki menedzser szakos hallgatók Ábrázoló geometria kurzusainak blogja, Elektronikusan elérhető: <http://abrazolottanitok.blogspot.hu/>
- [11] Papp Ildikó: Hallgatói véleményezés - Ábrázoló geometria 2., elektronikusan elérhető: https://docs.google.com/forms/d/17_fyGZjhpxIsA1eTUH0CjCdSx0rBKFQ3o71kEH5a9oA/viewform
- [12] Papp Ildikó: Hallgatói véleményezés elemzése, elektronikusan elérhető: https://docs.google.com/forms/d/17_fyGZjhpxIsA1eTUH0CjCdSx0rBKFQ3o71kEH5a9oA/viewanalytics

Kvadkopter drón az oktatásban és kutatásban

Quadcopter drone in education and research

Sári Zoltán, Schiffer Ádám, Várady Géza, Jancskárné Anweiler Ildikó

PTE Pollack Mihály Műszaki és Informatikai Kar

sari.zoltan@pmmik.pte.hu

Absztrakt: A beágyazott rendszerek, autonóm rendszerek, és irányításuk fontos területei a mérnök informatikus képzésnek, ahol a gyakorlati képzés során komplex rendszerek vizsgálatára is sor kerül. A cikk egy kvadkopter drón (mely a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú "Jól-lét az információs társadalomban" című pályázat keretében került beszerzésre), mint hallgatói oktató-kutató eszköz lehetőségeit, és potenciális alkalmazási területeit tárgyalja. A kvadkopter, mint oktatási platform, a beágyazott rendszerek alkalmazásának és összekapcsolási lehetőségeinek vizsgálatára, tesztelésére, önálló fejlesztések, és elképzelések kipróbálására, ötletek megvalósítására egyaránt alkalmas. Intézményünkben egyelőre szűk, kiválasztott hallgatói csoport foglalkozik - szoros együttműködésben az oktatókkal - a kvadkopter vezérlésével, irányításával és programozásával. Elsőként a platform megismerése, szenzorinformációk feldolgozása és egyszerűbb irányítási feladatok elvégzése a cél, mely megalapozza az összetettebb feladatok (pl. képfeldolgozás) megoldását. A feladatok elvégzése során a hallgatók gyakorlati tapasztalatokat szereznek a kvadkopter vezérléséhez szükséges programozással, számítógép-architektúrákkal és számítógép-hálózatokkal kapcsolatos problémák megoldásában, és sikerrel kamatoztathatják a korábban megszerzett elméleti tudásukat látványos végeredménnyel záruló projektekben. A kvadkopterrel kapcsolatos projektekbe bevont hallgatók motiváltak, sok önálló munkát fektetnek a projektekbe, melynek pozitív eredményi már jelenleg is megmutatkoznak.

Kulcsszavak: kvadkopter, oktatás-kutatás, komplex rendszerek

Abstract: Embedded systems, autonomous systems, and control theory are important fields of the education of engineering informatics, where complex systems, their usage and description is discussed. The paper deals with the general possibilities and possible applications of a quadcopter drone as an educational and research platform. The quadcopter can be a useful tool in the education of complex and embedded technologies, through its usage the students can get an insight into the behavior of complex, interconnected systems. In our faculty, currently a little group of students works on projects related to the drone. The goal of these projects is to investigate the possibilities of the quadcopter, work with sensory data, perform real-time image processing, and program basic manouvers and tasks. During these projects the students get experiences corresponding to various tasks, such as sending, receiving and processing data over a network, working with low-level architectures, and they can apply their theoretical knowledge in practice. The students involved in quadcopter related projects are motivated, they invest plenty of time and effort into these projects with encouraging results.

Keywords: quadcopter, education and research, complex systems

1. Bevezetés

A 2000-es évtized második felétől széleskörűen elérhetővé váltak a viszonylag olcsó és sokrétűen használható, mindamellett komplex technológiák, a drónok. Ezek az eszközök sok szempontból ideálisnak tekinthetők az oktatás területén, lévén összetett, sok mérnöki területet lefedő technológiák, és nem utolsósorban nagyon érdekes és látványos lehet a velük való munka. A megjelenésük korai szakaszában nagy volt a szakadék a professzionális, viszonylag nagy teherbírású, jó felszereltségű, de meglehetősen drága eszközök ill. az olcsó, elsősorban játéknak szánt drónok között, melyek azonban kutatási és fejlesztési feladatokra nem alkalmasak. A komolyabb, K+F célokra is alkalmas drónok megfizethetetlenül drágák voltak (részben még ma is azok) hallgatói projektekhez, így egészen a 2010-es évekig nem volt jellemző a drónok oktatási célú alkalmazása. Ezt, a professzionális és a játék szintű eszközök közötti szakadékot hidalta át a Parrot AR.Drone elnevezésű modellje, amely elérhető áron biztosít egy megfelelő szenzor-felszereltségű platformot, mely viszonylag flexibilis, és a kezdeti nehézségek áthidalása után könnyen programozható.

2. A drónok felépítése és működése

A drón elnevezés alatt egy vezető nélküli, autonóm, vagy távirányított légi járművet értünk. Felépítésüket tekintve igen változatosak, de alapvetően két nagy csoport különböztethető meg, a merevszárnyú (repülőgép-szerű) és a forgószárnyal rendelkező (helikopter-szerű) kialakítás. A cikkben részletesebben egy négyrotoros, ún. kvadkopter drónnal foglalkozunk, így ennek felépítését tekintjük át kicsit alaposabban.

A kvadkopter egyik nagy előnye, hogy a rotorok átmérője viszonylag kicsi, így a gép könnyen navigál beltérben, sokkal kisebb a valószínűsége, hogy a forgó rotor manőverezés közben nekiütközik valamilyen tárgynak, mint az egy nagy főrotorral rendelkező kialakítás esetében. Mindemelllett a kvadkopter fix rotorokkal kerül kialakításra, így az irányításhoz szükséges mechanizmusok is jóval egyszerűbbek. Dinamikáját tekintve a kvadkopter egy 6 szabadságfokú, alulaktuált nemlineáris rendszer, melynek viselkedése alapvetően instabil, így stabilan tartásához folyamatos és gyors szabályozásra van szükség, ennek hiányában nem irányítható.

Az AR.Drone architektúrája

Az AR.Drone szerkezetének alapját egy szénszálaz váz alkotja, mely könnyű és ellenálló, ez tartja a rotorokat, és a rotorok meghajtásáért felelős, saját mikrovezérlővel ellátott villanymotorokat, továbbá a drón 'alaplaját' (ld. 1. ábra). A drónnal való kommunikáció WiFi hálózaton valósul meg, a gyártó által definiált interface-en keresztül, alapvetően 3 portot használva az irányításhoz, a szenzorinformációkhoz való hozzáféréshez, és a képi adatok eléréséhez. A gépen két fedélzeti kamera helyezkedik el. Az egyik - lefelé irányított - kamera egy QVGA felbontású, a drón felszínhez viszonyított elmozdulását hivatott követni, így a gép stabilan tartásáért felelős, a másik - előre irányuló - HD-ready kamera egy viszonylag jó felbontású videót szolgáltat további feldolgozást ill. tárolást téve lehetővé.



1. ábra. Az AR.Drone 2.0 kültéri és beltéri testtel felszerelve

Az AR.Drone firmware a kamerák konkurens elérését teszi csak lehetővé, így vagy az egyik vagy a másik kamera által látott kép kérdezhető le. A kamerák közötti váltás kb. 300ms alatt megy végbe, így a kamerák szimultán, időosztásos lekérdezése sem ad kielégítő megoldást mindkét kép egyidejű feldolgozására.

A felhasználó által adott utasítások ún. AT parancsok formájában kerülnek kiküldésre a megfelelő UDP portra, melyeket az eszköz kb. 30ms-os ciklusidővel fogad és dolgoz fel, ugyanilyen ciklusidővel frissülnek az előfeldolgozott szenzoradatok, melyek a navigációs információkat (navdata) tartalmazzák. Ilyen adatok a gép térbeli orientációját jellemző Pitch, Roll, és Yaw, valamint a repülési magasság, sebességek (x,y,z), ill. az akku töltöttségi szintje (amennyiben a töltöttségi szint alacsony, a drón automatikusan landol).

Az ARM-alapú architektúrán működő Linux operációs rendszer lehetővé teszi az on-board feldolgozást is, természetesen a rendelkezésre álló erőforrások ésszerű kihasználásának szem előtt tartásával, így egyszerűbb feladatok megoldhatók extra hardver alkalmazása nélkül is [1].

3. A kvadkopter, mint dinamikai rendszer

A kvadkopter dinamikájának leírása az általánosított helykoordináták megadása után az Euler-Lagrange módszer alapján történhet [2][3]. A koordinátákat tartalmazó vektorok

$$\mathbf{q} = [x, y, z, \phi, \theta, \psi]^T, \mathbf{q} = [\xi, \boldsymbol{\tau}]^T, \quad (1)$$

$$\xi = [x, y, z], \boldsymbol{\tau} = [\phi, \theta, \psi]$$

ahol a \mathbf{q} vektor tartalmazza a szabadsági fokokhoz tartozó jellemzőket (3D helykoordináták és szögelfordulások), ξ és $\boldsymbol{\tau}$ pedig rendre a helykoordináták és a szögelfordulások vektorai a Lagrange-függvény felírásához, mely az alábbi alakú

$$L(q, \dot{q}) = \frac{1}{2} \dot{\xi}^T m \dot{\xi} + \frac{1}{2} \dot{\boldsymbol{\tau}}^T \mathbf{I} \dot{\boldsymbol{\tau}} - mgz, \quad (2)$$

és amelyből az Euler-Lagrange egyenlet felírható az alábbiak szerint

$$\frac{d}{dt} \frac{\partial L(q, \dot{q})}{\partial \dot{q}} - \frac{\partial L(q, \dot{q})}{\partial q} = F, \quad (4)$$

ahol a jobb oldalon álló F , a kvadkopterre ható külső erőket tartalmazza. Képezve a bal oldalon álló differenciálhányadosokat, a kvadkopter dinamikáját jellemző egyenletrendszer adódik.

A kvadkopterhez hasonló dinamikai rendszerek megismertetése a hallgatókkal egy egyszerűbb rendszer megismerésén, vizsgálatán és szabályozásán keresztül történhet. A Quanser 'QNET Vertical Take Off and Landing Trainer' eszköze [4] az NI ELVIS board-ra épülve LabView környezetben vizsgálható és programozható. Az egy rotort tartalmazó rendszer viselkedésének tanulmányozásán keresztül a hallgatók közelebb kerülhetnek a komplex dinamikai rendszerek viselkedéséhez, és megszerezhetik a szabályozásukhoz szükséges ismereteket és tapasztalatokat mind PID, mind Fuzzy szabályozó algoritmusokat használva. Egy a fentihez hasonló rendszert alkalmaz a TurAFA (Turkish Airforce Academy) a mérnökképzésben, a robotika és automatika témaköröket támogatandó [5].

4. Oktatási és kutatási alkalmazások

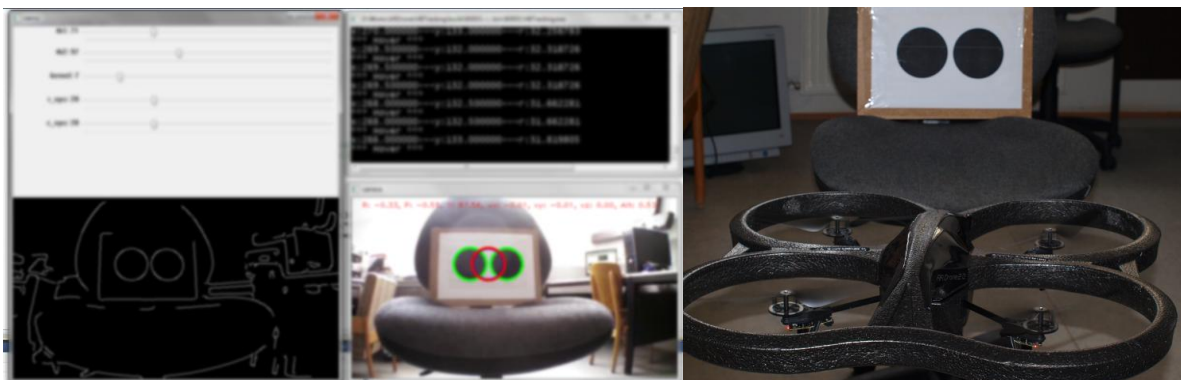
A drónok, mint oktatásban, hallgatói csoportokkal közösen végzett kutatásban használható eszközök igen hasznosnak bizonyulhatnak, első sorban sokrétűségük, viszonylag könnyű használhatóságuk, másodsorban vonzerejük miatt [6]. Didaktikai szempontból nézve elmondható, hogy a drónhoz kapcsolódó önálló hallgatói munka sokkal vonzóbb, mint az egyéb önálló laborok, házi feladatok, és a hallgatók szívesen foglalkoznak a kapcsolódó témákkal szabadidejükben is, szívesen próbálják ki ötleteiket a drón irányításával programozásával kapcsolatban.

Hallgatói projekteknél való felhasználás szempontjából további előny, hogy az AR.Drone a tapasztalataink alapján igen robusztus eszköz. A teszt-repülések során (különösen, ha valamilyen új, élesben még nem kipróbált algoritmus alapján történt az irányítás) gyakran

előfordult, hogy a drón valaminek nekiütközött, esetleg több méteres magasságból lezuhant, de az esetek döntő többségében nem szenvedett semmilyen strukturális sérülést, továbbra is működőképes maradt. A kisebb koccanások (a beltéri test alkalmazásával) semmilyen problémát nem jelentenek, nagyobb ütközéséknél, ha a rotor mozgása akadályozva van, a motorok automatikusan leállnak és a gép leesik, de ha a magasság nem túl nagy, akkor ez sem okoz különösebb gondot.

Az egyik, jelenleg is futó hallgatói projektünk egy C# alapú SDK fejlesztése az AR.Drone 2.0 hoz. Ez a fejlesztés egy olyan platform létrehozását célozza, amely lehetővé teszi, hogy az SDK leendő felhasználói egy magas szintű, objektumorientált környezetben elkészített keretrendszert használhassanak az eszköz programozására, irányítására, ill. valós idejű képfeldolgozási feladatok megoldására. A jelenleg fejlesztési fázisban lévő projekt már aktuális állapotában is lehetővé teszi a drón robusztus irányítását. A témával foglalkozó hallgatóink jártasságát tükrözi, hogy hallgatói csapatunk a 2014-es BEST-Continental programozó versenyen első helyezést ért el.

Egy másik fontos fejlesztési irány a valós idejű objektumkövetés és tájékozódás megvalósításához alkalmazható algoritmusok elemzése, tervezése, és implementációja. Ebben a témakörben az egyszerű színtér alapú küszöbölési eljárásoktól, a paraméteres geometriák transzformált térbe való leképezésén és felismerésén át, a SURF algoritmuson alapuló mintaillesztésig sok minden beletartozik. A 2. ábrán a kvadkopter egy ún. tag-et követ, melyet a háttérben működő képfeldolgozó algoritmus a minta geometriai jellemzői alapján ismer fel skála- és rotáció-invariáns módon.



2. ábra. Objektumkövetés a kvadkopterrel (a vezérlő program és a kopter)

Az képfeldolgozáshoz alkalmazott software környezet jellemzően az OpenCV ill. ennek C# wrappere az EmguCV, de van a kvadkopterhez kapcsolódó LabView alapú képfeldolgozással foglalkozó projektünk is.

5. Konklúzió

Összességében elmondható, hogy a kvadkopter drónok kétségkívül hasznos eszközei lehetnek számos, több mérnöki területet is érintő fontos téma oktatásának. Saját tapasztalataink azt mutatják, hogy a kvadkopterhez kapcsolódó kutatási feladatokhoz szívesen csatlakoznak a hallgatók. Az oktatásba bevonva pedig egy ilyen eszköz érdekes, és szemléletes platform lehet az autonóm rendszerek, dinamikai rendszerek, szabályozások, beágyazott rendszerek, számítógéphálózatok stb. tárgyak bizonyos témaköreinek tárgyalásakor.

6. Köszönetnyilvánítás

A projekt a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú "Jól-lét az információs társadalomban" című pályázat III./6. munkacsoport támogatásával valósul meg.

Irodalomjegyzék

- [1] T. Krajník, V. Vonásek, D. Fišer, and J. Faigl, "AR-Drone as a Platform for Robotic Research and Education," in *Research and Education in Robotics - EUROBOT 2011*, vol. 161, D. Obdržálek and A. Gottscheber, Eds. Springer Berlin Heidelberg, 2011, pp. 172–186.
- [2] P. Pounds, R. Mahony, J. Gresham, P. Corke, and J. Roberts, "Towards dynamically-favourable quad-rotor aerial robots," in *Proceedings of the 2004 Australasian Conference on Robotics & Automation*, 2004.
- [3] Y. Naidoo, R. Stopforth, and G. Bright, "Quad-Rotor Unmanned Aerial Vehicle Helicopter Modelling & Control," *Int J Adv Robotic Sy*, vol. 8, pp. 129–149, 2011.
- [4] Quanser Inc., *QNET Practical Control Guide*. 2009.
- [5] M. K. Bayrakceken, M. O. Kizilkaya, and A. Arisoy, "ENHANCING CONTROL EDUCATION WITH REAL-TIME EXPERIMENTS," in *INTERNATIONAL CONFERENCE of SCIENTIFIC PAPER AFASES 2012*, 2012.
- [6] Z. Dodds, "Can Quadrotors Succeed as an Educational Platform?," in *AAAI'11*, 2011, pp. 1–1.

Az informatikai kompetencia aktuális kérdései a műszaki felsőoktatásban

Current trends and changes of competences regarding IT skills in technical higher education

Dr. Szabó György^a, Dr. Czinkóczy Anna^b

^aBudapesti Műszaki és Gazdaságtudományi Egyetem
gyszabo@eik.bme.hu

^bCorvinus Egyetem Budapest
anna.czinkoczy@uni-corvinus.hu

Absztrakt : Napjainkban az informatikai tudásnak továbbra is kulcsszerepe van a felsőoktatásban, azonban az informatikai fejlődés felgyorsulásával az eszközök egyre bonyolultabbá válnak. Ezzel szemben a megszerzett tudás egyre gyorsabban évül el. A nagy szoftvergyártók komplikált és monolit rendszereket fejlesztenek ki és oktatási verziók kibocsátásával illetve a hallgatók/ oktatók számára nyújtott kedvezményes lehetőségekkel igyekeznek egy életen át magukhoz láncolni a felhasználókat. Az informatikai szoftverhasználat másik opciója lehet az ún. Open Source, azaz nyílt forráskodú, ingyenes szoftvercsalád használata, amelyet már több európai állam is támogat.

A függőség kialakításának veszélye mellett a hallgatók leginkább csak a konkrét szoftver őket érdeklő opcióinak megismerését tekintik fontosnak, holott az informatikai képzésnek szoftverfüggetlen és időtálló kompetenciákat kellene megtanítania. Cél, hogy a végzett szakembereknek nagyobb átlátása legyen a tanulmányikat érintő egész szakterület eszközrendszerére, s rugalmasabb tudásra tegyenek szert aminek segítségével módjuk legyen kitörni egy adott szoftver használatának csapdájából.

Kulcsszavak: szoftverfüggetlenség, Open source szoftver, kompetencia

Abstract: IT knowledge still plays an essential role in higher education. Nevertheless, the rapidly developing infrastructure and complex programs make it very difficult to stay up-to-date in IT applications. The main software manufacturers (Microsoft, Adobe, SPSS, etc.) are interested in developing huge and monolithic programs and by providing educational versions and student discounts, their aim is to bond professional users for lifetime.

The other option in education is to use Free Open Source programs, which is also becoming more popular in governmental usage due to the Open Source Initiative. The dilemma is how to define core IT skills and competencies in higher education concerning technical/ engineering Informatics that is software independent and enables students to switch between programs and applications and provides a flexible knowledge.

Keywords: software dependence, IT skills and competencies, Open Source initiative

1. Bevezetés: Az informatikai kompetencia változása a műszaki felsőoktatásban

Az informatika bizonyára az egyik legdinamikusabban fejlődő tudományterület, amely a felsőoktatásban - de különösen a műszaki felsőoktatásban- évtizedek óta állandóan megújuló szoftveralkalmazásokkal jellemezhető.

Általánossá vált, hogy nagy, összetett informatikai rendszerek, amelyek felölelik egyes tudományterületek egész eszköztárát, s amelyeket nagy tervezőcégek több ezer szakembere évtizedeken át fejlesztett (Mathematica, SPSS, CAD rendszerek), bizonyos tervezőfeladatok során megjelennek az egyetemi oktatóteremben. Azok a hallgatók, akik a középiskolából kikerülve a szoftver mögött álló tudományterület töredékét sem fogják fel illetve látják át, egyszerre felhasználóivá válnak bonyolult és intelligens rendszereknek anélkül, hogy tisztában lennének a használt program mögötti tudományos mélységgel. Ezért inkább csak olyan eszközt látnak benne, aminek működési elve fel sem fogható, mindössze a “hogyan kezeljem, mit nyomjak meg, hogy kijöjjön a megoldás?” motiválja őket.

Természetesen egyénenként és szakterületenként nagyon nagy az eltérés az informatikai tudásban és ismeretekben. Nyilván más igények merülnek fel az irodai munkakörnyezetben (Office ismeretek), mások a gazdasági tervezésekben (SPSS, R), míg az építészeti munkakörök speciális tervező rendszerek (ArchiCAD, AutoCAD) ismereteit igénylik.

Elmondható, hogy a szakmák specializálódása együtt jár a használt fő szoftverek eltérő alkalmazásával, tehát a szoftver egyben indikátora is lehet az adott szakterület összetettségének és alkalmazás-orientáltságának.

2. Jogdíjas szoftver vagy ingyenes (Free Open Source = FOS) szoftver használatának kérdései

A legtöbb informatikai szoftvergyártó (élén a Microsoft-tal) külön, kiemelt kedvezményeket nyújt az oktatásban résztvevők számára.

A sor folytatható az SPSS-sel, vagy az AutoCAD-del, ahol szintén oktatási verziót kaphatnak szinte térítés nélkül a felsőoktatásban dolgozók ill. hallgatók.

Ez látszólag jótékony gesztus az oktatás támogatására, de ugyanakkor a cégek szempontjából nagyon is racionális és jól megtérülő beruházásokká válnak hosszútávon.

Az informatikai szoftvergyártó cégek is pontosan felméri annak a jelentőségét, hogy milyen szoftveren tanul meg valaki egy szakmát, mert nagy valószínűséggel egy életen át elkötelezi magát az adott szoftver használata mellett.

És amíg egy oktatási verzió ingyenes, a profi tervezőirodák már igencsak borsos összegeket fizetnek évente az új verzióért és fejlesztett környezetekért.

Elmondható tehát, hogy a nagy szoftvergyártók szándékosan célozzák meg az oktatást és a monolitikus és összetett szoftvereiket megtanulva a hallgatók szinte egy életen át egy adott termék használatának fogságába esnek, ami egy hosszú távú gazdasági kiszolgáltatottságot jelent.

3. Free Open Source szoftverek elterjedése

A legtöbb kormány, illetve a profi informatikus társadalom nagy része felismerte a licenszes szoftverek csapdáját és célkitűzésként sok ország bevezette az államigazgatásban az Open Source, azaz nyílt forráskódú szoftverek használatát.

Ez természetesen egy évtizedek óta tartó folyamat eredményeként jött létre.

1998-ban megalakult az Open Source Initiative¹ (OSI), a világméretű együttműködés, amely azóta is egyre átfogóbban népszerűsíti a nyílt forráskódú szoftverek fejlesztését és használatát

4. Az átlagfogyasztó illetve az intelligens műszaki szakember eltérő kompetenciája

Természetesen ez az attitűd nem csak a felsőoktatásban résztvevő hallgatókra jellemző, hiszen egy átlagember is megnyomja a TV kapcsolót és nézi a műsorokat anélkül, hogy tisztában lenne a digitális képfeldolgozással vagy a jelátvitel problémáival.

Nyilvánvaló, hogy a felsőoktatásba bekerült hallgatók természetesnek veszik ezt az átlagemberre jellemző fogyasztói hozzáállást, hogy "csak az a lényeges, hogy tudjam uralni az eszközt, de nem számít, hogy miért van úgy és igazából felfoghatatlan a működési elve".

Azonban a felsőoktatás és különösen a műszaki felsőoktatás nem degradálódhat az átlag fogyasztói mentalitás szintjére, mivel a felsőoktatásban elsősorban olyan szakembereket képzünk, akik összetett rendszereket képesek átlátni és a műszaki problémákat absztrakt rendszerben is képesek felfogni (lásd 1. ábra).



1. ábra. Kompetenciák http://librarysciencelist.com/files/2013/03/core_competencies.png

A fogyasztói társadalmat a folyamatosan megújuló cikkek szolgáltatások vásárlása jellemzi ami részben arra épül, hogy a manipulált fogyasztó nincs olyan önálló döntés helyzetben, amelyben át tudná látni a dolgok működését és fel merné vállalni a saját maga erejével való megoldás útját.

Épp ez az attitűd ami eltérő kell, hogy legyen a felsőoktatásban résztvevő hallgatók esetében. Hiszen nekik kell a jövő műszaki szellemi termékeit megalkotni, tehát nem maradhatnak benne abban a csapdában, amit a fogyasztói társadalom feltételez és kínál az átlagfogyasztó számára.

A (műszaki) felsőoktatás igazi feladata, hogy a "bamba", kritikai szemlélettel alig vagy egyáltalán nem rendelkező hallgatót kiemelje az eddigi – a társadalom és az iskolarendszer

¹ <http://opensource.org/history>

sablonos átlagából- és gondolkodó, intelligens felhasználót neveljen belőle. Ezzel remélhetőleg több évtizeden keresztül alkalmas lesz az éltechnológiák uralására és akár fejlesztésére is.

5. Általános IT kompetenciák és azok változása

„Az informatikai műveltség alapja az, hogy az egyén felismerje, hogy milyen információra van szüksége és azt milyen módon tudja a leghatékonyabban elérni, feldolgozni és értelmezni. Az informatikai műveltség igen fontos a gyorsan változó technológiai rendszerekben, az egyetemi tanulmányok során, a munkában és a magánéletben egyaránt. Az egyetemi oktatásban a cél, hogy a tanuló az általa megalkotott absztrakt tervet, gondolatot meg tudja valósítani egy adott informatikai rendszerben...”

A fenti definícióból - amelyet az American Library Association (ALA) Elnöki Tanácsa fogalmazott meg 2000-ben kiemelnénk, hogy az IT kompetenciához hozzátartozik, hogy egy tetszőleges önálló mondanivalót, absztrakt tervet kell tudni konkrét informatikai **rendszerből függetlenül implementálni**.

Ebből szükségszerűen az következik, hogy **nem egy adott szoftver konkrét használatára** (melyik menüből mit válasszak ki, hogy kijöjjön az eredmény?) kell a tanulót megtanítani, hanem sokkal fontosabb az általános, rendszerből független struktúrák ismertetése. Ilyen lehet például az *Adatmodellezés*. Az sem lenne haszontalan, ha egy konkrét problémát többféle rendszerben (Open source vagy licenszes szoftverrel) megoldanának és elemeznék az egyes lépéseket, ezeket összehasonlítanák funkcionalitás szempontjából.

A realitás persze az, hogy az esetek többségében nincs mód egy adott feladatot többféle módszerrel is megoldani.

Ezért inkább azt ajánljuk, hogy önálló feladatként kapják meg a hallgatók azt a lehetőséget, hogy többféle rendszerrel dolgozhatnak, és aki tanári segítség nélküli megoldást talál az Internet felhasználásával, azt mindenképpen plusz ponttal jutalmazzuk.

Az informatikában különösen fontos az **önképzés**, hiszen a technológiai fejlődéssel legkönnyebben az Internet segítségével tarthatunk lépést. Ezt fontos kiemelni és a hallgatókat erre bátorítani.

A fentiekben szó volt arról, hogy az Informatika az egyik legjobb indikátorterülete a gyorsan változó és hamar elavuló műszaki tudásnak. Ezért nagyon fontos, hogy ne csak egy konkrét szoftver specifikus funkcióit tanítsuk, hanem megvilágítsuk a rendszer működési elvéhez tartozó alapvető elméleti összefüggéseket- ahol csak lehetséges.

Ide tartozik pl. az, hogy a térinformatikai vagy akár képfeldolgozó rendszerek milyen módon tárolják és kezelik az adatokat (vektoros- ill. raszteres rendszerek), mivel ezek olyan tulajdonságokra vannak befolyással mint a nagyíthatóság (zoom-olás) korlátai, adatpontosság, fájl méret, stb.

Azt is tudatosítani kell, hogy az adatok megjelenítése függ az *adatmodell*től, a modellben választott entitásoktól és a közöttük levő kapcsolatoktól.

Milyen összefüggések számíthatók ki, milyen eszközökkel és az eredmény megjelenítése milyen pontosságig értelmezhető.

Maga a konkrét szoftver rendszer igazából egy eseti megjelenítés megvalósítását adja, a menüpontok sorrendje is változhat, de a funkciók ennél rendszerint általánosabb elvek alkalmazásából adódnak.

Éppen azért, mert az informatika az egyik leggyorsabban változó terület, a naprakész tudáshoz elengedhetetlen az önálló tanulás.

Szerencsére az informatika nem csak igényli, de egyben biztosítja is a tudás megszerzésének folytonosságát.

Ma már megkérdőjelezhetetlen a Wikipedia mint online közösségi tudásbázis létjogosultsága, a Youtube video megosztókon nem csak a legújabb vagy a klasszikus zenei ritkaságok érhetők el, hanem bőségesen találunk tudományos előadásokat, számítási eljárásokat, szoftverhasználatot magyarázó videókat.

A felsőoktatásnak ezekben az anyagokban nem konkurenciát, hanem szövetségest kell látni, amely adott esetben tehermentesítheti az oktatót a mechanikus magyarázat alól- s egyben lehetőséget teremt egy magasabb, problémacentrikus, alkotó oktatás megteremtésére.

Saját oktatási tapasztalatom, hogy ha a kiadott informatikához kapcsolódó projektek anyagát döntően, de nem teljesen (kb. 95%-ban vesszük át, azonban mindig kapnak a hallgatók utalást online anyagokhoz,) sokkal motiváltabbá válnak, ha önállóan (Internet segítségével) elsajátított anyag beépítésére plusz pontot kapnak.

A felsőoktatásnak a célja nem egy végérvényes, fix tudás átadása tehát, hanem egy gondolkodó szemlélet kialakítása, amellyel a hallgató önállóan tud és mer ismereteket megtalálni és elsajátítani.

6. E-learning, Virtual learning

Az Open Source szoftverek igen elterjedtek az oktatásban, a Moodle e-learninges rendszerét több nemzetközi és hazai egyetem is használja. A Moodle rendszer további előnye az, hogy igen nagyszámú és lelkes felhasználó csoport véleményezi és fejleszti, s emiatt már alkalmas egy nagy intézmény összes oktatási feladatának kezelésére, szervezésére és tárolására. Ezen felül interaktív tananyagok létrehozását is támogatja a Moodle ; valamint ki lehet egészíteni plagizálást kiszűrő opciókkal is, azaz a virtuális oktatás teljesen átfogóan megvalósítható rendszere jön létre.

Összegzésként elmondható, hogy az Informatikai kompetenciák az informatikai rendszerek felgyorsult fejlődésével folyamatosan változnak, s ezek az oktatás minden szereplőjét – az oktatókat és hallgatókat egyaránt érintik.

Források, felhasznált irodalom:

[1] I.Information Literacy Standards in Higher Eduaction

<http://arizona.openrepository.com/arizona/bitstream/10150/105645/1/standards.pdf>

[2] Open Source Options For Education

Informatika a felsőoktatásban 2014 konferencia, Debrecen, 2014. augusztus 27–29.

by Mark Johnson on 13 January 2013 , last updated 3 January 2014

[3] CeLT / LRT Learning Technologies Review

[4] <http://lrt.mmu.ac.uk/lrtreview/>

Adatbázisok, adatbázis alapú programozás oktatása

Databases, education of database programming

Szendrői Etelka

Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Kar /Rendszer és Szoftvertchnológiai
Tanszék

szendroi@pmmik.pte.hu

Absztrakt: A mérnökinformatika BSC szakos hallgatók képzése során sokféle problémával kell megküzdenünk. Ezek egyike, hogy mennyire korszerű ismereteket adunk át a hallgatóknak. Komoly kihívás a hallgatók motiválása, érdeklődésük fenntartása. Előadásomban szeretném bemutatni az általunk tanított tematikát (Adatbázisok I és Adatbázisok II), valamint azt, hogyan épülnek egymásra az elsajátított ismeretek, hogyan alakul ki reményeink szerint a problémamegoldó gondolkodás hallgatóinkban. Ismertetni szeretném a tematikán túl a felhasznált szoftvereket, valamint azt, hogy milyen eszközökkel támogatjuk a hallgatókat a tananyag elsajátításában.

Kulcsszavak: adatbázis-kezelés, adatbázis programozás, problémamegoldás

Abstract: In the course of engineering information technologist (BSC) education we have to solve a lot of problem. One of these problems is how we can able to follow the rapid and great deal of change in the IT fields and to offer up-to-date knowledge to meet career track expectation for graduates. It is a big challenge to motivate our students and keep their interest. In my presentation I am going to introduce the syllabus of the courses titled Databases I and Databases II. and the courses prerequisite to them. We hope that we are able to prepare our students for the problem solving thinking. I am going to introduce the software, methods and technics used in our courses to help students to deepen their knowledge they gained in the lectures and practices.

Keywords: database management, database programming, problem solving

1. Bevezetés

A Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Karán a BSC mérnökinformatikus képzés nappali és levelező tagozaton történik. 2012-ben indult az MSC képzésünk szintén nappali és levelező tagozaton. Az alapszakon fő képzési célunk, hogy a gyakorlatban jól használható ismeretekre tegyenek szert a végzett hallgatóink. A képzési célnak megfelelően nagy hangsúlyt fektetünk arra, hogy alapos programozási ismeretekkel rendelkezzenek a hallgatóink. Az informatikai alkalmazások adatokkal dolgoznak, tehát a programozási ismereteken túl fontos az adatbázis-kezelés, adatmodellezés, adatbázis programozási ismeretek oktatása is.

A BSC képzés mintatantervének megfelelően az adatbázisokkal kapcsolatos ismereteket a harmadik és a negyedik félévben sajátítják el a hallgatók, a kötelező szakmai tárgyak blokkban, az Adatbázisok I. és Adatbázisok II. tantárgyak keretében. Az Adatbázisok I. tantárgy keretein belül a hallgatók megtanulják a relációs adatbázisokkal kapcsolatos alapfogalmakat, az adatmodell alkotás lépéseit, valamint az SQL nyelvet. Az Adatbázisok II tantárgy oktatása során ismerkednek meg a tanulók a tárolt eljárások, függvények, triggererek fogalmával, ezek gyakorlati felhasználásával. Megismerik az ADO.NET és az Entity

Framework technológiát, s ezek ismeretében adatbázisok használatán alapuló C# nyelvű programokat készítenek a Visual Studio fejlesztőeszköz segítségével.

A szakmai törzsanyag elsajátítása Microsoft SQL szerver és Visual Studio fejlesztőkörnyezetben történik. Az Adatbázis I. és Adatbázis II. tárgyakat követő programozással (pl. Webfejlesztés) kapcsolatos tárgyakban, valamint a szabadon választható tantárgyak keretében lehetőség van arra, hogy más technológiákat (MySQL, ORACLE, IBM DB2) is megismerjenek a hallgatók.

A továbbiakban bemutatom az Adatbázisok I. és Adatbázisok II. tantárgyak tematikáját és ismertetem a tantárgyak keretein belül folyó munkát.

2. Az adatbázisok kezelésével, programozásával kapcsolatos tantárgyak tematikái

Az Adatbázisok I. tantárgy a harmadik félévben kerül sorra, heti 2 óra előadás és 2 óra számítógépes gyakorlat formájában. Ekkor már a hallgatók túl vannak a Programozás I. (C programozási nyelv) és a Programozás II. (objektum-orientált programozás C# nyelven) tantárgyakon és az adatbázisokkal párhuzamosan hallgatják a Programozás III. (Programozás Java-ban) tárgyat. A kurzus vizsgával zárul.

Az Adatbázisok I. tantárgy fő témakörei

- Alapfogalmak, adatbázis, adatmodell, adatbázis-kezelő rendszerek, adatbázisrendszer. Rövid történeti áttekintés
- Az adatmodellezés három szintje. Adatmodellezési alapfogalmak. Egyed, egyedtípus, tulajdonság (attribútum), kapcsolat. Tulajdonságok és kapcsolatok osztályozása.
- A relációs adatmodell. Relációséma, reláció, megszorítások.
- Az ER modell leképezése relációs modellre. Leképezési szabályok.
- Funkcionális függés. Normalizálás, normálformák (1NF, 2NF, 3NF, BCNF).
- Többértékű függés, magasabb normálformák. (4NF, 5NF)
- Relációs műveletek. Relációs algebra.
- Az SQL nyelv. A SELECT utasítás. Csoportképző függvények. Táblák összekapcsolása. Beágyazott lekérdezések. Halmazműveletek.
- Nézetek (View)
- Az SQL DDL, DML, DCL utasításai.
- Adatbázis szerepkörök. Jogosultságok. Adatbázismentés, visszatöltés.
- Az ER modell kiterjesztése, EER modell. EER modell leképezése relációs modellre.
- Objektumorientált adatbázisok.

Az Adatbázisok II. tantárgy oktatására a negyedik félévben kerül sor. A tárgy oktatása heti 2 óra előadás és 2 óra laborgyakorlat formájában történik. A tárgy félévközi jeggyel zárul.

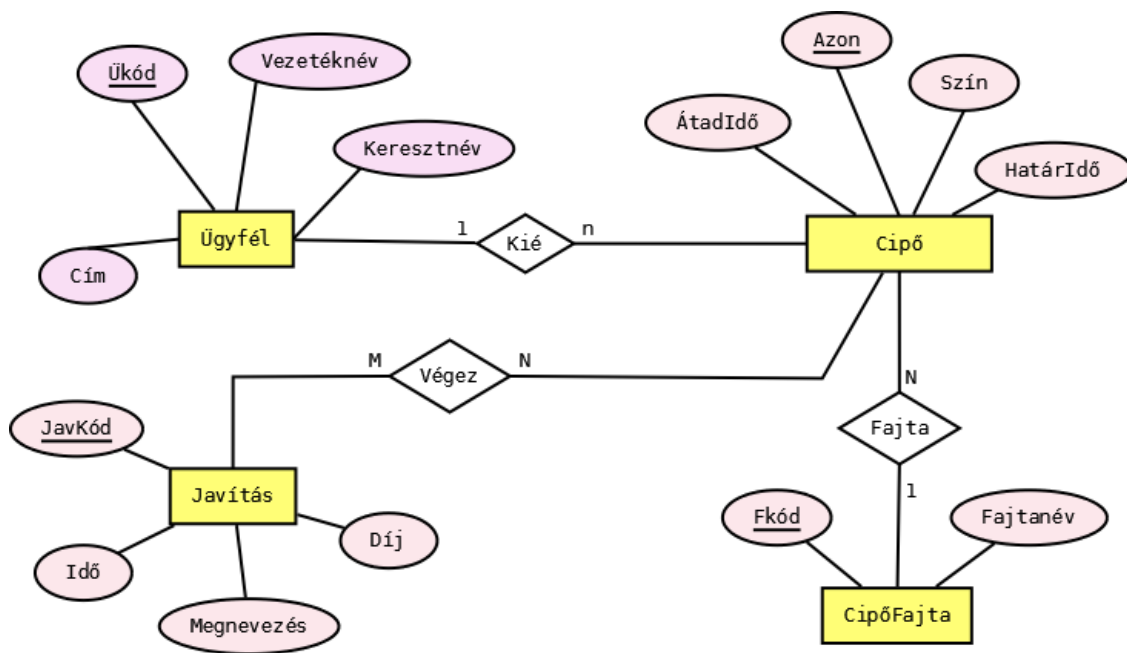
Az Adatbázisok II. tantárgy fő témakörei

- Térbeli információk kezelése SQL-ben (geometriai és geográfiai adatok).
- Adattárolás, szerver architektúra.
- Tárolt eljárások. Vezérlő szerkezetek (IF, ciklus stb.)
- Kurzorok. Felhasználói függvények.
- Triggerek.

- Tranzakciók. Tranzakciók tulajdonságai. Konkurenciakezelés. Zárolási technikák.
- Hatékonyság. Tuning, indexek.
- Programozás, ADO.NET
- DataReader, DataSet használata, kapcsolatalapú és kapcsolat nélküli modell.
- LINQ
- Entity Framework

3. Az oktatás menete, a tanulás támogatása

Az Adatbázisok I. tárgy keretében az előadásokon elhangzott témakörök elmélyítésére a gyakorlatokon kerül sor. A hallgatók egyedül, önállóan oldják meg a feladatokat, azonban összetett, szöveges modellezési feladatok esetén 2-3 fős csoportokban dolgoznak. A megoldásokat közösen megbeszéljük. Az egyed-kapcsolati modellek elkészítése a DIA nevű programmal történik. (lásd 1. ábra) Az ábrán bemutatott példát a [1] könyvből vettük be a tananyagba. Ezekre a szöveges feladatokra nagy hangsúlyt fektetünk, hiszen a szövegekörnyezetből a lényeget kiemelve, az egyed típusokat s azok tulajdonságait meghatározni, az egyedek közötti kapcsolatokat felismerni, s a kapcsolatok tulajdonságait meghatározni sok hallgató számára igen nehéz feladatnak bizonyul.



1. ábra. Egy cipésműhely működésének ER modellje

Vannak olyan feladatok, amelyek többször felbukkannak a félév során. Ilyen az előző ábrán látható feladat is. A koncepcionális tervezéskor elkészül az ER modell, majd később leképezik a hallgatók relációs modellé. A normálformák elsajátításakor szintén előkerülhetnek ezek a feladatok, megvizsgálhatják, hogy megfelelnek-e a különböző normálformáknak. Az SQL nyelv tanulásának szakaszában pedig a korábban már elkészített relációs modell megvalósítására kerül sor, létrehozzák a hallgatók a megfelelő adattáblákat, feltöltik adatokkal, majd különböző lekérdezési feladatokat oldanak meg. Természetesen minden témakör elsajátításakor új feladatokat is meg kell oldani.

Az SQL nyelv használatának gyakorlásához a laborokban az MS SQL szerver Developer változata van telepítve, minden gépre külön-külön. A hallgatók a DreamSpark előfizetésnek köszönhetően az otthoni számítógépükre is telepíthetik ezt a szoftvert, így könnyebb számukra a házi feladatok megoldása is. Az MSSQL szerverrel szállított mintaadatbázisokat is felhasználjuk az oktatásban.

Milyen módon támogatjuk a tananyag jobb elsajátítását? Az előadások látogatása ugyan nem kötelező, de erősen ajánlott. Sajnos a hallgatók nagy része nem jegyzetel, csak beül az előadásra füzet és toll nélkül. Az előadásokon vetített anyagot ezért nagyon fontos, hogy átadjuk a hallgatóknak. Korábban egy külön ftp szerverre raktuk fel a tananyagot, de amióta a Coospace rendelkezésre áll ezt sokkal szívesebben használjuk. Az előadásokon a figyelem felkeltésére időnként különböző „játékos” feladatok megoldását építjük be az anyagba. Ezek a rövid teszt jellegű játékok vagy az eXeLearning vagy a Microsoft LCDS szoftverrel készülnek. Mindkét szoftver ingyenesen letölthető az internetről. Többnyire sorrendbe rakás, osztályozás, szétválogatás, vagy puzzle kirakójátékokat használunk egyrészt az ismeretek megszerzésének ellenőrzésére, másrészt a fontos tudnivalók elmélyítésére. Ezen kívül különböző gyakorló tesztek is feltöltünk a tudás ellenőrzésére.

Egy nagyon egyszerű példán keresztül szeretném szemléltetni az előbb említett interaktív játékok beillesztésének hasznosságát. Előadáson részletesen beszélünk arról, hogy hogyan történik az SQL SELECT utasításának kiértékelése. Az előadás anyagában ez az alábbi PowerPoint dián látható (lásd 2. ábra) is.

A lekérdezés végrehajtási folyamata

Egy lekérdezésben szereplő utasítás elemeket az SQL szerver nem a leírás sorrendjében hajtja végre .

5:	SELECT	<select lista>
1:	FROM	<tábla(k)>
2:	WHERE	<szűrési feltétel>
3:	GROUP BY	<csoportosító mező(k)>
4:	HAVING	<csoport szűrési feltétele>
6:	ORDER BY	<rendezési lista>

2. ábra SELECT kiértékelési sorrendje

Tekintsünk egy egyszerű gyakorlaton megoldott lekérdezési feladatot: *Listázzuk ki a Northwind adatbázis Products táblájából a P kezdőbetűs termékek kódját és nevét! Az eredményben ProductName helyett a Terméknév szerepeljen az oszlop fejlécében, az eredménylistát a termék neve szerinti sorrendben írjuk ki a képernyőre.* Tipikus hallgatói megoldás látható a következő ábrán (lásd 3. ábra), amely hibás.

```

SELECT ProductID, ProductName as Terméknév
FROM Products
WHERE Terméknév LIKE 'P%'
ORDER BY Terméknév;
    
```

Msg 207, Level 16, State 1, Line 3
Invalid column name 'Terméknév'.

3. ábra. A feladat hibás megoldása

Ha a hallgató emlékezett volna az előadáson elmondottakra, akkor a WHERE záradékban nem az alias névvel (Terméknév), hanem a ProductName mezőnévvel hivatkozott volna a termék nevére, és helyes lenne a megoldása (lásd 4. ábra).

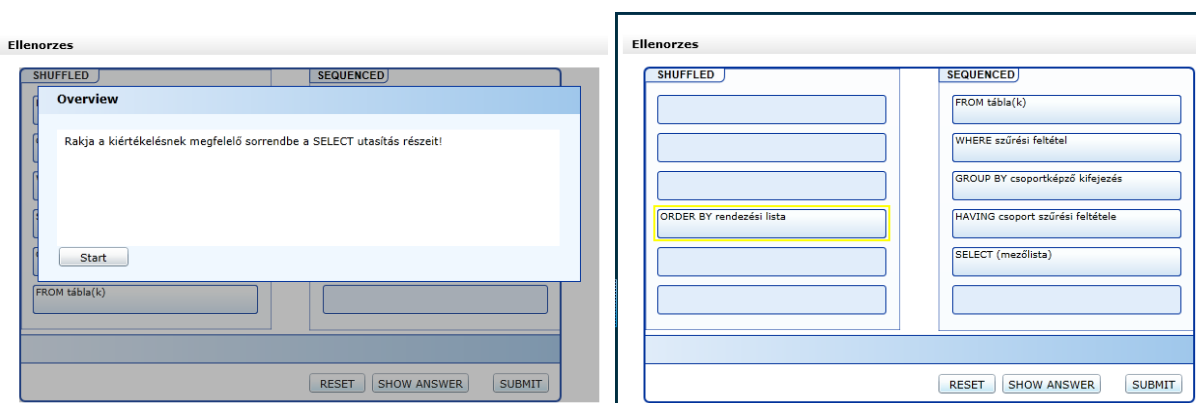
```

SELECT ProductID, ProductName as Terméknév
FROM Products
WHERE ProductName LIKE 'P%'
ORDER BY Terméknév;
    
```

	ProductID	Terméknév
1	55	Pâté chinois
2	16	Pavlova
3	53	Perth Pasties

4. ábra. A feladat helyes megoldása

Többek között ez a tipikus és igen gyakran előforduló hiba készített arra, hogy készítsék egy játékot, amelyben pont a második ábrán látható tudnivalókat lehet begyakorolni. (lásd 5. ábra). A játék indulásakor a képernyő bal oldali részén, a gép véletlen sorrendbe rakja a SELECT utasítás részeit tartalmazó kártyákat, s a hallgatónak a jobb oldali területre kell áthúznia ezeket, a helyes sorrendnek megfelelően. Úgy vélem ilyen játékos formában jobban megjegyzik a hallgatók, a fontos tudnivalókat, mintha csak felsorolnánk egy dián.














5. ábra Játék: Helyes sorrend meghatározása

Ezek az anyagok a Coospace-be feltölthetők, s mivel a Coospace-hez most már mobil eszközökről is lehet kapcsolódni, így bármikor, ha kedve támad a hallgatónak megoldhatja a játékokat. A gyakorlatokon kiadott feladatsorok megoldását, vagy iránymutatást a megoldáshoz szintén a Coospace felületén keresztül adunk a tanulóknak.

Az Adatbázisok II. tantárgy épít az Adatbázis I. kurzuson elsajátított anyagra. Mivel félévközi jegyes tantárgyról van szó, a félév közben végzendő folyamatos munkán van a hangsúly. Az ismeretek elsajátításának ellenőrzésére három dolgozatot írunk. Vannak határidőre beadandó kisebb házi feladatok is. A félév végére egy összetett beadandó feladatot is el kell készíteni, amelyet minden hallgató maga talál ki. A beadott munkának tartalmaznia kell a normalizált adatmodellt, létre kell hozni és be kell adni az adatbázist, feltöltött táblákkal, és készíteni kell egy Windows Form alkalmazást C# nyelven, amely az adatok karbantartását végzi, különböző szűréseket adatlistázást, adatmódosítást, törlést valósít meg. A jegy a dolgozatok és az önálló projektmunka eredményét figyelembe véve alakul ki.

Fontos megemlíteni, hogy a témakörök közé be kellett illesztenünk a geometriai és geográfiai új adattípusokat, hiszen a modern alkalmazások megkívánják, hogy földrajzi koordinátákkal dolgozzunk, geometriai adatokat tároljunk (lásd 6. ábra). Ez a lehetőség az MS SQL Server2008 óta áll rendelkezésre.[2] [4] Az SQL Server Management Studio lehetővé teszi, hogy az ilyen típusú adatokat lekérdezésének eredményét egy külön lapon megtekinthessük (lásd 7.-9 ábra). Az ábrákon néhány, a gyakorlatokon megoldott feladat eredménye látható.

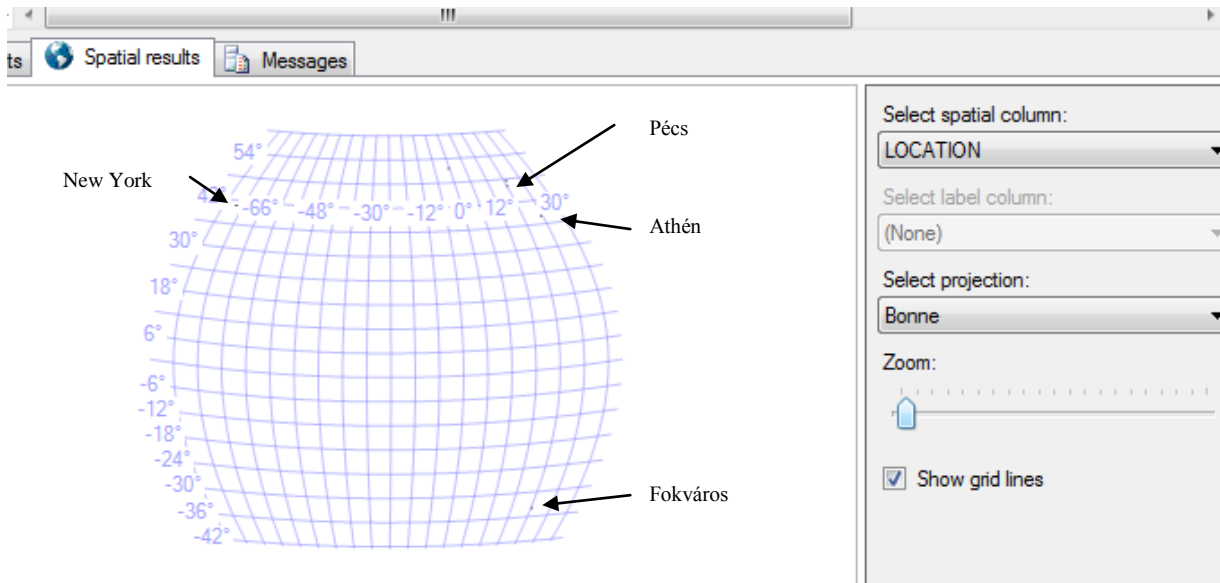
POINT		MULTIPOINT	
LINestring		MULTILINESTRING	
CIRCULARSTRING		COMPOUNDCURVE	
POLYGON		MULTIPOLYGON	
CURVEPOLYGON		GEOMETRY-COLLECTION	
FULLGLOBE			

6. ábra Geometriai adattípusok

```
CREATE TABLE City ( CityID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
Nev nvarchar(12), LOCATION Geography);

INSERT City (Nev, LOCATION)
VALUES('Budapest', 'POINT(19.064819 47.506221)')
INSERT City (Nev, LOCATION)
VALUES('Pécs', 'POINT(18.307489 46.126202)')
INSERT City (Nev, LOCATION)
VALUES('London', 'POINT(-0.127140 51.506321)')
INSERT City (Nev, LOCATION)
VALUES('Athén', 'POINT(23.736410 37.976150)')
INSERT City (Nev, LOCATION)
VALUES('NEW York', 'POINT(-74.007118 40.714550)')
INSERT City (Nev, LOCATION)
VALUES('Fokváros', 'POINT(18.421989 -33.919090)')
```

7. ábra Város tábla létrehozása és feltöltése geográfiai adattípussal



8. ábra A Város tábla Location adatmezőinek megjelenítése SQL Server Management Studioban

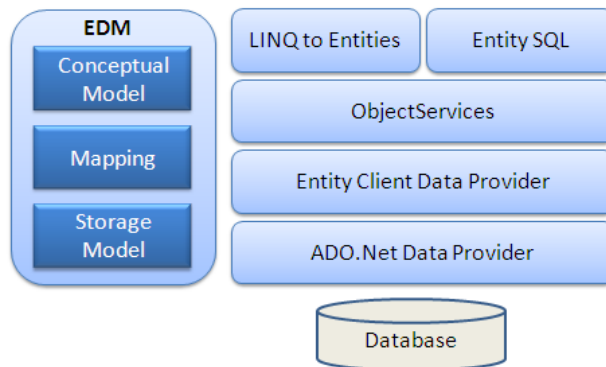
```

SELECT
  C1.Nev AS Város1,
  C2.Nev AS Város2,
  ROUND(C1.Location.STDistance(c2.location) / 1000, 2) AS Km
FROM
  City AS C1 JOIN City AS C2 ON C1.CityID < C2.CityID
ORDER BY C1.CityID
    
```

	Város1	Város2	Km
1	Budapest	Pécs	163,94
2	Budapest	London	1454,51
3	Budapest	Athén	1125,11
4	Budapest	NEW York	7027,36
5	Budapest	Fokváros	9018,44
6	Pécs	London	1474,88
7	Pécs	Athén	1010,07
8	Pécs	NEW York	7059,6
9	Pécs	Fokváros	8864,81
10	London	Athén	2396,11
11	London	NEW York	5585,27
12	London	Fokváros	9635,63
13	Athén	NEW York	7945,69
14	Athén	Fokváros	7978,51

9. ábra Az STDistance() függvény használata a távolságok kiszámítására

A félév során a hallgatók megismerkednek a szerver oldali programozással, tárolt eljárások, kurzorok használatával. Készítenek triggeret, függvényeket. Fontos témakör a több-felhasználós elérés, a tranzakciók kezelése. A félév második felében készítünk menüvezérelt, Windows Form alkalmazásokat. Az ADO.NET szolgáltatásainak, osztályainak megismerése jelent komoly kihívást a hallgatók számára. Meg kell ismerniük a kapcsolat és kapcsolat nélküli működés közötti különbséget, mikor melyiket célszerű használni. Jártasságot kell szerezniük a DataReader és a DataSet osztályainak használatában. A relációs adatmodell és az objektumorientált programozási nyelvek közötti leképezési problémák megoldására fejlesztették ki az Entity Framework technológiát. Az Entity Framework hivatott arra, hogy a relációs adatmodellt leképezze objektum modellé.



10. ábra. Az Entity Framework architektúrája [3]

El kell sajátítaniuk a hallgatóknak a LINQ lekérdező nyelv által biztosított lehetőségeket is. Ezeknek a témaköröknek az elsajátítását is támogatjuk. Az előadások anyaga és mintaprojektjei felkerülnek a Coospace-ba. Ezen kívül minden gyakorlat feladatainak megoldásához készítünk 12-15 oldalas részletes leírást, útmutatást. A korábban említett eszközök segítségével kisebb eLearning anyagok is készülnek. A nehezebb programozási feladatok megoldásának bemutatására pedig Camtasia Studióban készült video anyagokat is készítünk, amelyben a képernyőn folyó programozási munka kerül rögzítésre a tanári magyarázattal (hang) együtt. Ez is felkerül a Coospaceba.

4. Továbbfejlesztés

A törzsanyagban kötelezően elsajátítandó Adatbázisok I és Adatbázisok II tantárgyakon túl lehetőségük van a hallgatóknak arra, hogy érdeklődésüknek megfelelően bővítsék ismeretüket az adatbázisokkal kapcsolatos témakörökben. Szabadon választható tantárgyi blokkban MS SQL szerver adminisztrátori ismeretekre tehetnek szert. Az IBM DB2 fakultációban pedig az IBM adatbázis-kezelőjében mélyülhetnek el. Sajnos ez a két szabadon választható kurzus az utóbbi négy évben nem indult, de reméljük, hogy hamarosan ismét felvehetik a hallgatók. Az ORACLE Academy programhoz kapcsolódva szabadon választható tárgyként szándékozunk ORACLE interneten keresztül elvégezhető kurzusokat felkínálni az érdeklődők számára. Fontos a tananyagok folyamatos továbbfejlesztése, a legközelebbi feladat az, hogy a Big Data és NoSQL témakörök is beépüljenek valamilyen formában a BSC képzésbe.

Irodalomjegyzék

- [1] Tímár Lajos, Vigh Krisztina és Szigeti Judit: Így még könnyebb az adatmodellezés, *Veszprémi Egyetemi Kiadó* (2005), 42-46.
- [2] Richard Leonard Lobel és Andrew Brust: Programming Microsoft SQL Server 2012, *Microsoft Press* (2012), ISBN: 978-0-7356-5822-6
- [3] <http://www.entityframeworktutorial.net/EntityFramework-Architecture.aspx> , Látogatva 2014.01.03
- [4] Ross Mistry és Stacia Misner: Introducing Microsoft SQL Server 2012, *Microsoft Press*, (2012), ISBN: 978-0-7356-6515-6

Szoftverfejlesztés (programozás) az Informatikus Szakigazgatási Agrármérnök szakon

Software development at Agricultural Engineering in Agricultural Computing and Policy Administration BSc

Bakó Mária^a, Várallyai László^b

^aDebreceni Egyetem, GVK
bakom@unideb.hu

^bDebreceni Egyetem, GVK
varal@agr.unideb.hu

Absztrakt: A Debreceni Egyetem Gazdálkodástudományi és Vidékfejlesztési Karán a 2006/07-es tanévben indult el az Informatikus és Szakigazgatási Agrármérnök BSc szak. Diákjainknak lehetőségük van választani az Informatikus és a Szakigazgatási szakirány között. Mivel napjainkban az informatika végzettséggel rendelkező diákok elhelyezkedéséhez elengedhetetlen bizonyos szintű programozási ismeret, cikkünkben elemezni szeretnénk, hogy diákjaink milyen programozással kapcsolatos ismeretekre tehetnek szert. Rá szeretnénk mutatni arra, hogy a programozáshoz kapcsolódó tantárgyaink oktatásakor milyen nehézségekbe ütközünk, ezeket hogyan kezeljük; és hogyan lehetne még hatékonyabban kezelni a felmerülő problémákat. Három programozáshoz kötődő tantárgyunk van a tantervben: a Szoftverfejlesztés, az Internet alkalmazásfejlesztés és a Szakértői rendszerek. A Szoftverfejlesztés tárggyal kapcsolatban a leglényegesebb probléma az, hogy nem előzi meg olyan tárgyat, ahol a hallgatók algoritmizálási képességeit lehetne megalapozni. Hasonlóképpen hiányoznak a logikai előismeretek is, így ezen tantárgyak oktatása során igen sok időt elvesz a tantárgyhoz szükséges alapismeretek elsajátítása, így alig jut idő a tematikában felsorolt ismeretekre. Egyre sürgetőbb a felmerült problémákra megoldást találni, mert az MSc képzés során két tárgy is a szoftverfejlesztési ismeretekre épül, illetve a Szakértői rendszerek tárgynak is van egy folytatása ezen a szinten.

Kulcsszavak: szoftverfejlesztés, Visual C#, Prolog

Abstract: The Faculty of Economics and Rural Development of the University of Debrecen launched the Agricultural Engineering in Agricultural Computing and Policy Administration BSc in the 2006-07 academic year. Our students have the opportunity to choose between the fields of Agricultural Computing and Policy Administration. Nowadays companies require a certain level of knowledge in programming for students with a degree in computer science. In our paper we would like to analyze which kind of programming knowledge can acquire our students. We would like to point out to the difficulties related to the teaching of the subjects connected to programming, how we treat it; and how we can deal more effectively with the arising problems. Only three subjects of the curriculum are related to programming: Software development, Internet application, and Expert systems. One of the most important problems regarding the Software development subject is that it is not preceded by a subject where student's capabilities to creating algorithms could be improved. Similarly, the students have a lack of knowledge in mathematical logic, so we lost a lot of time teaching these basic elements and we do not have sufficient time to teach the necessary curricular elements. It is increasingly urgent to find a solution to these problems, because on the MSc level there are two subjects which are based on the Software development knowledge and the Expert systems also have a continuation on this level.

Keywords: software development, Visual C#, Prolog

1. Bevezetés

Az Informatikus és Szakigazgatási Agrármérnök BSc szakalapítási kérelmét 2004-ben nyújtottuk be közösen a Budapesti Corvinus Egyetemmel, a Szent István Egyetemmel és másik hat egyetemmel együttműködve. Mintegy négy hónappal később beadásra került a szakindítási kérelem is, amelyet a MAB pozitívan bírált el – csak úgy, mint a szakalapítási kérelmet –, így a Debreceni Egyetem Gazdálkodástudományi és Vidékfejlesztési Karán a 2006/07-es tanévben indult el az Informatikus és Szakigazgatási Agrármérnök BSc szak. Diákjainknak lehetőségük van választani az Informatikus és a Szakigazgatási szakirány között. Mivel napjainkban az informatika végzettséggel rendelkező diákok elhelyezkedéséhez elengedhetetlen bizonyos szintű programozási ismeret, cikkünkben elemeznénk, hogy diákjaink milyen programozással kapcsolatos ismeretekre tehetnek szert. Rá szeretnénk mutatni arra, hogy a programozáshoz kapcsolódó tantárgyaink oktatásakor milyen nehézségekbe ütközünk, ezeket hogyan kezeljük; és hogyan lehetne még hatékonyabban kezelni a felmerülő problémákat. Három programozáshoz kötődő tantárgyunk van a tantervben: a Szoftverfejlesztés, az Internet alkalmazásfejlesztés és a Szakértői rendszerek.

A félév során összeállított kérdőívvel felmértük diákjaink véleményét a programozási tantárgyakkal kapcsolatban, hogy segítségükkel a megfelelő lépéseket fogantassuk ezen tantárgyak oktatásának a javítására. Egyre sürgetőbb a felmerült problémákra megoldást találni, mert az MSc képzés során két tárgy is a szoftverfejlesztési ismeretekre épül, illetve a Szakértői rendszerek tárgynak is van egy folytatása ezen a szinten.

A következő két fejezetben a Szoftverfejlesztés és Internet alkalmazásfejlesztés és a Szakértői rendszerek oktatásában felmerülő konkrét problémákat ismertetjük. A 4. fejezet a kérdőív kiértékelését az utolsó fejezet pedig az összefoglalást tartalmazza.

2. Szoftverfejlesztés és Internet alkalmazásfejlesztés

2006/07-es tanévben, a képzés indításakor, a szoftverfejlesztés tantárggyal a hallgatók a második félévben találkozhattak. Ehhez a tantárgyhoz 2006-ban egy HEFOP pályázat keretében jegyzet is készült a Corvinus és a Debreceni Egyetem munkatársainak részvételével [3]. Azonban hamarosan rá kellett jönnünk, hogy a tantárgy második félévben történő oktatása két okból sem megfelelő. Az egyik, hogy az alapozó informatikai ismeretek két féléve még nem fejeződik be addigra, így egy „gyengébb” előképzettséggel rendelkező hallgatónak nem sok esélye maradt a tantárgyat sikeresen teljesíteni. A másik, hogy két tárgy is szorosan kapcsolódik illetve épül a szoftverfejlesztési ismeretekre az informatikus szakirányon (Internet alkalmazásfejlesztés és Szakértői rendszerek), valamint túl sok idő telik el a két tantárgy között programozással kapcsolatos tantárgy nélkül. Ezért kérelmeztük, hogy a szoftverfejlesztés tantárgy átkerülhessen a harmadik félévre, amellyel mindkét problémát kiküszöbölhetjük egyidejűleg. Ezt a Kari Tanács engedélyezte is négy évvel ezelőtt. Az említett szakirányos két tantárgynál már nem szerepel a tematikában a C programnyelv megismerése, hiszen ez már a szoftverfejlesztés során elsajátításra került, viszont az időközben megismert adatbáziskezelésre annál inkább hangsúlyt helyezünk. Az Internet alkalmazásfejlesztés során a PHP nyelvet ismertetjük meg a hallgatókkal, adatbáziskezelőként az ingyenes MySQL rendszert használjuk [4],[5]. Az óraszám sajnos kevés ahhoz, hogy e tárgy keretében bővebb kitekintést adjunk a hallgatóknak, hiszen meg kell ismerniük a HTML leírónyelvet, a CSS stílusnyelvet, a kliensoldali dolgok használatához a JavaScript-et és az AJAX technológiát.

A Szoftverfejlesztés tárgyval kapcsolatban azonban a legnagyobb probléma az, hogy nem előzi meg olyan tárgy, ahol a hallgatók algoritmizálási képességeit lehetne megalapozni. Így a tárgy keretében a grafikus felület adta előnyöket kihasználva kisebb programok, programegységek megírására nyílik mindössze lehetőség. A tárgyat záró gyakorlati vizsgán is hasonló nagyságrendű program elkészítését várjuk el. A félév során a hallgatók több közepes nagyságrendű feladatot is kapnak, amelyeket személyre szabottan adunk ki, azaz mindenki más-más három feladatot készít el a félév során. Az elkészült feladatokat a szorgalmi időszak végéig fel kell tölteniük a Moodle e-Learning rendszerbe. Erre is kapnak jegyet a hallgatók még ha ez kisebb súllyal is szerepel az átlagukban, mint a félév végén megoldott feladat eredménye. Természetesen a feladatba történő belekérdezéssel könnyen kideríthető, ki az, aki maga készítette el a feladatot és ki az, aki másnak a segítségét vette igénybe. Az elméleti számonkérések eredménye is érdekes, hiszen a félév során két jegymegajánlót is írnak, de mégis nagyon kevesen élnek ezzel a lehetőséggel. Jól látható az eredményekből, hogy a kétféle típusú kérdésből a több felelet választósakat még megpróbálják megtippelni, de az esszé részhez már nem írnak semmit. Ebből tisztán látszik, ki az, aki nem készült és esszé pontok nélkül nem lehet meg az elégséges szint. Érdekes módon a szóbeli vizsgára már meg tudják tanulni korrektül az elméleti anyagot. Eredetileg a HEFOP pályázat keretében készült jegyzet esetében vannak olyan fejezetek, amelyek a szoftver tervezését segítik, de erre sajnos idő hiányában nem tudunk sort keríteni a gyakorlatok keretében. Megoldás lehetne, ha az első féléves alapozó informatika tárgy keretében lehetőség nyílna az említett algoritmizálási készségek fejlesztésére. Azért is fontos erre hangsúlyt fektetni, hiszen az új MSc képzésünk keretében két tantárggyal is találkozhatnak a hallgatók. Az egyik a Szoftverfejlesztési technológiák, a másik az Agrárspecifikus alkalmazási szoftver rendszerek fejlesztése. A hallgatóknak ezt a hiányzó algoritmizálási képességét mindenképpen meg kell oldani, mert az MSc képzés keretében nem fogják tudni teljesíteni ezeket a tárgyakat. Mivel itt már nagyobb rendszerek tervezését, algoritmizálást és kódolást kell egyidejűleg alkalmazniuk a két tárgy keretében.

3. Szakértői rendszerek

A Szakértői rendszerek tantárgy oktatására a hatodik félévben kerül sor. Hasonlóképpen a Szoftverfejlesztés tantárgyhoz, itt is hiányoznak a tantárgy oktatásához szükséges előismeretek. A képzés során diákjaink oktatásából kimarad a Matematikai logika oktatása és középiskolából hozott tudásuk nem elegendő ahhoz, hogy erre építhessük a Prolog illetve Clips programozási nyelveket. A tantárgy oktatása során igen sok időt elvesz a tantárgyhoz szükséges alapismeretek elsajátítása, így alig jut idő a tematikában felsorolt ismeretekre.

A tantárgy alapvető célkitűzése az, hogy a hallgatók a képzés során ismerjék meg a szakértői rendszerek helyét a mesterséges intelligencia tudományterületén belül. Ehhez sajátítsanak el néhány alapvető ismeretrepresentációs módszert és az azokhoz kapcsolódó következtetési technikákat és algoritmusokat. Továbbá legyenek képesek a gyakorlatban is hasznosítani a megszerzett elméleti ismereteket. Ismerkedjenek meg a Prolog és Clips logikai programozási nyelvek alapfogalmaival és ebben implementáljanak egyszerű szakértői rendszereket.

A logikai programozási nyelvek elsajátításához elengedhetetlen a logikai következmény fogalmának megértése és készség szintű alkalmazása. Azonban a logikai következmény fogalmának megértéséhez diákjainknak ismerniük kell az alapvető logikai operátorokat, azok működését és tudniuk kell formalizálniuk különböző állításokat. Ami a szakértői rendszerek szempontjából nagyon fontos, hogy a logika segítségével ellenőrizhetjük magunk és mások érvelésének helyességét. A logika nem tudja megmondani, hogy adott premisszákból mire kell következtetni. Arra viszont tud válaszolni, hogy az adott állításoknak logikai

következménye-e egy további állítás, és mi az, ami nem következik. Diákjaink azonban a matematika logika fogalmával, ezen belül is csak a logikai összekötőjelekkel találkoztak középiskolában. Annak ellenére, hogy a harmadik félévben a szoftverfejlesztés tantárgy keretén belül ez felelevenítésre került, a tapasztalatunk azt mutatja, hogy a hatodik félévre a diákok egyáltalán nem emlékeznek ezekre az alapfogalmakra. A logikai alapfogalmak elsajátítására e-learning tananyag készült, amelynek segítségével megpróbáljuk felgyorsítani a logikai alapismeretek elsajátítását [1], [2]. A tananyag tartalmazza az elméleti alapfogalmakat a nulladrendű és az elsőrendű logikai nyelvekről, továbbá számos gyakorló feladatot a nulladrendű logika elsajátításához. Elsőrendű logikában a formalizálást, illetve azt, hogy adott állításoknak egy állítás logikai következménye-e, rengeteg gyakorló feladattal a táblánál próbálják meg elsajátítani a diákok.

A tantárgy fő célkitűzése azonban az lenne, hogy a diákok készség szintjén elsajátítsák a Prolog és a Clips programozási nyelveket, azonban a logikai alapozás olyan mértékben elviszi az időt, hogy a programnyelvekre alig négy dupla gyakorlati óra szokott maradni. Ez azonban csak egy betekintésre, a nyelvek alapjainak elsajátítására elegendő. Általában csak a családi kapcsolatok formalizására marad idő, az egyszerű szakértői rendszerek, osztályozási feladatok már nem férnek be az időkeretben.

A legelkeserítőbb a szakértői rendszerek oktatásában az, hogy egy tantárgy időkeretén belül, gyakorlatilag kénytelenek vagyunk két tantárgyat oktatni. Ennek az a következménye, hogy gyakorlatilag sem a matematikai logika sem pedig a logikai programozási nyelvek alapos oktatására nincsen elég időnk.

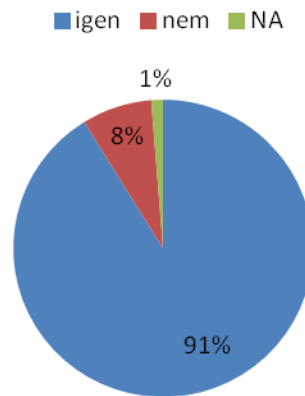
Az alapvető logikai összekötőjelek oktatásakor az implikáció fogalmát, ami később a logikai következménynél is igen fontos szerepet játszik, nehezen tanulják meg a diákok. Az igazi nagy problémába viszont a formalizálásnál ütközünk. A diákjaink gyakorlatilag ennyi idő alatt képtelenek megérteni a létezik és a bármely fogalmát és gyakorlatilag alig 20%-uk képes valamilyen szinten formalizálni elsőrendben egy állítást.

Tovább nehezíti a dolgunk az, hogy a diákok eddig csak imperatív programozási nyelvekkel találkoztak, így a Prolog – mint deklaratív programozási nyelv – számukra nagyon furcsa és nehezen érthető. Egy program megírásához diákjainknak elsősorban arra van szükségük, hogy átlássák a különböző elemek közötti logikai kapcsolatokat és képesek legyenek az állítások formalizálására. Erre azonban csak akkor lesz esélyük, ha sikerül a logikát a szakértői rendszerek előtti félévben oktatnunk.

4. A diákok véleménye a programozás oktatásáról

A félév során a korábbi tapasztalatokra támaszkodva összeállítottunk egy rövid kérdőívet, amelyből képet szeretnénk volna kapni a felvetődött kérdésekre a programozási tantárgyakkal kapcsolatban. A kérdőívben alkalmaztunk mind a zárt, mind pedig a nyílt kérdéseket, hogy pontosabb képet kaphassunk a hallgatók véleményéről az adott kérdésben. A kapott válaszokat feldolgozva diagramos és táblázatos formában mutatjuk be az eredményeket.

Az első legalapvetőbb kérdésünk az volt, hogy "Munkahelykereséskor hasznos-e, ha rendelkezik valaki programozási ismeretekkel?" A kérdőívet kitöltő hallgatók 91 százaléka (1. ábra) válaszolta azt, hogy hasznos. Ez szinte teljes mértékben egybevág a munkaerő-piaci tapasztalatokkal valamint az általunk felvázolt hipotézissel. A megkérdezett hallgatók mintegy 8 százaléka szerint viszont nem hasznos munkahelykereséskor a programozási ismeretek megléte. Náluk joggal vetődik fel a kérdés, hogy milyen munkaerő piaci szemléletet képviselnek.



1. ábra. Munkahelykereséskor hasznos-e, ha rendelkezik valaki programozási ismeretekkel?

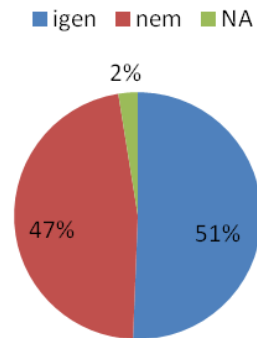
A kérdőívben megkérdeztük a hallgatókat arról is, hogy milyen programozási nyelveket ismernek. Ezt nyílt kérdésként tettük fel, tehát a hallgatók szabadon írhattak be programozási nyelveket. Ennek eredménye látható az alábbi táblázatban.

1. Táblázat Programozási nyelvek és ismertségük

Programozási nyelv	Darabszám
C#	28
JAVA	16
PROLOG	11
PASCAL	9
PHP	8
C++	7
DELPHI	5
VISUAL BASIC	2

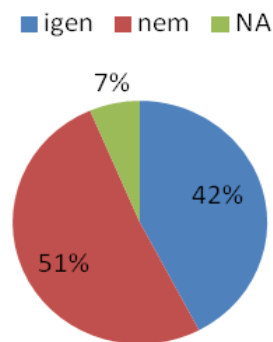
Az egyértelműen látszik a táblázat adataiból, hogy a C-alapú nyelvek viszik a pálmát. Ez több okra vezethető vissza. Egyrészt ezeken a nyelveken oktatjuk a hallgatókat, másrészt a munkaerőpiac is ezt követeli meg tőlük. A kettő természetesen szorosan összefügg egymással, hiszen ezen elv alapján választottuk ki az akkreditáció során az oktatandó nyelvet.

A következő kérdésünk arra vonatkozott, hogy használták-e már a hallgatók az előző táblázatban feltüntetett nyelvet vagy nyelveket (2. ábra). Természetesen az órán kívüli használatra voltunk kíváncsiak. Nos, ez meglepő eredményt mutatott, hiszen a megkérdezett hallgatóknak a fele már órán kívül is használta a megjelölt nyelvet fejlesztésre.



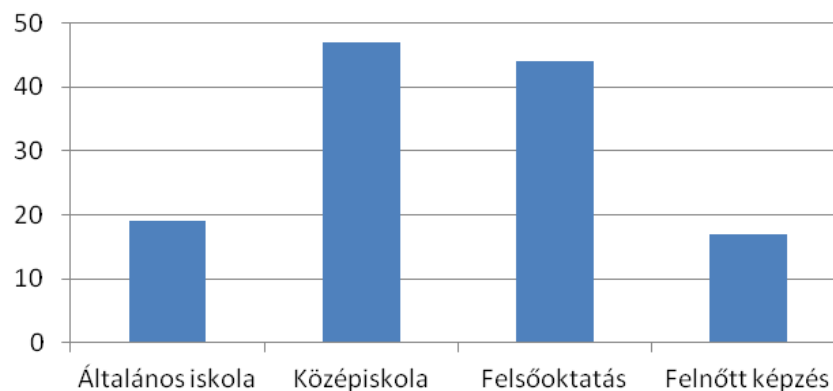
2. **ábra.** Használta-e az Ön által megjelölt nyelvet fejlesztésre az órai tevékenységen kívül?

A kérdőívünk következő kérdésére adott válaszok egy kicsit ellentmondásosnak tűnnek számunkra, mivel a hallgatók arra a kérdésre, hogy szeretnék-e más programozási nyelvet megtanulni, többségükben azt válaszolták, hogy *nem*. Ez különösen érdekes az 1. ábra adatainak ismeretében illetve annak összevetésében.



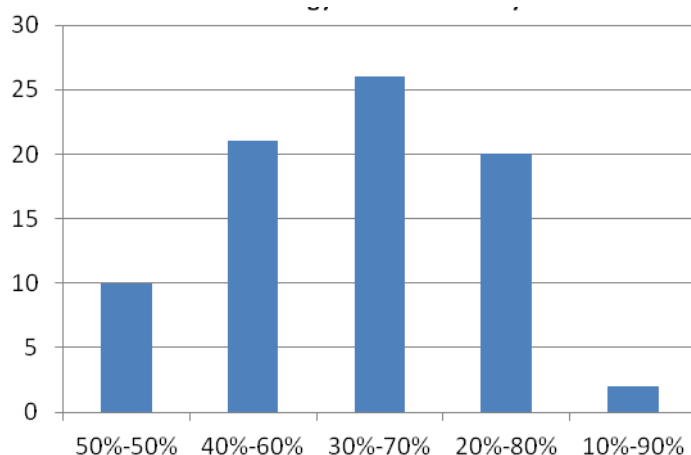
3. **ábra.** Szeretne-e más programozási nyelveket megtanulni?

A kérdőívünkben arra is kíváncsiak voltunk, hogy a hallgatók szerint melyik iskolai szinten lenne érdemes a programozást leginkább elkezdni tanítani.



4. **ábra.** Ön szerint az oktatás melyik szintjén kellene programozást tanítani?

A 4. ábra adatai alapján egyértelmű a hallgatók véleménye, hogy a középiskola és a felsőoktatás a programozás igazi terepe. Emellett persze érdemes említést tenni arról, hogy már általános iskolákban néhány kolléga elhivatottnak érzi magát a programozás oktatásában, csak sajnos az esetek többségében a diákok oldaláról nincs meg még a kereslet. A különböző szoftverkészítő cégek is igyekeznek olyan „diákközelí” programozási nyelveket kialakítani „játékos” környezettel, hogy kedvet csináljanak a diákok számára a programozáshoz (pld. Python, Comenius Logo). Sajnos ebben a korban úgy próbálják programozásra tanítani a diákokat, hogy nem rendelkeznek még egyáltalán algoritmizálási készséggel és képességgel. Ez sajnos több helyen a középiskolában, sőt még a felsőoktatásban is előfordul. A diákok gyors eredményt szeretnének elérni a programozásban, persze algoritmikus készségek nélkül. Arra is kíváncsiak voltunk, hogy a hallgatók szerint mennyi lenne az ideális elméleti-gyakorlati arány a programozás oktatásban.



5. ábra. Ön szerint programozásból mennyi lenne az ideális elméleti-gyakorlati arány?

Az 5. ábráról leolvasható, hogy nagyjából 1/3-2/3 lenne az ideális arány, amivel egyet is lehet érteni, hiszen nagyon fontos, hogy a hallgatók az elméleti alapokkal tisztában legyenek, de ami még lényegesebb ennél is, hogy mindezeket az ismereteket a gyakorlatban is tudják kamatoztatni, hiszen végzés után a munkaerőpiacon erre lesz elsősorban szükségük.

Kérdőívünk utolsó kérdése arra vonatkozott, hogy a hallgatók szerint milyen programozási nyelvet (nyelveket) kellene tanítani a felsőoktatásban. Érdekes módon az 1. táblázatban feltüntetett adatokat láthattuk viszont, ami véleményünk szerint egyfajta visszaigazolást jelent számunkra hallgatói oldalról.

5. Összegzés

A Szoftverfejlesztés tárggyal kapcsolatban véleményünk szerint az a leglényegesebb probléma, hogy nem előzi meg olyan tárgyat, ahol a hallgatók algoritmizálási képességeit lehetne megalapozni. Ezt talán az első éves alapozó tantárgyba kellene beintegrálni. Hasonlóképpen hiányoznak a logikai előismeretek is, így ezen tantárgyak oktatása során is igen sok időt elvesz a tantárgyhoz szükséges alapismeretek elsajátítása, így alig jut idő a tematikában felsorolt ismeretekre a Szakértői rendszerek tantárgy keretében. Az ötödik féléves Internet alkalmazásfejlesztés során, ahol PHP alapú MySQL adatbázissal dolgozó

webes alkalmazások készítését kell elsajátítani a hallgatóknak, azzal a problémával szembesültünk, hogy kevés az óraszám a kitekintéshez. Az óraszám sajnos kevés ahhoz, hogy e tárgy keretében bővebb kitekintést adjunk a hallgatóknak, hiszen meg kell ismerniük a HTML leírónyelvet, a CSS stílusnyelvet, a kliensoldali dolgok használatához a JavaScript-et és az AJAX technológiát. Erre talán a mesterképzésben szabadon választható ugyanilyen néven futó tantárgy kínálhat megoldást, legalábbis annak, aki ezt választja. Amint azt a bevezető részben is említettük, különösen fontos a fent említett problémákra minél előbb megoldást találni, mert a szeptemberben induló MSc képzés során két tárgy is a szoftverfejlesztési ismeretekre épül, illetve a szakértői rendszerek tárgynak is lesz folytatása ezen a szinten.

Irodalomjegyzék

- [1] László Aszalós: Online and offline logic tests, Proceedings of The Ninth International Conference on Technology in Mathematics Teaching, Metz (2009)
- [2] Mária Bakó: E-learning tananyag a szakértői rendszerek oktatásában, Informatika a felsőoktatásban, Debrecen, (2011)
- [3] Csetényi Arthur, Mohácsi László, Várallyai László: Szoftverfejlesztés (egyetemi jegyzet) (Informatikus és szakigazgatási agrármérnök BSc szakosok számára) (2006)
- [4] Várallyai László, Péntek Ádám: Internetes adatbáziskezelés (egyetemi jegyzet, Informatikus Agrármérnök hallgatók számára) (2007)
- [5] Várallyai László, Péntek Ádám: Kliens és szerveroldali webprogramozás (egyetemi jegyzet, Informatikus Agrármérnök hallgatók számára) (2007)

Képfeldolgozás a szegedi informatikus-képzésben

Image processing in the education of computer scientists at the University of Szeged

**Balázs Péter, Katona Endre, Kató Zoltán, Nagy Antal, Németh Gábor,
Nyúl László, Palágyi Kálmán, Tanács Attila, Varga László Gábor**

Szegedi Tudományegyetem (SZTE), Természettudományi és Informatikai Kar (TTIK),
Képfeldolgozás és Számítógépes Grafika Tanszék
{pbalazs, katona, kato, nagya, gnemeth,
nyul, palagyi, tanacs, vargalg}@inf.u-szeged.hu

Absztrakt: A digitális képfeldolgozás oktatása Szegeden több mint három évtizedes múltra tekint vissza. E tudományterület hazai művelőinek jelentős műhelye az SZTE TTIK Informatikai Tanszékcsoportján belül működő Képfeldolgozás és Számítógépes Grafika Tanszék. Az SZTE az egyetemi informatikus-képzés teljes kínálatát nyújtja, a programtervező informatikus, mérnök informatikus, és gazdaságinformatikus alap- és mesterszakok mellett számos egyéb szakon is, továbbá az informatika tanárképzésben és a doktori képzésben. A képfeldolgozás szakterület az egyes képzési programokban törzstárgyak, választható tárgyak, speciálkollégiumok és több szakon önálló szakirányok kurzuskínálatában is megjelenik. Az érdeklődő hallgatók már tanulmányaik elején találkozhatnak a képfeldolgozás érdekességeivel és bekapcsolódhatnak a tanszéken folyó kutatási és ipari projektekbe is. Évente számos diplomadolgozat készül témavezetésünkkel és hallgatónk rendszeresen érnek el kiemelkedő eredményeket a helyi és országos TDK konferenciákon. Rendszeresen tartunk kurzusokat az SZTE Informatika Doktori Iskolában, folyamatosan vannak doktoranduszaink és születnek nálunk sikeresen megvédett képfeldolgozás témájú PhD értekezések. Tudomásunk szerint a képfeldolgozás területén a hazai felsőoktatásban a mi kínálatunk a leggazdagabb.

Kulcsszavak: digitális képfeldolgozás, számítógépes grafika, számítógépes látás

Abstract: The teaching of digital image processing related subjects has a tradition of more than four decades in Szeged. One significant group of the Hungarian image processing community is at the Department of Image Processing and Computer Graphics, Faculty of Science and Informatics, University of Szeged. The University of Szeged offers the full set of informatics (computer science) majors, such as Computer Science, Computer Science Engineering, Business Informatics BSc and MSc, several other majors, as well as informatics teachers' education and doctoral programs. The field of image processing appears in the curricula as mandatory or selectable courses, special courses, or as a set of courses in an independent specialization in several programs. Interested students can meet with interesting image processing problems early in their studies, join ongoing research and development project at the Department. Many diploma theses are submitted annually under our supervision and our students regularly show exceptional performance at the local and the National Scientific Students' Associations Conference. We regularly teach courses in the Informatics Doctoral School of the University of Szeged, and continuously have doctoral students and produce successfully defended theses from the field of images processing. To our knowledge, we provide the most versatile education in the image processing field among the Hungarian universities.

Keywords: digital image processing, computer graphics, computer vision

1. Bevezetés

A digitális képfeldolgozás oktatása a Szegedi Tudományegyetemen (2000 előtt József Attila Tudományegyetem) több mint három évtizedes múltra tekint vissza. E tudományterület hazai művelőinek jelentős műhelye az SZTE Természettudományi és Informatikai Kar (TTIK) Informatikai Tanszékcsoportján belül működő *Képfeldolgozás és Számítógépes Grafika Tanszék*. Az SZTE az egyetemi informatikus-képzés teljes kínálatát nyújtja, a programtervező informatikus, mérnökinformatikus, és gazdaságinformatikus alap- és mesterszakok mellett számos egyéb szakon is, továbbá az informatika tanárképzésben és a doktori képzésben. Tudomásunk szerint a képfeldolgozás területén a hazai felsőoktatásban a mi kínálatunk a leggazdagabb. Tárgyainkhoz elérhetők a folyamatosan továbbfejlesztett elektronikus jegyzeteink.

A következőkben ismertetjük a képfeldolgozási és számítógépes grafikai tárgyú ismeretek megjelenéseit a különböző képzési programjainkban valamint a hallgatókkal és tanulmányaikkal kapcsolatos egyéb területeken.

2. Képfeldolgozás a BSc képzéseinkben

Az alapképzésben résztvevő hallgatók a képfeldolgozás és számítógépes grafika témáival alapozó törzstárgyakban és több speciálkollégium keretében találkoznak. Ennek köszönhetően a hallgatók már tanulmányaik korai fázisában kedvet kaphatnak ehhez a szakterülethez, aminek köszönhetően rendszeresen születnek eredményes TDK munkák és a mesterképzésben továbbtanulók közül is sokan választják a képfeldolgozás szakirányt.

A *Digitális képfeldolgozás* tárgy a programtervező informatikus és mérnökinformatikus alapszakokon kötelező, gazdaságinformatikus alapszakon választható, a társterületeket lefedő *Számítógépes grafika* és *Multimédia* kurzusok pedig mindhárom szakon kötelezők.

2.1. Törzstárgyak

Digitális képfeldolgozás

A tárgy tematikája a képfeldolgozás (matematikai) alapjait, a legfontosabb fejezeteit (bennük az alapvető módszereket, algoritmusokat) öleli fel: képképzés, Fourier-transzformáció, konvolúció, pontoperációk és hisztogram-transzformációk, szűrés képtérben és frekvenciatérben, jellemzők detektálása, szegmentálás, alakleírás, képek kódolása. Az előadás-anyag legfontosabb forrása Gonzales és Woods kézikönyve [1].

A laboratóriumi gyakorlaton a Matlab Image Processing Toolkit (mint korszerű prototípus-előállító eszköz) segítségével összetett alkalmazási feladatok megoldására, megvalósítására is képessé válnak a hallgatók.

Számítógépes grafika

A kurzus a 2D és 3D számítógépes grafika alapelveit, feladatait, matematikai alapjait és legfontosabb algoritmusait tárgyalja: grafikus primitívek rajzolása raszteres képernyőn, területkitöltő algoritmusok, geometriai transzformációk, grafikus primitívek vágása, vetítések, parametrikus görbék és felületek megadása, látható vonalak és felszínek meghatározása. Szó esik még a különféle színmodellekről, a megvilágítottság modelljeiről, árnyékolási algoritmusokról. A hallgatók megismerkednek az OpenGL programozás alapjaival és a

számítógépes laborgyakorlatokon a megismert módszereket konkrét példákon is kipróbálják, illetve egyszerűbb grafikai feladatokat oldanak meg. A tematika Foley klasszikus tankönyvére [2] illetve Szirmay-Kalos magyar nyelvű tankönyveire [3], [4] támaszkodik.

Multimédia

A kurzus egy áttekintő képet nyújt a multimédiával kapcsolatos fogalmakról, eszközökről, alapvető technikákról, alkalmazási területekről és tervezési szempontokról. A változatos hallgatói összetétel miatt csak a legszűkebb előismeretekre támaszkodik, és összefoglalásra kerülnek a kapcsolódó témák legalapvetőbb fogalmai, módszerei, valamint iránymutatókat ad az érdeklődők számára a mélyebb ismeretek felé. Tárgyalásra illetve áttekintésre kerülnek a digitális reprezentációval, digitalizálással, a grafikai elemekkel kapcsolatos fogalmak, különböző színterek, színek megválasztása, használata, tipográfiai alapismeretek, lapelrendezés, grafikus és szöveges elemek kombinálása, strukturált szöveg, jelölőnyelvek, stíluslapok alkalmazása. Külön témakört alkotnak a mozgóképpel (animáció, film- és videókészítés, feldolgozás) és hangelemekkel kapcsolatos technikák, formátumok, szabványok, valamint a különböző médiumok kombinálására és az interaktivitás kezelésére használt módszerek. A számítógépes laborgyakorlatokon a hallgatók megismerkednek egy kiválasztott szoftver eszközzel, amivel önálló multimédiás projekteket valósítanak meg. Az előadás anyaga nagyjából a Chapman páros könyvének [5] tárgyalásmódját követi, de az információkat egyéb nyomtatott és online források alapján bővítve és frissítve tartalmazza.

3. Képfeldolgozás az MSc képzéseinkben

A szakterület a programtervező informatikus mesterképzésben is jelentős szerepet kap, hiszen a *Képfeldolgozás haladóknak* és a *Fejlett grafikai algoritmusok* tárgyak kötelezők, a *Képfeldolgozás szakirányunk* pedig a PTI MSc program indulása óta (és a korábbi egyetemi programtervező matematikus szakon is) folyamatosan az egyik leggyakrabban választott specializáció változatos, folyamatosan bővülő kurzuskínálattal. Számos kurzusunk kapott helyet a mérnökinformatikus MSc szak kötelező és választható tárgyai között is. A következő tanévtől indítjuk az info-bionika MSc szakot *bio-nano mérőeszközök és képalkotók* szakiránnyal. A képfeldolgozás itt is jelentős szerepet kap mind a kötelező mind a szakirányos választható kurzusok között. Mindezekon felül több képfeldolgozás témájú speciálkollégium is gazdagítja kínálatunkat.

3.1. Törzstárgyak

Képfeldolgozás haladóknak

A tárgyat a fentebb bemutatott Digitális képfeldolgozás alapozza meg. Tematikája a kulcsproblémákra adott fejlett módszereken túl további érdekes fejezetet is tárgyal: pl. színes képek feldolgozása, a szegmentálás fejlett módszerei, textúra-elemzés, matematikai morfológia, mozgáskövetés, vázkijelölés, leképezések (warping, morphing), bináris képek feldolgozása. Az előadás anyagánál felhasznált forrásművek közül Sonka, Hlavac és Boyle kézikönyvét [6] emeljük ki, valamint elkészült a tárgy tematikáját felölelő magyar nyelvű egyetemi jegyzet is [7].

A laboratóriumi gyakorlaton a Matlab Image Processing Toolkit (mint korszerű prototípus-előállító eszköz) segítségével összetett alkalmazási feladatok megoldására, megvalósítására is képessé válnak a hallgatók.

Fejlett grafikai algoritmusok

A tantárgy keretein belül főleg olyan algoritmusokat mutatunk be, melyek a Számítógépes grafika tantárgy ismeretanyagára épülnek, és új módszereket adnak a 3D-s objektumok realiztikus megjelenítésére. Bemutatásra kerülnek különböző transzformációs, textúrázási és megvilágítási technikák valamint az algoritmusok mellett olyan programozási technikákat is ismertetünk, amelyek kihasználják a programozható grafikus hardverek képességeit.

A laboratóriumi gyakorlatokon az OpenGL függvény könyvtár és Cg eszköz segítségével szemléltetjük az előadáson ismertetett algoritmusokat.

Az utóbbi években készült el a tárgy tematikájához kapcsolódó egyetemi jegyzet [8] és előadásvázlat [9].

3.2. Képfeldolgozás szakirány

GPGPU: Grafikus processzorok felhasználása általános célú számításokra

A hallgatók jártasságot szereznek programozható grafikus kártyák általános célú számításokra való felhasználására. Különböző programozási architektúrák alkalmazásainak esettanulmányokon keresztül való bemutatásával a hallgatók képessé válnak párhuzamosítható algoritmusok GPU-kon való implementálására és optimalizálására. A hallgatók olyan ismereteket is elsajátítanak, amelyek a képfeldolgozás tudományterületen alkalmazott módszerek esetén hatékonyan használhatnak. Az általános GPU elvek elsajátítása mellett a hallgatók megismerkednek a CUDA architektúrával és használatával [10], valamint az OpenCL platform használatával [11].

Ipari képfeldolgozás

A hallgatók megismerkednek a vizuális minőségellenőrzés, kép alapú mérés és a kép alapú mobil robot navigáció algoritmusaival és azok legfontosabb ipari alkalmazásaival. A kurzus tananyaga kitér a következő témákra: Kamerák és optikák jellemzése, a képpalkotás modellje, kamera kalibráció és optikai torzítások. Speciális optikák, mint pl. telecentrikus, körbelátó (omnidirectional) képpalkotása és kalibrációja. Alapvető megvilágítások és azok használata. Kép alapú mérés (megvilágítás és optika megválasztása, sablon illesztés). Több kamerából/optikából álló rendszer és annak kalibrációja (mozaikozás, sztereo képpalkotás). 3D rekonstrukció strukturált fény segítségével (strukturált fény fajtái, azok használata). Mobil robot navigáció, SLAM (lokalizáció és feltérképezés).

Képi adatbázisok

A kurzus a képi információ adatbázisban való tárolásával illetve a képi tartalom és adatbázisok kapcsolatával foglalkozik. Két fő megközelítést vizsgálunk. Egyrészt képparchívumok kezelési technikáit tekintjük át, kiegészítve a képi tartalom szerinti keresés lehetőségeivel, érintve a kereséshez használt képi eredetű jellemzők kinyerését, a szegmentálás, képosztályozás, alakleírás kapcsolódó kérdéseit is. Másrészt, vektoros (például geoinformatikai) adatok relációs adatbázisban való hatékony kezelését vizsgáljuk [12], [13]: térbeli indexelési módokat, speciális lekérdezési és megjelenítési lehetőségeket tárgyalunk. A kurzushoz kapcsolódó laborgyakorlaton a hallgatók megismerkednek a témakörhöz kapcsolódó szoftverekkel, és projektmunka keretében mintaalkalmazást készítenek.

Képregisztráció

A képregisztráció a digitális képfeldolgozás napjainkban felmerült problémája, mely két, egymástól függetlenül felvett kép közötti geometriai viszony meghatározását jelenti. Ezen képek felvétele különböző időpontokban, különböző nézőpontokból vagy akár különböző

képkötő berendezésekkel is történhet. A regisztrációs problémával számos alkalmazási területen találkozunk, beleértve a számítógépes látást és az orvosi képek illesztését. Ez utóbbi esetben a betegekről különböző időpontokban, irányokból vagy módokon készült felvételek illesztése megbízhatóbb diagnosztikai információt biztosít, segíti a képmegértést, és javítja a műtét- és a terápia tervezését, valamint kiértékelését. A kurzus során bemutatása kerülnek a regisztrációs módszerek csoportosítási lehetőségei, általános jellemzői. Részletesen tárgyaljuk a pont-alapú, a felszínillesztő, és a képpont hasonlóságon alapuló módszereket. Az anyag zárásaként szó esik a nem-lineáris módszerekről, valamint műtéttervezési és -végrehajtási feladatokról is. Az előadás anyaga elsősorban Hajnal és munkatársai [14], valamint Goshtasby [15] könyveire épül.

Képrekonstrukció

A kurzus a vetületi képekből történő képkötés általános módszereibe és alkalmazásaiba nyújt bevezetést. A tárgyalt témák többek között: konvolúciós rekonstrukció, algebrai rekonstrukciós technikák, 3D tárgyak rekonstrukciója, számítógépes tomográfia, orvosi képek rekonstrukciója (CT, MRI, SPECT, PET), a tomográfia ipari alkalmazásai. Emellett a hallgatók megismerkedhetnek a diszkrét tomográfiával és annak alkalmazásaival is (angiográfia, elektronmikroszkópia, nemroncsoló anyagvizsgálat). Ez utóbbi témakör Herman és Kuba könyvei alapján kerül feldolgozásra [16], [17].

A gyakorlatokon a hallgatók feladata a bemutatott rekonstrukciós módszerek valós képeken történő alkalmazása.

Mikroszkópiai képfeldolgozás

A high content screening (HCS) ötvözi a sejtbológia (annak minden molekuláris eszközével), az automatizált nagy felbontású mikroszkópia, és a robotika legújabb fejlesztéseit. Olyan molekuláris alkotóelemek (pl. gyógyszerek, gátló RNS) felkutatására szolgál, melyek a kívánt módon változtatják a sejt fenotípusát. A high content analysis (HCA) a high-content screening során előállított adatok elemzésével és értelmezésével foglalkozik. Habár az elmúlt években nagy előrelépések történtek, az elemzés során mégis a legnagyobb limitáló tényező a számítási problémák megoldása, mind sebesség, mind pedig precizitás szempontjából. Ennek megoldására a közelmúltban számos képfeldolgozó és gépi tanulási eljárást mutattak be. A kurzus során megismerjük a HCA-ban használt legnépszerűbb módszereket, betekintést nyerünk az automatizált mikroszkópiába, képfeldolgozásba, és az adatok többváltozós analízisébe. A gyakorlat során, automatizált fluoreszcens mikroszkóp segítségével több 10.000-100.000 képet készítünk, majd képfeldolgozó és gépi tanulási módszereket fejlesztünk az adatok feldolgozására.

Mobil képelemzés és grafika

A kurzus célja a hallgatókat megismertetni a korszerű mobil platformok képkötési, képfeldolgozási és 3D grafikai feladatokhoz kapcsolódó programozási és felhasználási lehetőségeivel. Az anyag kitér ilyen jellegű rendszerek tervezésére, mobil eszközök képkötő és 3D grafika képességeit kihasználó rendszerek tervezésére és implementálására, valamint képkötési lehetőségek kiterjesztésére modern képfeldolgozó algoritmusok ismeretével. A tanórákon lefedett témakörök: képi adatok elérése népszerű mobil platformokon, OpenCV függvénykönyvtár használata Android platformon, 3D grafika OpenGL ES könyvtárakkal, szenzorok, szenzor vezérelt 3D grafika, kiterjesztett valóság, kép-alapú keresés alapjai, arcdetektálás, vonalkódok, QR kódok, HDR képkötés, karakterek és számok képi felismerése, 3D képkötés és megjelenítés.

Nem-konvencionális adatbázisok

A kurzus során a hallgató megismerkedik a képi adatbázisok tervezésének és kezelésének speciális megoldásaival, és a képi tartalom szerinti keresés alapjaival. A hallgató betekintést nyer a nagyméretű, elosztott, nem-relációs adatbáziskezelés elméletébe és megismeri ezen technológiák leggyakoribb alkalmazásait. Emellett elsajátít olyan speciális adattárolási és adatmanipulációs technikákat is, mint például az XML alapú vagy az időbeliséget támogató adatbázisok kezelése.

Orvosi képalkotás

A tárgy a legfontosabb orvosi képalkotó eljárásokkal, a képalkotás fizikai alapjaival, az egyes modalitásokra leginkább jellemző képfeldolgozó eljárásokkal foglalkozik. Ennek során ismerteti a Röntgen és CT készüléket, a vetületekből történő rekonstrukció elvét. A radioaktív sugárzás detektálására szolgáló szkennereket, a gamma kamerát és a SPECT készüléket. A gamma kamera kapcsán bemutatja a szokásos kalibrációs és korrekciós eljárásokat. A PET készüléket. A PET készülék segítségével végezhető két és több kompartment analízist. Az ultrahanggal végezhető vizsgálatok közül tárgyalja az időbeli és a térbeli felbontást biztosító és a Doppler hatást kihasználó felvételi módokat. Bemutatja a mágneses rezonancia készülék működési elvét, a proton sűrűség, a T1 és T2 képek készítményét. Szó esik a regisztráció fontosságáról, alkalmazási lehetőségeiről. Bemutatásra kerül néhány egyszerűbb klinikai program, és funkcionális jellemzőket bemutató 3 dimenziós megjelenítés. Az előadás anyagának bőséges kiegészítését szolgálja Gelfand és Thomas könyve [18], továbbá a három kötetes Medical Imaging [19], [20], [21].

A laboratóriumi gyakorlatokon a hallgatók az orvosi képfeldolgozás egyes fejezeteihez tartozó egyszerűbb feladatokat oldanak meg.

Számítógépes látás

A kurzus célja a számítógépes látás elméletének és gyakorlatának megismertetése a hallgatókkal. A számítógépes látás az emberi látás azon funkcióit valósítja meg, amelyek a retinai kép elemzését végzik. Ezek elsősorban a képi tartalom értelmezésére irányulnak: a látott képből következtet az objektumok 3D alakjára (felületrekonstrukció), az objektumok térbeli elhelyezkedésére, egymáshoz való viszonyára (mélységi információ kinyerése), illetve több, időben egymást követő képből a mozgás érzékelése és a mozgó objektumok követése. A kurzus előadásain a következő témakörökbe kapnak részletes betekintést a hallgatók: Az emberi és a számítógépes látás kapcsolata, a látás matematikai és pszichofizikai modelljei (Marr, Gestalt szabályok). Kamera geometria és a leképezés paraméterei. Felületrekonstrukció egyetlen kép segítségével. Szetero látás matematikai alapjai: epipolár geometria, esszenciális mátrix, fundamentális mátrix. 3D rekonstrukció egy ill. több képpárból, virtuális nézetek generálása. Mozcás mérése, Optical Flow kiszámítása, parametrikus mozcás modellek, mozcáskövetés.

A laboratóriumi gyakorlatokon egyrészt az előadáson elhangzott algoritmusok gyakorlati alkalmazását ismerhetik meg a hallgatók mintaprogramokon keresztül, másrészt a félév során intenzív projektmunka zajlik.

A kurzus anyagának megértéséhez szakirodalomként három kézikönyv ajánlott [22], [23], [24] és elkészült egy magyar nyelvű egyetemi jegyzet is [25].

Számítógéppel segített mütéti tervezés

A képfeldolgozás orvosi alkalmazásai között ma már nem csak a diagnosztika szerepel, hanem a terápia, közöttük a mütétek megtervezésének és végrehajtásának a számítógépes segítése is. A kurzus célja, hogy betekintést adjon a képfeldolgozás ilyen alkalmazásába.

Bemutatásra kerülnek a számítógéppel segített műtéti tervezés legfontosabb elemei: orvosi képalkotás (CT, MRI), geometriai és biofizikai modellezés, műtéti navigáció és képek regisztrációja [26], [14].

Térinformatika

A térinformatika (geoinformatika) ugyan önálló tudományterületet képez, de számos ponton kapcsolódik a digitális képfeldolgozáshoz. A szükséges térképészeti alapismeretek után a kurzusban a vektoros térinformatikára helyezük a hangsúlyt: áttekintjük a fontosabb adatmodelleket (spagetti és topológikus modellek, tartománytérképek, hálózatok), majd vektoros algoritmusokat és adatbázismodelleket tárgyalunk. Ezután a térinformatikában használatos jellemző raszteres algoritmusok következnek, majd a digitális terepmodellezés eszközszerét tárgyaljuk.

A géptermi gyakorlaton a hallgatók a MicroStation térinformatikai rendszerrel ismerkednek. Kötelező feladatként egy-egy kisebb térinformatikai alkalmazást kell elkészíteniük, amely egy térképrészlethez kapcsolt több táblás adatbázisból és megfelelő lekérdezési lehetőségekből áll.

4. Képfeldolgozás a képzési programokon túl

A Számítógépes grafika és a Digitális képfeldolgozás tárgyak kibővített változata, valamint a Számítógépes képelemzés speciálkollégium szerepel az Informatikai Tanszékcsoporton működő *BSc tehetséggondozó program* kiemelt kurzusai között. Az érdeklődő hallgatók már tanulmányaik elején találkozhatnak a képfeldolgozás érdekességeivel és bekapcsolódhatnak a tanszéken folyó *kutatási és ipari projektekbe* is, ahol valós problémákkal foglalkozhatnak. Évente számos *szakdolgozat és diplomamunka* készül témavezetésünkkel és hallgatóink rendszeresen érnek el kiemelkedő eredményeket a helyi és országos *TDK konferenciákon*. Rendszeresen tartunk kurzusokat az *SZTE Informatika Doktori Iskolában*, folyamatosan vannak doktoranduszaink és születnek nálunk sikeresen megvédett képfeldolgozás témájú PhD értekezések.

Az SZTE az 1993-ban indított *Summer School on Image Processing (SSIP)* nemzetközi képfeldolgozó nyári iskola alapító intézménye. Minden évben vállalunk előadásokat és delegálunk szegedi hallgatókat a rendezvényre, melynek 2015-ben immár 11. alkalommal adunk majd otthont Szegeden. Nemzetközi kapcsolataink révén hallgatóink ösztöndíjasként rövidebb-hosszabb időt tölthetnek külföldi egyetemeken.

A képfeldolgozás és grafika tárgyak oktatását, a TDK munkát, szakdolgozat és diplomamunka készítést, valamint a projekt-munkákat *több speciálisan felszerelt laborral* is támogatjuk, melyekhez a kurzusokra járó illetve az engedéllyel rendelkező hallgatók férhetnek hozzá.

- A *képfeldolgozás oktató labor* tanterem jellegű, többnyire szakirányos kurzusokat, speciálkollégiumokat tartunk benne, de a számítógépek (oktatási időn kívül) projektekhez számítási klaszterként is használhatók. Felszereltsége: 16 db számítógép, nagyteljesítményű, sokmagos GPU-val szerelve, CUDA programozás támogatása, képfeldolgozó és grafikai szoftverek: Matlab, Autodesk Product Design Suite, Adobe Creative Suite, Blender, MeshLab, OpenCV, valamint Android Development Tools a mobil képfeldolgozás oktatásához.
- A *mobil képfeldolgozás labor* támogatja az önálló feladatok elvégzését mivel itt szervezett oktatás nem történik és így a hallgatók saját időbeosztásuk szerint használhatják. Felszerelése: több tucat kamerával ellátott okostelefon különféle gyártó (HTC, LG, Nokia, Samsung, Sony Ericsson, Apple) és platform (Android, Windows

Mobile, iOS, Symbian) kombinációkban, 2 db MacBook Pro notebook, és néhány asztali számítógép.

- A *3D labor* a térbeli képadat előállításához és 3D modellek kezeléséhez biztosít lehetőséget. Felszerelése: NextEngine 2020i Desktop 3D Scanner, MultiDrive, AutoPositioner, egy nagyteljesítményű számítógép, 3D monitor és aktív 3D szemüveg.
- A *számítógépes látás* labor elsősorban a gépi látás és ipari képfeldolgozás jellegű kurzusokhoz és projektekhez kapcsolódik. Felszereltsége: JAI CB-140GE színes kamera, Pericentrikus optika, Boroszkópikus csővizsgáló optika, Csővizsgáló optika, Telecentrikus optika, Struktúrált minta vetítő, Háttérvilágítás (fehér), Diffúz fényforrás, Canon EF 8-15mm f/4 L fisheye zoom USM halszem optika, MESA SR4000 Time-of-flight 3D kamera, asztali számítógép.

5. Összefoglalás

A képfeldolgozás és rokonterületei több évtizedes hagyományokra építve komoly szerepet kapnak a szegedi informatikus-képzésben. Hallgatóink gyakorlatilag az alapképzés derekától találkoznak a szakterülettel, az érdeklődők később a mesterképzésben képfeldolgozás szakirányon, majd PhD tanulmányaik során teljesíthetik ki ismereteiket. Tanszékünk kutatási tevékenysége és ipari projektjei a képzések bármely szintjén résztvevő hallgatóknak lehetőséget biztosítanak arra, hogy elméleti ismereteiket valós problémák megoldásában kamatoztassák. Végzett hallgatóinkat örömmel fogadják a különböző cégek is képfeldolgozással kapcsolatos munkakörökbe.

Irodalomjegyzék

- [1] Gonzales R.C., Woods R.E.: Digital Image Processing, 3rd edition, Prentice-Hall, Inc. (2008)
- [2] Foley J.D., van Dam A., Feiner S.K., Hughes J.F.: Computer Graphics. Principles and Practice, Addison-Wesley (1999)
- [3] Szirmay-Kalos L.: Számítógépes grafika, Computerbooks (1999)
- [4] Szirmay-Kalos L., Antal Gy., Csonka F.: Háromdimenziós grafika, animáció és játékfejlesztés, Computerbooks (2003)
- [5] Chapman N., Chapman J.: Digital Multimedia, John Wiley & Sons, Ltd. (2000)
- [6] Sonka M., Hlavac V., Boyle R.: Image Processing, Analysis, and Machine Vision, 3rd edition, Thomson Learning (2007)
- [7] Palágyi K.: Képfeldolgozás haladóknak, egyetemi tananyag, Typotex Kiadó (2011)
- [8] Nagy A.: Fejlett grafikai algoritmusok, egyetemi tananyag, Typotex Kiadó (2011)
- [9] Nagy A.: Fejlett grafikai algoritmusok, előadásvázlatok, Szegedi Tudományegyetem (2014)

- [10] Sanders J., Kandrot E.: *CUDA by Example: An Introduction to General-Purpose GPU Programming*, Addison-Wesley (2010)
- [11] Kovács György: *Open CL, egyetemi tananyag*, Typotex Kiadó (2014)
- [12] Rigaux Ph., Scholl M., Voisard A.: *Spatial databases*. Morgan Kaufmann Publishers, San Francisco (2002)
- [13] Katona E.: *Térképi adatbázisok, egyetemi tananyag*, Typotex Kiadó (2011)
- [14] Hajnal J.V., Hill D.L.G., Hawkes D.J.: *Medical Image Registration*. CRC Press (2001)
- [15] Goshtasby A.A.: *2-D and 3-D Image Registration*, John Wiley & Sons, Inc. (2005)
- [16] Herman G.T., Kuba A. (eds.): *Discrete Tomography: Foundations, Algorithms and Applications*, Birkhauser (1999)
- [17] Herman G.T., Kuba A. (eds.): *Advances in Discrete Tomography and Its Applications*, Birkhauser (2007)
- [18] Gelfand M.J., Thomas S.R.: *Effective Use of Computers in Nuclear Medicine*, McGraw-Hill (1988)
- [19] Beutel J., Kundel H.L., Van Metter R.L.: *Medical Imaging, Vol. 1*, SPIE (2000)
- [20] Sonka M., Fitzpatrick J.M.: *Medical Imaging, Vol. 2*, SPIE (2000)
- [21] Kim Y., Horii S.C.: *Medical Imaging, Vol. 3*, SPIE (2000)
- [22] Faugeras O.: *Three Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press (1996)
- [23] Hartley R.I., Zisserman A.: *Multiple View Geometry in Computer Vision*, Cambridge University Press (2004)
- [24] Ma Y., Soatto S., Kosecka J., Sastry S.: *An Invitation to 3-D Vision*, Springer Verlag. (2003)
- [25] Kató Z., Czúni L.: *Számítógépes látás, egyetemi tananyag*, Typotex Kiadó (2011)
- [26] Kalender W.A.: *Computed Tomography Fundamentals, System Technology, Image Quality, Applications*. John Wiley & Sons (2011)

CAEDUS oktatási keretrendszer

CAEDUS Learning Management System

Biró Piroska

Debreceni Egyetem, Informatikai Kar
biro.piroska@inf.unideb.hu

Absztrakt: A XXI. század elején az IKT (Információs és Kommunikációs Technológia) eszközök megjelenése és használata szerves részévé vált a mindennapjainknak, az oktatásban is egyre nagyobb szerepet töltenek be. Nemzetközi és hazai kutatások eredményei is alátámasztják, hogy pozitív hatással vannak az eszközök a diákok motivációjára, ugyanakkor nem egyszerű feladat a tanárok számára az eszközök kezelésének tökéletes elsajátítása. Az IKT eszközök integrálásának feltétele a digitális kompetenciák jelenléte és az IKT eszközök működtetésére vonatkozó alapvető ismeretek. A CAEDUS (Computer Aided Educational System) egy oktatási keretrendszer, mely a közoktatásban tanító pedagógusok számára készült. A keretrendszer létrehozásának célja, hogy segítse a pedagógusok mindennapi munkáját, továbbá, hogy egy olyan rendszert fejlesszünk ki, melynek kezelése egyszerű, felhasználóbarát, platformfüggetlen, tanár, diák és szülő számára bármikor és bárhol elérhető. Előadásomban a CAEDUS, mint egy újonnan fejlesztett oktatási keretrendszer, használatát mutatom be.

Kulcsszavak: CAEDUS (Computer Aided Educational System), oktatási keretrendszer, IKT eszközök

Abstract: At the beginning of the 21st century the appearance and the use of ICT (Information and Communication Technology) devices became an organic part of our everyday life and are gaining growing importance in the field of education. The results of both the international and Hungarian researches support the fact that the devices have positive effect on the students' motivation, however, mastering the handling of the devices is not an easy task for the teachers. The integration of the ICT devices into the education system requires the presence of the digital competences and the basic knowledge related to operation of the ICT devices. CAEDUS (Computer Aided Educational System) is a learning management system, which has been designed for the teachers in primary and secondary education. The CAEDUS was created with the purpose to provide aid for the teachers in their everyday work, furthermore to develop a system which is user friendly, platform-independent and easily available for both students and parents, anytime and anywhere. In my presentation I will introduce the advantages and disadvantages of the CAEDUS learning management system.

Keywords: CAEDUS (Computer Aided Educational System), learning management system, ICT devices

1. Bevezetés

A CAEDUS (Computer Aided **E**ducational System) egy olyan tanulás/tanítás menedzsment rendszer, amely lehetővé teszi a tanár és a diákok közötti információcserét, továbbá támogatja a különböző platformokon való megjelenítést is.

A célunk egy felhasználóbarát eszköz fejlesztése volt, amely megfelel a 21. századi pedagógusok elvárásainak is. Ezért előzetes kutatások, tanári interjúk előzték meg a rendszer fejlesztését.

Az IKT (Információs és Kommunikációs Technológiák) eszközök, főként az interaktív táblák megjelenése, maga után vonta a digitális tananyagok megjelenését és gyakoribb használatát a közoktatásban [21], [22], [23], [24].

Egyre nagyobb igény jelentkezik a minőségi, felhasználóbarát, könnyen és főként ingyen hozzáférhető, szerkeszthető elektronikus/digitális tananyagokra, melyet a pedagógus saját, egyéni elképzelései szerint módosíthat és megoszthat a tanórák között a tanulókkal. Ennek

következtében szükség van az egyszerű, felhasználóbarát tanulás/tanítás segítő rendszerekre (Learning Management System – LMS), amelyek képesek a regisztrált felhasználók kezelésére, az új információk, tananyagok tárolására, menedzselésére, az oktatási folyamatok adminisztrálására, a kommunikációs lehetőségek elősegítésére.

2. Az oktatási keretrendszerekről

Az e-tanulási formák megjelenése a 90-es évekre nyúlnak vissza. Elsősorban az oktatási kurzusok hálózati platformjaként megjelenő virtuális tanulókörnyezetek (VLE – Virtual Learning Environment), melyek statikus tárhelyként szolgáltak az oktatási tananyagok, tesztek és vitafórumok számára. Ezen rendszerek külön komponensekből épültek fel.

A 90-es évek második felétől megjelent rendszerek már integrált tananyagbázissal rendelkeztek, melyek magába foglalták a felhasználók adatait tartalmazó adatbázist is. A technológia fejlődésével ezek a rendszerek újabb és újabb funkciókkal bővültek, mint például az adminisztrációs, tananyagfejlesztő és tudásellenőrző modulok. A rendszerek folyamatos fejlesztésének köszönhetően a kommunikációs és média-elemek mellett, kifinomult statisztikai modulok is megjelentek, továbbá fejlett pedagógiai és adminisztratív funkciókon túl az egyéni igényeknek megfelelő testreszabhatóság lehetősége is előtérbe került [26].

A szakirodalomban a virtuális tanulókörnyezetek számos szinonimájával találkozhatunk: tanulásmenedzsment rendszer (Learning Management System, LMS), kurzusmenedzsment rendszer (Course Management System, CMS), tananyagmenedzsment rendszer (Learning Content Management System, LCMS), irányított tanulókörnyezet (Managed Learning Environment, MLE), tanulástámogató rendszer (Learning Support System, LSS) és a tanulási platform (Learning Platform, LP) [26].

Az utóbbi években a virtuális keretrendszereket két nagyobb csoportba szokás sorolni: tananyagfejlesztő eszközre és oktatási keretrendszerekre. A közelmúltban fejlesztett rendszerek mindkét funkcióval rendelkeznek, ezért a továbbiakban a bővebb oktatási keretrendszer kifejezést fogom használni.

Az oktatási keretrendszerek megjelenése maga után vont a különböző elvárásokat is a keretrendszerekkel szemben. A következő lista a Chief Learning Officer Magazine 2003. januári számában a következő elvárásokat írja le az LMS rendszerekkel kapcsolatosan [2]:

- magas rendelkezésre állás (high availability) – nagyszámú felhasználó kezelése, párhuzamos kiszolgálása
- skálázhatóság (scalability) – az infrastruktúra bővítésének lehetősége (hardver, szoftver)
- használhatóság (usability) – felhasználóbarát, egyszerű, könnyen kezelhető, motiváló
- együttműködési képesség (interoperability) – különböző típusú tartalmak kezelése
- stabilitás (stability) – megbízhatóan és hatékonyan biztosítani az elérést
- biztonság (security) – korlátozott és korlátlan hozzáférés biztosítása
- platformfüggetlenség

A következő elemek szinte minden rendszerben megtalálhatóak [27]:

- adminisztrációs modul
- tananyag menedzselési modul
- tananyagfejlesztői modul
- statisztikai modul
- teszt, kérdőív, értékelési modulok
- felhasználói tevékenységet regisztráló modul

- felhasználó profil, profilszerkesztői modul
- kommunikációs modul
- elektronikus kereskedelmi modulok

Kutatásaink alapján a magyarországi közoktatásban tanító pedagógusok körében legnépszerűbb, leginkább ismert oktatási keretrendszer a Moodle [20], [25] (nevét a Moduláris Objektum-Orientált Dinamikus Tanulási Környezet angol kifejezésből kapta – Modular Object-Oriented Dynamic Learning Environment) [12], LMS rendszer, fejlesztői a konstruktivista pedagógia irányelveit követve fejlesztették a rendszert. Martin Dougiamas ausztráliai fejlesztő munkájának eredményeként a rendszer prototípusát, 2001 novemberében jelentették be, majd a Moodle 1.0 szabad szoftver letölthető, stabil verziójának elkészültét pedig 2002 augusztusában. Legújabb verziója a Moodle 2.7, amely 2014 májusában jelent meg. Jelenleg 235 országban használják, több mint 71 millió felhasználóval rendelkezik [12].

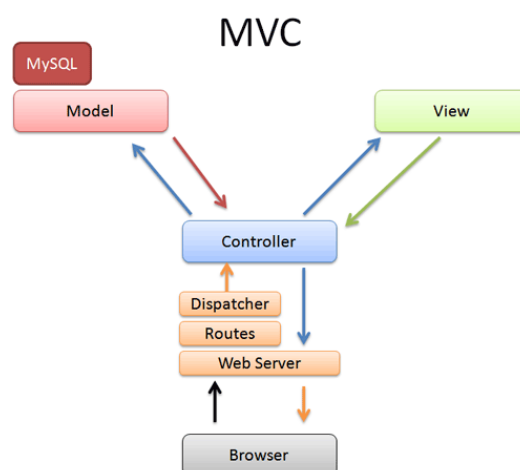
3. A CAEDUS fejlesztése

Technológiai háttér

A rendszer kliens szerver architektúrában működik. Az alkalmazás a központi szerveren fut, a felhasználók böngészőn keresztül lépnek vele kapcsolatba.

Fejlesztési technológia – Ruby és Ruby on Rails

A rendszert Ruby-ban fejlesztették. A Ruby script nyelv a 90' évek közepén jelent meg, Yukihiro Matsumoto (Matz) fejlesztette ki [11]. Maga a nyelv nagyon beszédes és könnyen tanulható. Dinamikus típusosság jellemzi és teljesen objektum orientált. Matz egy olyan nyelvet álmódott meg, amelyben a programozó élvezettel dolgozik. A Ruby nyelv elterjedéséhez az ezredforduló után nagyban hozzájárult a Ruby on Rails (RoR) keretrendszer kialakulása. Ez a webes keretrendszer egy MVC (model-view-controller) keretrendszer, amely megvalósítja a SOC (separation of concern) elvét miszerint az adatmodell réteget, a prezentációt és vezérlést erősen el kell különíteni egymástól (1. ábra) [1], [29], [28].



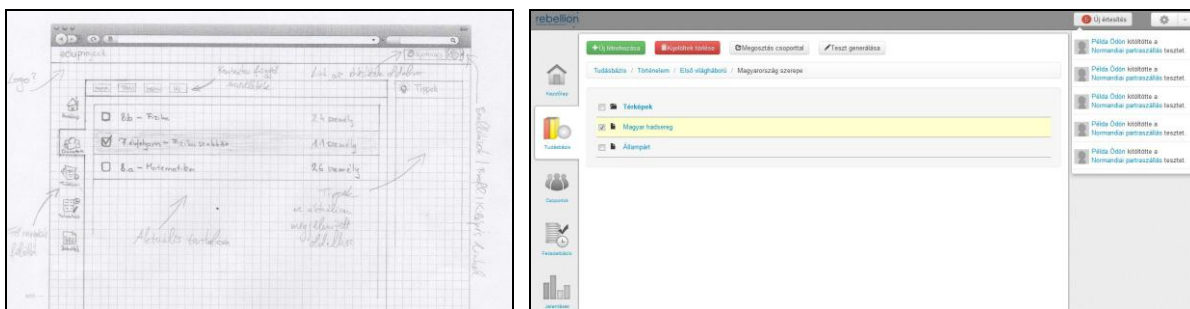
1. ábra. MVC – Model-View-Controller [13]

Adatbázis réteg

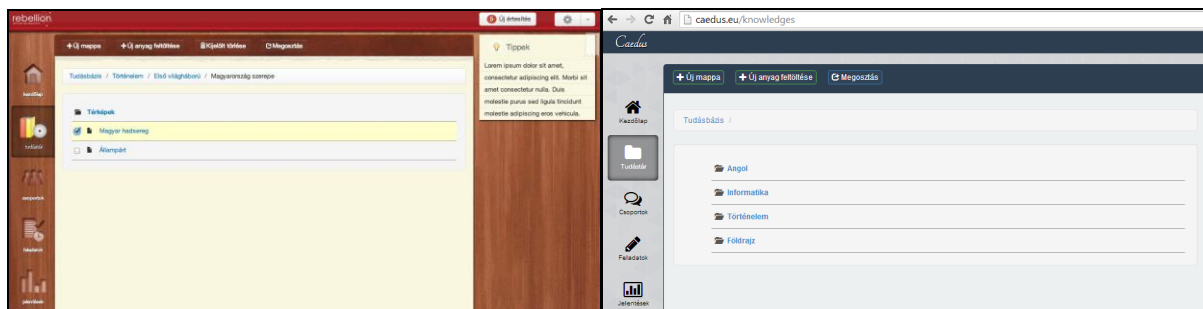
Az adatbázis réteg kialakítása során két ingyenes és relációs alapokon nyugvó adatbázis kezelő merült fel, a MySQL és a PostgreSQL. Először a PostgreSQL-t használtunk, mivel funkcionalitását tekintve például képes több sémát kezelni, van lehetőség az öröklődésre, de mivel idővel kiderült, hogy a MySQL-hez kényelmesebb fejlesztői készletek vannak, váltottunk [28].

Webes felület és megjelenés

A felület megtervezése (user interface) során elsődleges szempont az átláthatóság, a könnyű navigáció volt a cél. A 'skeuomorph desing' irányelveit követve a felületi elemeket a való világban megfelelő azonos funkcionalitással rendelkező tárgyokról mintáztuk [28].



2. ábra. A felület első terve (bal) és első megvalósítása (jobb)



3. ábra. A felület tervei 'skeuomorph design'-nal

A felület a felhasználói visszajelzések során folyamatos átalakuláson ment keresztül (2 és 3. ábra), mindaddig, amíg elérte jelenlegi formáját.

Jogosultságok, felhasználók, szerepkörök kezelése, biztonság

A CAEDUS rendszer következő fontos része a felhasználók kezelése, azonosítása. A rendszert három felhasználói szerepkör számára fejlesztettük, ezért három felületet kellett kialakítani: tanári, tanulói és szülői felületek. A három felhasználói szerepkörhöz különböző jogosultságokat rendeltünk, melyek alapvetően különböznek egymástól (további részletek a 4. fejezetben).

Az autentikáción kívül nagyon fontos feladat volt a rendszerbiztonság biztosítása. Autentikációra és autorizációra rengeteg 'gem' létezik a Ruby on Rails-ben, mi mégis saját algoritmust írtunk rá. Az autentikációnál a Ruby on Rails filtereit használtuk, például ilyen filter volt az a metódus, amivel kiszűrtük a HTML válaszból azokat az elemeket, amelyekhez a felhasználónak nincs joga [28].

Skálázhatóság

A rendszer fejlesztése szempontjából a következő fontos szempont a skálázhatóság volt. A rendszer képes kell legyen a hirtelen növekedésre és a nagy teherbírásra, hogy a felhasználókat kiszolgálja. A rendszert több Thin webservert szolgáltatja. A Thin egy elterjedt választás Ruby on Rails alkalmazások üzemeltetésére. Előnye, hogy rendkívül könnyen konfigurálható, kevés erőforrást igényel és skálázható [28].

4. A CAEDUS használata, funkciói

A CAEDUS elérése: caedus.eu

A rendszer a regisztráció után jelenleg ingyenesen használható, kipróbálható. Jelenleg tesztelési fázisban van, várjuk a felhasználói visszajelzéseket az esetleges hibák kijavítása céljából (4. ábra).



4. ábra. CAEDUS - Bejelentkezési felület

TANÁRI felület

A tanári felületen a következő funkciók elérhetőek:

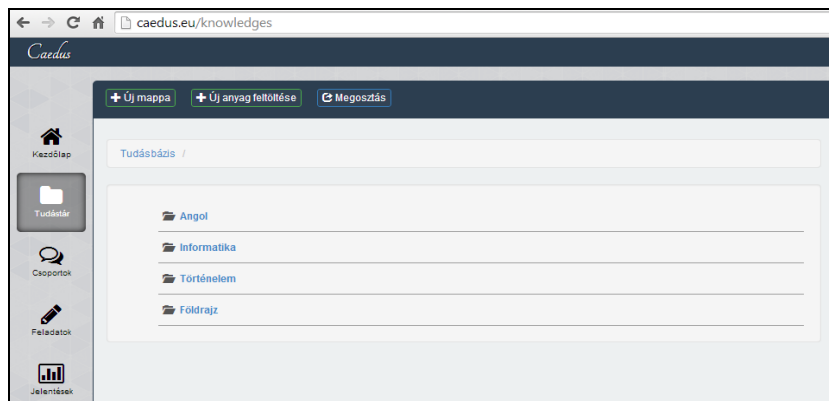
- Tudástár
- Csoportok
- Feladatok
- Jelentések

Tudástár

A Tudástár legfőbb funkciója a tanár digitális anyagainak tárolását szolgálja. A könnyű kezelés érdekében egy mappaszerkezetet alakítottunk ki, ahol a tanár rendszerezheti az anyagait (5. ábra).

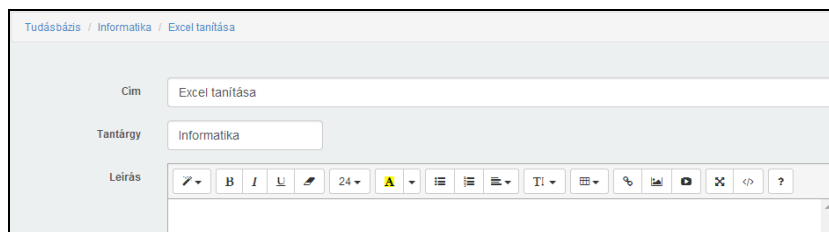
A mappában elvégeztető műveletek:

- új mappa készítése
- új anyag feltöltése
- mappa megosztása csoportokkal/tanulókkal
- anyagok megosztása csoportokkal/tanulókkal
- mappa törlése
- anyagok törlése



5. ábra. CAEDUS – Tudástár

A pedagógus nem csak előre elkészített anyagokat tölthet fel, hanem online is szerkesztheti azokat. A 6. ábrán látható szerkesztőléc lehetőséget nyújt a multimédiás anyagok készítésére, linkek képek, videók beillesztésére is.



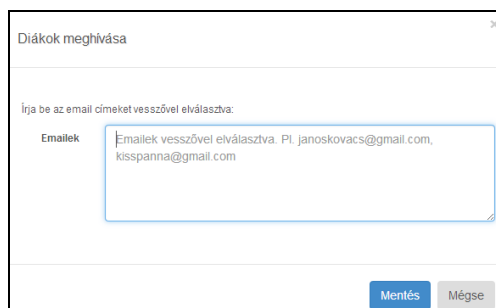
6. ábra. CAEDUS – Anyagok rendszeren belüli szerkesztése

Csoportok

Csoportok létrehozása

A csoport létrehozása során a pedagógus megadhatja a csoport nevét, hozzá rendelhet egy tantárgyat és részletezheti a csoport szerepkörét (7. ábra).

Diák meghívása



7. ábra. CAEDUS – Csoportok – Diákok meghívása

A csoport létrehozása után a tanár feladata a diákok felvétele a csoportba. Erre szolgál a 'Diákok meghívása' gomb, amely során a diákokat e-mail címeik alapján felveszi a csoportba. A diákok erről értesítést kapnak, továbbá hozzáférést nyernek a csoport anyagainak megtekintéséhez.

Anyagok megosztása

A tananyagok megosztását személyre is szabhatjuk az igényeknek megfelelően, akár néhány kijelölt tanulóval (8. ábra, bal) vagy egy teljes csoporttal is megoszthatjuk (8. ábra, jobb).



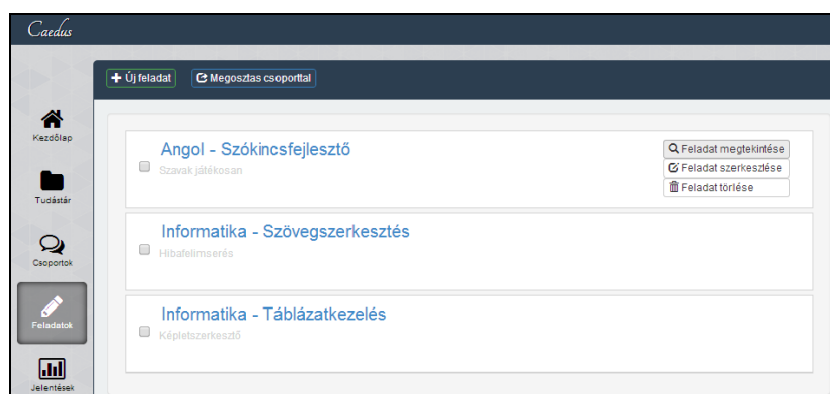
8. ábra. CAEDUS – Tanulók kijelölése (bal), csoportok kijelölése (jobb)

Társalgó – Kommunikáció

Minden csoporthoz tartozik egy közös 'Társalgó' funkció. Akár a tanár, akár a tanuló kezdeményezhet új beszélgetést, amely során egy adott tananyaghoz megvitathatnak felmerülő kérdéseket, ezáltal segíthetik egymást a tanulásban. A Társalgóba beérkező új bejegyzésekről, hozzászólásokról a csoport összes tagja automatikus e-mail értesítést kap [28].

Feladatok

A feladatok funkció nagyon kiemelkedő szerepet tölt be a rendszerben. Ez a legkomplexebb komponense. Nagy hangsúlyt fektettünk a kidolgozására. A célunk egy olyan feladatkészítő, gyakoroltató, számon kérő modul létrehozása volt, amely során a pedagógus a tananyagnak megfelelően könnyen létre tud hozni ellenőrző feladatsorokat.



9. ábra. CAEDUS - Feladatok

Komponensei:

- Új feladat létrehozása
- Feladatok megosztása csoportokkal
- Feladat megtekintése
- Feladat szerkesztése
- Feladat törlése

A teszt típusok közül a következő lista közül választhatunk:

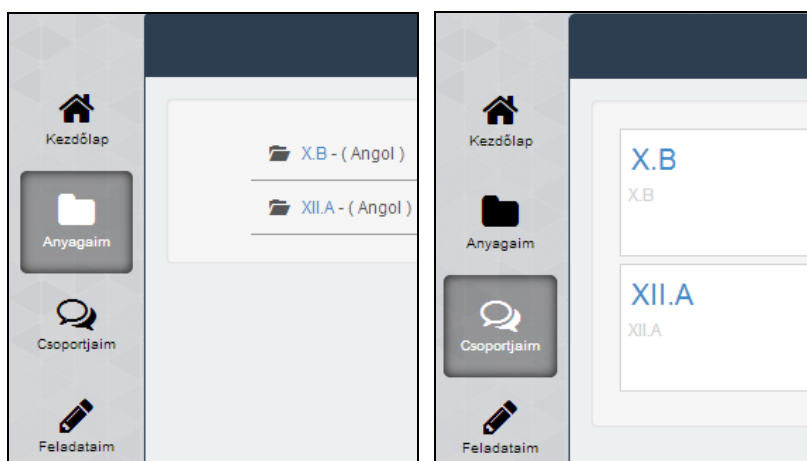
- Eldöntendő kérdések
- Két válaszos
- Helyes válasz (egy megoldás)
- Helyes válasz (több megoldás)
- Szabadszavas
- Képek közötti választás
- Rangsorolás
- Értékelő skála
- Lenyíló listás

Igyekezünk minél több kérdéstípust kezelni, de természetesen ezek akár a későbbiek során, az igényeknek megfelelően fejleszthetők, bővíthetők.

Jelentések

A pedagógus értesítést kap minden egyes feladatsor megoldása után, ezáltal követheti akár a csoport, akár az egyéni teljesítményt. Kimutatás készül arról, hogy a diák mennyi időt töltött a feladat megoldásával, milyen eredményt ért el, volt-e javítva a válasz stb. Ezáltal a pedagógus követni tudja, hogy a tanulók milyen hibákat követnek el, és lehetőség nyílik a problémák feltárására, indokolt esetben azok újbóli magyarázatára.

TANULÓI és SZÜLŐI felület



10. ábra. CAEDUS – Tanulói felület

A Tanulói felület (10. ábra) kevesebb funkcióval rendelkezik, mint a Tanári. A tanuló számára megosztott tananyagokhoz a tanulónak nincs szerkesztési joga. Csak azon anyagokat és feladatokat láthatja, melyeket a tanár kitűzött számára. A csoporton belül viszont lehetősége van a Társalgóhoz való hozzáféréshez, új bejegyzések létrehozásához.

A Szülői felület jelenleg fejlesztés alatt áll. A célja, hogy a szülő a gyereke tevékenységét nyomon tudja követni, lássa a számára kitűzött feladatokat, a feladatokon elért teljesítményét, illetve elősegítse a tanár-szülő közötti kommunikációt.

Mobil megjelenítés

A mai kor alapvető igénye a mobilitás, ma már a diákok is elvárják, hogy a szolgáltatások mobil eszközökön is megjelenjenek, ezért számunkra is alapvető igényként merült fel, hogy a szoftvernek elkészítjük a mobil klienseit is.

A klienseket főként táblagépekre optimalizálva tervezzük, mely lehetőséget nyújt a diákoknak, hogy akár utazás közben, vagy szabadidejükben bármikor el tudják érni a tananyagokat. Úgy tervezzük, hogy a mobil kliensek offline is képesek lesznek működni. Két fő platformot céloztunk meg az Androidot és iOS-t [28].

5. A CAEDUS használata a gyakorlatban

CAEDUS az informatikaoktatásban

Az Algoritmikus és Alkalmazói Készségek Tesztelése (AéAKT Projekt – TAaAS Project – Testing Algorithmic and Application Skills) című projektet a 2011/2012-es tanév első félévében indítottunk el. A projekt elsődleges célja az Informatikai Karon tanulóit megkezdő hallgatók algoritmikus és alkalmazói készségeinek tesztelése. Arra kerestük a választ, hogy a hallgatók milyen szakmai ismeretekkel rendelkeznek, hogyan értékelik saját tudásukat, továbbá, hogy mit jelent számukra az informatika, miért választották ezt a tudományterületet [3], [5], [6].

Az utóbbi években az oktatási tapasztalataink során és a felmérések eredményeit figyelembe véve egy új módszert fejlesztettünk ki a táblázatkezelés és a szövegszerkesztés algoritmikus szemléletét támogatva. A módszerek CAAD-alapú (computer algorithmic and debugging based) metakognitív megközelítések, amelyek lényegesen eredményesebbnek bizonyultak, mint a hagyományosnak számító TAEW (trial-and-error wizard based) megközelítések, amelyet több éves kutatási eredményeink is alátámasztanak [14], [15], [16], [18]. Ezen módszerek tesztelését beépítettük a rendszerbe.

CAEDUS a táblázatkezelés oktatásánál

Mivel a táblázatkezelő programok fordító programja csak szintaktikailag helyes képletek rögzítését engedélyezi, továbbá a felhasználó számára minimális, míg pedagógiai szempontból értékelhetetlen javaslatokat tesz a hibás képletek javítására, ezért létrehoztunk a CAEDUS-ban egy képletkiértékelői modult is (11. ábra), amely tartalmi javaslatokat tesz a képletek helyes megszerkesztésére. Nagyon hasznos a tanulók számára, mivel azonnali visszajelzést ad, hogy milyen hibát követettek el, illetve meghatározza a hiba helyét, ezáltal segít a feladat megoldásban [8], [17].

	A	B	C	D	E
1	ország	foldresz	fovaros	terulet	fo_ezer_02
2	Afghanistan	Asia	Kabul	647500	27756
3	Albania	Europe	Tirana	28748	3545
4	Algeria	Africa	Algiers	2381740	32278
5	American Samoa	Oceania	Pago Pago	199	69
6	Andorra	Europe	Andorra la Vella	468	68
7	Angola	Africa	Luanda	1246700	10593
8	Anguilla	Amerika	The Valley	102	12
233	Yemen	Asia	Sanaa	527970	18701
234	Yugoslavia	Europe	Belgrade	102350	10657
235	Zambia	Africa	Lusaka	752614	9959
236	Zimbabwe	Africa	Harare	390580	11377

1. Írassuk ki a H1 cellában megadott földrész országainak összterületét!

Megoldás:

11. ábra. CAEDUS – Táblázatkezelés

CAEDUS a szövegszerkesztés oktatásánál

Az általunk kidolgozott CAAD-típusú módszer lényege, hogy megtanítsunk olyan technikákat, amelyek segítségével a felhasználók képesek lesznek szöveges dokumentumokban a hibák felismerésére, azok kategorizálására, valamint a hibák kijavítására, és a későbbiekben egy jól szerkesztett dokumentum létrehozására [7], [9].

A tanulási folyamat első lépéseként már létező dokumentumok nyomtatott verzióit elemezzük (12. ábra), annak érdekében, hogy felszámoljuk azt a tévhitet, hogy a szövegszerkesztés elsődleges célja, hogy nyomtatásban jól nézzen ki a dokumentum. A papír alapú elemzés során felismerhetőek a szöveg szintaktikai, szemantikai, tipográfiai hibái. A módszer ezen szakaszának a begyakorlására használhatjuk a CAEDUS-t [4], [10].

The screenshot shows a document with several errors marked by blue arrows and the number '2.'. The errors include:

- Incorrect use of the Hungarian definite article 'a' before 'HATÉKONY-KOMMUNIKÁCIÓ-SZEREPÉ-A-VEZETÉSBEN'.
- Incorrect use of the definite article 'a' before 'szabályai'.
- Incorrect use of the definite article 'a' before 'Békéscsaba, 2002 november 8.'.
- Incorrect use of the definite article 'a' before 'Tisztelettel'.
- Incorrect use of the definite article 'a' before 'Szegegyvári Tamásné'.
- Incorrect use of the definite article 'a' before 'igazgató'.
- Incorrect use of the definite article 'a' before 'igazgató'.

Below the document, there is a legend for error types:

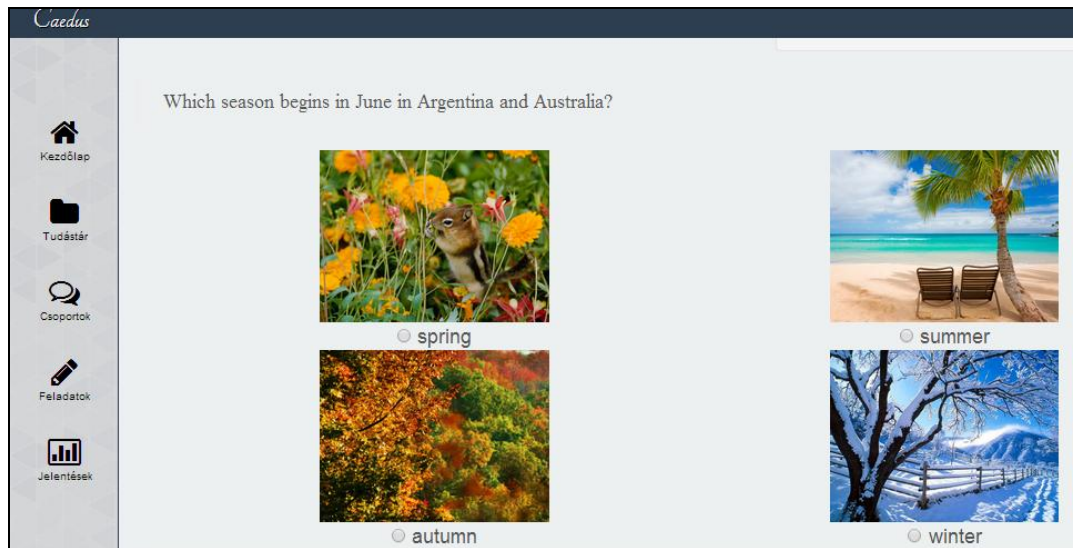
- fölösleges szóközök
- fölösleges enterek
- hibás margó
- hibás sorköz

The task instruction is: "Nevezze meg a képen megjelölt hibát/hibákat!"

12. ábra. CAEDUS –Szövegszerkesztés

CAEDUS az angol nyelv oktatásában

A CAEDUS oktatási keretrendszert nem csak az informatikában, hanem gyakorló angol tanárokkal is teszteltük. Kíváncsiak voltunk arra, hogy az informatikában kevésbé járatos tanárok, hogyan tudják kezelni. A visszajelzések pozitívak voltak, önállóan fedezték fel a rendszer lehetőségeit (13. ábra), a tesztelés során felmerülő nehézségek építő jellegű visszajelzéseként szolgáltak, hogy a rendszer esetleges gyengeségeit a jövőben kijavíthassuk.



13. ábra. CAEDUS –Angol nyelv

6. Összefoglalás

A CAEDUS oktatási keretrendszert a közoktatásban tanító pedagógusok számára fejlesztettük, hogy megkönnyítsük a mindennapi munkájukat.

Létrehoztunk egy olyan tanulás/tanítás menedzsment rendszert, amely képes a több tízezer felhasználót párhuzamosan kiszolgálni, skálázható – a felhasználói visszajelzések alapján bővíthető –, egyszerű, képes a különböző tartalmakat kezelni, felhasználóbarát, stabil és biztonságos.

Úgy gondoljuk, hogy sikerült egy könnyen hozzáférhető és használható rendszert fejleszteni. A jelenlegi tesztelési fázis és az apróbb hibák javítása után reményeink szerint egy széles körben elterjedt szoftverről beszélhetünk. Az igazi sikerét, eredményességét majd a felhasználói visszajelzések fogják igazán megadni.

7. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Köszönettel tartozom Ujlaki Imre és Cselle Lajos szoftverfejlesztőknek, továbbá Dr. Csernoch Máriának a szakmai tanácsaiért és Árva Anita angol tanárnőnek a rendszer teszteléséért.

Irodalomjegyzék

- [1] Amir Salihefendic: Model View Controller: History, theory and usage, <http://amix.dk/blog/post/19615>. Letöltés dátuma: 2014.01.29.
- [2] John Hall: Assessing Learning Management Systems; *Chief Learning Officer Magazine* – 2003 Január, 27-35 old.
- [3] Mária Csernoch és Piroska Biró: Spreadsheet misconceptions, spreadsheet errors, Oktatóskutatás határon innen és túl. *HERA Évkönyvek I.*, ed. Juhász Erika, Kozma Tamás, Publisher: Belvedere Meridionale, Szeged, (2014), 370–395.
- [4] Mária Csernoch és Piroska Biró: Szövegszerkesztés algoritmikus szemlélettel. *Digitális Módszertár. Jó gyakorlatok az oktatásban*. Letöltés dátuma: 2014. 07. 07. http://oktataskepzes.tka.hu/pages/idea/index.php?page_id=1113&idea_id=126.
- [5] Mária Csernoch és Piroska Biró: Teachers' Assessment and Students' Self-Assessment on The Students' Spreadsheet Knowledge. *EDULEARN13 Proceedings July 1st-3rd, 2013* – Barcelona, Spain. Publisher: IATED, (2013) 949–956.
- [6] Mária Csernoch és Piroska Biró: Button-up technikák hatékonyságának vizsgálata informatika szakos hallgatók táblázatkezelés-oktatásában. T. Kozma és I. Perjés, *Új kutatások a neveléstudományokban 2012*, ELTE Eötvös kiadó, (2013), 352–369.
- [7] Mária Csernoch: Clearing Up Misconceptions About Teaching Text Editing. *Proceedings of ICERI2011 Conference, ICERI 2011*, 4th International Conference of Education, Research and Innovation, Madrid, (2011). 407–415.
- [8] Mária Csernoch: Introducing Conditional Array Formulas in Spreadsheet Classes. *EDULEARN12 Proceedings*. Barcelona, Publisher: IATED, (2012), 7270–7279.
- [9] Mária Csernoch: Teaching word processing – the practice. *Teaching Mathematics and Computer Science*. 8/2 (2010). 247–262.
- [10] Mária Csernoch: Teaching word processing – the theory behind. *Teaching Mathematics and Computer Science*. 1(2009), 119–137.
- [11] Matz: The Father Of Ruby. <http://davincicoders.com/davinci-college-a-transformationaleducation-experience-designed-specifically-for-your-future/matz/>. Letöltés dátuma: 2014.01.29.
- [12] Moodle statisztika, verziók, történet. <https://moodle.org/>. Letöltés dátuma 2014. 07. 07.
- [13] MVC minta Ruby on Rails-ben. <http://betterexplained.com/wpcontent/uploads/rails/mvc-rails.png>. Letöltés dátuma: 2014.01.29.
- [14] Piroska Biró és Mária Csernoch: Deep and surface structural metacognitive abilities of the first year students of Informatics. *4th IEEE International Conference on Cognitive Infocommunications, Proceedings*, Budapest, (2013), 521–526.

- [15] Piroska Biró és Mária Csernoch: Programming skills of the first year students of Informatics. *XXIII. International Conference on Computer Science 2013*, EMT, in Hungarian, (2013). 154–159.
- [16] Piroska Biró és Mária Csernoch: Táblázatkezelés algoritmikus megközelítése. *Kiss Árpád Emlékkonferencia Tanulmánykötete 2013*, Debrecen, (2013). Elfogadva.
- [17] Piroska Biró és Mária Csernoch: Táblázatkezelés könnyen, hatékonyan, algoritmikus szemlélettel. *Digitális Módszertár. Jó gyakorlatok az oktatásban*. Letöltés dátuma: 2014. 07. 07. http://oktataskepzes.tka.hu/pages/idea/index.php?page_id=1113&idea_id=124.
- [18] Piroska Biró, Mária Csernoch, Kálmán Abari és János Máth: First year students' algorithmic skills in tertiary Computer Science education, *9th International Conference on Knowledge, Information and Creativity Support Systems*. Cyprus, November 6–8., (2014), submitted.
- [19] Piroska Biró: Attitude of Interactive Whiteboard Users. *Proceedings of the 4th International Conference on Computer Supported Education*, Vol. 1, (2012), 348–356.
- [20] Piroska Biró: E-learning, B-learning és Projektpedagógia. *Projektpedagógia-Projektmódszer VI.*, Kecskemét, (2006), 53–59.
- [21] Piroska Biró: Interactive whiteboard in mathematics education. *Acta Universitatis Sapientiae, Social Analysis*, Vol. 2, No. 1, (2012), 111–127.
- [22] Piroska Biró: Interaktív tábla az oktatásban, *Tavaszi Szél Doktoranduszok Országos Szövetsége Konferencia*, Budapest, (2008), 665–671.
- [23] Piroska Biró: Students and the Interactive Whiteboard, *Acta Didactica Napocensia*, No. 2-3, Vol. 4, (2011), 29–38.
- [24] Piroska Biró: Teachers and the Interactive Whiteboard. *Teaching Mathematics and Computer Science*, (2012), 10(2), 281–298.
- [25] Piroska Biró: Újdonságok a tanítási órán, INFODIDACT 2010 Konferencia, *CD konferencia kiadvány*, Szombathely.
- [26] Tóth Zsolt és Bessenyei István: A konstruktivista oktatás környezete és a Moodle. *Információs Társadalom folyóirat*, 2008. VIII. évfolyam 3. szám, http://www.infonia.hu/digitalis_folyoirat/2008_3/2008_3_toth_istvan_bessenyei_istvan.pdf. Letöltés dátuma: 2014. 07.07.
- [27] Török Mátyás: e-Learning oktatásmenedzsment keretrendszerek. *Szakedolgozat*. Budapesti Közgazdaságtudományi és Államigazgatási Egyetem, Gazdálkodástudományi Kar, Információrendszerek Tanszék, Budapest, 2003.
- [28] Ujlaki Imre: Kollaborációs rendszer fejlesztése a közoktatás számára. *Diplomamunka*. Debreceni Egyetem, Informatikai Kar, Debrecen, 2013.
- [29] Wikipédia Observer minta: http://en.wikipedia.org/wiki/Observer_pattern. Letöltés dátuma: 2014.01.29.

A számítógéppel támogatott oktatás egy SQC alapú minőségbiztosítási modelljéről

On an SQC-based Model of Computer Aided Teaching

Fazekas K. Anna^a, Fazekas Gábor^b, Várterész Magda^b

^aDebreceni Egyetem Kossuth Lajos Gyakorló Gimnáziuma, 4029 Debrecen, Csengő u. 4.
kfanna@gmail.com

^bDebreceni Egyetem, Informatikai Kar, 4032 Debrecen, Kassai út 26.
{fazekas.gabor, varteresz.magda}@inf.unideb.hu

Absztrakt: Az ISO9000 szabványoknak az 1980-as évek végén történt bevezetése egyfajta forradalmat váltott ki a minőségbiztosítás területén. Széles körben lehetséges alkalmazhatóságuk miatt mára majdnem fél millió szervezet adaptálta ezeket a szabványokat. A regisztrált felhasználók között azonban viszonylag kevés oktatási intézményt és még kevesebb középiskolát találunk. Ennek oka részben az „oktatás termékének”, azaz a végzett hallgatóknak és tanulóknak „komplexitása és bonyolultsága”. Másrészt az is a probléma forrása lehet, hogy hiányoznak és szegényesek azok, az oktatás teljesítményének kvantitatív mérésére ténylegesen használt eszközök, amelyek effektív visszacsatolási módszerek kidolgozását megkönnyítenék.

A dolgozatban S. Karapetrovic-nak a *statisztikai minőségirányítás* (SQC) elvére alapozott ötletét felhasználva, az SQC-nak a nyelvoktatásban lehetséges alkalmazásáról számolunk be. De úgy gondoljuk, ez az ötlet más tárgyak oktatásánál is felhasználható. Az ötlet megvalósításában alapvető szerepet játszanak a számítógépek.

Kulcsszavak: e-Learning, SQC, ISO 9000

Abstract: Ever since their introduction in the late eighties of the previous century the ISO 9000 standards have resulted in a revolution in the quality assurance issues. Because of their wide range of applicability, today almost half a billion organizations have adopted these standards. However, relatively few educational institutions, and even fewer secondary schools are registered worldwide. This can be partly due to the complexity and difficulty of the “product of education”, i.e. the graduated students and pupils. Another problem can be the lack of clearly defined quantitative tools for measuring the performance of education and establishing the most effective feedback processes. Statistical Quality Control was first introduced by W.A. Shewhart for the control of classical (factory) production processes in the early 1920s.

In this paper, following the pioneer SQC based ideas of S. Karapetrovic we present a possible adaptation/extension of SQC for secondary school language teaching. We believe that these ideas can be applied in the teaching of many other subjects. Computers play a very important role in the realization.

Keywords: e-Learning, SQC, ISO 9000

1. Bevezetés, történeti megjegyzések

Az E-learning/(CAI) ötlete „rég”, majdnem egyidős a számítógépekkel. Az 1962-ben már rendelkezésre álló számítógép terminálok, lehetővé tették, hogy létrejöjjön a BASIC programozási nyelv, kifejezetten a programozás oktatására, John F. Kemeny (Kemény János) ötletéből és gondozásában.

Fontos állomás volt a személyi számítógépek megjelenése: a számítógép a tanuláshoz szükséges motivációs bázis megteremtésének fontos eleme lett!

Az elektromos oktatás társadalmilag elismert értékeinek elismerését példázta az EU szándéknyilatkozata az oktatásról és képzésről (Lisbon objectives 1999/2000) []:

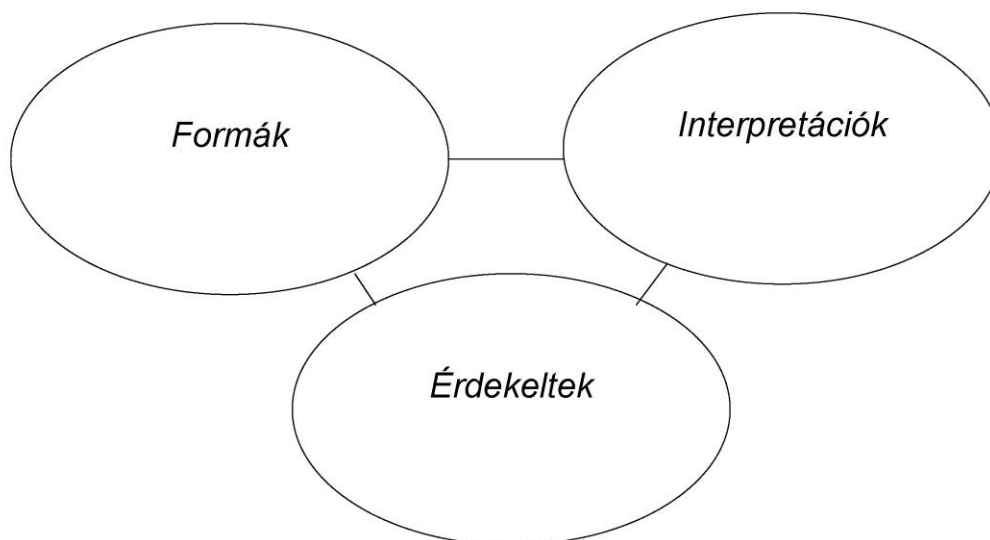
- „Europe should become, by 2010, ...
 - the most competitive and dynamic knowledge-based economy in the world, capable of sustainable economic growth with more and better jobs and greater social cohesion”
- „The future objectives for education & training ...
 - Increasing the quality and effectiveness of education and training systems in the European Union
 - Facilitating the access of all to the education and training systems
 - Opening up education and training systems to the wider world”

A kérdés új megvilágításba került az újabb technológiák megjelenésével: ICT, Internet, WWW. Aktívak maradtak az EU kapcsolódó szervezetei, - a teljesség igénye nélkül említve - kibocsátották az EU „e-Learning Action Plan” (2001) dokumentumot, amely a közösségi kezdeményezések koordinálását segítette. Kibocsátották az „eLearning Initiative” (2000-2003) projekt felhívást, amely 65 projektet finanszírozott, 27.2 millió Euro keretből. Kibocsátották az „eLearning Programme” (2004-2006) dokumentumot a fő koncepciók megfogalmazására.

Ezek lényeges következménye lett az a felismerés, miszerint a minőség a képzés és oktatás modernizálásának kritikus elemévé vált. Erről egy kiváló „State of the Art” leírást találhatunk *Ehlers and Pawlowski gyűjteményes kötetében []*.

2. Minőség koncepciók

A minőség egy normatív szabályrendszerként interpretálható, amely bizonyos viszonyokra vonatkozik. Ezeket a viszonyokat általános értelemben szemlélteti az alábbi ábra:



A részleteket leginkább az e-learning szabványok fejezik ki, amelyek komoly – érdekeket tükröző/védő – vitákban alakultak ki. Ennek megfelelően létezik:

Kormányzati / befektetői nézőpont:

Itt a domináns célok: költségcsökkentés, titkos befektetés, új piaci térhódítások.

Létezik az oktatói közösség nézőpontja:

Ezt leginkább félelem dominálja az (egyéni) kreatív megoldások korlátozásától, merthogy a szabványokat a flexibilitást és kreativitást korlátozó eszközöknek tekintik.

Általában elmondható, hogy a szabványok követése hatalmas további munkaráfordítást igényel.

A megoldás kompromisszumokat igényel. Jelenlegi kompromisszumok:

A minőségi szabványok új generációi csupán egy alapszintű keretrendszert adnak meg.

A szervezetek saját minőségi rendszert dolgozhatnak ki.

Rugalmasság megoldása lehet, hogy különböző szabványokat alkalmazunk a különböző alkalmazási környezetekhez.

Az e-learning szabványok osztályozása jelenleg három aspektusból történik:

I. Minőség irányítás és minőség biztosítás

Folyamat-orientált

ISO 9000:2000, EFQM,

ISO/IEC 19796-1, ...

Termék-orientált

Követelményrendszerek-Criteria (dmmv- *German Multimedia Association*, *ASTD - American Society for Training & Development*),

DIN Reference Criteria

Kompetencia orientált

Kompetencia definíciók és kiértékelések (assessment) képezik az alapját.

II. Tanulási technológiák szabványai

Tartalom (Content)

Learning Object Metadata (LOM)

Szereplők (Actors)

Learner information Package (LIP), PAPI

Irányítás/kezelés (Management)

SCORM, IMS Content Packagig

Didaktika/módszertanok (Didactics)

EML/IMS Learning Design, DIN Didactical Object Model

Környezet (Context)

Situation and Context Descriptions / KM Interface

Hordozhatóság / mobilitás (Mobility)

Location and Context Awareness, Synchronization

III. Kapcsolódó szabványok

Technológia szabványok

SGML,XML, ...

GPRS,UMTS, ...

MPEG-x, ...

TCP/IP, ...

Folyamat szabványok

E-Business Standards (e.g. EBXML, OASIS)

School Standards (e.g. School Interoperability Framework)

Curriculum Standars (e.g. ECTS)

Jogi szabványok

Digital Rights Expression Language (DREL)

3. Egy új megközelítés: SQC

A szabványok hiányosságai:

- Kvalitatív ismérvekkel dolgozik (jobb, érdekesebb, hatékonyabb, stb ...).
- Mérési módszerek helyett „irányelvek” (pl.: legyen szép, barátságos, jó; az oktatás segítse a gyereket „jobbá válni”, stb ...)
- Ha vannak is numerikus jellemzők, ezekre legfeljebb a leíró statisztikákat (átlag, minimum, maximum, ...) alkalmazzák.
- Más: értelmezési, terminológiai problémák.

Megoldás lehet:

- Felismerni, hogy az oktatás folyamatát milyen számszerűsít-hető adatokkal lehet jellemezni.
- Felismerni ezeknek az adatoknak a sztochasztikus jellegét.
- Megkeresni és alkalmazni a releváns statisztikai módszereket.

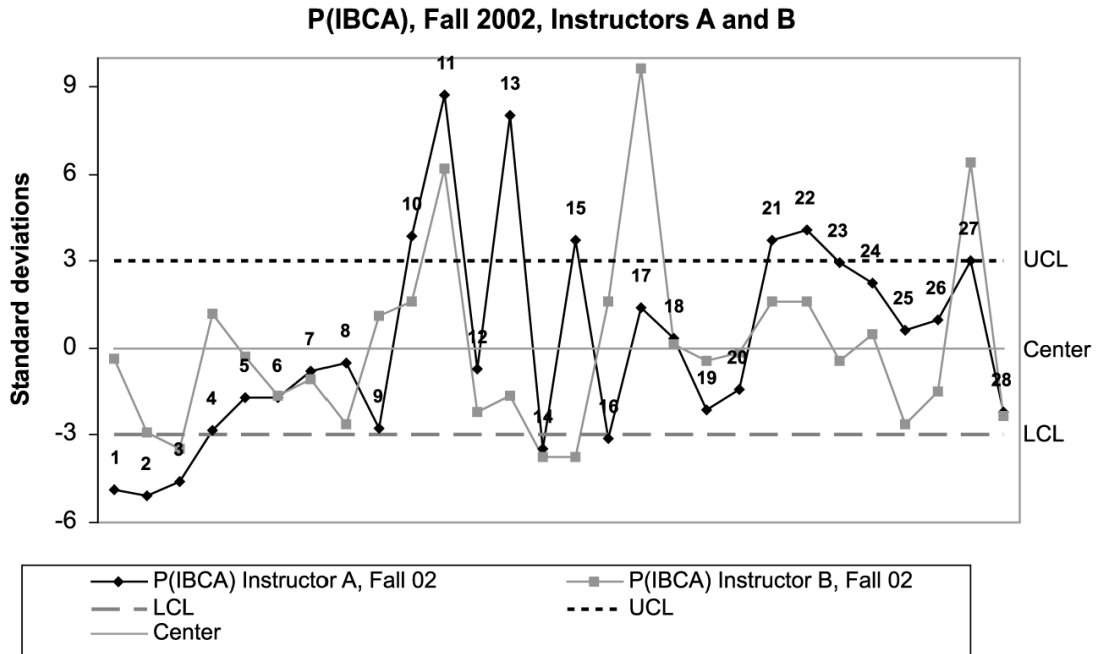
- Az oktatási /képzési folyamat produkciós / termelési folyamat: „terméke a végzett diák /tanuló”.
- Termékek minőségbiztosítására alkalmazott bevált módszer a statisztikai minőség-ellenőrzés (SQC).
- Az SQC alapváltozatát **Walter Shewhart** fogalmazta meg 1931-ben megjelent híres dolgozatában, majd később mások (pl. **Deming, Ishikawa**) tökéletesítették.
- **Alapötlet:**
 - Egy folyamat számszerűen mérhető és véletlenszerűen változó kimeneti értékeit két okra vezethetjük vissza: általános és speciális okokra.
 - Az általános okokból bekövetkező változásokat normálisnak tekintjük, a speciális okokra visszavezethetőket problémaként interpretáljuk.
 - Lehetséges modell: **kontrol kártya**

A kontrol kártyák szerkezete:koordináta rendszer

- Függőleges tengelyen: a statisztikák értékei.
- Vízszintes tengelyen: csoportok (a statisztikák argumentumai).
- Három vízszintes vonal: a statisztikák átlaga, felső-, ill. alsó korlátja.
- Ha a statisztikák értékei a korlátok közé esnek, akkor a folyamat normális viselkedésű.
- Ha a statisztikák értékei a korlátokon kívül vannak, a folyamat viselkedése nem normális, beavatkozás szükséges.

Példák:

- Palacktöltő futószalag, anyag darabolás, teszt kérdéssor(!)
- Az alábbi ábra egy kontrol kártyát szemléltet.



Alkalmazások(1: USA Mérnökképzés)

- Alap: ABET 2000 criteria.
- A számítógépek csak a kiértékelésben vesznek részt!

Alkalmazások(1: DE, Informatikus képzés)

- Az E-learning folyamat a számítógépen naplózható.
- Az összegyűjtött adatok könnyen feldolgozhatók.
- Előny: gyors visszacsatolás lehetséges (mesterséges intelligencia módszerek)!
- Egy debreceni projekt (2007):
 - MOODLE + R
 - Moodle forráskód céloknak megfelelően módosítható.
 - R szkriptek hozhatók létre a Moodle adatainak elemzésére.
- Nyelvoktatási projekt keretei körvonalazódnak.

Irodalomjegyzék

- [1] Karapetrovic, Stanislav, Rajamani, Divakar, *An approach to the application of statistical quality control techniques in engineering courses*, Int. Journal of Engineering, July 1998.
- [2] Grygoryev, Konstantin, Karapetrovic, Stanislav, *An integrated system for education performance measurement, modeling and management at the classroom level*, The TQM Magazine, Vol.17 No2, 2005. pp. 121-136.
- [3] U-D. Ehlers, J.M. Pawlowski, (eds.), *Handbook on Quality and Standardization in E-Learning*, Springer Verlag, Berlin, Heidelberg, 2006.
- [4] Fazekas G., Ispány M, Juhász I, Kis B., *Elektronikus oktatási környezetek minőségbiztosítási modelljei*, Informatika a felsőoktatásban, Aug. 27-29. 2008, Debrecen, proceedings, Debrecen, 2008.
- [5] Anna K. Fazekas, Gabor Fazekas, Magda Varteres, *An SQC based model for the quality improvement of computer aided teaching*, Abstracts MaCS'12, Zoltan Csörnyei (ed.) , ELTE, Budapest, 9.2.2012, p.37.

Mikrokontroller programozó diákműhely a Gábor Dénes Főiskolán

Microcontroller Programming Student Workshop in Dennis Gabor College

Friedel Attila

Gábor Dénes Főiskola Informatikai Intézet

friedel@gdf.hu

Absztrakt: A Gábor Dénes Főiskola mérnök informatikus, műszaki alkalmazások sávós hallgatóinak a hatodik szemeszterben kötelező tantárgya a Beágyazott rendszerek. A tárgy keretében a hallgatók megismerkedhetnek a Microchip cég PIC16F690-es, 8 bites mikrokontrollerével, illetve annak assembly nyelven történő programozásával. Tapasztalataim szerint a hallgatók nagyobb részének problémát okoz, hogy az előzőleg megszerzett objektumorientált negyedik generációs nyelvi tapasztalataival szemben itt alacsony szintű, gépközeli nyelven kell programozni és a sikeres program megírásához elengedhetetlen a hardver részletes ismerete is. A cikkben bemutatom a Gábor Dénes Tehetségpont gondozásában létrehozott Mikrokontroller programozó Diákműhelyt, a létrejöttének körülményeit, az abban folyó munkát. Részletesen tárgyalok egy konkrét fejlesztési projektet, annak hardveres és szoftveres oldaláról. A projekt keretei között két hat oldalú dobókockával történő dobások sorozatát szimulálok, különböző véletlenszám generáló algoritmusok felhasználásával. A generált számok összegét soros kapcsolaton keresztül juttatom el a mikrokontrollerről egy számítógépre, ami az elosztást, a gyakorisági adatokat oszlopdiaqramon ábrázolja. A feladat része a soros kommunikáció hardveres paramétereinek meghatározása, illetve a protokoll szoftveres megvalósítása is.

Kulcsszavak: felsőoktatás, mikrokontroller, hardver közeli programozás, soros kommunikáció, diákműhely, tehetséggondozás

Abstract: Those students of Dennis Gabor College, during their Bachelor's studies in Computer Engineering, who are enrolled in Engineering Application specialisation of said Bachelor's course, are required to complete a compulsory course named "Embedded Systems". Said course features the Microchip company's PIC16F690 model 8-bit microcontroller and students are required to write their programs using the Assembly language. It is my experience that students find this exercise more than challenging, bordering on it being considered especially difficult. I have found that the main reason behind this phenomenon is the students' unfamiliarity with low-level programming languages that is due to their exclusive exposure to those languages and programming practices that are inside highly object-oriented environments and are dealing with fourth generation computers. Certain concepts of machine-oriented programming, along with detailed knowledge of the specific hardware, is needed in order to be able to write successful programs during this course at Dennis Gabor College. During the course of this article, I will discuss the Microcontroller Student Workshop that was housed by the Dennis Gabor Talentpoint, elaborating on the factors that led to the starting of the workshop and the programming exercises the workshop students were faced with. I will go into further detail about a certain development project, examining both its hardware and software aspects. During this project, I simulated a series of dice rolls using the model of two six-sided dice and a set of random generator algorithms. The sum of the randomly generated pair is transferred from the microcontroller to the computer using a serial connection. These pieces of data are then analysed for distribution and frequency, with the results displayed as bar charts. The exercise contained defining the hardware parameters of the particular serial connection and also the software implementation of said protocol.

Keywords: higher education, microcontroller, machine-oriented programming, serial communication, student workshop, talent management

1. Bevezetés

A Magyar Tehetségsegítő Szervezetek Szövetségét [1] magyarországi és határon túli tehetséggondozással foglalkozó szervezetek hozták létre. Célul tűzték ki, hogy biztosítsák a tagszervezetek közötti álláspontok egyeztetését, tanulmányozzák a hazai és külföldi példákat, az így szerzett tapasztalatokat szakmai fórumok szervezésével továbbadják, támogatási lehetőségeket és új támogatási formákat kutassanak fel. Pályázatok kiírásával segítik a tehetséggondozás rendszerének további fejlődését. A szövetség nyitott, bármely olyan szervezet tagja lehet, amely teljesíti a szövetség által támasztott követelményeket. A szövetség küldetése az, hogy bárki számára lehetővé tegye, hogy megtalálja és kibontakoztassa a magában rejlő tehetséget. Ennek a küldetésnek a teljesítésére alakult a tehetségpont hálózat. A hálózat tagjai szoros kapcsolatban állnak egymással, ezáltal lehetőség nyílik a helyi értékek közvetítésére már területek számára. A tehetségpontok feladata, hogy segítséget nyújtsanak az érdeklődő fiatalok számára tehetségük felismeréséhez és kibontakoztatásához, keressék az együttműködés lehetőségét más tehetségpontokkal és segítsék elő a tehetséggondozó kezdeményezések emberi és anyagi erőforráshoz jutását.

A Gábor Dénes Főiskolán [2] működő Gábor Dénes Tehetségpont [3] tagja a tehetségpont hálózatnak, és idén második alkalommal szerzett sikeres akkreditációt kiváló minősítéssel. Feladatai között szerepel diákműhelyek létrehozása és fenntartása, ahol az érdeklődő, tehetséges diákok a kötelező tananyagot kívül többlet tudást szerezhetnek. A diákműhely témájának összeállítása hallgatói érdeklődés alapján ölt formát, de nem ritka az önálló hallgatói kezdeményezés sem. A diákműhely foglalkozások nyíltak, nem korlátozódnak a főiskola hallgatóira, középiskolás diákoktól a már végzett, de érdeklődő hallgatókig mindenkit szívesen fogadnak. Az aktuális szemeszter diákműhelyeiről a főiskola honlapjáról [4], Ilias LCMS rendszeréből [5], szórólapokról, facebook üzenetből, a Neptun tanulmányi rendszeren keresztül küldött körlevélből és természetesen tanári ajánlásokat követve szerezhetnek tudomást az érdeklődők.

A Mikrokontroller programozó diákműhely célja megismertetni a hallgatókat a belépő szintű, 8 bites mikrovezérlők világával, gyakorlatban mutatva be a hardverközeli programozást. Nagyobb hangsúlyt fektet a gyakorlati megvalósításra, az elméletből csak az éppen szükséges ismereteket tárgyalva. Ezzel a módszerrel több kis sikerélményen keresztül lehet fenntartani az érdeklődést. A tehetségpont nyílt filozófiájából adódóan a célközönség szerteágazó: óraadók és külsős diákok ugyanolyan szeretettel látott vendégek, mint a főiskola hallgatói. Nem feltétel a főiskolai tantárgyak előzetes teljesítése, de előny, ha az érdeklődő rendelkezik alapszintű programozási ismeretekkel és hardveres szemlélettel. A műhely működése idejére igénybe veheti az elektronika labor tulajdonában álló PICkit2 Starter Kit [6] nevű, PIC16F690-es [7] mikrokontrollert tartalmazó fejlesztő készletet, az ingyenes mikroC for PIC [8] szoftveres fejlesztőkörnyezet demo változatát, és a labor számára fenntartott termet. A műhelyfoglalkozások 4 alkalomra, alkalmanként 6 óra időtartamra kerülnek ütemezésre, a nap délutáni felében. Minden érdeklődő számára külön számítógép és fejlesztőkészlet áll rendelkezésre. A foglalkozások három egységre bonthatók: az elmélet ismertetésére, az elméletben hallott ismeretek gyakorlati alkalmazására és az alkalmazás közben felmerülő kérdések/hibák megbeszélésére, kijavítására. Kisebb feladatok esetén ez a hármas egység többször is megismétlődhet egy foglalkozás alkalmával. A tehetségpont részéről elvárás, hogy a diákműhely a meghirdetett időtartam alatt valamilyen kézzel fogható, működőképes végtermékkel álljon elő.

2. Kockadobás projektfeladat

A kockadobás feladat megvalósítása ideális választás a diákműhelyben dolgozó résztvevők számára, mert kivitelezhető a rendelkezésre álló idő alatt, nagyrészt benne marad a tesztpanel biztosította hardveres lehetőségek határait, és a működtető szoftver sem lépi túl a fejlesztőkörnyezet demo változatának korlátait. Tehát készüljön két program, amelyek közül az egyik C# nyelvű Windows Forms-os alkalmazás, a másik pedig a mikrokontrolleren fut. A két program együttműködve szimuláljon előre megadott dobásszámú kísérletet, ahol egy kísérlet két szabályos dobókocka véletlenszerű feldobását jelenti és a dobott számok összegét adja meg. A dobások száma előre rögzített, de több ilyen lehetőség közül lehessen választani. A dobásokat véletlenszám generálással kell szimulálni. Ez a program a mikrokontrolleren kell, hogy fusson, a dobások számának kiválasztását, és a két kockával dobott számok összegét soros kapcsolaton keresztül kell kommunikálni. A soros kapcsolathoz készüljön olyan egyszerű protokoll, ami lehetővé teszi a tesztelést soros terminál emulátor program használatával. A Windows operációs rendszer alatt futó program biztosítson grafikus felhasználói felületet a parancsok kiadására a mikrokontroller felé, illetve az onnan érkező dobások összegeinek eloszlását ábrázolja grafikomon. A feladat megoldásához a rendelkezésre álló PICkit2 Starter Kit hardver, és a mikroC for PIC fejlesztőkörnyezet használható.

A jól működő program elkészítéséhez meg kell ismerni a mikrokontroller hardverét, annak felépítését és perifériáit. Minden szükséges információ részletesen megtalálható az adatlapon [7]. Szoftver oldalon C nyelvű fejlesztőkörnyezetben érdemes dolgozni, mert a fordító sok hardver-specifikus műveletet elrejt (például a bankváltásokat és a nyolcbitesnél hosszabb változók kezelését), vagy leegyszerűsít (például a speciális funkciójú regisztereknek és bitjeiknek a névvel való hivatkozásának lehetőségét). Ezen felül a mikroC-hez jól használható, lényegre törő és példakódokat is tartalmazó súgó is tartozik.

Két szabályos dobókockával végzett dobások összegeinek eloszlásáról tudjuk, hogy normális eloszlást követ. Ezért a véletlenszámmal történő szimulációt is úgy kell elkészíteni, hogy kövesse a normális eloszlást. A jól működő véletlenszám generátor az összes generálható értéket azonos valószínűséggel állítja elő, ezért helytelen egy számot készíttetni vele a 2-12 zárt intervallumban. A helyes megoldás két darab véletlen szám készítése az 1-6 zárt intervallumban, majd a két szám összegzése. Arra lehet számítani, hogy minél nagyobb mintát veszünk figyelembe, az eloszlás annál inkább meg fogja közelíteni a Gauss görbét. A jelenséget bemutatandó négy féle dobásszám közül lehessen választani: 10, 100, 1000 és 5000, továbbá legyen lehetőség megállítani a dobások generálását futás közben.

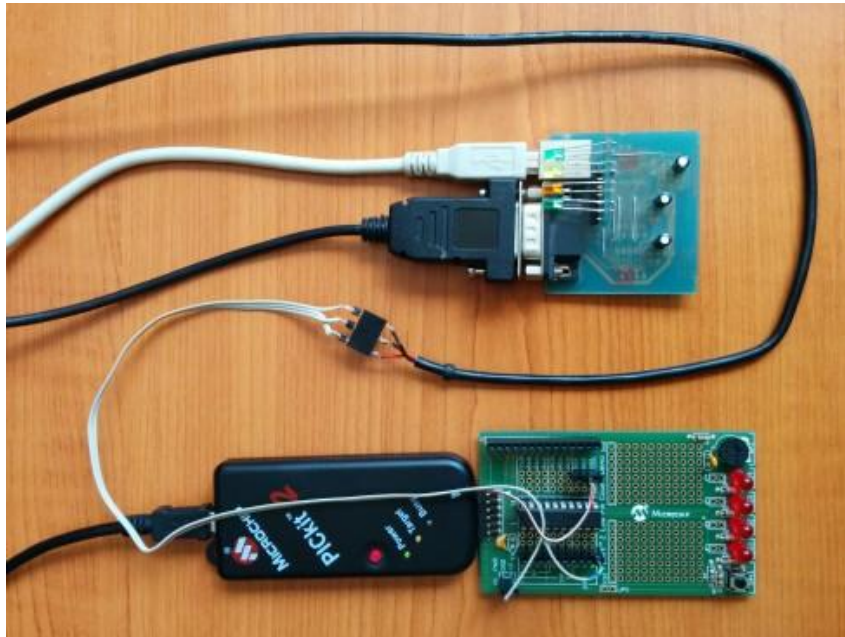
A soros vonali kommunikáció tehát a következők szerint megvalósuljon meg: ötféle különböző üzenetet lehessen küldeni a mikrokontrollernek, az üzenetek ne tartalmazzanak lezáró részt. A mikrokontroller felől tizenkét féle válasz érkezhetsz, kettőtől tizenkettőig a dobott számok, illetve egy kérdőjel, hogy legyen lehetőség hibás üzenetre reagálni. A kérdőjel után, illetve a generált véletlen számsorozat után érkezzen egy CR+LF karakter páros, így jelezve a válasz végét. A kommunikáció ideje alatt a mikrokontroller adjon vizuális visszajelzést a felhasználó számára.

3. Hardveres megvalósítás

A PIC16F690-es mikrokontroller 8 bites, Harvard architektúrájú, RISC felépítésű, 20 lábú, 7 KB programtárral és 256 byte adattárral rendelkező áramköri elem. Ez a vezérlő foglal helyet a PICKit2 Starter Kit panelen, ami programozó és általános célú csatlakozót, egy nyomógombot, egy potenciométert és négy darab ledet biztosít a felhasználó számára. A vezérlő sokkal több perifériával rendelkezik mint ahány lába van, ezért egy-egy lábán több funkció is megjelenik [7: p.8]. A fejlesztő dolga kiválasztani, hogy adott lábón éppen milyen funkciót engedélyez. A feladathoz a digitális ki- és bemeneti perifériák (PORTx) [7: p.61], valamint a soros kommunikációs (EUSART) [7: p.153] periféria szükséges. Az egyes perifériákat a speciális célú regisztereken (SFR) keresztül lehet engedélyezni, letiltani vagy működési paramétereiket beállítani. A ki- és bemeneti funkciót 8 bites szervezésű, Rx0-7 elnevezésű lábak biztosítanak, ahol x A-tól C-ig terjedhet. Két regiszter tartozik egy-egy porthoz, a TRISx és a PORTx. A TRISx regiszter egyes bitjei határozzák meg, hogy az adott láb kimenet(0) vagy bemenet(1) funkciót tölt be. A PORTx regiszterbe való írással a kimenetek állíthatóak, a PORTx regiszterből olvasással a bementek kérdezhetőek le.

Néhány láb analóg bemenetként is tud funkcionálni, ami a feladathoz nem szükséges, de alapállapotban be van kapcsolva a bemeneti pufferük [7: p.62/4.2.1], ezért le kell tiltani a hozzájuk tartozó ANSEL, ANSELH regiszterekbe való nulla érték írásával. Ha ez a lépés kimarad, a későbbiekben össze fogja zavarni a kommunikációt. A be/kimeneti lábak alapállapota bemenet üzemmódban van. Módosítani kell az összes lábat, hogy kimenetként működjenek, kivéve az RA3 lábat, ahova a nyomógomb van kötve, az RA1 és RA0 lábat, amelyek a későbbi kommunikáció tesztelésénél fognak szerepet kapni és az RB5 lábat, ami a soros port bemeneti (RX) lába lesz. Mindhárom port értéke legyen nulla.

A soros adó/vevő periféria beállításánál meg kell határozni a működési módot (szinkron/aszinkron), a kommunikáció sebességét és paramétereit (bithosszúság, paritás, stop bitek száma). A PC-n használatos soros portok aszinkron üzemmódban dolgoznak, így a vezérlőt is ennek megfelelően kell beállítani. A kommunikáció sebességét nagyban meghatározza a mikrokontroller órajele, mert abból kerül leosztásra az SPBRG és SPBRGH regiszterek használatával. A rendszer órajel 31 kHz és 8 MHz között állítható [7: p.53/3.5], alapértelmezetten 4 MHz. Ezen diszkrét értékek egyike sem osztható el kettő hatványival maradék nélkül úgy, hogy szabványos soros port sebesség értéket kapnánk. Ebből következik, hogy úgy kell órajelet és soros sebességet választani, hogy az időzítésből eredő hiba minél kisebb legyen. Ebben segítenek az adatlapon található táblázatok [7: p.166], amelyekből kiolvasható, hogy 9600 bps sebesség választása esetén 8 MHz és 4 MHz alap órajel mellett is a hiba 0,16%. Nem érdemes ennél kisebb sebességet választani, mert nem érhető el vele hibacsökkenés. Legyen tehát az órajel 8 MHz, és a soros kapcsolat sebessége 9600 bps. A kommunikációs paraméterek kövessék az alapértelmezett 8 bit, nincs paritás, egy stop bit beállítást. Szerencsére ebben az esetben nem kell a regiszter beállításokkal foglalkozni, mert a fejlesztőkörnyezet átvállalja ezt a feladatot. A kommunikáció teszteléséhez fel lehet használni a PICKit2 égető hardvert, mert rendelkezik terminál emulátor üzemmóddal is. Ehhez össze kell kötni az égető 4-es és 5-ös lábát a vezérlő RB7-es és RB5-ös lábával, így két darab vezetéken kívül nem szükséges további hardver csatlakoztatása.



1. ábra. A fejlesztéshez használt hardver konfiguráció

A mikrokontroller rendelkezik egy olyan konfigurációs regiszterrel [7: p.201], amelyet csak a program beégetése alkalmával lehet állítani, és a hardver elindulásának pillanatától meghatározza a működést. Ebben a regiszterben is be kell állítani, vagy le kell tiltani néhány szolgáltatást: a külső reset lábat (MCLR), a watchdog időzítőt (WDT), a brown-out reset funkciót (BOR), az órajel biztonsági funkciót (FCMEN) és a kód védelmet (CP, CPD). Az órajel forrásának (FOSC) a belső oszcillátort (INTOSCIO) kell választani, meghagyva az RA4 és RA5 láb I/O működését.

4. Szoftveres megvalósítás

A mikroC fejlesztői környezetnek két kitüntetett metódusa van, amelyek szükségesek a programhoz. Az első a main() metódus, az itt található programkódot kezdi el futtatni a CPU a bekapcsolás után. A második az interrupt() metódus, ahová egy megszakítás bekövetkezésekor adódik át a futás. Mivel a dobókocka projektben nem lesz szükség megszakítás kezelésre, ez a metódus elhagyható. A fordító nem támogatja az objektumorientált paradigma szerint létrehozott kódot, helyette a strukturált mintát követi. Az első ábrán látható a változók deklarálása, ahol érdemes figyelembe venni a hardver által biztosított csekély méretű adatmemóriát, ezért a tárolandó adatoknak elegendő, de legkevesebb helyet foglaló típust kell használni.

```
1 char randType = 0;
2 unsigned char rxbyte = 0, txbyte = 0, ledek = 0;
3 unsigned int koc1 = 0, koc2 = 0, dob = 0;
4 const long int MM = 2147483647, AA = 48271, QQ = 44488, RR = 3399;
5 long int ss = 12345678; //random seed - kezdőérték
```

2. ábra. A változók deklarációja

A randType változóba kerül a véletlenszám generálást kiválasztó érték, ami nulla vagy egy lehet, attól függően, hogy a fordító által használt eljárás fog futni, vagy a saját, Lehmer-féle generáló algoritmus [9] implementációja. Az rxbyte és txbyte változók a soros kommunikációhoz tartozó egy bájtos pufferek, a ledok változó a panelen található led-ekhez tartozó puffer. Ez utóbbi létrehozását az a tény indokolja, hogy kimenetként konfigurált I/O portról nem lehet olvasni, mert az eredmény megbízhatatlan lesz, azaz egy egyszerű port olvasással nem lehet eldönteni, hogy az adott led világít-e vagy sem. A koc1 és koc2 változókba kerül tárolásra az aktuális dobás értéke, a dob változóban a hátralévő dobások száma kerül. Az MM, AA, QQ és RR változók a Lehmer véletlenszám generátor működéséhez szükséges konstansok, az ss változó a generátor kezdőértékét adja meg (lásd 2. ábra). A mikrokontroller programja [10] három fő részre osztható: a hardver beállítására, a soros kommunikáció kezelésére és a véletlenszámok generálására. A hardver konfigurációs rész a 3a. ábrán látható és igen rövid, köszönhetően a fejlesztőkörnyezetbe épített soros port kezelő könyvtárnak. Csak a digitális és analóg beállításokat kell a felhasználónak elvégeznie, a soros kapcsolat beállítása egy utasításból áll.

<pre> 24 TRISA = 0b00001011; 25 TRISB = 0b00100000; 26 TRISC = 0; 27 PORTC = 0; 28 PORTB = 0; 29 PORTA = 0; 30 ANSEL = 0; 31 ANSELH = 0; 32 Usart_Init(9600); 33 Delay_ms(100); </pre>	<pre> 42 43 44 45 46 47 48 49 50 51 </pre>	<pre> switch (rxbyte) { case 65: dob = 10; randType = 0; break; case 66: dob = 100; randType = 0; break; case 67: dob = 1000; randType = 0; break; case 68: dob = 5000; randType = 0; break; case 69: dob = 10; randType = 1; break; case 70: dob = 100; randType = 1; break; case 71: dob = 1000; randType = 1; break; case 72: dob = 5000; randType = 1; break; } </pre>
--	--	--

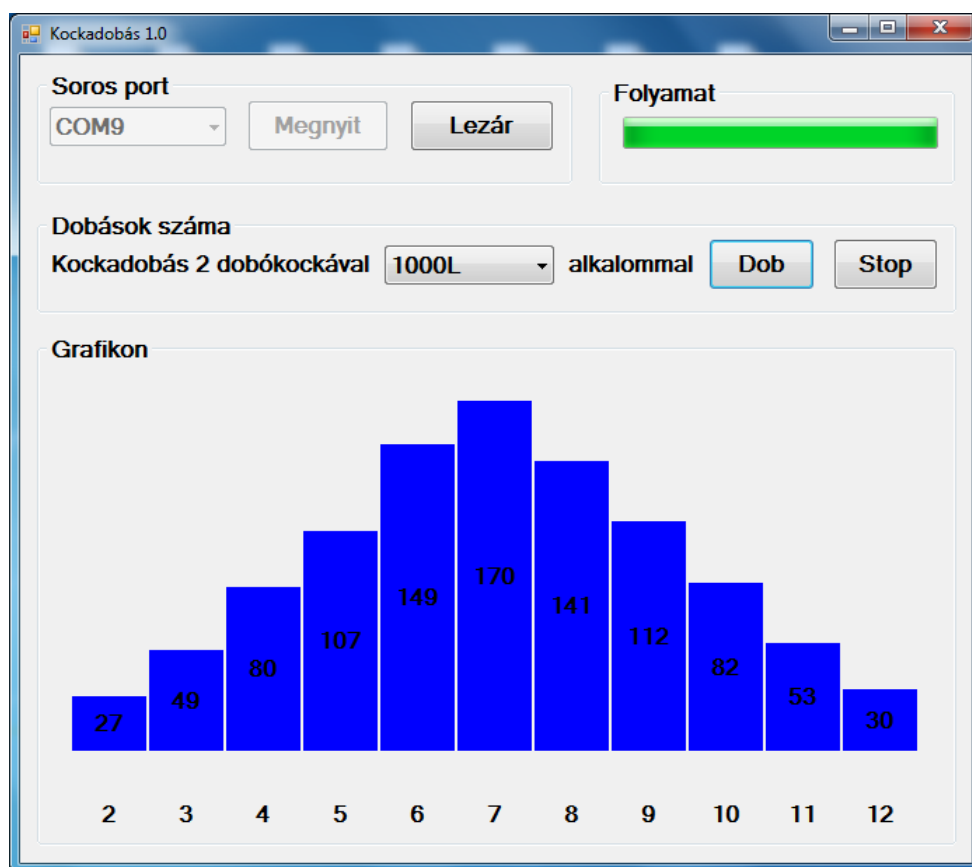
3a. ábra és 3b. ábra. A hardver konfigurálása és a soros parancsok feldolgozása

A beállítás után végtelen ciklus kezdődik, ahová a program többi része kerül, hiszen egy hardvert meghajtó vezérlő program esetében az az elvárás, hogy a bekapcsolástól a kikapcsolásig az eszköz folyamatosan végezze a feladatát. A 3b. ábrán a mikrokontrollerhez érkező soros parancsok feldolgozása látható switch-case elágazással. A pontosabb megértéshez specifikálni kell a soros protokollt. A mikrokontrollerhez érkező parancsokból összesen kilenc van, kétszer négy dobás és egy megállj jelzés. Ezek hibajavítás nélküli átviteléhez elegendő egy bájt. Mivel a működést terminál programból is tudni kell ellenőrizni, ezért célszerű az ASCII tábla szerinti jeleket használni, a nagy A betűtől kezdve. A válasz is hasonló sémát követ: tizenegy szimbólum kell a dobott számok jelzésére (C-M), egy szimbólum a hibás parancsra való reagálásra – ez a kérdőjel lesz – és két szimbólum a válasz vége jelezésére, amelyek a kocs vissza és az újsor karakterek lesznek. A kommunikáció polling eljárással valósul meg. Ha az Usart_Data_Ready() függvény értéke nem nulla, akkor a fogadó pufferben kiolvasásra váró bájt van, amit az Usart_Read() függvény ad vissza. Bájtok küldésére az Usart_Write() metódus szolgál. Kétféle véletlenszám generáló algoritmus is beépítésre kerül a programba. Az első változat a mikroC saját rand() függvénye, ami 0 és 32767 között tud számokat generálni. 32768 lehetőség nem osztható maradék nélkül 6-tal, ezért az alkalmazandó modulo 6 eljárással nem lesznek egyenletes eloszlásúak a dobások. Ezért kerül implementálásra a Lehmer-féle algoritmus is, amely esetében biztosított az egyenletes eloszlás. A program elkészültével, a CTRL+F9 gombok lenyomásával lefordítható a forráskód. A projekt könyvtárban több állomány is keletkezik, ilyen az ASM fájl, amiben

assembly nyelvre lefordítva látható a program és a HEX fájl, amit a PICkit2 égető segítségével a mikrokontrollerbe lehet írni.

5. A Windowson futó vezérlő program

A vezérlőprogram a Microsoft .Net keretrendszer 2.0-ás verziójában, Windows Forms-os alkalmazásként készül el. A felületen jelenjen meg egy kombinált lista, ami a rendszerben szereplő soros portokat tartalmazza. A listából kiválasztott portot a Megnyit gombbal lehessen használatba venni, és ekkor aktiválódjon a többi vezérlő is az oldalon. A Lezár gombbal lehessen megszüntetni a kommunikációt, deaktiválni a vezérlőket és új portot választani. Egy második kombinált listában lehessen kiválasztani a dobások számát és a generáló algoritmust. Ez a lista a 10, 100, 1000 és 5000 lehetőségeket tartalmazza. A lista mellett legyen a Dob gomb, amivel elindítható, illetve a Stop gomb, amivel leállítható a dobások generálása. Az ablak aljára kerüljön a eloszlást és egyben az előfordulást is ábrázoló oszlopdiagram. A grafikon egyes oszlopainak magassága jelentse az adott számhoz tartozó előfordulások számát. Az oszlopok egymáshoz viszonyított magassága egyben az előfordulások közötti arányt is szemlélteti. A grafikon a dobások beérkezése közben folyamatosan aktualizálja magát. A 4. ábrán látható a tervezett grafikus felület megvalósítása. Kiegészítésként egy folyamatjelző is helyet kapott, ami mutatja a beérkező számok mennyiségét a kiválasztott összes dobáshoz képest.



4. ábra. A vezérlőprogram működés közben

6. Tapasztalatok, visszajelzés

A diákműhely foglalkozásaira kezdetben 15 fő jelentkezett, az utolsó alkalomra 5 fő maradt. Az érdeklődők nagyobb része külső oktatási intézményből érkezett, és eddig nem, vagy csak keveset foglalkozott mikrokontrolleres rendszerekkel. A tőlük kapott visszajelzések alapján a jövőben érdemesebb lenne délután 17 órától kezdődő, három órás alkalmakat tartani, így jobban lehetne alkalmazkodni a munkából, vagy nappali oktatási rendszerből érkező hallgatók igényeihez. A mintapéldák programjainak elkészítése közben többször is előfordult, hogy az elkészített forráskód nem, vagy nem a várt eredménnyel futott. A hibák feltárása után jellemzően arra a tapasztalatra lehetett jutni, hogy nagyon oda kell figyelni a mikrokontroller hardveres beállítására, különösen abban az esetben, ha egy lábon többféle funkció is megjelenhet. A hardver közeli programozásnál elengedhetetlenek a pontos hardver ismeretek.

7. Továbbfejlesztési lehetőségek

A diákműhely foglalkozásainak folytatásaként a dobókocka projekt több szempont szerint is kibővíthető. El lehetne hagyni a fejlesztőkörnyezetbe épített soros port kezelő könyvtárat, és saját rutint írni helyette. Ebben a rutinban ki lehetne használni, hogy a mikrovezérlő támogatja a megszakítások generálását a soros átvitel közben, így tehermentesíthető lenne a program fő ciklusa. Fejlesztteni lehetne az átviteli protokollt is, hogy észlelni lehessen az átviteli hibákat, illetve akár hibajavító algoritmust is kaphatna. Szintén a protokoll része lehetne a véletlenszám generáló algoritmusához tartozó random seed érték beállításának lehetősége. Meg lehetne ismerkedni olyan mikrokontrollerrel, ami beépítetten tartalmazza az USB portot, így kikerülhető lenne az RS-232 konverter használata. Egy haladó csoporttal assembly nyelven újra lehetne írni a kockadobó programot, ezáltal maximálisan megismerni a hardver képességeit.

8. Összefoglalás

A cikkben bemutatásra került a Gábor Dénes Tehetségpont gondozásában működő Mikrokontroller programozó Diákműhely, annak célja és az abban folyó munka. A tehetségpont részéről elvárás volt, hogy a diákműhely a rendelkezésére bocsájtott idő alatt kézzel fogható, konkrét végtermékkal álljon elő. Ez a végtermék a kockadobás projekt formájában öltött testet. Részletesen tárgyalásra került a feladat-specifikáció, a mikrokontroller hardveres és szoftveres oldaláról szükséges ismeretek, és elvégzendő feladatok, valamint a PC-n futó vezérlő programmal szemben támasztott követelmények. A feladat megvalósítása közben többször is merültek fel problémák, amelyek vizsgálata és további ismeretanyag elsajátítása után sikeresen megoldásra kerültek, továbbá a hallgatók számára emlékezetes tanulásként szolgáltak. A tehetségpont diákműhelyei minden tanév végén gálát [12] tartanak, ahol az érdeklődők megismerkedhetnek az elkészült alkotásokkal. Ezen a gálán a Mikrokontroller programozó Diákműhely is megjelent egy prezentációval [13]. Véleményem szerint a diákműhelyben elvégzett munka alkalmas arra, hogy a hatodik szemeszteren tanuló, műszaki alkalmazások sávós mérnök informatikus hallgatókat felkészítse a Beágyazott rendszerek tantárgyra, megalapozva egy másfajta, inkább gépközeli gondolkodásra.

9. Köszönetnyilvánítás

Köszönetet mondok Dr. Gambár Katalinnak, a Gábor Dénes Tehetségpont vezetőjének, akinek bátorításával és támogatásával létrejöhett a Mikrokontroller programozó Diákműhely. Köszönetet mondok továbbá Kaczur Sándornak, aki ötletet adott a megvalósítandó projekthez és segítséget nyújtott számomra a cikk megírásához.

Irodalomjegyzék

- [1] Magyar Tehetségsegítő Szervezetek Szövetsége, <http://www.tehetsegpont.hu>, 2014.06.25.
- [2] Gábor Dénes Főiskola honlapja, <http://www.gdf.hu>, 2014.06.25.
- [3] Gábor Dénes Tehetségpont | tehetség.hu, <http://tehetseg.hu/tehetsegpont/tp-150-000-248>, 2014.06.25.
- [4] GDF.hu » Szervezet » GD Tehetségpont » Diákműhelyek, <http://www.gdf.hu/szervezet/gdf-tehetsegpont/bemutatkozunk>, 2014.06.26.
- [5] Gábor Dénes Tehetségpont Ilias lapja, http://ilias.gdf.hu/repository.php?ref_id=26326&cmd=render, 2014.06.26.
- [6] PICkit2 Starter Kit, <http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=DV164120>, 2014.06.26.
- [7] PIC16F690 mikrokontroller adatlapja, <http://www.microchip.com/wwwproducts/Devices.aspx?product=PIC16F690>, 2014.06.26.
- [8] MikroElektronika mikroC for PIC, <http://www.mikroe.com/mikroc/pic/>, 2014.06.26.
- [9] Lehmer véletlen szám generáló algoritmus, http://en.wikipedia.org/wiki/Lehmer_random_number_generator, 2014.06.27.
- [10] Kockadobás PIC mikrokontroller program forráskódja http://ilias.gdf.hu/repository.php?ref_id=50839, 2014.06.27.
- [11] Kockadobás Windowson futtatható vezérlőprogramja http://ilias.gdf.hu/repository.php?ref_id=50840, 2014.06.27.
- [12] Tanév végi gála, http://ilias.gdf.hu/goto.php?target=pg_39822_48985&client_id=ilias-ha, 2014.06. 28.
- [13] Kaczur Sándor – Friedel Attila: A programozáshoz kötődő diákműhelyek a 2013/2014-es tanévben, http://ilias.gdf.hu/ilias.php?ref_id=48985&obj_id=39822&from_page=39822&cmd=downloadFile&cmdClass=ilImpresentationgui&cmdNode=8n&baseClass=ilLMPresentationGUI&file_id=il__file_102253, 2014.06.28.

Csatlakozás az Európai tudásközpontokhoz

Joining to the European Knowledge Centres

Gonda János^a, Horváth Zoltán^b, Pap László^c

^aELTE IK
andog@inf.elte.hu

^bELTE IK
hz@inf.elte.hu

^cBME VIK
pap@hit.bme.hu

Absztrakt: 2008-ban döntött az Európai Unió arról, hogy létrehozza az Európai Innovációs és Technológiai Intézetet (European Institute of Innovation and Technology, EIT), és székhelyét Budapestet jelölte ki. Az intézmény 2009-ben kezdte meg a működését. Az intézet egyik legfontosabb feladata a felsőoktatás, a kutatás és az innováció minél szorosabb együttműködésének kialakítása, a „tudásháromszög” megvalósítása, az ehhez szükséges intézményi keretek létrehozása, az egyetemek, kutatóintézetek és az ipar közös szervezetének létrehozása. Ennek érdekében tudás- és innovációs közösségeket, KIC-eket (Knowledge and Innovation Communities) szervezett három kiemelt, kulcsfontosságúnak ítélt területen: a klímaváltozás (Climate KIC), az energetika (InnoEnergy) és az infokommunikáció (EIT ICT Labs) kutatására és fejlesztésére.

Az EIT ICT Labs úgynevezett csomópontok (node) köré szerveződik. Egy-egy csomópontban rangos egyetem(ek), nemzetközileg is elismert kutatóintézetek és vállalatok működnek együtt. Jelenleg hét csomópont van, és e mellett két társult tagja van a szervezetnek, amelyek egyike Budapest. A budapesti csomópontban két egyetem működik szorosan együtt: a konzorcium vezetője az ELTE, a másik egyetemi partner a BME, míg a konzorcium ipari partnerei a Cisco Systems Magyarország, az Ericsson Magyarország, a Nokia Solutions and Networks, a Magyar Telekom és a General Electric Healthcare. A csomópont működéséhez jelentősen hozzájárul az ELTE-Soft Nonprofit Kft is. A konzorcium több jelentős magyarországi kisvállalkozással is szorosan együttműködik, ilyen cégekkel partnerségi megállapodást kötött.

A magyar állam és a magyar kormány a teljes jogú tagság elérése érdekében 2012-ben pályázatot írt ki, amelyet a fentebb említett két egyetem a Cisco-val kiegészülve megnyert. A pályázat 2013-ban indult, és 2014. júniusában fejeződött be. Ennek eredményeiről, tapasztalatairól szól az előadás.

Kulcsszavak: EIT, ICT Labs, tudásháromszög, budapesti csomópont, pályázat

Abstract: In 2008, the European Union decided to establish the European Institute of Innovation and Technology (EIT) in Budapest. The institution started his function in 2009. One of the most important tasks of the Institute is to base the cooperation between the higher education, research and innovation as tight as possible, to implement of the "knowledge triangle", to create the necessary institutional framework, the common organization of the universities, research institutes and industry. To this end, EIT organized Knowledge and Innovation Communities, abbreviated as KIC-es, in three key areas: for the research and development of the climate change (Climate KIC), the energy (InnoEnergy) and the info communication (EIT ICT Labs).

The EIT ICT Labs is organized as a net of the so-called nodes. In a node, prestigious university (universities), internationally recognized research institute(s) and companies work together. Currently there are seven nodes, and in addition, two associate members. One of these associate members is Budapest. In the Budapest node two universities work closely together: the Consortium is led by the Eötvös Loránd University, the other university-partner is the Budapest University of Technology and Economics, while the industrial partners are the Cisco Systems Hungary, Ericsson Hungary, Nokia Solutions and Networks, the Hungarian Telekom and General Elec-

tric Healthcare. To the activities of our node contributes significantly the ELTE-Soft Non-profit Ltd. The consortium works closely together with several important small and medium sized hungarian companies (SME-s), and signed partnership agreements with them.

The consortium of the abovementioned two universities and the Cisco Systems Hungary won a tender in 2012, granted by the Hungarian State and the Hungarian Government in order to achieve the full membership. The work began in 2013, and was completed in June of 2014. The present paper tries to summarise the results and the experiences of the tender.

Keywords: EIT, ICT Labs, knowledge triangle, Budapest Node, tender

1. Bevezetés: Az EIT és a tudásháromszög

Az Európai Unió a kétezres évek első évtizedének közepén, a megújult lisszaboni stratégia keretében arra a megállapításra jutott, hogy bár a kutatásban igen komoly eredményeket tud felmutatni, ezek megvalósításában, az eredmények felhasználási, alkalmazási szintre való transzformálásában, termékké való konvertálásban jelentős a lemaradása az Egyesült Államokhoz vagy más, ázsiai, a kutatásban-fejlesztésben jelentős eredményeket elérő országokkal, térségekhez viszonyítva (ezt hívják információs paradoxonnak). Az előbbieket röviden összefoglalva azt mondhatjuk, hogy a kétségtelenül jelentős kutatási eredmények mellett nagy volt a lemaradás az innováció, és ezen belül is kiemelten a termékinnováció területén, és ennek következtében a gazdasági növekedésben. A probléma elemzése során azt a megállapítást tették, hogy a versenyképesség fenntartása, sőt, a minél több területen való elsőbbség megszerzése érdekében szükséges egy kifejezetten ezen cél megvalósítását szervező, koordináló intézmény létrehozására. 2008. március 11-én hozta létre az Európai Parlament és a Tanács a 294/2008-as rendeletével az Európai Innovációs és Technológiai Intézetet (European Institute of Innovation and Technology, röviden EIT). Öten pályáztak az új intézmény székhelyeként: Budapest, Wrocław, Jéna, Sant Cugat del Vallès (Katalónia), valamint Bécs és Pozsony között. Közben az Unió megfogalmazta a székhellyel kapcsolatos elvi álláspontját: olyan országban legyen, amely új tagállam, és amelyben még nincs más, uniós intézmény. Ennek a két feltételnek csak Wrocław és Budapest felelt meg. 2008. június 18-án született meg a döntés Budapest javára. Az intézmény 2009-ben kezdte a működését Lágymányoson, az Infoparkban. Az Infopark Kelet-közép Európa első innovációs és technológiai parkja, ahol elsősorban az informatika, a telekommunikáció és a szoftverfejlesztés van jelen. Itt, az Infopark közvetlen szomszédságában található az ELTE Természettudományi valamint Informatikai Kara, a BME Villamosmérnöki és Informatikai Kara, valamint legújabb „lakóként” az MTA Természettudományi Kutatóközpontja (MTA TTK), benne többek között az Anyag- és Környezet-kémiai Intézet (AKI).

Az Európai Unió, az Európai Innovációs és Technológiai Intézet a felmerült problémákat, az innováció felgyorsítását a tudásháromszög megvalósításával akarja elérni. A tudásháromszög három csúcspontjában a kutatás (a tudás létrehozása), az oktatás (a tudás továbbadása) és az innováció (a tudás hasznosítása) található. A háromszög három éle a csúcspontok közötti kölcsönhatást, együttműködést és egymásra utaltságot fejezi ki, azt, hogy a három láncszem közül egyik sem hiányozhat. Az egymásra hatás mindhárom vonatkozásban kölcsönhatás, kétirányú kapcsolat.

A fentebbi célok elérése, megvalósítása céljából az EIT úgynevezett Tudás és Innovációs Közösségeket (Knowledge and Innovation Community, KIC) szervez. Jelenleg három KIC működik:

- az éghajlatváltozással és annak mérséklésével foglalkozó KIC (Climate-KIC);
- a fenntartható energia kérdéseit vizsgáló KIC (KIC InnoEnergy);
- az információs és kommunikációs technológiák KIC-e (EIT ICT Labs).

2. Második fejezet: EIT ICT Labs

Az EIT ICT Labs missziója az, hogy Európát a világ vezető innovátorai közé emelje az információs és kommunikációs technológiák terén (*Information and Communication Technologies – ICT*) az oktatás, kutatás és üzleti szféra szoros integrációjával. Az EIT ICT Labs írja önmagáról¹:

EIT ICT Labs egyike az első Tudás és Innovációs Közösségeknek, KIC-eknek, amelyeket az Európai Innovációs és Technológiai Intézet hozott létre az Európai Unió kezdeményezésére. Az EIT ICT Labs missziója, hogy előmozdítsa Európa vezető szerepét az infokommunikációs technológia területén a gazdasági növekedés és az életminőség érdekében. 2010 óta az EIT ICT Labs következetesen hozza össze a kutatásban, felsőoktatásban és az üzleti világban tevékenykedő embereket. Az oktatás, kutatás és üzlet összekapcsolásával felkészíti az IKT talentumait a jövőre, és életre kelti az IKT-s innovációkat. Az EIT ICT Labs partnerei között az IKT multinacionális vállalatai, vezető kutatóközpontjai és a legmagasabban jegyzett egyetemek vannak jelen.

A fenti idézetben olvashatunk a tudásháromszög szerepéről. Magát a tudásháromszöget mutatja az 1. ábra.

Az EIT ICT Labs célja tehát röviden, hogy olyan szakembereket képezzen, akik a technológiai és üzleti szaktudásukra támaszkodva sikeresen tudják termőre fordítani az európai kutatóintézetekben létrehozott kutatási eredményeket.

Fontos szerepet játszik az elérendő célok megvalósításában az EIT ICT Labs finanszírozási modellje. A KIC az úgynevezett katalizátor-programokat támogatja, azokat a programokat, amelyek jelentős új hozzáadott értékkel rendelkeznek. A katalizátor egy kutatás, egy kísérleti fejlesztés eredményét viszi tovább a piacra jutás feltételeinek megteremtésével. A teljes program költségének maximum 25%-át biztosítja az EIT ICT Labs, és csak a katalizátor részt támogatja. Ezekhez a támogatásokhoz pályázat útján lehet hozzájutni. Ugyanakkor a fennmaradó 75% akár teljes egésze is származhat más pályázatokból. A modell szemmel láthatóan és egyértelműen az információs paradoxon megszüntetését célozza, az innovációt támogatja, a gondolatok termékké válását, eladható, piacra vihető termékké válását támogatja.

Az innovációs szakértők hálózata azzal a feladattal, hogy iránymutatást adjon a várható feladatokról, célokról, kihívásokról és trendekről, amelyek a jövő infokommunikációs világát jellemzik majd.

Az EIT ICT Labs által preferált innovációs területek 2014-ben az alábbiak:

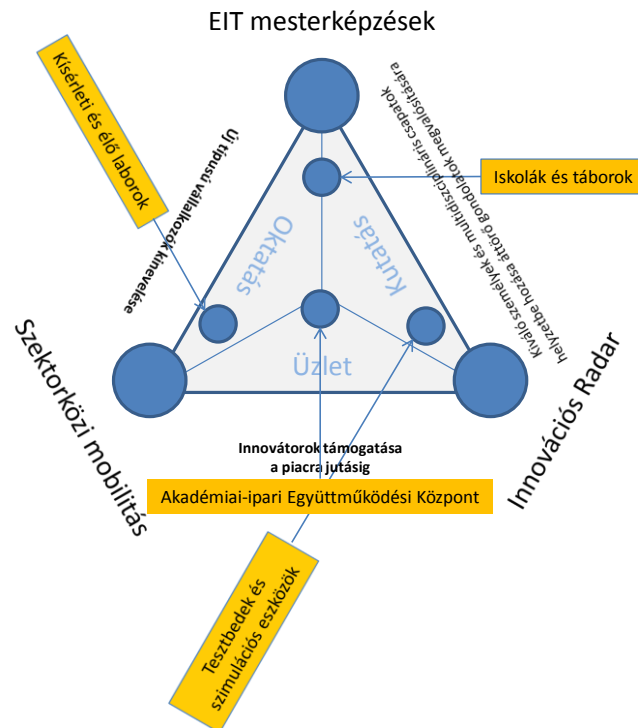
- kiber-fizikai rendszerek;
- jövőbeli felhő;
- a jövőbeli hálózati megoldások;
- egészség és jólét;
- magánszféra, információs biztonság és az információs közösségekben való megbízhatóság;

¹ <http://www.eitictlabs.eu/about-us/>

- „okos” energiarendszerek;
- „okos” terek;
- városi élet és mobilitás.

A fentiekhez kapcsolódnak az alábbi kiemelt területek:

- (egyetemi) mesterképzés;
- doktori iskola.



1. ábra. A tudásháromszög²

Az EIT ICT Labs a fenti feladatok megoldására csomópontokat hozott létre, amelyek feladata az EIT ICT Labs stratégiájának végrehajtása. A csomópontok az oktatás, kutatás és az innováció központjai a legkiválóbb egyetemi és ipari kutatókkal, amelyre alapozva a kiváló regionális csoportok világszínvonalú innovációs központtá alakulhatnak. Jelenleg hét csomópont van: Helsinki, Stockholm, Eindhoven, Párizs Berlin, Trento és London. A csomópontok 3-10 tagból és további, kapcsolódó partnerekből állnak. A hét csomópont mellett két társult tag van: Budapest és Madrid.

A felsorolásból látszik, hogy kelet-közép Európát csupán Budapest képviseli ezen a palettán.

A budapesti csomópontot alkotó konzorcium tagjai:

- ELTE, egyben mint a konzorcium vezetője;
- BME;
- Ericsson Magyarország;
- Magyar Telekom;
- Cisco;
- GE Healthcare.

² A kép a <http://www.eitictlabs.eu/about-us/strategy/the-knowledge-triangle/> honlapról származik magyar fordításban

Budapest aktív szerepet játszik az EIT ICT Labs életében. Három mesterképzésben (Security and privacy, ennek a témának a gondozója, koordinátora, vezetője az EIT ICT Labs egészében, Service design and engineering, Digital Media Technology) és a doktori képzésben érintett, több EIT által kiírt, nemzetközi konzorciumban megvalósuló pályázat részese, és koordinátora, szervezője az úgynevezett outreach programnak. Ennek feladata, hogy az Európai Unió azon 21 országában szervezze, toborozza a tehetséges hallgatókat és támogassa start-up vállalkozások létrehozását, alapítását, beindítását, amelyekben nincs teljes jogú tag.

A csomópontok úgynevezett Akadémiai-ipari együttműködési központokat (Co-location centre, CLC) szerveznek. Ezek fizikai valóságukban létező központok, amelyekben az EIT ICT Labs adott csomópontbeli tevékenységét működtetik, szervezik, a csomópont legtöbb tevékenysége ezekben a központokban történik.

3. Harmadik fejezet: a budapesti csomópont

Budapest az EIT ICT Labs társult tagja, amely sok, de ennek ellenére korlátozott lehetőségeket jelent. Fontos kiemelni, hogy bár nem teljes jogú tagja a szervezetnek, ennek ellenére fontos szerepe van ennek életében, mert, mint azt az előző fejezetben írtuk, két területen is Budapest tölti be a vezető, irányító, koordináló szerepet, az egyik mesterképzési programban, illetve az úgynevezett outreach programban. A feladatok ellátására, koordinálására Budapesten is létrejött az akadémiai-ipari együttműködési központ, a CLC az ELTE Informatikai Karán. A központ az ELTE szenátusának CLXXIX/2013. (V. 27.) Szen. sz. határozatával alakult meg, bár már 2012. szeptember 1-én megkezdte működését. A CLC az ELTE lágymányosi telepén, az egykori ELTE Számítóközpont épületében működik.

Ha megnézzük a teljes jogú tagsággal rendelkező csomópontokat, akkor láthatjuk, hogy mindegyikben rangos egyetem, nemzetközileg is jegyzett kutatóközpontok és igen jelentős, erős, az innovációban komoly eredményekkel büszkélkedhető multinacionális nagyvállalatok vesznek részt. A budapesti csomópontban két, nemzetközileg is ismert és elismert egyetem, valamint négy ipari vállalkozás működik. Alapvető probléma, hogy Magyarországon nincs nemzetközileg is jegyzett, erős innovációs eredményekkel rendelkező nagyvállalat. Ezt ellenőrizendő, a budapesti konzorcium jelentős magyar kis- és középvállalatokkal kötött partneri megállapodást. Ezek a vállalatok az alábbiak:

- AITIA International Zrt.;
- Attrecto Innovations;
- BHE Bonn Hungary Elektronikai Kft.;
- IntelliFactory Kft.;
- BInergy Kft.;
- Netvisor Kft.;
- BalaBit IT Security;
- IND Group;
- Prezi.

A konzorcium partnerségi megállapodást kötött ezen kívül az Informatikai, Távközlési és Elektronikai Vállalkozások Szövetségével (IVSZ) is. A budapesti csomópont tevékenységében részt vesz a Számítástechnikai és Automatizálási Kutató Intézet (Sztaki) is az ELTE IK oda kihelyezett tanszéke által, és a doktori képzésben partner a Nokia Solutions and Networks KFT (NSN) is.

Budapest részt vesz a mesterképzésben, és részese az európai doktori programnak, amelynek fontos sajátossága, hogy maga a szakmai képzés az adott egyetem eddig is létező és akkreditált doktori programjának keretében történik, de kiegészítésként jelentős gazdasági, üzleti, vállalkozói ismereteket nyújtó, szervezett képzésben is részt vesznek a hallgatók, akik így (remélhetőleg) képesek lesznek az elképzeléseiket valós termékekben is megvalósítani, képesek lesznek vállalkozást létrehozni új termékek előállítására, ilyen vállalkozásokat működtetni és vezetni. Az EIT által szervezett mesterképzés két éve két különböző csomópont egyetemén történik. Jelenleg Budapest három mesterképzésben vesz részt, egyet-egyét az ELTE illetve a BME jegyezi, míg a harmadik, a biztonsági problémákkal foglalkozó szakirány a két egyetem közös gondozásában valósul meg. Mindhárom képzésben Budapest a kimenet, a tanulmányok második évét töltik itt a hallgatók, de várhatóan a közeljövőben a belépő hallgatóknak is lehetőségük lesz Budapestet választani. A három mesterkurzus az alábbi:

1. Service Design and Engineering (SDE; ELTE IK);
2. Security and Privacy (S&P; ELTE IK és BME VIK);
3. Digital Media Technology (DMT; BME).

Az S&P teljes ICT Labs-re kiterjedő vezetője Sziklai Péter, az ELTE TTK Számítógéptudományi Tanszékének docense.

A budapesti CLC végzi a teljes outreach program szervezését, koordinálását, irányítását, ennek vezetője Benczúr András, az ELTE TTK korábbi dékánja, jelenleg az egyetem professzor emeritusa.

4. Negyedik fejezet: a pályázat

A fentiek alapján elég világosan látszik, hogy egyrészt nagy elismertséget, elfogadottságot jelent az EIT ICT Labs (és a másik két KIC) csomópontjának lenni, másrészt igen sok előnnyel is jár a „klubtagság”. Ezt ismerte fel a magyar állam és a magyar kormány, és a cél elérésének érdekében, ennek támogatására pályázatot írtak ki „EIT KIC társulásokban történő magyar részvétel és partneri közreműködés támogatása” címmel³.

A pályázatban két pont került megfogalmazásra, közülük az „A” pontra pályáztunk (az alábbi rész a pályázati felhívásban olvasható):

A” alprogram:

· Az EIT KIC-ek partnerségi hierarchiájának csúcsán kiemelkedő tudományos és innovációs kapacitással bíró, önmagukban is több partner együttműködéséből kialakuló csomópontok állnak. Az alprogram célja a KIC keretében hivatalos státusszal bíró azon hazai kutatóhelyek és konzorciumaik támogatása, amelyek potenciálisan legfelső szintű csomóponttá válhatnak az adott KIC keretében. ... A támogatás célja, hogy a potenciális csomópont a lehető legmagasabb számú, a KIC KFI tevékenységéhez kapcsolódó projekt megvalósításában tudjon részt venni, az ehhez szükséges biztos finanszírozási keretet felmutatva, ami előfeltétele a legfelső szintű csomóponti státusz elérésének.

A pályázatra egy háromtagú konzorcium jött létre az ELTE, a BME és a Cisco Magyarország részvételével, és ez a konzorcium el is nyerte a támogatást. A pályázati munka 2013. január 1-én kezdődött és 2014. június 30-án zárult.

³ PÁLYÁZATI FELHÍVÁS a Kutatási és Technológiai Innovációs Alap EIT KIC társulásokban történő magyar részvétel és partneri közreműködés támogatása tárgyú pályázathoz Kódszám: EITKIC_12

A pályázat keretében, a felhívásban olvashatóaknak megfelelően, külön részprogram feladata volt a budapesti csomópont CLC-jének felállítása, működésének beindítása. Ennek keretében egyrészt szervezetenként is megalakult az Akadémiai-ipari Együttműködési Központ mint az ELTE IK szervezeti egysége, megfogalmazódott a CLC víziója és missziója (lásd alább), megszerkesztésre és elfogadásra kerültek a működéshez szükséges jogszabályok, kialakultak a különböző feladat ellátására a csoportok. Mivel a CLC egyik fontos feladata az induló vállalkozások támogatása, a pályázat keretében a szakma jeles képviselői kézikönyvet készítettek a vállalkozások megindításához szükséges legfontosabb tudnivalókról [1], valamint megszerzésre került egy Befektetői Fórum.

A budapesti csomópont az alábbi célokat és feladatokat fogalmazta meg⁴:

Vízió

Egy olyan helyi és regionális meta-ökoszisztéma, ahol az innovatív (innovációs) ötletek kiteljesednek, megvalósulnak, új ökoszisztémák jönnek létre.

A Budapesti Akadémiai - Ipari Együttműködési Központ (Budapest CLC) segítségével és támogatásával olyan folyamatok indulnak el, olyan szervezetek jönnek létre és működnek, valamint olyan környezet alakul ki, amely inspirálja, elősegíti és támogatja az innovációt, az új ötletek megszületését, kialakulását, felhasználását és üzleti megvalósítását az egyetemeken, a kutatóintézetekben, valamint a kis- és középvállalatoktól kezdve a nagyvállalatokig.

Kialakul és megerősödik szoros partnerség és együttműködés, hálózat az egyetemek, kutatóintézetek, kis- és középvállalatok, nagyvállalatok között, továbbá a külföldi partnerek között az oktatás, kutatás és üzleti területen. Aktív együttműködés hazai és nemzetközi kutatási projekteken, közös vállalkozási tevékenységeken. Az együttműködések kiterjednek egész Európára, és a Központ által koordinált partnerség az infokommunikációs területen Kelet-közép Európában vezető szerepet tölt be. Ez a hazai és regionális vezető szerep, olyan központ és célpont, amely jelentős kapcsolati tőkével rendelkezik, és amely képes a régióban összehangolni a kutatás-fejlesztési és innovációs folyamatokat, és azokat további értékeket teremtő üzleti folyamatokba bekapcsolni.

Misszió

Célunk, hogy a Magyarországon, illetve ezen túlmenően Közép- és Kelet-Európában keletkező legújabb kutatás-fejlesztési eredmények sikeresen átkerüljenek az innovációs folyamatba, különös tekintettel az infokommunikációs technológiák területére, és az EIT ICT Labsban betöltött egyedülálló szerepünknek – egyedüli kelet-közép európai tagként – köszönhetően ezen eredmények széles körben elterjedhessenek, illetve a partnerségből származó kapcsolati tőke kiaknázásával üzleti eredmények születhessenek. Ezt a létrejött Budapesti Akadémiai - Ipari Együttműködési Központ (Budapest CLC) az alábbiakkal segíti elő:

- *a kutatás-fejlesztés és az innováció területén biztosítja a kezdeményezésben résztvevő egyetemek, kutatóközpontok és vállalkozások együttműködéséhez szükséges feltételeket;*
- *megvalósítja a szakemberek cseréjét, az oktatás megújítását, a hallgatók kutatás-fejlesztési és innovációs folyamatokba történő bevonását;*

⁴ Vízió és misszió (a CLC szabályzatai és dokumentumai között)

- *serkenti az akadémia és az ipar, a hazai kis- és középvállalati szektor közötti, illetve az EIT ICT Labs partnerei közti nemzetközi együttműködést és mobilitást;*
- *közös EIT és Európai Unió projektjein keresztül lehetőséget teremt új partnerségek kialakítására, az eddigi együttműködések elmélyítésére, a kutatási eredmények gyakorlatba való átültetésére.*

Konkrét célok:

- *hazai és regionális vezető szerep az infokommunikációs technológiák területén a gazdasági fejlődés és ez életminőség javítása céljából;*
- *az innovációs szellemiség és potenciál lehető legjobb honi, regionális és európai kihasználása;*
- *innovációs képességek megerősítése, a jövő vállalkozóinak kinevelése, felkészítése és elősegítése innovációs áttörések eléréséhez;*
- *az innovatív vállalkozói kedv és a versenyképesség javulásának, növekedésének elősegítése;*
- *innovációs és vállalkozási ökoszisztémák kialakítása;*
- *Kelet-közép európai informatikai területen működő oktató, kutató, vállalkozó intézmények hálózatának kiépítése.*

Stratégiai tevékenységek:

- *az oktatás, a kutatás és az üzleti világ összekapcsolása, a tudásháromszög megvalósítása;*
- *a tudásháromszög résztvevőinek összefogása, munkájának összehangolása, és számukra hozzáadott érték termelése;*
- *üzleti szemlélet által vezérelt, működőképes területi (regionális) oktatási, kutatási és innovációs központ megvalósítása;*
- *innovációs potenciál (képességek) felismerése, felfedezése, felkarolása, elismerése, fejlesztése, támogatása, segítése, elősegítése;*
- *az ötlettől a termékig, a kutatóintézetektől a piacig, valamint a hallgatóktól az üzletemberekig tartó útvonalak kialakítása, megerősítése és megkönnyítése, lerövidítése;*
- *hálózatépítés, partnerek összegyűjtése.*

A megvalósításba bevont partnerek:

- *partnerek bevonása és összekapcsolása a tudásháromszög mindhárom oldaláról, azaz az akadémiai (egyetemi), a kutatási-kutatóintézeti, valamint az ipari és üzleti oldalról;*
- *a partneri együttműködés és mobilitás biztosítása a hazai, a közép- és kelet-európai régióban, valamint az EIT ICT Labs és partnerei között;*
- *új, innovatív, innovációs tevékenységüket erősíteni kívánó partnerek hazai és regionális felkutatása és bekapcsolása az innovációs folyamatokba.*

A pályázat nagyobb része kutatási-fejlesztési feladatokról szólt, és négy nagy témát ölelt fel, mindegyik témában külön-külön az ELTE és a BME részéről, illetve az egyik témában kiegészítve a Cisco saját kutatásával. Az egyes témakörök az alábbiak voltak:

1. Komplex infokommunikációs hálózatok megoldásai (ELTE, BME);
2. A jövő digitális közösségi terei és mobilitás (ELTE, BME);
3. Humán-számítógép együttműködés (ELTE, BME, Cisco);
4. Kritikus rendszerek biztonsága.

A pályázat első hat hónapjában alapkutatás illetve alkalmazott kutatás, míg a további tizenkét hónapban, egy kivételével, mindenütt kísérleti fejlesztés történt, ráépülve az első szakaszban elért kutatási eredményekre. A kutatott és megvalósított témák között volt többek között:

- új szemléletű, a kiber-tér és a fizikai eszközök kapcsolatának leírására alkalmas programozási nyelv kifejlesztése;
- nagy, peer to peer felépítésű hálózatok adatforgalmának optimalizálása;
- országhatárokon átnyúló mérőhálózatokhoz kapcsolódó adatbázisok keretrendszerének kidolgozása és megvalósítása, metrikák vizsgálata és felépítése;
- közlekedés különböző aspektusainak optimalizálása;
- aktív eszközzel nem rendelkező objektumok helymeghatározása;
- igen nagy adathalmazban való keresés módszereinek vizsgálata, ajánló rendszerek kidolgozása;
- sérült emberek kommunikációjának támogatása;
- szövegfelolvasó rendszerek vizsgálata és fejlesztése;
- infokommunikációs eszközök munkahelyi tanulásban betöltött szerepének vizsgálata;
- kiemelten fontos infrastruktúrák információs rendszerének védelme, valamint az egyénekhez köthető szenzorok adatainak a személyes adatok védelméhez fűződő jogokkal kapcsolatos védelme.

Láthatóan igen széles területet ölelt fel a pályázat. A kutatás és fejlesztés eredményeként számos cikk, konferencia-előadás, prezentáció, poszter és tanulmány született. Jónéhány szakdolgozat és diplomamunka készült vagy teljesen, vagy legalább részben a pályázati munka eredményeként, és sikeresen megvédett doktori disszertáció is kapcsolódott a pályázati munkához. A pályázati kutatások és fejlesztések részben bekerültek az oktatásba, részben új témákat generáltak, amelyek új kurzusokban jelennek meg. A pályázat további fontos eredményeként lehet megemlíteni, hogy több témában gyümölcsöző, a jövőben is hasznos, mindkét fél számára előnyös kapcsolat jött létre ELTE-s és BME-s csoportok között.

A pályázat indikátorai között szerepelt egy szabadalom benyújtása és egy spin-off cég alapítása. Egy további indikátor a jövő szempontjából is meghatározó mindkét egyetem számára. A pályázatban a két egyetem vállalta, hogy a korábban meglévő három, nemzetközi konzorciumban elnyert pályázat mellé további legalább tíz sikeres, neves külföldi partnerekkel közösen benyújtott pályázatban vesznek részt. Ezt a feltételt is sikeresen teljesítettük – valójában túlteljesítettük – , ami az egyetemek külföldi elismertségének egy fontos kritériuma, és másrészről az egyetemek oktatói és kutatói számára magasabb szintű kihívások is. Ezek olyan szempontok, amelyek a pályázattól függetlenül is fontosak az egyetemek számára.

Oktatók, kutatók, doktoranduszok és hallgatók egyaránt dolgoztak, kutattak, fejlesztettek a pályázatban. A kutatási és fejlesztési témák egy része beépült az MSc-képzésbe, többek között projektmunka keretében. A doktoranduszok és a hallgatók igen jelentős mértékben vettek részt a pályázati munkában, ami minden bizonnyal nagyban járul hozzá a szakmai fejlődésükhöz. De emberi kapcsolatok terén is sokat tanultak és fejlődtek, hiszen ezen időszakban csapatmunkában vettek részt, így meg kellett azt is tanulniuk, hogy egyénenként hogyan kell egy csapat tagjaként az együttműködésben részt venni, a közös eredmény elérése érdekében hogyan kell másokkal együtt-tevékenykedni. És minden ilyen pályázat fontos és hasznos az oktatóknak és kutatóknak, mert az ilyen munka során sok tehetséges és szorgalmas diákot ismernek meg, akiknek egy jó része később, a tanulmányok bevégeztével, az egyetemen folytatja tevékenységét.

5. Köszönetnyilvánítás

Ez a cikk az EITKIC_12 kódszámú, a Kutatási és Technológiai Innovációs Alapból finanszírozott, Az EIT KIC társulásokban történő magyar részvétel és partneri közreműködés támogatása tárgyú, EITKIC_12-1-2012-0001 azonosítójú pályázatról szól, így mindenekelőtt ennek a pályázatnak köszönhető a cikk megszületése. És a cikk nem jöhetett volna létre azon oktatók, kutatók, doktoranduszok és hallgatók, valamint a konzorciumi partnerek sok más dolgozójának munkája és segítsége nélkül, akik valamilyen formában részt vettek a pályázati munkában. A cikkel az ő áldozatos munkájukról adtunk számot.

Irodalomjegyzék

- [1] András Gábor, Miklós Gábor és Péter Racskó: Kézikönyv IKT startup cégek számára, *Teaching Mathematics and Computer Science* **11** (2013), 21–40.

A jövő szelleme. Bevezetés a hálózattudományba.

Dr. Prof. Gyarmati G. Péter

Absztrakt: Az internet szelleme ébresztő az emberiség számára. Telhard de Chardin egy jezsuita pap, 1938-ban állította fel tézisét, amely a Földet körülvevő „gondolatok szférája” létezéséről szól, még el is nevezte: nooszféra, tulajdonképpen kiegészítve a geoszféra, bioszféra elméletünket az agyunk működésének velük azonos szintre emelésével. Az írás csak 1959-ben jelent meg The Phenomenon of Man címmel. Mint minden új, ez is számos ellenkezésre talált, mind vallási, mind akadémiai körökben. A nooszféra ma már közismert és az eredeti leírás szerint „a tudatosság élő anyaga, amely átjárja a Földet és idővel egyre sűrűbbé válik”. Gondoljuk csak meg: a világháló éppen egy – talán kezdetleges – realizációja a nooszférának.

Kulcsszavak: hálózattudomány, második kulcsszó, harmadik kulcsszó

1. Bevezetés

Az internet szelleme ébresztő az emberiség számára.

Telhard de Chardin egy jezsuita pap, 1938-ban állította fel tézisét, amely a Földet körülvevő „gondolatok szférája” létezéséről szól, még el is nevezte: nooszféra, tulajdonképpen kiegészítve a geoszféra, bioszféra elméletünket az agyunk működésének velük azonos szintre emelésével. Az írás csak 1959-ben jelent meg The Phenomenon of Man címmel. Mint minden új, ez is számos ellenkezésre talált, mind vallási, mind akadémiai körökben. A nooszféra ma már közismert és az eredeti leírás szerint „a tudatosság élő anyaga, amely átjárja a Földet és idővel egyre sűrűbbé válik”. Gondoljuk csak meg: a világháló éppen egy – talán kezdetleges – realizációja a nooszférának.

2. Magyarázat

A mostani előadásomhoz magyarázatul az alábbiakat említem:

1. Jelentős előrehaladást értünk el a mérnöki tudományokban, hatalmas alkotásokkal vagyunk körülvéve, amelyek életünket szolgálják, de a lényegyet tekintve csak bonyolultabbá tette életünket és tömegek érzik azt, hogy egyre feleslegesebbé válnak.
2. Fontos eredményeket tudhatunk a biotechnológia, a nanotechnika és más tudományok területén, amelyek egyre távolabb visznek a természetes léttől.
3. Kihasználjuk a földet, vizeket egészen addig a pontig, hogy a biológiai élet már nem lesz fenntartható többé bolygónkon.

Az Internet a milliárdnyi elme összekapcsolásával lehetőséget nyújt a technikai zsákutcából való kijutásra, amennyiben odafigyelünk a szellemére. Ilyen spekulációk a Virtuális Valóság, a Virtuális Világ benépesülése, a Cyberspace közismerten.

Mások az Információs társadalom képét vetítik elénk, amelyben a javakat egy fenntartható fejlődés elméletével, döntően robotokkal állítják elő, míg az ember - mint valamiféle nagy adatfeldolgozó - irányítja az egész folyamatot, vagyis az életet.

Ezek a sommás jövőképek nem képzeletszülemények, sokkal inkább valóság képek, amelyek mindenképpen indokolják a rendszer lényegének megértését, szerkezetének megismerését. Az Internet egy eddig soha nem látott gigantikus hálózat, amely lehetőséget ad a kapcsolatok

tanulmányozására, a tudományos módszereink, az elmélet mellett a kísérletek is, egészen összemérhető méretűekig lehetővé váltak. Ez az új tudományterület a hálózattudomány. A számítástechnika története, a számítógépek fejlődése, a matematikai módszerek kiteljesedése megnyitotta az utat az emberi kapcsolatok kutatását egészen új utakon.

A hálózat nem informatika, a hálózat alkalmazza az informatikát, a hálózat nem írható le pusztán informatikai módszerekkel, holott elemei annak alkotásai, a hálózat ennél magasabb szintű: az emberi kapcsolatok szerkezete. Ezért másik tudományterület.

3. Megjegyzések

Alapvető fontosságúnak tartom tehát, a szokásos informatikai oktatáson túlmenően hálózattudományi tanulmányok bevezetését a felsőbb oktatás körébe. A téma valamennyire interdiszciplináris jellegű, mivel minden tudományterület emberi kapcsolatok mibenlétén alapul és eredmények ma már csak közös munkával érhetők el. Ezek szervezése, működtetése, az adott kapcsolatok maga a hálózat, amelynek ismeretéről van szó. Több éves külföldi tapasztalatom alapján javaslom az alábbi Bevezetés a hálózat tudományba kurzus létesítését.

Meg kell jegyezni, hogy a területnek számos nemzetközi élvonalbeli magyar művelője van itthon és a világban, mint például Barabási Albert László, vagy Albert Réka, aztán itthon Csermely Péter, Vicsek Tamás, Kertész János, Lovász László, Bollobás Béla a teljesség igénye nélkül. Természetesen a „régiek: Erdős Pál, Rényi Alfréd, Neumann János és ne feledkezzünk meg Karinthy Frigyesről, aki 1929-ben (!) a Láncszemek novellájában az emberiség történelmében először írta le a kisvilág tulajdonságát, amelyet évekkel később bizonyítottak be.

4. Javaslat multidiszciplináris egyetemi kurzusra

Az alábbiakban olvasható a tervezett kurzus Syllabusa:

Javaslat multidiszciplináris kurzusra a ... Egyetem ... Informatikai Karán.

Bevezetés a hálózat tudományba.

Kurzuskód:

Felelős kapcsolat:

Előadók

-

Időtartam, időpontok

- 18 hét, egy félév: heti 2 óra előadás, további heti 1 óra gyakorlat, vagy házi munka:
- Konzultáció minden héten 1 óra:

Minősítés

- házi feladat: 30%
- évközi dolgozat: 30%
- vizsgafeladat: 40%
- pontszám: ?

Ajánlás

A kurzust ajánljuk az informatikai, számítástudományi, ismeretelméleti, statisztikai, pszichológiai, biológiai, kommunikációs, fizikai tanulmányokban résztvevőknek.

Résztvételi feltétel

- számítógép használat
- angol nyelv tudás, legalább olvasás/megértés szinten
- programozási alapismeretek bármilyen programnyelvből
- matematika: valószínűség, statisztika, komplex (?)
- általános ismeretek az érintett szakterületeken
- vizsga, látogatási bizonyítvány: kurzusokból

Feltétel az alábbi kurzusokhoz

- ?

Tartalom, cél

Hálózatok veszik körül az életünket szinte minden területen. Barátok hálózata, kommunikáció, számítógéphálózat, úthálózat, szállítás csak néhány példa, amelyeket mindennap tapasztalhatunk. Mindezeket az agyunk celláival, az idegsejtekkel és fehérjék bonyolult láncolataival érzékeljük, amelyek alkotják és meghatározzák intelligenciánkat, végsősoron életünket.

A hálózat egy általános, mégis alkalmas eszköz kapcsolatok leírására és tanulmányozására az egyszerűtől a tetszőlegesen bonyolultakig.

A kurzus célja bemutatni ennek a most formálódó tudománynak az alapjait, megismertetni a résztvevőket az eredetével, a tudományok szerteágazó területein való alkalmazhatóságával és bemutatni hálózat analitikai módszereket, hogy megérthessük az életünket alkotó bonyolult kapcsolatokat és viszonyokat.

A kurzus gyakorlati részében ismertetésre és kipróbálásra kerülhetnek számítógépes eljárások és programok, amelyek segítik az analízist.

A gyakorlatok analízise az egyszerű hálózatmodelltől fokozatosan egyre bonyolultabb társadalmi-, infrastrukturális-, műszaki-, informatikai-, vagy éppen fizikai-, biológiai modellekkel foglalkozik.

Tematika, terv

1. Áttekintés
2. Gráfelmélet, struktúrák, algoritmusok
3. Folyamathálók, elektromos áramkörök
4. Társadalmi hálózatok, szomszédosságok, mértékek
5. Szemantikai hálók
6. Média kommunikációs hálózatok
7. Üzleti hálók, gazdasági modellek
8. Számítógép hálózatok: diagrammok, felépítés, internet
9. Ideghálók, mesterséges modellek
10. Dinamikus analízis
11. Perceptron
12. Skála-mentes hálózatok
13. Hatvány-függvény eloszlások
14. Csoportosulások, cluster diagram és coefficient
15. Pareto-törvényesség, Pólya-eloszlás
16. Fokszám eloszlás
17. Fontossági algoritmusok
18. Összefoglaló, következmények: monopóliumok, gazdag-szegény, kisvilág, leggyengébb láncszem

Kötelező irodalom, jegyzet

- P. G. Gyarmati: Some words about networks
- Barabási A. L.: Behálózva

Ajánlott irodalom

- Gyarmati Péter: Informatikai elemek
- Karinthy F.: Minden másképpen van, Láncszemek
- Magyar Tudomány: 2006-11. száma

Irodalomjegyzék

- [1] L. Hagerty: The Spirit of the Internet, Kindle Edition, http://www.amazon.com/Spirit-Internet-ebook/dp/B005UQHXM2/ref=sr_1_1?
- [2] P. G. Gyarmati: Computer Science course at UC, University of Cambridge, Centre of Mathematical Sciences, Millenium Mathematical Project
- [3] J. A. Bondy & U. S. R. Murty: Graph Theory with Applications, North Holland, ISBN 0-444-19451-7
- [4] M. Bohus : Számítógép-hálózatok felépítése, oktatási segédlet, Szeged, 1995
- [5] A. I. Galushkin: Neural Networks Theory, Springer, ISBN 978-3-540-48124-9
- [6] P. G. Gyarmati: Next Generation Science Standards (NGSS), <http://www.corestandards.org>
- [7] S. Hewson: The Mathematical Problems Faced by STEM Students, <http://nrich.maths.org/6458/index>
- [8] Bureau of Educational and Cultural Affairs: Special Announcement for P. G. Gyarmati: TEACH THE WEB, ECA-11p/EX, SA-5Floor 42200 C Street NW Washington, DC, 20522, UNITED STATES
- [9] Dr. Prof. P. G. Gyarmati: Javaslat egy neurális hálózati modellre. Megjelenés előtt. <http://szentendreszalon.fw.hu/13-2014-I/140417/sz/neuron-1.pdf>
<http://szentendreszalon.fw.hu/13-2014-I/140417/sz/neuron-2.pdf>
- [10] A. L. Barabási: *How Everything is Connected to Everything Else*, 2004, ISBN 0-452-28439-2
- [11] P. G. Gyarmati: Hálózatok, emberi kapcsolatok, előadás-sorozat, <http://www.gyarmati.dr.hu/lectures/network2.pdf> <http://szentendreszalon.fw.hu/8-2010-II/101111/sz/network2.pdf>
- [12] P. G. Gyarmati: MyCCNet, Network Courses, https://ccnet.stanford.edu/cgi-bin/home.cgi?action=course_search&form_submitted=1&cc=&dir_selected=1&dir_selected_win12=1&dir_selected_spr12=1&dir_selected_sum12=1&dir_selected_fall12=1&dir_selected_win13=1&dir_selected_spr13=1&dir_selected_sum13=1&dir_selected_fall13=1&dir_selected_win14=1&keyword=network&search_code=1&search_name=1&keyword_search=Search+by+Keyword, Stanford University

Programozási gyakorlófeladatok és a tömegbe kiszervezés

Programming exercises and crowdsourcing

Kollár Lajos

Debreceni Egyetem, Informatikai Kar

kollarl@inf.unideb.hu

Absztrakt: A cikkben bemutatom, hogy milyen út vezetett egy, a programozási gyakorlófeladatok tömegbe kiszervezett értékelését segítő webalkalmazás kifejlesztéséhez. Gyakori visszajelzés a programozást épp csak tanulni kezdő hallgatók részéről, hogy nem találnak a szintjüknek megfelelő gyakorlófeladatokat, vagy, ha mégis, akkor sokszor nem tudják eldönteni, hogy az általuk adott megoldás helyes-e, illetve ha elakadnak, nem tudnak segítséget kapni. Ennek egyrészt az egészségesnek sokszor messze nem nevezhető hallgató–oktató arány az oka, hiszen egyszerűen nem áll rendelkezésre annyi oktatói erőforrás, hogy ezek a – vélt vagy valós – problémák megoldódjanak. Ennek a problémának a megoldására került kifejlesztésre egy olyan webalkalmazás, amelynek segítségével a tömeg (elsősorban a hallgatótársak, akár felsőbbévesek) visszajelzései alapján kaphat értékelést illetve egyéb hasznos visszajelzéseket a felhasználó, így tehermentesítve az oktatókat, akiknek így elsősorban majd csak azokra az esetekre kell külön odafigyelniük, ahol a rendszerben beérkezett visszajelzések alapján nem egyértelmű, hogy egy megoldás helyes-e avagy sem. Az alkalmazás tervezése és kifejlesztése során felmerülő kérdésekre, problémákra is kitérve mutatom be, hogy hogyan segítheti a tömegbe kiszervezés a gyakorlásra szánt feladatok értékelését, illetve felvázolom a jövőbeli terveket és elképzeléseket is a továbbfejlesztés lehetséges irányairól.

Kulcsszavak: programozás, gyakorlás, feladatmegoldás, értékelés, tömegbe kiszervezés

Abstract: In this paper I will show the background of the development of a web application that allows the crowd-based evaluation of programming exercises. Students who just start to learn programming often whine that they cannot find exercises of their own level of expertise, they are not able to decide whether the solution they gave is correct or not or they cannot find any help if they get stuck. Root causes of these problems include the bad student–teacher rate so the department does not have enough resources (mainly, instructors) for solving such problems. As a supporting tool, a new web application has been developed which can be used by the crowd (especially other students) to provide feedback to the user. It can ease the work of instructors since they need to focus only on problematic cases where it is hard to decide (based on the feedbacks) whether a given solution is correct or not. I will demonstrate how crowdsourcing can help the evaluation of exercises through some issues that have been raised during the design and implementation of the system. Possible future development directions will also be discussed.

Keywords: programming, practicing, problem solving, evaluation, crowdsourcing

1. Bevezetés

A Debreceni Egyetem Informatikai Karának számos hallgatójától hallottam már különféle programozással kapcsolatos kurzusok (Magas szintű programozási nyelvek 1-2, Programozási technológiák) gyakorlatvezetőjeként, hogy a problémát okoz nekik az, hogy megfelelő programozási gyakorlófeladatokat találjanak otthoni gyakorlás céljából, illetve, hogy ha találnak is ilyeneket, sokszor nehezen tudják eldönteni, hogy az általuk készített megoldás megfelelő-e. Ezek a kérdések annál gyakrabban fordulnak természetesen elő, minél inkább kezdő az adott hallgató, hiszen a felsőbb éves hallgatók idővel különféle tesztelési eszközöket

és tesztesettervezési módszereket is tanulnak, amelyek segítségével remélhetőleg ezen problémák egy jó részén át tudnak lendülni.

Nyilvánvalóan megoldaná a problémát, ha a gyakorlatvezetők és/vagy demonstrátorok képesek lennének megfelelő mennyiségű gyakorlófeladatot előállítani, illetve volna arra lehetőség, hogy azokat ellenőrizzék is. Erre azonban sajnos olyankor, amikor egy-egy gyakorlatvezetőnek 3-5 gyakorlati csoportja (45-90 fővel) is van, sajnos nem nagyon jut erőforrás. Sokszor épp elég nehézséget jelent a számonkérések feladatsorainak összeállítása és értékelése, különösen, hogy nem ritkán 4-10 különféle, de mégis lehetőleg azonos nehézségi szintű feladatsor összeállítására van szükség, mivel ennyi különféle időpontban vannak gyakorlati zárhelyik. Épp ezért a gyakorló feladatok összeállítására és különösen megoldásaiknak igény szerinti ellenőrzésére sajnos alig-alig van mód.

A hallgatókkal erről a témáról többször beszélgetve fogant meg bennem az az ötlet, hogy jó lenne kialakítani egy olyan webalkalmazást, amelyen keresztül hallgatók feltölthetnék feladatmegoldásaikat, amelyeket bárki értékelhet, akinek elegendő ideje, kedve és képessége van. A feltöltött feladatok értékelését így nem (feltétlenül) a gyakorlatvezető végzi, hanem az önkéntes alapon szerveződő tömeg. Ez gyakorlatilag a tömegbe kiszervezésnek, vagyis a crowdsourcingnak eg megjelenési formája.

2. Felhasználók és funkciók

A rendszer tervezése során négy felhasználói szerepkör került meghatározásra. Regisztrációt és bejelentkezést követően szerepkörtől függetlenül mindenkinek lehetősége van a feltöltött feladatok böngészésére, a közöttük történő keresésre, valamint minden regisztrált felhasználó hozzászólást is fűzhet az egyes feladatokhoz. Természetesen a felhasználói adatok és beállítások megváltoztatását is mindenki elvégezheti. A kialakított négy szerepkör az alábbi:

1. **Feltöltő:** bárki, aki a tömeg segítségét szeretné kérni egy feltöltött feladat értékeléséhez. A fenti funkciók mellett saját (értékelendő) feladatmegoldás feltöltését végezhetik (1. ábra), illetve megtekinthetik (nyomon követhetik) saját feltöltéseiket (2. ábra).

Bejelentkezve, mint
Kollár Lajos
Feltöltő
Kilépés

Feladat feltöltése Összes feladat Feladataim Személyes adatok

Feladat feltöltése

Afeladat címe:

Afeladat leírása:

A Kemighan-Ritchie könyv klasszikus példája.

Milyen formában szeretnéd feltölteni a megoldást? Szöveg

Fájl: Tallózás... Nincs kijelölve fájl.

Programkód:

#include <stdio.h>

int main() {
 printf("Hello, World!\n");
 return 0;
}

Programozási nyelv: C

Címkék:

1. ábra. Feladat feltöltése szövegmezőbe írt közvetlen inputtal.

Bejelentkezve, mint
Kollár Lajos
Feltöltő
Kilépés

Feladat feltöltése Összes feladat **Feladataim** Személyes adatok

Feltöltött feladataim

Cím	Feltöltés ideje	Nyelv	Értékelések száma	Pozitív értékelések
Hello	2014-07-08 16:51	Java	1	100%
Hello	2014-07-23 03:46	C	0	0%

2. ábra. Saját feltöltések megtekintése.

2. **Értékelő:** az ő feladata a Feltöltőknek történő segítségnyújtás. Az Értékelők értékeléseikkel és megjegyzéseikkel mondanak szakmai véleményt a feltöltött megoldásokkal kapcsolatban. A minden szerepkör számára közös funkciókon túlmenően megoldások értékelését (3. ábra), az értékelendő illetve már értékelt megoldások megtekintését (4. ábra), valamint a mások által írt értékelések véleményezését végezheti.

Összes feladat **Értékelt feladatok** Értékelői rangsor Személyes adatok Nyelvek Címkék
 Felhasználók kezelése

Bejelentkezve, mint
Kollár Lajos
 Oktató Admin
 Kilépés

A feladat részletei

Hello

A Kernighan-Ritchie könyv klasszikus példája.

Feltöltő: Felhasználó#31
 A megoldás nyelve: C

```
#include <stdio.h>

int main() {
    printf("Hello, World!\n");
    return 0;
}
```

- Megoldás értékelése

Szerinted helyes a megoldás?

Indoklás:

Értékelem

+ Hozzászólás írása

3. ábra. Megoldás értékelése.

Összes feladat **Értékelt feladatok** Értékelői rangsor Személyes adatok Nyelvek Címkék
 Felhasználók kezelése

Bejelentkezve, mint
Kollár Lajos
 Oktató Admin
 Kilépés

Általam értékelt feladatok

+ Szűrés

Cím	Feltöltés ideje	Nyelv	Értékelések száma	Pozitív értékelések	Műveletek
Hello	2014-07-08 16:51	Java	1	100%	Törés

4. ábra. A már értékelt megoldások listája.

3. **Oktató:** az Informatikai Kar valamely oktatója. Az Oktató minden esetben Értékelő is egyben; e szerepkör létjogosultságát elsősorban az adja, hogy legyenek a rendszerben olyan szereplők, akik bizonyos fokú megbízhatóságot adnak a rendszernek, hiszen egy Feltöltő egy Oktató véleményére talán jobban ad, mint egy Értékelő szerepkörű – de akár névtelenséget választó – hallgatótársáéra. (Holott természetesen akár egy Oktató értékelése is lehet hibás.) Az Értékelő funkcióin kívül az Oktató megtekintheti az értékelői

rangsort, ahol a névtelenséget választó Értékelők kódoltan, míg a többiek névvel együtt láthatóak számára.

4. **Adminisztrátor:** a felhasználók kezelésével (5. ábra), az Oktatók regisztrációs kérelmeinek elbírálásával (hiszen nem szabad engedni, hogy bárki Oktató szerepkört kapjon), valamint a feltöltések és/vagy hozzászólások törlésével foglalkozó felhasználó.

[Összes feladat](#)
[Értékelt feladatok](#)
[Értékelői rangsor](#)
[Személyes adatok](#)
[Nyelvek](#)
[Címkék](#)

Bejelentkezve, mint
Kollár Lajos
Oktató Admin

Kilépés

Felhasználók kezelése

Inaktív felhasználók

Név	E-mail cím	Jogosultságok	Műveletek
No records found.			

Aktív felhasználók

Név	E-mail cím	Jogosultságok	Műveletek
Kollár Lajos	kollarl@inf.unideb.hu	Oktató Admin	Inaktíválás Szerepkörök kezelése
Kollár Lajos	kollarl@unideb.hu	Feltöltő	Inaktíválás Szerepkörök kezelése

5. ábra. Felhasználók kezelése.

Egy-egy felhasználó egyidejűleg több szerepkörrel is rendelkezhet, ahogyan az az 5. ábra Jogosultságok oszlopában is látható.

Az elkészült rendszer a szociális tanulókörnyezetek [5]-ben leírt számos elemét alkalmazza. A felhasználók a feltöltött programokat különféle címkékkel annotálhatják, amelyek segítenek a tartalom alapú visszakeresésben. Rendszerünk [5]-tel összhangban nagy jelentőséget tulajdonít az értékeléseknek, ugyanis, ha nem áll rendelkezésre megfelelő számú értékelés, az a felhasználók számára problémás lehet, hiszen nem kapnak megfelelő visszajelzést arról, hogy megfelelő megoldást adtak-e vagy sem. Ezt a fajta bizalmat a rendszerben kétféleképpen érzük el: egyrészt az Oktató szerepkör jelenléte szolgál erre, másrészt pedig az értékelés nélküli feladatok egy idő után automatikusan hozzárendelődnek bizonyos értékelőkhöz (akik természetesen nem kötelesek elvégezni az értékelést), ezzel katalizálva az értékelések darabszámát. Ehhez kapcsolódóan szintén fontos, hogy megtaláljuk azokat a megfelelő értékelőket, akikhez egy adott feladat hozzárendelhető, és bírnak a megfelelő szaktudással annak eldöntésére, hogy a megoldás jó-e, avagy sem.

3. Továbbfejlesztési lehetőségek

Az alkalmazás továbbfejlesztésére több fronton is szükség van. Elsőként olyan újabb fejlesztéseket szeretnénk elvégezni, amelyek a meglévő rendszer bizonyos értelemben vett finomhangolásaként is értelmezhetőek, így javítva a használhatóságot. Ilyen funkció például a feltöltött feladatok értékelő(k)höz történő automatikus hozzárendelése akkor, ha egy bizonyos (konfigurációs paraméterként beállítható) ideig senki nem értékelt ők. Ezzel reményeink szerint elejét vehetjük annak, hogy hosszú ideig vélemény nélkül maradjon egy-egy beküldés. Mindenképpen rövid távon megvalósítandó, hogy komplexebb programok feltöltésére is lehetőség nyíljon, hiszen jelenleg csak egyetlen állomány feltöltésére van mód. Ez például

egy több osztályból álló Java program feltöltésekor túlságosan korlátozó megoldás. Szintén rövidesen tervezzük annak a funkciónak a megvalósítását, amely az oktatók értesítését végzi akkor, ha az értékelések alapján nem egyértelmű, hogy egy adott megoldás jó-e avagy sem.

Középtávon be szeretnénk építeni különböző statikus elemző eszközök [4] (mint amilyen a C nyelvhez a LINT, a Javához a FindBugs vagy éppen a CheckStyle) megállapításait, amelyek a beküldött programokra automatikusan végrehajtva bizonyos javítási javaslatokat tehet.

Harmadrészt a távolabbi tervek között a ProgCont [1, 2] rendszerrel történő együttműködés és integráció is szerepel. Egyrészt ott számos programozói feladat (elsősorban verseny-, de gyakorló és számonkérésre szolgáló feladatok is) szövege is megtalálható, amelyhez a ProgCont rendszer automatizált ellenőrzést is végez. Ez azonban a kezdő hallgatóknak sokszor nem elegendő, hiszen egy elbukó program esetén nem kapnak arról visszajelzést, hogy mi a gond a programjukkal. Erre a problémára nyújthatna segítséget a közösségi értékelés.

Végül, de nem utolsósorban, a többi továbbfejlesztési lehetőség megvalósításával párhuzamosan analitika alkalmazásával tehető intelligenssé a rendszer. Itt olyan funkciók statisztikai megalapozására kell gondolni, mint amilyen például a rendszeresen helytelen értékelést adó értékelők kiszűrése, és a későbbiekben véleményük kisebb súllyal történő figyelembe vétele.

4. Összefoglalás

A Bevezetésben említett célok minimális elérését – vagyis az értékeltetni kívánt programok, programrészek feltöltését valamint azok böngészését és értékelését – támogató szoftver elkészült. Egyelőre tesztüzemben működik, élesben a most következő (2014/2015 őszi) félévben kezdenek használni a hallgatók. A rendszer már jelenlegi állapotában is használható, azonban a Továbbfejlesztési lehetőségek fejezetben említettek megvalósítása tovább növelheti a használhatóságot és a felhasználói élményt. Természetesen fontosnak tartjuk a felhasználói visszajelzéseket is, ennek megvalósításához a Smart Campus alkalmazás [3] kérdőív moduljával történő szorosabb integráció válik szükségessé.

5. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Külön köszönet illeti Kovács Ádám PTI MSc szakos hallgatót, aki diplomamunkája elkészítésének kereteiben nyújtott segítséget a rendszer megvalósításában.

Irodalomjegyzék

[1] Kádek Tamás, Kósa Márk, Pánovics János: A ProgCont szoftverrel támogatott programozó versenyek tapasztalatai, *New Technologies in Science, Research and Education, XXV. DIDMATTECH Konferencia*, 2012, Révkomárom, Szlovákia, 152–157.

- [2] Kádek Tamás, Kósa Márk, Pánovics János: Programozó versenyek támogatása webes alkalmazással, *XXI. Számítástechnika és Oktatás Konferencia*, 2011, Kolozsvár, Románia, 184–187.
- [3] Kádek Tamás, Kósa Márk: Intelligens campus Debrecenben, *XXIII. Számítástechnika és Oktatás Konferencia*, 2013, Kolozsvár, Románia, 210–213.
- [4] Michael Striwe, Michael Goedicke: A Review of Static Analysis Approaches for Programming Exercises, in: *Computer Assisted Assessment. Research into E-Assessment* Vol. 439, 2014, Springer, 100–113.
- [5] Julita Vassileva: Toward Social Learning Environments, *IEEE Transactions on Learning Technologies* **1** (4), 2008, 199–214.

Információvédelmi képzés - egy próbát megér (?)

Information security training course – it is worth a try (?)

Michelberger Pál

Óbudai Egyetem, Keleti Károly Gazdasági Kar, Szervezési és Vezetési Intézet
michelberger.pal@kgk.uni-obuda.hu

Absztrakt: 2012-ben az Óbudai Egyetem két kara összefogott és elkészítette az „Adat- és információvédelmi szakember” szakirányú továbbképzési szak szakindítási dokumentációját. A képzést 2013-ban az Oktatási Hivatalban befogadták, létesítési határozatát kiadták, de elegendő jelentkező hiányában még nem indult. Az előadás röviden bemutatja a szakindítást motiváló tényezőket, a tantervet és a hasonló jellegű képzések, valamint a megszerezhető és nemzetközileg is elfogadott végzettségek által befolyásolt „piaci” környezetet.

Kulcsszavak: ISO 27001, információbiztonság, ISACA, etikus hacker

Abstract: In 2012, two faculties of Óbuda University joined forces and compiled the launch material of „Data and information security specialist” postgraduate training course. Although the training course was accepted by Hungarian Educational Authority in 2013, and its establishment decision was issued, the course has not started yet in absence of enough applicants. The presentation briefly describes the motivating factors of course launching, the curriculum and the “market” environment influenced by similar courses and obtainable, internationally accepted qualifications.

Keywords: ISO 27001, information security, ISACA, ethical hacker

1. Bevezetés

Az információk sértetlenségéhez, rendelkezésre állásukhoz és bizalmas kezelésükhöz – magához az információbiztonsághoz - nem elegendő az információtechnológiai védelmi megközelítés [1,2]. A brit eredetű ISO 27000-es szabványcsomag [9] szervezetek információvédelmi irányítási rendszerének kialakításához nyújt útmutatót, ill. ad támogatást. Az ISO 27001-es szabvány [3] alkalmazása világszerte terjed [7]. A kialakított irányítási rendszer független szervezet által tanúsítható. Az ilyen irányú szabályozást az üzleti környezet is egyre jobban igényli [6]. Természetesen információbiztonsági vonatkozásai más szabványoknak vagy ajánlásoknak is vannak. Nem feledkezhetünk meg a COBIT-ről [19] vagy az ITIL-en alapuló MSZ ISO/IEC 20000-1 és 20000-2 szabványokról [4,5] sem. A statisztikák alapján [8] feltételezhető, hogy nő az igény az információbiztonsági irányítási rendszer kialakításához és működtetéséhez értő szakemberek iránt is.

2. Információbiztonság oktatása

Felsőoktatási intézmények és üzleti, ill. szakmai szervezetek képzéseiben egyre többször kap helyet az informatikai biztonságon túlmutató információbiztonsági témakör; önálló tanfolyamként, külön tantárgyként, esetleg egy-egy előadás vagy gyakorlati foglalkozás erejéig. A most következő áttekintés a teljesség igénye nélkül készült. A rövid, pár napos

üzleti szempontok alapján folyó információ-, ill. IT biztonsági auditor képzések is kimaradnak [22,24].

2.1. Információbiztonság az alap- és mesterképzésben

Informatikai alapképzésekben (Gazdaságinformatikus, Mérnök informatikus és programtervező informatikus) általában egy kötelező alaptárgy jelenik meg „Informatikai biztonság alapjai” címmel, de néhány képzési helyen további kötelező és választható tárgyak is felvehetők ebben a témakörben (pl. „Informatikai rendszerek minőségbiztosítása és auditja” – DUF; „Kriptográfia” – BGF, EKF, NYME; „IT biztonság tervezése” – SZE). Az alapszakok képzési és kimeneti követelményei között egyébként megtalálhatók az „informatikai audit” (gazdaságinformatikus BSc) és „informatikai rendszerek biztonsága” (mérnök informatikus BSc), témakörök is.

Az informatikai mesterképzések tanterveiben számos információbiztonsággal foglalkozó tantárgy található. Néhány példa:

- „Adatbiztonság” – BMGE
- „IT controlling és audit” – DE
- „Informatikai auditálás” – ÓE
- „Információs rendszerek biztonságtechnikája” – PE
- „IT audit” – Corvinus

Egyéb - nem informatikai alap és mesterképzésen – az információbiztonság (mint valamilyen tantárgy jellemző témája) csak elvétve fordul elő, többségében szabadon választható tárgyaknál. Oktató kollégák gyakran érintik a témát 1-2 óra erejéig minőségirányítással foglalkozó tárgyakban, hiszen szervezeteknél lehet ún. integrált (minőség – környezet – információbiztonság – munkahelyi biztonság) irányítási rendszereket is működtetni.

2.2. Szakirányú továbbképzések

Szakirányú továbbképzések kapcsán három „kísérletet” érdemel említést.

2009-ben a **Gábor Dénes Főiskola** próbálkozott egy „**Informatika biztonsági szakmérnök**” Oktatási Hivatal által engedélyezett (regisztrált) képzéssel [14]. A kétszemeszteres, 60 kredites kurzus bemeneti feltétele főiskolai vagy alapképzésben szerzett mérnöki oklevél volt. A gyakorlatorientált képzésben 22 kreditet elméletből, 28 kreditet gyakorlatból és 10 kreditet a diplomamunkából lehetett megszerezni.

Az előremutató tematika a következő részekből állt:

- Alapismeretek
(védelem és jogosultságkezelés, alkalmazás- és web szerverek, operációs rendszerek)
- A biztonság összetevői, az információvédelem és az informatikai biztonság
- Hálózatok biztonsága
(határvédelem, Network Access Control, logelemzési eszközök, tartalomszűrés, IP telefonok)
- Adatok biztonsága
(adatszivárgás, adatvesztés elleni védelem, mentési, archiválási eszközök, eljárások)
- Az elektronikus szolgáltatások védelme
(kriptológia, protokollok, nyilvános kulcsú infrastruktúrák, az elektronikus aláírás)
- Az informatikai biztonság vizsgálata
(sérülékenység-vizsgálat, kockázatelemzés, az információs biztonság auditálása)
- Az informatikai biztonság irányításának alapelvei, előírásai
(szabványok és ajánlások)
- Projektmenedzsment ismeretek

A **Nemzeti Közszolgálati Egyetem** az állami és önkormányzati szervek elektronikus információbiztonságáról szóló **2013. évi L. törvény** alapján indítja az **Elektronikus**

információbiztonsági vezető szakirányú továbbképzési szakot [10], amely a közszolgálati információs rendszerek biztonságáért felelős személyek feladatellátáshoz szükséges szakmai kompetenciáinak átadásra és a biztonságtudatos szemléletmód kialakítására hivatott. A 60 kredit, két féléves képzés során a hallgatók az alábbi ismeretanyagot szerezhetik meg;

- Alapismeretek - jogi, vezetéselméleti és technológia ismeretek; 12 kredit (Minőségügyi ismeretek, Biztonságtechnika, Biztonságpolitika, Jogi és közigazgatási ismeretek, Vezetéselmélet, Stratégia és szervezettámogatás)
- Rendszerirányítási szakismeretek; 14 kredit (Információbiztonsági szabványok, Irányítási rendszerek, Információbiztonsági stratégia és vezetés, Biztonság támogatása)
- Információbiztonsági szervezési szakismeretek; 18 kredit (Információbiztonsági program, Biztonsági technológiák alkalmazása, Biztonságtudatossági gyakorlat, Rendszerek biztonsága, Hálózatok biztonsága, Biztonsági tesztelés gyakorlat)
- Információ kockázatok kezelése és a megfelelés; 6 kredit (Kockázateértékelés, Kockázatmenedzsment, Kockázatmenedzsment gyakorlat)
- Információbiztonsági események kezelése (incidenskezelés) ismeretek; 6 kredit (Incidens menedzsment, BCP, DRP integráció, Incidens menedzsment gyakorlat)

Az **Óbudai Egyetem** Kandó Kálmán Villamosmérnöki Kara és Keleti Károly Gazdasági Kara „**Adat- és információvédelmi szakember**” szakirányú továbbképzési szakot engedélyezett 2013-ban [11,12]. Képzés elegendő érdeklődő hiányában még nem indult. A 3 szemeszter 90 kreditet jelent, amiből 10 kredit a szakdolgozat. A képzésbe bevonásra kerülnek a Neumann János Informatikai Kar oktatói is.

Bemeneti feltétel régi főiskolai v. alapképzésben szerzett mérnöki, közgazdász (!) vagy informatikus oklevél.

A képzés tematikája:

- Információbiztonsági alapismeretek (informatikai projektmenedzsment, informatikai környezet védelme, informatikai jog, kriptográfia, információbiztonsági szervezetek és szabványok)
- Szakmai törzsanyag („etikus hackelés”, szervezeti információbiztonság tervezése, személyes adatok védelme, ISO 27001 audit, számítógépes hálózati biztonság, gyakorlati adatbiztonság, felhasználói eszközök és alkalmazások biztonsága)

A képzés levelező rendszerű. A hallgatóknak 12 vizsgát és 1 évközi jegyet kell teljesíteniük. (1 tárgy 25 óra konzultációt jelent...)

Záróvizsga tárgyaink:

- Szervezeti információbiztonság tervezése
- ISO 27001 audit
- Gyakorlati adatbiztonság

A tervek között szerepel, hogy a képzés során a hallgatók számára a TÜV Rheinland „információbiztonsági irányítási rendszer auditor”-i bizonyítvány megszerzésének lehetőségét is biztosítsuk [22,23].

2.3. „Etikus hacker” képzés

Az információbiztonsági terület speciális megközelítését, művelését teszik lehetővé az ún. „etikus hacker” képzések. Az általában kétszemeszteres (és a szakirányú továbbképzésekhez képest magasabb részvételi díjú) kurzusokat felsőoktatási intézmények (pl. NKE, ÓE) [17] és üzleti vállalkozások (pl. Kürt Akadémia, NetAcademia Oktatóközpont) is indítanak, indítottak [13].

A képzési tematikákban nem csak szűken vett információtechnológiai biztonsági kérdések szerepelnek (külső és belső hack, web, wifi, mobil), hanem a jogi ismeretek mellett megjelennek az információbiztonság emberi tényezői (social engineering) is. A képzések végén nemzetközileg elismert „Certified Ethical Hacker (CEH)” vizsgát is lehet tenni (ld. 3.1. fejezet).

3. Nemzetközi vizsga- és képzési rendszerek

A munkaerőpiacon elismertek (elismertebbek) a nemzetközi szakmai szervezetek által kiadott, bizonyos időközönként megújítandó szakmai végzettségek. Itt a teljesítendő (általában angol nyelvű) vizsgákon van a hangsúly. A felkészítő tanfolyamok elvégzése csak ajánlott.

3.1. EC Council vizsgák

Az „International Council of Electronic Commerce Consultants” szakmai szervezet egy többszintű vizsga-, ill. végzettségrendszert kínál [16].

EC-Council Network Security Administrator (ENSA – „biztonságcentrikus rendszerüzemeltető”) végzettséget egy 50 kérdéses, 2 órás teszt kitöltése után lehet megszerezni. Az Amerikai Egyesült Államok kormányzati szférájában dolgozó informatikusnál ez a végzettség alapkövetelménynek tekinthető.

A **Certified Ethical Hacker** (CEH – etikus hacker) végzettséghez kötött vizsga egy 4 órás, 150 kérdésből álló tesztet jelent, amely később az EC-Council **Certified Security Analyst** (ECSA - haladó etikus hacker) szintre továbbfejleszhető egy további 50 kérdésből álló 2 órás teszttel. A kettő együtt ad egy újabb szakmai végzettséget (**Licensed Penetration Tester**).

Magasabb tudásszintet jelent a **Computer Hacking Forensic Investigator** (CHFI - számítógépes bűnözés elleni szakértő), akinek a végzettség megszerzéséhez 4 óra alatt kell 150 kérdésre válaszolnia.

A megszerzett minősítések 3-5 évig élnek, utána megújításuk szükséges...

3.2. ISACA vizsgarendszere

Az ISACA (Information Systems Audit and Control Association) magyarországi szakmai szervezete révén (is) 4 végzettség megszerzésére biztosít angol nyelvű vizsgalehetőségeket [15].

- **CISA** – Certified Information System Auditor
- **CISM** – Certified Information Security Manager
- **CGEIT** – Certified in Governance of Enterprise IT
- **CRISC** – Certified in Risk and Information System Control

A megszerzhető végzettségek széles körben (még az államigazgatásban is) elismertek.

Felkészítő tanfolyamok néha Magyarországon is indulnak. Az e-learning-es tananyagok általában könnyen elérhetők. A végzettség fenntartásához az ISACA által elfogadott szakmai rendezvényeken történő igazolt részvétel (pontgyűjtés) szükséges.

3.3. CISSP (Certified Information Systems Security Professional)

Az (ISC)² (International Information Systems Security Certification Consortium, Inc.) non-profit, nemzetközi szervezatként tart tanfolyamokat és szervez vizsgákat **CISSP** (Certified Information Systems Security Professional) tanúsítványok megszerzése céljából [18,20]. Magyarországon ezek a Net Academia-nál és a Budapesti Műszaki és Gazdaságtudományi Egyetemen (voltak) elérhetők.

A rövid, pár napos tanfolyam során a résztvevők áttekintik az információ- és IT biztonság legfontosabb területeit (rendszerüzemeltetés, szoftverfejlesztés, fizikai biztonság, architektúra tervezés, hálózatbiztonság, titkosítás, kockázatkezelés, valamint a jogi- és menedzsmenttel kapcsolatos kérdések).

4. Melyiket válasszuk?

A fejezetcímekben szereplő kérdést a információbiztonsági tudást megszerezni kívánó szakember és a felsőoktatási intézmény is felteheti.

Üzleti vállalkozások maguk dönthetnek arról (általában környezetük nyomására), hogy előírják-e egyáltalán ilyen jellegű kötelezettséget információvédelemmel foglalkozó munkatársaiknak. Az állami és önkormányzati szervezetek azonban már nem rendelkeznek ekkora mozgástérrel. A 2013. évi L. törvény (Az állami és önkormányzati szervek elektronikus információbiztonságáról) 13. § 8. bekezdése szerint csak olyan személy végezheti az elektronikus információs rendszer biztonságáért felelős személy feladatait, aki büntetlen előéletű, rendelkezik a feladatellátáshoz szükséges felsőfokú végzettséggel és szakképzettséggel. A törvényhez készült képzési rendelet előírja, hogy ilyen pozícióhoz milyen végzettséget, ill. oklevelet fogadnak el. A Nemzeti Közszolgálati Egyetem szakirányú továbbképzése mellett (ld. 1.2. fejezet) az ISACA (ld. 2.2. fejezet; CISA, CISM, CRISC) és az (ISC)² (ld. 2.3. fejezet; CISSP) szervezetek szakképzettségeivel lehet ilyen pozíciót betölteni [21].

A többi felsőoktatási intézménynek „saját fejlesztésű”, szakirányú továbbképzéssel csak az üzleti szféra szereplőinél érdemes próbálkozni. A közigazgatás és önkormányzati területeken szintén a képzési rendeletben előírt – a Nemzeti Közszolgálati Egyetem Átképzési és Szakirányú Továbbképzési Központjával egyeztetett – rövidebb (8 vagy 50 óras) továbbképzések jelenthetnek esetleg oktatási piacot. Korábban néhány egyetem (Corvinus, ÓE) indított CISA vizsgára felkészítő kurzusokat.

A tanulmány a Magyarországon elérhető információbiztonsági jellegű képzési lehetőségek áttekintő bemutatására vállalkozott. Az érdekelt hazai felsőoktatási intézményeknek a környezet és a piaci igények tanulmányozása alapján kell/lehet újragondolni képzési kínálatukat ezen a területen.

Irodalomjegyzék

- [1] Ködmön István (szerk.): Hétpecsétés történetek (Információbiztonság az ISO 27001 tükrében). Hétpecsét Információbiztonsági Egyesület, Budapest, 2008.
- [2] Ködmön István (szerk.): Hétpecsétés történetek II. (Információbiztonság az ISO 27000 szabványcsalád tükrében). Hétpecsét Információbiztonsági Egyesület, Budapest, 2014.
- [3] MSZ ISO/IEC 27001:2014 Informatika. Biztonságtechnika. Információbiztonság-irányítási rendszerek. Követelmények
- [4] MSZ ISO/IEC 20000-1:2013 Informatika. Szolgáltatásirányítás. 1. rész: A szolgáltatásirányítási rendszer követelményei
- [5] MSZ ISO/IEC 20000-2:2007 Informatika. Szolgáltatásirányítás. 2. rész: Alkalmazási útmutató

- [6] The ISO Survey of Management System Standard Certifications 2012
www.iso.org/iso/iso_survey_executive-summary.pdf (letöltés dátuma: 2014.06.10.)
- [7] www.standardsconsultants.com/2010-top-10-countries-certified-to-iso-27001
(letöltés dátuma: 2014.06.10.)
- [8] www.iso.org/iso/home/standards/certification/iso-survey.htm
(letöltés dátuma: 2014.06.10.)
- [9] Information Security Standards www.iso27001security.com
(letöltés dátuma: 2014.06.10.)
- [10] Az Elektronikus információbiztonsági vezető szakirányú továbbképzési szak képzési és kimeneti követelményei (KKK) – Nemzeti Közszolgálati Egyetem http://vtki.uni-nke.hu/srv/www/vtki.uni-nke.hu/web/downloads/Arop/vtki/ASZT/kkk/Elektronikus_Informaciobiztonsagi_Vezeto_KKK.pdf (letöltés dátuma: 2014.06.10.)
- [11] http://kvk.uni-obuda.hu/kepzesek/szakiranyu_tovabbkepzes/adat-es-informaciovedelem (letöltés dátuma: 2014.06.10.)
- [12] <http://kvk.uni-obuda.hu/sites/default/files/adat-es-informaciovedelem-szakember-szakiranyu-tovabbkepzes-targyleiras.pdf> (letöltés dátuma: 2014.06.10.)
- [13] www.kurt-akademia.hu/hu/kepzesek/5/etikus_hacker_kepzes (letöltés dátuma: 2014.06.10.)
- [14] <http://gdf.info.hu/informatika-biztonsagi-szakmernok/> (letöltés dátuma: 2014.06.10.)
- [15] www.isaca.hu/index.php/minositesek (letöltés dátuma: 2014.06.10.)
- [16] www.training360.com/Szolgáltatások/Vizsgák/ECcouncil-vizsga.aspx
(letöltés dátuma: 2014.06.10.)
- [17] <http://nik.uni-obuda.hu/etikushack/> (letöltés dátuma: 2014.06.10.)
- [18] www.netacademia.hu/Info/CISSP (letöltés dátuma: 2014.06.10.)
- [19] www.mtaita.hu/hu/Publikaciok/ISACA_HU_COBIT_41_HUN_v13.pdf
(letöltés dátuma: 2014.06.10.)
- [20] <http://ik.bme.hu/node/17> (letöltés dátuma: 2014.06.11.)
- [21] http://net.jogtar.hu/jr/gen/hjegy_doc.cgi?docid=A1300026.KIM
(letöltés dátuma: 2014.06.11.)
- [22] www.tuv.com/hu/hungary/services_hu/education_personnel_hu/further_education_hu/information_safety_critical_infrastructure_hu/information_safety_critical_infrastructure_hu_1.html (letöltés dátuma: 2014.06.11.)
- [23] www.irca.org/ (letöltés dátuma: 2014.06.12.)
- [24] www.euzert.hu/informatika.php (letöltés dátuma: 2014.06.12.)

Ismeretalapú keretrendszerek oktatása az ELTE Informatikai Karán¹

Knowledge-based System Shells in Education at ELTE

Nagy Sára^a

^aELTE Informatikai Kar/ Algoritmusok és Alkalmazásai Tanszék

saci@inf.elte.hu

Absztrakt: Az ELTE Informatikai Karán a programtervező informatikusok mesterképzésében szerepel az Ismeretalapú keretrendszerek című tárgy oktatása a Szoftvertchnológia szakirány Intelligens rendszerek választható blokkjában.

Az ismeretalapú (vagy másképpen tudásalapú) technológiák szemléletmódja más, mint a hagyományos adatbázis-kezelő technológiáké, az alkalmazások építésének kérdései is eltérőek. Napjainkban általában hibrid módon épülnek be ezek a technológiák a hagyományos rendszerekbe.

A tárgy keretében a hallgatók többféle ismeretalapú keretrendszerrel ismerkedhetnek meg. Szerepel cél- és adatvezérelt szabályalapú, frame alapú és fuzzy bizonytalanság kezelő, valamint eset alapú keretrendszer is. A hallgatók a fejlesztő eszközök gyakorlati kipróbálásával mélyebben megérthetik az elméleti órákon felvetett problémákat.

Kulcsszavak: szakértő rendszerek, tudásbázis, következtetés

Abstract: The knowledge-based technology is different than everyday software technology. This paper presents the place of the “Knowledge-based System Shells” subject in the graduate program at Eötvös Loránd University Faculty of Informatics. The students become acquainted with some knowledge-based system shells. Using these tools the students understand the principles of expert systems. They have to develop their own consulting systems using the knowledge-based system shells.

Keywords: expert systems, knowledge base, inference

1. Bevezetés

A mesterséges intelligencia kutatások elméleti eredményeinek gyakorlatban történő alkalmazásaképpen jöttek létre a szakértő rendszerek. Ezek a rendszerek a problémamegoldás teljesen más útját követik, mint a hagyományos számítástechnikai rendszerek, ezért lehetőséget nyújtanak olyan feladatok megoldására is, amelyek korábban más eszközökkel nem voltak lehetségesek vagy nehézkesek voltak. Bár a kifejlesztett különböző tudásreprezentációs és következtetési technikák ma már megfelelő elméleti háttérrel biztosítanak szakértő, tanácsadó rendszerek építéséhez, azonban a helyes rendszer építés módszerei még mindig tisztázatlanok.

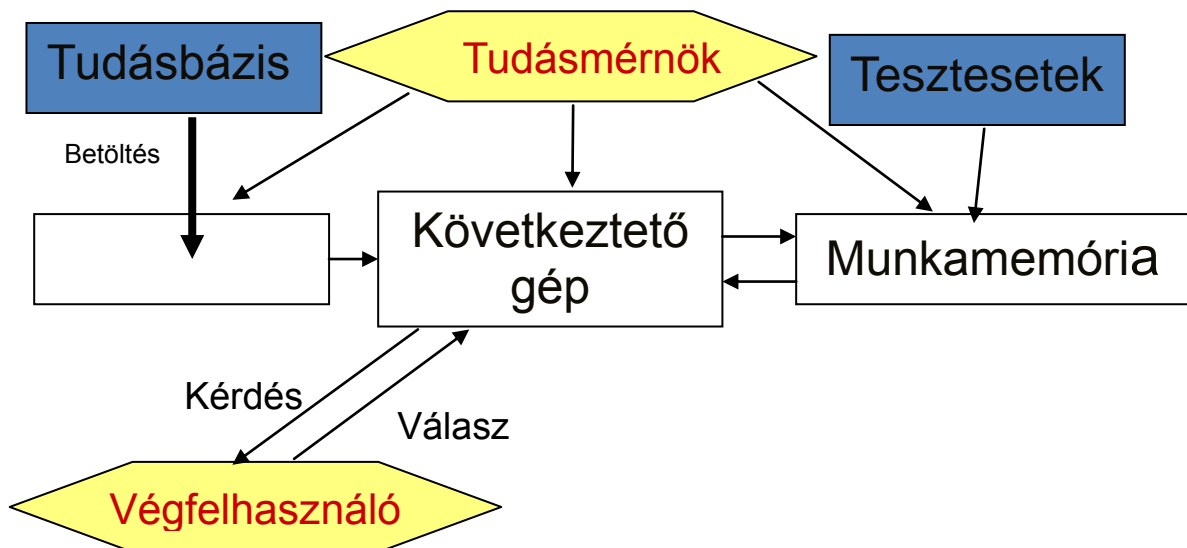
¹ A TÁMOP-4.1.2.B2-13/1 „Pedagógusképzést segítő szolgáltató és kutatóhálózatok továbbfejlesztése és kiszélesítése, 2. Országos módszertani és képzésfejlesztési komponens” pályázat támogatásával.

A tudásreprezentációs nyelvek nincsenek szabványosítva és nincs egységesen elfogadott fejlesztési módszertan sem, ezért az „Ismeretalapú keretrendszerek” tárgy keretében több, eltérő ismeretreprezentáción alapuló eszközt mutatunk be a hallgatóknak.

2. Ismeretalapú keretrendszerek felépítése

A szakértő rendszerek fejlesztése során különféle ismeretleíró nyelvek, reprezentációk születtek. Az adott reprezentációnak megfelelő következtető eljárások, következtető gépek íródtak. Sok szakterületen a tudás hasonlóképpen írható le. Így egy korábban jól megírt rendszerben elég csak a tudást „lecserélni”, és máris egy új szakértő rendszerhez jutunk. Így alakultak ki az úgynevezett shell-ek vagy magyarul keretrendszerek, amelyek megőrizték a rendszert működtető következtető gépet, a felhasználóval kommunikáló felületet, és kibővültek a tudásbázist fejlesztő funkciókkal. A rendszer felépítését az 1. ábra mutatja.

Így, ha egy új tanácsadó, szakértő rendszert kívánunk fejleszteni, akkor elég egy keretrendszer tudásleíró nyelvét megismerni, annak formájában megfogalmazni a szakterületi tudást, és készen is van az új szakértő rendszerünk. Természetesen hatékony rendszer készítését nagyban segíti a szabályalapú, esetalapú és más elveken működő rendszerek elméleti hátterének ismerete is, de korábbi minták alapján is készíthetünk kisebb tanácsadó rendszereket. Így egyszerűbb, tanulásra szánt, shell-ekkel akár középiskolás hallgatók is készíthetnek tanácsadó rendszereket. Ezért javasoljuk az „Ismeretalapú keretrendszerek” tárgy felvételét tanár szakos hallgatóknak is.



1. ábra. Ismeretalapú keretrendszer (shell) felépítése

3. A tárgy helye az ELTE Informatikai Kar oktatásában

Az „Ismeretalapú keretrendszerek” tárgy közel két évtizede szerepel az ELTE Informatikai Kar oktatásában. Korábban a Programtervező matematikusok Mesterséges intelligencia sávjában szerepelt választható tárgyként, és az Informatika tanár szakosok is felvehették szabadon választható tárgyként.

Jelenleg a Programtervező informatikus MSc képzés Szoftvertchnológia szakirány S3-as Intelligens rendszerek blokkjában szerepel. Ebben a blokkban kapott még helyet az „Ismeretalapú modellezés” című tárgy is, amely az „Ismeretalapú keretrendszerek” tárgy elméleti kiegészítőjének is tekinthető. Ez utóbbi tárgy bemutatja, hogy mit értünk ismeretalapú technológián, miként kell a megfelelő területeket kiválasztani, ahol alkalmazható, és hogyan építhető segítségével intelligens rendszer.

A két tárgy külön-külön is elvégezhető, de természetesen teljesebb képet kap az a hallgató a tudásalapú technológiáról, aki a kettőt együtt végzi el.

Az Intelligens rendszerek blokk további tárgyai: Megerősítés tanulás, Mesterséges neuronhálók, Robotika, Autonóm rendszerek.

4. A tárgy tematikája

A gyakorlati jellegű tárgy fő oktatási célkitűzése, hogy bemutassa, hogy az elméleti alapokra építve miként készíthetünk ismeretalapú alkalmazásokat, a különféle tudásreprezentációkat támogató keretrendszerek segítségével. Fontos, hogy a hallgatók minél több, más-más elven működő eszközt kipróbálhassanak. Sajnos az anyagi lehetőség hiánya miatt, csak olyan keretrendszereket tudunk használni, amelyek ingyenesen érhetők el. Így ezek általában régi vagy csak tanuló rendszerek, amelyek az internetről letölthetőek. Az évek során használt eszközök ennek megfelelően folyamatosan cserélődtek.

A kurzus teljesítéséhez a hallgatóknak 3 keretrendszer segítségével saját maguknak kell készíteni saját maguk által kitalált témában tanácsadó rendszert. Ezen keretrendszerek jellemzőit a 1. táblázatban láthatjuk. Valamint a kurzus végén a megismert rendszerekkel kapcsolatos tesztet kell sikeresen kitölteniük.

A beadandók elkészítéséhez használt rendszereken kívül még bemutatásra kerül egy esetalapú keretrendszer is a CBR-Works 4.

Keretrendszer neve	Típusa	Következtetés iránya	Leíró nyelv	Bizonytalanság kezelés	Operációs rendszer
M1	szabályalapú	visszafele haladó	Prolog szerű	bizonyossági faktoral	DOS
CLIPS	szabályalapú	előrefele haladó	LISP szerű	nincs beépített	DOS, UNIX, Windows
FLEX	frame- és szabályalapú	mindkét irányú	Prolog szerű	fuzzy logikával	Windows

1. táblázat. Bemutatandó keretrendszerek jellemzői

5. Összegzés

Az elmúlt évek oktatási tapasztalata szerint a hallgatók szívesen választják a cikkben ismertetett tárgyat. Egy programtervező számára nem elég, ha csak a hagyományos szoftver technológiában jártas, mivel ma már a legtöbb piacképes szoftver (eszköz, alkalmazás) rendelkezik intelligens komponensekkel, szolgáltatásokkal. A hallgatók érdeklődése abban is megnyilvánul, hogy mind az elméleti, mind a gyakorlati tárgy témáiból rendszeresen készülnek diplomamunkák, szakdolgozatok. Tervezzük az MI rendszerhez hasonló, tanulásra igen alkalmas, egyszerű keretrendszer kidolgozását is különféle platformokra.

Irodalomjegyzék

- [1] Sántáné-Tóth E.: Tudásalapú technológia, szakértő rendszerek, Javított és módosított kiadás (Dunaújvárosi Főiskola Kiadói Hivatala, Dunaújváros, 2000, 2007)
- [2] Gregorics T.: Mesterséges intelligencia (ELTE IK Digitális Könyvtár 2008)
- [3] Futó I. (szerk.): Mesterséges intelligencia (AULA Kiadó, 1999)
- [4] Russell, S. and Norvig, P.: MI - modern megközelítésben (Panem Kft, 2000)
- [5] Fekete I., Gregorics T., Nagy S., „Bevezetés a Mesterséges intelligenciába. Heurisztikus gráfkeresés, kétszemélyes játékok, automatikus tételbizonyítás”, LSI Oktatóközpont, 1990.
- [6] Harmon, P. and Sawyer, B., “Creating Expert Systems for Business and Industry”, Wiley&Sons, Inc., 1990.
- [7] N. J. Nilsson: Principles of Artificial Intelligence (Springer-Verlag, 1982)

Felhő alapú szoftverek használata az informatikatanár képzésben

Cloud Technologies in ICT Teacher Training

Pozsgai Tamás^a, Lipovits Ágnes^b, Gál Balázs^c

^aPannon Egyetem

pozsgai@almos.uni-pannon.hu

^bPannon Egyetem

lipovitsa@szt.uni-pannon.hu

^cPannon Egyetem

galb@almos.uni-pannon.hu

Absztrakt: A középiskolás diákok számítógép használati szokásai jelentősen megváltoztak az új eszközök (tablet, okostelefon) megjelenésével. Az új eszközök más hozzáállást követelnek meg a használóiktól. A tabletek, okostelefonok sokkal kevesebb adatot, programot tárolnak háttértárakon, viszont erősebben támaszkodnak az internet adta lehetőségekre. A középiskolai informatika tananyag jelenleg számítógépközpontú. Az érettségi feladatsor megoldásakor nem lehet internetet használni. A diákok azonban a megszerzett tudásukat más platformokon szeretnék használni, ami nem minden esetben egyszerű feladat. A középiskolai tanárok sincsenek felkészítve az új eszközök használatára, nem, vagy csak nagyon kevésbé ismerik az elérhető programokat, alkalmazásokat.

Az irodai programcsomagok jelentős fejlődésen mentek át az utóbbi néhány évben. Megjelentek a felhő alapú változataik. Új szereplők jelentek meg (Google). A felhő alapú irodai programok tudása ugyan még elmarad az asztali változatokétól, de néhány szoftver már eléri a középiskolai érettségiben elvárt szintet. Néhány ilyen szoftvert vizsgálunk meg olyan szempontból, hogy mennyire éri el a jelenleg középiskolában elvárt szintet, milyen szerepet kaphatnak ezek a szoftverek az oktatásban, van-e, lesz-e létjogosultságuk a jövő oktatásában.

Kulcsszavak: informatikatanár, irodai szoftverek, felhő

Abstract: With the invention of new devices (tablets, smartphones) ICT usage habits of secondary school students has changed extensively. New devices demand different approaches. Tablets and smartphones store much less data on storage devices, while rely much heavier on internet resources. Secondary school curriculum tend to be focused on personal computers. During the school leaving exam internet use is prohibited. Students are eager to use the acquired knowledge on various platforms, however difficult it may be. Secondary school teachers also need further assistance in using the new technologies, software solutions.

Office software suites have developed dynamically in the past few years, cloud based solutions evolved. New participants like Google appeared on this market segment. Though the functionality of the cloud based solutions is still limited as compared to their desktop version counterparts, some of them are already suitable for solving the practical part of the school leaving exam. In this paper we are investigating some available on-line solutions, focusing on their functionality, their appropriateness for the task. We are focusing on their potential future role in education.

Keywords: *teacher of ICT*, office software, cloud technology

1. Bevezetés

A középiskolai informatikaoktatás az asztali számítógépet használja eszközü. A tananyagok célja is a számítógép és a rajta található szoftverek használatának elsajátítása. Ennek megfelelően a képzésben is az asztali számítógépen található szoftverek megismerése kap kiemelt szerepet a tanárképzésben. Az elmúlt években azonban a középiskolás korosztály számítógépezési szokásai megváltoztak. Olyan eszközök jelentek meg a mindennapjaikban, amiket már nem csak hobbiként használnak a diákok, hanem a tananyagot is fel tudják dolgozni segítségükkel. Ezek az új eszközök, elsősorban a táblagépekre és az okostelefonokra gondolva, az asztali gépeknél sokkal jobban támaszkodnak az internetre. Már nem csak információszerzésre, tárolásra használják, hanem olyan alkalmazásokat használnak, melyek hamarosan alkalmassá válnak arra, hogy kiválthassák az asztali gépekre telepített programokat.

A középiskolai tananyag jelentős része az irodai programcsomag valamelyik részéhez köthető (szövegszerkesztés, táblázatkezelés, prezentációkészítés, adatbázis kezelés). A számítógépekre telepített irodai programok jelenleg még sokkal jobban használhatók, nagyobb tudásúak, mint az interneten található megfelelőik. Azonban már most rendelkeznek ezek a programok olyan előnyökkel, amik miatt érdemes megismerkedni velük. Ilyen előnyök lehetnek a dokumentumot egyidejű szerkeszthetősége több felhasználónak, a szerkesztések nyomon követhetősége stb.

Ebben a dolgozatban az online irodai programok felhasználhatóságát vizsgáljuk, illetve azt, milyen módon jelennek az informatikatanár képzésben egyetemünkön [1].

Irodai szoftverek

Az iskolák döntő többsége irodai szoftverként a Microsoft Office programcsomag valamelyik verzióját használja. Jóval kisebb arányban dolgoznak LibreOffice programmal. Ezen két program található meg az érettségi vizsga szofverlistájában is [2]. A két szoftver közül a Microsoft Office programnak létezik online változata, a LibreOffice-nak még nem.

Az Alkalmazói rendszerek és az Oktatási média ismeretek kötelező tárgyak oktatása során kerülnek elő az irodai programok. Ezekben a kurzusokon kerülnek szóba az irodai szoftverek. Az órákon döntően az offline verziókkal kapcsolatos feladatok, újdonságok kerülnek megbeszélésre, de mint újdonságok említésre kerülnek az online programok is. Az irodai szoftverek alkalmazásai közül három nagyobb témakör kerül feldolgozásra ezeken a kurzusokon: szövegszerkesztés, táblázatkezelés, prezentációkészítés. Az adatbázis kezelés némileg speciális helyzetben van. Külön kurzusként oktatja egyetemünk, de ebben a témakörben a legnagyobb a különbség az offline változat (Access, Base) és az online változatok között (MySQL, Phpmyadmin). Az órákon az online programok között előkerül a két legnépszerűbb változat, a Microsoft Office online változata [3] és a Google Docs [4] alkalmazásának elemei. Ezek mellett minden évben megvizsgálunk néhány ingyenes alkalmazást is. Ezek évről évre változnak, sokszor a hallgatók igénye szerint.

I. Szövegszerkesztés

A szövegszerkesztés témakörének vizsgálata hálás feladat, tapasztalataink szerint ezekben a programokban mozognak leggyakrabban a hallgatók. A legtöbb problémát az ismeretlentől való félelem okozza, akik először próbálkoznak egy böngészőprogram segítségével szöveget szerkeszteni, eleinte nehezen találják meg az alapvető funkciókat is (mentés, visszavonás). Szintén problémákat okoz a már meglévő feladatok (példatárak, saját, előre elkészített feladatok esetén is) átfogalmazása az új környezethez alkalmazkodva (távolságok megadása centiméter helyett pixelben, angol nyelvű menü). Azonban azt sem tagadhatjuk, hogy az

online programok esetén sok tudáselem még nem megvalósítható. Azonban ezek a szoftverek rohamosan fejlődnek és egyre többen használják őket. Évről-évre egyre több tudáselemet lehet megoldani velük.

A Microsoft Word Online-t a tavalyi év során fejlesztette jelentősen a Microsoft, de folyamatosan jelennek meg a frissítések. Legnagyobb hibája, hogy nincs még magyar nyelvű változata (miként az összes Microsoftos online programnak sincs). Megjelenésében nagyon hasonlít az Word 2013-as változatára.

A Google Docs Writer programja rendelkezik magyar menüvel, azonban itt más jóval több eltérést tapasztalhatnak a felhasználók. Kicsivel másabb a menürendszer, néhány menüt másképpen nevezték el, azonban hamar megszokhatók a különbségek. Néhány fontos elemmel azonban nem, vagy nehezen birkózik meg a legfrissebb változat is (hasábok kezelése). Ami nagy előnye lehet ennek a programnak a mögötte álló cég. Az online világban a Google termékei sok helyen megkerülhetetlenek (keresések, levelezés). Mivel sokan használják az egyéb termékeket és szeretnének egy platformon dolgozni, ezért választják ezt. Szövegszerkesztési tudásban azonban jelenleg elmarad a Word Online programtól. Amiben viszont könnyebben kezelhető, a megosztott használat. Ötletesen külön színekkel jelölik a különböző felhasználókat, akik közben megbeszélhetik a kiosztott feladatokat, segíthetnek egymásnak.

A két népszerű alkalmazás mellett minden évben igyekszünk felhívni a hallgatók figyelmét néhány egyéb alkalmazásra is. Több fórumon a Zoho cég [5] alkalmazásai kerülnek ki győztesen az összehasonlításokból, ezért ezeket a programokat legalább beadandó szintjén megismertetjük a hallgatókkal. általában összehasonlításokat várunk tőlük, illetve feladatmegoldásokat, előre elkészített feladatok megoldhatóságát (érettségi feladatok). A Zoho alkalmazásokat kicsivel nehezebb megtalálni a weboldalon, de utána egy jól kezelhető alkalmazást kapunk. Azonban furcsán hat, hogy jelen állapotában csak félig van magyarítva.

Összességében mindhárom program jelen állapotában már alkalmas a középszintű érettségi feladatok közel 90%-os megoldására. Ez az arány néhány évvel ezelőtt még csak 30-40% volt. Vannak azonban olyan témák, amikkel nem tudnak megbirkózni a programok (körlevélkészítés).

II. Táblázatkezelés

Az online táblázatkezelő programokról általánosságban elmondható, hogy nagyobb a lemaradásuk a telepített változathoz, mint a szövegszerkesztők esetében. Azonban az is igaz, hogy döntően olyan elemek hiányoznak belőlük, melyek a középiskolai tananyagban nem szerepelnek. Természetesen a beállítási lehetőségek szegényesebbek minden menüben, de az igazi hátrány a függvények kezelése. A helyzet azonban folyamatosan javul.

Az Excel Online programot teljesen hasonlóan tervezték, mint az offline változatot. A legújabb változata már nem csak a legszükségesebb függvényeket tartalmazza. Ami hátránya, hogy csak angol nyelven érhető el minden függvény. Tapasztalataink szerint az angol nyelvű programhasználat sok esetben inkább a tanárokat zavarja jobban, mint a diákokat.

A Google Docs Spreadsheet alkalmazás sokkal kevesebbet tud, mint a microsoftos vetélytársa. A legnagyobb lemaradás talán a függvények kezelésében mutatkozik. A megszokott beszűrős módon csak az alapfüggvények érhetőek el jelenleg, a többi függvényt felismeri ugyan a program, de beszűrni nem, csak beírni tudjuk őket. Emellett kevesebb formázási lehetőségünk van. Előnye az Excel Online-hoz képest, hogy a menürendszer itt is magyar nyelvű (a függvények azonban itt is angolul vannak meg).

A Zoho Sheet a kettő között helyezkedik el tudását tekintve. Függvényeket jobban használja, mint a Google alkalmazása, de kevesebb függvényt ismer, mint az Excel Online. A félig magyar, félig angol menürendszer itt kifejezetten zavaróan hat. A menükezelése viszont talán a leghatékonyabb a három alkalmazás közül.

III. Prezentációkészítés

A három vizsgált alkalmazási terület közül itt érezhető a leglátványosabb fejlődés mindhárom programban. Néhány évvel ezelőtt az online módban készített prezentációk még statikus bemutatók voltak, azonban a Html5 szabvánnyal és az ezzel járó fejlődéssel mára már animációkkal teletűzdelt prezentációk készíthetők.

A PowerPoint Online menürendszere a szokásos módon a telepített PowerPoint programra hasonlít, ám jóval kevesebb mindent tud megvalósítani. Az előző két programhoz képest ez van leginkább elmaradva az offline változathoz képest. Sem a formázási lehetőségek, sem a beépített témák mennyisége nem éri el a várt szintet. Ami viszont jól használható, az új dia beszúrásakor az adott témához elkészült több változat is, attól függően, mit szeretnénk a dián megjeleníteni.

A Google Docs Presentation alkalmazása lényegesen több beépített sablonnal képes dolgozni, mint a Microsoftos vetéltársa. A menürendszere is könnyebben használható, ráadásul magyarul is elérhető. A témákból itt többet találunk, viszont az újonnan beszúrt diák egyformák lesznek.

A Zoho Show alkalmazása már csak angolul érhető el, ami talán szerencsésebb, mint a félig magyarított Writer. A menürendszer megvalósítása eltér a többitől, de jól átgondolt, könnyen használható. Beépített témákból itt sem találunk túl sokat és az animációk is szegényesek még. Az elkészített prezentációk azonban jó minőségűek lesznek.

Az online prezentáció készítésénél nem hagyható figyelmen kívül az egyre népszerűbb Prezi [6] alkalmazás sem. A prezentációkészítés ezzel az alkalmazással nem egy lineáris diasorozat, sokkal inkább hasonlítható egy virtuális vászonhoz [7], ahol a diáinkat tetszőlegesen rendezhetjük. Ez az újfajta rendezési mód új feladatok kitalálását is megköveteli. A már meglévő feladatok nehezen fogalmazhatók újra, mivel a Prezi ingyenes változata elég kevés formázási lehetőséget biztosít. Az újszerűsége miatt azonban lassan megkerülhetetlen lesz ebben a témakörben.

2. Adatbázis kezelés

Az adatbázis kezelés, mint középiskolai tananyag teljesen az Access program éppen aktuális verziójára épül. A tankönyvek feladatai is erre a programra vannak kitalálva. A nehézséget az adja, mikor szeretnénk felhő alapú szoftverrel megoldani ezeket a feladatokat, hogy nem létezik alternatívája az Accessnek. Természetesen tudunk adatbázist kezelni az interneten is, de azok a kezelő programok más elveken épülnek fel, jóval nagyobb informatikai ismeretet tételeznek fel a diákoktól (SQL nyelv ismerete). Az internetes adatbázis kezelést nem tudjuk átültetni a középiskolai adatbázis kezelő feladatokra, más kurzusokon történik az oktatása, ahol nem cél a középiskolai feladatok megoldása ezekkel az eszközökkel [8]. Azonban több kurzuson is előkerülnek feladatok (Adatbázis kezelés, Web programozás), melyek hasonlóak a középiskolai feladatokhoz, de másfajta megoldást kívánnak.

3. Összefoglalás

Az informatika oktatása különleges helyzetben van középiskolai tanításban. Annyira gyorsan változik a tananyag, mint talán egyetlen másik tárgynál sem. Az eszközök változnak, a szoftverek frissülnek, lecserélődnek, amit folyamatosan követniük kell az oktatóknak, a tankönyveknek, tanmeneteknek. Az informatikatanárok képzésénél igyekszünk nagy hangsúlyt fektetni az új eszközök használatának megismertetésére. Bízgatjuk a hallgatókat,

hogy bátran próbáljanak ki új szoftvereket akkor is, ha nem minden esetben lehet elérni teljes sikert elérni (nem oldható meg minden részfeladat, vagy máshogyan oldható meg). Az oktatás során olyan helyeken is kipróbálunk online szoftvereket, ahol előre tudjuk, hogy jelen pillanatban még nem férnek be a középiskolai oktatás keretei közé. Az irodai szoftverek világa tipikusan ilyen terület, hiszen az érettségi során nem használható internet. A diákok azonban egyre növekvő arányban használják ezeket az alkalmazásokat, emiatt a tanárok sem hagyhatják figyelmen kívül. Mindezen hátrányok mellett több előnnyel is jár, ha ezeket az alkalmazásokat is bevonjuk a tanításba. Az újdonság, a szokatlan a diákokra is motiválóan hat, csoportmunkára ösztönözhetjük őket, új típusú feladatokat készíthetünk [9].

A megvizsgált alkalmazások nem érik el az offline programok szintjét. A jövő egyik lehetséges útja azonban lehet ez az út is. A programok nagyon dinamikusan fejlődnek, több multinacionális cég is nagy pénzt és energiát fektet a fejlesztésükbe. A közösségi média robbanásszerű elterjedésével ezek a programok is létjogosultságot nyertek, talán a középiskolai oktatásban is.

Irodalomjegyzék

- [1] <https://docs.google.com/file/d/0B2q7RYfGVqa2c05MNkU0bHhoX2c/edit>
- [2] http://dload.oktatas.educatio.hu/erettsegi/nyilvanos_anyagok_2014majus/info_emelt_szoftverlista_2014maj.pdf
- [3] <https://onedrive.live.com>
- [4] <http://google.hu>
- [5] <http://zoho.com>
- [6] <http://prezi.com>
- [7] <http://hu.wikipedia.org/wiki/Prezi>
- [8] <http://phpmyadmin.net>
- [9] <http://www.sulinet.hu/etwinning/kepzes/web2.html>

Magyar informatikusképzés a Babeş-Bolyai Tudományegyetemen

IT Training at Babeş-Bolyai University in Hungarian

Robu Judit^a, Soós Anna^b

^aBabeş-Bolyai Tudományegyetem
robu@cs.ubbcluj.ro

^bBabeş-Bolyai Tudományegyetem
asoos@math.ubbcluj.ro

Absztrakt: Kolozsváron az egyetem megalapítására visszanyúló hagyománya van a természettudományos kutatásnak és a reál tárgyak oktatásának. Az 1971-ben induló informatika képzés a már meglévő magas szintű matematika oktatás alapjaira építkezhetett. 1993 óta működik magyar nyelvű informatikusképzés a Babeş-Bolyai Tudományegyetemen, mely egyre növekvő beiskolázási számokkal, fejlődő infrastruktúrával és kapcsolatrendszerrel szolgálja a számítástechnikai szakemberek színvonalas képzését. Az alapképzés mellett több mesterszak is indult, melyek a piaci követelményeknek megfelelően rugalmasan alakulnak, hogy a képzési kínálatot a keresletnek megfelelően alakítsák. Ezt szolgálja a tanterv alakulása is, melyben a matematikai tárgyak kezdeti súlya csökkent, ma már az informatika műveléséhez specifikusan szükséges témák, területek kapnak hangsúlyt. Az opcionális tárgyak tág kínálata lehetővé teszi, hogy a hallgatók már az alapképzésben több területbe beleszólhassanak, esetleg már itt elinduljanak a szakosodás irányába. A kötelező céges gyakorlat bevezetésének, illetve széleskörű belföldi és nemzetközi vállalati kapcsolatrendszernek köszönhetően hallgatóink gyakorlati készségeiket is fejleszthetik. A folyamatos fejlődés ellenére azonban számos nehézséggel is találkozunk, hiszen a céges fizetések és a folyamatosan érkező munkajánlatok már az egyetemi évek alatt elvonják a hallgatók energiáit a tanulástól és csökkentik a mesterképzés elvégzésére irányuló motivációikat.

Kulcsszavak: kolozsvári magyar informatikusképzés, képzési kínálat, vállalati kapcsolatok

Abstract: In Cluj there is a tradition of research in natural sciences and teaching of these subjects that dates back to the foundation of the university. The informatics study program started in 1971 and was based on the high level mathematics program. The Hungarian program of informatics was launched in 1993 and has been working ever since on training well-educated and capable professionals. Besides the BA level informatics program have been founded several MA level programs that orient themselves dynamically according to the ever changing requirements of the employment market. The development of the curricula is also market-oriented, there has been a decrease in the emphasis on mathematics, nowadays the purpose of the mathematical subjects is to give a foundation to computer science skills and knowledge. The broad range of optional subjects gives the possibility to students to try out different fields of informatics or to start specialization early if they wish to. Our students have the possibility to develop their practical skills during their compulsory practical training at firms in Romania or abroad. In spite of the continuous evolution, we also encounter several difficulties since the salaries offered by firms and the continuous flow of employment possibilities lure away students from university training, their motivation of finalizing their university studies with a masters diploma decreases.

Keywords: computer science training in Cluj, educational offer, industrial relationships

1. Történeti bevezető

A kolozsvári magyar nyelvű felsőfokú oktatás több évszázados múltra tekinthet vissza, mely a korántsem volt viszontagságtól mentes. A Báthory István által 1581-ben alapított akadémia alig néhány évtizednyi működés után bezárta kapuit s az ezt követő két évszázad

felsőfokú oktatásra vonatkozó próbálkozásai sorra kudarcba fulladtak. 1872-ig kellett várni arra, hogy szervezett módon, a tudományok tág spektrumát átfogva, intézményesített módon elinduljon az erdélyi értelmiség képzése a frissen megalapított Kolozsvári Magyar Királyi Tudományegyetemen, mely 1881-ben vette fel a Magyar Királyi Ferenc József Tudományegyetem nevet. Berde Áron első rektor irányításával négy karon (Jog- és Államtudományi Kar, Orvosi Kar, Bölcsészlet-, Nyelv- és Történettudományi Kar, Matematikai és Természettudományi Kar) indult el az oktatás. [3]

A Matematikai és Természettudományi Kar tanárai között már a kezdetektől megtaláljuk a kor neves matematikusait, akik lelkes és szakszerű munkájukkal lehetővé tették, hogy a kolozsvári egyetem kezdetben provinciális helyzete, mostoha anyagi körülményei és történelmi hányattatásai közepette is jelentős tudományos eredmények szülessenek Erdélyben. Vályi Gyula, Farkas Gyula, Fejér Lipót, Haar Alfréd, Riesz Frigyes, Dávid Lajos neve a kor (és nem csak) matematikusai számára mind ismerősen csengett, talán túlzás nélkül állíthatjuk, hogy világszínvonalú és jelentőségű matematikát műveltek. [2, 6, 7].

A kommunista rezsim megpróbálta ellehetetleníteni a magyar nyelvű egyetemi oktatást, ezt szolgálta a Babeş és Bolyai egyetemek 1959-es összeolvasztása is. Az egyre nehezebbé váló körülmények ellenére továbbra is tanítottak az egyetemen magyar matematikusok, akik óráik többségét románul tartották ugyan, de a klasszikus tanár-tanítvány, mentor-diák kapcsolatok keretében továbbra is gondoskodtak arról, hogy a tehetséges magyar fiatalok megfelelő képzésben részesüljenek. [2]

Az informatikusképzés 1971-ben indult el a Babeş-Bolyai Tudományegyetemen román nyelven. Az egyetemi képzés első 2 éve azonos volt a matematikusokéval, majd az utolsó 3 évben tanultak a hallgatók speciális tárgyakat is. A kezdeti 75 fős beiskolázási számot néhány tanév után 25 fősre csökkentette a román állam. [5]

A rendszerváltás után néhány évvel, 1993-ban vált lehetővé, hogy a teljes informatikai képzés magyarul történjen, ekkor kapott az egyetem először külön beiskolázási számokat a magyar nyelvű hallgatók számára. Az első tanévben 20 magyar hallgatót vettek fel költségtérítéssel helyre, a következőkben már 25-öt. [4, 5] Az informatika oktatása fontosságának erősödését jelzi az is, hogy ez a kar nevében is megjelent 1994-ben, ekkor nevezték át a Matematika Kart Matematika és Informatika Karrá.

A '90-es években elinduló fejlődés azóta is tart, hiszen 2011-ben a tanügyi törvény lehetővé tette az etnikai alapokon szerveződő intézetek létrehozását, így megalakulhatott a Magyar Matematika és Informatika Intézet, mely jelentős autonómiával rendelkezik a karon belül. A 2014-2015-ös tanévre 80 tandíjmentes és 20 tandíjas magyar helyre pályáznak az alapképzésre felvételizők, a mesterképzésen pedig 3 szakon 29 tandíjmentes és 56 tandíjas hely áll a továbbtanulni vágyók rendelkezésére.

A romániai egyetemek 2011-es rangsorolása szerint a BBTE osztozik az 1-2 helyen a Bukaresti Egyetemmel, azaz kutató- és oktatóegyetem, az egyetlen romániai tudományegyetem, mely az alapképzéstől a doktoriig biztosítja az magyar nyelven való tanulás lehetőségét. Egészen friss eredmény, hogy a 2013-as Shanghai listán a kolozsvári egyetem matematika doméniuma bejutott az első 150 közé. [1]

2. Informatika az alap- és mesterképzésben a BBTE-n

Az 1993 óta folyamatosan működő magyar nyelvű informatika alapszak mellett 2002-ben indítottuk az első, ekkor még egy éves magyar mesteri szakot Komputacionális matematika címmel. Ez mai napig működik (2 éves rendszerűvé átalakítva), ám az újabb informatikai mesterszakok megjelenésével ma már sokkal inkább a matematikára került benne a hangsúly, matematika mesteri szakká alakult. A bolognai rendszer bevezetése után, 2006-ban indult el a Számítógépes rendszerek optimalizálása mesterszak, mely kezdetben 1 éves

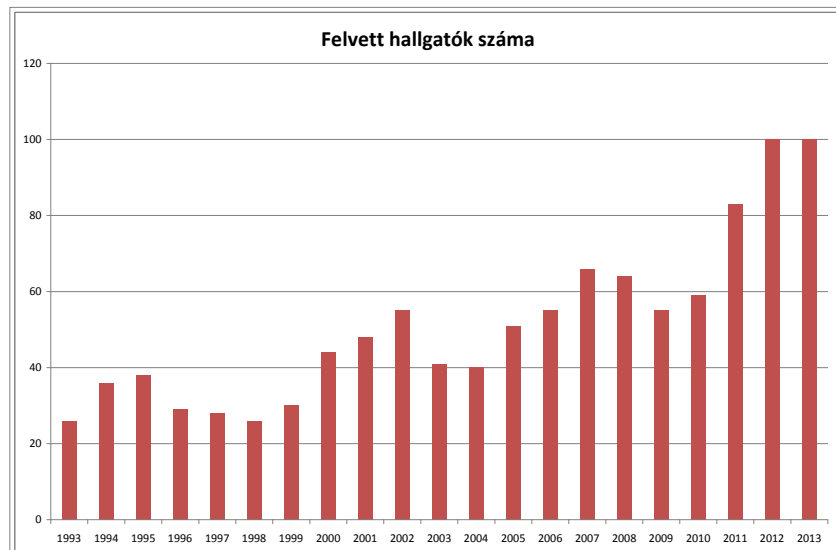
volt, 2008 óta pedig 2 éves rendszerben működött tovább. 2013-ban újabb 2 éves magyar mesteri szakot indítottunk Vállalati szoftvertervezés és alkalmazásfejlesztés néven, a következő tanévtől pedig az Adatelemzés és modellezés mesteri szak is útjára indul. A 3 szakon összesen 29 költségtérítéssel és 56 önköltséges helyre pályáznak a hallgatók.

A képzés 3. fokaként 2008-ban indult matematika-informatika doktori iskola a karon, azóta 14 magyar hallgató szerzett doktori fokozatot. 2009 óta működik kettős doktorképzésünk az ELTE-vel, melynek keretében eddig 3 hallgató szerzett fokozatot.

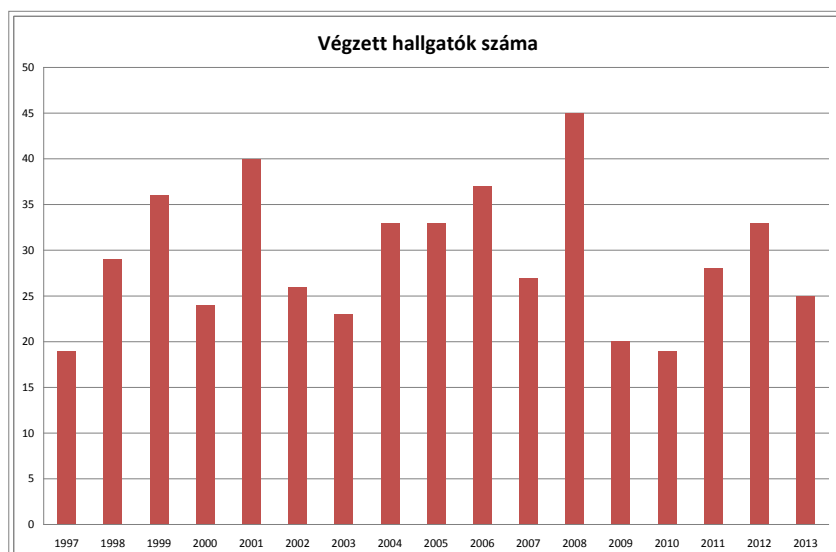
3. A hallgatók számának alakulása

Mint a fentiekben már részletesebben kifejtettük, az informatikusképzés a Babeş-Bolyai Tudományegyetemen 1971-ben indult el román nyelven, majd 1993-ban magyar nyelven is.

A beiskolázási számok a kezdeti 20-25-ről fokozatosan nőttek, ezzel párhuzamosan a jelentkezők száma is növekvő tendenciát mutat. Az 1. ábrán látható, hogy az előző két tanévben sikerült mind a költségtérítéssel, mind pedig az önköltséges helyeket betölteni.



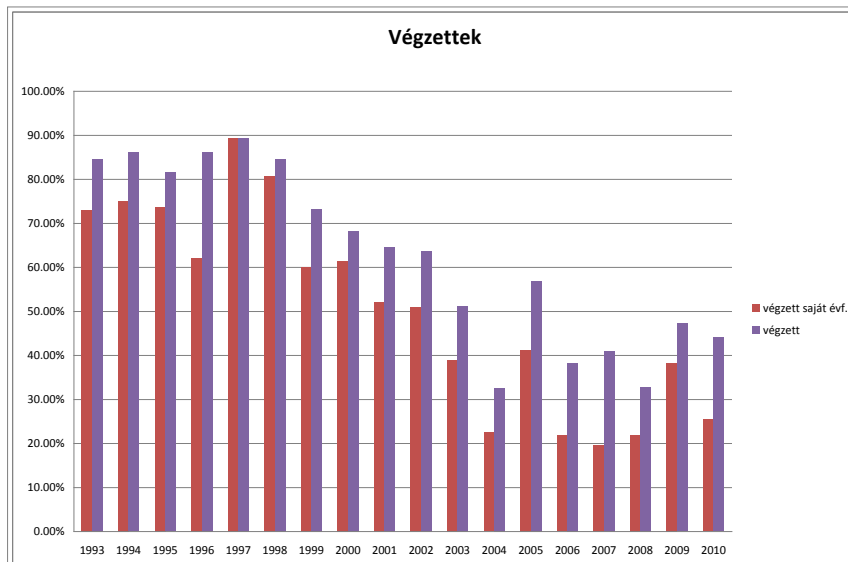
1. ábra. Felvett hallgatók létszámának alakulása



2. ábra. Abszolutóriumot szerzett hallgatók létszámának évenkénti alakulása

A 2. ábra az abszolutoriumot szerzett hallgatók számát mutatja éves bontásban. Ez minden hallgatót tartalmaz, aki az adott tanévben sikeresen letette minden vizsgáját függetlenül attól, hogy melyik évben kezdte meg tanulmányait. A 2008-as csúcs annak köszönhető, hogy az adott tanévben a bolognai rendszerre való átállás miatt két évfolyam (a 2004-ben és 2005-ben felvételt nyertek) is végzett egyszerre. Látható, hogy a felvett hallgatók számának növekedésével nem jár együtt a végzettek számának növekedése.

Annak érdekében, hogy árnyaltabb képet kaphassunk, érdemes lehet megvizsgálni, hogy az egyes években felvett hallgatók hány százaléka végzett a saját évfolyamával, illetve hány százalékuk szerzett abszolutoriumot összesen (a saját évfolyamával vagy később). Ezeket az adatokat mutatja be a 3. ábra.



3. ábra. Saját évfolyammal / összesen abszolutoriumot szerzett hallgatók létszámának alakulása

Látható, hogy már a kezdetektől volt egyfajta szelekció, nem minden felvett hallgatónak sikerült be is fejeznie az egyetemet, az első vagy második tanév után a hallgatók egy része pályát módosított. A saját évfolyammal, illetve később abszolutoriumot szerzők közötti különbség részben azzal magyarázható, hogy a hallgatók egyre nagyobb százaléka kezdett el dolgozni már az egyetemi évek alatt, így nem sikerült minden követelményt teljesíteniük az egyetemi évek alatt.

Az eredmények azonban a 2000-es évek elejétől kezdenek egyre lehangolóbbá válni, a tanulmányaikat elkezdett hallgatók mind kisebb százaléka fejezi be az egyetemet akár saját évfolyamával, akár később. Kezdetben ennek az az oka, hogy a matematika szakos hallgatóknak lehetősége volt az egyetemi képzés ideje alatt átiratkozni informatikára, így azok, akik a felvételin nem jutottak be, a kevésbé népszerű matematika szakon kezdték meg tanulmányaikat azt remélve, hogy menet közben válhatnak informatikára, azonban menet közben sokszor bebizonyosodott, hogy nem bírnak eleget tenni a követelményeknek, így esetükben erősebben érvényesült a szelekciós hatás. A 2004-es mélypont oka, hogy a 2001-2002-es tanévtől megszűnt a felvételi vizsga, a hallgatók a középiskolás jegyeik átlaga és az érettségi eredményeik alapján jutottak be, így nem volt semmiféle szűrés arra, hogy matematikából, illetve informatikából milyen előzetes tudással rendelkeznek. Az ő esetükben ez a szelekciós hatás még erősebben jelentkezett, sokan már az első tanév vége előtt abbahagyták az informatika szakot.

A 2011-es tanügyi törvény lehetővé tette a felvételi vizsga visszavezetését. A tapasztalat az, hogy az azóta bejutottak között sokkal kisebb a pályamódosítók aránya, azonban az, hogy hány százalékuknak sikerül be is fejeznie az egyetemet csak 2016-tól lesz vizsgálható.

4. A tantervek jellegzetességei

Az 1993-ban elinduló magyar informatika-oktatásban a bolognai rendszer bevezetéséig viszonylag kevés változás történt [4]. A matematika tárgyak túlsúlya volt jellemző az első néhány félévben, melyeket a matematika szakos hallgatókkal közösen (azonos színvonalon) tanultak az informatikus hallgatók is, erre épültek rá a képzés második felében a sajátosan informatikai tárgyak. A bolognai rendszer 2005-ös bevezetésével a matematikai tárgyak száma csökkent, ezeket külön tanulják az informatika szakos hallgatók azzal a céllal, hogy a számítástechnika szempontjából fontos témákra, területekre kerülhessen a hangsúly.

Jelenleg a tantervben szereplő tárgyak körülbelül 80% kötelező tantárgy, a maradék választható. Az angol nyelv tanulása és a sport 2-2 féléven keresztül kötelező.

A 2007-es tantervtől kötelezővé tettük a cégeknél végzett 3 hetes (90 órás) nyári gyakorlatot.

5. Együttműködések

A BBTE informatikusképzése kiterjedt bel- és külföldi ipari, illetve nemzetközi akadémiai együttműködés kontextusában zajlik. Lehetőséget biztosítunk a hallgatóinknak arra, hogy neves kolozsvári cégeknél végezzék el kötelező szakmai gyakorlatukat (pl. Evoline, Codespring), de vannak nemzetközi ipari együttműködéseink is az EIT ICT Labs Budapest keretében, illetve a németországi Siemens vállalattal is, mellyel mesteri és doktori együttműködésünk is van. A tavaszi félévben gyakorlat-vásárokat szervezünk, ahol az erre nyitott vállalatok bemutatják az általuk felkínált gyakorlat-lehetőségeket. Ezen kívül a hallgatók az Erasmus szakmai gyakorlatok keretében Európa számos országában fejleszthetik gyakorlati készségeiket.

Részesei vagyunk két CEEPUS hálózatnak és Európa számos egyetemével van Erasmus kapcsolatunk. A magyar hallgatók ezen kívül rendszeresen részt vesznek magyarországi részképzéseken, nyári egyetemeken is.

6. Gondok az informatikusképzésben

A folyamatos fejlődés ellenére számos nehézséggel is küzdünk a kolozsvári magyar informatika-oktatásban.

Hosszú évekig a felvételi vizsga hiánya nehezítette a tanárok munkáját, hiszen így számos olyan hallgató is felvételt nyert, aki magas érettségi jeggyel és középiskolás átlaggal rendelkezett, ám humán vagy társadalomtudományi osztályba járt, így komoly lemaradásai voltak a matematika és informatika terén. Ennek volt köszönhető a viszonylag nagy lemorzsolódás az 1. és 2. tanév végére. Ez a probléma megoldódni látszik a felvételi vizsga visszavezetésével, lényegesen kevesebb hallgató hagyta abba tanulmányait a jelenlegi elsőévesek közül.

Folyamatos problémát jelent az is, hogy a hallgatók egy része már az egyetemi évek alatt elkezd a szakmában dolgozni és jóval kevesebb ideje, energiája marad a tanulmányokra. Sajnos sok cég nem követeli meg az egyetemi diploma megszerzését, így nem motiváltak arra, hogy egyetemi képzésüket befejezzék. Ez a helyzet hatványozottan igaz a mesterképzésre. A mesteris hallgatók szinte kivétel nélkül dolgoznak, így nehézségekbe ütközik az órarend elkészítése és a követelmények következetes számonkérése is. A cégek, vállalatok nem biztatják őket a mesterképzés elvégzésére, így csak a személyes motivációjukra számíthatunk. Sajnos országos szinten probléma, hogy nincs törvényi szabályozás arra, mely munkakörök betöltéséhez van szükség mesteri diplomára, így nehéz a

mellett érvelni a hallgatóknak, hogy a saját érdekükben kellene egyetemi tanulmányaikat a mesterképzéssel együtt befejezniük.

Hasonló nehézségekbe ütközünk akkor, amikor fiatal kollegákat szeretnénk bevonni az oktatásba. A doktori fokozat megszerzése után legtöbbször néhány év tanítás után elcsábulnak, és programozóként vállalnak munkát, hiszen a tanári fizetés sok nagyságrenddel kisebb annál, mint amit egy-egy cég fizet nekik. Ha hivatásuknak érzik is a tanítást, amikor szembesülnek egy család eltartásának költségeivel, arra kényszerülnek, hogy a sokkal jobban fizetett állást válasszák.

Irodalomjegyzék

- [1] *Academic Ranking of World Universities in Mathematics – 2013*, <http://www.shanghairanking.com/SubjectMathematics2013.html>
- [2] BALÁZS Márton, SZENKOVITS Ferenc: Az erdélyi magyar matematikusok, csillagászok és informatikusok tudományos munkássága az 1945–1990 időszakban. *Műszaki Szemle*, 37, 2007, 22–37.
- [3] GAAL György: *Egyetem a Farkas utcában*. Erdélyi Magyar Műszaki Tudományos Társaság, Kolozsvár, 2001
- [4] KÁSA Zoltán, ROBU Judit: Informatikusképzés magyarul Erdélyben, *Informatika a felsőoktatásban '2005, Konferenciakiadvány'*, Pethő Attila, Herdon Miklos (szerk.), ISBN 963-472-909-6, 264–268.
- [5] KÁSA Zoltán, ROBU Judit, VARGA Ibolya: *Az informatika oktatásának múltja és jelene a kolozsvári egyetemen*, Informatika a felsőoktatásban '96 és Networkshop '96 közös rendezvény, Debrecen, 1996. augusztus 27-30.
- [6] KOLUMBÁN József: A kolozsvári matematikai iskola kialakulása, Cseke Péter, Hauer Melinda(szerk): *125 éves a kolozsvári egyetem*, KOMP-PRESS Kiadó, 1999, 65-76.
- [7] KOLUMBÁN József: Matematikai tanszékek a Ferenc József Tudományegyetemen 1940 és 1945 között, Batiz Enikő, Nagy László, Soós Anna (szerk.): *140 éves a kolozsvári magyar nyelvű egyetemi oktatás/140 de ani de învățământ universitar în limba maghiară*, Presa Universitară Clujeană/Kolozsvári Egyetemi Kiadó, 2013, 205-214.

GRAFIKUS PROGRAMOZÁS A MÉRNÖK INFORMATIKUS KÉPZÉSBEN

Graphical programming in the information technologist engineer's education

Schiffer Ádám^a, Sári Zoltán^a, Várady Géza^a, Jancskár Ildikó^a

^aPécsi Tudományegyetem, Pollack Mihály Műszaki és Informatikai Kar, Műszaki Informatika Tanszék
schiffer.adam@pmmik.pte.hu

Absztrakt: A bolognai folyamat sztenderdjeinek, irányelveinek bevezetése után a műszaki képzésben fontos szempont az egyes kutatóhelyek, egyetemek specifikus képzésének, egyedi tudásbázisának megőrzése. Karunkon a mérnök informatikus oktatásban immáron 18 éve használjuk a National Instruments LABVIEW grafikus fejlesztőkörnyezetét, melynek segítségével mérnöki problémák megoldására koncentrálhatnak a hallgatók a programozás árnyalatainak tanulása helyett. Mivel a LABVIEW szignifikánsan leegyszerűsíti és meggyorsítja a kurzusok projektjeinek megoldását, így már minimális programozási tudás mellett hatékonyan használható az alapképzésben. Az NI LABVIEW bevezetését megvizsgáljuk költséghatékonyság, fejlesztési idő, hardver kompatibilitás szempontjából. A szoftver környezet hatékony kiegészítője az NI ELVIS II laboratóriumi munkaállomás, melynek moduljait az BSc, MSc oktatásban, önálló projektmunkában használjuk. A karunkon országosan elsőként megnyitott NI akadémia, a projektek, a hallgatók és az oktatók felkészültsége számos sikert hozott a szak számára.

Kulcsszavak: LabView, projekt, mérnök informatikus

Abstract: After the introduction of the standards of the Bologna process it is a main viewpoint to keep specific qualification and knowledge in the university research centers. In our faculty we used the LABVIEW graphical development environment in the education since eighteen years, which allows to solve engineering problems rather than focus on learning the nuances of programming. As the LABVIEW significantly simplifies and speeds up the solution of the course projects it can be used very effectively in the undergraduate training with minimal programming knowledge. The application of the NI LABVIEW will be analyzed in terms of cost efficiency, development time and hardware compatibility. The NI ELVIS II laboratory workstation is an effective addition of the software environment, which modules in the BSc, MSc education and in the independent projectworks are used. Our country first opened National Instruments Academy, the projects, and the know-how of our educators and students brought many success for our department.

Keywords: Labview, project, information technologist engineer

1. Bevezetés

A grafikus programnyelvek, mint a National Instruments LABVIEW népszerűsége töretlenül növekszik, egyre több egyetem helyez hangsúlyt használatára világszerte mind az oktatásban, kutatásban, mérnöki fejlesztésben. A mérnök informatikus képzésben a grafikus fejlesztői környezetek mint a LABVIEW vagy a SIMULINK a kiegészítő moduljaikkal nagyfokú szabadságot biztosítanak a különböző szintű és típusú egyetemi kurzusok összeállításában.

Kutatócsoportunk Pécsi Tudományegyetem Pollack Mihály Műszaki és Informatikai Karának Műszaki Informatika Tanszékén tevékenykedik. Tanszékünk autonóm rendszerekkel, szabályozástechnikával, ipari automatizálással, mérésadatgyűjtéssel, képfeldolgozással

foglalkozik, melyhez immáron 18 éve használjuk a LABVIEW-t, mint grafikus fejlesztőkörnyezetet.

A virtuális műszerek ideálisak a valós hardverek szoftveres implementálására, és kombinálva az adatfolyam hálózatok intuitív természetével az oktatóknak sokkal gazdagabb lehetőséget kínál a gyakorlati szemináriumok kidolgozására. Az előzőekre példaként a hallgatók képesek már az első pár alkalommal olyan applikációkat fejleszteni, melyek nem igényelnek különös programozás elméleti előképzettséget. Ezek segítségével a gyakorlat során nem a programozási paradigmákkal, elmélettel kell foglalkozni, hanem a mérnöki problémák megoldására koncentrálhatnak a hallgatók.

2. Népszerű grafikus programnyelvek mérnök informatikusoknak

A grafikus programozás népszerűvé válása az ipari alkalmazásokban hamar megjelent, egyre több cég fejleszt grafikus alkalmazást a hardvereihez, s ehhez igazodva az egyetemi képzésbe is bekerültek világszerte. Tanszékünkön két grafikus programnyelv bevezetését látjuk célszerűnek a mérnök informatikus képzésben:

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) ipari szabvány, 1986 óta van a piacon, s amelyet elsősorban mérési, szabályozási, illetve tesztelési feladatokra fejlesztettek. Mivel a LABVIEW segítségével szignifikánsan lerövidíthetőek az egyes programozási feladatok, ezért az oktatók széleskörűen használják az általános iskolai (!) oktatástól kezdve a PhD képzésig. Használható laboratóriumi feladatokhoz, ipari alkalmazásokhoz, házi feladatokhoz, szimulációs problémákra, mérnöki feladatok megoldásához [1].

SIMULINK, amely a MathWorks által lett kifejlesztve, s mely segítségével összetett dinamikus rendszereket modellezhetünk, szabályozhatunk nagyon egyszerűen. A SIMULINK nagyon felhasználóbarát, könnyen kezelhető programcsomag, melynek részei blokkokból épülnek fel. Komplexitása miatt inkább felsőfokú oktatási intézményben használatos. Műszaki informatikai képzésben a jelfeldolgozásban, rendszermodellezésnél használható, differenciál egyenletekkel vagy állapotváltozókkal leírt, visszacsatolt rendszerek modellezésénél [2].

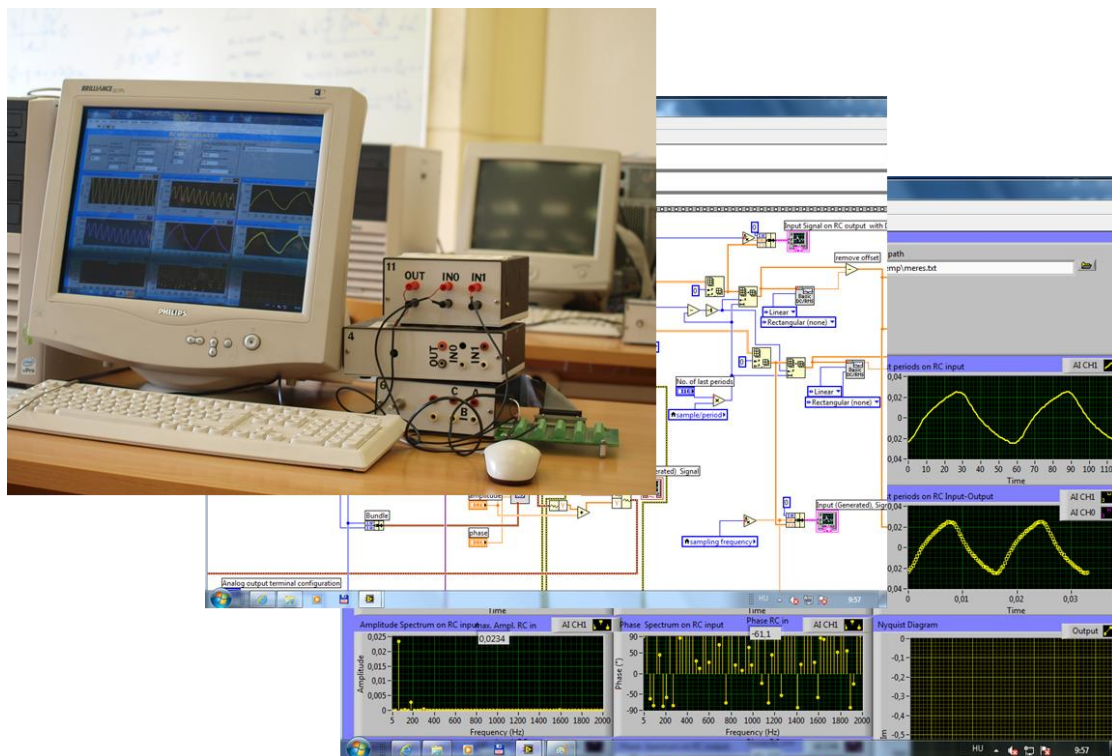
3. A LABVIEW bevezetésének vizsgálata

A mérnök informatikus autonóm szakirányú képzésünkben az oktatás és a kutatás során mind a hardverre, mind a szoftverre szükség van. Úgy gondoljuk, a LABVIEW nagyon hatékony eszköz a mérnöki oktatásban, így vizsgáljuk meg a szoftverkörnyezet bevezetését költséghatékonyság, fejlesztési idő, hardver kompatibilitás szempontjából.

Költségek

A szoftver használatának hallgatói oldalát nézve az NI számukra viszonylag olcsó LABVIEW verziót kínál, mely tudását tekintve megegyezik a LABVIEW Full Development System-el (egyetlen különbség a vízjel, mely mind a szoftver interfészben, mind a nyomtatásban megjelenik). Együtt használva egy alacsony költségű NI DAQ hardverrel, vagy esetleg az NI

ELVIS II műszertáblával, széles spektrumú gyakorlati lehetőséget kínál az oktatásban, melyet a képzésünkben számos tárgynál használni tudunk. Elkezdve az egyszerű jelek monitorozásától, megjelenítésétől, mérésétől egészen a szabályozási rendszerekig. Mérésadatgyűjtés mérései során például általunk gyártott egyszerű analóg modellek ki- és bemeneteit mérik a hallgatók LABVIEW segítségével, illetve az eredményeket feldolgozzák (RC tag, hiszterézis karakterisztika, ...stb), s mellyel igazolható a jelfeldolgozáson tárgyalt és számított modellek viselkedése. A mérésadatgyűjtés tárgy egy mérőállomását az 1. ábra mutatja.



1. ábra. A mérésadatgyűjtés egy munkaállomása, illetve a hallgatók által kidolgozandó szoftver

Fejlesztési idő

A fejlesztési idő fontos szerepet játszik egy egyetemi gyakorlat hatékonyságában, és nincs kétség, a LABVIEW rendkívül lerövidíti egy mérnöki programozási feladat elvégzését. Ezen felül rendelkezésre állnak olyan szoftverek eszközök, mint az Express VI-ok, mérési segédprogramok, amelyek még gyorsabbá teszik a fejlesztést. Például mérési feladatoknál a *Spectral Measurement* VI olyan bemenetekkel ellátott, ahol egyszerűen kiválasztható a mérési típus (Erősítés– RMS, Erősítés – peak, ...stb), az eredmények ábrázolása (lineáris vagy dB), FFT ablakfüggvények, átlagolási és fázis paraméterek. Ezek beállítása, az eredmények ábrázolása, hardver eszközök kezelése egyszerűen, gyorsan történik. A jelfeldolgozásra szintén jelentős VI készlet áll rendelkezésre (torzítás, FFT és frekvencia analízis, időtartomány vizsgálat, IIR és FIR szűrők, ...stb). A hálózati adatátviteli szabványos protokollok lehetőséget adnak a kommunikációra (EtherCAT, UDP, TCP/IP, VISA Serial, Web ...stb). Továbbá a LABVIEW számos példaprogramot kínál minden feladathoz, így új applikációk fejlesztésénél ezt kiindulásként a hallgatók használni tudják, s ezzel a fejlesztési idő szintén lecsökken.

Hardver kompatibilitás

A mérnök informatikusok oktatásában általunk használt NI és harmadik féltől származó hardvereket a LABVIEW meghajtóprogramjai teljes egészében lefedik. Az NI támogatásában a saját eszközeik meghajtóprogramjain felül több, mint 10000 egyéb hardver támogatása biztosított. Ezen felül lehetőség van saját meghajtóprogram elkészítésére az NI VISA segítségével, amely egy standard API (Application Programming Interface). Mivel a hardvereszközök elérése eleve biztosított, így a GPIB, soros, USB, Ethernet, VXI interfészek elérése pár egyszerű VI megtanulásával biztosított.

Kiegészítő szoftver és hardver modulok

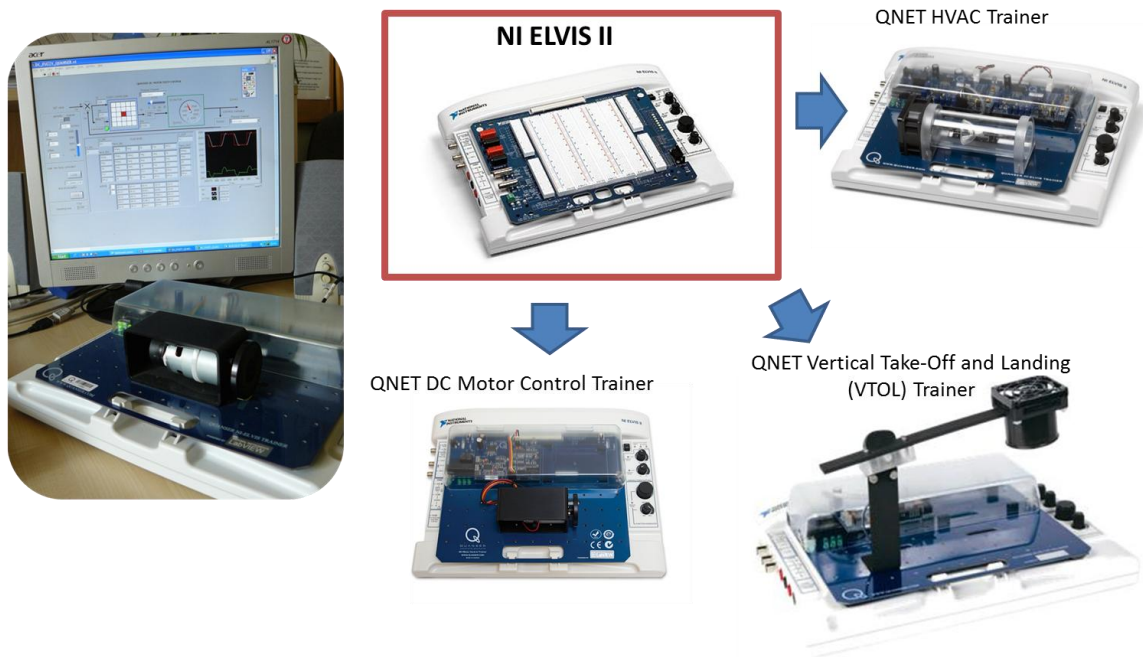
A LABVIEW kiegészítő modulok használata az oktatásban szintén indokolt, mivel ezek segítségével specifikus feladatokat adhatunk a hallgatóink számára. A beágyazott rendszerek, képfeldolgozás, szabályozási témakörökhez 25 modul áll rendelkezésre. Az NI ELVIS II segítségével megvalósítható az a koncepció, hogy különböző oktatási feladatokat, módszereket több félével keresztül is ugyan azon, a hallgatók által ismert hardver platformon végezzék. Ez csökkenti a költséget, illetve a beszerzendő eszközök mennyiségét. A rugalmasan konfigurálható NI ELVIS II (NI Educational Laboratory Virtual Instrumentation Suite) műszertáblát, illetve kiegészítő moduljait számos oktatási projektben használja tanszékünk. Az NI ELVIS II számtalan laboratóriumi eszközt egyesít: oszcilloszkópot, multimétert, jelgenerátort, AC tápot, valós idejű spektrum analizátort, ... stb. Az oktatás során az alábbi NI ELVIS II kiegészítő modulokat használjuk, melyek a 2. ábrán láthatóak:

- QNET DC motor trainer: ennek segítségével bemutatható a DC motor szabályozásának alapjai, miközben megfoghatóvá és érthetővé teszi azt a hallgatók számára. Bemutatható rajta a PID szabályozás alapjai, rendszeridentifikáció, sebesség és pozíció szabályozás.
- QNET HVAC Trainer Board: a PID szabályozás elsajátítása után a hallgatóknak ezen modullal lehetőségük van termodinamikai szabályozó algoritmusok tesztelésére. Itt alkalmuk nyílik hőmérséklet vezérlésre, paraméter identifikációra, rendszer modellezésre
- QNET Vertical Take-off Landing Trainer: irányítástechnikai és mechatronikai oktatáshoz használható egy szabadságfokú helikopter, mely segítségével a hallgatókat bevezetjük a mozgás vezérlésbe, dinamikába, kinematikába.

4. Grafikus programozás a PTE PMMIK mérnök informatikus képzésben

Az előzőekből is látható, szakunkon nagy hangsúlyt fektetünk a LABVIEW-n alapuló grafikus programozásra, amelynek előnyeit ismertettük. Mivel a LABVIEW hallgatói verziója rendelkezik a legtöbb kiegészítő modullal is, így számos tárgyba, megfelelő egymásra épüléssel alkalmazható. A rendszerelmélet kapcsán egyszerűbb rendszermodelleket állítatunk elő hallgatóinkkal, majd a jelfeldolgozás témakörökben a jelek idő- és frekvenciatartománybeli viselkedését tanulmányoztatjuk. Erre épülő képfeldolgozás, gépi látás kapcsán szoftveresen a VISION modullal foglalkozunk. A BSc képzés végén, illetve az MSc képzésben az önálló projektmunkák, órai feladatok esetén a képfeldolgozás hardveres hátterét NI SMART kamera, illetve VarioSCAN hőkamera adja. Szabályozástechnika, intelligens irányítások kapcsán a DSC, illetve real-time modulokat használunk, melyhez NI CompactRIO, PXI valós idejű rendszerek adják a hardveres hátteret. Az autonóm rendszerek tárgyon belül részben az AR.Drone 2.0 quadkopterek autonóm viselkedését oktatjuk, melyben

a hallgatók projekt munkájában felhasználják a képfeldolgozásban, grafikus programozásban, szabályozástechnikában, jelfeldolgozásban szerzett tudásokat [3][4][5].



2. ábra. NI ELVIS II és a szakunk által használt kiegészítő modulok a diszciplináris oktatás segítéséhez

A hallgatói aktivitást jellemzi, hogy a 2013 évben Müller Péter MSc-s hallgatónk megnyerte a National Instruments országos pályázatát. Az általa kidolgozott szemmozgáskövető rendszer fejlesztését tanszékünkön végezte, valós idejű NI Smart kamerával, illetve LABVIEW szoftverkörnyezetben. A mérési rendszer sematikus ábrája, illetve a mérőkör működés közben a 3. ábrán látható.

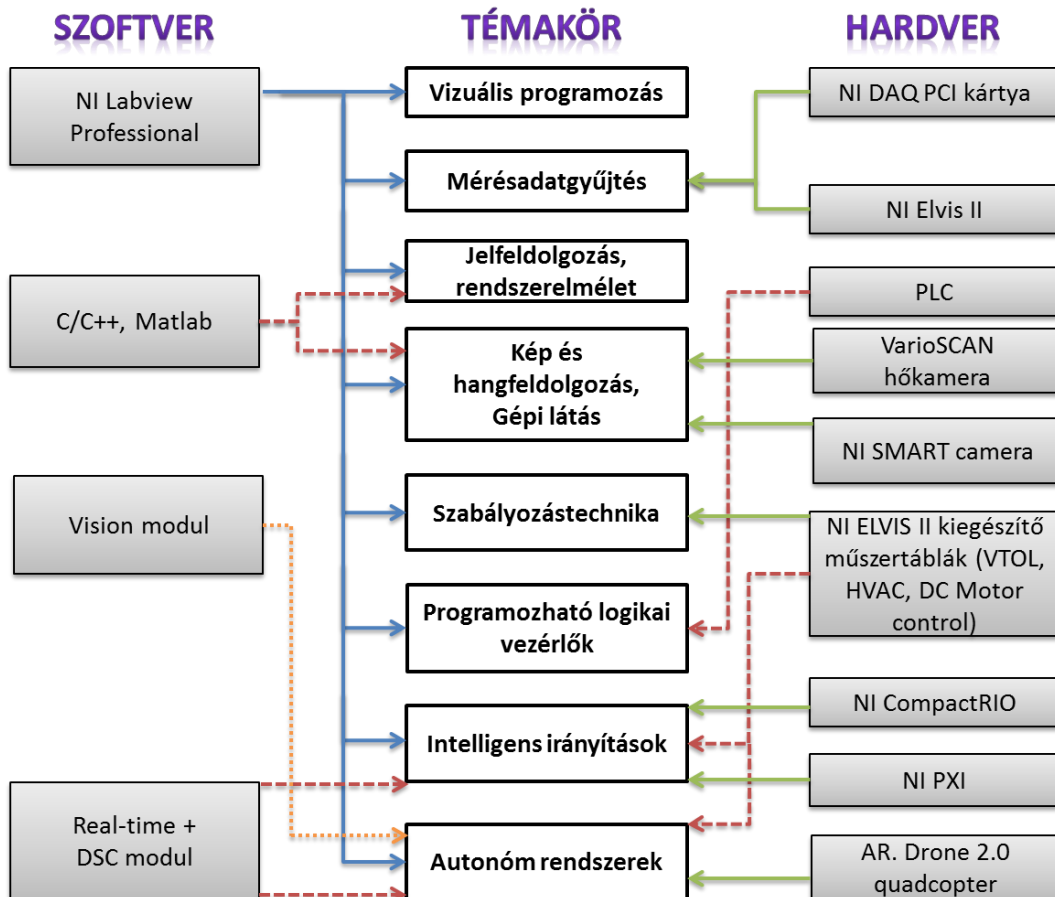


3. ábra. A 2014-es NI pályázaton nyertes valós idejű szemmozgáskövető rendszer

A 4. ábra a mérnök informatikus képzésben bevezetett grafikus programozás alkalmazását, illetve az ehhez kapcsolódó hardvereket mutatja be az egyes oktatott témakörökhöz. Látszik,

hogy a felsőoktatásban és az iparban elvárt C/C++ oktatással párhuzamosan viszonylag hamar, a második szemeszterben, külön tárgyon belül elkezdjük a LABVIEW bevezetését. Az ábra szemléletesen mutatja az oktatott témakör– szoftver–hardver használatát szakunkon.

A National Instruments átfogó képzési programját, az NI Akadémiát az országban elsőként, 2011. Január elsején hivatalosan bevezettük. Azóta a hallgatók ingyenesen NI CLAD vizsgát tehetnek intézményünkben. A program keretén belül minden, a tárgyat felvevő hallgató kap egy teljes NI LABVIEW Student Edition licenst.



4. ábra. A LABVIEW alkalmazása mérnök informatikus szakon, különböző oktatási területeken

5. Összegzés

Ebben a cikkben a szerzők tapasztalatait próbáltuk meg összefoglalni a grafikus programozással kapcsolatban. Az NI LABVIEW 1996-os bevezetés mindenképpen sikernek mondható a szakunkon, melynek eredménye az utóbbi pár évben jelentkezik. Számos hallgatónk talál állást a megfelelő szabályozástechnikai, irányítástechnikai ismeretei miatt.

A projekt a TÁMOP-4.2.2.C-11/1/KONV-2012-0005 számú "Jól-lét az információs társadalomban" című pályázat III./6. munkacsoport támogatásával valósult meg.

Irodalomjegyzék

- [1] LabVIEW is a trademark of National Instruments Corporation
- [2] SIMULINK is a trademark of MathWorks Inc.
- [3] Ildiko Jancsakar, Zoltan Sari, Adam Schiffer, Geza Varady, “Soft control of a 1-degree-of-freedom helicopter”, in *Architectural, Engineering and Information Sciences - 9th International PhD& DLA Symposium: Abstracts Book*, University of Pécs Pollack Mihály Faculty of Engineering, 2013, pp. 164-164.
- [4] QNET HVAC, VTOL are trademarks of Quanser Inc.
- [5] Quanser Inc., QNET Practical Control Guide. 2009.

Középiskolai informatikai tehetséggondozás az ukrain magyar tannyelvű iskolákban

Szimkovics Tamás

Ungvári 10. Számú Dayka Gábor Magyar Tannyelvű Középiskola

sztommy18@gmail.com

Absztrakt: Kárpátalján 99 magyar tannyelvű iskola található. Az ezek között fennálló interakció elmélyítéséhez és kibővítéséhez jött létre az anyaországi alapítvány példájára a „Genius” Jótékonyági Alapítvány, amely feladatának tekinti, hogy a magyar nyelvű középfokú oktatást és a felsőoktatást összekapcsolja, áthidalja. Az alapítvány létrehozását két intézmény valósította meg: a Kárpátaljai Magyar Pedagógusszövetség és a II. Rákóczi Ferenc Kárpátaljai Magyar Főiskola. Az alapítvány többek között képzéseket szervez, amelyek a Kárpátalján található magyar tannyelvű iskolákban zajlanak több mint 2000 diák és 300 oktató részvételével. A tehetséggondozás során több tantárgyat is oktatnak, humán és reál tárgyakat egyaránt. Ezekből majd a tantárgyi vetélkedőkön mérhetik össze tudásukat a diákok.

Én az informatikai képzéseket tanulmányozom. Az informatikai képzéseket fontosnak tartom, mivel a középiskolai informatikaoktatás sokszor nem biztosít megfelelő alapokat a felsőoktatási képzéshez. Az iskolai állami tanterv a programozást és weblapszerkesztést csak felületesen érinti, ráadásul azt is csak a 11. osztályban. A „Genius” Jótékonyági Alapítvány képzése során a gyerekek már 5-6. osztályban megismerkedhetnek a programozás alapjaival a Logo nyelv segítségével. A weblapszerkesztésbe is betekintést nyernek a 10. osztályos tanulók. Akik már rendelkeznek ilyen irányú előismeretekkel, kibővíthetik, elmélyíthetik azokat. Az alapítvány figyelemmel kíséri a tehetséges diákokat, és segíti őket a felsőoktatásban is különböző szakkollégiumok és konferenciák szervezésével. Az alapítvány munkásságán kívül a II. Rákóczi Ferenc Kárpátaljai Magyar Főiskola egy most induló szakképzését is be szeretném mutatni.

A „Genius” Jótékonyági Alapítvány irodavezetőjével és informatika tantárgyfelelőseivel, valamint az informatikai tantárgyi vetélkedők főszervezőjével készítettem interjút, hogy bemutassam a Kárpátalján zajló informatikai tehetséggondozás szervezési és oktatási módszereit.

Kulcsszavak: tehetséggondozás, informatikai képzések, informatikai vetélkedők.

1. Bevezetés

Kárpátalján összesen 672 iskola található, amiből 99 magyar tannyelvű. A megye székhelye a 114 ezer lakosú Ungvár, ahol is egy magyar középiskola és egy magyar gimnázium található.

Az ukrain állami tanterv szerint az informatikát 5–11-ik osztályban oktatják, egy éve viszont már a második osztálytól kezdődik az informatika oktatása. Az informatikaoktatás azért fontos a diákok számára, hogy kellőképpen be tudjanak illeszkedni az információs társadalomba és kultúrába. Az informatika megfelelő színvonalú oktatásához sokszor több feltétel is hiányzik:

- Az oktatóknak nincs meg a megfelelő képzettsége, sokszor nem tanult informatikus helyettesíti őket.
- A technikai eszközök elavultak. A legtöbb számítógépet és technikai eszközt az állam 2007-ben vásárolta az iskolák számára, ráadásul azok már akkor sem voltak épp minőséges termékek.

- Az ukrán iskolai állami tantervben több fontos témakör nincs megfelelően hangsúlyozva
- A diákok alulmotiváltak

A fenti okok teszik szükségessé és elengedhetlené Kárpátalján az informatikaképzések és versenyek rendezését a diákok számára. Ezzel többek között motiválhatják a gyerekeket az informatikatanulására.

2. A „Genius” Jótékonysági Alapítvány képzései

Kárpátalján magyar nyelven informatikai képzéseket és vetélkedőket a Kárpátaljai Magyar Pedagógusszövetség (KMPSz) és a „Genius” Jótékonysági Alapítvány (GJA) rendez. A tehetségek felkutatása és felkarolása a 2011-ben alakult „Genius” Jótékonysági Alapítvány legfőbb profilja. Az alapítvány a KMPSz és a II. Rákóczi Ferenc Kárpátaljai Magyar Főiskola (II. RFKMF) közreműködésével jött létre, és a Nemzeti Tehetségsegítő Tanács akkreditálta. A GJA anyagi támogatását a Bethlen Gábor Alap és az Emberi Erőforrások Minisztériuma biztosítja. Az alapítványra azért is nagy szükség volt, mivel megalakulásáig nem volt a közép- és felsőoktatást áthidaló egyetlen intézmény sem. Számos elért eredményét jól példázza, hogy 2013-ban megkapta a Magyar Termék Nagydíjat külföldi szervezetek számára kiírt kategóriában [5].

Az alapítvány főbb tevékenységei közé sorolandóak a középfokú oktatásban résztvevő diákok számára rendezett képzések. A képzésekkel kapcsolatos főbb kérdéseket a Kárpátaljai Tehetségsegítő Tanács tárgyalja. A foglalkozások Kárpátalja-szerte összesen 8 tehetségpontban zajlanak (Aknaszlatina, Tiszaújlak, Beregszász, Mezővári, Bátyú, Nagydobrony, Munkács, Ungvár). Ezekbe összesen 88 oktatási intézményből utaznak be a tanulók, hogy részt vehessenek a tanórákon. Utaztatásukat a GJA szervezi és finanszírozza. A nyolc tehetségpontot kiválasztásának szempontjai a következők voltak: földrajzi helyzet, szaktanárok száma, illetve, hogy mely intézményekkel sikerült korábban kapcsolatot kialakítani. A földrajzi szempont szerint igyekeztek úgy kiválasztani az intézményeket, hogy lefedjék Kárpátalja magyarlakta településeit (*1. ábra*).

A foglalkozások középiskolák épületeiben vannak levezetve, kivéve a beregszászi tehetségpontot, mivel ott a túljelentkezés miatt egy hasonló intézmény sem tudta volna befogadni a jelentkezőket. Így a II. RFKMF támogatásával a tehetségpont oktatói a főiskola termeit és technikai eszközeit vehették igénybe a képzések levezetéséhez. A képzéseken hétvégeként zajlik az oktatás, a tehetségsegítő tanács szakmai bizottsága által kidolgozott egységes tanterv alapján. A tehetségpontokban minden alkalommal, több mint 2000 tanuló, 300 tanár és majdnem 90 magyar tannyelvű iskola vesz részt [1]. A képzési pontokban fontos feltétel a megfelelő számítástechnikai eszközök biztosítása, így az informatikafoglalkozások is megfelelő támogatottsággal rendelkeznek. Továbbá a jótékonysági alapítvány technikai eszközökkel is segíti a tehetségpontokat. A 2013-2014 tanévben három az egyben funkciójú nyomtatókat kaptak azok a középiskolák, amelyekben a képzések voltak levezetve, a következő tanévben pedig laptopot és vetítőt is kapnak, amelyet természetesen szabadon felhasználhatnak az iskolai oktatás során is.

A tehetségpontoknál a diákokat három korcsoportra osztják a képzéseken: 5-6, 7-8, 9-11 osztályos. Egy-egy képzés 75 percig tart.



Jelmagyarázat
● - járásközpontok
● - tehetségpontok

1. ábra. Kárpátalján található „Genius” tehetségpontok (eredeti kép URL címe: http://lehetvelem.blog.hu/2014/01/31/ha_torik_ha_szakad_309 h. i.: 14.06.25.)



2. ábra. Informatikai képzés a Beregszászi „Genius” tehetségpontban (az eredeti kép:URL címe: <http://www.genius-ja.uz.ua/keptar/2013-genius-tehetsegpont-2013-november-23-beregszasz> h. i.: 14.06.25.)

Egy-egy szombati napon három foglalkozás kerül levezetésre csoportonként a tehetségpontokban. A második foglalkozás után ebédszünet következik, amelyen a diákok ingyenesen étkezhetnek. A képzések tantárgyait rotációsan változtatják hétvégénként, váltogatva a humán és reál tárgyakat. Egy korosztálynak egy tantárgyat három foglalkozáson keresztül oktatnak. Tehát informatikából összesen 9 foglalkozást vezetnek le egy tehetségpontban. Az egységes képzési tantervet a tantárgyfelelősök állítják össze.

Informatikából a tantárgyfelelős Beregszászi István tanár úr, aki a II. Rákóczi Ferenc Kárpátaljai Magyar Főiskolán oktat, többek között informatikai tantárgy-pedagógiát, informatikai alkalmazói ismereteket és programozói ismereteket is. A képzési tantervet egyedi minta alapján állította össze, bevonva az ECDL-oktatási minták alkalmazói ismereteit és az ukrainai, magyarországi oktatási tantervek programozói ismereteit. A tanterven változtattak minden évben, úgy, hogy azok jobban illeszkedjenek az oktatásban résztvevő diákok szintjéhez és a modern oktatási trendekhez. Legutoljára az egységes tanterv összeállítása, összeegyeztetése az oktatásban résztvevő oktatók bevonásával történt, a feladatok kidolgozásának részleges átruházásával. Így a kilenc képzési foglalkozás anyagát négy oktató állította össze: Beregszászi István, Bátyi András, Horváth Zoltán és Jakab Géza tanár urak.

Az 5-6. osztályosok a *Windows* operációs rendszer használatával, a *Paint* rajzprogrammal és a *Logo* programozási nyelvvel foglalkoztak a legutóbbi GJA informatikafoglalkozásokon. A 7-8. osztályosoknál a szövegszerkesztéssel és táblázatkezeléssel kapcsolatos ismeretek bővítése volt a cél. A 9-11. osztályosoknál a *HTML* nyelvvel, valamint az *Excel* és a *Pascal* szoftverekkel folyt a munka. A tantervek kidolgozásánál gondoltak a képességek szerinti csoportbontásra, ezért minden képzéshez alap- és emeltszinten készítették el a tantervet. A csoportbontásra szükség is volt, egyrészt a differenciáloktatás elősegítése, másrészt az előismeretek nagy különbsége okán. Például a képzésen előadott *Logo* programozást nem oktatják minden iskolában. Így a diákok egy részének a *Logo* szoftver alapjaival, a program felépítésével és a fontosabb parancsokkal kellett megismerkedniük. Viszont akik járatosabbak voltak a technógrafikában, azok összetettebb feladatokkal dolgozhattak. Nagy különbségek mutatkoztak a *HTML* és a *Pascal* nyelv ismereteiben is: sokszor az alapok is hiányoztak. Így szintén az egyszerűbb, pársoros kódoktól a keresési és rendezési algoritmusokat használó, fájlkezelő programokig különböző bonyolultságú programok létrehozását sajátíthatták el a diákok. Így a gyengébb és a jobb előismeretekkel rendelkező tanulók is részesülhettek sikerélményben a képzések során. További motiváció a diákoknak, hogy a képzések a KMPSz által rendezett következő félévi tantárgyi vetélkedőre is felkészítik őket, amelyen a kimagasló eredményeket elért diákok értékes jutalmakban részesülhetnek.

Perspektívák terén a „Genius” Jótékonyági Alapítvány jelenlegi célkitűzése a középfokú oktatással kapcsolatban, hogy magyarországi vetélkedőkre juttassák ki a Kárpátalján tanuló gyerekeket. Ezáltal egy nagyobb, erősebb mezőnyben mérhetik össze tudásukat a diákok.

3. Visszacsatolás, informatikai tantárgyi vetélkedő

Idén immár ötödik alkalommal került sor informatikai vetélkedőre Beregszászban, minden eddiginél több diák részvételével. A KMPSz és GJA által rendezett tantárgyi vetélkedőre esélyegyenlőségi szempontok terén nagy szükség van, hiszen az ukrán nyelven lebonyolított tantárgyi vetélkedőkön egy magyar ajkú diáknak nehezebb boldogulnia. A vetélkedőn a középiskolák 5-11. osztályos, a gimnáziumok 1-7. évfolyamos diákjai és líceumok tanulói

vehettek részt. Az informatikai vetélkedők 2010-től folynak. Kezdetben csak programozói kategóriában nyolcan jelentkeztek, ám amint megjelent a felhasználói kategória is, ez a szám évről-évre növekedett:



3. ábra az informatika tantárgyi vetélkedőre jelentkezők száma

A diákok elhelyezése nem volt egyszerű feladat, mivel 114-en jelentkeztek. Több intézmény összefogásának köszönhetően sikerült ekkora mennyiségben *Windows* operációs rendszert futtató számítógépeket előkészíteni a vetélkedőre. A II. RFKMF, a Kölcsény Ferenc Szakkollégium és a Beregszászi Magyar Gimnázium számítógéptermeire is szükség volt (4. ábra).



4. ábra. Az informatikavetélkedő látható négy helyszínen, a jobb alsó képen a szakkollégium, a többi képen a főiskola termei láthatóak. (az eredeti képek: URL címe: <http://www.genius-ja.uz.ua/keptar/2014-kmpsz-genius-informatika-vetelkedo-2014-februar-8> h. i.: 14.06.25.)

Mint látható, a verseny lebonyolításához használt számítástechnikai eszközök jelentősen különböztek egymástól. Így a feladatokat is úgy kellett megszerkeszteni, hogy azokat megfelelőképpen, az összes számítógépen meg lehessen oldani. Korosztályok szerint négy csoportra bontották a felhasználói kategóriában résztvevő diákokat: 5-6., 7-8., 9-10. és 11. osztályosra, a programozási kategóriában pedig 9-10. és 11. osztályosra. A versenyen egységesen három óra volt adva a feladatok megoldására.

A vetélkedőn az összetett szövegszerkesztési feladatok: körlevél, tabulátorok, táblázatok megfelelő használata ment nehezen. A táblázatkezelésnél pedig a logikai gondolkodást igénylő feladatok megoldása. Programozói kategóriából kevés a jelentkező, itt a gimnáziumok vannak többségben. Sajnos kevés középiskolában megy megfelelő színvonalú programozás oktatás.

A dobogós helyezést elért diákok oklevélben részesültek és ingyenesen részt vehetnek a KMPSz és GJA által rendezet nyári táborokban. A diákok egyéb ösztöndíjakra is pályázhattak, amelyeket a vetélkedőkön elért eredményektől függően ítéltek oda.

4. Szakképzés a II. Rákóczi Ferenc Kárpátaljai Magyar Főiskolán

A Kárpátalján található magyar nyelvű informatikaoktatással kapcsolatban fontos megemlíteni a II. Rákóczi Ferenc Kárpátaljai Magyar Főiskola által nyújtott új képzési lehetőséget, az „ifjú szakembert”. Erre a képzésre 2014-2015-ös tanévtől lehet jelentkezni óvodapedagógia, szociálpedagógia, könyvvitel, alkalmazott matematika és turisztikai szolgáltatások szakirányokban. A képzésre a diákok már a 9. osztályt elvégezve jelentkezhetnek, a négy féléves képzés befejezése után pedig a főiskola második évfolyamára felvételizhetnek a megfelelő szakokra. A felvételihez szükséges a 9. osztály befejezése után kapott kis érettségi bizonyítvány és a felvételi vizsgák teljesítése. A vizsgákból a diákoknak kettőt kell teljesíteniük, egy ukrán nyelvű tollbamondást és egy a szakiránytól függő felvételi tárgyat. A választható szakirányok között az alkalmazott matematika az, ami érdekes az informatikaoktatás szempontjából. Az alkalmazott matematika szak felvételijéhez az ukrán tollbamondáson kívül szükséges a matematika felvételi vizsga. Az alkalmazott matematika szak tulajdonképpen programozók és rendszergazdák képzését takarja. Befejezése után a diákok a főiskolán is folytathatják tanulmányaikat matematika szakon, második évfolyamtól. [3]

A képzés nagy előnye amellet, hogy a főiskola második évfolyamától kezdhetik meg majd tanulmányaikat, az, hogy nem szükséges ukrán nyelvből emelt szintű érettségit tenniük a felvételizőknek. Ugyanis a magyar ajkú diákok között sokan megbuknak az ukrán nyelv érettségi vizsgán és Ők csak akkor felvételizhetnek, ha a szaktárgyi vizsgájuk legalább 170 pont az elérhető 200-ból.

5. Összefoglalás

Egyre több diák jelentkezik a képzésekre és vetélkedőkre, tehát van igény az informatika tanulására is. A tömeges rendezvények megszervezése költséges és sok munkával jár. Erre megoldás egy elektronikus oktatási rendszer beüzemeltetése, amely ezeket a terheket részben enyhíthetné. Erre gondolt is a GJA, ezért elindította az elektronikus online képzését a Debreceni Egyetem segítségével az Arany János Programon belül. A magyar nyelv,

matematika és angol tantárgyakon jelenleg 150 regisztrált tanuló van. A képzések során kimagasló eredményeket elérő diákok tárgyi jutalmakban részesülhetnek.

Az elektronikus oktatási képzéseket fontos lenne kiterjeszteni több tantárgyra és korosztályra is. Mély és alapos tudás elsajátítására ugyanis sokszor nem alkalmasak az alkalmanként rendezett képzések, ám azok a szükséges érdeklődést felkelthetik a gyerekekben a tanulás iránt. Ezért fontos, hogy a diákoknak lehetőségük legyen otthon is tanulni, önállóan. Egyes jobb képességű tanulók ezt teszik. De sajnos kevesekben van meg a motiváció és az önálló tanuláshoz való készség, hajlam. Az önálló tanulást nagyban elősegítené egy elektronikus oktatási rendszer, amelyben a diákok könnyen megtalálhatnák a szükséges tananyagokat. A legtöbb gyereknek van otthon számítógépe és internetkapcsolata, így a technikai eszközök első fele megvan az elektronikus oktatáshoz. Szükséges az elektronikus oktatási rendszer beüzemelése, illetve feltöltése magyar nyelvű, megfelelően strukturált oktatási anyaggal. Oktatási és oktatást segítő rendszerekből nagy a választék. Az ingyenes és sok funkcióval ellátott rendszerek közül talán a *Moodle* rendszere a legkézenfekvőbb és legelterjedtebb. A főiskola web-szerverén telepítve van egy *Moodle* elektronikus oktatási rendszer, amely megfelelne a diákok képzéséhez (<http://ml.kmf.uz.ua>). Fontos konfigurálni és oktatási tartalommal feltölteni a rendszert. Az informatika oktatáshoz szükséges tartalom elkészítése nagy munkát jelent, mivel informatikából magyar nyelven, ukrán állami tanterv mintájára, tudomásom szerint még nem készültek elektronikus oktatási anyagok, ezeken belül a programozást és algoritmikus gondolkodást fejlesztő tananyagokat hangsúlyoznám. A képzések és a vetélkedők eredményein egyértelműen látszanak ebbéli hiányosságok a diákoknál. A rendszer hatékony működéséhez fontos a létrehozott és feltöltött anyagokat mindenki számára könnyen elérhetővé tenni.

6. Köszönetnyilvánítás

Köszönettel tartozom az interjúalanyoknak, akik elláttak a szükséges információkkal a cikk írása közben, és megosztották velem tapasztalataikat. Ők: Dr. Váradi Natália, a „Genius” Jótékonyági Alapítvány irodavezetője, Beregszászi István tanár úr, a „Genius”- képzések informatika tantárgyfelelőse, Pallay Ferenc tanár úr, az informatikai tantárgyi vetélkedő főszervezője.

Irodalomjegyzék

- [1] Gál Adél: Civil szervezetek Kárpátalján: „Genius” Jótékonyági Alapítvány (2014).
<http://www.karpatalja.ma/karpatalja/oktatas/26000-civil-szervezetek-karpataljan-genius-jotekonyyasi-alapitvany>
- [2] Gál Adél: Informatika vetélkedő Beregszászban (2014).
<http://www.karpatalja.ma/karpatalja/oktatas/24945-informatikavetelkedo-beregszaszban>
- [3] Gál Adél: A II Rákóczi Ferenc Kárpátaljai Magyar Főiskola weboldalának egy hirdeteménye (2014).
<http://www.kmf.uz.ua/hun114/index.php/fiskolai-hirek/1630-uj-lehetsegek-a-karpataljai-magyar-felsoktatasban--molodsij-szpecialiszt.html>

- [4] A Kárpátaljai Magyar Pedagógusszövetség weboldala.
<http://kmpsz.uz.ua/>
- [5] A KMPSz cikke a Magyar Termék Nagydíj átadásáról a „Genius” jótékonyági alapítványnak (2013).
<http://kmpsz.uz.ua/hirek/magyar-termek-nagydi-jat-kapott-a-genius-jotekonysagi-alapitvany.html>
- [6] A „Genius” Jótékonyági Alapítvány weboldala.
<http://www.genius-ja.uz.ua/>

UNI@HAND – offline mobil alkalmazás az egyetem és a hallgatók támogatására

UNI@HAND – offline mobile application supporting the university and the students

Demetrovics Zsolt^b, Lőrincsik Éva^b, Márkus Zsolt László^a, Szkaliczki Tibor^a, Szántó György^a

^aMagyar Tudományos Akadémia Számítástechnikai és Automatizálási Kutatóintézete (MTA SZTAKI)

markus.zsolt@sztaki.mta.hu, szkaliczki.tibor@sztaki.mta.hu,
szanto.gyorgy@sztaki.mta.hu

^bEötvös Loránd Tudományegyetem Pedagógiai és Pszichológiai Kara

demetrovics.zsolt@ppk.elte.hu, lorincsik.eva@ppk.elte.hu

Absztrakt: A UNI@HAND egy többnyelvű komplex információs mobil szolgáltatás a hallgatók életével kapcsolatos naprakész információk (tanárok, tárgyak, események, stb.) kezelésére és megjelenítésére. Az alkalmazás az Eötvös Loránd Tudományegyetem Pedagógiai és Pszichológiai Karán érhető el. Az alkalmazás funkcionalitása a kar munkatársaival közösen került meghatározásra. A fejlesztést az MTA SZTAKI végezte, felhasználva a turisztikai célokra kifejlesztett GUIDE@HAND mobil alkalmazásban szerzett sokéves tapasztalatait. Az okostelefonok használatának gyors terjedése lehetővé teszi, hogy az alkalmazáshoz a hallgatók széles köre hozzáférjen. Az UNI@HAND az egyetemi élettel kapcsolatos számos hasznos információt tartalmaz: bemutatja az egyetem szervezeti egységeit, hallgatói önkormányzatát, oktatóit, a különböző hallgatói ügyek elintézésének menetét, a tantárgyakat, az egyetemi épületeket, egyéb hasznos helyeket és az aktuális eseményeket. Az egyetemi rendezvények és programok időpont és helyszín szerint kilistázhatók. A hallgatók a meghirdetett tantárgyak közül válogatva állíthatják össze a saját órarendjüket, amelyet táblázatos formában tekinthetnek meg. Az alkalmazás offline módon használható, tehát nincs szükség hozzá internetkapcsolatra. Webes adminisztrációs felület segíti az alkalmazás tartalommal való feltöltését, de lehetőség van a már meglévő webes egyetemi adatbázisokkal való összekapcsolásra is.

Kulcsszavak: offline, mobil alkalmazás, egyetem, hallgató

Abstract: UNI@HAND is a multilingual complex mobile information service for managing and displaying up-to-date information (teachers, courses, events, etc.) related to the life of students. The application is available at the Faculty of Education and Psychology of Eötvös Loránd University. The functionality of the application has been defined in cooperation with the staff of the faculty. Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI) developed the application by building on their experiences gained in the development of the GUIDE@HAND mobile tourist application in recent years. Many students have access to the application due to fast spreading of smart phones. UNI@HAND provides various useful information related to the university life: it introduces the organisational units, the student government, the staff, the academic affairs, the courses, the university buildings, further useful places and a list of current events. The events and programmes organised at the university can be displayed according to their location and time. The students can setup their own timetable by selecting the courses they visit from those organised by the university. The application can be used in an offline manner, i.e. it does not require an Internet connection. A Web administration interface helps in uploading content to the service but there is also an option to synchronise the application with the existing data bases of the web portals of the universities.

Keywords: offline, mobile application, university, student

1. Bevezetés

Az internet megjelenésével, majd elterjedésével eltűntek a nyomtatott tanrendek és egyetemi illetve kari tájékoztatók az egyetemi mindennapokból. Helyüket a honlapok illetve az elektronikus tanulmányi rendszerek vették át. Ezek gyorsan és könnyen elérhető információt szolgáltatnak, ráadásul a módosulások, korrekciók is pillanatok alatt keresztülvihetők ezeken az elektronikus rendszereken. A könnyű elérhetőség mellett azonban továbbra is szükségesek az internet, illetve ideálisan egy számítógép vagy notebook. Az utóbbit tekintve a telefonok, tabletek természetesen jelenthetnek köztes, de nem ideális megoldást, hiszen ezeket a felületeket elsődlegesen nem az okostelefonok kijelzőire tervezték. Mindezek tükrében joggal merülhet fel az igény egy olyan offline alkalmazásra, amely a szükséges információkat akár internet hiányában is elérhetővé teszi, illetve amelyet kifejezetten, az egyetemisták többsége által ma használt okos telefonokra terveztek.

Az előadás célja, hogy bemutassa az MTA SZTAKI által készített UNI@HAND ELTE PPK nevű, mobil eszközökön használható többnyelvű alkalmazást, amelynek célja az ELTE Pedagógiai és Pszichológiai Kar hallgatóinak széleskörű tájékoztatása. A fejlesztés során felhasználtuk a GUIDE@HAND turisztikai célokra kifejlesztett mobil alkalmazásban szerzett sokéves tapasztalatokat. Az egyetemi alkalmazás funkcionalitása a kar munkatársaival közösen került meghatározásra. A fejlesztés során törekedtünk arra, hogy az alkalmazás a megrendelő kívánságainak megfelelően gyorsan és rugalmasan testre szabható legyen.

2. UNI@HAND: Miért jó az Egyetemnek és a hallgatóknak?

Számos olyan információ van, amelyeket fontos, hogy az egyetemek minél hatékonyabban és minél naprakészebben juttassanak el hallgatóikhoz. A kurzusokkal kapcsolatos információkon túl ilyen az egyetem, illetve egyes szervezeti egységeinek a felépítése, az ott dolgozó oktatók elérhetősége és egyéb adatai, de ilyenek a különböző épületek, oktatói szobák, tantermek elérhetőségének adatai is. A tanulmányaikat gyakran számukra idegen nagyvárosban megkezdő diákok számára nagy segítség lehet, ha a városban vagy akár az egyetemi épületeken belül segíteni tudjuk a tájékozódásukat. Az is segítség lehet, ha az egyéb fontos helyek, mint a jegyzetbolt, a könyvtár, a fénymásoló, de akár a kikapcsolódás, szórakozás lehetőségei is megjelennek egy ilyen szolgáltatásban. Annak ellenére, hogy ezek az információk többségükben elérhetők különböző forrásokból, mégis, különösen az egyetemi tanulmányait frissen megkezdő diák számára nagyban segítheti az eligazodást, ha ezeket egy forrásból, könnyen elérhető, a telefonján hordozható módon, az ő számára célzottan összerendezett szerkesztésben kapja meg. Az egyetemi élet megkezdése nagyon örömteli, de ezzel együtt stresszes és számos kihívást jelentő feladat is, amely gyakran lakóhelyváltással, életformaváltással, a környezet, a társas kapcsolatok és számos más tényező változásával jár együtt. A UNI@HAND ebben a változásokkal teli időszakban nyújthat praktikus segítséget.

Az alkalmazás bevezetésének néhány további előnye az egyetem számára:

- a meglévő honlap többnyelvű kiterjesztése, azaz külföldi diákoknak, oktatóknak, vendégeknek is ideális,
- nem kell új adatbázist építeni, mert a meglévő integrálható a mobil megjelenítéshez,
- meglévő, kész keretrendszer, amely az egyetem kívánságainak megfelelően gyorsan és rugalmasan testre szabható, választható modulokat tartalmaz (például órarend szerkesztési lehetőséget),

- előre (otthon, egyetem, stb.) letölthető alkalmazás, amely internet használata nélkül használható,
- szabadban GPS, az épületen belül QR használatával segít a tájékozódásban előre elkészített külső, belső térképek segítségével,
- kész keretrendszer, gyorsan bevezethető.

3. A GUIDE@HAND alkalmazáscsalád és szolgáltatásainak általános bemutatása

Az MTA SZTAKI eLearning Osztálya 2005 óta folytatott turisztikai célú szoftverfejlesztési tapasztalataira alapozva a mai kornak és trendeknek megfelelő megoldást dolgozott ki GUIDE@HAND néven [1]. A fejlesztés eredményeit több hazai és nemzetközi szakmai díjjal jutalmazták. A GUIDE@HAND alkalmazás turisztikai célú alkalmazásainak használatával nem csak a látogató érezheti magát otthonosan egy múzeumban, egy idegen városban, hanem egy lokálpatriótának is felejthetetlen élményt nyújtanak az elérhető tartalmak, szolgáltatások.

A GUIDE@HAND alkalmazáscsalád tagjainak a legfontosabb szolgáltatásai:

- **okostelefonokon** vagy **tableteken** érhetőek el,
- elérhetőség a legelterjedtebb mobil platformokon (**iOS és Android**),
- előzetes letöltés esetén használatához **nincs szükség Internet** kapcsolatra (nem kell Internetért fizetni),
- **interaktív**, offline vagy online **térkép** használata, valamint GPS használata helymeghatározáshoz (szabad térben),
- területileg vagy tematikusan gyűjtött **látványosságok** felfedezésének lehetősége hangos **vezetett séták** vagy **szabad barangolás** révén,
- hasznos helyek (éttermek, helyi szolgáltatók, gyógyszertárak, stb.) bemutatása, megjelenítése a térképen,
- hírek, **események**, rendezvények ajánlása,
- **GPS használata** helymeghatározáshoz szabad térben,
- **QR kódok alkalmazása** (zárt térben, szabadban):
 - látványosságok azonosítása,
 - PR tevékenység támogatása,
 - személyre szabott tartalmak kezelése.
- offline **túratervezési lehetőség** tematikus adatbázisok (gyalogos, kerékpáros, stb.) használatával,
- **élmények megosztása** a mobilon és megtekintése weben,
- többnyelvű web-es portál a következő funkciókkal:
 - tartalmak előkészítése, szerkesztése és publikálása,
 - használati adatok gyűjtése és rendszerezése,
 - tartalmak előzetes bemutatása, a közvetlen alkalmazásletöltés és közösségi funkciók támogatása.
- Integráció külső adatbázisokkal (pl.: turisztika adatbázis).

Alkalmazási területek:

- Turisztikai alkalmazások (Budapest, Tokaji régió, Mátra stb.)
- Múzeumi/belső térben használt alkalmazások (Duna Múzeum, Petőfi Irodalmi Múzeum, Bolgár Ikonográfiai Digitális Könyvtár, Budapesti Történeti Múzeum Vármúzeuma, Győri Fotóklub Egyesület)

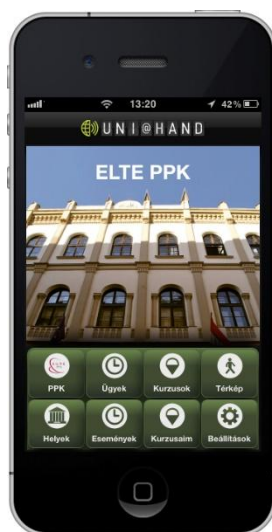
- Egyetemi alkalmazás (ELTE PPK)
- Település-információs alkalmazás (Törökbálint)
- Esemény bemutatók (Múzeumok éjszakája, Kassa, Könyvfesztiválok, Nyitott Balaton, stb.)

A folyamatos bővülés eredményeképpen hazánkban és egyre több külföldi országban, egyre több város és régió (Esztergom, Vác, Budapest, Gödöllő, Badacsony, Győr, stb.) kerül be önállóan, saját arculattal a GUIDE@HAND alkalmazások közé. Az alkalmazás honlapja a következő linken érhető el: <http://guideathand.com/>

4. UNI@HAND funkciói és használata

Az UNI@HAND egy többnyelvű komplex információs mobil szolgáltatás a hallgatók életével kapcsolatos naprakész információk (tanárok, tárgyak, események, stb.) kezelésére és megjelenítésére. Az ELTE PPK számára elkészített alkalmazás főmenüje az alábbi menüpontokat tartalmazza:

- PPK
- Ügyek
- Kurzusok
- Térkép
- Helyek
- Események
- Kurzusaim
- Beállítások



1. ábra. UNI@HAND ELTE PPK főmenü

A *PPK* menüpont bemutatja a kart, a hozzá tartozó szervezeti egységeket és az oktatókat. Az alábbi almenüket tartalmazza:

- ELTE
- ELTE PPK
- HÖK
- Intézetek
- Oktatók

Az *Ügyek* menüpont egyetemi ügyintézéshez szolgál hasznos információkkal:

- Nemzetközi ügyek
- Tanulmányi ügyek

A *Kurzusok* menüpont áttekintést nyújt az ELTE PPK tantárgyairól. Egy listát jelenít meg, amiből egy tantárgyat kiválasztva, arról részletes információk jeleníthetők meg, mint például a tantárgy leírása, szakirodalom, időpont, tantermek és oktatók. A kiválasztott tárgyakból áll össze a hallgatók egyéni órarendje, amely *Kurzusaim* menüpont alatt tekinthető meg.



2. ábra. Egy kurzus adatlapja

A *Térkép* menüpont alatt egy interaktív térkép érhető el Budapestről és az egyetemről. A térképen megjelenik a felhasználó aktuális tartózkodási helye és a környéken található hasznos helyek. Az egyes helyekre kattintva további információ érhető el róluk.

A *Helyek* menüpont bemutatja a Kar épületeit és a diákok képzéséhez és kikapcsolódásához kapcsolódó szolgáltatásokat az egyetemen és környékén:

- Épületek
- Kikapcsolódás
- Szolgáltatások

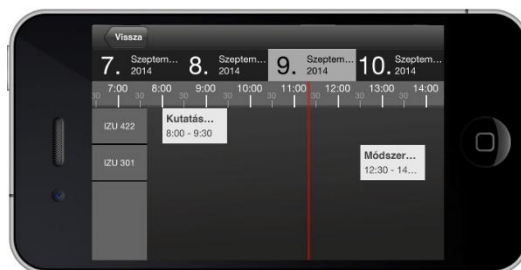
A helyek többszintű kategória hierarchiában érhetők el, és lista formájában áttekinthetők a különböző kategóriákhoz tartozó helyszínek. Az egyes helyekhez részletes információ tartozik képekkel, pontos címmel és elérhetőségekkel együtt. Az alkalmazás megjeleníti az épület távolságát a felhasználó aktuális tartózkodási helyétől.



3. ábra. Egy helyszín adatlapja

Az *Események* menüpont tartalmaz egy többszintű, tematikus eseményajánlót a helyszínek bemutatásával az egyetemi rendezvények bemutatására. Az alkalmazás segítségével emlékeztető értesítés állítható be az egyes események kezdete előtt.

A *Kurzusaim* menüpont a hallgató órarendjének táblázatos megjelenítésére szolgál. A táblázat oszlopai felelnek meg a napoknak, míg a sorai a tantermeknek. Offline egyéni órarend formájában jelenik meg. Az alkalmazás egyelőre nincs összekötve a Neptun rendszerrel, a hallgató a Kurzusok menüpontban saját maga választhatja ki az órarendjében szereplő tárgyakat.



4. ábra. Órarend

A *Beállítások* menüpontban az alkalmazás általános beállításai (pl. nyelv, online vagy offline térkép használata, frissítések, stb.) érhetők el.

Az alkalmazás tartalommal való feltöltését webes adminisztrációs felület segíti, de lehetőség van a már meglévő webes egyetemi adatbázisokkal való összekapcsolásra is. Az UNI@HAND igény szerint más oktatási intézmények számára is tesztre szabható. Az intézmények különböző szolgáltatáscsomagok közül választhatnak:

- *Alap szolgáltatáscsomag* - az egyetemi honlap közvetlen kiterjesztése:
 - Egyetem bemutatása,
 - Interaktív térkép a városról és az egyetemről,
 - Több szintű, tematikus eseményajánló a helyszínek bemutatásával az egyetemi rendezvények bemutatására,
 - Emlékeztető értesítés kezelése az egyes eseményekhez.
- *UNI információs szolgáltatáscsomag* - egyes szervezeti egységek részletes bemutatása / szervezeti egységként:
 - Egyetemi szolgáltatások,

- Tanulmányi ügyek,
- HÖK,
- Szervezeti egységek, stb.
- minden állandó tartalommal, ami az egyetem honlapjáról elérhető
- *Választható kiegészítő szolgáltatáscsomagok - szabadon variálhatók:*
 - szolgáltatás ajánló (éttermek, üzletek stb.),
 - online kérdőívek, egyetemi felmérések,
 - útvonal tervező (külső helyszínen GPS, belső helyszínen QR kód segítségével),
 - az én órarendem (Órarend készítő, helyszín, oktató hozzárendelésével, megjegyzés modullal (mikorra milyen beadandóm van... stb.),
 - további modulokkal az egyetem igényei szerint kiegészíthető, amelyhez fejlesztőink várják az újabb igényeket.

5. Összefoglalás

Az UNI@HAND rendszer fejlesztése és az ELTE PPK által szolgáltatott tartalmakkal való feltöltése a befejezéséhez közeledik. 2014 őszén élesben teszteljük a rendszert. Reményeink szerint a tesztelés igazolja, hogy valós szükséglet mentén kezdtük el megalkotni az egyetemi mobil alkalmazásunkat. Az alkalmazást igyekeztünk úgy elkészíteni, hogy minél könnyebben testre szabható legyen. Bízunk benne, hogy az alkalmazás más felsőoktatási intézményeknek is felkelti érdeklődését, és az UNI@HAND további helyeken is megjelenhet.

Irodalomjegyzék

- [1] Zsolt László Márkus, Balázs Wagner, *GUIDE@HAND: Digital GPS Based Audio Guide that Brings the Past to Life, Proceedings of the Digital Presentation and Preservation of Cultural and Scientific Heritage International Conference, Veliko Tarnovo, Bulgária, (2011), pp. 15-25*

A JavaScript nem Java?

JavaScript is not Java?

Kádek Tamás^a, Pánovics János^b

Debreceni Egyetem, Informatikai Kar

^akadek.tamas@inf.unideb.hu

^bpanovics.janos@inf.unideb.hu

Absztrakt: A Java programozási nyelv világába a Java 8 – amely több év kihagyás után követte elődjét – több olyan újítást is hozott, mellyel igyekszik megnyerni a fejlesztőket. Egyike ezen eszközöknek a Nashorn JavaScript motor. A Java 8-cal tehát „A JavaScript nem Java” szlogen, melyet éveken keresztül olvashattunk a JavaScript kézikönyvek bevezetőjében, már nem is annyira egyértelmű. És bár a két programozási nyelv közti koncepcionális különbség valóban éles határt von a két eszköz közé, a felhasználók tábora egyesülhet.

Cikkünkben arra a kérdésre keressük a választ, hogy milyen lehetőségeket nyit meg a két nyelv közös használatával készített programkód a rendszerfejlesztők számára. A két nyelv együttműködésének előnyeit két gyakorlati példával illusztráljuk.

Kulcsszavak: programozási nyelvek, Java 8, JavaScript, Nashorn

Abstract: Java 8, which followed its predecessor after years, has brought a number of new features to the world of Java, trying to win the developers. One of these features is the Nashorn JavaScript engine. As a result, the slogan “JavaScript is not Java”, which could have been read for years in the introductions of JavaScript manuals, is not so evident anymore. Although the conceptual difference between the two programming languages indeed makes a sharp border between them, the groups of their users may unite.

In our paper, we are looking for the answer to the question: “What potentials can be opened up for system developers using a program code featuring both languages?” We illustrate the benefits of the cooperation of Java and JavaScript using two practical examples.

Keywords: programming languages, Java 8, JavaScript, Nashorn

1. Bevezetés

A JavaScript tankönyvek előszeretettel használt mondata, hogy „a JavaScript nem Java”. Bár ez valóban így van, a Java 8 megjelenésével az Oracle egy új szintre emelte a két nyelv kapcsolatát. JavaScript-kódokkal eleinte weblapok dinamikus tartalmának kliens oldali előállításakor találkozhattunk, webböngészők (pontosabban a Netscape) szkriptnyelveként mutatkozott be. Az első Java-verziókban a két nyelv kapcsolata kimerült abban, hogy a JavaScript nyelvű programok kommunikálni tudtak a böngészőben futó applettekkel (ahogy a böngészőben elérhető egyéb objektumokkal is). A megnevezésük hasonlatosságának ellenére a két nyelv között számos különbözőség van, amelyeket az alábbi táblázat foglal össze:

Tulajdonság	Java	ECMAScript
<i>paradigma</i>	objektumorientált	prototípus-alapú
<i>a nyelv típusa</i>	általános célú fordítóprogramos és interpreteres nyelv	interpreteres szkriptnyelv
<i>típusrendszer</i>	gyengén típusos, statikus	dinamikus
<i>párhuzamos programozás</i>	van	nincs
<i>programkönyvtár</i>	gazdag	szegényes
<i>funkcionális nyelvi eszközök</i>	Java 8-tól	minden verzióban

Idővel a JavaScript nyelv, illetve az ECMAScript szabvány szerepe változásnak indult: a web nemcsak kliens, hanem szerver oldalán is megjelent (lásd Node.js), és mára már akár mobil alkalmazásokat is készíthetünk vele (lásd WinRT).

A Java 6-ban és 7-ben is volt már ugyan JavaScript-motor (Rhino), de az még az ECMAScript 3-as szabványát valósította meg, illetve a JRE nem tartalmazott JavaScript interpreter alkalmazást. E tekintetben a Java 8 nagyot lépett előre: a Nashorn JavaScript-motor az ECMAScript 5.1 szabványára épül, és a JRE tartalmaz egy `js` nevű külső JavaScript interpretert. Az 5.1-es szabvány legfőbb előnye a 3-ashoz képest, hogy saját Java objektumokat is tudunk manipulálni JavaScript kódból (lásd a beépített `Java` objektumot és annak metódusait).

A következőkben két életszerű példán keresztül szeretnénk bemutatni a két nyelv kapcsolatát:

- egyfelől annak módját, hogy hogyan lehet Java-kódot kiegészíteni JavaScriptben megírt funkcionalitással,
- másfelől annak módját, hogy a Java-világ elemeit hogyan építhetjük be JavaScript-kódjainkba.

2. Java kód kiegészítése funkcionális JavaScript betéttel

Első példánkban azt szeretnénk illusztrálni, hogy JavaScript-kódok segítségével hogyan bővíthető a Java-programunk funkcionális elemekkel. Erre kiváló példa a megoldáskereső algoritmusok működése, ahol a megoldandó probléma leírásától függetlenül készítjük el az algoritmus implementációját, majd külön implementáljuk a megoldandó problémákat. Ez utóbbit is tehetjük Java nyelven, de ehhez szükség van ezen Java-kódok lefordítására is, tehát a JDK-ra. A mi célunk viszont az, hogy ne kelljen csak azért JDK-t telepítenünk, hogy lefordíthassuk az egyes konkrét problémák reprezentációit, és a problémák leírására szolgáló új szakterület-specifikus nyelvet (domain-specific language, DSL) se kelljen kitalálni és szemantikáját implementálni.

A példában egy p probléma leírására az alábbi állapottermodellt használjuk:

$$p = \langle A, \text{kezdő}, c, o \rangle,$$

ahol A az állapotter, $\text{kezdő} \in A$ a kezdőállapot, $c : A \rightarrow \{\text{igaz}, \text{hamis}\}$ egy függvény, amely egy állapotról eldönti, hogy célállapot-e, $o : A \rightarrow 2^A$ pedig egy függvény, amely megadja, hogy az egyes állapotokból mely más állapotok érhetők el közvetlenül. Az állapotteret a

State interfész reprezentálja, amelynek az `isGoal()` metódusa a *c* logikai függvényt, a `getAccessibleStates()` metódusa pedig az *o* függvényt reprezentálja.

A példánkban a konkrét megoldáskereső algoritmusokat tehát Java nyelven készítjük el. Az alábbi forráskód az egyszerűség kedvéért a próba-hiba módszert implementálja:

```
package hu.unideb.inf.java8;

import java.io.InputStreamReader;
import java.util.Random;
import javax.script.*;

public interface State {
    State[] getAccessibleStates();
    boolean isGoal();
}

class Main {
    private static final Random RANDOM = new Random();
    public static void main(String[] args) throws ScriptException {
        ScriptEngine se = new ScriptEngineManager().getEngineByName("js");
        State state = (State)se.eval(
            new InputStreamReader(ClassLoader.getSystemClassLoader().
                getResourceAsStream("hu/unideb/inf/java8/statespace.js")));
        while (true) {
            if (state.isGoal()) {
                System.out.println("Sikeres: " + state);
                break;
            }
            State[] accessible = state.getAccessibleStates();
            if (accessible.length == 0) {
                System.out.println("Sikertelen.");
                break;
            }
            state = accessible[RANDOM.nextInt(accessible.length)];
        }
    }
}
```

A `ScriptEngine` objektum (*se*) nyújtja azokat a szolgáltatásokat, amelyeket már az 1.6-os verzióban is elérhettünk. A JavaScript-motor neve a „js” helyett lehetett volna „nashorn” is, főleg akkor, ha garantálni szeretnénk, hogy csak az új, 5.1-es szabványú JavaScript-kódot értelmezzük. A `ScriptEngine` osztály `eval` metódusa szolgál egy JavaScript-kód interpretálására. A kódot megadhatjuk sztringként vagy – mint jelen esetben – állományként is.

A `statespace.js` állomány tartalmazza a `State` interfészt implementáló konkrét osztályt, amely az alábbi példában a három kancsó problémájának egy lehetséges reprezentációját valósítja meg:

```
var StateType = Java.type("hu.unideb.inf.java8.State");
var StateArrayType = Java.type("hu.unideb.inf.java8.State[]");
var JarType = Java.extend(StateType, {
    amounts: [5,0,0],
    toString: function() { return "("+this.amounts.join(";")+")"; },
    isGoal: function() { return this.amounts[0] == 4; },
});
```

```

getAccessibleStates: function() {
  var m = [5,3,2], res = [];
  for (var i = 0; i < this.amounts.length; i++)
    for (var j = 0; j < this.amounts.length; j++)
      if (i != j && this.amounts[i] > 0 && this.amounts[j] < m[j]) {
        var n = this.amounts.slice();
        var k = Math.min(m[j]-n[j], n[i]);
        n[i] -= k, n[j] += k;
        var t = this;
        var child = new (Java.extend(StateType, {
          amounts: n,
          isGoal: t.isGoal,
          getAccessibleStates: t.getAccessibleStates,
          toString: t.toString
        }));
        res.push(child);
      }
  return Java.to(res, StateArrayType);
}
});

new JarType();

```

A három kancsó problémája a következő: Van 5 egységnyi vizünk és három különböző méretű kancsónk. Az első kancsó pontosan 5 egységnyi vizet tud tárolni, és kezdetben tele van. A másik két kancsó 3, illetve 2 egységnyi kapacitásúak, és kezdetben üresek. A feladat, hogy kimérjünk pontosan 4 egységnyi vizet az első kancsóban az alábbi szabályok figyelembevételével:

- Nem önthetünk ki vizet a kancsókból, és nem is adhatunk vizet a rendszerhez. Kizárólag átönthetjük a vizet az egyik kancsóból a másikba.
- Mindig a lehető legtöbb vizet önthetjük csak át. Ez a mennyiség legfeljebb a forrás kancsóban található mennyiség, illetve legfeljebb a cél kancsóban lévő üres hely lehet.

A Nashorn új eszközeinek zömét a `Java` nevű globális JavaScript-objektum segítségével érhetjük el. A szkript eleje a Java-beli típusok használatát és azok kitejesztésének lehetőségét mutatja be. A `State` interfész implementációjában egyszerűen a benne szereplő metódusok törzsének definícióját adjuk meg JavaScript-függvények formájában egy objektum literálban. Szintén a `Java` nevű objektumot hívhatjuk segítségül, ha a Java kollekcióit és tömbjeit a JavaScript-kódban szeretnénk kezelni. Erre láthatunk példát a `getAccessibleStates` metódus visszatérési értékének meghatározásakor. A `Java.eval` metódus visszatérési értéke a paraméterként megadott JavaScript-kód utolsó kifejezésének értéke lesz, ami a mi esetünkben a kezdőállapotot reprezentáló `JarType` típusú Java-objektum.

A bemutatott megoldás előnye, hogy könnyen implementálhatók a funkcionális részek JavaScript nyelven, és könnyen is cserélhetők az osztályok újrafordítása nélkül. Sajnos a két nyelv még mindig nem teljesen átjárható, nem tudjuk kihasználni a Java nyelv összes eszközét JavaScriptből. Van beépített támogatás a `JavaBean`ek használatára, melynek segítségével a bemutatott kód futási sebessége is javítható volna, de ebben az esetben az állapottér implementációja a Java-rétegbe kerülne át (egy-egy állapotot egy Java-osztály példányai reprezentálnának), és így elveszítenénk a megoldáskereső algoritmus problémafüggetlen Java nyelvű implementációját. Ezért cserébe – ahogy a példában is látható – a `getAccessibleStates` metódus törzsében állandóan új implementációt kell készíteni a `State` interfészhez.

Egy másik, gyakorlatban is használható, életszerű példa lehet heurisztika függvény implementálása JavaScript nyelven állapotér-reprezentációhoz. Előnye, hogy ha már egyszer bájtkód lett a függvényből, akkor a JavaScript-kódot már nem kell újra interpretálni, viszont a függvény törzse könnyen cserélhető, függetlenül a Java-kódtól.

3. Java-elemek beépítése JavaScript-kódba

A JavaScript-kódból a Java-elemek elérését egy olyan példával illusztráljuk, amelyben a JavaScript-kódok klasszikus előfordulását, a webet célozzuk meg. Manapság már az sem szokatlan, hogy a JavaScript-kódok nemcsak a kliens oldalon (a böngészőprogramokban), hanem akár a szerver oldalon jelennek meg. Egy nagyon egyszerű CGI-szkript (`/srv/jjs.sh`) formájában mutatjuk be a `jjs` interpreter használatát:

```
#!/bin/sh
/opt/java8/jre/bin/jjs -classpath /srv/cgi-bin -scripting $1
```

Ez a szkript a JRE-ben található `jjs` segédprogramot hívja meg az argumentumában megadott JavaScript-kód futtatása céljából. A weblapkészítő példa JavaScript-kódunk a következő:

```
#!/srv/jjs.sh
print("Content-Type: text/html\n\n")

var lng = $ENV["HTTP_ACCEPT_LANGUAGE"];
var greetings = typeof lng == "string" && lng.indexOf("hu-HU") > -1
  ? function(h) { return h < 9 ? "Jo reggelt" : "Jo napot"; }
  : function(h) { return h < 12 ? "Good morning" : "Good afternoon"; }

var page = {
  html: {
    head: { title: "Kezdolap" },
    body: {
      h1: "Hello JavaScript.",
      div: {
        script: "document.write( '"+greetings+"' (new Date().getHours()) )",
        noscript: greetings((new java.util.Date()).getHours())
      }
    }
  }
}

function write(obj) { ... }

write(page)
```

A `write` függvény segítségével JavaScript-objektumokból állítjuk össze a HTML-oldalt. Az egyes objektumok attribútumai HTML-tagek nevei, az attribútumok értéke pedig a tagek tartalma. A bemutatott példa a napszaknak megfelelő köszöntést állít elő, figyelembe véve a kliens által elfogadott nyelvet, valamint az aktuális óra értékét. Először a megfelelő nyelvű üzenetet generáló függvény kerül kiválasztásra. Ha van rá lehetőség, akkor ez a függvény a kliens oldalon, a helyi dátumnak megfelelően kerül futtatásra, ennek hiányában alapértelmezés a szerver oldalon ugyanezen függvény segítségével számított érték.

A következő eszközöket emelnénk ki a fenti kódból:

- A `jjsc` interpreter segítségével nemcsak a beépített Java-osztályok, hanem a `-classpath` paraméterben megadott erőforrások is használhatók.
- A `-scripting` opció teszi lehetővé többek között a környezeti változók elérését (a `$ENV` objektumon keresztül), amelyeket vagy a webszerver, vagy a parancsértelmező állít be (a fenti kódot Apache 2 webszerveren próbáltuk ki).
- A szkript egyes részei szerver oldalon, más részei kliens oldalon futnak. Ugyanaz a szkript ráadásul futtatható szerver és kliens oldalon egyaránt (mint pl. a `greetings` függvény).
- A szerver oldalon elérjük a Java-világ objektumait (mint pl. a `java.util.Date`).

Természetesen senkinek nem ajánljuk, hogy a weboldalakat ilyen módon állítsa elő, mivel minden egyes oldallekérés a virtuális gép (JVM) betöltését teszi szükségessé, ami nagyban lelassítja az oldalak generálását. Ugyanakkor a példa jól mutatja, hogy hogyan keverhető és használható fel újra a kliens és a szerver oldali JavaScript-kód.

4. Összegzés

Bár tudjuk, hogy a Java nyelv nem úttörő a JavaScript beépítésének tekintetében, a Nashorn (elsősorban a funkcionális eszközök vonatkozásában) sok esetben hiánypótló, és tényleges előrelépést jelent a korábbiakhoz képest. A bemutatott példák segítségével láthattuk, hogy milyen új utak nyílnak meg a Java-világot előnyben részesítő programozók előtt, ugyanakkor azt is láthattuk, hogy az új lehetőségek kiaknázásának sokszor nagy ára van. Összefoglalva tehát sok előnye van a két nyelv házasításának, de a futási teljesítmény érdekében JavaScript-kódot csak akkor célszerű Java-kódba ágyazni, ha az elsősorban valamilyen funkcionalitással egészíti ki azt.

5. Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-0001 számú projekt támogatta.

TÁMOP-4.2.2.C-11/1/KONV-2012-0001

Jövő Internet kutatások az elmélettől az alkalmazásig

Nemzeti Fejlesztési Ügynökség
www.ujszachenyiterv.gov.hu
06 40 638 638



A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] Java Platform, Standard Edition Nashorn User's Guide, <http://docs.oracle.com/javase/8/docs/technotes/guides/scripting/nashorn>, letöltve 2014.07.03-án.
- [2] Julien Ponge: Oracle Nashorn: A Next-Generation JavaScript Engine for the JVM, <http://www.oracle.com/technetwork/articles/java/jf14-nashorn-2126515.html>, letöltve 2014.07.03-án.
- [3] Standard ECMA-262, 5.1 Edition, ECMAScript Language Specification, <http://www.ecma-international.org/ecma-262/5.1/Ecma-262.pdf>, letöltve 2014.07.03-án.

Tudományos repozitóriumok az MTA-ban: a KDK¹ és a SZTAKI² tanulságai

Scientific repositories at the Hungarian Academy of Sciences: the Research Documentation Centre and SZTAKI

Micsik András^a, Gárdos Judit^b

^aMTA SZTAKI

micsik@sztaki.mta.hu

^bMTA Társadalomkutató Központ

Gardos.Judit@tk.mta.hu

Absztrakt: Két olyan repozitóriumot mutatunk be a hazai palettáról, melyeknek elsődleges célja a kutatás szolgálata. Ez a feladat új követelményeket támaszt a repozitóriumokkal szemben, kezdve az örvendetesen terjedő nyílt hozzáféréstől a kutatási adatok kezeléséig. A kutatások közzététele ma már nem korlátozódhat a közlemények megjelentetésére, a kutatási adatok és a módszerek, algoritmusok nyilvánosságát is meg kell teremteni. Egy tudományos repozitóriumban lehetővé kell tenni minél színesebb, sokfélebb háttéranyag tárolását, ezzel is elősegítve a tudományos eredmények terjesztését, újrafelhasználását, reprodukálását és ezáltal ellenőrizhetőségét. A tudományágak szokásai határozzák meg, hogy a másodelemzés mennyire elfogadott és elvárt, mely kutatási dokumentumok válnak értékessé és megőrizendővé. Az MTA irányelvei az utóbbi időben erőteljesen változtak e tekintetben. A kutatási adatok az általános problémákon túl, mint például hivatkozhatóság, tudományáganként speciális követelményeket is támasztanak. A szociológia esetében az adatok között találhatunk videókat, vagy kérdőíves válaszokat. Mind a két esetben felmerül a kérdés, hogyan tudná ezeket egy külföldi kutató használni? A tudományos adatok létrehozásának kontextusa fontos értelmezési szempont a másodelemzés során, ennek érdekében tudományágra szabott, extenzív metaadat tárolási igényeket kell kiszolgálni. A repozitóriumok a kutatástámogatás és kutatómenedzselés számára is nyújthatnak értékes adatokat. Ahhoz, hogy a repozitóriumunk mindezeket lehetővé tegye, legtöbbször az informatikai megoldás erőteljes konfigurálása és testre szabása szükséges. A társadalomtudományos adatok szenzitivitása, az anonimizálási lehetőségek és nehézségek korlátozhatják a nyílt hozzáférhetőséget. Két gyakorlati megvalósítás kapcsán nyújtunk ízelítőt a felmerült problémákból, azok megoldásaiból, valamint kitérünk a világ élvonalbeli trendjeire, fejleményeire is e téren.

Kulcsszavak: repozitórium, digitális archívum, kutatási adat

Abstract: The paper presents two recently created repositories established for supporting researchers and scientific dissemination. Scientific publication today requires solutions beyond publishing papers for various other scientific data management issues such as the access, re-use and citation of raw data. This vastly increases the formats and types of items stored in repositories. Raw data for example produced in sociology may need detailed and translated documentation of its internal structure. Repositories may also be an important data source for funding and research management. In this paper we describe the process of creating two scientific repositories in the fields of computer science and social sciences, and provide examples of the problems and solutions we faced during the implementation.

Keywords: scientific repositories, digital archives, research data

¹ MTA TK Kutatási Dokumentációs Központ, <http://www.openarchive.tk.mta.hu>

² SZTAKI Repozitórium, <http://eprints.sztaki.hu>

1. Bevezetés

A kutatás szolgálata új követelményeket támaszt a repozitóriumokkal szemben, kezdve az örvendetesen terjedő nyílt hozzáféréstől a kutatási adatok kezeléséig. Ezek a feladatok konkrét kutatási témákként jelentkeznek az informatika, a tudománymenedzsment, sőt a tudományelmélet területén manapság. A témák között a nyers adatok közzététele, azok hivatkozása, a szerzők egyértelmű azonosítása, de a digitális archívumok működését vizsgáló elméleti kérdések is megjelennek [1]. A fejlődés elég gyors, példaként említhető az ORCID [2], amely egy globális szerző azonosító, vagy a DataCite [3] hálózat, amely a kutatási adatok megőrzését és publikálását tűzte ki célul. A kutatási adatok archiválásának általános kérdésein túl tudományáganként speciális követelmények is megjelennek. A szociológia esetében az adatok között található videókat, vagy kérdőíves válaszokat. Mind a két esetben felmerül a kérdés, hogyan tudná ezeket egy, a kutatás részleteit nem ismerő, vagy akár egy külföldi kutató használni. Az informatikatudomány esetében az eredmények ellenőrizhetősége, reprodukálhatósága egyre fontosabb, és ehhez kapcsolódóan nem csak a nyers adatok, hanem az algoritmusok, szoftverek publikálása is szükségessé vált.

Európában, de világszerte is egyre-másra szerveződnek társadalomtudományos archívumok. Ezek sokszor intézményi vagy nemzeti archívumok. Talán a legnagyobb európai, társadalomtudományos adatokat kezelő szervezet a CESSDA.³ Világszinten a legnagyobb társadalomtudományos archívumokat tömörítő szervezet az IASSIST. Mindkét intézmény fő profilja a *kvantitatív* módszerrel készült adatok tárolása (tipikusan kérdőíves felvételek adatbázisai). A társadalomtudományos, *kvalitatív* adatokkal foglalkozó archívumoknak is van egy kis európai szervezete.⁴

Az Európai Uniónak sok éve célja, hogy úgynevezett kutatási infrastruktúrákat (Research Infrastructure) alakítson ki. Ennek keretében több európai bölcsészet- és társadalomtudományos archívumot támogatott. Az utóbbi néhány évben ezeket az archívumokat egységes rendszerbe kívánja terelni, két éve fut a DASISH⁵ projekt, amely az öt eddig támogatott európai bölcsészet- és társadalomtudományos archívum közötti szinergiákat, együttműködési lehetőségeket és lehetséges közös standardokat tárja fel. Magyarországon 2014 tavaszán kapott újra életre a NEKIFUT rendszer, amely a magyarországi kutatási infrastruktúrákat térképezi fel (ideértve a társadalomtudományosokat is).⁶

2. A SZTAKI Repozitórium

Az MTA SZTAKI 2013 elején létrehozta a saját repozitóriumát azzal a céllal, hogy az intézetben keletkező minden olyan a tudományos kutatás és fejlesztés, valamint műszaki innováció során keletkezett digitális anyagot tároljon, mely valamilyen megőrzendő értéket tartalmaz. Az archívum elsődleges gyűjtőköre tudományágak szerint: műszaki tudományok és technológiák, számítástechnika, számítógép tudomány, irányítás és vezérlés. A SZTAKI Repozitórium nem csak a tudományos közlemények teljes szövegének tárolására ad lehetőséget, hanem a kapcsolódó anyagok, például prezentációk, nyers adatok, videók, stb. elhelyezésére is. A tudományos anyagok mellett műszaki jelentések, tanulmányok, segédletek, stb. is feltölthetők. A repozitóriumban az MTA SZTAKI munkatársai

³ <http://www.cessda.net>

⁴ EQUALAN, <http://www.iqda.ie/content/equalan>

⁵ <http://www.dasish.eu>

⁶ <http://www.nih.gov.hu/strategiaalkotas/kfi-infrastruktura/kutatasi-infrastrukturak>

helyezhetnek el anyagokat, és azok elérhetőségét három szinten korlátozhatják: mindenki, az intézet munkatársai, a repozitórium adminisztrátorai.

A fenti célok megvalósítására a rugalmasan konfigurálható EPrints rendszert választottuk, mely a University of Southampton által immár 13 éve fejlesztett szabad szoftver. Az elmúlt másfél év alatt jó tapasztalatokat szereztünk a rendszerről, mivel jelentős felhasználói közössége van, kicsi az erőforrásigénye (gyengébb hardver konfigurációban is gyors válaszidővel működik), és hasznos kiegészítő bővítményekkel rendelkezik.

Az EPrints konfigurálását az autentikációval kezdtük: a SZTAKI LDAP szerverét használjuk a felhasználó azonosítására, az EPrints wikiben található recept alapján. Az autentikáció során a felhasználó LDAP adatainak egy részét áttöltjük az EPrints-be, így például a név vagy a részleg változásai automatikusan átkerülnek az EPrints-be. Ilyenkor megkapjuk a felhasználó LDAP azonosítóját is, és jobb híján ezt kezdtük el használni a szerzők egyértelműsítésére is. Később ezt kiegészítettük a szerző ORCID és MTMT azonosítóival is. Ezen szerző-azonosítás alapján megoldottuk, hogy a bejelentkezett felhasználó az általa feltöltött tételeket tudja módosítani, amely egyébként nem megoldott az EPrints-ben, mivel a szerzők és a felhasználók két külön rekordkészletet alkotnak. Az új tételeket feltöltés után a szerkesztők hagyják jóvá, a bevitt metaadatok ellenőrzése után.

Az EPrints tételek (ún. eprint-ek) metaadat mezőit könnyedén lehet módosítani, így felvettünk több olyan mezőt is, amelyek a pályázati támogatást rögzítik, illetve a weblapunkon való megjelenést segítik (pl. magyar nyelvű kulcsszavak). Továbbá az EPrints-ben szabályozni tudjuk, hogy mely mezők legyenek kereshetőek, illetve mely mezők alapján készüljenek automatikus listázások, és azok hogyan jelenjenek meg. Erre példaként készítettünk olyan listázást, amely a szerző LDAP azonosítója alapján gyűjti ki a szerző publikációit.

A repozitóriumot feltöltöttük a korábban más rendszerben tárolt publikációs metaadatokkal, ezzel lehetőséget adva, hogy kollégáink utólagosan egyszerűen feltölthessék cikkeik teljes szövegét. A 2013-tól életbe lépett MTA open access rendelet [4] kötelezővé tette a közleményeink teljes szövegének elérhetővé tételét, amelyet már a SZTAKI Repozitóriummal is tudunk teljesíteni, mivel az megfelelt az MTA repozitórium minősítési folyamatban. A folyamat során definiálnunk kellett a repozitórium működésével kapcsolatos keretfeltételeket, amely egy hasznos ellenőrző listaként szolgált számunkra is. Ennek során a repozitórium működtetésének személyi és műszaki feltételeit kellett tisztázni, valamint a tartalommal kapcsolatos minőségi jellemzőket felmérni. Ezek között említendő elemek: a hardver üzemeltetésének, a mentések rendszerességének biztosítása, a bekerülő anyagok ellenőrzési folyamata, és a más szerverekkel való kapcsolatok.

A repozitóriumok hasznossága nagymértékben a külső láthatóságon, kapcsolatokon múlik. A legáltalánosabban használható kapcsolódási mód az OAI-PMH protokoll [5], ez alapértelmezésben be van kapcsolva az EPrints-ben. Az OAI kapcsolódási pontot a ROAR [6] és OpenDOAR [7] nyilvántartásokban jelentettük be.

Az OpenAIRE project [8] az EU FP7 és H2020 által támogatott közleményeket gyűjti. Az ő nyilvántartásukba is lehet csatlakozni, ennek feltétele az, hogy az általuk megadott módon tároljuk a finanszírozási adatokat, és az FP7 illetve H2020 által támogatott tételek egy OAI set-ben kigyűjthetőek legyenek. Mindezeket egyetlen EPrints bővítmény telepítésével meg lehet oldani.

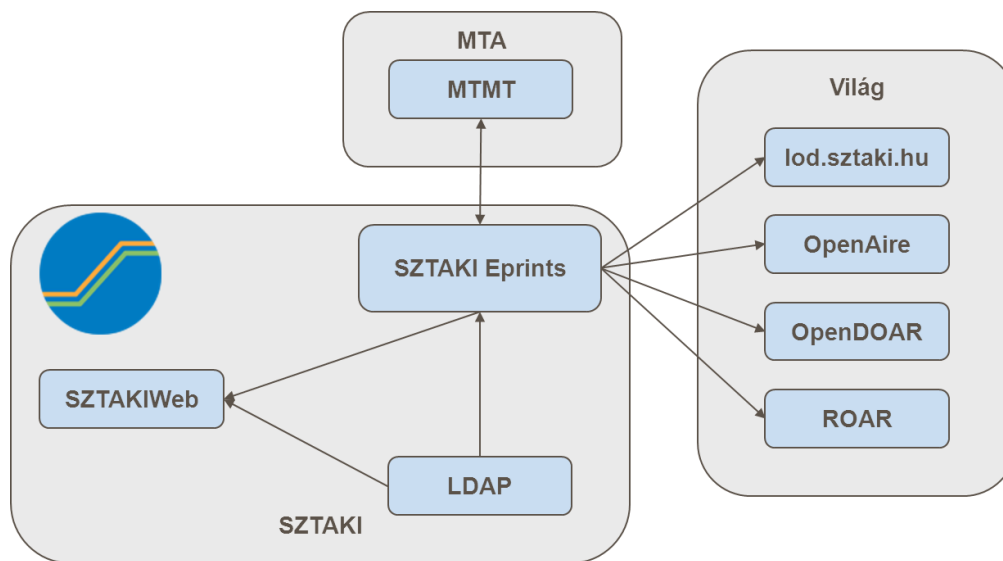
Egy másik kapcsolódási pontunk az MTMT-ből teszi lehetővé a PDF fájlok közvetlen feltöltését a közlemény adatainak megadásával egyidejűleg. Ehhez a SWORD protokollt használjuk, melyet az EPrints-ben könnyen be lehet kapcsolni, és minősített repozitóriumok esetén az MTMT rövid tesztelés után engedélyezi a SWORD kapcsolatot. A SWORD

kapcsolaton keresztül feltöltött közlemények a szerkesztők postaládájában jelennek meg, a főbb metaadatokat kitöltve látjuk már, de ellenőrzésre szorulnak. Fontos, hogy az ilyen tételek esetén tároljuk a közlemény MTMT azonosítóját is, ebből láthatjuk, hogy a tétel az MTMT-ben már rögzítve van.

Sok belső vitánk volt arról, hogy a publikációkat először az MTMT-be töltsük fel, vagy a saját repozitóriumunkba. Univerzális megoldás nem született, ezért mindkét irányt támogatjuk, az egyik irányban a SWORD kapcsolaton keresztül, míg a másik irányban az EPrints RIS export lehetőségét igazítottuk az MTMT RIS import funkciójához. Ez utóbbit akkor érdemes használni, ha még a közlemény megjelenése előtt közzé akarjuk tenni eredményünket preprint-ként, és ilyenkor a megjelenés adatait (oldalszám, DOI, stb.) még nem tudnánk az MTMT-ben kitölteni.

Az MTMT és a SZTAKI repozitórium szinkronban tartására írtunk egy szkriptet, amely az MTMT-ből exportált XML alapján kiegészíti az Eprints-ben tárolt tételeket.

A repozitórium adatait RDF-be konvertálva áttöltjük saját Linked Open Data szolgáltatásunkba is (lod.sztaki.hu). Végezetül, a SZTAKI intézeti portáljába több helyen be vannak integrálva az EPrints szerverről letöltött adatok. A személyek oldalain és a részlegek oldalain is megjelennek a kapcsolódó publikációk, melyek többféleképpen rendezhetők is. Ezen kívül kulcsszavak és témakörök szerint is kereshetők a publikációk. Mindezek az adatok a repozitórium felé továbbított egyedi keresések eredményeként kerülnek be a portál weboldalaiba. Az ismertetett kapcsolatokat összegzi az 1. ábra.



1. ábra. A SZTAKI Repozitórium kapcsolatrendszere más szolgáltatásokkal

3. A KDK

Az MTA Társadalomkutató Központjának (TK) Kutatási Dokumentációs Központja (KDK) az MTA központi határozata után született meg 2013 februárjában. A KDK szakmai vezetője (Kovács Éva) és munkatársai a 20. Század Hangja Archivum és Kutatóműhely⁷ alapítóiként több éves társadalomtudományos archivumi tapasztalattal rendelkeznek. A 20. Század Hangja

⁷ <http://www.20szadhangja.hu>

Archívum az Open Society Archive-val (OSA) együttműködésben, az OSA szerverén⁸, a DSpace nyílt szoftver felhasználásával működik.

A KDK kettős céllal jött létre. Mint repozitórium, a TK-ban létrejött kutatási nyersanyagok digitális másolatait archiválja és kutatási adatokat szolgáltat, elsősorban TK-s kutatóknak. Mint sokrétű és sokfajta metaadatot szervező rendszer pedig kutatásokról szolgáltat alapinformációkat; pl. a kutatás témájáról, résztvevőiről, idejéről, finanszírozójáról stb. lehet tájékozódni.

Az MTA TK négy intézetében (Jogtudományi, Kisebbségtudományi, Politológiai, Szociológiai) létrejövő nyersanyagok, dokumentumok, adatok nagyon sokrétűek. Találhatunk köztük táblázatos adatbázisokat (kérdőíves felmérésből, sajtószemléből, stb.), interjúszövegeket, fókuszcsoporthoz tartozó kutatásokból származó videófelveteleket, fényképeket, terepnaplót, stb. Olyan szoftverre volt szükségünk, amely tehát dokumentumszinten tud metaadatokat kezelni és kereshetőséget biztosítani, másfelől a dokumentumokat gyűjteményekbe tudja rendezni, és egységesen megjeleníteni. Szükségünk volt egy differenciált hozzáférési beállításokat kezelni tudó szoftverre. Alapvetően egy olyan digitális archívumot terveztünk, amely a kutatóknak közvetlenül letölthető, jó minőségű kutatási anyagokat kínál.

Ahogy már fenn bemutattuk, a társadalomtudományos mező Európában és világszerte dinamikusabban fejlődik és mindig változó képet mutat. Tehát nem létezik (még?) Európában egy olyan társadalomtudományos archívumi tér, ideértve a kanonizált és elfogadott digitális megoldásokat is, amelyben könnyen és magától értetődően el tudná magát helyezni és el tudna kezdeni működni egy új, komplex társadalomtudományos adatokat archiválni kívánó intézményi archívum Magyarországon.

A hasonló intézmények európa- és világszerte általában saját igényeikre szabott informatikai megoldásokat használnak digitális repozitóriumukban. Létezik már 1-2 olyan kész, angol nyelvű megoldás is, amely sokfajta felmerülő igényt ki tud szolgálni (Dataverse⁹, CKAN¹⁰).

Az MTA tudományos közleményeket tároló repozitóriuma, a REAL, szintén a nyílt forráskódú Eprints szoftverrel működik. A viszonylag könnyű kezelése, a magyarra már lefordított felülete, és a jó konfigurálhatósága miatt, valamint az MTA repozitóriumok közötti átjárhatóság biztosítása érdekében 2013 elején úgy döntöttünk, hogy Eprints alapokon fejlesztjük a KDK repozitóriumot.

A repozitórium kialakítása 2013 őszén indult el a SZTAKI segítségével. A tudományos publikációkra tervezett Eprints-en alapvető változásokat eszközöltünk. A legfontosabb a kutatási gyűjtemény, mint egység bevezetése volt, melyhez az Eprints ReCollect bővítményét használtuk fel. A ReCollect a UK Data Archive és a University of Essex fejlesztése, és a következő főbb változtatásokat adja a rendszerhez: a kutatási gyűjtemény típusához egy új, egyedi nézetet rendel, amely sok kapcsolt fájl esetén is jól áttekinthető. A gyűjtemény feltöltésére szolgáló workflow-t is jelentősen leegyszerűsíti, miközben hozzáadja azokat metaadat-mezőket, amelyek a DataCite, INSPIRE és DDI alapján a kutatási adatok leírásához elengedhetetlenek (pl. időbeli és geográfiai lefedettség), végül az új leíró mezők szerinti kereshetőséget is beállítja.

⁸ <http://voices.osaarchivum.org>

⁹ <http://datascience.iq.harvard.edu/dataverse>

¹⁰ <https://orbital.blogs.lincoln.ac.uk/2012/09/06/choosing-ckan-for-research-data-management>

A fentiekén túl még számos további metaadatmezőt is létre kellett hozni, hogy a KDK életének első néhány hónapjában begyűjtött kutatásleírások átvehetőek legyenek. Ezek alapján a kutató már keresés és böngészés közben láthatja, hogy az adott anyag milyen és mennyi dokumentumból áll, milyen a lefedettsége, milyen adatgyűjtési módszereket használtak, stb.

További újításként kereszthivatkozási lehetőséget hoztunk létre, mellyel a kapcsolódó tudományos publikációk és a kutatási gyűjtemények mindkét irányból hivatkozhatóak.

A kutatási gyűjtemény permanens URL-lel rendelkezik, ezáltal egy tudományos publikációban az adatokra való hivatkozás lehetségessé válik. A későbbiekben minden gyűjteményt DOI-val szeretnénk ellátni, egyelőre az EPrints saját URL kiosztási módszerét használjuk. A kutatási adat hivatkozása ma még nagyon egyszerű, nem részletekbe menő, és nem megoldott a közvetlen kapcsolat a konkrét adat és az azt elemző szövegrész között; mégis egy olyan lehetőség bevezetése, amellyel ma Magyarországon még alig élnek, de nemzetközileg egyre elfogadottabbá válik. Ennek megoldásával a Research Data Alliance (RDA) külön munkacsoportja¹¹ valamint az egyik legnagyobb európai társadalomtudományos adatarhívum, a UK Data Service egy projektje¹² is foglalkozik jelenleg.

A gyűjtemények egyes dokumentumainak más és más hozzáférési beállításai lehetnek. A metaadatok mindig nyilvánosak (kivéve egy mezőt, amelyben esetleges szenzitív adatokat tárolunk, pl. egy interjúalany nevét). Társadalomtudományos kutatási anyagok számos esetben szenzitív adatokat tartalmaznak, amelyekhez való hozzáférést korlátozni kell. Ezért a KDK-ban többfajta hozzáférési beállításokat dolgoztunk ki, a SZTAKI Repozitóriumhoz hasonlóan. Vannak gyűjteményeink, amelyek bármelyik honlaplátogató előtt nyitottak, vannak csak MTA TK-s felhasználói adatokkal rendelkezők számára letölthetőek, és vannak olyanok, amelyekhez egyedi, kutatói döntés után férhetnek csak hozzá az érdeklődők.

Mivel a KDK repozitóriumában a személyek beazonosításához megfelelő kutatási anyagok csak a kutatásban résztvevők kifejezett beleegyezési engedélyével tárolhatók, az engedéllyel nem rendelkező dokumentumok esetében a szenzitív adatokat leválasztjuk a többi adatról. Ez egy komplex anonimizálási folyamat része, ahol a folyamat minden lépését dokumentáljuk, és a személyes adatokat tartalmazó dokumentumokat zártan tároljuk. Amennyiben a későbbiekben mégis lehetségessé válik az adatok újbóli összerakása, a dokumentáció segítségével megoldható.

Ez az eljárás megfelel a Data Documentation Initiative¹³ -Lifecycle (DDI-L) ajánlásnak. A DDI-L alapján a metaadat dokumentálása nem a kutatás *után*, hanem már közben történik, minden, az adatok milyenségét befolyásoló lépcsőben. Gyakorlatilag tehát már a kutatás létrejöttékor érdemes elkezdni rögzíteni az adatokkal kapcsolatos információkat. A 2. ábra szemlélteti a DDI-L működési sémáját, és látszik, hogy melyek azok a lépések, ahol ezen ajánlás szabályai szerint dokumentálni kell azt, ami befolyásolta az adatok létrehozását és milyenségét.

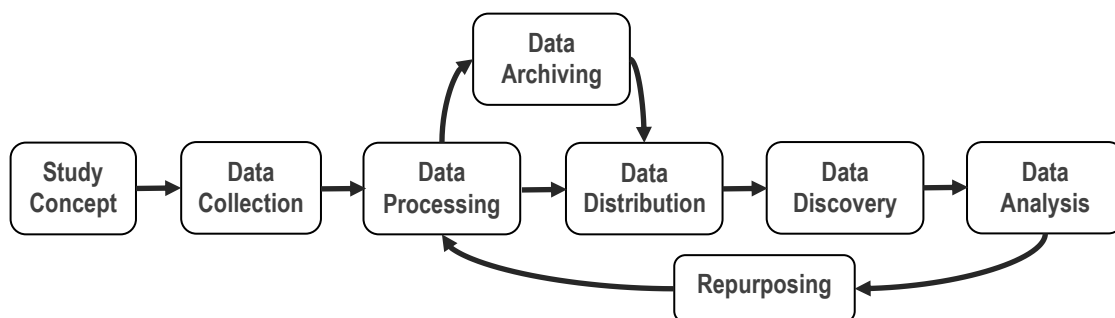
A dokumentáció gépileg is feldolgozható, és számos olyan informatikai fejlesztéssel találkozhatunk, amely a DDI szerint dokumentált metaadatokat kezeli, tárolja, elemzi és jeleníti meg komplex módon. Léteznek olyan szoftverek, amely segítségével a metaadatok és maguk az adatok egy közös felületre rendeződnek ezáltal, ahol látszódik az adatok keletkezésének és kezelésének, változtatásának több eleme.

¹¹ <https://rd-alliance.org/group/data-citation-wg.html>

¹² <http://ukdataservice.ac.uk/about-us/projects/digital-futures/details.aspx>

¹³ <http://www.ddialliance.org>

A DDI és a DDI-L jelenleg leginkább kvantitatív adatok esetében használatos, de már kvalitatív adatok dokumentációjánál is felbukkan.¹⁴ Alkalmazásukkal hatékonyan és nemzetközi összehasonlítást és használhatóságot biztosító módon lehet dokumentálni a kutatási adatok, dokumentumok élettörténetét.



2. ábra. A DDI-L működési sémája¹⁵

A DDI-L bevezetéséhez a kutatók együttműködésére van szükség. A társadalomtudományokban jelenleg sokféle munkamegosztási modell létezik; léteznek olyan projektek, amelyek esetében elengedhetetlen már a kutatás közben dokumentálni az adatok létrejöttét és manipulálását, hogy a kutatótársak is dolgozni tudjanak az adatokkal. Más projektek esetében, főleg egy- vagy néhány személyes kutatások esetében ez ritkábban valósul meg, ezért később már követhetlenné válik számos dokumentum létrejötte és életútja. A kutatási adatmenedzsmentnek mint kutatók felé támasztott eljárásnak a bevezetése¹⁶ egy módja lehet annak, hogy később olyan dokumentumok, adatok álljanak rendelkezésre, amelyeket el lehet helyezni egy repozitóriumban és mások újra tudják elemezni.

4. Összefoglalás

A rendelkezésre álló technológia ma már lehetővé teszi, hogy kevés hardver erőforrással is gyors repozitóriumi megoldásokat hozhassunk létre rövid idő alatt. A probléma inkább ezen repozitóriumok használata, amely technológiai, emberi és szervezeti kihívásokat egyaránt rejt. Még csak kialakulóban van a „best practice” a kutatási adatok kezelésére, és hasonlóképpen kutatások nyilvántartása, a közreműködők, intézmények, támogatók és újrafelhasználók kapcsolatrendszerének nyilvántartása sem általános még. Viszont ezzel egyidejűleg a kutatókat fel kell készíteni ezekre az új kihívásokra, meg kell ismertetni velük az open access jelentőségét, és azt, hogy közleményeiket másképpen érdemes ezután publikálniuk. Az intézményeknek pedig folyamatos tevékenységként be kell vezetniük a kutatási adatok megőrzését, dokumentálását és gondozását, amely újfajta speciális szaktudást igényel. Mindezek teljesülése esetén élvezhetjük csak a tudományos munka új lehetőségeit, arathatjuk le ezek gyümölcseit a támogatások, kapcsolatok vagy újrafelhasználás területein.

A társadalomtudományos adatok, mind a kvantitatívok, mind a kvalitatívok, többlépcsős tisztítási, összevonási adatkezelésen esnek keresztül, mielőtt felhasználják őket a kutatók. A nyers adatokat gyakorlatilag soha nem használják a kutatók, tehát az, amire adatként

¹⁴ <http://ukdataservice.ac.uk/about-us/projects/digital-futures/details.aspx>

¹⁵ forrás: http://odaf.org/papers/DDI_Intro_forNSIs.pdf

¹⁶ Egy jól használható kézikönyv az adatmenedzsmentről: Corti et al (2014) [12].

tekintünk, a kutatói adattisztítási eljárás eredménye. Ez az adatkezelés nem csak másodfelhasználás előtt, hanem már az első felhasználáskor megtörténik. Természet- és társadalomtudományos kutatási adatok és nyersanyagok esetében a kutatók beszámolójára, emlékezetére és (főleg régebbi kutatási anyagok esetében) sokszor levéltári munkára is szükség van, hogy egy-egy kutatás létrejöttének és sorsának aspektusait megismerjük. Az ilyen feltáró munka törvényszerűen csak egy-egy olvasatát adhatja egy kutatás történetének, amelyet a DDI-L keretei között is lehet többé-kevésbé dokumentálni. Az adatok létrejöttét és milyenségét (és nem csak a társadalomtudományosokét) alapvetően meghatározza pl. a felhasznált módszer és a kutatási kérdéseket motiváló érdeklődés; ezeket mind befolyásolják többek között a személyes motivációk, az intézményi prioritások, a tudományos hagyományok, valamint a tudományos mező specifikus működése egy adott időben és tudományterületen [9][10]. Minden tudományos diszciplínában a kutatási adatok és a valóság közötti viszonyt a kutatók munkájuk (elemzéseik, írásaik) során újra és újra létrehozzák [11].

Az idők során általában változik ez a folyamat, és a kutatási adatok más és más valóságot írnak le, más korokban a kutatók más következtetéseket vonnak le, másra helyezik a hangsúlyt, mást tartanak irreleváns és nem elemzendő adatnak. Maguk az adatok is átalakulnak, például ugyanazt az adatbázist felhasználva a társadalomtudósok új változókat hoznak létre a korábbi adatokból, régi interjúk alapján új tudást termelnek, stb. Ahhoz, hogy a kutatói, az értelmezési szabadságot minél inkább segítsük, digitális repozitórium tervezőiként és építőiként arra kell törekedjünk, hogy egy kutatás minél több szeletét minél több nézőpontból mutathassuk be. A szűkös keretek között jelenleg most csupán arra van lehetőségünk az MTA TK Kutatási Dokumentációs Központjában, hogy rugalmas és széles körű metaadatolást tegyünk lehetővé és lehetőleg minél több dokumentumot szerezzünk be a kutatóktól, továbbá az adatokhoz kapcsoljuk az azokat értelmező tanulmányokat. De a jövőben olyan digitális platformokat is el lehet képzelni, ahol a tudósok együtt gondolkodva értelmezik az adatokat, ahol az eredmények és az adatok szimbiózisban mutatkoznak, és ahol ezáltal a tudományos adatokat, történetüket, létrejöttüket, és elemzésük sokszínűségét és sokrétűségét is be lehet mutatni.

Irodalomjegyzék

- [1] Mauthner, N., Parry, O. and Backett-Milburn, K.: The data are out there, or are they? Implications for archiving and revisiting qualitative data', *Sociology*, 32/4 (1998), 733-45.
- [2] ORCID, <http://orcid.org/>
- [3] DataCite, <https://www.datacite.org/>
- [4] MTA Open Access rendelkezés, <http://real.mtak.hu/eprints/mandate.html>
- [5] The Open Archives Initiative Protocol for Metadata Harvesting, 2008-12-07, <http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm>
- [6] ROAR: Registry of Open Access Repositories, <http://roar.eprints.org>
- [7] The Directory of Open Access Repositories – OpenDOAR, <http://www.openoar.org/>
- [8] OpenAIRE, <https://www.openaire.eu/>

- [9] Bourdieu, Pierre: The Peculiar History of Scientific Reason. *Sociological Forum*. Volume 6, Issue 1, March. (1991), 3–26.
- [10] Kuhn, Thomas: *The Structure of Scientific Revolution*. Chicago: University of Chicago Press. (1962).
- [11] Latour, Bruno - Woolgar, Steve: *Laboratory Life. The Construction of Scientific Facts*. Princeton University Press. (1996).
- [12] Corti, Louise - Van den Eynden, Veerle - Bishop, Libby - Woollard, Matthew: *Managing and Sharing Research Data. A Guide to Good Practice*. Sage. (2014).

Alternatív dokumentumszervezési módszerek

Alternative Methods of Document Organizing

Piller Imre^a, Fegyverneki Sándor^b

^aMiskolci Egyetem, Alkalmazott Matematikai Intézeti Tanszék

`pilller@iit.uni-miskolc.hu`

^bMiskolci Egyetem, Alkalmazott Matematikai Intézeti Tanszék

`matfs@uni-miskolc.hu`

Absztrakt: A dokumentumok szervezési módszere döntően befolyásolja a velük való munka hatékonyságát. A döntő tényezőket ebben nyilvánvalóan a dokumentumkezelés szempontjából a dokumentumok elhelyezése és a visszakeresése jelenti. Az ezekre fordított időnek és energiának kell célszerűen minimálisnak lennie. Jelenleg már számos olyan eszköz elérhető, amely ezt segíti. Fontos kiemelni ezek közül a dokumentumkezelésre kifejlesztett rendszereket, illetve az indexelő szolgáltatásokat, amelyeknek szintén ez az egyik legfontosabb feladatuk. A különféle módszereket a humán tényezők szempontjából is vizsgálunk kell. A hatékonysághoz gyakran szükséges az, hogy a felhasználónak komoly előismeretei legyenek a rendszer működésével kapcsolatban. Ez egy természetes elvárás, és ez alapján a módszereket is külön csoportokba sorolhatjuk. Ezek bemutatásának a célja nem az, hogy egy preferált módszert kiemeljünk a többi közül, hanem hogy egy áttekintést kapjunk arról, hogy milyen felhasználási módok esetében milyen módszert érdemes használni. A dolgozat javaslatot tesz egy újszerű dokumentumkezelési mód alkalmazására is. Ennek alapja egy háló, amely rugalmasabb navigációt és szűrési feltételeket biztosít már modell szintjén.

Kulcsszavak: dokumentumkezelés, tartalomkezelés, indexelő szolgáltatások

Abstract: The organizing method of the documents has a role impact for the effectiveness of everyday work. The main factors are the storing and retrieving of the documents. The aim is to minimize the required time and energy. Many applications are available which help us. It is important to consider the special document management systems and the indexing services. We must consider the various methods from the aspect of human factor. The productivity often requires higher amount of previous knowledge about the internal structure and processes of the systems. This is a natural requirement and we can classify the methods based on it. We would not like to denote preferred method only give a brief overview about the available tools and give some advices which is the most appropriate method for the given situation. This paper shows a new alternative document management method. It based on lattice and results more flexible navigation and filters for document collections.

Keywords: document management, content management, indexing services

1. Bevezetés

A számítógépen tárolt adatok logikai egysége az állomány. Felhasználói szempontból egy praktikusabb egységet jelöl a dokumentum. Az utóbbi a hétköznapi szemlélethez közelebb áll, tulajdonképpen a hagyományos dokumentumkezelés fogalmi és műveleti jelennek meg kihasználva a modern eszközök adta lehetőségeket.

A számítógéppel dolgozók többsége munkája során különféle dokumentumokat kezel. Az ehhez használt hardver- és szoftverrendszerek fő feladatai az állományok tárolása, visszakeresésük biztosítása, illetve módosítási műveletek végrehajtása. Korábban az elsődleges problémát ezeknek a műveleteknek a végrehajtása jelentette. Manapság a technológiai fejlődésnek köszönhetően a helyzet annyiban változott, hogy az elvégzendő komplex műveletsor megfogalmazásának segítése legalább olyan fontossá vált, mint a végrehajtás optimalizálása.

A felhasználói elvárásokban komoly változások történtek. Az online keresőszolgáltatások kényelmessé és gyorsá teszik a keresést az Interneten. Ez természetesen kedvező a dokumentumok kezelési hatékonyságára nézve, viszont amíg nem válnak általánossá a vékonykliens technológiák, addig ezek térnyerése azt jelzi, hogy a fájlrendszeri keresőszolgáltatások nem megfelelőek. Olyan egyszerű esetekre kell itt gondolni, mint például amikor tudjuk, hogy egy állományt már letöltöttünk és lementettünk valahová, viszont már nem emlékszünk pontosan a célhelyre, és gyorsabb ismételt letölteni azt.

A következő pontok röviden összefoglalják a legelterjedtebb állományszervezési elvet, a könyvtárstruktúrát. Ez után a műveletek tárgyalása következik, főként humán oldalról, majd pedig a további alternatív eszközök bemutatása.

2. A fa struktúra megjelenése és használata

A dokumentumok rendszerezése több okból is fa struktúra szerint történik. Ezek közül foglalnak össze néhányat az alábbi pontok.

- Szemléletes, a felhasználók számára természetesen adódik. Abból az egyszerű tényből kiindulva, hogy a rendszerezendő dolgok fizikailag egy helyen lehetnek csak következnek, hogy a csoportosítandó objektumok elrendezésének is hasonlóknak kell lennie.
- A könyvtári nyilvántartó rendszerek így épültek fel, többek között az előző okok miatt. A kialakuló informatikai rendszerek fokozatosan átvették ezeket az elemeket.
- A fa struktúra egyszerűen implementálható. Megjelenítése, benne a navigáció egyszerűen megoldható. Mutatók segítségével felépítve az elvégezhető általános műveletek (*új elem beszúrása, részfa áthelyezése, törlése*) konstans időben elvégezhető.

Az egyszerű elv hatékonysága sajnos a nyilvántartott állományok mennyiségi változása miatt romlott. Míg egy pár évtizede néhány száz elektronikus dokumentummal rendelkeztek a felhasználók (a kis kapacitású tárolók, és a lassú hálózati kapcsolatok miatt), addig ma már az ezres, tíz- vagy akár százezres nagyságrend is tipikusnak mondható. Ezek karbantartása, az ilyen méretű fájlrendszerekben a keresés lassú és nehézkes lehet, ha nem tudjuk pontosan, hogy hol található az állomány. Legrosszabb esetben, vagyis ha nem tudjuk az állománynak

sem a nevét, sem a formátumát, sem pedig a módosítás dátumát, akkor kénytelenek vagyunk a jegyzékstruktúra részfáját csomópontonként bejárni.

A fastruktúra említett problémáját taglalja a [1] cikk. Ebben a hierarchikus rendszerezés érzékeny pontjaként említi azt, hogy a bevett rendszerezési módszer nem alkalmas az átfedéssel/átlapolódó kategóriák kezelésére.

3. Elhelyezés és visszakeresés

A dokumentum kezelése során a két alapvető művelet a dokumentum adott helyen való elhelyezése, illetve a dokumentum visszakeresése. Érdemes megvizsgálni az ezekre fordított időt, amely tulajdonképpen jellemzi a dokumentumkezelés hatékonyságát.

A dokumentumok a számítógépünkre többféle módon is kerülhetnek, például újakat hozhatunk létre, háttértárról vagy az Internetről letölthetjük azokat. Az elhelyezésnél a leggyorsabb megoldás, ha a legközelebbi helyre másoljuk őket. Ez tipikusan lehet az asztal, a dokumentumok vagy a letöltések mappája. Ez a használati mód abban az esetben szerencsés, ha az új dokumentumokra gyakrabban van szükség, mint a régebbiekre.

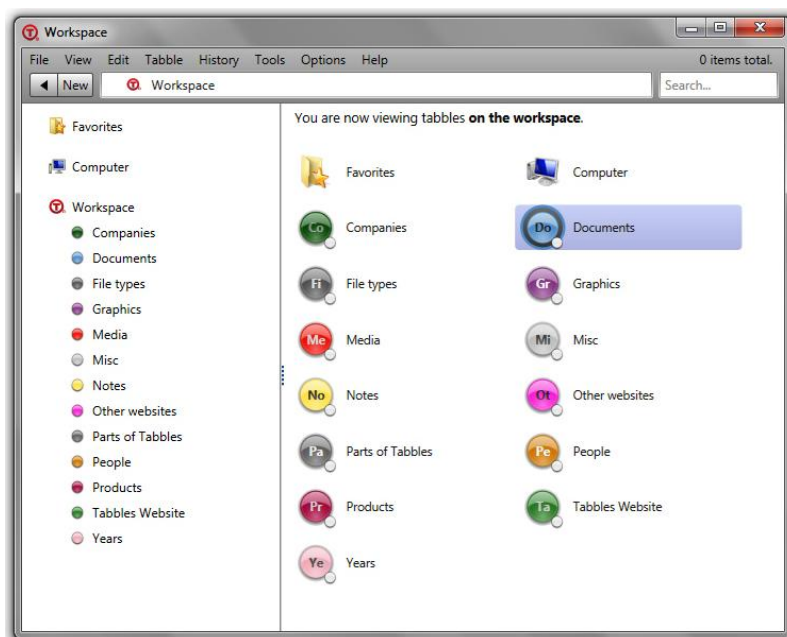
A másik véglet, ha saját taxonómiát szeretnénk kialakítani. Ilyenkor minden újonnan megjelenő dokumentum esetében pontosan be kell sorolni, hogy az hova tartozik, és annak megfelelően elhelyezni.

Különböző ajánlásokat, konvenciókat találhatunk, amelyek segítenek felépíteni a saját rendszerezési struktúráinkat, majd eligazodni benne [2]. Olyan lényeges dolgok szerepelnek ezek között, mint például, hogy a fájlok és jegyzékek elnevezése legyen minél rövidebb, tükrözze a tartalmat. Az elnevezési konvenciók szintén fontosak, illetve hogy a jegyzékstruktúra hasonló szerkezetű részei következetesen szerepeljenek.

Az átlagos, napi szintű állományhasználat mellett az archiválás további, nehezen megoldható problémákat hordoz. Az archívumok esetében az adatmentés, vagy a munkatér felszabadítása szokott lenni az elsődleges cél, magára az archívum rendszerezésére már ritkábban marad idő. Ez viszont azt eredményezi, hogy néhány fontos, de ritkán használt állomány megkeresése az archivált jegyzékeken belül kritikusan sokáig tarthat.

4. Jelenleg elérhető eszközök

Az alapvető fájlkezelő alkalmazásokon túl használhatjuk az asztali környezetbe integrált indexelő, kereső szolgáltatásokat (*Desktop Search*). A Windows esetében a Windows Desktop Search segítségével végezhető a keresés [3]. A keresési kifejezésben a fájlok neve, kiterjesztése, dátuma, szerzője vagy címkék szerepelhetnek. Az Apple OS X operációs rendszerén a Spotlight lát el ugyanilyen feladatokat [4]. A GNU/Linux operációs rendszerek esetében nincs egységes asztali környezet. A KDE-hez a NEPOMUK szemantikus desktop [5], és a Dolphin fájlkezelő alkalmazás [6] használható. A másik irányvonalon az Ubuntu most már alapértelmezetté vált Unity [7] felülete rendelkezik integrált kereső szolgáltatással.



1. ábra. A tabbles grafikus felülete a címkeként használható jegyzékekkel

Az előbbieken említett keresőszolgáltatások a dokumentumok visszakeresését segítik, de az állományok kezelése a felhasználó szemszögéből tulajdonképpen változatlan. Vannak a fa struktúrát bővítő lehetőségek, mint például a linkek és a virtuális könyvtárak. Ezek már strukturális változást jelentenek. Egy egyedi, de praktikus megoldást képvisel ezek közül a tabbles [8]. Ebben a speciális kerek ikonok címkeket jelölnek, amelyeket mappáknak is lehet tekinteni (1. ábra).

5. Háló alapú struktúra használata

Az említett problémákra az egyik megoldást a háló alapú struktúrák használata jelentheti. Ezek jóval általánosabbak, mint a jegyzékstruktúrák, így lehetőséget biztosítanak az átfedéses kategóriák kezelésére, illetve rugalmasabb navigációt tesznek lehetővé. A hálón (a fákhoz hasonlóan) megfogalmazhatók keresési kifejezések [9]. Ez a struktúra tehát a címkézés egy, a jelenlegi rendszerek működéséhez közelebb álló megvalósítása. A fájlrendszerben való navigációhoz parancssori eszközök is kényelmesen használhatók, illetve a fogalomháló alapú modell úgy van kialakítva, hogy a grafikus felület is könnyen átlátható, praktikus legyen.

Irodalomjegyzék

- [1] Margo Seltzer, Nicholas Murphy: Hierarchical File Systems are Dead, Harvard School of Engineering and Applied Sciences.
- [2] How-To Geek: Zen and the Art of File and Folder Organization, <http://www.howtogeek.com/howto/15677/zen-and-the-art-of-file-and-folder-organization>, Internet, 2014.
- [3] Windows asztali kereső, <http://support.microsoft.com/kb/917013/hu>, Internet, 2014.
- [4] Apple, Spotlight, <http://support.apple.com/kb/ht2531>, Internet, 2014.
- [5] NEPOMUK, <http://nepomuk.semanticdesktop.org>, Internet, 2014.
- [6] Dolphin fájlkezelő alkalmazás, <http://dolphin.kde.org>, Internet, 2014.
- [7] Unity, <https://unity.ubuntu.com>, Internet, 2014.
- [8] Tabbles, <http://tabbles.net>, Internet, 2014.
- [9] Piller Imre: Fogalom alapú állománykezelés, Diplomamunka, Miskolci Egyetem, 2013.

Felhő alapú dokumentumkezelő rendszerek

Cloud Based Document Management Systems

Piller Imre^a, Fegyverneki Sándor^b

^aMiskolci Egyetem, Alkalmazott Matematikai Intézeti Tanszék
piller@iit.uni-miskolc.hu

^bMiskolci Egyetem, Alkalmazott Matematikai Intézeti Tanszék
matfs@uni-miskolc.hu

Absztrakt: A felhő alapú szolgáltatások jelentős segítséget jelentenek a dokumentumok megosztásában. Itt minden olyan Interneten keresztül elérhető webalkalmazást annak tekinthetünk, amely alkalmas arra, hogy dokumentumokat feltöltsünk rá, illetve azt mások számára is elérhetővé tegyük. A nagyobb ilyen szolgáltatások például a Google Drive, SkyDrive és a DropBox. Ezek használatához elegendő regisztrálnunk és a szolgáltatások egy része már elérhető is lesz webböngészőn keresztül. Lehetőség van saját felhőszolgáltatás kialakítására is. Ez olyan esetekben lehet indokolt, ahol a szükséges hardverek rendelkezésre állnak és a kezelt állományok mennyisége már túllépné az ingyenes szolgáltatások kereteit. Ehhez használhatjuk például az OwnCloud, Sparkleshare, Seafile vagy Pydio alkalmazásokat. Érdeemes szót ejteni még a közösségi oldalak adta fájlmegosztási lehetőségekről. A közösségi oldalak ugyan nem elsődlegesen dokumentumok megosztására lettek kifejlesztve, de népszerűségüknek köszönhetően gyakran látnak el ilyen funkciókat is. A dolgozat célja kiemelni és összehasonlítani a rendszerek és használati módok fő jellemzőit, illetve saját megoldásokkal hatékonyabbá tenni azokat.

Kulcsszavak: tartalomkezelés, felhő alapú szolgáltatások, fájlmegosztó rendszerek

Abstract: The cloud based services has major importance in the sharing of documents. In this paper we consider all kind of webservices as a cloud based services which enables to upload documents and makes them available for other people. The main services are the Google Drive, SkyDrive and the DropBox. Enough to register on their webpages and many of the functions become accessible via webbrowser. We can install our own webservice for document management also. This is reasonable when the required hardware infrastructure is provided and the number of managed document is over the limit of the free services. We can use for example the OwnCloud, Sparkleshare, Seafile or Pydio applications. It is worth to consider the document sharing possibilities of the social networks. These are not designed primarily for document management but have similar features because they are frequently used in this role. The aim of the paper is to highlight and to compare the main features of these type of systems and propose some methods to make them more effective.

Keywords: content management, cloud services, file sharing

1. Bevezetés

A felhő alapú szolgáltatások használata már általánossá vált. A felhasználók egyre kevésbé érzik, és egyre kevésbé szükséges érezniük a különbséget a között, hogy egy alkalmazás helyben fut, vagy pedig egy szolgáltató távoli gépén. A webböngészőn keresztül elérhető szolgáltatások számos hasonlóságot mutatnak a példaként szolgáló korábbi natív alkalmazásokkal. Ez a használatukat nagyban megkönnyíti, mivel így nem szükséges új fogalmakat, módszereket megtanulniuk a felhasználóknak. Dokumentumok esetén például a hagyományos jegyzékstruktúrát ki tudják alakítani akár a saját gépükön lévő jegyzékstruktúra mintájára.

2. Vállalatok és felsőoktatási intézmények

A dokumentumok kezelése szerves részét képezi a vállalati és az intézményi működésnek. Mindkettőnek szüksége van arra, hogy a felhalmozott ismereteket biztonságos helyen tárolja, és szükség esetén minél gyorsabban elérhetővé tudja tenni.

A vállalatok számára a dokumentumkezelő rendszerekből nagy a választék. Szerepel közöttük alkalmazás, illetve szolgáltatás (*Software as a Service*). A vállalatok szemszögéből az utóbbi több előnnyel rendelkezik. Ezek közül a két legfontosabb, hogy a használatához kevesebb informatikust elegendő foglalkoztatni, illetve a használatuk megtanulása kevesebb időbe telik.

A nagyobb dokumentumkezelő rendszerek között a következőket érdemes megemlíteni: Alfresco [1], DocStar tartalomkezelő rendszere [2], digitalbucket.net [3], M-files [4], netdocuments [5], Cabinet [6], Box [7] illetve Spring CM rendszer [8]. Számos további elérhető alkalmazás van a felsoroltakon kívül [9].

Az oktatási intézmények esetében a szempontok kissé mások, mint a vállalatoknál. A hardveres erőforrások a nagyobb intézmények esetében rendelkezésre állnak. A problémát főként ezek kihasználtsága szokta okozni. A kifejezetten felsőoktatási intézmények számára készített dokumentumkezelő rendszerek piaca nem olyan jelentős, viszont akár a kisebb nyílt forrású rendszerek képesek lehetnek ellátni a szükséges feladatokat. Érdemes ezért saját kialakítású rendszerekkel foglalkozni.

3. Elterjedt felhő alapú szolgáltatások

A Google Drive [10] a Google saját tárhely és dokumentummegosztó szolgáltatása. A cég 2012-ben tette publikusan elérhetővé. Egy kézenfekvő lépés volt a következő okok miatt.

- A GMail levelezőrendszert egyre többen kezdték fájlok tárolására és megosztására használni, akár úgy, hogy saját maguknak küldtek el állományokat.
- Dokumentumkezeléshez már készen és aktív használatban volt a GoogleDocs, általános célú fájlmeosztó rendszere viszont még nem volt a Google-nek.
- A Google szolgáltatások integrációjának keretein belül lehetőség volt a fájlok kezelésével kapcsolatos funkciókat egy termékbe összefogni.

A SkyDrive (amely később a OneDrive nevet kapta) a Microsoft cég központosított tárhelyszolgáltatása [11]. A Skype felvásárlása után megnövekedett az igény a saját alkalmazások integrációjára.

A DropBox [12] egész más utat követett. Egy különálló fájlmegosztó szolgáltatásként indult, amely webes és mobil platformokon is elérhető volt. Népszerűségének köszönhetően verziókezelő és megjelenítő funkciók is kerültek bele.

A dokumentumok kezelésénél az alábbi szempontok játszanak fontos szerepet.

Tárolás: Összesen mennyi dokumentum tárolására van lehetőség. A tárolás milyen megbízhatósággal történik. A tárolási módhoz kapcsolódóan érdemes figyelembe venni a fel- és letöltési időket.

Keresés: Megkülönböztethetjük a metaadatokban és a tartalomban történő keresést. A metaadatok esetében például a fájlnevek, fontos műveletekhez köthető időpontok, jogosultságok használhatók fel.

A hatékonyság szempontjából nagyon fontos a keresőfelület kialakítása is. A Google példája jól mutatja, hogy nem feltétlenül kell a felületnek komplexnek lennie ahhoz, hogy a komplex funkcionalitást el lehessen érni.

Ma már lehetőség van teljes szöveges keresésre PDF fájlok, vagy akár képek esetében is. Ez egy bonyolult előfeldolgozási folyamatot feltételez. Képek indexelésére szintén vannak már példák.

Jogosultságkezelés: A dokumentumok másokkal való megosztása tulajdonképpen a láthatósági jogosultságok publikussá tételével egyezik meg. A jogosultságkezelésnél tipikusan csoportok, jogosultsági körök alakíthatók ki. A megvalósítása a többfelhasználós operációs rendszereknél használatos jogosultságkezelésnek feleltethető meg általában.

Szerkesztés: Az egyszerű állománymegosztó rendszerek esetében csak az állományok feltöltésére, letöltésére, esetleg metaadataik szerkesztésére van lehetőség. A napjainkban használt böngészők lehetővé teszik, hogy teljes értékű grafikus szerkesztőfelületet biztosítsanak a felhasználók számára.

Szinkronizálás: A felhő alapú technológia segítségével többen is el tudják érni egyazon dokumentumot. Ezzel a szolgáltatás kollaborációs eszközként is használhatóvá válik.

Verziókezelés: Egy szerzős esetben, de főként amikor többen szerkesztenek egy adott dokumentumot, a verziókezelés létfontosságú ahhoz, hogy követni, korrigálni lehessen a szerkesztési folyamatokat. Mivel a szolgáltatások nem kifejezetten informatikusoknak készültek, ezért csak az egyszerűbb verziókövetési módszerekre van szükség.

Az említett rendszerek közül nyilvánvalóan a leginkább kifinomult keresési módszerrel a Google rendelkezik. A Google Drive már támogatja képek keresését úgy, hogy a képen ábrázolt objektumok nevei szerepelnek a keresési kifejezésben.

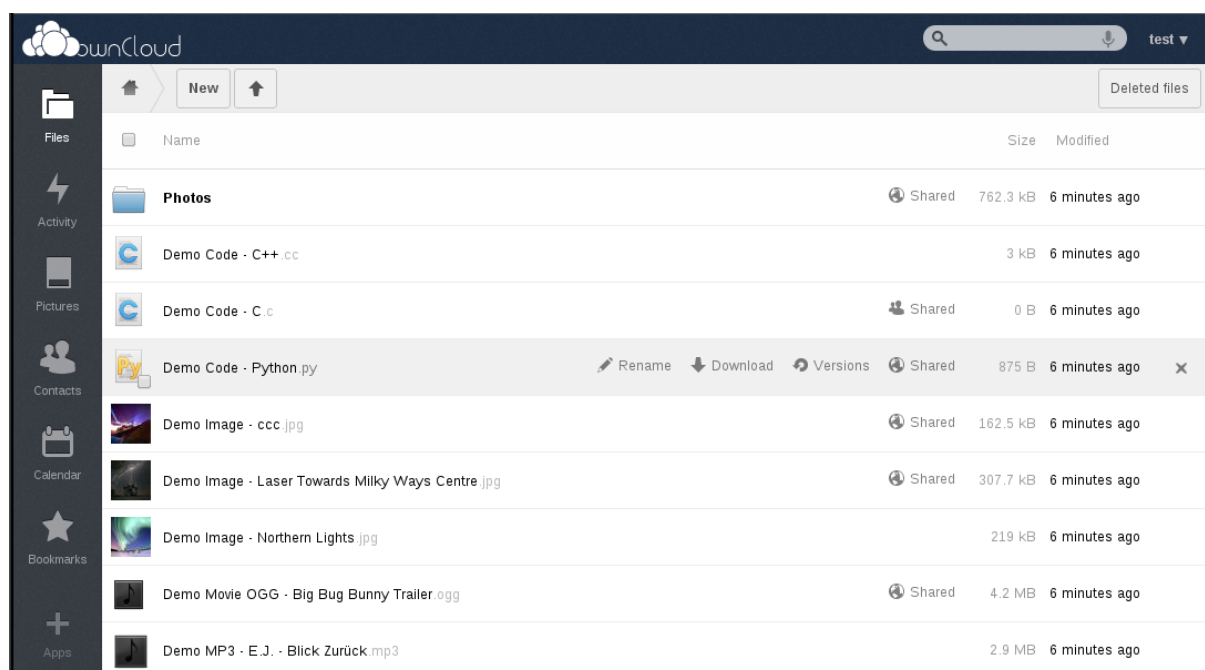
A szerkesztőfelületek, mint külön alkalmazások jelennek meg, amelyek képesek megnyitni a megfelelő formátumú állományokat. A OneDrive esetén az Office 365, míg a Google Drive esetében a Google Docs látja el ezt a feladatot.

Szinkronizálási és verziókezelési lehetőségekkel mindhárom szolgáltatás rendelkezik.

4. Saját rendszerek kialakítása

Kevésbé gyakori, de nagyobb vállalatok, intézmények számára szükséges lehet saját felhő alapú dokumentumkezelő rendszer kialakítására. Amennyiben a dokumentumkezelésnél speciális igények jelennek meg, akkor új alkalmazás kifejlesztésére van szükség, viszont vannak nyílt forráskódú, ingyenesen elérhető alkalmazások is, amellyel a gyakori feladatok megoldhatók.

Az OwnCloud [13] egy teljeskörű dokumentumkezelő rendszer számos kiegészítő funkcióval. A szerkesztéshez a LibreOffice alapértelmezett nyílt formátumát, az ODT (ODF Text) használja. Tartozik hozzá galéria és PDF fájlnézegető. A webes felületről az 1. ábrán láthatunk egy képet.



1.ábra. Az OwnCloud webes felülete

A Sparkleshare [14] egy kisebb alkalmazás. Egyszerűbben konfigurálható, mint az OwnCloud, ami főként annak az oka, hogy nem támogatja az alkalmazások, mint bővítmények telepítését és használatát.

A Seafile [15] szintén nyílt forráskódú, és a Sparkleshare-hez hasonlóan aránylag kevés funkciót támogat. Van üzleti változata is, amelyekbe viszont már belekerült a teljes szöveges keresés, Microsoft Office dokumentumok előnézeteinek létrehozása, idővonal a fájlok végzett műveletek követéséhez, illetve nagyobb klaszterek kialakításához szükséges funkciók.

A Pydio [16] (korábbi nevén AjaXplorer) segítségével webböngészőn keresztül szintén elérhetővé tehetők a fájlrendszerben lévő állományok. Az alkalmazás egyik egyedi

sajátossága, hogy részletes keresést tesz lehetővé, illetve hogy a keresések külön elmenthetők és szükség esetén újra végrehajthatók.

5. Közösségi oldalak funkciói és szerepük

A közösségi oldalak szerepe a magán- és üzleti jellegű kommunikációban, illetve az oktatásban fokozatosan nő. A dokumentumkezelés kapcsán két nagy közösségi oldalt érdemes említeni úgy, mint a FaceBook-ot és a Google+-t.

A FaceBook a dokumentumkezelést illetően a képekre és a videókra szorítkozik. Elterjedtsége miatt viszont gyakran jut fájlmegosztó-rendszer szerephez. Jelenleg még nem érhető el hozzá olyan szolgáltatás, amely lehetővé tenné a feltöltött tartalmakban való keresést.

A Google+ kevésbé elterjedt, viszont a Google Drive-nak köszönhetően közvetve támogatja a dokumentumkezelést is.

Ez tehát azt mutatja, hogy egy elterjedtebb, általános funkcionalitás előnyt tud élvezni a speciális célú alkalmazásokkal szemben, mivel a felhasználók jobban kötődnek egy adott szolgáltatáshoz.

6. Folyamatos integráció

A felsőoktatási intézmények szükségszerűen működésük során nagy mennyiségű dokumentumot halmoztak/halmoznak fel. Ezek tárolása és megjelenési módja kevésbé egységes, mint a vállalatoknál. Ez annak tudható be, hogy a vállalatoknál nincs akkora mozgástér a formátumok és egyéb módszerek kiválasztására, mint az oktatásban. Egy adott dokumentumformátumhoz ragaszkodni egyébként sem lenne célszerű. Az oktatóknak megvannak a saját jól bevált módszereik, amelyek oktatóként hatalmas eltéréseket mutathatnak.

További probléma, hogy egy adott cég terméke, vagy szolgáltatása komoly költségekbe kerül, ha azt a hallgatók számára elérhetővé szeretné tenni az intézmény. A cégek természetesen biztosítanak kedvezményeket a diákok részére, viszont a dokumentumkezeléssel járó problémák olyanok, amelyek rontják ugyan a hatékonyságot, viszont nem látványosak annyira, hogy magasabb prioritást élvezhessenek más finanszírozandó dolgokkal szemben.

Az informatikus hallgatók esetében szempontként még megjelenik továbbá az, hogy jobb, ha több, egymástól eltérő rendszert használnak, mivel nem lehet tudni biztosan, hogy később a cégeknél majd melyik ismeretére lesz szükségük.

Irodalomjegyzék

- [1] Alfresco, <http://www.alfresco.com/products/cloud>, Internet, 2014.
- [2] docStar Content Management, <http://www.docstar.com/document-management-systems/cloud>, Internet, 2014.
- [3] digitalbucket.net, <http://digitalbucket.net>, Internet, 2014.
- [4] M-files Inc, <http://www.m-files.com/en/cloud-vault>, Internet, 2014.
- [5] netdocuments, <http://www.netdocuments.com>, Internet, 2014.
- [6] Cabinet, Document Management Solutions, <http://cabinetpaperless.com/products/cabinet-online>, Internet, 2014.
- [7] Box, Document Management, <https://www.box.com/business/document-management>, Internet, 2014.
- [8] SpringCM, Content Cloud Services for the Enterprise, <https://www.springcm.com/products/what-springcm>, Internet, 2014.
- [9] ITBusinessEdge, <http://www.smallbusinesscomputing.com/News/Software/10-top-cloud-storage-services-for-smb.html>, Internet, 2014.
- [10] Google Drive, <https://drive.google.com>, Internet, 2014.
- [11] Microsoft OneDrive, <https://onedrive.live.com/about/hu-hu>, Internet, 2014.
- [12] DropBox, <https://www.dropbox.com>, Internet, 2014.
- [13] OwnCloud, Web services under your control, <http://owncloud.org>, Internet, 2014.
- [14] SparkleShare, <http://sparkleshare.org>, Internet, 2014.
- [15] Seafile, Next-Generation Open Source Cloud Storage, <http://seafile.com>, Internet, 2014.
- [16] Pydio, <http://pyd.io>, Internet, 2014.

Multiset számítások R-ben

Multiset computations in R

Takács Péter^a, Csajbók Zoltán Ernő^a, Mihálydeák Tamás^b

^aDebreceni Egyetem, Egészségügyi Kar
{takacs.peter, csajbok.zoltan}@foh.unideb.hu

^bDebreceni Egyetem, Informatika Kar
tamas.mihalydeak@inf.unideb.hu

Absztrakt: A klasszikus halmazelmélet nem különbözteti meg az azonos tulajdonságokkal rendelkező elemeket. A multihalmazok (multisets) elméletében viszont ez lehetséges. A számítógép-tudományon belül a membrán számítások elmélete az, amely a biológiai sejtek tulajdonságainak tanulmányozása révén fejleszt új számítási modelleket. A membrán számítások elosztott és párhuzamos modellekkel dolgoznak. A régiók és környezetük közötti kommunikáció (határtartományokon keresztül) fontos szerepet játszik a folyamatokban. A P rendszerek és a multihalmaz approximációs terek kombinációja elvezet a „membránhoz elég közel” fogalmához. Ebben az írásban néhány olyan R függvény kerül bemutatásra, amelyek lehetővé teszik a multihalmaz approximációs terekben (MAS) és membrán rendszerekben történő számításokat. A számítások korábbi közleményekben szereplő példákon keresztül kerülnek bemutatásra. Az eredmények igazolják az R programozási környezet alkalmazhatóságát a membrán számítások területén.

Kulcsszavak: R nyelv, multihalmazok, multihalmaz approximációs terek

Abstract: The classical set theory does not distinguish element with same properties. The theory of mutiset can manege this task. The membrane computing is an area within computer science which aims to develop a new computational model through the study of the characteristics of biological cells. Membrane computing deals with distributed and parallel computing models. Communication between the regions and their environment (via boundary) play an important role in the process. Combination of P systems with multiset approximation spaces leads to an abstract concept of ‘to be close enough to a membrane’. In this paper some R functions have been presented that allow to carry out some calculations in multiset approximation spaces (MAS) and membrane systems. The calculations are illustrated with the examples coming from earlier papers. The results presented in this paper prove the usability of the R program in the field of membrane calculations.

Keywords: R language, multisets, multiset approximation spaces

1. Bevezetés

A klasszikus halmazelmélet nem teszi lehetővé az azonos tulajdonságokkal rendelkező objektumok előfordulását egy halmazon belül. A multihalmazok elmélete (theory of multisets¹) azonban lehetőséget ad erre – multihalmazokban megengedett egy elem többszöri szerepeltetése [1] [2] [3]. Többek között a Gh. Păun által bevezetett membrán-elmélet is multihalmazokkal dolgozik [4]. Mihálydeák T. és Csajbók Z. E. a membránszámításokhoz kapcsolódva alakított ki egy olyan modellt, amelyben értelmezhető a „membránhoz elég közel” fogalma [5] [6]. A munka a Z. Pawlak által kidolgozott [7], mára már klasszikusnak nevezhető

¹ A leírás során sok esetben a használt fogalmak angol eredetije is szerepel zárójelek között, mivel a terület jelenlegi magyar terminológiai rendszere még nem minden ponton egységes.

elmélet multihalmazokra történő általánosítását használja – multiset approximációs tér (multiset approximation space - *MAS*). A kutatások egyik iránya a kapcsolódó *MAS* számítások gyors és pontos elvégzése.

A membránszámítások egyik gyakran alkalmazott, multihalmazokkal dolgozó programozási környezete a P-lingua [8]. Az R programozási rendszerhez is található olyan csomagok, amelyek lehetővé teszik a multihalmazok kezelését (sets csomag [9], roughsets csomag [10]). Ezen alkalmazások használata a *MAS* számítások során azonban nehézkes, viszonylag bonyolult. Jelen írásban bemutatásra kerül egy olyan R programnyelvben fejlesztett függvénykönyvtár, amely segítségével a *MAS* számítások egy része elvégezhető.

A következő, második fejezetben a multihalmaz pontosabb meghatározására és néhány kapcsolódó reláció és művelet leírására kerül sor. A bemutatott fogalmak megértését az R példák segítik. Az ezt követő fejezetek a Pawlak-féle elmélet és a *MAS* számítások eddig kidolgozott R implementációját tárgyalják – bemutatva a csatlakozó membrán számításokat. Az R programozási nyelven létrehozott függvényeket a melléklet tartalmazza, megfelelő iránymutató (angol nyelvű) kommentekkel.

2. Multihalmaz függvények R-ben

U legyen egy véges nem üres halmaz – nevezzük univerzumnak (universe). Egy M multihalmaz (multiset, mset) U felett egy leképezés, amely U elemeihez egy természetes számot rendel (\mathbb{N} a természetes számok halmazát jelöli):

$$M: U \rightarrow \mathbb{N}$$

Például, ha $a \in U$ és M egy multihalmaz, amelyben az a elem háromszor fordul elő, akkor a fenti leképezés a 3 értéket veszi fel: $M(a) = 3$. Ez a többszöri előfordulás gyakran felső indexet használva kerül jelölésre: a^3 . Amennyiben egynél több fajta elem is szerepel egy multihalmazban, a jelölés hasonlóan felső index alakban lehetséges. Például a $B = a^3b^2$ egy olyan multihalmaz, amelyben az a elem háromszor, a b elem pedig kétszer fordul elő. A gyakorlat a multihalmazban nem szereplő elemeket nem jelöli.

Szükséges az elmélet teljesebb leírásához a multihalmazok halmazának bevezetése. Jelölje $MS(U)$ az U univerzum feletti összes multihalmazok halmazát. Jelölje $MS^n(U)$ az összes olyan M multihalmaz halmazát U felett, amelyre $M(a) \leq n$ minden $a \in U$ esetén ($n \in \mathbb{N}$). Legyen $MS^{< \infty}(U) = \bigcup_{n=0}^{\infty} MS^n(U)$.

A multihalmazok fogalmának pontosítása után mód nyílik a kapcsolódó multihalmaz relációk és műveletek bemutatására. Az R nyelvi implementáció feltételezi, hogy az U univerzum rögzített, egy számítás során az nem változik. Például² legyen

$$U = \{ a, b, c, d, e \}.$$

R-ben ez az

```
> U <- c("a","b","c","d","e")
```

paranccsal rögzíthető. Itt a relációs jel (>) az R promptot (készenléti jel) jelöli, a többi rész pedig parancsként begépelendő. A parancs végrehajtása az *Enter* billentyű leütésével indul.

Ezen kívül az implementációban az is kötött, hogy a multihalmazok leírása Parikh vektor reprezentációt használt. Ez esetünkben azt jelenti, hogy az univerzum azon elemei, amelyek ép-

² A példa a 2014-ben megjelent [11]-es forrás számításait követi. Melléklet – Example 1.

pen nem szerepelnek egyszer sem a multihalmazban nulla kitevővel jelöltek. Így egyértelmű kapcsolat alakítható ki az R-beli vektorforma és a multihalmazok között.

Legyen például a $W1$ multihalmaz a következő: $W1 = ce^5$. Ennek R-beli reprezentációja az $a^0b^0c^1d^0e^5$ alakon keresztül érhető el. Az R-beli parancs:

```
> W1 <- c(0,0,1,0,5)
```

Néhány további példa (az előzőt is ismételve), amelyek később még felhasználásra kerülnek³:

R-kódok	Multihalmazok jelölése	
> W1 <- c(0,0,1,0,5)	$a^0b^0c^1d^0e^5$	ce^5
> W2 <- c(3,2,0,1,0)	$a^3b^2c^0d^1e^0$	a^3b^2d
> W3 <- c(1,2,3,4,0)	$a^1b^2c^3d^4e^0$	$ab^2c^3d^4$
> W4 <- c(3,1,0,0,0)	$a^3b^1c^0d^0e^0$	a^3b
> W5 <- c(1,1,2,3,0)	$a^1b^1c^2d^3e^0$	abc^2d^3
> B1 <- c(2,0,0,0,0)	$a^2b^0c^0d^0e^0$	a^2
> B2 <- c(1,1,0,0,0)	$a^1b^1c^0d^0e^0$	ab
> B3 <- c(0,1,0,0,0)	$a^0b^1c^0d^0e^0$	b
> B4 <- c(0,0,1,1,1)	$a^0b^0c^1d^1e^1$	cde
> C1 <- c(3,2,0,1,0)	$a^3b^2c^0d^1e^0$	a^3b^2d
> C2 <- c(2,3,0,1)	$a^2b^3c^0d^1$	a^2b^3d

A multihalmazok R-beli megvalósításakor követelményként előírt reprezentáció szükségessé teszi a vektorok többszöri ellenőrzését. Erre az **mcheck** függvény szolgál. Ez egy egyszerű, vektor-hosszakat vizsgáló függvény, amely minden reláció és művelet végrehajtásakor meghívásra kerül.

mcheck(mS, SU) - ellenőrzés

Paraméterek: mS multihalmaz; SU univerzum

Leírás: Ez a függvény ellenőrzi az mS multihalmaz R-beli Parikh vektor reprezentációjának elemszámát (hosszát). Amennyiben az mS elemszáma megegyezik az SU univerzum elemszámával, akkor a függvény 1-es értékkel tér vissza, máskülönben a függvény visszatérési értéke 0.

*Példa:*⁴

```
> mcheck(W1,U)
[1] 1
> mcheck(C2,u)
[1] 0
```

A következő függvények multihalmaz relációkat valósít meg.

³ A közölt példák a későbbiekben a következő jelentéssel bírnak: $W1$; ...; $W5$ membrán régiókat jelölő multihalmazok; $B1$; ...; $B4$ bázis multihalmazok; $C1$ egy multihalmaz U felett; $C2$ egy multihalmaz egy másik U' univerzum felett.

⁴ A függvények leírásában szereplő példák először minden esetben az R parancsok kerülnek leírásra – a '>' kénszenléti jellel kezdődően. Ezután (új sorban) szerepelnek a végrehajtás eredményei, amit általában a '[1]' jel indít. A '# ' jel utáni részek megjegyzések, a R nem veszi figyelembe ezeket – a kiegészítések a könnyebb ellenőrizhetőséget, az eredmények bemutatását segítik.

mequal(mS1, mS2, SU) - egyenlőség

Paraméterek: $mS1$ multihalmaz; $mS2$ multihalmaz; SU univerzum

Leírás: Ez a függvény két multihalmaz egyenlőségét vizsgálja.

Definíció szerint $mS1 = mS2$, ha $mS1(a) = mS2(a)$ minden $a \in U$ esetén.

A függvény visszatérési értéke 1 , amennyiben a két multihalmaz egyenlő és 0 , amennyiben nem.

Példa:

```
> mequal(W1,W2,U)           # W1(0,0,1,0,5); W2(3,2,0,1,0)
[1] 0
> mequal(W2,C1,U)          # W2(3,2,0,1,0); C1(3,2,0,1,0)
[1] 1
```

mpartof(mS1, mS2, SU) - tartalmazás

Paraméterek: $mS1$ multihalmaz; $mS2$ multihalmaz; SU univerzum

Leírás: Definíció szerint a multihalmazok tartalmazási relációja a következő. Legyenek $mS1$ és $mS2$ multihalmazok U felett. $mS1 \sqsubseteq mS2$, ha $mS1(a) \leq mS2(a)$, minden $a \in U$ esetén. Ez az R függvény ezt a relációt vizsgálja. A függvény visszatérési értéke 1 , amennyiben $mS1$ része $mS2$ -nek; és 0 a visszatérési érték, ha $mS1$ nem része $mS2$ -nek.

Példa:

```
> mpartof(W1,W2,U)         # W1(0,0,1,0,5); W2(3,2,0,1,0)
[1] 0
> mpartof(W4,W2,U)        # W4(3,1,0,0,0); W2(3,2,0,1,0)
[1] 1
```

gmpartof(mS1, mS2, le) - tartalmazás

Paraméterek: $mS1$ multihalmaz; $mS2$ multihalmaz; le multihalmazok elemszáma

Leírás: Az $mpartof$ függvény módosított verziója. A bemeneti érték nem az univerzum tulajdonságaihoz kötött, hanem egy elemszámot megadó természetes szám.

Példa:

```
> gmpartof(W1,W2,5)        # W1(0,0,1,0,5); W2(3,2,0,1,0)
[1] 0
> gmpartof(W4,W2,5)        # W4(3,1,0,0,0); W2(3,2,0,1,0)
[1] 1
```

Most következzen néhány multihalmaz művelet.

mintersec(mS1, mS2, SU) - közös rész, metszet

Paraméterek: $mS1$ multihalmaz; $mS2$ multihalmaz; SU univerzum

Leírás: A függvény két multihalmaz közös részét határozza meg.

Definíció szerint $mS1 \sqcap mS2 = M \in MS(U)$, ahol $M(a) = \min \{mS1(a); mS2(a)\}$; minden $a \in U$ esetén.

Példa:

```
> mintersec(W2,W3,U)      # W2(3,2,0,1,0); W3(1,2,3,4,0)
[1] 1 2 0 1 0             # ab2d
> mintersec(B1,B2,U)     # B1(2,0,0,0,0); B2(1,1,0,0,0)
[1] 1 0 0 0 0             # ab
```

munion(mS1, mS2, SU) – egyesítés, unió

Paraméterek: mS1 multihalmaz; mS2 multihalmaz; SU univerzum

Leírás: A függvény két multihalmaz unióját számítja ki.

Definíció szerint $mS1 \sqcup mS2 = M \in MS(U)$, ahol $M(a) = \max \{mS1(a); mS2(a)\}$; minden $a \in U$ esetén.

Példa:

```
> munion(W4,W5,U)           # W4(3,1,0,0,0); W5(1,1,2,3,0)
[1] 3 1 2 3 0                # a3bc2d3
> munion(B2,B4,U)           # B2(1,1,0,0,0); B4(0,0,1,1,1)
[1] 1 1 1 1 1                # abcde
```

madd(mS1, mS2, SU) - addíció, összeadás

Paraméterek: mS1 multihalmaz; mS2 multihalmaz; SU univerzum

Leírás: Értelmezett a multihalmazok összege. Definíció szerint:

$mS1 \oplus mS2 = M \in MS(U)$, ahol $M(a) = mS1(a) + mS2(a)$; minden $a \in U$ esetén.

Példa:

```
> madd(W4,W5,U)             # W4(3,1,0,0,0); W5(1,1,2,3,0)
[1] 4 2 2 3 0                # a4b2c2d3
> madd(B2,B4,U)             # B2(1,1,0,0,0); B4(0,0,1,1,1)
[1] 1 1 1 1 1                # abcde
```

mdiff(mS1, mS2, SU) - differencia, különbség

Paraméterek: mS1 multihalmaz; mS2 multihalmaz; SU univerzum

Leírás: Két multihalmaz különbségénél figyelni kell a negatív értékű eredmények kiszűrésére.

$mS1 \ominus mS2 = M \in MS(U)$, ahol $M(a) = \max \{mS1(a) - mS2(a); 0\}$; minden $a \in U$ esetén.

Példa:

```
> mdiff(W3,W2,U)            # W3(1,2,3,4,0); W2(3,2,0,1,0)
[1] 0 0 3 3 0                # c3d3
> mdiff(W5,B4,U)            # W5(1,1,2,3,0); B4(0,0,1,1,1)
[1] 1 1 1 2 0                # abcd2
```

Végezetül következzen egy újabb reláció, az n-szeres tartalmazás relációja.

mnpartof(mS1, mS2, SU)

Paraméterek: mS1 multihalmaz; mS2 multihalmaz; SU univerzum

Leírás: Legyenek $mS1 \neq mS2$ multihalmazok U felett. $mS1 \sqsubseteq^n mS2$ ha

$\oplus_n mS1 \sqsubseteq mS2$, de $\oplus_{n+1} mS1 \not\sqsubseteq mS2$ minden $a \in U$ esetén.⁵ Az R függvény visszatérési értéke 0, ha mS1 nem része mS2-nek. Amennyiben mS1 n-szeres része (legalább egyszeres) mS2-nek, akkor a visszatérési érték n.

⁵ Az n-szeres multihalmaz összeg pontos definícióját illetően [12] forrásra hivatkozunk.

$\oplus_0 M = 0$; $\oplus_1 M = M$; $\oplus_{n+1} M = \oplus_n M \oplus M$.

Példa:

```
> mnpartof(W2,W4,U)          # W2(3,2,0,1,0); W4(3,1,0,0,0)
[1] 0
> mnpartof(W4,W2,U)          # W4(3,1,0,0,0); W2(3,2,0,1,0)
[1] 1
> mnpartof(B2,W2,U)          # B2(1,1,0,0,0); W2(3,2,0,1,0)
[1] 2
```

A multihalmaz relációk és műveletek R-beli interpretációja ezen a szinten lezárható. A további munkához elegendők ez eddigi eredmények.

3. A Pawlak-féle multihalmaz approximációs tér

A multihalmaz relációk és műveletek rövid áttekintése után áttérünk az (általános) multihalmaz approximációs terek (general multiset approximation spaces, MAS) elméletének és R-beli interpretálásának bemutatására. A precíz és egzakt matematikai definíciókat, kapcsolódó tételeket többek között [11] és [12] források részletesen tárgyalják; itt csak egy rövid összegző áttekintés szerepel – az R implementációt szem előtt tartva.

Legyen U ismét egy véges nem üres halmaz, az univerzum. Tekintsünk egy \mathcal{B} bázisrendszert U felett, amely a B_1, B_2, \dots, B_n multihalmazokból áll. A \mathcal{B} bázisrendszeren alapuló, definiálható multihalmazok a multihalmaz összeadás \oplus műveletére támaszkodva alakíthatók ki. Jelölje ezen definiálható multihalmazok halmazát $\mathcal{D}_{\mathcal{B}}$. A MAS rendszerek egyik feladata, hogy egy tetszőleges U feletti M multihalmazt leírjunk – közelítsünk, approximáljunk – a $\mathcal{D}_{\mathcal{B}}$ elemeinek segítségével. Az elmélet egy $\langle l, u \rangle$ függvénypárt (weak approximation functions, gyenge közelítő függvények) alakít ki ebből a célból. l az alsó, u pedig a felső approximációs függvény (lower approximation function; upper approximation function). A felsorolt összetevők alkotják együtt a $MAS(U)$ multiset approximációs teret (multiset approximation space)

$$MAS(U) = \langle MS^{<\infty}(U), \mathcal{B}, \mathcal{D}_{\mathcal{B}}, l, u \rangle$$

A $MAS(U)$ sok érdekes és fontos tulajdonsága vizsgálható, de számunkra a legfontosabb eset a Pawlak-féle multiset approximációs tér. Ebben az esetben, ha $M \in MS^{<\infty}(U)$ tetszőleges multihalmaz U felett, akkor l és u speciális alakban írható fel:

$$l(M) = \bigsqcup \{ \oplus_n B \mid n \in \mathbb{N}^+, B \in \mathcal{B} \text{ and } B \sqsubseteq^n M \}$$

$$u(M) = l(M) \sqcup b(M)$$

ahol

$$b(M) = \bigsqcup \{ \oplus_n B \mid B \in \mathcal{B}, B \not\sqsubseteq M, B \sqcap M \neq 0 \text{ and } B \sqcap M \sqsubseteq^n M \}$$

az M multihalmaz határtartománya. A fenti eredményekhez kapcsolódó R függvények a következők.

plow(M, BASE, SU)

Paraméterek: M multihalmaz, $BASE$ bázisrendszer (mátrix), SU univerzum

Leírás: Ez a függvény az M multihalmaz alsó közelítését számítja a $BASE$ bázisrendszer alapján.

Példa: A függvény meghívása előtt rögzíteni kell a bázisrendszert – a báziselemek n számát és magát a bázist:

```
> n <- 4
> Bázis <- c(B1,B2,B3,B4)
> B <- matrix(Bázis, nrow=n, ncol=length(U), byrow=T)
```

Ezután hívható meg a *plow* függvény:

```
> plow(W1,B,U)
[1] 0 0 0 0 0 # ures multiset
> plow(W2,B,U)
[1] 2 2 0 0 0 # a2b2
> plow(W3,B,U)
[1] 1 2 0 0 0 # ab2
> plow(W4,B,U)
[1] 2 1 0 0 0 # a2b
> plow(W5,B,U)
[1] 1 1 0 0 0 # ab
```

pbound(M, BASE, SU)

Paraméterek: M multihalmaz, $BASE$ bázisrendszer (mátrix), SU univerzum

Leírás: Ez a függvény az M multihalmazhoz tartozó határtartományt számítja a $BASE$ bázisrendszer alapján.

Példa: A függvény hívása előtt *plow* függvény leírásakor bemutatott inicializáló lépéseket végre kell hajtani – ha azok nem voltak már korábban végrehajtva.

```
> pbound(W1,B,U)
[1] 0 0 1 1 1 # cde
> pbound(W2,B,U)
[1] 0 0 1 1 1 # cde
> pbound(W3,B,U)
[1] 2 0 3 3 3 # a2c3d3e3
> pbound(W4,B,U)
[1] 0 0 0 0 0 # ures multiset
> pbound(W5,B,U)
[1] 2 0 2 2 2 # a2c2d2e2
```

pupp(M, BASE, SU)

Paraméterek: M multihalmaz, $BASE$ bázisrendszer (mátrix), SU univerzum

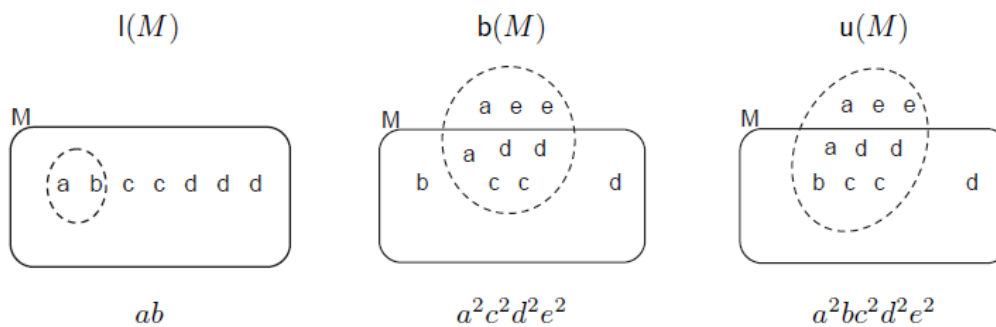
Leírás: A függvény az M multihalmaz felső közelítését számítja a $BASE$ bázisrendszer alapján. A függvény a *plow* és *pbound* függvények eredményeit használja a számítás során.

Példa: A függvény hívása előtt *plow* függvény leírásakor bemutatott inicializáló lépéseket végre kell hajtani – ha azok nem voltak már korábban végrehajtva.


```

> pupp(W1,B,U)
[1] 0 0 1 1 1           # cde
> pupp(W2,B,U)
[1] 2 2 1 1 1           # a2b2cde
> pupp(W3,B,U)
[1] 2 2 3 3 3           # a2b2c3d3e3
> pupp(W4,B,U)
[1] 2 1 0 0 0           # a2b
> pupp(W5,B,U)
[1] 2 1 2 2 2           # a2bc2d2e2
    
```

Az R reprezentáció eredményeit ellenőrzésképpen összevethetők a már közölt számítási eredményekkel [11]. Az alsó, felső közelítések ábrázolását az 1. ábrán láthatjuk.



1. ábra – Az $M = abc^2d^3$ multihalmaz approximációi.⁶

4. Határok számítása membrán rendszerekben

Ebben a fejezetben a multiset approximációs terek és a membrán rendszerek (P rendszerek) közötti kapcsolat kerül bemutatásra. A továbbiakban szükséges, a membrán rendszerekkel kapcsolatos fogalmak és eredmények az [4] forrásban találhatóak.

Egy V véges ábécé feletti, m csomópontot magába foglaló IP rendszer meghatározása tartalmazza magát a V -t, a régiók μ hierarchikus struktúráját, a w_i ($i=1, \dots, m$) multihalmazokat és a régiókhöz rendelt R_i ($i=1, \dots, m$) szabályhalmazok véges halmazát. Jelölje $MAS(IP)$ a V ábécé (mint univerzum) feletti Pawlak-féle multiset approximációs teret amit *kapcsolódó membrán approximációs térnek* (joint membrane approximation space) nevezük. Az elmélethez igazodva értelmezhetők azok a határtartományok, amelyek a membrán körül az „elegendően közel” fogalom kialakítását teszik lehetővé.

A w_i ($i=1, \dots, m$) multihalmazok régiók esetén definiálunk $bnd(w_i)$ membrán határokat (membrane boundaries), külső $bnd^{out}(w_i)$ határokat (outer membrane boundaries) és belső $bnd^{in}(w_i)$ határokat (inner membrane boundaries). A definíciók képletei a következők (a $w_{parent(i)}$ alsó index a membránstruktúrához kapcsolódó tartalmazási viszonyokat jelöli; bővebben az implementáció leírásánál):

⁶ Az ábra forrása: T. Mihálydeák and Z. E. Csajbók: On the membrane computations in the presence of membrane boundaries. Megjelenés alatt.

$$bnd(\omega_i) = \sqcup \{ \oplus_{N(B,i)} B \mid B \in \mathcal{B} \}$$

ahol

$$N(B, i) = \begin{cases} 0, & \text{if } B \sqsubseteq \omega_i \text{ or } B \sqcap \omega_i = \emptyset; \\ n, & \text{if } i = 1 \text{ and } B \sqcap \omega_1 \sqsubseteq^n \omega_1; \\ \min\{k, n \mid B \sqcap \omega_i \sqsubseteq^k \omega_i, \text{ and } B \ominus \omega_i \sqsubseteq^n \omega_{parent(i)}\}, & \text{otherwise} \end{cases}$$

$$bnd^{out}(\omega_i) = bnd(\omega_i) \ominus \omega_i$$

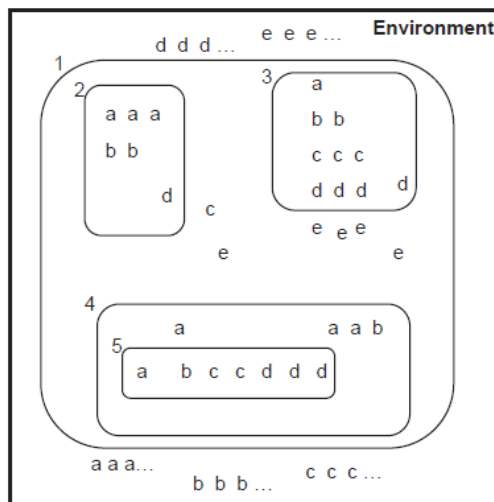
$$bnd^{in}(\omega_i) = bnd(\omega_i) \ominus bnd^{out}(\omega_i)$$

A fenti függvények R-beli realizációja több lépésben történik. Az első R függvény az $N(B, i)$ összefüggés egy részét számítja ki.

NB(REGION, i, BASE, j, SU, SMU)

Paraméterek: REGIO membrán régiók (mátrix); i mely régió esetén történjen a számítás; BASE bázisrendszer (mátrix); j mely báziselemre történjen a számítás; SU univerzum (ábécé); SMU membrán struktúra vektor

Leírás: Az NB függvény az $N(B, i)$ összefüggés értékét határozza meg rögzített régió (i) és rögzített bázis elem (j) esetén. A membránstruktúrát az SMU vektor írja le. Az SMU vektor i . eleme a i . membrán régió szülő régióját tartalmazza. Például a (0,1,1,4) SMU vektor legfelső régiója (első vektorelem, nincs szülője) három régiót tartalmaz (második, harmadik és negyedik vektorelem). Az ötödikként jelölt régió szülő régiója a negyedik régió (2. ábra).



2. ábra – Az $M = abc^2d^3$ multihalmaz approximációi.⁷

⁷ Az ábra forrása: T. Mihálydeák and Z. E. Csajbók: On the membrane computations in the presence of membrane boundaries. Megjelenés alatt.

Példa: A függvény futtatásához előkészítő lépések szükségesek. Rögzíteni kell a bázisrendszert – a báziselemek n számát és magát a bázist (ez a *plow* függvény leírásánál már szerepelt). Ezenkívül rögzíteni kell a régiók számát (m); ki kell alakítani a régiók leírását multihalmazokból képzett mátrix rögzítésével (R); valamint meg kell adni a membránstruktúrát leíró vektort (SMU).

```
> m <- 4
> Regio <- c(W1,W2,W3,W4,W5)
> R <- matrix(Regio, nrow=m, ncol=length(U), byrow=T)
> MU <- c(0,1,1,1,4)
```

Ezután hívható meg a *plow* függvény:

```
> NB(R,5,B,1,U,MU)
[1] 0
```

Az NB függvényt alkalmazva már könnyen kiszámíthatók a határok meghatározásához szükséges $N(B,i)$ értékek.

NBi(i)

Paraméterek: i mely régióra történjen a számítás

Leírás: A függvény kiszámítja az $N(B,i)$ összefüggés értékét rögzített i régióra (az összes báziselemen futtatva az NB R -beli függvényt).

Példa:

```
> NBi(1)
[1] 0 0 0 1
> NBi(2)
[1] 0 0 0 1
> NBi(3)
[1] 0 0 0 3
> NBi(4)
[1] 0 0 0 0
> NBi(5)
[1] 1 0 0 0
```

Az $N(B,i)$ értékeinek ismerete már lehetővé teszi a határregiók számítását

bnd(REGION, i, BASE, SU)

Paraméterek: $REGIO$ membrán régiók (mátrix); i mely régió esetén történjen a számítás; $BASE$ bázisrendszer (mátrix); SU univerzum (ábécé)

Leírás: A megadott i . régió határát számítja ki a függvény.

Példa:

```
> bnd(R,1,B,U)
[1] 0 0 1 1 1 # cde
> bnd(R,2,B,U)
[1] 0 0 1 1 1 # cde
> bnd(R,3,B,U)
[1] 0 0 3 3 3 #c3d3e3
```

```
> bnd(R,4,B,U)
[1] 0 0 0 0 0 # ures multiset
> bnd(R,5,B,U)
[1] 2 0 0 0 0 # a2
```

bndout(REGION, i, BASE, SU)

Paraméterek: REGIO membrán régiók (mátrix); i mely régió esetén történjen a számítás; BASE bázisrendszer (mátrix); SU univerzum (ábécé)

Leírás: A függvény az i. régió külső határtartományát számítja.

Példa:

```
> bndout(R,1,B,U)
[1] 0 0 0 1 0 # d
> bndout(R,2,B,U)
[1] 0 0 1 0 1 # ce
> bndout(R,3,B,U)
[1] 0 0 0 0 3 # e3
> bndout(R,4,B,U)
[1] 0 0 0 0 0 # ures multiset
> bndout(R,5,B,U)
[1] 1 0 0 0 0 # a
```

bndin(REGION, i, BASE, SU)

Paraméterek: REGIO membrán régiók (mátrix); i mely régió esetén történjen a számítás; BASE bázisrendszer (mátrix); SU univerzum (ábécé)

Leírás: A függvény az i. régió belső határtartományát számítja.

Példa:

```
> bndin(R,1,B,U)
[1] 0 0 1 0 1 # ce
> bndin(R,2,B,U)
[1] 0 0 0 1 0 # d
> bndin(R,3,B,U)
[1] 0 0 3 3 0 # c3d3
> bndin(R,4,B,U)
[1] 0 0 0 0 0 # ures multiset
> bndin(R,5,B,U)
[1] 1 0 0 0 0 # a
```

5. Kommunikációs szabályok

A P rendszerekben a számítások egy kiinduló állapotból, diszkrét lépéseken keresztül, egy végállapotig tartanak. Egy több régióból álló rendszerben minden lépésben minden régióra történnek számítások a kommunikációs szabályok által vezérelve. A szabályok alkalmazása maximálisan párhuzamos (maximal parallel) módon, nem-determinisztikusan történik [4]. A számítások akkor érnek véget, amikor már nem értelmezhető végrehajtandó szabály.

A kapcsolódó membrán approximációs tér esetén, az általános elmélettől eltérően a szabályok a régiók határán hajtandók végre – a *bnd*; *bndin*; *bndout* függvények által meghatározott membrán határokon.

A szabályok három típusba sorolhatók. Egy szabályt u -val, egy régiót w_i -vel jelölve a következők lehetségesek

- $\langle u, in \rangle$ alakú szabály; amely akkor alkalmazható, ha $u \sqsubseteq bnd^{out}(w_i)$ teljesül.
- $\langle u, out \rangle$ alakú szabály; amely akkor alkalmazható, ha $u \sqsubseteq bnd^{in}(w_i)$ teljesül.
- $\langle u, in; v, out \rangle$ alakú szabály; amely akkor alkalmazható, ha $u \sqsubseteq bnd^{out}(w_i)$ és $v \sqsubseteq bnd^{in}(w_i)$ teljesül.

A továbbiakban csak a második eset kerül tárgyalásra. Ennek elméleti háttere a már többször is hivatkozott [11] és [12] forrásokban található meg. A számítások során az első lépések között szerepel a kommunikációs szabályok multihalmazainak az összegyűjtése, amelyek alkalmazhatók az adott régió esetén. Ezt az R megvalósításban az *Appl* függvény számítja. A függvény eredményét egy tömb formában veszi át az *ApplMax* függvény, amely a maximális párhuzamos módnak megfelelően szelektálja a végrehajtható szabályokat. Az *ApplMax* függvény outputja a soronként leírja, hogy a kommunikációs szabályok milyen módon, hányszor hajtandók végre. Minden sor egy-egy lehetséges végrehajtási esetet jelent.

A következő függvényleírásokban szereplő példák a [12] közleményben megjelent példákat számítják (Melléklet – Example 2.).

Appl(nr,SQ,M,SU)

Paraméterek: nr szabályok száma; SQ szabálmátrix; M régió; SU univerzum

Leírás: A függvény az alkalmazható kommunikációs szabályokat gyűjti össze.

Példa:

A, Az alkalmazható kommunikációs szabályok összegyűjtése hagyományos számítási környezetben. A régió: $a^2b^{11}c^3d^9e$

```

> R                                     # régió
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    2   11    3    9    1    0
> Q                                     # szabályok
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    0    1    0    0    0
[2,]    0    6    0    6    0    0
[3,]    0    0    0    3    1    0
> Appl(r,Q,R,U)                         # alkalmazható szabályok
      [,1] [,2] [,3]
[1,]    0    0    1      # r3
[2,]    0    1    0      # r2
[3,]    0    1    1      # r2r3
[4,]    1    0    0      # r1
[5,]    1    0    1      # r1r3
[6,]    1    1    0      # r1r2
[7,]    1    1    1      # r1r2r3
[8,]    2    0    0      # r12
[9,]    2    0    1      # r12r3
[10,]   2    1    0      # r12r2
[11,]   2    1    1      # r12r2r3
    
```

B, Az alkalmazható kommunikációs szabályok összegyűjtése membrán határok esetén. A régió: $a^2b^{11}c^3d^9e$

```

> bnd(R,1,B,U)                         # a régióhoz tartozó határ
[1] 1 9 1 6 3 3                         # ab9cd6e3f3
    
```

```

> L <- bndin(R,1,B,U)           # belső határtartomány
> L
[1] 1 9 1 6 1 0                #  $ab^9cd^6e$ 
> bndout(R,1,B,U)              # külső határtartomány
[1] 0 0 0 0 2 3                #  $e^2f^3$ 
> Appl(r,Q,L,U)                # alkalmazható szabályok
      [,1] [,2] [,3]          # belső határtartományban (out szab.)
[1,]  0    0    1            #  $r_3$ 
[2,]  0    1    0            #  $r_2$ 
[3,]  1    0    0            #  $r_1$ 
[4,]  1    0    1            #  $r_1r_3$ 
[5,]  1    1    0            #  $r_1r_2$ 

```

ApplMax(nr,SQ,M,SU)

Paraméterek: nr szabályok száma; SQ szabálymátrix; M régió; SU univerzum

Leírás: A függvény az alkalmazható kommunikációs szabályok közül a maximálisan párhuzamosan futtathatókat gyűjti össze.

Példa:

A, A maximális párhuzamos mód szabályai hagyományos számítási környezetben – lásd Appl függvény példája A, rész.

```

> ApplMax(r,Q,R,U)
[1] 2 1 1                       #  $r_1^2r_2r_3$ 

```

B, A maximális párhuzamos mód szabályai membrán határok esetén – lásd Appl függvény példája B, rész.

```

> L <- bndin(R,1,B,U)
> L
[1] 1 9 1 6 1 0                #  $ab^9cd^6e$ 
> ApplMax(r,Q,L,U)
      [,1] [,2] [,3]
[1,]  1    0    1            #  $r_1r_3$ 
[2,]  1    1    0            #  $r_1r_2$ 

```

Az A, esetben az $a^2b^{11}c^3d^9e$ régió az $r_1^2r_2r_3$ szabályok alkalmazásával b^5c régióvá alakul.

A B, esetben már két „lefutása” lehet a szabályok alkalmazásának.

Az egyik szabályhalmaz (r_1r_2) alkalmazása azt eredményezi, hogy az ab^9cd^6e belső határtartományból kiválik az ab^6cd^6 multihalmaz, és belép a külső határtartományba.

A belső tartomány a művelet után: b^3e .

A másik szabályhalmaz (r_1r_3) alkalmazása azt eredményezi, hogy az ab^9cd^6e belső határtartományból kiválik az acd^3e multihalmaz, és belép a külső határtartományba. A belső tartomány a művelet után: b^9d^3 .

6. Összegzés

A bemutatott R függvények a jelzett forrásokban leírt multihalmaz relációkat, műveleteket, alsó és felső közelítéseket, membrán határokat és az alkalmazható kommunikációs szabályokat számítják. A számítások a [11] és [12] forrásokban közölt példákon keresztül az olvasó számára is ellenőrizhetők. További feladatot jelent az R implementáció számára az $\langle u, in \rangle$ és

az $\langle u, in; v, out \rangle$ szabályok megvalósítása. A közölt eredmények bizonyítják az R programozási nyelv alkalmazhatóságát a membrán számítások területén.

Irodalomjegyzék

- [1] A. Syropoulos: Mathematics of Multisets. Multiset Processing, Mathematical, Computer Science, and Molecular Computing Points of View, Workshop on Multiset Processing, WMP 2000, Curtea de Arges, Romania, August 21-25, 2000, 347–358.
- [2] D. Singh, A. M. Ibrahim, T. Yohanna and J. N. Singh: An overview of the applications of multisets. *Novi Sad Journal of Mathematics* Vol. **37**. Iss. 2 (2007), 73-92.
- [3] Wildberger N. J.: A new look at multiset. School of Mathematics, UNSW Sydney 2052, Australia, 2003.
- [4] Gh. Păun, G. Rozenberg: An introduction to and an overview of membrane computing. The Oxford Handbook of Membrane Computing. Edited by Gheorghe Paun, Grzegorz Rozenberg, and Arto Salomaa. Oxford Handbooks in Mathematics The Oxford Handbook of Membrane Computing, Oxford University Press (2010).
- [5] T. Mihálydeák, Z. E. Csajbók: Membranes with local environments. In: Csuhaj-Varjú, E., Gheorghe, M., Vaszil, G. (eds.) 13th International Conference on Membrane Computing, CMC13, Budapest, Hungary, August 28 - 31, 2012. Proceedings. pp. 311–322. MTA SZTAKI, the Computer and Automation Research Institute of the Hungarian Academy of Sciences, Budapest, Hungary (2012).
- [6] T. Mihálydeák, Z. E. Csajbók: Membranes with boundaries. In: Csuhaj-Varjú, E., Gheorghe, M., Rozenberg, G., Salomaa, A., Vaszil, Gy. (eds.) CMC 2012. LNCS, vol. 7762, pp. 277–294. Springer, Heidelberg (2013).
- [7] Z. Pawlak: Rough sets. *International Journal of Parallel Programming* **11** (1982): 341–356.
- [8] D. Díaz-Pernil, I. Pérez-Hurtado, M. J. Pérez-Jiménez, A. Riscos-Núñez: A P-Lingua Programming Environment for Membrane Computing. In: Membrane Computing, Proceeding of the 9th International Workshop, WMC 2008, Edinburgh, UK, July 28-31, 2008, Revised Selected and Invited Papers. p. 187-203.
- [9] D. Meyer, K. Hornik: Generalized and Customizable Sets in R, *Journal of Statistical Software*, Vol. 31 (2009), 1–27.
- [10] L. Septem Riza, A. Janusz, C. Cornelis, F. Herrera, D. Slezak, and J. M. Benitez: Data Analysis Using Rough Set and Fuzzy Rough Set Theories. Reference manual of the RoughSets package. Web source: <http://cran.r-project.org/web/packages/RoughSets/>
- [11] Z. E. Csajbók, T. Mihálydeák: Maximal Parallelism in Membrane Systems with Generated Membrane Boundaries. In: Proceedings of 10th Conference on Computability in Europe, CiE 2014, Budapest, Hungary, June 23-27, 2014. p. 103-112.

- [12] T. Mihálydeák, Z. E. Csajbók, P. Takács: Communication Rules Working in Generated Membrane Boundaries, In: A. Alhazov, S. Cojocaru, M. Gheorghe, Y. Rogozhin (eds.) Proceedings 14th International Conference on Membrane Computing, CMC14, Chisinau, Moldova, August 20-23, 2013, 241–254.

Mellékletek

A melléklet a törzsszövegben bemutatott R függvények kódját tartalmazza. A közölt kódsorok egyféle implementációját rögzítik az elméleti ismereteknek. A véglegesítéshez további futtatási tesztek, hibakeresés és javítás, valamint kódoptimalizálás (végrehajtási idő, tárhely; esetleg objektum alapú átdolgozás) szükséges. Ezen lépések elengedhetetlenek nagyobb számítási szükségletű feladatok optimális futási megoldásaihoz.

```
#####
# INITIALIZATION PARTS
#####

#####
# EXAMPLE 1. - Source: [11]
# Z. E. Csajbók and T. Mihálydeák: Maximal Parallelism in Membrane Systems with
# Generated Membrane Boundaries. In: Proceedings of 10th Conference
# on Computability in Europe, CiE 2014, Budapest,
# Hungary, June 23-27, 2014. p. 103-112.
#####

# universum
U <- c('a','b','c','d','e')

# number of bases
n <- 4

# parts of base
B1 <- c(2,0,0,0,0)
B2 <- c(1,1,0,0,0)
B3 <- c(0,1,0,0,0)
B4 <- c(0,0,1,1,1)

# assembling base components
Bazis <- c(B1,B2,B3,B4)
B <- matrix(Bazis, nrow=n, ncol=length(U), byrow=T)

# number of regions
m <-5

# regions
W1 <- c(0,0,1,0,5)
W2 <- c(3,2,0,1,0)
W3 <- c(1,2,3,4,0)
W4 <- c(3,1,0,0,0)
W5 <- c(1,1,2,3,0)                                # this region is in the example
```



```

# assembling regions
Regio <- c(W1,W2,W3,W4,W5)
R <- matrix(Regio, nrow=m, ncol=length(U), byrow=T)

# membrane structure
MU <- c(0,1,1,1,4)

# number of rules
r <- 5

# rules - here one region, one rule
R1 <- c(0,0,1,0,0)
R2 <- c(0,0,1,0,1)
R3 <- c(0,0,0,0,3)
R4 <- c(0,0,0,0,3)
R5 <- c(1,1,0,0,0)

# type of rules
RX <- c(1,2,2,2,1)                # out, in, in, in, out

# assembling rules
Rules <- c(R1,R2,R3,R4,R5)
Q <- matrix(Rules, nrow=r, ncol=length(U), byrow=T)

#####
# EXAMPLE 2. - Source: [12]
# T. Mihálydeák, Z. E. Csajbók, P. Takács: Communication Rules Working in Generated
# Membrane Boundaries, In: A. Alhazov, S. Cojocaru, M. Gheorghe, Y. Rogozhin (eds.)
# Proceedings 14th International Conference on Membrane Computing, CMC14, Chisinău,
# Moldova, August 20-23, 2013, 241–254.
#####

# universum
U <- c('a','b','c','d','e','f')

# number of bases
n <- 10

# parts of base
B1 <- c(2,1,0,0,0,0)
B2 <- c(1,1,1,1,1,1)
B3 <- c(1,0,1,0,0,0)
B4 <- c(0,3,1,2,0,0)
B5 <- c(0,3,0,2,0,0)
B6 <- c(0,3,0,2,0,1)
B7 <- c(0,0,1,0,0,0)
B8 <- c(0,0,0,0,3,0)
B9 <- c(0,0,0,0,0,2)
B10 <- c(0,0,0,0,0,4)

# assembling base components
Bazis <- c(B1,B2,B3,B4,B5,B6,B7,B8,B9,B10)
B <- matrix(Bazis, nrow=n, ncol=length(U), byrow=T)

# number of regions

```

```

m <- 1

# regions
W1 <- c(2,11,3,9,1,0)

# assembling regions
Regio <- c(W1)
R <- matrix(Regio, nrow=m, ncol=length(U), byrow=T)

# membrane structure
MU <- c(0,1)

# number of rules
r <- 3

# rules
R1 <- c(1,0,1,0,0,0)
R2 <- c(0,6,0,6,0,0)
R3 <- c(0,0,0,3,1,0)

# type of rules
RX <- c(1,1,1) # out, out, out

# assembling rules
Rules <- c(R1,R2,R3)
Q <- matrix(Rules, nrow=r, ncol=length(U), byrow=T)

#####
# GENERAL FUNCTIONS
# Run these rows to calculate boundaries, etc.
#####

#####
# MULTISETS
#####

### mcheck – length check
mcheck <- function(mS, SU) {
  a <- 1
  if (length(mS) != length(SU))
    {a <- 0
     #message("Check the lengths!")}
  return(a)
}

### mequal – equality of multisets
mequal <- function(mS1, mS2, SU) {
  mcheck(mS1, SU); mcheck(mS2, SU)
  ncol <- length(SU)
  a <- 1
  for (i in 1:ncol) {if (mS1[i] != mS2[i]) a <- 0 }
  return(a)
}

### mpartof – inclusion relation

```

```
mpartof <- function(mS1,mS2,SU){
  mcheck(mS1,SU); mcheck(mS2,SU)
  ncol <- length(SU)
  a <- 1
  for (i in 1:ncol) { if (mS1[i] > mS2[i]) a <- 0 }
  return(a)
}
```

mintersec – intersection of multisets

```
mintersec <- function(mS1,mS2,SU){
  mcheck(mS1,SU); mcheck(mS2,SU)
  ncol <- length(SU)
  A <- 0
  for (i in 1:ncol) { A[i] <- 0 }
  for (i in 1:ncol) { A[i] <- min(mS1[i],mS2[i]) }
  return(A)
}
```

munion – union of multisets

```
munion <- function(mS1,mS2,SU){
  mcheck(mS1,SU); mcheck(mS2,SU)
  ncol <- length(SU)
  A <- 0
  for (i in 1:ncol) { A[i] <- 0 }
  for (i in 1:ncol) { A[i] <- max(mS1[i],mS2[i]) }
  return(A)
}
```

madd – addition of two multisets

```
madd <- function(mS1,mS2,SU){
  mcheck(mS1,SU); mcheck(mS2,SU)
  ncol <- length(SU)
  A <- 0
  for (i in 1:ncol) { A[i] <- 0 }
  for (i in 1:ncol) { A[i] <- mS1[i]+mS2[i] }
  return(A)
}
```

mdiff – differentiation of two multisets

```
mdiff <- function(mS1,mS2,SU){
  mcheck(mS1,SU); mcheck(mS2,SU)
  ncol <- length(SU)
  A <- 0
  for (i in 1:ncol) { A[i] <- 0 }
  for (i in 1:ncol) { A[i] <- max(mS1[i]-mS2[i],0) }
  return(A)
}
```

mnpartof – n-times

```
mnpartof <- function(mS1,mS2,SU){
  mcheck(mS1,SU); mcheck(mS2,SU)
  ncol <- length(SU)
  a <- 1
  b <- 0
  n <- 1
```

```

for (i in 1:ncol) {if (mS1[i]>0) b <- 1 }
repeat{
  if (b==0) { n<-1; break() }
  nmS1 <- n*mS1
  for (i in 1:ncol) { if (nmS1[i]>mS2[i]) a <- 0 }
  if (a == 0) break()
  n <- n+1 }
return (n-1)
}

```

```

#####
# MAS(U) PAWLAKIAN
#####

```

plow – lower approximation

```

plow <- function(M,BASE,SU){
  nrow <- length(BASE)/length(SU)           # number of rows
  ncol <- length(SU)                         # number of columns
  A <- 0                                     # collector A
  for (i in 1:nrow) { A[i] <- 0 }
  D <- 0                                     # collector D
  for (i in 1:ncol) { D[i] <- 0 }
  C <- BASE                                  # collector C
  for (i in 1:nrow) {                         # how many times Y2 part of X2
    A[i] <- mpartof(BASE[i,],M,SU)           # collect A
  }
  for (i in 1:nrow) { C[i,] <- A[i]*C[i,] }   # compute n-fold parts, collect C
  for (i in 1:nrow) {                         # comparison and collect max tags
    for (j in 1:ncol)
      if (C[i,j] > D[j]) D[j] <- C[i,j]
  }
  return(D)
}

```

pbound - boundary

```

pbound <- function(M,BASE,SU){
  nrow <- length(BASE)/length(SU)           # number of rows
  ncol <- length(SU)                         # number of columns
  A <- 0                                     # collector A
  R <- 0                                     # collector R
  S <- BASE                                  # collector S
  Sx <- 0                                    # collector Sx
  C <- BASE                                  # collector C
  D <- 0                                     # collector D
  for (i in 1:nrow) {                         # inits
    R[i] <- 0
    Sx[i] <- 0
    A[i] <- 0
    C[i,] <- 0
  }
  for (i in 1:ncol) { D[i] <- 0 }             # init
  for (i in 1:nrow) R[i] <- mpartof(BASE[i,],M,SU) # compute R, Y5 partof? X5
  for (i in 1:nrow) S[i,] <- mintersec(BASE[i,],M,SU) # compute S, Y5 intersec X5
  for (i in 1:nrow) {                         # compute Sx, S equal? zero
    for (j in 1:ncol) {
      if (S[i,j]>0) Sx[i]<-1
    }
  }
}

```

```

    }
  }
  for (i in 1:nrow) A[i] <- mnpartof(S[i,],M,SU)      # ? times S part of X5 coll.A
  for (i in 1:nrow) {                               # comp. n-fold parts, collect C
    if ((R[i] == 0) & (Sx[i] == 1))
      C[i,] <- A[i]*BASE[i,]
    }
  for (i in 1:nrow) {                               # compar. and collect max tags
    for (j in 1:ncol)
      if (C[i,j] > D[j]) D[j] <- C[i,j]
    }
  return(D)
}

```

pupp – upper approximation

```

pupp <- function(M,BASE,SU){
  ncol <- length(SU)                               # number of columns
  A <- 0                                             # collector A
  L <- 0                                             # collector L
  B <- 0                                             # collector B
  # calculate upper approximation
  L <- plow(M,BASE,SU)
  B <- pbound(M,BASE,SU)
  A <- munion(L,B,SU)
  return(A)
}

```

```

#####
# MEMBRANE BOUNDARIES
#####

```

NB -

```

NB <- function(REGION, i, BASE, j, SU, SMU) {
  ncol <- length(SU)
  # N(B,i)=0
  s1 <- 0
  s2 <- mintersec(BASE[j,],REGION[i,],SU)
  for (s3 in 1:ncol) { if (s2[s3] != 0) s1 <- 1 }
  s4 <- mpartof(BASE[j,],REGION[i,],SU)
  if ((s4==1) | (s1==0)) return(0)
  # N(B,i)=n
  s5 <- mnpartof( mintersec(BASE[j,],REGION[1,], SU ) ,REGION[1,], SU )
  if (i == 1) return(s5)
  # N(B,i)=min(k,n)
  s6 <- mnpartof( mintersec(BASE[j,],REGION[i,], SU ) ,REGION[i,], SU )
  s7 <- mnpartof( mdiff(BASE[j,],REGION[i,], SU), REGION[SMU[i] ,],SU )
  s8 <- min(s6,s7)
  return(s8)
}

```

NBi

```

NBi <- function(i){
  s1 <- length(B)/length(U)
  A <- 0
  for (s3 in 1:s1) {A[s3] <- 0}
  for (s2 in 1:s1) {A[s2] <- (NB(R,i,B,s2,U,MU))}
}

```

```
return(A)
}
```

bnd - boundary

```
bnd <- function(REGION,i,BASE,SU){
  bcol <- length(SU)
  brow <- length(B)/length(SU)
  A <- 0
  for (s1 in 1: bcol) {A[s1] <-0 }
  for (s1 in 1: brow) {
    A <- munion(A, (BASE[s1,]*NBi(i)[s1]),U)
  }
  return(A)
}
```

bndout – boundary

```
bndout <- function(REGION,i,BASE,SU){
  return(mdiff(bnd(REGION,i,BASE,SU),REGION[i,],SU))
}
```

bndin – boundary

```
bndin <- function(REGION,i,BASE,SU){
  return(mdiff(bnd(REGION,i,BASE,SU),bndout(REGION,i,BASE,SU),SU))
}
```

```
#####
```

MEMBRANE RULES

```
#####
```

Appl – applicable rules

```
Appl <- function(nr,SQ,M,SU){
  ncol <- length(SU) # length universum
  # how many times containment
  A <- 0 # A collector
  for (i in 1 : nr) { A[i] <- 0 } # initialization
  for (i in 1 : nr) { A[i] <- mnpartof(SQ[i,],M,SU) } # collect # print (A)
  # create matrix of applicable rules
  a <- 1 # number of rows
  for (i in 1 : nr) { a <- a * (A[i]+1) } # print(a)
  B <- array(0, dim = c(a,nr)) # create zero matrix
  j <- nr # number of rules
  k <- 1 # running variable
  for (i in 1 : r) {
    B[,j] <- rep( rep ( 0: A[j] , eac = k) , length.out=a)
    k <- k * (A[j]+1)
    j <- j-1
  } # print(B) - test
  # check - rules part, create multisets from rules
  C <- array(0, dim = c(a, ncol))
  for (i in 1: a) {
    D <- 0 # D collector
    for (k in 1 : ncol) { D[k] <- 0 } # initialization
    for (j in 1 : nr) {
      D <- madd(D,B[i,j]*SQ[j,],SU)
    }
    C[i,] <- D
  }
}
```

```

    }
    #print(C) # calculated rules - test
    #print(B) # possible rules - test
    # clear C (what if C[i,] does not part of M) - test
    E <- array(0, dim = c(1, nr)) # E collector
    E[1,] <- B[2,]
    # print(E)
    for (i in 3 : a) {
      if (mpartof(C[i,],M,SU)==1) {
        E <- rbind(E, B[i,])
      }
    }
    return(E)
  }
}

### ApplMax – maximal parallel applicable rules
ApplMax <- function(nr,SQ,M,SU){
  B <- Appl(nr,SQ,M,SU) # possible rules
  a <- nrow(B) # number of rows
  #print(B) # test
  #print(a) # test
  E <- 0 # E collector
  E <- array(0, dim = c(1, nr)) # E initialization
  for (i in 1 : a) { # double-cycle 1.
    c <- 0 # semafor
    for (j in 1 : a) { # double-cycle 2.
      if (i==j) next # step over the same index
      if (gmpartof(B[i,],B[j,],nr) == 1) c <- 1 # change semafor
    }
    if (c == 0) {E <- rbind(E, B[i,])} # if semafor 0, attach to E
  }
  F <- E[-1,] # delete first row
  print(F) # print
}

```

CoSTiP Szeminárium - ISTQB Syllabus a gyakorlatban

CoSTiP Seminar – ISTQB Syllabus in practise

Boda Béla

Neuron Szoftver Kft.

boda.bela@neuron.hu

bodabela@gmail.com

Absztrakt: Bemutatom, hogy a Debreceni Egyetem Informatikai Karán 2014-ben indított CoSTiP (Kompetens szoftvertesztelés a gyakorlatban) című tárgy "Tesztelés a szoftver életciklusán át" című témaköréhez tartozó szemináriumi gyakorlatokat milyen tartalommal töltöttük meg, és milyen feladatok és eszközök segítették a hallgatókat a téma feldolgozásában. A gyakorlati órákon rámutattunk a folyamatos, kreatív munkavégzésre, a feladatok becslésének nehézségeire valamint e nehézségek kezelésére. Megismerkedtünk a legelterjedtebb egységteszt keretrendszer (JUnit) eszközrendszerével néhány egyszerű példán keresztül. Végül a stub és mock technikákat vizsgáltuk. Valamint betekintést nyertek a diákok a ciklomatikus komplexitás alkalmazásába és a lefedettség vizsgálatára. Az órák menetén túl szeretném elmondani a fogadtatást és a hallgatók véleményét is.

Kulcsszavak: istqb, syllabus, gyakorlat, szeminárium, egységteszt, integrációs teszt, junit, lefedettség, szoftverfejlesztési modell, becslés, vezető, beosztott, tesztszintek, tesztípusok

Abstract: I would like to show how I demonstrated the chapter „Testing Throughout the Software Life Cycle” of course CoSTiP at University of Debrecen Faculty of Informatics in 2014. What were the topics, the tasks and tools that helped the students to proceed the lecture. I show methods of creative working, the pain of estimation and how to reduce this pain. The students get familiar with JUnit framework through some examples. We also have some stubbing and mocking tasks. We get a short insight of cyclomatic complexity and coverage. Besides the topics of the course I would like to cover how the students welcome each exercise.

Keywords: istqb, syllabus, seminar, unit test, integration test, junit, coverage, software development model, estimation, leader, employee, test levels, test types

1. Bevezetés - feladatom a CoSTiP kurzusban

A CoSTiP tantárgy az ISTQB Syllabus „Foundation Level” anyagát hivatott átültetni egyetemi keretek közé kiegészítve a piaci résztvevők tapasztalatával és azok gyakorlati megközelítésével. Az anyagot több vállalattól érkező vendégelőadó osztotta fel egymás között ki-ki a munkaköréhez legközelebb álló témakört dolgozva ki.

Az én feladatom a második fejezet –„Tesztelés a szoftver életciklusán át”– kidolgozása volt, mely a „Szoftverfejlesztési modellek”, „Tesztszintek” és „Tesztípusok” témaköröket foglalja magába. E rész három előadást és három szemináriumi alkalmat jelent egy szemeszterben. A következőkben a három szeminárium anyagát ismertetem.

2. Szoftverfejlesztési modellek szeminárium

Komoly kihívásnak tűnt, hogy ezt a témakört egyetlen szemináriumi órába sűrítve gyakoroljuk a diákokkal, ezért igyekeztem néhány elvárást megfogalmazni az órával kapcsolatban.

- Keltse fel a hallgatók érdeklődését
- Valós élethelyzetekben felmerülő szituációkat mutassunk be
- Módszertani alkalmazás is jelenjen meg

Egy a teszteléssel kapcsolatban is megjelenő feladattal kezdünk, mely a becslések nehézségeire próbál rámutatni, tovább arra, hogy hogyan könnyíthetjük a helyzetünket akkor, amikor becslést kell végeznünk.

Az alábbi állatokat kell kifejezniük a diákoknak lemmingben az átlagos tömegük alapján:

- Macska
- Tacsó
- Gerenuk
- Bordeaux-i dog
- Ember
- Hippopotamus
- Afrikai elefánt
- Kék bálna

Azaz a kérdés az, hogy például egy afrikai elefánt tömege körülbelül hány lemming tömegével egyezik meg, ha az átlagos tömegüket vizsgáljuk.

A feladat célja, hogy megmutassuk a diákoknak, hogy milyen, amikor a komfort zónában, azon kívül, illetve nagy különbségek esetén kell becslést végezni. Az állatokat e feladatban nem rendezzük sorba, hanem szétszórva helyezzük el, ezzel is nehezítve a feladatot. A macska, a tacsó fogalmak, és jól látható, hogy körülbelül hány lemming tömegét teszik ki körülbelül. A gerenuk teljesen ismeretlen mindenki számára, de ezt is becsülniük kell. A bordeaux-i dog és az ember szintén ismerős, és egymással jól összehasonlíthatók, viszont lemmingben kifejezni őket már kicsit nehezebb. Az utolsó három állat esetén nagy különbségek vannak egymáshoz képest is, és sokkal nagyobbak, minthogy kényelmesen meg lehessen becsülni őket.

Ezzel a feladattal a dekompozíció és a részfeladatokra bontás jelentőségére hívjuk fel a figyelmet: a komfortzónánkba eső méretűre kell bontani a feladatokat ahhoz, hogy jól becsüljünk.

E feladat megoldásakor a diákok az elvárásoknak megfelelően becsültek. Természetesen voltak, akik nagyon messze voltak az igazságtól, és olyanok is, akik pontosan eltalálták vagy nagyságrendileg jól tippeltek.

A következő feladatban sorba rendezzük az állatokat és azokat egymásban kell kifejezni hasonlóan a tömegük alapján. Azaz egy macska hány lemming, egy tacsó hány macska és így tovább. E feladat célja, hogy demonstrálja a tapasztalat alapján történő relatív becslést, és azt, hogy a becsült értékeket hogyan használhatjuk arra, hogy a nagyobb különbségeket is becsülni tudjuk. Például, ha egy kék bálna 22 elefánt és egy elefánt 3 víziló, akkor a kékbálna körülbelül 66 víziló lehet.

Az óra utolsó feladata egy toronyépítési feladat, melyet párokban hajtunk végre. A páros egyik tagja a vezető a másik a beosztott. A beosztott szemét bekötjük. A vezető feladata, hogy utasításokat adjon a beosztottnak, és egy perc alatt a lehető legmagasabb tornyot építsék egyszerű kockákból. Ha a torony ledől, újra lehet kezdeni. A mért egy perc előtt lehetőségük van fél perc gyakorlásra. A párosok sorban követik egymást.

A jelenség az, hogy a „balra”, „jobbra”, „fel”, „le” utasítások hangzanak el. A következő páros általában tanul az előzőek hibáiból, és próbálnak újabb módszereket kitalálni, de alapjaiban nem változik a tornyok építése, és hasonló magasságig jutnak. Esetünkben 8 és 11 közötti elemszámot sikerült egymásra rakni.

Miután minden páros megépítette a tornyát, én veszem át a vezető szerepet minden párosban, és én adok utasításokat. Hangsúlyozom, hogy nem baj, ha összedől a torony, kezdje újra nyugodtan. Azt mondom a beosztottnak, hogy építse, ahogy jónak látja, akár két kézzel vagy többet is megfoghat egyszerre, de én kérem majd, hogy mikor álljon meg. Amíg épít, folyamatosan megerősítem: „rendben”, „ez az”, „jó”. Ha ledől: „nem baj, még van idő”, „rakd újra”. Sőt az is előfordul, hogy arra utasítom, hogy döntse le, mert nem jól áll, és kezdje előlről minél hamarabb. Ez esetben minden beosztottal 2-5 kockával többet sikerült felépítenünk, és ami sokkal fontosabb, hogy úgy érezték, kevésbé stresszes az építés, bátrabbak voltak, mertek kísérletezni.

E feladat célja, hogy megmutassa, mennyivel hatékonyabb és kellemesebb az intuícióna bízott munkavégzés, és mennyivel jobb, ha a feladat végzője határozza meg a munkavégzés módját, ezáltal szerevezve tapasztalatot akár kudarcok árán is.

3. Tesztszintek és tesztípusok szemináriumok

A tesztszintek szemináriumon a két alsóbb szint gyakorlati alkalmazásával foglalkozunk: egység és integrációs tesztek. A tesztípusok gyakorlat pedig egy lefedettség vizsgálatból áll, melynek segítségével vizsgáljuk azt is, hogy hány tesztetesre lenne szükség a teljes lefedettséghez (ciklomatikus komplexitás számítás). A témakör első része jóval nagyobb, mint a második, így a tesztszintek szeminárium anyaga átnyúlik a tesztípusok szemináriumába.

A JUnit eszköztárszerének alapjait tekintjük át:

- Tesztosztály és tesztmetódus - @Test annotáció
- Assertek, azaz az elvárt eredmény ellenőrzése
- Teszt életciklus annotációk - @Before, @After, @BeforeClass, @AfterClass
- Elvárt kivétel vizsgálata – @Test(expected=...)
- Időtúllépés vizsgálata – @Test(timeout=...)
- Runner osztályok - @RunWith
- Tesztgyűjtemények - @RunWith(Suite.class), @SuiteClasses
- Paramétervizsgálat - @RunWith(Parametrized.class), @Parametrized.Parameters

Hangsúlyozzuk azt, hogy a JUnit teszt metódusokkal szemben alapvető elvárás, hogy egymástól függetlenül fussanak, így a globális hatáskört csak olvasásra szabad használni, statikus tagok állapotát nem változtatjuk! Kitérünk arra, hogy a teszt futtatók mindig új példányt hoznak létre az adott tesztosztályból, amikor egy-egy tesztmetódust futtatnak. Ezáltal az osztályszintű tagok tekinthetők a tesztmetódus saját hatáskörének, azaz minden példány szintű tagot inicializálni kell a teszt metódus futása előtt.

Az alapok megismerés után folytatjuk a Stubbing és Mocking technikákkal. Megnézünk néhány példát mindkét technikával, és rámutatunk a legfőbb különbségekre a megvalósítás tekintetében: stub implementációkkal és mock objektumokkal történő megoldás.

A Mockito nevű keretrendszerrel dolgozunk. Célunk, hogy e keretrendszerrel demonstráljuk a mocking technika lehetőségeit. Kitérünk a mockítóval rögzíthető elvárásokra (when) és verifikációkra, melyek új szemlélet adnak a teszt objektumok megvalósításának. Vizsgáljuk a mockobjektum metódusainak visszatérési lehetőségeit, a kivételek kiváltását és az egyedi válasz alkalmazását. Ez utóbbit arra használjuk, hogy az adott egység belső működésbe betekinthessünk és értékeket ellenőrizzünk futás közben. Továbbá kitérünk a hívások verifikálására, melyet a viselkedés ellenőrzésére használjunk, azaz gray-box technikát alkalmazunk. Azt vizsgáljuk, hogy meghívódtak-e a mockobjektum metódusai. Ha igen, hányszor, milyen paraméterekkel. Ezeket a Mockito verify műveleteivel hajtjuk végre. A Mockito segítségével azt is vizsgáljuk, hogy a tesztelendő metódus hívása előtt érintetlen volt-e a mockobjektum (verifyZeroInteractions), illetve a hívás és a viselkedés ellenőrzése után maradt-e még ellenőrizetlen hívás a mockobjektumon (verifyNoMoreInteractions).

A lefedettség vizsgálatot SonarCube illetve Jacoco segítségével vizsgáljuk. Kitérünk az utasítás és az elágazás alapú lefedettség közötti különbségre, és megtekintjük azt egy Jacoco jelentésen belül kiválasztott metóduson. Kiszámítjuk a ciklomatikus komplexitást, és belátjuk, hogy mennyi az a minimális teszteset mennyiség, amennyivel teljesen le lehet fedni a vizsgált metódust, ha az elágazásokat vesszük alapul.

A fennmaradó időben néhány nem-funkcionális teszt típusra nézünk gyakorlati példát, illetve sorra vesszük azok jelentőségét a különböző típusú rendszereknél.

4. Köszönetnyilvánítás

Köszönet a Debreceni Egyetem Informatikai Kar dékánjának, Dr. Mihálydeák Tamásnak, Juhász István Tanár Úrnak és Bodrogyózi Lászlónak, hogy részt vehetek a CoSTiP tantárgy oktatásában.

Irodalomjegyzék

[1] © HTB – Hungarian Testing Board Magyar Szoftvertesztelői Tanács Egyesület: ISTQB® Certified Tester Foundation Level Hivatalos magyar nyelvű tanterv Alapszintű képzés (2012)

[2] junit.org

[3] <http://docs.mockito.googlecode.com/hg/org/mockito/Mockito.html>

Szoftvertechnológia - mobilfejlesztés

Softvertechnology – mobile development

Dr Illés Zoltán^a, Heizlerné B. Viktória^b

^a ELTE Informatika Kar
illes@inf.elte.hu

^b ELTE Informatika Kar
hbv@inf.elte.hu

Absztrakt: A BSC-MSC-s hallgatók jól ismerik a klasszikus, bevált szoftvertechnológiai, programfejlesztési elveket, környezeteket, miközben majd mindegyikük kezében ott van már egy-egy okos eszköz (telefon). Ezek szintén programozhatóak, a programozásuk mikéntje azonban talán nem teljesen magától értetődő. A korábban elsajátított programozási paradigmák természetesen ugyanúgy felhasználhatóak az újabb platformokon is, így a most elterjedő mobil eszközök területén is. Annak ellenére, hogy a tervezési alapelvek azonosak, mégis szükség van speciális ismeretekre is ahhoz (új érzékelők, erőforrások kezelése, UI), hogy az egyes platformokra hatékony, minőségi alkalmazások készíthessenek. Mind az ipar, mind pedig a hallgatói igények arra sarkallják a felsőoktatási intézményeket itthon és külföldön is, hogy valamilyen módon emeljék be tematikájukba a mobilfejlesztési területet, akár speciál kollégiumok, akár blokkok, szakirányok vagy akár teljes képzések vonatkozásában. Az ELTE IK-n eddig is rendelkezésre álltak különböző speciál kollégiumok, de 2014 tavaszán elindult egy négy féléves választható blokk is, a szoftvertechnológia laboratórium keretén belül, amely elvégzése során az MSC képzésében résztvevő informatikushallgatók jobban elmélyedhetnek ezen a területen. A blokk anyagának összeállításánál fontosnak tartottuk, hogy az egyes platformok és fejlesztési lehetőségeik naprakész ismerete mellett bevezessük a hallgatókat a multiplatform megoldások világába is. A feladat nehézségét mutatja a ma elterjedt különböző mobilplatformok, eszközök különbözősége az Androidtól a Windows Phone-ig, de gondolhatunk akár egyetlen platformon belüli gyors változásra is, például a Silverlight „eltűnésére”.¹

Kulcsszavak: szoftvertechnológia, mobilfejlesztés, oktatás

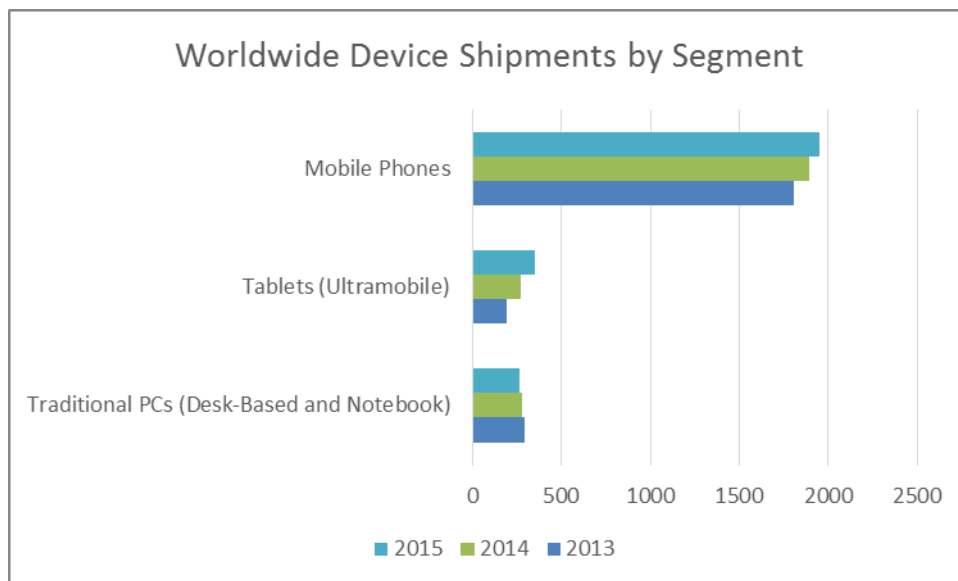
Abstract: Informatics students well know the classical and long standing theories of software technology, program development and programming environments, while most of them use daily regularity smart devices (smart phones). These devices are also programable, but the „know how” of it is not going without saying. Programming paradigmas, learnt before are naturally can be used up on new platforms as well, on mobile platforms too. Though the planning methods, theories are the same developers most have special knowledge (new sensors, new resources) to be able to develop efficient, qualitative applications. Both of the needs of industry and the students urge the institutes of higher education to built in mobile development into their syllabus, either as special courses, blocks, specializations or as unique trainings. In ELTE IK there were different special courses dealing with this area before too, but during the spring semester of 2014 a choosable four semester long block was started as a new offer in softwaretechnological laboratory. During the block MSC students can deepen their knowledge in this special field. Creating the content of the block we kept very important to drive students into the world of multiplatform solutions too along with the deep, daily knowledge of developing on unique platforms. The task is not easy at all, while well-spread platforms and devices from Android to WP differ to each other quite much, moreover there are big changes on each platform too..

Keywords: softwaretechnology, mobile development, education

¹ A TÁMOP-4.1.2.B2-13/1 „Pedagógusképzést segítő szolgáltató és kutatóhálózatok továbbfejlesztése és kiszélesítése, 2. Országos módszertani és képzésfejlesztési komponens” pályázat támogatásával.

1. Bevezetés

Napjainkban az okos eszközök rohamos térnyerését tapasztaljuk, kezdve az okos telefonoktól a hordható okos eszközökig (lásd 1. ábra).



1. **ábra** Eszköz-eladások <http://www.gartner.com/newsroom/id/2692318>

Miért is ilyen népszerűek a mobil eszközök? Ezek az eszközök új dimenziókat nyitnak a mindennapok embere számára a gyors információkeresés, a szórakozás, a tanulás [1] és a kommunikáció területén. Szemben egy asztali PC-vel, az okos telefonok bárhol és bármikor használhatóak, könnyűek, hordozhatóak, mára az árak is elérhető a többség számára. Ráadásul teljesítményük is összemérhető az asztali gépekével például nézzük a jó minőségű Sony Xperia Z2 telefon jellemzőit: 4 magos processzor, 2,3 GHz, 3 GB RAM, 5,2” képernyő, 20MP kamera stb. Talán a képernyő méretét lehet az egyetlen „szűk keresztmetszetnek” tekinteni. Az okos telefonok ezen felül rendelkeznek olyan érzékelőkkel is, amelyek az asztali gépekre nem jellemzők, mint például az érintőképernyő, a GPS vagy a gyorsulásmérő, amelyek nagyon vonzóak a felhasználók számára. Ide tartozónak érezzük Barack Obama szavait, aki a következőket mondta: Don’t just play on your phone. Program it. [6]

Természetesen az ezeken az eszközökön futatható alkalmazások elkészítéséhez programozó szakemberekre is szükség van. Nyilvánvaló, hogy az ilyen gyors és radikális változások egy adott területen belül komoly kihívások elé állítják mind a szakma művelőit, mind pedig az adott területtel foglalkozó oktatókat, a képzési programok összeállítóit. [3] Szerencsére azonban egy-egy új platform megjelenése (az asztali gépek, a webes alkalmazások után a mobil platform) nem jelenti azt, hogy a jól bevált programtervezési módszereket, technikákat nem használhatjuk tovább - éppen ellenkezőleg, ezek az elvek biztos alapot adnak a szoftverfejlesztés bármely területén. Ennek ellenére szükség van arra, hogy az újabb és újabb platformok speciális tudnivalóit a hallgatók megismerjék már a képzésük során is.

A mobil eszközök programozása megjelenik az oktatás különböző szintjein a szakképzéstől a felsőoktatásig. Az új szakképzések tekintetében említhetjük a Mobilalkalmazás fejlesztő 55 213 04 számú OKJ képzést vagy akár a 2012-ben induló programtervező informatikus asszisztens felsőoktatási képzést, ahol már hangsúlyosan megjelenik a mobilfejlesztési terület. A magyar felsőoktatásban a különböző BSC illetve MSC informatikai képzésekben is majd mindenhol megjelenik speciál kollégiumok (pl. BME Android alapú szoftverfejlesztés),

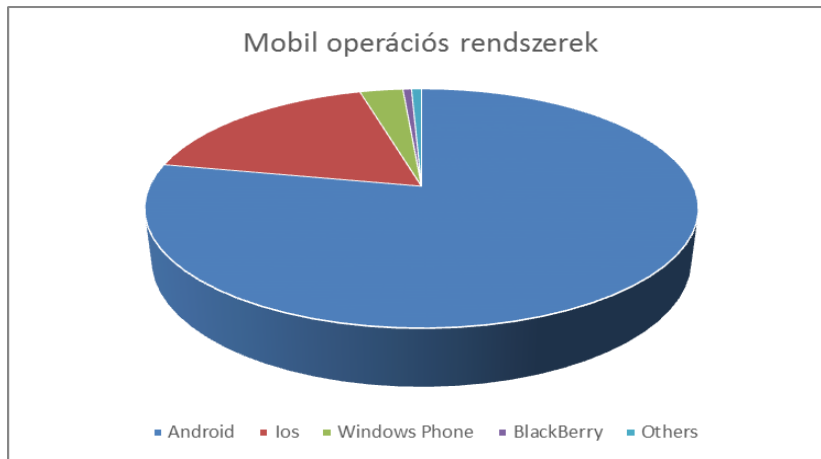
blokkok (ELTE IK Mobil Akadémia) vagy szakirányok (Óbudai Egyetem Beágyazott és mobil információs rendszerek szakirány) formájában. Külföldön, például Angliában már önálló szakként is jelen van ez a terület.

http://www.uk-universities.net/Universities/Programs/Mobile_Computing.html.

2. Mobilprogramozás specialitásai

Miben is jelent mást a mobil telefonokra készült alkalmazásfejlesztés a hagyományossal szemben? Állítjuk, hogy nincs szükség paradigmaváltásra, a jól bevált programfejlesztési módszerek alkalmazhatóak ezen a területen is, azonban a különböző érzékelők programozása, kezelése mindenképpen újdonságot jelent a fejlesztők számára. Új szempontot jelent az, hogy a hordozhatóság miatt az alkalmazások erőforrással való takarékoskodása különösen fontos.

A ma elterjedt mobil telefonok különböző hardverparaméterekkel (képernyőméret, érzékelők) rendelkeznek, rajtuk különböző operációs rendszerek futnak, lásd 2. ábra. Ezek után már nem meglepő, hogy az egyes készülékek esetében a fejlesztőeszközök és a használható programozási nyelvek sem egységesek.



2. ábra A legelterjedtebb okos telefon operációs rendszerek
(Forrás: IDC Worldwide Mobile Phone Tracker, February 12, 2014)

Nyilvánvaló, hogy a fejlesztők célja az lenne, hogy egy-egy elkészült alkalmazás mindenki számára elérhető legyen, de ez a fentiek miatt egyáltalán nem magától értetődő feladat, hiszen ugyanazt az alkalmazást el kellene készíteni minden egyes platformra. Van-e, lehet-e találni elegánsabb megoldást erre a problémára?

Természetesen több különböző lehetőség is adódik. Az egyik ezek közül az, hogy böngészőben futó alkalmazásokat kell készíteni, hiszen ezek bármelyik mobilkészüléken, sőt asztali PC-n is ugyanúgy fognak viselkedni. (Gondoskodni kell ilyenkor a különböző méretekből adódó helyes megjelenítésekről a CSS-en keresztül.) Sajnos ennek a megoldásnak hátránya is van, mégpedig az, hogy a böngészőből az eszközbe beépített érzékelőket nem érhetjük el – ezzel leszűkülnek az elkészíthető alkalmazások típusai. Erre kínálnak megoldás a különböző keretrendszerek (pl. PhoneGap), amelyek megfelelő API-t szolgáltatnak ezekhez a feladatoknak a végrehajtásához is.

A másik megoldás az, hogy olyan fejlesztőeszközt választunk például a Xamarin Studio-t vagy a Visual Studio-t, amely több platformra is tartalmaz alkalmazásablont. Nem szabad

azonban elfelejteni, hogy ezek az eszközök sem mindentudóak. Az alkalmazáslogikát ugyan elegendő egyszer megírni, de például a felhasználói interfészt platformtól függően azért el kell készíteni. A mobiltelefonok világa talán a többi informatikai területnél is gyorsabban változik, az eszközök egy-két év alatt elavulnak, az egyes platformokhoz kapcsolódó szoftveres ismeretek is folyamatosan megújulnak - elegendő csak a Windows Phone platformról hirtelen eltűnő Silverlight technológiára és az e helyett megcélzott DirectX-es megoldásra gondolni. Így az ezen a területen dolgozó fejlesztők és oktatóik állandó megújulásra kényszerülnek.

3. Internet of Things, Eszközök hálózata

A számítógépek, számítóközpontok fejlődése, általánossá válása azt eredményezte, hogy a felhasználóknak egyre személyesebb „élménnyé” vált a hozzá kapcsolódó munka. A hétköznapiakban ez abban jelentkezett, hogy a személyi számítógépek jelentősége megnőtt, a számítógépes munkák tömegessé váltak, a szoftveres megvalósítások sok esetben „varázslat”, „művészet” kategóriába sorolhatók. [7]

A számítógépek jótékony hatását, segítségét a korai informatikai világ örökségeként elsősorban a különböző tudományágak használták. Matematikai modellek készítése, fizikai, kémiai stb. szimulációk elvégzéséhez magától értetődik, hogy a számítógépek használata nélkülözhetetlen. A külvilággal való kapcsolat, annak érzékelése, befolyásolása egy következő lépés volt. Ahogy a számítógépek fejlődésére nagy hatással volt az elektronikai eszközök fejlődése, miniatürizálódása, úgy igaz ez minden eszközre.

A mobiltelefonok megjelenésének idején kevesen gondolták, hogy ezek ma olyan mini számítógépekké válnak, amelyeknek már csak elenyésző feladata a telefonálás.

Hasonlóan elmondható, hogy a környezetünk eszközein is érzékelhető az elmúlt évek fejlődése, a klasszikus logikai áramköröket, vezérléseket SoC alapú mikrovezérlők, mikroszámítógépek helyettesítik. Sok általános lego szerű vezérlő építőcsalád is elérhető, mint például az Arduino, BeagleBone, Raspberry Pi, Udoo stb.

Általánosan elmondható, hogy a mobil világ korábban talán csak a mobiltelefonokkal kapcsolatos fejlesztéseket jelentette, de mára sokkal általánosabbá vált. A körülöttünk lévő eszközök, nem csak a nagyobbak, mint autó, televízió, hanem a kisebbek is, mint például egy óra, szemüveg, hőmérő, kabát, a „lelkük mélyén” egy-egy mikroszámítógép, amiknek a viselkedését a rajta futó program határozza meg. Ma természetesen az is igaz, hogy ezen eszközök kommunikációra születtek. Mit kezdene a például a hőmérő az aktuális hőmérséklet értékkel? Ennek csak úgy van értelme, ha ezek az információk más eszközök rendelkezésére állnak. Ha tehát a hőmérő hideget érez, akkor a kabát kicsit jobban fűti a viselőjét.

Ez a programkészítés oldaláról jelentős feladatokat generál. Egyrészt biztosítani kell a mikroszámítógép alap operációs rendszerét, másrészt olyan fejlesztési lehetőségeket, melyekkel testre szabhatók ezek az egymással kapcsolatban lévő eszközök.

4. Szoftvertechnológia laboratórium – Mobil Akadémia

Az ELTE Informatika Karán a mobilfejlesztés iránt érdeklődő hallgatók több különböző speciál kollégium közül választhattak:

- C# mobil környezetben (önálló tanulásukat on-line tananyag is segíti [4])

- Gazdag funkcionalitású webes és mobilos alkalmazások Flex-szel
- Mobil alkalmazásfejlesztés

Ezek a tárgyak jellegüknél fogva nem épültek egymásra, nem adtak, nem is adhattak mélyebb és részletesebb ismereteket erről a teljes területről. Megismerve a hallgatói igényeket több szempontból is megvizsgáltuk ennek a témakörnek az oktatásba emelését. [2] Végül így született meg a 4 féléves szoftvertechnológia laboratórium mobilfejlesztéssel foglalkozó legújabb blokkjának terve. [5] Ez a 16 kredités blokk az MSC hallgatók számára van meghirdetve, mivel ők már rendelkeznek a megfelelő szoftvertechnológiai ismeretekkel, így elegendő a mobilterület specialitásaira koncentrálni.

Módszertanilag előtérbe helyeztük az iparban is használt team-munkát, projekt szemléletet, a SCRUM módszert.

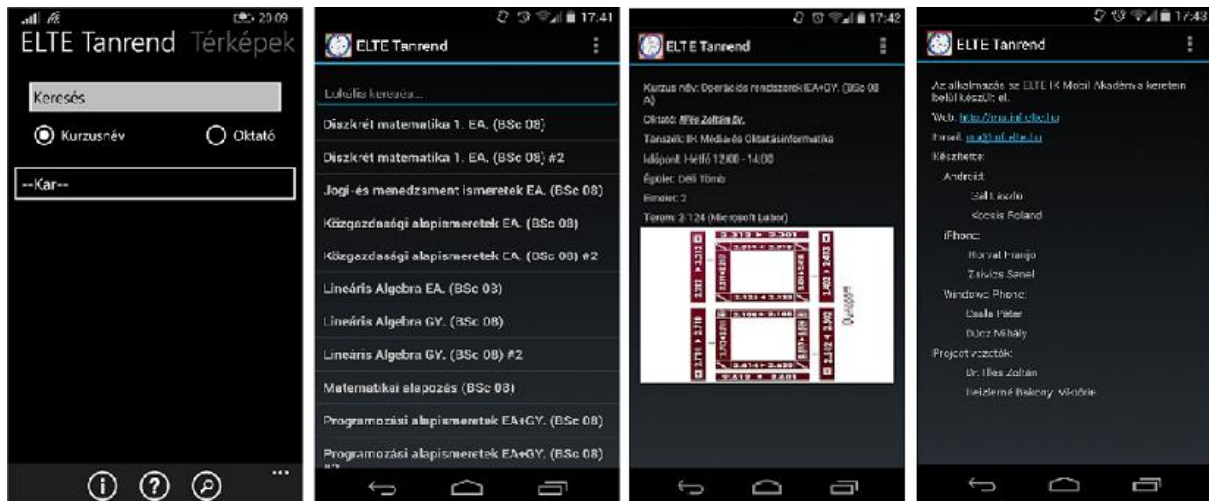
Tartalmilag fontosnak tartottuk, hogy a hallgatók megismerjék részletesen az egyes platformokat, mindeközben átlássák és kiismerjék magukat a multiplatform megoldásokban és egyúttal gyakorlatot szerezzenek összetett alkalmazások készítésében is. Nem szabad elfelejteni, hogy az okos telefonok mellett ma már más okos eszközök (okos óra, okos szemüveg) jelentősége, elterjedtsége is rohamosan nő, így programozásuknak is előbb-utóbb meg kell jelennie az informatikai képzésekben.

A fent leírtaknak megfelelően a 2014 tavaszi félévében útjára indult blokk egyes féléveinek tartalmi felépítése a következő (<http://ma.inf.elte.hu>):

- Mobil platformok - Alkalmazások készítése több platformra
- Kód újrahasonosítás multiplatform környezetben
- Összetett projektek tervezése és implementálása
- A jelen és a jövő okos eszközei - "Okos kabát"

Az ELTE IK tanrendjének eddig web-en elérhető változata rosszul olvashatónak bizonyult a mobilos környezetben. Az első félév feladata így egy tanrend alkalmazás elkészítése volt a legelterjedtebb platformokra: Windows Phone-ra, Android-ra és iPhone-ra. (A munkában résztvevő lelkes hallgatók Gál László, Kocsis Roland (Android), Horvat Franjo, Zsivics Sanel (iPhone), Csala Péter, Dúcz Mihály (Windows Phone)).

A félév végére elkészült verziók, amelyek felkerültek az AppStore-ba is, egyelőre a leggyakrabban használt webes funkciókat, az egyes oktatók és kurzusok adatainak lekérdezését támogatják, lásd 3. ábra



5. ábra ELTE Tanrend mobilalkalmazás

Az alkalmazás az adatait az ELTE TTK által üzemeltetett tanrendi adatbázisból nyeri (amiért köszönet illeti őket), így biztosítva, hogy a megszokott webes és a mobilos felületen keresztül mindig azonos eredményt kapjanak a felhasználók.

Az adatbázis lekérdezését egy C#-ban írt web-szolgáltatás végzi, amely a választ Json formában küldi a rákapcsolódó klienseknek.

A projekt eredményeit elérhetik a <http://tanrend.inf.elte.hu/index.html> oldalon.

6. Összegzés

A gyorsan elterjedt okos telefonok, okos eszközök működtetéséhez megfelelő szakembergárdára van szükség, köztük programozókra is, így ez a terület meg kell, hogy jelenjen valamilyen formában a szakképzésben középszinttől egészen a felsőoktatásig. Kijelenthetjük, hogy a fejlesztéshez nincs szükség paradigmaváltásra, de új technológiai ismeretekre igen. Mivel az eszközök hardveres és szoftveres jellemzői ma ugyancsak különbözőek, megoldást kell találni a különböző platformok mind hatékonyabb együttes kezelésére – ez az útkeresés azonban még nem zárult le, napjainkban is folyik.

Az eszközök sokszínűsége és a rendkívül gyors fejlődése megnehezíti a képző intézmények dolgát, hiszen állandóan biztosítani kell az oktatáshoz a legújabb eszközöket, technológiákat.

Irodalomjegyzék

- [1] Dr. Illés Zoltán, Dr Psenaková Ildikó, Heizlerné B. Viktória: New Perspectives of M-Learning, *Acta Electrotechnica Et Informatica* 8 (3) pp. 36-39. (2008)
- [2] Dr Psenaková Ildikó, Dr Illés Zoltán, Dr Szabó Tibor, Heizlerné B. Viktória: Mobilvilág és mobilprogramok fejlesztése WP7-re, *Tudomány az Oktatásért – Oktatás a Tudományért, Constantine the Philosopher University in Nitra, Faculty of Central European Studies*, 2014. 11 p. (2014)

- [3] Heizlerné B. Viktória, Dr Illés Zoltán, Menyhárt László: Viewpoints for the development of teaching contents in the field of informatics, *Acta Didactica Napocensia ISSN: 2065-1430* 6: (2) Paper 5. 10 p. (2013)
- [4] Dr Illés Zoltán, Dr Psenáková Ildikó, Dr Szabó Tibor, Baráth Ádám, Illés Zoltán, Kapos Ádám, Kiss Dávid, Heizlerné B. Viktória: Mobil világ és fejlesztése WP7 környezetben (ISBN:ISBN 978-963-284-465-7) (2013)
- [5] Menyhárt László, Dr Illés Zoltán, Heizlerné B. Viktória: Birth of Mobile Academy, *ICAI (International Conference of Applied Informatics) 2014*. Eger (2014.01.30-02.2)
- [6] Barack Obama: „Don't just play on your phone. Program it.” , <http://www.wired.com/wiredenterprise/2013/12/obama-code>
- [7] D.E.Knuth: A számítógép programozás művészete, Műszaki könyvkiadó, 1994

DSPNSim: Szimulációs Szoftver Determinisztikus és Sztochasztikus Petri Hálókra

DSPNSim: Simulation Software for Deterministic and Stochastic Petri Nets

Molnár András^a, Horváth Ádám^b

^aNyugat-magyarországi Egyetem, gazdaságinformatikus M.Sc. hallgató
molnaran@gain.nyme.hu

^bNyugat-magyarországi Egyetem, Informatikai és Gazdasági Intézet
horvath@inf.nyme.hu

Absztrakt: A teljesítményvizsgálatoknak számos eszköze áll rendelkezésünkre, különösen a markovi modellek esetében. Sok folyamat azonban nem tekinthető markovinak, például egy gyártási folyamatban a különböző munkafázisok eloszlása általában determinisztikus. Determinisztikus és sztochasztikus késleltetésű folyamatok jól leírhatók egy determinisztikus és sztochasztikus Petri háló (DSPN) segítségével, melyet először Ajmone Marsan és Chiola javasolt [1]. A DSPN-ek analitikus megoldása néhány speciális esetet leszámítva [2, 3] arra az esetre korlátozódik, amikor a háló minden állapotában legfeljebb egy determinisztikus átmenet engedélyezett. Analitikus megoldás hiányában szimulációs szoftvereket használhatunk a háló egyensúlyi vagy tranzien eloszlásának meghatározására.

Bár vannak a területen ismert szoftverek, ezek némelyike elavult (DSPNExpress [4]), némelyike pedig tömeges eredményeket csak nehézkesen képes előállítani (GreatSPN [5], TimeNET [6]). Így született meg a DSPNSim, egy saját fejlesztésű szoftver, amely lehetővé teszi a DSPN paraméterfájllal való leírását, a kötegelt futtatást, s egy olyan statisztikai modullal egészül ki, amely biztosítja a szimuláció pontosságát (relatív hiba, konfidenciaszint). A statisztikai modulban egyes eljárásokat pontos közelítő algoritmusok segítségével valósítottunk meg, amely a szimulációnál kritikus futási időt rövidíti le.

Kulcsszavak: determinisztikus és sztochasztikus Petri hálók, szimulációs szoftver

Abstract: Performance analysis can be carried out in several ways, especially in case of Markovian models. However, many processes cannot be considered Markovian. For example, in a manufacturing process, the distribution of the work phases are generally deterministic. A process in which the delays are either deterministic or exponential, can be described by a deterministic and stochastic Petri net (DSPN), which was first proposed by Ajmone Marsan and Chiola [1]. Except some special cases [2, 3], the analytical solution of DSPNs is restricted to the case when only one deterministic transition is enabled in each marking. Therefore, simulation software can be used to obtain the steady state or transient solution of the DSPN.

Although there are some well-known simulation software for DSPNs, some of them are obsolete (DSPNExpress [4]), while some are not appropriate for generating results in batch mode (GreatSPN [5], TimeNET [6]). These facts were the main motivation factors of developing a new simulation software, which supports the parametric description of DSPN and the batch processing. Moreover, DSPNSim is complemented with a statistical module, which ensures the precision of the simulation (relative error, confidence level). In the statistical module, certain procedures were implemented using accurate approximate methods. So, the generally critical run time of the simulation process is shortened.

Keywords: deterministic and stochastic Petri nets, simulation software

1. Bevezetés

A determinisztikus és sztochasztikus Petri hálók olyan folyamat leírására alkalmasak, melyben a késleltetések eloszlása determinisztikus vagy exponenciális lehet, vagyis a DSPN a determinisztikus átmenetek megengedésével tekinthető a sztochasztikus Petri háló (SPN) kiterjesztésének is. A DSPN így nagyobb modellezési erőt jelent az SPN-nél, azonban a modell kiértékelésekor nagyobb korlátokba ütközünk, mint az SPN esetében. Általánosan csak abban esetben tudunk módszert a háló analitikus megoldásának előállítására, ha a háló minden állapotában legfeljebb egy determinisztikus átmenet engedélyezett [1]. Mivel ez egy erős megszorítás, a gyakorlatban sokszor merül fel az igény szimulációs technikák alkalmazására.

Bár Petri hálók szimulációjára számos alkalmazás született, s ezek közül több képes DSPN analitikus megoldására vagy szimulációjára, az egyes szoftverek mindegyike esetében hiányosságokkal szembesültünk. A DSPNExpress szoftver új, hatékony eljárásokat tartalmazott a DSPN-ek analitikus megoldására, ennek ellenére a szoftvert ma már nem fejlesztik tovább, és a régi változat sem elérhető.

A TimeNET szoftvert a mai napig fejlesztik az ilmenauai műszaki egyetemen (Technische Universität Ilmenau). Fejlesztői az általánosan alkalmazott eljárásokat építették bele a Petri hálót megoldó modulokba. Jól parameterezhető a szimulációkhoz tartozó statisztikai modul. Grafikus felülettel rendelkezik, biztosítva a Petri hálók hatékony tervezését. A szoftver legfőbb hátránya, hogy a grafikus felületen nehézkes a szimulációs eredmények generálása, mivel a szimulációkat csak egyesével, a grafikus felületen történő parameterezéssel futtathatjuk. További hátránya a szoftvernek, hogy a tüzelési politika nem állítható paraméter, hanem minden esetben az „enabling memory” szabályt [1] alkalmazza.

A GreatSPN szoftvert a torinói egyetemen (Università di Torino) kezdték fejleszteni 1985-ben. Az eszköz azóta számos változáson ment keresztül. A legnagyobb hátránya, hogy a jelenlegi verzió instabil, emellett a TimeNET-hez hasonlóan nehézkes a szimulációs eredmények tömeges generálása.

A DSPNSim a szimulációs eredmények hatékony generálására született meg. Grafikus felülettel nem rendelkezik, a szimulálni kívánt DSPN-t és a paramétereket egy XML fájl segítségével írhatjuk le. Az eredmények pontosságát a programhoz szorosan kapcsoló statisztikai modul biztosítja.

A cikk további felépítése a következő. A második fejezetben röviden összefoglaljuk a DSPN formalizmus alapjait. A harmadik fejezetben leírjuk a DSPNSim felépítését, míg a negyedik fejezetben összehasonlítások segítségével értékeljük annak eredményességét. Az ötödik fejezetben összefoglaljuk a cikket és megemlítjük a továbbfejlesztési lehetőségeket.

2. DSPN formalizmus

Ebben a fejezetben rövid leírást adunk a DSPN formalizmusról, míg részletes leírást például Ajmone Marsan és Chiola munkájában található [1].

A DSPN formálisan egy irányított gráf, amely két típusú csomópontot tartalmaz: helyeket és tranzíciókat. A helyek a rendszer állapotváltozóinak, míg a tranzíciók az állapotváltozásokat előidéző eseményeknek feleltethetők meg. A háló élei a helyeket kötik össze a tranzíciókkal, s a rendszerállapotok és a rendszerben értelmezett események kapcsolatát írják le. A helyek tokeneket tartalmazhatnak, s rendszerállapotok leírását az adja, hogy az adott állapotban hány token van a háló helyein. Általánosságban a rendszerállapotot

M -mel jelöljük, míg $M(p)$ jelöli az M rendszerállapotban a p helyen lévő tokenek számát. A következőkben néhány DSPN-nel kapcsolatos alapfogalmat írunk le.

A DSPN rendszer a következő paraméterekkel írható le:

$$(P, T, I, O, H, M_0, \tau, \omega, e_{t_1, t_2}).$$

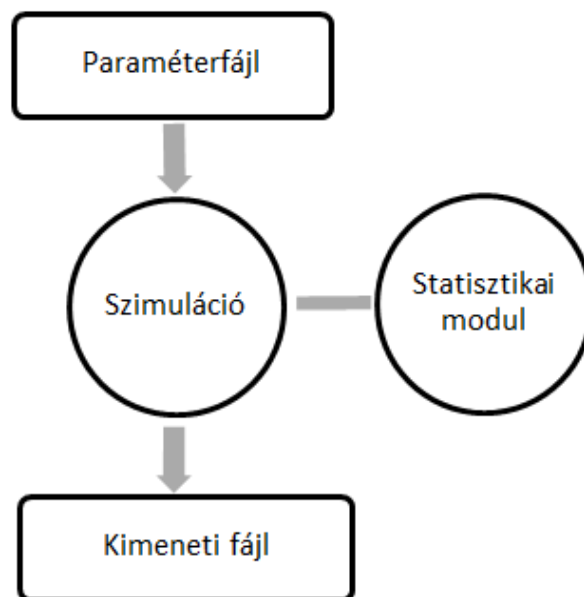
ahol:

- P a helyek véges halmaza. Egy $M \in \mathbb{N}^{|P|}$ rendszerállapot minden $p \in P$ helyen lévő tokenek számát definiálja.
- T is a tranzíciók véges halmaza, amely a következő diszjunkt halmazokra osztható: T^Z a rögtöntranzíciók halmaza, T^E az exponenciális tranzíciók halmaza, míg T^D a determinisztikus tranzíciók halmaza. $M \cap T = \emptyset$.
- $I, O, H : \mathbb{N}^{|P|} \rightarrow \mathbb{N}$ sorrendben a p -ből t -be menő bemenő élek, a t -ből p -be menő kimenő élek, és a p -ből t -be menő élek multiplicitása.
- $M_0 \in \mathbb{N}^{|P|}$ a kezdeti rendszerállapot.
- $\tau : \mathbb{N}^{|P|} \rightarrow \mathbb{R}^+$ az átlagos késleltetés $\forall t \in T^E \cup T^D$ esetén (τ függhet az aktuális rendszerállapottól).
- $\omega : \mathbb{N}^{|P|} \rightarrow \mathbb{R}^+$ a tüzelési súly $\forall t \in T^Z$ esetén (ω függhet az aktuális rendszerállapottól).
- $e_{t_1, t_2} : \mathbb{N}^{|P|} \rightarrow \{R, C\}$ a t_2 -re alkalmazott a tüzelési politika, amikor t_1 tüzel $\forall t_1 \in T^Z \cup T^E, t_2 \in T^D$ (e_{t_1, t_2} függhet az aktuális rendszerállapottól). R és C sorrendben az újraindítás (restart) és a folytatás (continue).

3. A DSPNSim program leírása

A programot felépítését és működését tekintve a következő részekre bonthatjuk fel (1. ábra):

- szimulációt végző egység,
- statisztikai modul,
- beolvasást végző egység.



1. ábra. A program felépítése

A szimuláció megkezdése előtt a külső paraméterfájl beolvasásra kerül. Ebben a fájlban adhatjuk meg a szimulálni kívánt modell Petri-hálóját és további, szimuláció szempontjából fontos paramétereit. Az adatok leírása XML formátumban történik, ezért a háló megadása nem igényel külön eszközt, valamint átlátható hierarchiát eredményez.

A Petri-háló megadásánál meghatározhatunk helyeket és különböző tranzíciókat is. Helyek esetében kötelező megadni nevet, opcionális paraméterként definiálhatunk kezdeti tokenszámot, ennek segítségével megadható a kezdeti tokeneloszlás a modellben.

A tranzíciók megadásánál kötelezően meg kell határozni egy azonosítót, továbbá megadható a bemenő illetve kimenő helyek listája, valamint az egyes helyekhez tartozó élsúly is. Az élsúlyoknál a nullás érték különleges szereppel bír, ez jelenti ugyanis a tiltó élt. A megadható tranzíciók közül először a rögtön tüzelő tranzíciót tekintjük át. Itt késleltetést nem definiálunk, hiszen ez a tranzíció várakozás nélkül mozgatja a tokeneket. Opcionális paraméterként azonban megadhatunk prioritást, melynek segítségével a konkurens tranzíciók között definiálhatjuk a tüzelési sorrendet. Exponenciális késleltetésű tranzíció esetén az átlagos késleltetést tudjuk megadni, amely a tüzelési intenzitás reciproka. Megadhatjuk a kiszolgáló típusát is, amely exkluzív illetve végtelen kiszolgáló lehet. Végtelen kiszolgáló esetén a tüzelési intenzitás a bemeneti helyeken található tokenek számától függ, exkluzív esetében ez az intenzitás konstans. Rögtön tüzelő tranzíciók és exponenciális tranzíciók leírásakor tudjuk megadni a determinisztikus tranzíciókra vonatkozó gyújtási politikát is. A gyújtási politika azt határozza meg, hogy egy adott $t \in T^E \cup T^Z$ tranzíció tüzelése esetén egy $t \in T^D$ tranzíció tüzelési eseményét újrageneráljuk-e. Az alapértelmezett szabály a *folytatás*, ez lép életbe, ha nem adunk meg gyújtási politikát. A determinisztikus késleltetésű tranzíciókat az exponenciálisához hasonlóan a késleltetéssel definiáljuk.

A Petri-hálón kívül a szimuláció beállításai is a bemenő paraméterfájlban írhatók le, meghatározhatjuk a szignifikáns helyeket, amelyekre a szimuláció a pontosságot mérni fogja, eldönthetjük, hogy a szimuláció rendelkezzen-e felfűtési időszakokkal, valamint, hogy ez a felfűtési időszak milyen pontosság elérése után fejeződjön be. Definiálhatunk minimális tranziens illetve minimális futási időt, továbbá maximális futási időt is. A szignifikáns értékek vizsgálata szempontjából megadható a konfidenciaszint és a maximális relatív hiba. A program lehetőséget ad kötegetelt futtatásra, ez az opció is ebben a fájlban határozható meg. Végül megadhatjuk a kimeneti fájl nevét és helyét.

A paraméterfájl beolvasása után a szimulációt végző egység a beolvasott hálón a megadott paraméterek mellett elkezd a szimulációt. Ebben az egységben található a logika túlnyomó része, ez a program központi része, mely a többi osztályt és annak metódusait mint szolgáltatást hívja meg. A futás végét háromféleképpen érhetjük el: *i*) az események elfogynak az eseménysorból; *ii*) elérjük a maximális futási időt; *iii*) elérjük a megadott pontosságot a szignifikáns helyek vizsgálatakor.

A pontosság vizsgálata a szimuláció minden iterációjában megtörténik, erre a mérésére a statisztikai modul ad lehetőséget. Ez a modul csupán a szimulációs modul által használt metódusokat tartalmazza, s a metódusok kifejezetten szimulációs célra optimalizált eljárásokat használnak.

Az itt található intervallumbecslő függvény a szignifikáns helyek átlagos tokenszámára határozza meg a konfidenciaintervallumot (a paraméterfájlban megadott paraméterek mellett). Az adott pontosság elérésének vizsgálata során az intervallum hosszát vetjük össze az általunk definiált megengedett hibával. A 3.1-es egyenlőtlenség a pontosság vizsgálatát írja le. A képlet bal oldala a maximális relatív hibát fejezi ki, a jobb oldal pedig az általunk definiált megengedett hiba mértékét jelenti az adott hely átlagos tokenszámára nézve.

$$z_{\alpha/2} \cdot \hat{S}_{\bar{y}} < \bar{x} \cdot e \quad (3.1)$$

Mivel ez a vizsgálat a szimuláció minden iterációjában kiszámításra kerül, ezért fontos, hogy a szükséges paraméterek számítása is költséghatékony legyen. Az aktuális iterációnál történő számításoknál a megelőző állapotban meghatározott eredmények kerülnek felhasználásra. Ez az átlagszámításnál súlyozott átlag használatával valósult meg (3.2-es egyenlet).

$$\bar{x}_n = \frac{(\sum_{i=1}^{n-1} t_i) \cdot \bar{x}_{n-1} + t_n \cdot x_n}{\sum_{i=1}^n t_i} \quad (3.2)$$

A szórásnégyzet számításához használt képletnél is ez a logikát használtuk (3.3-as egyenlet).

$$S_{n+1}^2 = \frac{\sum_{i=1}^n t_i}{\sum_{i=1}^{n+1} t_i} S_n^2 + \frac{t_{n+1} (x_{n+1} - \bar{x}_n)^2}{\sum_{i=1}^{n+1} t_i} + d^2 - 2d\bar{x}_{n+1} + 2d\bar{x}_n, \quad (3.3)$$

ahol

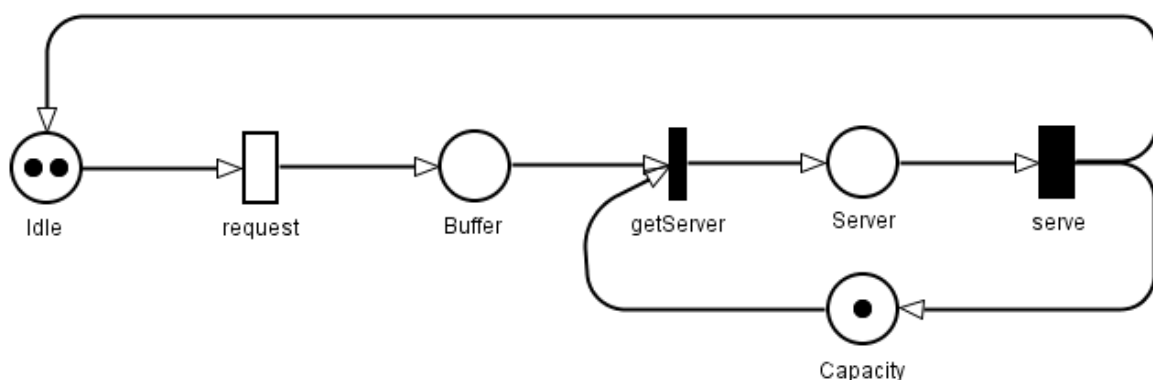
$$d = \bar{x}_{n+1} - \bar{x}_n. \quad (3.4)$$

Miután a szimuláció véget ért, a program a megadott fájlba kiírja a szimulációs eredményeket.

4. Az eredmények értékelése

Ebben a fejezetben a szimuláció által szolgáltatott eredményeket vizsgálunk meg két modell esetében: először egy egyszerűbb rendszert vizsgálunk, melynek könnyen megkapható az analitikus megoldása, majd egy összetettebb rendszert, melyben az eredményeket összevetjük a TimeNET [6] által meghatározott eredményekkel.

Az egyszerűbb rendszer az M/D/1/2/2-es sor. A vizsgálatot 95%-os konfidenciaszinten, 1%-os maximális hiba mellett végeztük. A következő ábrán a sornak megfelelő Petri-hálója látható (2. ábra).



2. ábra - Az M/D/1/2/2-es rendszer Petri hálója

2. táblázat – A TimeNET és a DSPNSim eredményeinek összevetése

Helyek	TimeNet	DSPNSim
Accepted	189.81933	190.8610859
CutMembers + RecutMembers	0.13467	0.133401
Planned + RePlanned	0.63745	0.63921723
Molded + ReMolded	3.25834	3.261711721
Glued + ReGlued	0.10666	0.107125
Dipped + ReDipped	0.70611	0.710063224
Sanded + ReSanded	0.35967	0.356171065
PrimerApplied+ PrimerReapplied	0.35056	0.354114951
Coated	584.55711	592.564000
Fixed	0.52422	0.510605551
Joined	3021.378	3017.342600

5. Összefoglalás

Bár sok esetben hasznos a determinisztikus és sztochasztikus Petri hálós modellek alkalmazása, numerikus megoldásuk gyakran problémákba ütközik, az elérhető szimulációs szoftverek pedig különböző korlátokkal rendelkeznek, főként a tömeges eredmények generálása nehézkes. A DSPNSim grafikus felület nélkül, egy paraméterfájl segítségével futtatható, így hatékonyan szállítja az eredményeket.

A tisztán szimulációs szoftverek természetesen nem korlátozódnak a DSPN-ekben megengedett eloszlásokra. A program egyik továbbfejlesztési iránya a további eloszlások beépítése, illetve ennek megfelelően a gyűjtési politika finomítása. A másik továbbfejlesztési irány a szimuláció pontosságának meghatározása tranziens szimuláció esetén, ugyanis a jelenlegi verzióban a szimuláció pontosságának növelése a független futtatások számának növelésével lehetséges.

Irodalomjegyzék

- [1] M. A. Marsan and G. Chiola, „On petri nets with deterministic and exponentially distributed firing times,” in *Advances in Petri Nets 1987*, pp. 132-145, Springer Berlin Heidelberg, 1987.
- [2] C. Lindemann and G. S. Shedler, „Numerical analysis of deterministic and stochastic petri nets with concurrent deterministic transitions,” *Perform. Eval.*, vol. 27-28, pp. 565-582, Oct. 1996.
- [3] C. Lindemann and A. Thümmler, „Transient analysis of deterministic and stochastic petri nets with concurrent deterministic transitions,” *Performance Evaluation*, vol. 36, pp. 35-54, 1999.
- [4] C. Lindemann, „DSPNexpress: A software package for the efficient solution of Deterministic and Stochastic Petri Nets.” In: *Proc. 6th Int. Conf. on Modelling Techniques*

and Tools for Computer Performance Evaluation, Edinburgh, Great Britain, pp. 15–29 (1992)

- [5] E. A. Gilberto, S. Donatelli: „DSPN-Tool: A New DSPN and GSPN Solver for GreatSPN,” In Proc. of the Seventh International Conference on the Quantitative Evaluation of Systems (QEST), pp.79-80, 15-18 Sept. 2010.
- [6] A. Zimmermann, M. Knoke, „TimeNET 4.0: A software tool for the performability evaluation with stochastic and colored Petri nets; user manual,” TU, Professoren der Fak. IV, 2007.
- [7] Á. Horváth, „Usability of Deterministic and Stochastic Petri Nets in the Wood Industry: a Case Study,” in Proc. of the 2nd International Conference on Computer Science, Applied Mathematics and Applications, pp. 119-127, 2014.

Calliope - Szakdolgozat Nyilvántartó és Követő Rendszer

Calliope – Thesis management system

Windisch Gergely^a, Fend Péter^b

^a Óbudai Egyetem, Neumann János Informatikai Kar
windisch.gergely@nik.uni-obuda.hu

^bNokia
peter.fend@nsn.com

Absztrakt: Egy felsőoktatási intézményben a szakdolgozatok zárják a hallgatók tanulmányait. Az elkészült szakdolgozatokat tárolni kell, így egy jól működő katalógus rendszer elengedhetetlen. A hosszú távú tárolás mellett egy megfelelő rendszer segítheti a konzulensek munkáját is, hogy az egy időben nagyszámú szakdolgozót rendszerezetten kezelhesse. A cikkben az Óbudai Egyetemen fejlesztett, jelenleg teszt üzemben használt szakdolgozat követő rendszert mutatjuk be. A rendszer több, mint egy egyszerű archívum, mivel a végleges dokumentumok tárolásán kívül alapvető funkciója a dolgozat készítése közben a haladás követése. A hallgató ide töltheti fel az aktuális állapotot, a konzulens ezt letöltheti, megjegyzéseket tehet, majd visszatöltheti ugyanide. Az alkalmazás felületet nyújt a szakdolgozati témakiírásoknak, a hallgatók ezeket böngészhetik, és a rendszeren keresztül van lehetőség a témára jelentkezni. A rendszernek része naptár és ütemező modul, verzió követő rendszer, valamint értesítést végző koordinációs alrendszer. A Calliope rendszer első verzióját 2013-ban vezettük be az Óbudai Egyetemen, azóta teszt üzemmódban használatban van, jelenleg a továbbfejlesztésén dolgozunk, amelynek keretében teljeskörű dokumentum tárrá lesz átalakítva.

Kulcsszavak: Calliope, szakdolgozat követő, diplomamunka

Abstract: The article describes Calliope – the thesis management system developed and used at Obuda University. It is a web application that helps students and advisors keep track of the theses. The students can upload up-to-date versions of the document that the advisors can download, comment and upload to the system, thus keeping all the files at a central location. The advisor can see up-to-date status info on all the documents making it easier to keep track of the multitude of students associated to him/her. The system has management capabilities as well: advisors can upload topics that students can see and apply to. Right now Calliope is used at Obuda University John von Neumann Faculty of Informatics as a beta version since 2013, we are currently working on upgrading it to Calliope 2.0 which will be a full fledged document archive and management system.

Keywords: Calliope, thesis management

1. Bevezetés

Egy egyetemen az oktatók sok hallgató szakdolgozatát és diplomamunkáját vezetik. A nagy számú hallgató nyilvántartása, haladásának folyamatos monitorozása elég komoly adminisztrációs terhet ró az oktatóra. A Calliope rendszert azért kezdtük el fejleszteni az Óbudai Egyetemen, hogy erre a problémára nyújtsunk egy automatizált megoldást.

A megvalósítandó rendszerrel nem csak egy dokumentum tárat szerettünk volna készíteni, ahová a hallgatók elkészült szakdolgozata felkerül, hanem egy olyan rendszert tűztünk ki célul, ami a dolgozat készítése közben központi szerepet kap a hallgató és a konzulens közötti kommunikációban, valamint az adminisztrációban. Olyan megoldást találtunk ki, ahová a konzulens bejelentkezik, látja egy helyen az összes szakdolgozatát, egy szakdolgozatot

kiválasztva láthatja annak jelenlegi státuszát és elérhető minden dokumentum friss változata. A hallgató ugyanide belépve a saját dolgozatát látja, ő is értesülhet a státuszról. Célunk volt, hogy a rendszerben a konzulensek kiírhassanak témákat, ezekre hallgatók jelentkezessenek. A Calliope első verziója egy hallgatói szakdolgozat keretében készült [1], jelenleg többen dolgoznak a továbbfejlesztésén.

1. 1. Szakdolgozatok és diplomamunkák az Óbudai Egyetemen

Az Óbudai Egyetem Neumann János Informatikai Karán a BSc képzésen szakdolgozatot, az MSc képzésen diplomamunkát készítenek a hallgatók. A tanterv szerint mindkettő két féléves, a képzés sikeres befejezéshez teljesíteni kell a Szakdolgozat I és II, illetőleg a Diplomamunka I és II tantárgyakat.

Mivel a két dolgozat típus között adminisztratív nincs különbség, így mostantól szakdolgozat néven hivatkozunk a dolgozatra, függetlenül attól, hogy BSc-s szakdolgozat, vagy MSc-s diplomamunka-e.

1. 2. Fontosabb mérföldkövek szakdolgozat készítésekor

Egyetemünkön a szakdolgozatok esetében a következő ügyrendet kell tartani:

- Szakdolgozat I tárgy felvétele (szemeszter kezdete előtt a tárgyfelvételi időszakban)
- Téma és konzulens választása (szemeszter elején)
- Témacím leadása magyarul és angolul, konzulenssel aláíratva (szemeszter 2. hete)
- Feladatkiírás elkészítése és leadása (szemeszter 4. hete)
 - Feladatkiírás elfogadása (1-2 hét a leadás után)
- Szakdolgozat I tárgy teljesítése - irodalom kutatás + szóbeli beszámoló (vizsgaidőszak 1. hete)
- Szakdolgozat II tárgy felvétele (szemeszter kezdete előtt a tárgyfelvételi időszakban)
- Szakdolgozat befejezése
- Szakdolgozat leadása (szorgalmi időszak vége előtt 2 héttel)
- Szakdolgozat megvédése államvizsgán (záróvizsgakor)

A Calliope rendszer jelenleg ezeknek a feladatoknak csak egy részét látja el, a későbbiekben tervezzük a teljes életciklus lefedését és központosított adminisztrációját.

2. Hasonló rendszerek

A Budapesti Műszaki Egyetemnek létezik egy hasonló rendszere, a Diplomaterv Portál [2], amely egy elektronikus szakdolgozat és diplomaterv nyilvántartó és archiváló rendszer. Ennek a rendszernek a célja a BME-VIK szakdolgozat készítéssel kapcsolatos tantárgyainak (Szakdolgozat készítés valamint Diplomatervezés) teljes ügymenetének elektronikus formában való lebonyolításának segítése. Feladata minden szakdolgozatokhoz kapcsolódó dokumentum tárolása és archiválása. A rendszernek nem hatásköre a szakdolgozati témák meghirdetése csak a meglévő, futó szakdolgozatok kezelése.

Lehetőség van a szakdolgozat dokumentumaiból több verziót is feltölteni. Egy változási naplóban minden dokumentum vagy egyéb adat változás nyomon követhető.

Ez a rendszer a szakdolgozatok teljes életútját nyomon követi a szakdolgozati téma kiírásától a szakdolgozati bírálat elkészültéig egy meglehetősen komplex folyamatban.

A portál lehetőséget ad a vendég felhasználóknak a témák böngészésére. Rendelkezésre áll szabad szavas illetve kulcsszó alapú keresés, valamint lehetőség van böngészésre tanszékek illetve félévek szerint.

A szakdolgozatok feladatainak menedzsmentjére semmilyen eszközt nem ad. A folyamatot az ügyvitel szempontjából közelíti meg, nem pedig a szakdolgozatok és azok szereplői felől.

kezdőlap keresés

Írja be a keresett szót

Kezdőlap » Keresés és böngészés » Témák böngészése

Témák böngészése

Ezen az oldalon a Portálra feltöltött szakdolgozat és diplomatervezési témák között böngészhet.

LISTÁZÁSI FELTÉTELEK

Tanszék:

Konzulens: Konzulens választásához előbb válassza ki a tanszéket!

Félév:

Szak:

Képzés:

Kulcsszó:

1. ábra: BME Diplomatervezési Portál

3. Calliope rendszer

3.1. Célok

A fejlesztés kezdetekor egy olyan "ügyvivő" szoftvert szeretnénk volna megvalósítani, ami a konzulens számára megkönnyíti a szakdolgozók nyilvántartását. Saját és kollégák tapasztalata az volt, hogy 8-10 konzultált hallgatónál már problémás számon tartani, hogy ki hogy áll. „Le lett-e adva a feladatkiírás?” „El lett-e fogadva a feladat kiírás?” „Mi is volt az utolsó verzió, amit átküldött a hallgató?” „Átnéztem mindenkinek a dolgozatát?”

Ezeket a problémákat mindenki saját adminisztratív megoldásokkal oldotta meg korábban - jegyzetomb, excel tábla, post-it cetlik az iroda falán. A nyilvántartáson kívül logisztikai probléma is felmerült sokszor a dolgozat küldözgetésekor (melyik az új fájl, mikori a legfrissebb verzió - az, amit emailben küldött, vagy valamikor hozott lemezen egyet stb.).

A Calliope főként ezekre a problémákra hivatott egy központosított választ adni.

Az elképzelésünk egy olyan webalkalmazás volt, ahova a konzulens beléphet, láthatja az összes szakdolgozóját a dolgozatok éppen aktuális státuszával egyetemben, azonnal látszik, ha valamelyik dolgozattal valami tennivaló van (pl. nincs még leadva a feladatkiírás, vagy beérkezett egy új verzió, amit el kellene olvasni). A hallgató pedig csak a saját dolgozatát látja, szintén azonnal látható számára a státusz. A kitalált alkalmazásba szándékoltnak nem terveztünk túl nagy szabadságfokot, hogy a konzulenseknek ne kelljen saját megoldásokat kitalálni, egyszerűen tudják követni a standard ügyrendet. Természetesen számoltunk azzal, hogy szükség lesz valamiféle kivételkezelésre bizonyos esetekben, ezeket terveztük beilleszteni később az alkalmazásba.

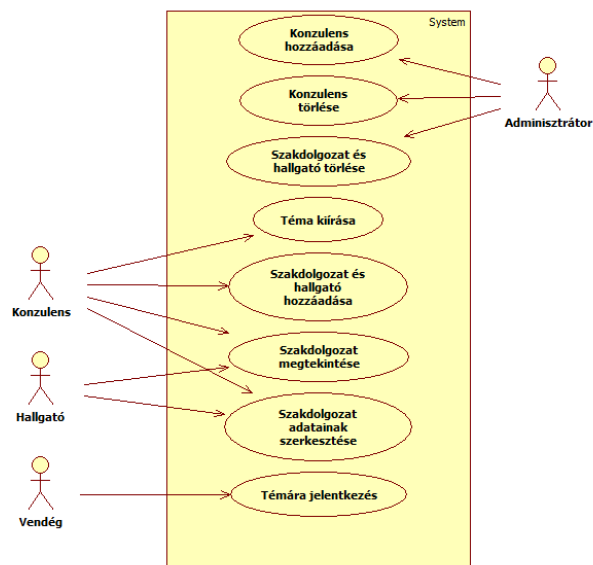
Mind a hallgatónak, mint a konzulensnek van lehetősége a dolgozat új verzióját feltölteni - a konzulens jellemzően a korrektúrázott, megjegyzéseivel ellátott verziót tölti inkább fel. Ezzel megoldódik az állandó email küldés problémája, és a legfrissebb, valamint minden korábbi változat egy helyen, a szerveren megtalálható.

Ugyanígy évekkel később a konzulens bármikor elérheti bármelyik korábbi hallgatójának dolgozatát, nem kell saját magának az archiválással bajlódnia.

A fájlok feltöltésénél kétféle dokumentum feltöltésére gondoltunk: a feladatkiírás és maga a dolgozat. Jellemzően mindkettő több iterációs körön megy keresztül a konzulens és a hallgató között, ezt nagyon megkönnyítheti a központi feltöltés és elérhetőség.

3.2. Funkciók

- Hallgatóként témakiírások böngészése
- Konzulensként témák kiírása
- Hallgatóként témára jelentkezés
- Konzulensként a jelentkező hallgató elfogadása vagy elutasítása. Elfogadás esetén automatikusan elkészül a szakdolgozat a rendszerben.
- Konzulensként kézi szakdolgozat létrehozás - jellemzően a hallgató saját témája alapján.
- A szakdolgozat státuszának beállítása konzulensként.
- Fájlok feltöltése
- Ütemterv készítés, naptár bejegyzések létrehozása

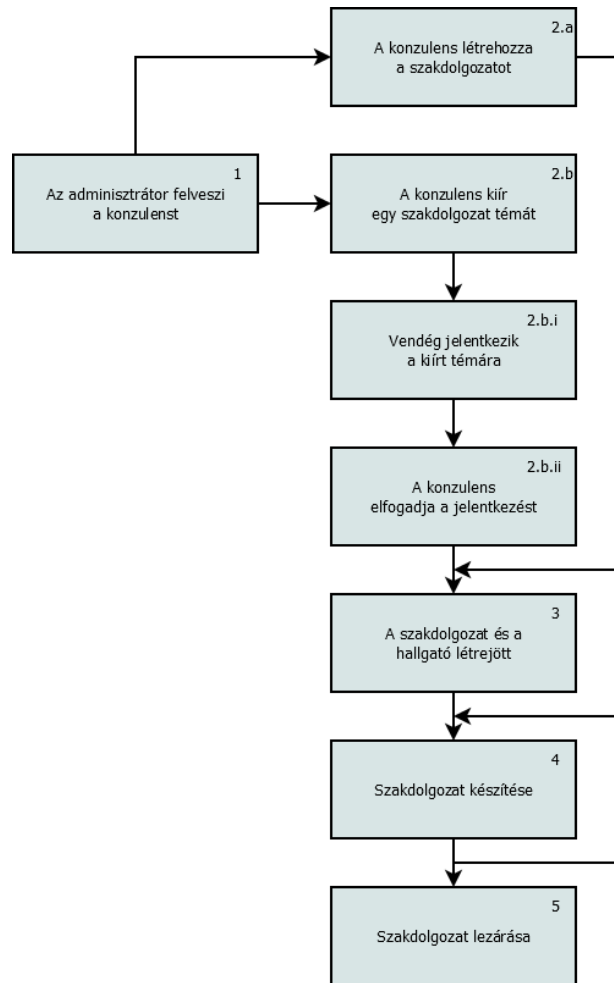


2. ábra: Calliope rendszer felhasználóinak funkciói

3.3. Implementáció részletei

A fájl tárolás mögött egy SVN szerver [3] van, ennek segítségével a változás kezelést, verzió követés automatikusan megoldódik. Úgy van megvalósítva a fájlok tárolása, hogy létrehozunk egy dolgozat állományt, majd pedig ennek újabb verzióit "commit"-oljuk, vagyis küldjük be az SVN szervernek. Eredeti terveink szerint ez megfelelő volt, mivel a dolgozat fájl újabb verziói bekerülnek így a központi tárbá. A való életben azonban a megoldás több hátránya is előjött. A legnagyobb probléma jelenleg az, hogy a gyakorlatban nem áll meg az az elképzelés, hogy egy dolgozat - egy fájl. A hallgatók és konzulensek az állományokat át szeretnék nevezni (pl. verziószámok bevezetése a fájlba - KissJolan_dolgozat_v2_formazott.doc), gyakran térnek át egyik formátumból a másikba (elkezdik a szöveget Microsoft Officeban írni, majd áttérnek Libre Office csomagra, és megváltoztatják a kiterjesztését az állománynak), vagy néha doc, néha véglegesített pdf formátumban küldik át a konzulensnek a friss verziót.

Jelenleg ezeket helyzeteket a Calliope mögött futó SVN nem tudja kezelni, mivel az SVN úgy van kitalálva, hogy van egy fájl, akkor annak az újabb verzióit tárolja, átnevezetten, más formátumban stb. már nem ugyanannak a fájlnak számít.



3. ábra: Szakdolgozat életciklusa a Calliope rendszerben

4. Elkészült rendszer

A Calliope rendszer PHP nyelven készült, Yii keretrendszer [4] használatával. Több másik MVC keretrendszer megvizsgálása után döntöttünk a Yii mellett annak kiterjedt funkcionalitása, biztonsági megoldásai, alapos dokumentációja és elterjedtsége miatt. A fájlok tárolására SVN szerveret használunk, a rendszert kiszolgáló adatbázis motor PostgreSQL, webserverként pedig egy apache szolgálja ki a kéréseket. A tervezésnél és a megvalósításnál egyaránt fontos szerepet játszott a biztonság, az alkalmazás XSS [5] és SQL injection [6] támadások ellen is védve van. Az előzetes tervek megvalósításra kerültek, Calliope használható szakdolgozatok írása közbeni adminisztrációra. A rendszer része lett egy levelező modul, ami folyamatosan email értesítéseket küld az összes érintettnek, amennyiben valami változik a dolgozat állapotával. Implementálásra került egy naptár modul is, ahol az ütemterv fontosabb állomásait lehet feltüntetni és megfigyelni. Gantt diagramok létrehozására és követésére is van lehetőség az ütemterv modul segítségével.

Szakdolgozat Címe	Hallgató
Közeledési lámpa felismerő és figyelmeztető rendszer	Pellicer Róbert
Képfeldolgozáson alapuló automatikus könyvkategorizáló alkalmazás	Mihályi Martin
Jegyzetelő alkalmazás labirintgore	Farkas Tamás
Augmented Reality könyvkatalógus	Kiss Árpád
EasyPic - fényképválogató alkalmazás mobiltelefonra	Durgó Veronika
Képfeldolgozáson alapuló körülmérés	Vilics Péter
Fényképválogató alkalmazás okostelefontáblat PC-ra	Vincze Imre
Apilis szoftverfejlesztési keretrendszer tervezése és fejlesztése	Kurucz András
Képfeldolgozáson alapuló könyvkatalógus alkalmazás	Jenei Zoltán
GPS alapú flottakövető rendszer	Tiborcz Gábor

4. ábra: Szakdolgozatok listája

Mérőidő	Tejlesztve
Cím leadva	2012/05/09 12:02
Cím elfogadva	2012/05/09 12:02
Feladatírás leadva	2012/05/09 12:02
Szakdolgozat 1 tartárgy tejlesztve	2012/05/09 12:02
Szakdolgozat 2 tartárgy felvéve	2012/05/09 12:02
Szakdolgozat leadva	2012/05/14 18:37
Szakdolgozat 2 tartárgy tejlesztve	2012/05/30 09:38
Szakdolgozat 1 tartárgy felvéve	2012/05/09 12:02
Feladatírás elfogadva	2012/05/09 12:02

5. ábra: Egy adott dolgozat

4.1. A pilot rendszer üzemeltetésekor felmerült problémák, tapasztalatok

A rendszer alapvetően jól vizsgázott az első éles bevetéskor. A tesztidőszakban 13 konzulens és 56 regisztrált hallgató használta az alkalmazást. A felmerülő hibákat és továbbfejlesztési ötleteket a kari Redmine szerveren gyűjtjük, mindezidáig 29 hiba bejegyzés, valamint 49 továbbfejlesztési ötlet került be. A bugok egy része programozási hibából eredt, pl. üres fájlnev esetén is megpróbált feltölteni dokumentumot, de nem sikerült neki és server error üzenettel tért vissza az alkalmazás, ezek azonnal kijavításra kerültek. Előkerültek olyan hibák is, amelyek inkább a tervezési problémákból következnek, itt a hibák egy részét javítottuk, egy részét továbbfejlesztési lehetőségnek jelöltük meg. A felhasználókat leginkább zavaró probléma a korábban már említett átnevezhetetlenség okozza, ami az SVN használatából ered. Többen javasolták még egy alaposabb kereső implementálását is, jelenleg a témakörök szűrése nem megoldott, így ha valóban beindul a széleskörű használat, akkor a hallgatók nehezen fognak tudni szűkíteni azokra a témákra, amelyek őket érdekelnék.

5. Továbbfejlesztési tervek

Az első éles tesztelési fázissal a fejlesztés első szakasza lezárult. A rendszert használó hallgatók és konzulensek alapvetően elégedettek vele. A pilot projekt után a fejlesztés a második ütemébe lépett, ami lényegében a továbbfejlesztésről szól. Két fő csapásiránya van a fejlesztéseknek: egyrészt továbbfejlesztjük az 1.0-ás változatot, hogy a jelenleg elérhető funkciók kényelmesebbek legyenek (pl. keresés, dolgozatok szűrése, friss dolgozat megjelenítése stb.), ennél nagyobb lélegzetvételű fejlesztési feladat a 2.0-ás verzió elkészítése, ami lényegében a teljes szakdolgozat életciklust fel tudja ölelni majd, valamint kiegészül egy kari szintű dokumentumtárral. Sok olyan fejlesztés van, amit technikailag egyszerű volna megoldani, ám a kényelmes és egyértelmű megoldást még nem sikerült egyértelműen megtalálni (ilyen például az az ötlet, hogy célszerű volna a témákat tagelhetővé - címkézhetővé tenni – pl. képfeldolgozás, robotika, hálózatok, biztonság stb. Itt felmerül a kérdés, hogy a címkék hogyan legyenek kezelve? Szabadon megadható legyen a sok címke?

Ekkor fennáll a veszélye, hogy a konzulensek ugyanazt a címkét másképp írják, és a hallgatók nem fogják tudni, hogy mire keressenek (pl. biztonság, hálózat biztonság, security). Legyen egy címke lista? Ez ekkor egyértelmű, de hogy kezeljük le, ha nincs meg egy adott címke?).

5.1 Calliope 2.0

A jelenlegi verzió javításával, fejlesztésével párhuzamosan folyik a munka a következő fő verzió elkészítésére. Jelen pillanatban a tervezési fázisban vagyunk, az igények alapján készül a specifikáció. A legfontosabb változásokat röviden összefoglaljuk.

Teljes szakdolgozati életciklus követése

A szakdolgozati témák megjelennek a Calliopén, a hallgató innen választhat, a konzulens a jelentkezést elfogadhatja, a szakdolgozat ezzel a rendszerben létre jön, azonban ez jelenleg független az egyetemi szintű ügymenettől. A hallgatónak ki kell töltenie egy formanyomtatványt a szakdolgozat címével, konzulens nevével és egyéb alapvető adatokkal, majd azt le kell adni a TO-n. Ezt követően a Calliope rendszeren belül elkészülhet a feladatkiírás, ám ezt szintén off-line módon kell benyújtani a TO felé - és így megy ez egészen a szakdolgozat leadásáig. A célunk az, hogy a teljes életciklus minden fázisa a Calliopén belülről elérhető legyen. Tehát hogy a hallgató jelentkezik egy kiírt témára, erről a TO automatikusan értesül. Elkészül a feladatkiírás, akkor a TO-s szereplő innen töltheti le, és küldheti tovább bírálatra a kari vezetőség irányába. Az elkészült végleges szakdolgozatot a Calliopéból tudná elérni a bíráló, és ide töltené fel a bírálatát.

Ennek megoldásához új szereplőkre van szükség a rendszerben (TO, bírálók), de a fő nehézségét inkább az adja, hogy itt mindenképpen egy kényelmes megoldást kell kialakítani a felhasználók számára, hogy az új rendszer bevezetése ne nyűg legyen, hanem az életüket megkönnyítő segítség.

A másik nagyobb feladat a kari szintű dokumentum tár kialakítása lesz, ebben már nem csak szakdolgozatok, hanem többféle dokumentumok, fejlesztési projektek, tananyagok is felkerülhessenek a rendszerbe. Itt ki kell még dolgozni a jogosultság kezelés bonyolult kérdéskörét.

6. Összefoglalás

A cikkben bemutatásra került az Óbudai Egyetem Neumann János Informatikai Karán fejlesztett Calliope nevű szakdolgozat követő rendszer, ami megkönnyíti a szakdolgozatok kiírását, a témára jelentkezést valamint a dolgozat készítése közben az adminisztrációt. A Calliope nem egyszerűen egy adattár, ahol a kész dolgozatok tárolódnak, hanem egy olyan rendszer, ami a teljes életciklus alatt végigköveti a szakdolgozatot, készítés közben ide tölthető fel az új verzió mind a hallgató, mind a konzulens által, a szakdolgozat aktuális státusza beállítható, hogy a nyomon követés könnyebb legyen.

Köszönetnyilvánítás

Szeretnénk köszönetünket nyilvánítani az Óbudai Egyetem Neumann János Informatikai Karának a fejlesztés- és az üzemeltetés alatt nyújtott támogatásért, valamint minden konzulens és hallgató kollégának, akik a teszt üzem alatt használattal értékes visszajelzésekkel segítettek a rendszer továbbfejlesztését.

Irodalomjegyzék

- [1] Fend Péter: Szakdolgozat követő rendszer megvalósítása nyílt forráskódú eszközökkel, ÓE-NIK szakdolgozat, 2012
- [2] Tevesz, G.: BME-VIK. Diplomaterv Portál. Felhasználói útmutató. (<https://www.vik.bme.hu/files/00005962.pdf>)
- [3] Collins-Sussman, B., W. Fitzpatrick, B., Pilato C, M: Version Control with Subversion. O'Reilly Media, 2008
- [4] Yii Framework: Fundamentals: Model-View-Controller (MVC). (<http://www.yiiframework.com/doc/guide/1.1/en/basics.mvc>)
- [5] Fleiner, R.: SQL injekcióra épülő támadások és védekezési lehetőségek. Hadmérnök. 2008., 3. kötet, 4.
- [6] Yii Framework: How to write secure Yii applications. (<http://www.yiiframework.com/wiki/275/how-to-write-secure-yii-applications>),

Az „Elektronikus Oktatási Környezetek” tananyag bemutatása

Introducing the „e-learning environments” study material

Balla Tibor^a, Kocsis Gergely^b

^aDebreceni Egyetem, Informatikai Kar, Informatiótechnológia Tanszék

balla.tibor@inf.unideb.hu

^bDebreceni Egyetem, Informatikai Kar, Informatikai Rendszerek és Hálózatok Tanszék

kocsis.gergely@inf.unideb.hu

Absztrakt: A Gyires Béla Tananyag Tárház tananyag fejlesztési projekt keretében a szerzők elektronikus tananyagot készítettek Elektronikus Oktatási Környezetek címmel [1]. Az előadásban az elkészült tananyag kerül röviden bemutatásra. Az előadásban a fejezetek rövid tartalmi ismertetése után rámutatunk, hogy hogyan illeszkedik a téma a jelenleg is meglévő kurzusok tematikájába.

Kulcsszavak: e-learning, elektronikus oktatási környezet, moodle

Abstract: Within the framework of the „Gyires Béla Study Material Library” study material developing project the authors have created the study material titled e-Learning environments. In the presentation on the conference and here in this paper a short introduction is given to this material. After presenting the material chapter-by-chapter the authors point out how this material fits to the tematics of the current subjects.

Keywords: e-learning, moodle

1. Bevezetés

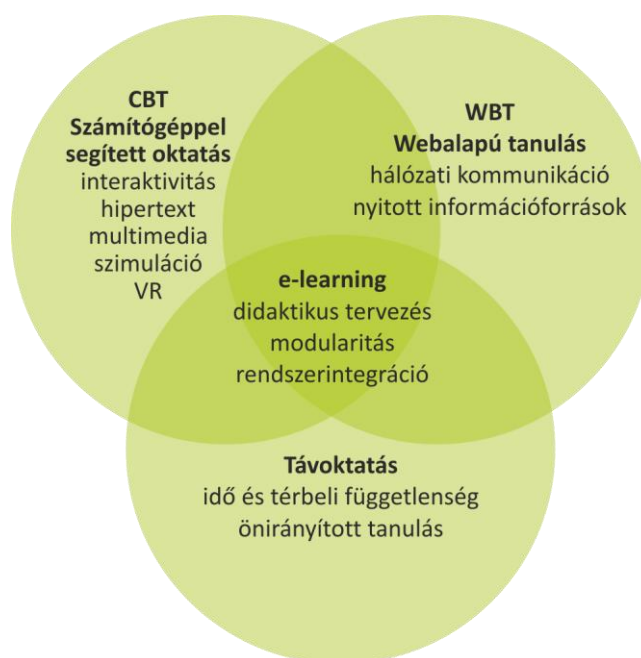
Napjainkban rengeteg szöveggörnyezetben elhangzik az a kifejezés, hogy e-learning, sokszor helytelenül használva, félreértelmezve, vagy nagyon sajátos értelmet kölcsönözve ennek a szónak. Az elmúlt évek során rengeteg koncepció került kidolgozásra, melyek újabbnál újabb fogalmat, hoztak be az e-learning világába, egyre átláthatatlanabbá téve, ezt az egyébként túlterhelt fogalmi rendszert. Mindezek ellenére az e-learning a mai napig sem rendelkezik teljesen egységes és kiforrott fogalmi meghatározással. Ennek oka a terület újdonsága, sokfélesége, és a megjelenése óta folyamatos átalakulások. Az Elektronikus Oktatási Környezetek tananyagban elsődleges célunk volt, hogy az irodalom mai állására alapozva egy olyan összefoglalót készítsünk egyetemi, vagy akár középiskolai oktatók számára, ami a legszélesebb körben elfogadott értelmezés szerint mutatja be az e-learning mai jelentését.

A tananyag gyakorlati és elméleti ismereteket egyaránt tartalmaz, és számos videó anyaggal is kiegészítettük. Az átláthatóság kedvéért az anyag 14 fejezetét két szakaszra osztottuk. Az első hat fejezet minden elméleti ismeretet tartalmaz, mely szükséges ahhoz, hogy a hallgatók megválaszolják a kapcsolódó szakok záróvizsgáinak ehhez a témakörhöz köthető tételeit. Az elméleti rész végén ellenőrző kérdésekkel segítjük az ismeretek elsajátítását. A következő nyolc fejezetben a gyakorlati oldalról közelítjük meg a témát. Ennek a résznek az elején elsősorban a Moodle keretrendszer általános bemutatásával illetve a SCORM tananyagok készítésének lehetséges módjaival foglalkozunk. Az utolsó négy

fejezetben konkrét tantárgyak/témakörök esetében szemléltetjük a megvalósítás lehetőségeit. A második szakasz szintén az elsajátított tudást próbára tevő feladatokkal zárul.

2. Az elméleti rész bemutatása

Az elméleti rész anyagát 6 fejezetre bontottuk. Ezek közül az első az e-learning fogalmának meghatározását tűzte ki célul. Ennek megfelelően itt kísérletet teszünk a kapcsolódó fogalmak tisztázására és rendszerezésére. Így ismertetjük a *számítógéppel támogatott oktatás*, a *webalapú oktatás*, a *távoktatás* és az *e-learning* fogalmak közötti határokat és összefüggéseket. (lásd 1. ábra). Az első fejezet végül az e-learning helyzetét vázolja napjainkban.



1. ábra Egy mintául szolgáló ábra.

A második fejezetben bemutatjuk, milyen szerepet tölthet be az e-learning az oktatásban. Ezt a fejezetet két részre osztottuk. Elsőként tisztázzuk a különbséget a tiszta és a kevert e-learning között, majd ismertetjük az e-learning használatának előnyeit és hátrányait.

A harmadik fejezetben bemutatjuk, hogyan is néz ki egy e-learning rendszer. A fejezetben külön hangsúlyt fektetünk a rendszer architektúrájának ismertetésére, Ennek megfelelően az átfogó kép érdekében sorra vesszük a tanulásmenedzsment rendszert (LMS), a tartalomkezelő rendszert (LMCS) és a virtuális tanulási környezetet. Az architektúra ismertetése az egyes egységek kommunikációs modelljének bemutatásával zárul (lásd 2. ábra).



2. ábra Az e-learning architektúra

Szintén a harmadik fejezetben kerül sor a jelenleg elérhető nyílt forráskódú e-learning keretrendszerek működési tulajdonságainak és funkcionális szolgáltatásainak bemutatására.

A negyedik fejezet az e-learninggel kapcsolatos legfontosabb szabványok ismertetését tűzi ki célul. Ezek közül ismertetésre kerül az LO (Learning Object), a LOM (Learning Object Metadata) és a legfontosabb szabványként a SCORM (Shareable Content Object Reference Model). Utóbbi igazából nem is egy szabvány, hanem egy szabvány együttes, mely integrálja a szakterület legfontosabb szabványait. Jelentőségénél fogva a SCORM-mal a későbbi gyakorlati részben is részletesen foglalkozunk. Az aktualitás érdekében igyekeztünk, hogy itt a lehető legújabb technológiákat ismertessük, ezért szót ejtünk a legújabb hivatalos SCORM szabványról, a SCORM 1.3.4-ről, de megemlítjük az xAPI-t is, melyet több helyen a következő generációs SCORM-ként emlegetnek.

Miután a korábbi fejezetekben bepillantást nyerhettünk az e-learning világ elméleti hátterébe és a szabványok jelenlegi állásába, az ötödik fejezetben rövid betekintést adunk a tananyagszerkesztés módszertani alapjaiba. Fontos, hogy az ebben a fejezetben leírtak igazából nem csak e-learning tananyagok készítésénél lehetnek hasznosak, de egyéb számítógéppel támogatott, vagy akár offline anyagok szerkesztése során is. Ez a fejezet is két nagyobb részre osztható. Az első szakaszban igyekszünk konkrét tanácsokkal ellátni a leendő tananyagszerkesztőt, milyen módszerekkel építhet jól használható informatív, de ugyanakkor robosztus anyagot. Hangsúlyozzuk a megfelelő technikák megfelelő helyen történő alkalmazását is. A fejezet második szakasza egy kicsit szárazabban, de pontosabban foglalja össze a tananyagfejlesztés módszertanának kapcsolódó részét. Ez utóbbi szakasz lehet hasznos a diákoknak az elmélet elsajátításához, míg az előtte lévő szakasz szolgál a konkrét szerkesztési folyamat segítésére, némileg előremutatva a gyakorlati fejezetekre.

Az első szakasz utolsó rövid, de az előzőeknél nem kevésbé fontos fejezete, a hatodik fejezet a szerzői rendszerekről szolgál némi ismertetővel. A fejezet egyik legfontosabb jelentősége, hogy kapcsolatot teremt a közvetlenül utána következő gyakorlati résszel. A tananyag elméleti része 40 ellenőrző kérdéssel zárul, melyek segítségével a hallgató visszajelzést kaphat, mi az amit igazán fontos ismerni a korábban olvasottakból.

3. A gyakorlati rész bemutatása

A tananyag gyakorlati részét összesen nyolc fejezet képezi, melyek közül négy az elméleti részben már bevezetett szerzői rendszerekkel, SCORM típusú tananyagok előállításával, illetve a Moodle keretrendszerrel foglalkozik. Ez utóbbi különösen jelentős, hiszen az általános ismertetésen kívül a gyakorlati rész utolsó négy fejezete ebben a keretrendszerben igyekszik példát szolgáltatni e-learning alkalmazásokra különböző tantárgyak esetében.

A hetedik fejezet ennek megfelelően rögtön a Moodle keretrendszer általános ismertetésével indít. Mivel gyakorlati részről van szó itt már sokkal kevésbé leírás jellegű a szöveg. Inkább fogható fel egyfajta gyakorlatsornak is. A tananyaghoz készült videók nagy része is ehhez és az ezután következő fejezetekhez kapcsolódik. A hallgató a fejezetet követve először megismerhet egy viszonylag egyszerű módot egy Moodle tesztkörnyezet kialakítására BitNami segítségével. A tananyagban is megjegyezzük azonban, hogy ez a környezet inkább csak a keretrendszer használatának elsajátításához ajánlott, az éles rendszerek telepítését és üzemeltetését bizzuk a rendszergazdákra. A telepítés után röviden ismertetjük a legfontosabb adminisztrátori funkciókat úgymint szerepkörök, kurzusok, kategóriák, tesztek, bővítmények, appletek és animációk. Vegyük észre, hogy az érintett témakörök szintén nem a portál magas szintű adminisztrációjának bemutatását célozzák meg. Csak azokra a funkciókra fókuszálunk, amire egy kurzus oktatójának szüksége lehet, de ha nem is ő végzi el, jó ha tisztában van a feladatok hátterével és a megvalósításuk lehetőségeivel.

A nyolcadik fejezet egy konkrét szerzői rendszert, az eXe XHTML szerkesztőt ismerteti a SCORM tananyag szerkesztésének szempontjából. Ez a fejezet is jócskán tartalmaz videós illusztrációt a leírtakhoz, hogy megkönnyítse az anyag feldolgozását. Ugyanakkor, ahogy a korábbiakban is, itt is igyekeztünk a leendő oktatók számára igazán fontos dolgokat kiemelni, épp csak említés szintjén hagyva az eXe szerkesztő egyéb lehetőségeit. Tematikáját tekintve ez a fejezet az eXe szerkesztő telepítésével indít, majd a kezelőfelület bemutatása után az egyszerű illetve a matematikai elemek anyagba illesztését ismerteti. A fejezet természetesen az anyag SCORM tananyagként történő mentésével zárul.

A kilencedik fejezet sokban hasonló az előző fejezethez. Itt egy másik kedvelt lehetőséget mutatunk be SCORM tananyagok készítésére, a Reload szerkesztőt. A fejezet tematikáját tekintve szintén hasonló a nyolcadik fejezethez. A telepítés bemutatását a kezelőfelület követi, majd a tananyag készítésére és összeállítására kerül a sor. Ahogy az anyagban is hangsúlyozzuk, a két ismertetett szerkesztő közötti jelentős különbség, hogy míg az eXe leginkább egy egyszerű HTML szerkesztőként használható, melyből aztán szabványos SCORM tananyagot exportálhatunk, a Reload sokkal inkább eleget tesz a SCORM eredeti céljának. Esetében ugyanis nem csak a szabvány minimális kielégítése a cél, de lehetőséget kapunk a szabványok igazi erősségeinek alkalmazására is (noha az anyag megjelenítésébe itt jóval nehezebb beleszólnunk, mint az eXe szerkesztő használatakor).

A tizedik fejezet a gyakorlati rész második szakaszát, a konkrét tantárgyakat felvonultató Moodle esettanulmányokat készíti elő, összefoglalva, hogy általánosságban miért szükséges más és más látásmód a különböző tárgyak esetében, ugyanakkor miért lehet mégis az e-learning a közös nevező. Azaz hogyan valósítható meg az e-learning oktatás a különböző tárgyi sajátosságok ellenére.

A következő fejezetek egymás után veszik az informatika, a matematika, a fizika és az idegen nyelvek tantárgyak sajátosságait és az azokra adott e-learning technológiai válaszokat. Néhány esetben egy-egy jó példát is mutatunk, ami ötletet adhat, hogy a hasonló témaköröket hogyan érdemes feldolgozni.

A sort a tizenegyedik fejezetben az informatikával kezdjük. Ahogy az a tárgykör sajátosságainak leírásánál is megfogalmazzuk: Az informatikai tárgyak anyaga szorosan kötődik az online elérhető információkhoz. Sok probléma megoldásához konkrétan az internet

használata szükséges. Az online már jelenleg is elérhető nagymennyiségű információt több okból sem szabad figyelmen kívül hagyni. Egyrészt a már elkészült anyagok használatával jelentősen csökkenthető a elvégzendő munka a tananyag szerkesztésekor, másrészt pedig a megfelelő forrásokat kiválasztva állandóan naprakész információkkal szolgálhatunk a hallgatóknak. Ezt szem előtt tartva a fejezetet érdemileg az alkalmazható hasznos Moodle bővítmények bemutatásával kezdjük, különös tekintettel a VPL bemutatására mind tanári, mind hallgatói szemmel. A fejezet végén egy konkrét programozás tárgy használatban lévő Moodle oldalt mutatjuk be, mely ugyan nem használja ki a lehetséges eszközrendszer minden lehetőségét, példát ad arra, hogyan lehet egy ilyen rendszert ténylegesen az oktatás szolgálatába állítani az informatika esetében.

A tizenkettedik fejezetben az előzőhöz hasonló tanulmányt adunk a matematika tantárgy esetében. A tantárgyi sajátosságok között itt kiemeljük a matematikai formalizmusok nehézkes kezelését, ugyanakkor megoldásként két hasznos bővítmény is említünk és ismertetünk nem csak szövegesen, de videós formában is. Így ismertetjük a Geogebra szűrőt és a Wiris matematikai eszközrendszert a telepítéstől az alkalmazásig.

A tizenharmadik fejezet a fizika tantárgyra fókuszál. Első megközelítésben természetesen gondolhatnánk, hogy a matematikával igen szoros kapcsolat miatt a tárgy sajátosságai is egyeznek azzal. Ugyanakkor hamar észre kell vennünk, hogy a tárgy a szemléletességét tekintve messze túlmutat a matematikán. Szerencsére akár Moodle bővítményként is számos lehetőségünk van különböző szemléltető szimulációs programok használatára, így téve könnyebbé az anyag elsajátítását. A lehetőségek közül mi a legnagyobb ismert csomagot, az Open Source Physics modult mutatjuk be, illetve a lehetőségek bővítésére itt mutatjuk meg, hogyan illeszthetünk Moodle oldalunkba Java appleteket, vagy Flash animációkat.

A fejezet végén ismét egy jó példát ismertetünk: egy a Debreceni Egyetemen jelenleg is oktatott fizika tárgy Moodle oldalát tekintjük át röviden.

A tizennegyedik fejezet a Moodle használatának lehetőségét mutatja meg az idegennyelv oktatásban. Szerkezetét tekintve ez a fejezet sem különbözik a korábbi háromtól. Elsőként a tárgykör sajátosságait foglaljuk össze röviden, kiemelve az írásbeli nyelvtanulást. Következően egy igen hasznos Moodle bővítményt, az OWL Language Lab-ot mutatjuk be. Erre a bővítményre építve igazából a legtöbb nyelvi feladattal kapcsolatos problémát meg is oldhatjuk. A tizennegyedik fejezet is egy konkrét tárgy Moodle oldalának rövid ismertetésével zárul,

A gyakorlati szakaszt ismét feladatokkal zárjuk, melyek azonban ezúttal nem egyszerűen megválaszolható kérdések, hanem komolyan energia befektetést igénylő összetett feladatok. Ezek elvégzéséhez természetesen a legtöbb esetben nem is csak szöveges, de videós példát is mutatunk korábban az anyagban.

A tananyag a két érdemi tartalmat leíró szakasz után a videójegyzékkel és az irodalomjegyzékkel zárul.

4. A tananyag felhasználásának lehetőségei

A tananyag készítését az *Elektronikus oktatási környezetek* tantárgy elmúlt években szerzett oktatási tapasztalataira alapoztuk. Oktatóként a tárgy gyakorlatainak összeállításakor nagy hangsúlyt fektettünk arra, hogy olyan példákon keresztül mutassuk be az eszközöket melyek által a hallgatók megláthatják a lehetőséget ezek használatában. A gyakorlati órák témáit évről-évre a hallgatói igények és visszacsatolások szerint alakítottuk, így került bele az informatika mellett a matematika, fizika és az idegennyelv oktatása is. A tananyag megírásakor ezeket a tapasztalatokat vettük figyelembe és egy olyan jegyzetet készítését

tűztük ki célul, mely egyaránt alkalmas a tantárgy kiegészítése mellett az ismeretek önálló elsajátítására is.

A hallgatók, akik leendő, vagy már dolgozó középiskolai tanárok az első pillanattól lehetőséget, munkájuk megkönnyítésének eszközét látták az e-learning rendszerekben, emiatt gyakran kaptunk pozitív visszacsatolást a tárgy hasznosságáról. Saját sikerként könyveljük el, hogy az elmúlt években oktatott diákjaink közül munkahelyükön – egy-egy általános, vagy középiskolában – sokan bevezettek valamilyen e-learning rendszert.

Bár jelenleg a középiskolai oktatásban nem támogatott a tiszta e-learning használata, mégis remek kiegészítései lehetnek ezek a rendszerek a hagyományos tantermi oktatásnak és számonkérésnek. A felsőoktatásban is érezzük annak előnyeit, hogy a hallgatók nem az egyetemen találkoznak először az elektronikus oktatási rendszerekkel. Hallgatóink gyakran már úgy érkeznek, hogy ismernek és használtak már hasonló rendszert. Az elektronikus tanulás vitathatatlanul segít az önálló tanulás kultúrájának kialakításában, melyre a felsőoktatás az első pillanattól támaszkodik.

5. Összefoglalás

A tananyag fejlesztési projekt keretében az Elektronikus Oktatási Környezetek és a hozzá kapcsolódó egyetemi tantárgyakhoz fejlesztettünk elektronikus tananyagot. Jelen leírásban röviden ismertettük az anyag felépítését a felhasználásának lehetőségeit. Eszerint az anyag két nagyobb részből áll: az első szakasz a tárgykör elméleti hátterét dolgozza fel, lehetőséget adva a hallgatóknak, hogy szert tegyenek a kapcsolódó tárgyakon elvárt elméleti ismeretekre. A második szakasz ezzel szemben elsődlegesen gyakorlati ismertetőkből áll. Bemutatunk két szerkesztő rendszert és a Moodle környezetet. A második szakasz utolsó négy fejezetében konkrét példákat mutatunk a felhasználásra. Jelen cikk végén megmutattuk, hogy milyen módon és milyen tárgyakhoz tartjuk jól illeszhetőnek az elkészült anyagot.

A tananyag önmagában szándékunk szerint teljes egészet alkot. Ugyanakkor a technológiák folyamatos fejlődése miatt valószínűleg 2-3 évente szükséges lenne a leírtak aktualizálása, újragondolása az újonnan megjelenő eszközök tükrében. Mivel az egyetemen egyre növekvő számban képzünk idegen nyelvű hallgatókat is, szükségesnek tartanánk a jövőben az anyag angol verziójának elkészítését is.

Irodalomjegyzék

- [1] Fazekas Gábor, Balla Tibor, Kocis Gergely, Elektronikus Oktatási Környezetek tananyag, Gyires Béla Informatikai Tananyag Tárház, Debrecen, (2014).

BSc és MSc szintű agrárinformatikai képzési programok a Debreceni Egyetemen

BSc and MSc level agricultural informatics training programs in the University of Debrecen

Herdon Miklós^a, Petó Károly^b, Várallyai László^c, Szilágyi Róbert^d

^{abcd} Debreceni Egyetem/Gazdaságtudományi Kar

herdon@agr.unideb.hu

peto@agr.unideb.hu

varal@agr.unideb.hu

szilagyi@agr.unideb.hu

Absztrakt: A Magyar Felsőoktatásban a 2003/2004-es tanévben jelent meg az agrárinformatika szakterületre irányuló képzés egyetemi 5 éves alapképzési szakként. Három évfolyam kezdte meg tanulmányait a Debreceni Egyetem 5 éves informatikus agrármérnök szakán. A bolognai folyamatban a képzési tapasztalatainkra alapozva társintézményekkel készítettük el az Informatikus és szakigazgatási agrármérnök BSc szak alapítási és indítási kérelmet, amelyet a 2006/2007-es tanévben vezetünk be. A mesterképzési szak létesítésére, vizsgálva a képzés iránti érdeklődést, a szakemberigényt, az informatika agrárgazdasági alkalmazásának trendjeit csak később került sor. A létesítési és indítási kérelem együttes benyújtása 2013 tavaszán történt a MAB részére. Kisebbségi korrekciókkal a MAB véleménye alapján az Emberi Erőforrások Minisztériuma és az Oktatási Hivatal jóváhagyta a szakigazgatás-szervező és informatikus agrármérnöki MSc szak létesítését és indítását. Mind a BSc mind az MSc szintű képzés programjában az alapozó, a szakmai törzsanyag és differenciált szakmai ismeretkörökben meghatározó az informatikai-, agrárszakmai-, gazdasági- és szakigazgatási tantárgyak, ismeretek oktatása. A cikk a képzési programok létrehozásának előzményeit, a fontosabb mérföldköveket, a képzések tartalmát, a tapasztalatokat és terveket ismerteti.

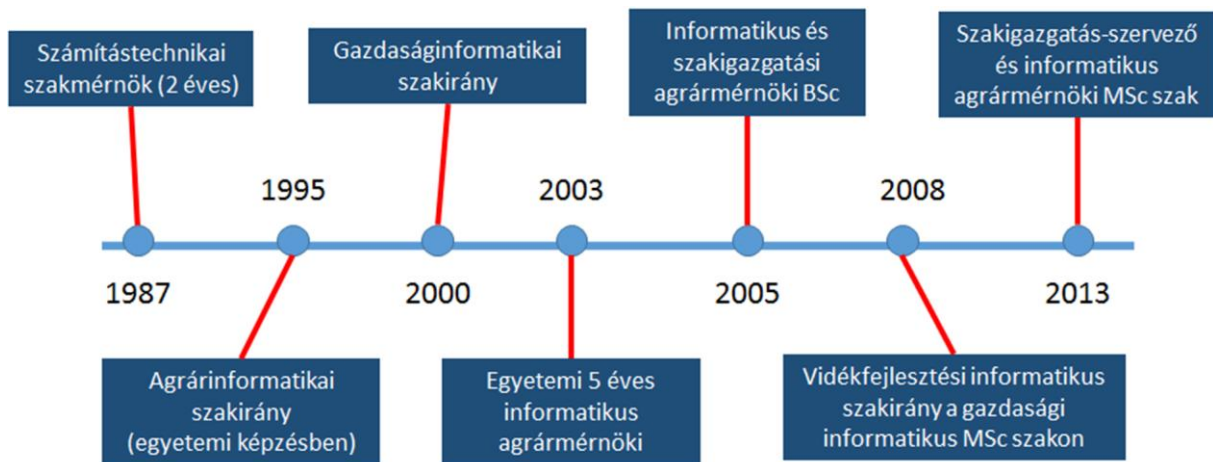
Kulcsszavak: agrárinformatika, BSc, MSc, tantervek

Abstract: In Hungarian higher education in the 2003-2004 school year, introduced the training for information technology professionals in the agricultural area as a five-year bachelor's degree program. Three grades began his studies at the University of Debrecen in the 5-year-old information technology in agricultural engineering training. Based on the experience we have prepared with the cooperation of our partner institutions the BSc course formation and start-up application and technical management in Agricultural Information Technology and Policy Administration BSc course, which we introduced in the 2006-2007 school year. Looking at the Master's degree courses have an interest in education, the need for professionals in the IT application of agro-economic trends came later. Submit the application installation and start-up in the spring of 2013 for the MAB. Minor corrections are based on the HAC and the opinion of the Ministry of Education, Office of Human Resources has approved the technical and IT management, organizing agricultural engineering MSc course creation and launching. Both the BSc and MSc level training program of the basic knowledge, the professional core and differentiated professional circles defining the IT knowledge, agricultural trade, economic and technical management courses, teaching skills. The paper describe the history of the establishment of training programs, major milestones, the content of the training, experience and describes the future plans.

Keywords: agricultural informatics, BSc, MSc curricula

1. Bevezetés

A Magyar Felsőoktatási törvény úgy rendelkezett, hogy a korábbi kettős (főiskolai / egyetemi) képzési rendszer helyett egy új, lineáris felsőoktatási képzési rendszert alakítson ki, valamint ezeknek a feltételeknek megfelelő új képzési programok induljanak. Magyarország - kivéve a művészeti képzést - áttért a többciklusú képzésre. Az első kísérleti képzések 2004-ben és 2005-ben indultak [4], a BSc képzések nagyobb intenzitással 2006-ban jelentek meg. Azokon a területeken, ahol a kísérleti képzések korábban már elindultak (informatikai, műszaki), az első végzett hallgatók 2008-ban; azonban nagyobb számban 2009-ben fejezték be tanulmányaikat. A képzési programok keretei a különböző programok által biztosított képzési és kimeneti követelményekből állnak, amelyeket a rendszeresen kiadott miniszteri rendelet tartalmaz. 2009-től a mester szintű képzési rendszer kialakítása kezdődött, hiszen a felmenő rendszerben a Bachelor szintű képzési rendszer hallgatói tömegesen ekkor végeztek. Ennek érdekében a mester szintű programok előkészítése már az első képzési ciklusban elkezdődött. Ennek kialakításában a korábbi egyetemi és főiskolai szintű képzési programok kevésbé jelentős szerepet játszottak [5]. Ugyanakkor a felsőoktatási intézmények általában arra törekedtek, hogy a második képzési ciklusú programot építsenek az első ciklusú képzésekre. Szakmai egyetemi műhelyek át is vették kezdeményezést, hogy a bennük rejlő tudományos háttér és a személyi feltételek alapján készítsenek speciális képzési programokat, amelyek jellemzően szűkebb területet öleltek fel [1]. Az agrárinformatikai képzésekre visszatekintve a 80-as években több mint 30 évvel ezelőtt kezdődött el a számítástechnikai tárgyak tanítása. Néhány tantárgy bevezetése után hoztuk létre specializáció tantervet alkalmazás orientált témákban [3]. Ez idő alatt több tantervet és akkreditált oktatási programot fejlesztettünk. A legfontosabb mérföldköveket az 1. ábra mutatja:



1. ábra. Az agrárinformatikai képzések fejlesztésének mérföldkövei

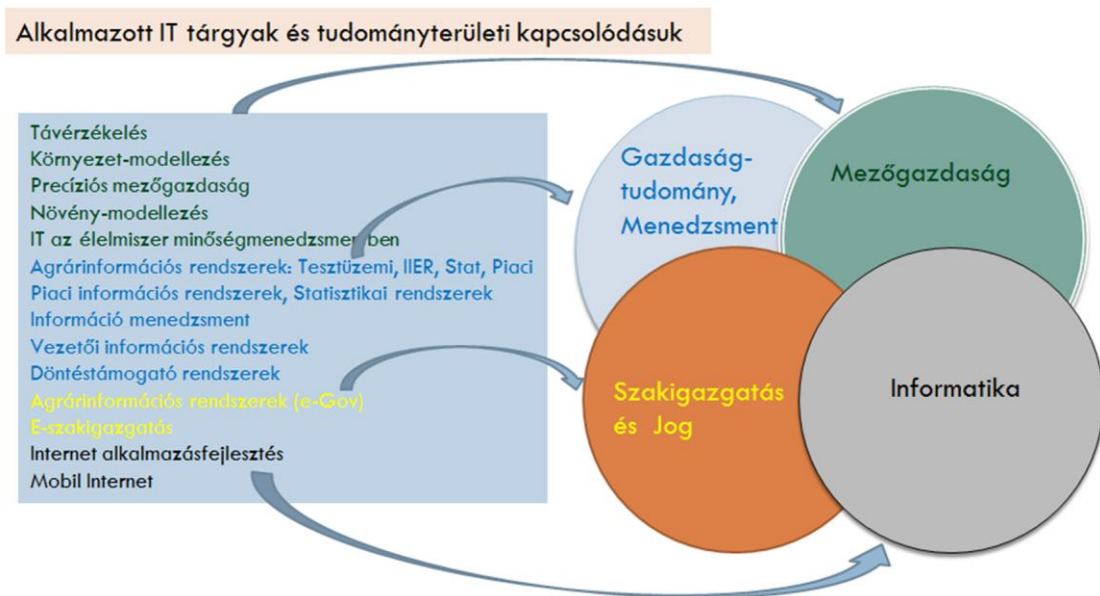
2. A fejlesztési folyamat

Magyarországon három egyetem, a Debreceni Egyetem, a Budapesti Corvinus Egyetem és a Szent István Egyetem kifejlesztett egy BSc tantervet az Informatikus és szakigazgatási agrármérnök BSc képzésre [4]. A képzés során a hallgatók 180 kreditet szerezhhetnek. A kreditek arányai a következők: a. Általános ismeretek 20%, b. Agronómiai ismeretek 16%, c. Szakigazgatás 18%, d. Gazdasági ismeretek és kapcsolódó témák 13%, e. Informatikai és speciális informatikai tudás 33%.

A fő tantárgycsoportokhoz tartozó kreditek a következők:

a. Általános ismeretek:	28 kredit
b. Agronómia:	21 kredit
c. Szakigazgatás:	18 kredit
d. Gazdasági ismeretek és kapcsolódó témák:	24 kredit
e. Informatikai és speciális informatikai tudás:	36 kredit
f. Egyék szaktantárgyak:	29 kredit

A képzéshez két szakirány a közigazgatás és az informatika tartozik. Az informatikai specializáció során a következő témákban tanulhatnak: Agrárinformációs rendszerek (tesztüzemi rendszer, integrált irányítási és ellenőrző rendszer, piaci információs rendszerek, agrárstatisztikai rendszer), Internetes alkalmazás-fejlesztés, Információ menedzsment, Menedzsment és szervezés, IT az élelmiszer-minőségbiztosításban, Vezetői információs rendszerek, Szakértői rendszerek, Projekt menedzsment, Távérzékelés, Ágazatspecifikus megoldások. Az alkalmazott informatikai tárgyak különböző diszciplínákhoz kapcsolódnak. Néhány fontosabb tárgy kapcsolatát mutatja a 2. ábra.



2. ábra. Az alkalmazott IT tárgyak szakterületi kapcsolatai

Jelenleg az Informatikus és szakigazgatási agrármérnök BSc a Debreceni Egyetem Gazdálkodástudományi Karán, valamint a Budapesti Corvinus Egyetem Kertészettudományi Karán folyik. Korábban a Pannon Egyetemen, Szent István Egyetemen és a Szegedi Tudományegyetemen is indult ilyen képzés. A képzés kialakítása során a lehetséges munkahelyek igényeit is figyelembe vettük. Véleményünk szerint a potenciális munkahelyek a következők lehetnek: kormányhivatalok (minisztérium, a mezőgazdasági szakigazgatás), az élelmiszeripar (főként a vállalati információs rendszer szakértő), a hagyományos mezőgazdaság (termelés, tevékenység controller) és az oktatás (mezőgazdasági szakoktatás, agrárinformatika).

3. Az új agrár-informatikai MSc szak

3.1 Az oktatási program tervezése

Az informatikus és szakigazgatási agrármérnök alapszakra épülő tervezett mesterképzésként az „informatikus agrármérnök” szak létesítése lett volna célszerű. Az egyetemi szintű 5 éves képzés előzménye az egyetemi szakokon közel egy évtizede működő gazdasági- és agrárinformatikai szakirányok voltak. A tervezett informatikus agrármérnök MSc szak személyi, tárgyi feltételei adottak voltak, melyek az egyetemi alapképzési szak keretében kialakításra kerülnek. Az informatikus agrármérnök egyetemi alapképzési szak a 2003/2004-es tanévben került bevezetésre, az első egyetemi évfolyam 2007/2008-es tanévben végzett. Tapasztalataink alapján az igények oldaláról szükséges, a képzési feltételek oldaláról pedig biztosított a tervezett MSc szak indítása, mely képzés főbb tanulmányi területei a következők:

a.) Természettudományi ismeretek

A matematika és informatika speciális fejezetei (ezen belül kiemelten a matematikai módszerek alkalmazásai, operációkutatás, matematikai statisztika, információ- és rendszerelmélet) és az ezekhez szükséges további természettudományi alapismeretek.

b.) Gazdasági- és agrár-szakismeretek

A szak sajátosságai által megszabott tartalommal és mértékben mezőgazdasági szakterületi ismeretek (környezet- és vízgazdálkodás, gyepgazdálkodás, növényélettan, állatélettan, növénytermesztés és állattenyésztés), illetve agrárgazdasági ismeretek (vállalati gazdaságtan, szervezési és vezetési ismeretek, számvitel, logisztika, marketing).

c.) Alkalmazott informatikai ismeretek

A szak területén belül biztosítja azt a konvertálható informatikai ismeretet, amely az agrárgazdaság legkülönbözőbb alkalmazási területein hasznosítható: adatbázisrendszerek, alkalmazás-fejlesztés, infrastruktúra menedzsment, integrált információs rendszerek, multimédia, precíziós mezőgazdaság.

d.) Differenciált informatikai ismeretek

Mélyreható, részletes szakmai ismeretek nyújtását biztosítják, kutatás-fejlesztési munkához szükséges készségek kialakítását a szakhoz kapcsolódó, az informatikai alkalmazások különböző területein teszik lehetővé: ágazati irányítás információs rendszerei, e-kormányzat, e-business, számítógéppel támogatott tervezés, modellezés, szakterületi alkalmazások (talajinformációs rendszerek, állategészségügyi információs rendszerek, élelmiszer-biztonság informatikai támogatása), stb.

3.2 A képzési követelmények

A képzés igénye, hogy a vidék és az agrárgazdaság fejlődésének, az innovatív képesség növelésének, a hatékonyság növelésének igénye a gazdálkodási, vállalkozási, a társulási, térségi, szakigazgatási szinten és a teljes ágazat irányítási szintjén megnyilvánul. A különböző ágazati problémák, mint az élelmiszertermékek minősége, a termékek nyomon követése, a logisztikai rendszerek üzemeltetése, a vidéki térségek fejlesztése, az elektronikus üzletviteli rendszerek elterjesztése, a szükséges szolgáltatások, a biológiai és szakmai adat- és tudásbázisok fejlesztése, hasznosítása nem nélkülözheti a megfelelő informatikai, gazdasági és agrárszakismerettel rendelkező szakemberek meglétét, akik képesek az informatikai rendszerek fejlesztésében és üzemeltetésében való alkotó, kreatív részvételre. Az e-

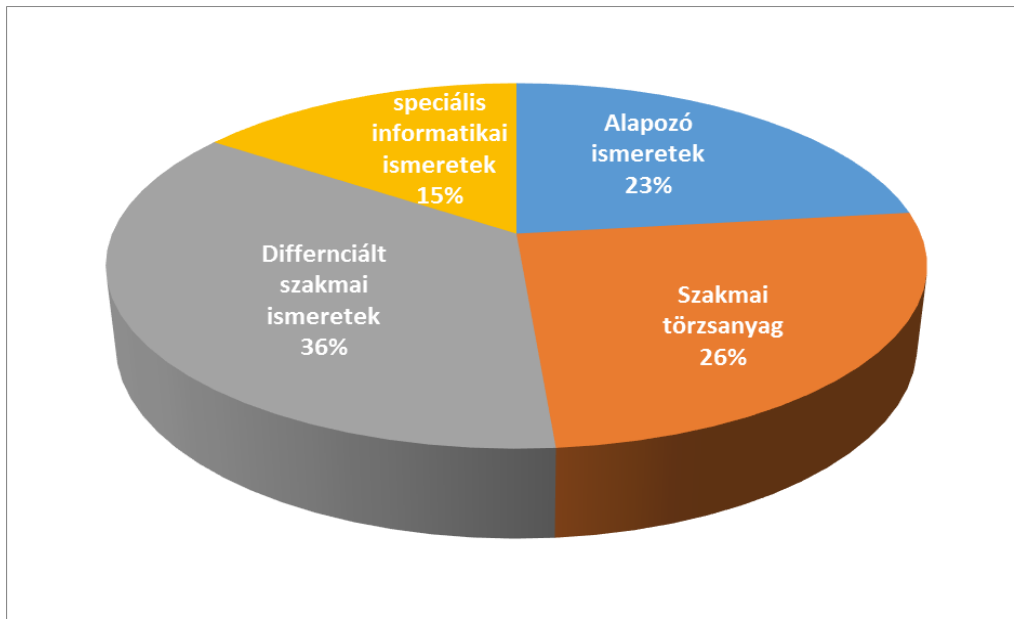
szakigazgatás, az elektronikus üzletvitel, a mezőgazdasági szaktanácsadás, a környezetvédelem területein, valamint az agrárágazat irányítási és információs rendszereiben specifikus információtechnológiák jelentek meg és kerülnek egyre inkább gyakorlati felhasználásra. A megjelenő új feladatok és rendszerek ismerete és az ezen ismeretek hasznosítása nélkülözhetetlen a vidék térségek, az agrárgazdaság és az agrárgazdasággal szoros kapcsolatban lévő szektorok számára.

Az EU és tagállamainak 2007-2013-ra vonatkozó fejlesztési stratégiájában a Vidékfejlesztési Stratégiai Terveiben az informatika a korábbi évekhez képest egyre fontosabb szerepet kapott [2]. A szakon végzett hallgatók iránti igényeket jelentősen növelte a Mezőgazdasági Szakigazgatási Hivatal országos hálózata, a Vidékfejlesztési Hivatalok országos hálózatának folyamatban lévő kialakítása is. Az Európai Unió tagállamaiban elkészültek és kötelezően működnek a nagy információs rendszerek, mint a Piaci Árinformációs rendszer, az Integrált Irányítási és Ellenőrzési Rendszer, Mezőgazdasági Számlák rendszere, az Agrárstatisztikai Rendszer, stb. Ezen rendszerek működtetése mellett számos szakigazgatási területen működnek az országos hálózati rendszerek. Ilyenek az erdészeti, a növény-egészségügyi, állategészségügyi rendszerek, az ingatlan nyilvántartás stb. E rendszerek működtetése, továbbfejlesztése, új rendszerek létrehozása a Vidékfejlesztési Minisztérium és háttérintézményinek véleménye szerint is egyre több olyan szakembert igényel, akik a magas szintű informatikai ismeretek mellett rendelkeznek vidékfejlesztési és gazdasági ismeretekkel.

3.3 A tanterv

Tapasztalataink és a hallgatói igények arra sarkalltak bennünket, hogy egy új MSc szak tantervét kidolgozzuk, ahol a végzős Informatikus és Szakigazgatási Agrármérnök BSc hallgatóink közvetlenül folytatni tudják tanulmányaikat. Ez a munka 2013-ban kezdődött a Gazdaság- és Agrárinformatikai Tanszéken. A tananyag három fő részből áll: az alapozó, a szakmai törzsanyag és differenciált szakmai ismeretekből.

A cél az MSc szintű képzésben, hogy olyan szakembereket képezzünk, akik képesek informatikai rendszereket fejleszteni és irányítani a mezőgazdasághoz kapcsolódó területeken, képesek megérteni a valós termelési, működési és üzleti modelleket, képesek azok megalkotásában való közreműködésre, továbbfejlesztésre és azok működtetésére. Ezeknek a problémáknak a megoldására informatikai eszközöket tudjon használni a gyorsan fejlődő és változó informatikai és telekommunikációs eszközök felhasználásával. Fontos kompetencia a modellezési képesség, a modell kialakítása, illetve a megfelelő megoldó algoritmusok megtalálása az adott feladathoz. Fontos célként fogalmaztuk meg, hogy a frissen végzett szakemberek megértsék a mezőgazdasági, gazdasági és adminisztrációs folyamatokat, melyeket informatikai rendszerek támogatnak a különböző vállalati területeken. Az MSc szak 2 éves, amelyben a szükséges tanulási idő 1500 óra, a szükséges kredit 120. A tárgycsoportok megoszlása az 1. ábrán látható.



1.ábra. Tárgycsoportok a Szakigazgatás-szervező és informatikus agrármérnöki Mester szakon

a. Az alapozó ismeretek tantárgyai (26 kredit) a következők: Szoftverfejlesztési technológiák, Fenntartható mezőgazdaság (Növénytermesztés), Gazdasági jog, Agroökológiai rendszerek modellezése, Agrár-környezetvédelem, Vállalatgazdaságtan, Számvitel vezetőknek, Kutatásmódszertan, Fenntartható mezőgazdaság (Állattenyésztés).

b. A szakmai törzsanyag tantárgyai (37 kredit): Ezek a tantárgyak az alapvető mezőgazdasági közgazdaságtan és az informatikai ismereteket megszerzését szolgálják, melyek a következők: Agrárspecifikus alkalmazási szoftver rendszerek fejlesztése, Infokommunikációs hálózati rendszerek, Adatbázisrendszerek fejlesztése, Az agrár-szakigazgatás irányítási-szervezési rendszere, Vállalatirányítási információs rendszerek, Környezetinformatika – Távérzékelés, Agrárgazdaságtan és agrárpolitika, Agrár- és környezetstatisztika, Agrárinformációs rendszerek, Döntéstámogatási módszerek és alkalmazások.

c. A differenciált szakmai ismereteken belül a hallgatók az alábbi tantárgyakkal ismerkedhetnek meg (31 kredit): Adatbányászat, e-Agrárszakigazgatás, e-Kereskedelem, Biometria, Környezetgazdasági modellezés, Precíziós mezőgazdaság, Szakértői rendszerek az agrárgazdaságban, Lokális vidékfejlesztés és szaktanácsadás, Környezet- és természetvédelmi politika, Vezetés- és szervezetelmélet, Mezőgazdasági ágazatok gazdaságtana, Minőségmenedzsment, Ökológiai földhasználat, Élelmiszerlánc menedzsment, logisztika.

d. A fentiekén kívül az alábbi szabadon választható tantárgyakból választhatnak a hallgatók minimum 6 kredit értékben. A hallgatók konvertálható informatikai ismereteket kapnak, amelyeket fel lehet használni a különböző mezőgazdasági és a vidékfejlesztési területeken. Ezek a tárgyak a következők: Mobil internet alkalmazások az agrárgazdaságban, e-Learning technológiák és tudástranszfer, Nyomonkövethetőség az élelmiszerláncban, Információs rendszer-fejlesztési módszertanok, Internet alkalmazás-fejlesztés, IT kontrollig és audit.

A szakmai képzés befejezése után a hallgatók különböző területeken helyezkedhetnek el (mezőgazdasági államigazgatási szervezetek, helyi hatóságok, a kutatási és fejlesztési intézmények, gazdálkodó szervezetek (mezőgazdasági és élelmiszeripari, informatikai cégek). Nagyon fontos hangsúlyoznunk azt, hogy ez a képzés egyedülálló a magyar felsőoktatási rendszerben. Reményeink szerint a képzést ez év szeptemberében tudjuk indítani és a

későbbiekben más egyetemek is fogják indítani a képzést (Corvinus Egyetem, Szent István Egyetem, akikkel együtt alapítottuk ezt az MSc szakot).

3.4 A képesítési és kimeneti követelmények

A képesítési és kimeneti követelményekben megfogalmazott kompetenciák és ismereteket az alábbiakban fogalmazzuk meg. A végzett hallgatókkal szembeni elvárásnál különös fontosak a különböző ágazati kérdések, mint például a különböző jellegű információs rendszerek, az agrár-környezetvédelem fejlesztése, a szükséges szolgáltatások, a mezőgazdasági és a műszaki adatok és a tudás fejlesztése sem nélkülözheti a megfelelő információs technológiákat, a szakemberek gazdasági és a mezőgazdasági ismereteinek meglétét, akik képesek az információs rendszerek fejlesztésére és üzemeltetésére, valamint a kreatív alkotói részvételre ezek fejlesztésében és alkalmazásában. Az e-kormányzatban, e-kereskedelemben, a környezetvédelem területén, valamint a mezőgazdaságban az információs rendszerek-specifikus technológiáit egyre gyakrabban alkalmazzák a gyakorlatban. Az új funkciók és rendszerek ismerete és felhasználása elengedhetetlen a mezőgazdasági területeken és a hozzá szorosan kapcsolódó ágazatokban. Ez szolgáltathat indokot a Szakigazgatás-szervező és Informatikus Agrármérnök MSc szak létrehozására és indítására. Ennek az MSc kurzusnak a célja, hogy olyan agrár-szakembereket képezzen, akik képesek a tudásalapú ágazati fejlesztésekre, a kapcsolódó informatikai rendszerek segítségével értékteremtő folyamatokat kialakítására, irányítására és működtetésére. Az ilyen típusú szakembereknek ismernie kell az alapvető informatikai rendszereket és ezek segítségével támogatni a mezőgazdasági, gazdasági, közigazgatási rendszer folyamatait. Együtt kell tudni működni a szakértőkkel az egyes területeken, a szervezetek stratégiai és operatív feladatainak megoldásában.

4. A tapasztalatok értékelése

A közel 3 évtizedes múltra visszatekintő oktatás-fejlesztési munka mérföldkövei jelzik azokat az eredményeket, amelyek egyfajta szakinformatikai terület fejlődését, fejlesztését szolgálták. Az 1970-es és 80-as években nemzetközi és hazai szinten is a számítástechnika alkalmazásának és a kutatási-fejlesztési tevékenységeknek a reneszánszát éltük. Számos, ma már olyan piacvezető nemzetközi informatikai cég kezdte tevékenységét mezőgazdasági szoftverek fejlesztésével, mint az ESRI, SAS, stb. Már az 1970-es években az USA kongresszusi tanulmánya deklarálta a vidéki térségek szélessávú hálózatfejlesztésének szükségességét. Az egyetemi szintű informatikus agrármérnöki szak 10 évvel ezelőtti indítása sikeres volt, az első évfolyamra nagyon jó középiskolai felkészültséggel érkeztek a hallgatók és legtöbbször kiváló eredménnyel végzett. Ebben a képzésben 3 évfolyam végzett, ezt váltotta fel az informatikus és szakigazgatási agrármérnöki BSc szak. Az erre épülő MSc szak tervezete már 2007-ben elkészült, de különböző szakmai viták miatt a létesítési kérelem nem került benyújtásra a MAB részére. Emiatt a végzett hallgatók, akik mester szakon kívánták folytatni tanulmányaikat a legkülönbözőbb MSc szakokra jelentkeztek, ahol sikeresen megállták helyüket a jelentős többletteljesítési követelmény ellenére. Számunkra ez is mutatta, hogy a BSc szak nehéz, amely több tudományterület ismeretanyagára épül, de ezek sikeres elsajátításával több területen helyt tudnak állni a végzős hallgatók. Végül 4-5 év késéssel más felsőoktatási intézmények felvetésére, végzett és jelenlegi hallgatóink, valamint az agrár-ágazati igények hatására elkészítettük az MSc szak létesítési és indítási kérelmét, amelynek képzési és kimeneti követelménye és a szak elnevezése a MAB bírálatok alapján, kompromisszumokkal alakult ki. Reméljük, hogy a napjainkban ismét népszerűvé váló

agrárinformatika a szakterület fejlődéséhez a szakigazgató-szervező és informatikus agrármérnök MSc szak jelentős mértékben hozzájárul.

Irodalomjegyzék

- [1] Csirik, J.: Bologna Process, National Reports 2007-2009, Budapest (2008).
- [2] Csótó, M. és Herdon, M.: Information technology in rural Hungary: plans and reality. Rural Futures: Dreams, Dilemmas and Dangers. The University of Plymouth, United Kingdom, 1-4 April 2008. (2008), 1-6.
- [3] Herdon, M.: Agri-informatics Curriculum and Education. Why and how we need training agri-informatics experts ? *Demeter Conference. European Higher Education Conference on Virtual Mobility - Information and Communication Technologies in Agriculture and Related Sciences (Video conferencing)*, Gent-Copenhagen-Montpellier, June 16-17,1997. Lecture. Real Audio on the DEMETER Proceedings, (1997), 61-69.
- [4] Herdon, M.: Experiences in Curriculum Development for Agricultural Informatics Specialists at BSc and MSc level – Towards European Masters. *EFITAWCCA2005 Conference*. Vila Real Portugal. Proceedings. Published by Universidade de Tras-os-Montes e Alto Douro. Edited by J. Bouventura Cunha and Antonio Valente.Vila Real Portugal, July 25-28, (2005), 733-738.
- [5] Herdon, M., Magó, Zs. és Kormos, J.: Curriculum for Agricultural Engineers and Economists Specializing in Informatics Science. Is this a Good Way to Train Agri informatics experts? *EFITA2003 Conference Proceedings*. Debrecen, Hungary, 2003 July. (2003), 515-519.

Mérnökinformatikus mesterképzés a Debreceni Egyetemen

Engineering Information Technology MSc at University of Debrecen

Kuki Attila^a

^aDebreceni Egyetem, Informatikai Kar

kuki@inf.unideb.hu

Absztrakt: Bemutatjuk a Debreceni Egyetem Informatikai Karán sikeresen akkreditált és 2014. februárjában beindított mérnökinformatikus mesterképzés történetét. Kitérünk a korábbi nem elfogadott pályázatok értékelésére, valamint arra, milyen megoldások születtek a megfogalmazott észrevételek kezelésére. Ismertetjük a közbeeső átmeneti megoldást, ahol a végzett mérnökinformatikus BSc-s hallgatóknak próbáltunk lehetőséget biztosítani a programtervező informatikus mesterképzésen belül létrehozott szakirány segítségével. Végezetül bemutatjuk a most beindult szakot az eddigi tapasztalatok tükrében, kitérve arra, miként próbáltuk az oktatást optimalizálni a már működő (gazdaságinformatikus és programtervező informatikus) mesterképzésekkel

Kulcsszavak: mesterképzés, mérnökinformatikus, tantervek

Abstract: This paper deals with the history of the Engineering Information Technology MSc course. After the successful accreditation it was launched in February, 2014. Before the accepted accreditation there were two unsuccessful attempts. The main statements of the reviews of the unsuccessful applications are described here. It is shown, what kind of solutions were applied towards a successful accreditation. There was an intermediate construction for the Engineering Information Technology BSc students to be able to continue their studies at MSc level: a new specialization was formed inside the Software Design MSc. Finally, the accepted new course is described based on the short educational experiences, and specially focused on connecting points with the other two MSc course at Faculty of Informatics, University of Debrecen.

Keywords: master course, engineering information technology, syllabus

1. Bevezetés

Az európai felsőoktatás harmonizációját szolgáló bolognai folyamat során a Debreceni Egyetem Informatikai Kara az elsők között szervezte át az informatikai alapképzését. Itt történt meg a Programtervező informatikus (PTI) BSc szak alapítása 2004-ben, 2005-ben elindult a Mérnökinformatikus (MI) BSc, majd 2006-ban a már a harmadik alapinformatikai szakkal, a Gazdaság informatikus (GI) BSc-vel is rendelkezünk.

A PTI BSc és a GI BSc természetes folytatására a megfelelő mesterképzési szakok akkreditációi sikeresek voltak, így azok elindulhattak 2007-ben, illetve 2009-ben. Már csak a mérnökinformatikus mesterképzés hiányzott az Informatikai Kar alapinformatikai képzésének palettájáról. Az alábbiakban azt a folyamatot mutatjuk be, amely során végül is el tudtuk érni, hogy ez a képzés is szerepeljen oktatási rendszerünkben.

2. Az első pályázat

A mérnökinformatikus mesterképzés Debreceni Egyetemre történő akkreditációjának első kísérletére 2007-ben került sor. Az Informatikai Kar megfelelő alapképzésén a következő specializációk voltak ekkor:

- Infokommunikációs hálózatok (IKH),
- Mérés- és folyamatirányítás (MFI),
- Vállalati információs rendszerek (VIR).

A Kar másik MSc szakjának tapasztalatai alapján egyértelműnek látszott, hogy sok szakirányt nem érdemes tervezni a mérnökinformatikus MSc szak beadványában. Az egyeztetések eredménye az lett, hogy a szakot egy szakiránnyal kívánjuk indítani. Mivel a három szakirányból az IKH volt a Kar felügyelete alatt, természetesnek látszott, hogy az MSc szaknál ennek a szakiránynak a haladó tárgyait szerepeltessük. A szak tervezett tantárgylistája így nézett ki:

Kód	Tárgynév	Előfelt.
MIM01	Automaták és formális nyelvek	
MIM02	Számítástudomány	
MIM03	Információelmélet	
MIM04	Kódelmélet	MIM03
MIM05	Modellezés és szimuláció	
MIM06	Adatbiztonság	
MIM07	Számítógépes képfeldolgozás és alakfelismerés	
MIM08	Hálózatok teljesítményelemzése	MIM05
MIM09	Adatbiztonság labor	MIM06
MIM18	Kommunikációs eszközök és technológiák	
Szakirány - Infokommunikációs hálózatok		
MIM10	IP alapú kommunikációs hálózatok és rendszerek	
MIM11	Mobil infokommunikáció	MIM10
MIM12	Kapcsolás és útválasztás	MIM11
MIM13	Integrált hálózati technológiák laboratórium	MIM10
MIM14	Útválasztási labor	MIM10
MIM15	Önálló labor tárgy 1	MIM10
MIM16	Önálló labor tárgy 2	MIM15
	Szabályozottan választható	
	Szabadon választható	
MIM17	Távközlő hálózatok	
	Szabadon választható	
MIM21	Új irányzatok az infokommunikációs hálózatok modellezésében	MIM05
MIM22	Gazdasági és humán ismeretek 1	

MIM23	Gazdasági és humán ismeretek 2	
MIM24	Diplomatervezés	

A tantárgyak összeállításánál követtük az Informatikai Kar hagyományait, az erős, matematikai alapú alapoó tárgyak (Számítástudomány, Automaták ...) mellett hangsúlyosan megjelennek a teljesítményelemzésre, szimulációra vonatkozó kurzusok is. A szakirány pedig továbbvitte azt a vonalat, amely a BSc szinten is vonzóvá és sikeressé tette az ottani megfelelő szakirányt.

A beadvány sikertelen volt, az okai alapvetően az intézményi hagyományok és a mérnök végzettségű kollégák hiánya volt.

3. A második pályázat

A megkezdett munka lendülete nem tört meg. A Kar a következő beadvány elkészítésekor nyitott más karok felé, és tantárgyfelelősnek meg tudunk nyerni számos mérnök végzettségű kollégát. A tantervi háló, mint lentebb látható, jelentősen nem változott. Nevesítettük a Humán, gazdasági blokk tárgyat. Ide is kifejezetten a szakmának megfelelő kurzusokat választottunk, amelyek tárgyfelelősei tovább erősíthették a kérelmet. A tantárgylista ekkor így nézett ki:

Kód	Tárgynév	Előfelt.
MIM01	Automaták és formális nyelvek	
MIM02	Számítástudomány	
MIM03	Információelmélet	
MIM04	Kódelmélet	MIM03
MIM05	Hálózatok teljesítményelemzése	
MIM06	Adatbiztonság	
MIM07	Számítógépes képfeldolgozás és alakfelismerés	
MIM08	Térinformatika/Távérzékelés	
MIM09	Adatbiztonság labor	MIM06
MIM18	Kommunikációs eszközök és technológiák	
Szakirány - Infokommunikációs hálózatok		
MIM10	IP alapú kommunikációs hálózatok és rendszerek	
MIM11	Mobil infokommunikáció	MIM10
MIM12	Kapcsolás és útválasztás	MIM11
MIM13	Integrált hálózati technológiák laboratórium	MIM10
MIM14	Útválasztási labor	MIM10
MIM15	Önálló labor tárgy 1	MIM10
MIM16	Önálló labor tárgy 2	MIM15
	Szabályozottan választható	
Gazdasági és humán ismeretek		
MIM22	Termelés és folyamat	

	menedzsment	
MIM25	Controlling	
	Szabadon választható	
	Szabadon választható	
MIM27	Diplomatervezés	

Ez a kérelem is elutasításra került. Az indoklásban ismét szerepelt a mérnök végzettségű kollégák alacsony száma, illetve a tantervi háló nem fedi le a KKK-ban szerepeltetett összes szükséges témakört. Noha a tantervi háló lényegesen nem módosult, az első beadványnál ez utóbbi nem merült fel problémaként. Valamint a háló kialakításakor tanulmányoztuk más intézmények már akkreditált, weben megtekinthető tantervét is. Az első probléma orvoslása azonban sürgősebbnek tűnt, mivel ez egy hosszabb folyamat eredményeképpen történhet csak meg. Ezért a Kar úgy döntött, hogy egy időre felfüggeszti a szak akkreditálására irányuló törekvését, és más, áthidaló megoldásban gondolkodik.

4. Hardverprogramozás

Ez az áthidaló megoldás a PTI MSc szak keretein belül egy ún. Hardverprogramozás szakirány létrehozása volt. Elsődlegesen az MI BSc szak legjobb hallgatóit kívántuk ezzel megcélozni. A szakirányt élénk szakmai viták során a megbízott csoport létrehozta, majd némi átalakítás során az alábbi képet öltötte (egyszerűsített forma, részletesebben lásd itt: [1]):

Szakirány kötelező tárgyai – teljesítendő 20 kredit

Kód	Tárgynév	Kredit
PTIMHP01	Prototípuskészítés Verilog nyelven	6
PTIMHP02	Valós idejű és beágyazott rendszerek programozása	6
PTIMHP03	Mikrokontrollerek programozása	6
PTIMHP04	Új irányzatok a hardverfejlesztésben	2

Differenciált szakmai törzsanyag – teljesítendő 16 kredit

Kód	Tárgynév	Kredit
PTIMHP05	Nemklasszikus logikák	4
PTIMHP06	Többértékű logikák	4
PTIMHP07	Új számítási paradigmák	4
PTIMHP08	Kvantumszámítógépek	6
PTIMHP09	Perifériák és meghajtók	6

A szakirányon a következő három kötelezően választható blokkból kettőt kötelező teljesíteni. A nem teljesített blokk tárgyait szabadon választható szakmai tárgyként lehet elszámolni. Szabadon választható szakmai tárgyakból **6 kredit** teljesítendő.

Kötelezően választható blokk 1 – teljesítendő 14 kredit

Kód	Tárgynév	Kredit
PTIMHP10	Hálózatok teljesítményelemzése	6
PTIMHP11	Mobil infokommunikáció	6
PTIMHP12	Kapcsolás és útválasztás	6

Kötelezően választható blokk 2 – teljesítendő 14 kredit

Kód	Tárgynév	Kredit
PTIMHP13	A jelfeldolgozás matematikai alapjai	6
PTIMHP14	Kommunikációs eszközök és technológiák	6
PTIMHP15	IP alapú kommunikációs hálózatok és rendszerek	6
PTIMHP16	Térinformatika/távérzékelés	6

Kötelezően választható blokk 3 – teljesítendő 14 kredit

Kód	Tárgynév	Kredit
PTIMHP17	Genetikus algoritmusok	2
PTIMHP18	FPGA használata nagy teljesítményű számításra	6
PTIMHP19	Párhuzamos programozás	6

Figyelembe véve, hogy ez a háló a PTI MSc szak hetedik szakirányaként lett létrehozva, az első dolog, ami feltűnik a szemlélő számára vele kapcsolatban az az, hogy igen összetett, kicsit nehezen áttekinthető. Megjelennek viszont a korábbi beadványoknál kidolgozott kurzusok közül azok, amelyek relevánsak voltak a tekintett szakiránynál. Továbbá, hogy a választható blokkokból nem egy esetben a szükséges krediteket csak túlteljesítéssel lehet megszerezni. Mindezek ellenére a szakirány egyre népszerűbb lett a hallgatók körében, hamarosan mind a nappali, mind a levelező tagozaton a PTI MSc egyik vezető szakirányává nőtte ki magát. Az oktatásszervezés során felmerülő problémák azonban egyre sürgették a szakirány felülvizsgálatát, átalakítását, ami bizonyos fokú egyszerűsítést jelentett volna. Erre végül is nem került sor, mert 2012-ben a Kar vezetése elhatározta, hogy ismét megpróbálkozik az MI MSc szak akkreditálásával.

5. A harmadik pályázat

A korábbiakhoz képest nagyobb bizakodással kezdtünk a munkához. Oktatói gárdánk jelentősen megerősödött a mérnöki vonal tekintetében, mögöttünk volt a Hardverprogramozás szakirány több féléves oktatási tapasztalata, pályázatok, projektek révén számos ipari kapcsolatot építettünk ki. Ezek közül kiemelkedik az IT Services Hungary Kft-vel létrehozott szoros együttműködés, amely révén a Deutsche Telekom 650 partneregyeteme közül a Debreceni Egyetem elsőként nyerte el a **Best in Class** kitüntető címet.

Szakítva az előzményekkel, teljesen új alapokra helyeztük a készülő beadványt. Ezúttal két szakirányban gondolkodtunk:

- Beágyazott és újrakonfigurálható rendszerek (BUR)

- Infokommunikációs hálózatok (IKH)

A BUR a Hardverprogramozás szakirány tapasztalataira támaszkodott, az IKH pedig továbbra is a BSc szak megfelelő szakirányát kívánta továbbvinni. A hálót (egyszerűsített forma, részletesebben lásd itt: [2]) az alábbiak szerint alakítottuk ki:

alapozó ismeretek		
1. Hálózatok teljesítményelemzése		3. Alkalmazott matematika
2. Haladó adatbiztonság		4. Számítástudomány
szakmai törzsanyag		
1. Bevezetés az új hálózati kommunikációs technológiákba		4. Adat- és rendszermodellek
2. Logikai tervezés hardverleíró nyelven		5. Haladó információ- és kódelmélet
3. Számítógépes képfeldolgozás és alakfelismerés		6. Adatbiztonság labor
differenciált szakmai ismeretek		
Infokommunikációs hálózatok szakirány		
1. Haladó kapcsolás és útválasztás		4. Multimédia hálózatok
2. Nagysebességű kommunikációs technológiák		5. Szakmai választható
3. Haladó hálózati hibaelhárítás		6. Szakmai választható

differenciált szakmai ismeretek					
Beágyazott és újrakonfigurálható rendszerek szakirány					
1. Újrakonfigurálható beágyazott rendszerek		4. Mikrokontrollerek programozása			
2. Intelligens szenzorhálózatok		5. Szakmai választható			
3. Valós idejű és beágyazott rendszerek programozása		6. Szakmai választható			
gazdasági és humán ismeretek		diplomatervezés			
1. Menedzsment és szervezési ismeretek		1. Diplomatervezés 1			
2. Gazdálkodási és jogi ismeretek		2. Diplomatervezés 2			
diplomatervezés					
		4/60 gy.	10	gyj.	
			10/150 gy.	20	gyj.
összesen		4/60 gy.	10/150 gy.	30	2 gyj.

szabadon választható szakmai tantárgyak		
1. LabView FPGA		4. Új generációs hálózatok laboratórium
2. Perifériák és meghajtók		5. Hardver-szoftver együttes tervezés labor
3. Bevezetés a felhő hálózat informatikai szolgáltatásokba		6. Digitális jelfeldolgozás programozható áramkörökkel

Ez a pályázat sikeres volt. A MAB-határozat tartalmazott ugyan néhány megjegyzést, de azok egy része csak értelmezési kérdés volt, más részüket meg könnyen meg lehetett oldani. A szak 2014. februárjában elindult. Népszerűségét az mutatja leginkább, hogy mindkét szakirány megfelelő nagyságú hallgatói létszámmal rendelkezik. Mivel a szaknak ez volt az első féléve, lényegi oktatási és egyéb tapasztalatokkal nem rendelkezünk, de a továbbiakban a többi szakunkhoz hasonlóan a megszerzett tapasztalatokat felhasználva a tekintett szak minőségének további emelésére törekszünk majd.

6. Köszönetnyilvánítás

A publikációt a TÁMOP 4.2.2. C-11/KONV-2012-0001 projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Irodalomjegyzék

- [1] A Debreceni Egyetem Informatikai Karának weboldala, www.inf.unideb.hu (2014).
- [2] A Debreceni Egyetem Informatikai Karának új weboldala, w1.inf.unideb.hu (2014).

Az osztatlan képzés lehetőségei és kihívásai a mérnökinformatikus képzésben

Opportunities and challenges of the university level education in information technology

Szolgay Péter^a, Szederkényi Gábor^a, Nyékyné Gaizler Judit^a

^aPázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar

1083 Budapest, Práter u. 50/a

szolgay@itk.ppke.hu

szederkenyi@itk.ppke.hu

nyekyne.gaizler@itk.ppke.hu

Absztrakt: A mérnökinformatikus BSc és MSc képzéssel rendelkező Pázmány Péter Katolikus Egyetem és a Pannon Egyetem vezetése (további intézmények várható támogatása mellett) célul tűzte ki, hogy visszaállítja - újra megalapítja - az ötéves mérnökinformatikus képzést. Kiemelt célunk a klasszikus értelemben vett "egyetemi képzés" lehetőségének megteremtése, meghagyva a BSc szintű diploma lehetőségét, mint "letérést" és az MSc szintre való bekapcsolódás lehetőségét is. A szakalapítással és szakindítással kapcsolatos tapasztalatokról kívánunk beszámolni.

Kulcsszavak: mérnök informatikus szak, osztatlan képzés, bolognai rendszer

Abstract: Two universities having BSc and MSc level education in Information Technology, namely the Pázmány Péter Catholic University and the University of Pannonia (together with the probable support of further institutions) have decided to make an attempt to re-establish the five year university level education in Information Technology. Our aim is to create the possibility for the traditional university level education allowing the chance to earn a BSc degree or to join the training at the MSc level. We describe our experiences about establishing and starting of such a course.

Keywords: Information Technology courses, university level education, Bologna system

1. Bevezetés

Magyarországon az egyetemi szintű műszaki informatika képzés az 1980-as évek második felében indult el. Az 5 éves műszaki informatika (később mérnökinformatikus) szak bizonyítottan kiváló minőségű és rugalmasan továbbfejleszthető képzettséget adott országszerte az absztrakció és a gyakorlati ismeretek megfelelő arányának biztosításával. A bolognai rendszer bevezetésével a hazai egyetemek a műszaki informatika területén is elindították az osztott, BSc/MSc szintű képzéseket. Az osztott képzés üzemeltetése során elegendő tapasztalat gyűlt össze ahhoz, hogy át tudjuk tekinteni a jelenlegi helyzetet ill. az aktuális tendenciákat, és javaslatot tegyünk új megoldásokra.

A bolognai rendszer kritikáját különböző szakok esetén már többen megfogalmazták itthon és külföldön egyaránt [1,2]. Közismert, hogy a tanárképzés esetében a modell annyira gyengén működött, hogy 2013-tól az általános és középiskolai tanárokat is újból egységes, osztatlan rendszerben képzik Magyarországon. Az egységes, osztatlan, egyetemi szintű (10 féléves) képzés nem példa nélküli a műszaki vagy ahhoz közel álló területeken, így például már 2007-

ben akkreditálták az osztatlan építészmérnöki, majd 2009-ben az osztatlan erdőmérnöki szakot. Manapság az informatikai alkalmazások létrehozása és üzemeltetése sokszor nem kevésbé biztonságkritikus, mint egy építményé. (Elég, ha pl. feldolgozóüzemek vagy erőművek informatikai rendszereire, járművek szoftver-alapú szabályozórendszereire vagy elosztott banki számítógépes rendszerekre gondolunk.) Szükség van tehát olyan szakemberekre, akik képesek komplex informatikai modellezési, tervezési és rendszerintegrációs feladatokat felelősen és magas szinten megoldani. Mindezek alapján határozott meggyőződésünk, hogy az egységes, osztatlan mérnökinformatikus képzés lehetőségének újbóli megteremtése ilyen igények kielégítését nagyban elősegítené.

2. Aktuális helyzetkép különös tekintettel a mérnökinformatikus képzésre

A műszaki és informatikai mesterszakokon jelenleg (a demográfiai trendeket is figyelembe véve) lényegesen kevesebb hallgató végez, mint a bolognai rendszer bevezetése előtt az okleveles mérnök szakokon (néhány szak esetén az arány kevesebb mint 50%). A gazdaságnak és a társadalomnak ennél nagyobb létszámú egyetemi szintű képzettséggel rendelkező szakemberre lenne szüksége. Az informatikus BSc hallgatók nem elhanyagolható része a diploma kézhezvétele után könnyen munkát talál, és nem vesz részt tovább a képzésben. Problémát jelent a tehetséges végzett BSc hallgatók külföldre távozása is munkavállalás vagy további tanulmányok céljából. A (potenciálisan) magasan képzett szakemberek elvándorlása az EU perifériáiról a gazdaságilag jóval erősebb központi országok felé vagy a tengeren túlra tehát már a tanulmányok alatt megkezdődik. Eközben az egyetemek oktatási és témavezetési erőforrásaik túlságosan nagy részét kénytelenek valójában főiskolai szintű oktatásra fordítani, emiatt az egyetemek és főiskolák közötti különbség eléggé elmosódik. Hosszabb távon pedig (már most láthatóan) problémássá válik a doktori iskolák minőségi doktorandusz hallgatókkal való ellátása.

Az itthon maradók közül igen sokan munka mellett végzik MSc tanulmányaikat, így kevesebb idő jut az egyetemi önálló munkára. A külső cégeknél sokszor pusztán anyagi kényszerből folytatott munka gyakran nem jelent különösebb intellektuális kihívást, és nem is kapcsolatos azzal a tématerülettel/szakiránnyal, amellyel a hallgató az egyetemen foglalkozik. A kevesebb időráfordítás miatt gyengül az önálló munka minősége, és nehezebb tudományos vonatkozású feladatokba bevonni a hallgatókat. Bizonyos intézmények úgy reagálnak erre a helyzetre, hogy a kötelező MSc tárgyakat nappali képzésben résztvevő hallgatóknak is csak késő délutánonként tartják, azaz de facto esti képzést folytatnak. Véleményünk szerint az említett problémákra más megoldási lehetőségeket is kell keresni.

A BSc képzési szint bevezetésével az alapozó természettudományos és szakmai tárgyak óraszámát csökkenteni kellett, és az ebből adódó osztatlan képzéshez képesti hiányosságokat MSc szinten a hallgatók már sok esetben nem képesek megfelelően pótolni. A komplex modellezési és tervezési feladatok ellátásához szükséges magas szintű absztrakt gondolkodás képessége a tapasztalatok szerint pusztán munkahelyi gyakorlattal nem szerezhető meg, hiszen ez érdemben még a középiskola elvégzése utáni néhány évig (azaz az egyetemi tanulmányok elején) fejleszhető. Ezt az időt tehát nem lenne szabad elveszteni! Ebből a szempontból az MSc képzés nem tökéletesen épül a BSc tanulmányokra, hiszen a BSc képzés sokszor hangsúlyozott célja gyakorlati szakemberek képzése kevesebb elméleti ismeret átadásával. Ezen kívül a különböző egyetemek BSc / MSc szakjai között nem tökéletes az illeszkedés: szinte minden esetben vannak pótolandó ismeretek és jelentős átfedések is. Ezek illesztésére (különbözeti vizsgák stb.) sokszor jelentős erőfeszítésre van szükség az MSc képzés első egy-két félévében, amikor viszont ideális esetben a hallgatóknak már komoly önálló munkát kellene végezniük a szakterületen.

A fent felvázolt jelenségek egyúttal komolyan veszélyeztetik a tudományos és oktatói utánpótlás nevelését a műszaki informatika területén, az egyéni tehetséggondozásra pedig az intézményeknek általában kevés erőforrásuk van.

3. Igény az osztatlan képzésre, lehetséges előnyök

Ismert tény, hogy Magyarországon jelenleg komoly munkaerőhiány mutatkozik a mérnöki és ezen belül az informatikusi képzettséggel rendelkező munkaerő területén. A párhuzamosan jelen lévő mennyiségi és minőségi informatikai munkaerőhiány jelentős veszteséget okoz a magyar gazdaságban. A hazánkban tevékenykedő műszaki, elektronikus és internetes szolgáltatásokat nyújtó, innovátor- és termelő vállalatok hatalmas informatikai tudásbázist használnak, melynek megfelelően kvalifikált emberi erőforrással történő ellátása komoly kihívás. Mindemellett ismeretes, hogy Magyarország a környező országokhoz hasonlított relatív innovációs elmaradottságának oka többek között a minőségi munkaerő hiánya, mely részben a hosszabb oktatási időtartamot és egységesebb tanulási ívet magukban foglaló képzéseken résztvevők viszonylag alacsony számára vezethető vissza. A létesítendő képzés tehát egyszerre szolgálná a munkaerőpiac igényeit, és segítené a minőségi, magasan képzett informatikai tudáspotenciál kialakulását és fenntartását. A tervezett képzés indításának további célja, hogy a tehetséges hallgatók számára a lehető legkorábban lehetőséget adjon akadémiai jellegű szakosodásra, így támogatva a tudományos és felsőoktatási utánpótlás képzését is az informatika területén. Elmondható tehát, hogy az osztatlan mesterszintű mérnökinformatikus képzés indítása időszerű, és a piaci ill. társadalmi igények alapján egyértelműen indokolható.

Véleményünk szerint a következő előnyök, lehetőségek rejlenek az osztatlan képzés újraindításában: Lehetővé válna, hogy, hogy már a bemeneti pontnál (a felvételinél) eldönthető legyen, hogy a BSc képzésnél komolyabb elméleti megalapozást igényel a hallgató. Ennek lehetővé tételére magasabb felvételi pontszám és esetlegesen emelt szintű érettségi feltételek követelhetők meg az osztatlan képzésben való részvételhez. A tervezett 10 féléves képzési idő miatt a tanterv jelenleginél kiegyensúlyozottabban kialakítható: a tanulmányok elején több idő szánható a komoly elméleti megalapozásra, hiszen az első három évben kevesebb szakmai tárgyat és gyakorlatot kell teljesíteni. Egyúttal több idő állna rendelkezésre, hogy a hallgatók elmélyülten foglalkozzanak egy önálló feladatkörrel, így nagyobb eséllyel bevonhatók lennének az intézményben folyó tudományos kutatásokba. Ily módon az osztatlan képzés minden bizonnyal támogatná az érdeklődő hallgatók lehető legkorábbi kutatás-fejlesztési fókuszú orientálását és tehetséggondozását. A kizárólagosan osztatlan képzést indító egyetemi szintű intézmények szakkal kapcsolatos adminisztratív terhei várhatóan lényegesen csökkennének az osztott képzés fenntartásával összehasonlítva.

4. A jelenlegi képzési rendszerbe illeszkedés és jogi lehetőségek

A létesíteni kívánt képzés formája osztatlan mesterképzés. Ennek megfelelően általánosságban feltételezi a teljes 5 éves képzési ciklusban való részvételt. A korábban elvégzett tanulmányok az oktatási intézmény kreditátviteli bizottsága által történő elfogadása után azonban továbbtanulási lehetőséget biztosíthat elsősorban mérnökinformatikus BSc, ezen kívül pedig gazdasági informatikus BSc, programtervező informatikus BSc ill. villamosmérnök BSc végzettséggel rendelkező hallgatóknak.

A cikk írásának idején érvényben lévő 2011. évi, a nemzeti felsőoktatásról szóló törvény 111. § (3) bekezdésében a következő olvasható: „Ha a ciklusokra bontott, osztott képzés az adott

szakon a továbbiakban osztatlan formában szervezhető meg, az adott alapképzési szakon és a teljes alapképzési szak beszámításával indított mesterszakon szakindítási engedéllyel rendelkező felsőoktatási intézmény - kormányrendeletben meghatározottak szerint - osztatlan képzés indítását kezdeményezheti.” A törvény értelmében az osztatlan mérnökinformatikus szak indítására (is) alkalmazható kormányrendelet azonban – legjobb tudomásunk szerint – jelen pillanatig még nem született meg. Ezért (sajnos) az indításhoz mindenképpen szükséges az osztatlan mérnökinformatikus szak létesítése is.

5. Összefoglalás

Rövid cikkünkben az egységes, osztatlan mérnökinformatikus képzés újbóli bevezetésének motivációit, igényét, lehetőségeit és várható előnyeit vizsgáltuk. Természetesen nem remélhetjük, hogy az osztatlan képzés bevezetése gyógyírt jelentene valamennyi felvázolt kedvezőtlen folyamatra (hiszen ezek okai nem kizárólag a képzési forma megváltozásában keresendők), de minden bizonnyal lehetőséget adna arra, hogy a tehetséges és erősen motivált hallgatók tanulmányaik kezdetétől fogva stimuláló közösségben, a jelenleginél egységesebb, a tananyag tekintetében egyenletesebben elosztott és átgondoltabb képzésben részesüljenek, és a képzést bevezető karok közelebb kerüljenek egyetemi küldetésük teljesítéséhez.

Irodalomjegyzék

- [1] Laczkovich Miklós. Bologna és a tanárképzés, *Fizikai Szemle* 6 (2013), 218 skk.
- [2] Bencze Ágnes: A magyar felsőoktatás helyzetéről: problémák és javaslatok.
<http://keszei.chem.elte.hu/Bologna/BenczeAgnes2010.doc>
- [3] 2011. évi CCIV. törvény a nemzeti felsőoktatásról.
http://net.jogtar.hu/jr/gen/hjegy_doc.cgi?docid=A1100204.TV

Osztatlan informatika tanárszak az ELTE-n

Dr. Zsakó László

Eötvös Loránd Tudományegyetem Informatikai Kar,
Média – és Oktatásinformatika Tanszék
zsako@caesar.elte.hu

Absztrakt: A 2013/2014-es tanévben újraindult a korábban sok éven ár sikeresen működött osztatlan tanárképzés, ezen belül is az informatika tanárszak.

Az előadás az újraindult képzés struktúrájáról, felépítéséről, a programtervező informatikus szakkal való átfedéseiről szól. Külön kitérünk az osztott képzésből az osztatlan képzésre való átmenetre.

Kulcsszavak: Tanárképzés, osztatlan, informatika tanár, tanterv, ekvivalenciák

1. Bevezetés

Az informatikatanár képzés több mint 30 éves múltra tekinthet vissza. Kezdetben matematika-fizika szakos tanárok vehették fel harmadik szakként (nappali, esti, illetve levelező tagozaton). A szakos megkötés a későbbiekben fokozatosan enyhült, lényegében bármely szakos tanárból lehetett informatika szakos tanár.

Még a 80-as években elindult az informatika (akkor még számítástechnika néven) tanárképzés fő szakként, alapvetően kétszakos képzésben.

Az informatika tanárképzés különböző fajtáinak indítási éve az ELTE-n:

1983/84. Számítástechnika nappali C-szak
1984/85. Számítástechnika esti C-szak
1985/86. Számítástechnika levelező C-szak
1987/88. Informatika tanárszak főszakként, nappali
2008/09. Tanári mesterszak (nappali, levelező) – osztott
2013/14. Osztatlan informatika tanárszak

2. Informatika tanári szak a múltban

A képzés időtartama 10 félév, két tanári szak esetén 10-11 félév, kreditek száma: 300-330 kredit. Az informatika tanárszakot első alapképzésben más tanári szakkal történő párosításban kell megkezdeni. A másik szak az ELTE tetszőleges tanárszakja lehet. Azon hallgatók részére, akik a másik szakjuk követelményeinek nem tesznek eleget, de a szakon legalább 60 kreditpontot gyűjtöttek, egyszakos egyetemi szintű informatika tanári oklevél adható ki.

A képzés mindenki számára egyformán kötelező alapképzésből, fakultációs részből, valamint az egy szakos tanároknak előírt tárgyakból áll. A fakultáció két szakosok számára összesen (heti óraszámban) 28 óra, egy szakosok számára 46 óra. Ezen belül mindenkinek választania kell egy 8 órás blokkot és 16 óra blokkos tárgyat (egy szakosoknak további 8 órát blokkok-

ból). További 4 óra (egy szakosoknak 14 óra) szabadon választható informatika előadások és blokkok közül.

A képzés főbb tanulmányi területei és azok aránya:

- Közismereti tárgyak: 6 kredit
- Matematikai alapozó ismeretek egyszakosoknak: 24 kredit
- Elektronikai ismeretek: 10 kredit
- Kötelező informatikai ismeretek: 94 kredit (a szigorlatokkal együtt)
- A tanári mesterséghez szükséges tanegységek: 50 kredit (szigorlattal, szakmódszer-tannal, gyakorló tanítással), egyszakosoknak 40 kredit
- Választható informatikai ismeretek: 28 kredit+egyszakosoknak további 18 kredit
- Szakdolgozat 20 kredit
- A másik szaktól 122 kredit, egyszakosoknak 60 kredit

3. Informatika tanárok osztott rendszerű képzése

A tanárképzés első hat féléve egy szakmai BSc képzésbe volt beágyazva. Esetünkben az Informatika + X tanár vagy a Programtervező informatikus BSc-ben kezdett el tanulni, vagy egy másik BSc-n belüli részismereti képzésként. A tanári mesterszak csak az MSc szinten jelent meg.

A tanárság kétszakos képzés, ami egy fő- és egy mellékszaktól jelent (első, ill. második szak, major ill. minor szak). A tanárságra készülő hallgatónak el kellett végeznie megadott, összesen 10 kredit értékű pedagógiai, pszichológiai tárgyakat. Ez a választható közismereti 5% keret terhére volt megtehető – némi szabálytalansággal. A minor szak teljesítése a BSc-ben 50 kredit elvégzését jelentette. A major szakra maradt $180 - (50 + 10) = 120$ kredit, ami a szakdolgozatot is tartalmazza.

A programtervező informatikus informatikatanár szakirány egyetemi oklevél kredit követelményei:

- 110 kredit természettudományos alapozó és kötelező szakmai tárgy;
- 50 kredit a minor szak;
- 10 kredit pedagógiai, pszichológiai tárgy;
- 10 kredit szakdolgozat.

A tanári mesterszak követelményei (5 félév, 150 kredit):

- Első tanári szak 30-40 kredit
- Második tanári szak 40-50 kredit
- Pedagógia-pszichológia 40 kredit
- Iskolai gyakorlat 30 kredit

Így a korábbi rendszerű (sok választási lehetőséget tartalmazó) 122 kredit helyett főszakként 140, mellékszakként 100 kredit szakmai képzést kaptak az informatika tanárszakos hallgatók. (Az már csak a múlt kérdése lehetne, hogy a főszakban megnövelt kreditszám ellenére miért csökkent a hallgatók szakmai és módszertani tudása.)

4. Új osztatlan informatika tanárképzés

Az új osztatlan tanárképzés kötelezően kétszakos, közös képzési szakaszt tartalmazó általános, illetve középiskolai tanárképzés.

Az általános iskolai tanár:

- Két tanári szakképzettség párhuzamos megszerzésekor: 10 félév, 300 kredit, amiből mindkét szak szakmai anyaga 100-100 kredit, a további 100 kredit pedig pedagógia, pszichológia, szakmódszertan, valamint iskolai gyakorlatok.
- Az általános iskolai és a középiskolai informatika tanári szakok közös képzési szakaszának ideje és kreditmennyisége: 6 félév, 72 kredit.
- Az általános iskolai informatika tanári szak önálló képzési szakaszának ideje és kreditmennyisége: 2 félév, 18 kredit.

A középiskolai tanár:

- Két tanári szakképzettség párhuzamos megszerzésekor: 10 félév, 360 kredit, amiből mindkét szak szakmai anyaga 130-130 kredit, a további 100 kredit pedig pedagógia, pszichológia, szakmódszertan, valamint iskolai gyakorlatok.
- Az általános iskolai és a középiskolai informatika tanári szakok közös képzési szakaszának ideje és kreditmennyisége: 6 félév, 72 kredit.
- A középiskolai informatika tanári szak önálló képzési szakaszának ideje és kreditmennyisége: 4 félév, 48 kredit.

A közös képzési szakasz:

Tanegység	1. félév	2. félév	3. félév	4. félév	5. félév	6. félév
Programozási alapismeretek 1	2+2+1					
Számítógépes alapismeretek	2+2+1					
WEB fejlesztés 1		1+2+1				
Programozási nyelvek 1,2,3,4			2+2+1	2+2+1		
Algoritmusok és adatszerkezetek 1				2+2+1		
Az informatika elméleti alapjai 1						2+1+1
Informatikai alapismeretek 1,2,3,4	1+2+1	1+2+1	1+2+1			2+0
Tantárgyi alkalmazások 1,2,3					0+4+1	1+2+1
Oktatásinformatika 1,2,3			0+2+1	0+2+1	0+2+2	
Algoritmizálás, adatmodellezés tanítása 1		2+2+1				
Feladatmegoldó szeminárium 1,2					0+2	0+2

A középiskolai tanárképzés önálló szakasza:

Tanegység	7. félév	8. félév	9. félév	10. félév	11. félév	12. félév
Programozási alapismeretek 2		2+1+1				
WEB fejlesztés 2			1+2+1			
Programozási nyelvek 3,4	1+2+1	2+2+1				
Adatbázisok			2+2			
Algoritmusok és adatszerkezetek 2	2+2+1					
Mesterséges intelligencia				2+0		
Operációs rendszerek				1+1		
Az informatika elméleti alapjai 2				2+0		
Tantárgyi alkalmazások 4			1+2+1			
Oktatásinformatika 4				0+2		

Feladatmegoldó szeminárium 3,4	0+2	0+2				
Szaktudományi záróvizsga				2		
Elemi informatika 1,2			0+2	0+2		
Informatika oktatása 1,2	0+2	0+2				
Összefüggő egyéni gyakorlatot kísérő szaktárgyi szeminárium 1,2					0+1	0+1
Szaktárgyi tanítási gyakorlat				2		

Az általános iskolai tanárképzés önálló szakasza:

Tanegység	7. félév	8. félév	9. félév	10. félév
Informatikai alapismeretek 5	0+2			
Tantárgyi alkalmazások 5	0+2			
Algoritmizálás, adatmodellezés tanítása 2		1+2+1		
Oktatóprogramok értékelése és tervezése	0+2			
Feladatmegoldó szeminárium 5,6	0+2	0+2		
Szaktudományi záróvizsga		2		
Informatika oktatása 1,2	0+2	0+2		
Elemi informatika 1,2	0+2	0+2		
Összefüggő egyéni gyakorlatot kísérő szaktárgyi szeminárium 1,2			0+1	0+1
Szaktárgyi tanítási gyakorlat		2		

A képzés érdekessége, hogy a korábbi rendszer minor szakjánál lényegesen több szakmai ismeretet tartalmaz. Ennél fontosabb azonban, hogy a major szak kreditszámánál kisebb kreditértékben sokkal magasabb szakmai színvonalat lehet biztosítani.

5. Új informatika tanárképzés és a programtervező informatikus képzés

Az osztott rendszerű tanárképzésből (mind a részismereti, ún. minor képzésből, mind a tanári mesterszokról) egyirányú átjárást kell biztosítani az új tanárképzésbe. Ezt lényegében minden, a képzésekben megjelent tárgy esetében biztosítani tudjuk. A maradék néhány tárgy pedig szabadon választható tárgyak körébe számítható be.

Ennél fontosabb kérdés az informatika tanárszak és a programtervező informatikus alapszak közötti átjárhatóság kérdése, amit a felsőoktatásról szóló törvény, illetve rendeletek mindkét irányban biztosítanak – új külső felvételi eljárás nélkül.

Az alábbi táblázat mutatja, hogy a közös tárgyak kétirányú kapcsolatot biztosítanak (vagy teljesen azonosak, vagy kreditekvivalensek), egyetlen esetben pedig a kapcsolat egyirányú:

Programtervező informatikus ABC szakirány – osztatlan informatika tanár kreditmegfeleltetés		
ABC szakirány		Osztatlan informatika tanár
Logika és számításelmélet Formális nyelvek és automaták	→	Az informatika elméleti alapjai 1
Algoritmusok és adatszerkezetek 1,2	↔	Algoritmusok és adatszerkezetek 1,2
Programozási alapismeretek	↔	Programozási alapismeretek 1

Programozás	↔	Programozási alapismeretek 2
Funkcionális programozás	↔	Programozási nyelvek 2
Programozási nyelvek JAVA	↔	Programozási nyelvek 4
Számítógépes alapismeretek	↔	Számítógépes alapismeretek
Adatbázisok 1	↔	Adatbázisok
Mesterséges intelligencia	↔	Mesterséges intelligencia
Operációs rendszerek	↔	Operációs rendszerek

Külön eset természetesen a Programtervező informatikus mesterszak utáni informatika tanárképzés, akkor ugyanis – a tartalom vizsgálata nélkül – az összes szakmai tantárgyból fel kell menteni az érintett hallgatót; a diplomához csak szakmódszertani tárgyakat kell elvégeznie. Ez a két képzés különbözősége miatt súlyos szakmai aggályokat vet fel.

Irodalomjegyzék

- [1] Juhász I. – Zsakó L. (1995): Informatikai tanárképzés és az Informatika tantárgy. Informatikai alkalmazások'95. Az NJSZT VI. Országos Kongresszusa. (Siófok, 1995. május 28-31), NJSZT, Budapest, 400-407.
- [2] Harangozó É. – Zsakó L. (1996): A levelező számítástechnika szakos tanárképzés jelene és jövője. Informatika a felsőoktatásban '96. (Debrecen, 1996. augusztus 27-30), Debreceni Universitas, Debrecen, 524-533.
- [3] Zsakó L. (2008): Az informatika TANÁRI mesterszak felépítése az ELTE-n. Informatika a felsőoktatásban 2008. (Debrecen, 2008. augusztus 27-29), Debreceni Universitas, Debrecen.
- [4] Zsakó L. , Harangozó É. (2011): Az informatika tanárképzés koncepciója. Informatika a felsőoktatásban 2011. (Debrecen, 2011. augusztus 28-30), Debreceni Universitas, Debrecen.

Linked Open Data az egyetemen

Linked Open University Data

Fleiner Rita^a, Micsik András^b

^aÓbudai Egyetem

fleiner.rita@nik.uni-obuda.hu

^bMTA SZTAKI

micsik@sztaki.mta.hu

Absztrakt: A Linked Data a Szemantikus Web technológiák egy praktikus alkalmazása adatok világméretű összekapcsolására. Az Open Data törekvés Európában is már komoly előrehaladást ért el, és áttekintő tanulmányok szerint a felsőoktatás területén is pozitívan befolyásolhatja az oktatás minőségét. Létrejött a Linked Universities szövetség is, amely az adatok Linked Open Data irányelvek szerinti publikálását segíti és fogja össze a tagintézményekben. Adatok Linked Data formátumú publikálásának kulcsfontosságú mozzanata az adatok felépítésének, sémájának a megtervezése. Jelen cikkben felvázoljuk az adatséma tervezés fő lépéseit, feladatait és a kapcsolódó fogalmak, kifejezések jelentéseit. Beszámolunk egy általunk megvalósított tervezési folyamatról, és ismertetjük az ennek eredményeként létrejött Egyetemi Adatmodellt. Munkánkban áttekintünk számos, egymást átfedő RDF sémát, és ismertetjük, hogy segítségével hogyan tudjuk leírni az egyetemi információs infrastruktúrát Linked Data formátumban. Adatmodellünkben négy fő adathalmaz osztályt különböztetünk meg, ezek a szervezetek, személyek, publikációk és tanegységek. Megadjuk a különböző osztályok adatait leíró tulajdonságokat, az osztályok adatai közötti kapcsolatokat és az adatmodell bővítés lehetőségeinek irányait.

Kulcsszavak: Linked Data, Open Data, Szemantikus Web, nyílt egyetemi adatok

Abstract: Linked Data is a practical application of the Semantic Web technologies for connecting data worldwide. The Open Data pursuit has achieved remarkable progress in Europe as well, and studies have shown that it has a positive impact on the quality of education at university level too. Schema design is a key moment in the publication of Linked Open Data. In this paper, we analyse the main steps and objectives of the Linked Open Data schema design, and we outline the related concepts. The set of schemas are chosen to describe university organization, persons, roles, courses and publications. We also describe the data properties of the different classes, the relations of the classes and the possible extensions of our model.

Keywords: Linked Data, Open Data, Semantic Web, Linked Universities

1. Bevezetés

A kapcsolt nyílt adatok (angolul Linked Open Data, rövidítve LOD) alapjait 2006-ban Tim Berners-Lee, a világháló feltalálója fektette le. A kapcsolt nyílt adatok használatával strukturált adatokat a világhálón számítógép számára értelmezhető módon jelenítünk meg, az adatokat más adatsoportokkal kapcsoljuk össze és lehetővé tesszük, hogy külső adatforrásokból az adatainkhoz kapcsolat legyen létesíthető. A kapcsolt nyílt adatok négy alaptulajdonságát Tim Berners-Lee, a Web szülőatyja 2006-ban a következőképpen fogalmazta meg [4]:

1. Adatok elnevezésére és azonosítására URI-eket (Universal Resource Identifier) használjunk,

2. A URI-k feloldható HTTP URI-k legyenek, ezáltal az azonosított adathoz tartalom társítható,
3. A URI-k mögötti tartalom szabványokra (pl. RDF, SPARQL) épüljön,
4. Az adatokhoz tartozó tartalom rendelkezzen más adathalmazra mutató kapcsolatokkal.

Az utolsó pont biztosítja azt, hogy a különböző forrású adathalmazok egymással összeköttetésben állhatnak és egymáshoz kapcsolódhatnak. Kapcsolt nyílt adatok használatát több területen is megfigyelhetjük, például kormányzati adatokat, egészségügyi kutatásokkal kapcsolatos információkat, publikációk adatait és különböző metaadatbázisokat találhatunk meg ebben a formában. Kapcsolt nyílt adatok publikálásának kulcsfontosságú mozzanata az adatok felépítésének, sémájának a megtervezése.

Az Open Data törekvés világszerte és Európában is már komoly előrehaladást ért el, és áttekintő tanulmányok szerint a felsőoktatás területén is pozitívan befolyásolhatja az oktatás minőségét. Létrejött az európai egyetemeket összefogó Linked Universities szövetség is, amely az adatok Linked Open Data irányelvek szerinti publikálását segíti és fogja össze a tagintézményekben. Az Óbudai Egyetemen elindítottunk egy projektet, aminek célja egyetemi nyílt adatok kapcsolt adatként való publikálása és ezzel a Linked Universities szövetséghez történő kapcsolódás.

Jelen publikáció célja áttekinteni a kapcsolt nyílt egyetemi adatok használatának lehetőségeit, bemutatni ezen speciális területen az adatséma tervezés folyamatát és végül javaslatot tenni egy egyetemi LOD adatsémára.

2. LOD a felsőoktatásban

A Linked Universities [8] és a Linked Education [9] két olyan kezdeményezés Európában, amely az oktatásban kívánja a kapcsolt nyílt adatok előnyeit kihasználni. A University of Southampton például 40-nél is több adathalmazt tett elérhetővé webes felületen, és ezekre immár 18 alkalmazás épül. Az Open University majdnem 3 millió állítást (triple-t) publikál saját magáról, de említhetnénk még számos spanyol és görög egyetemet is.

A LOD alkalmazásától várt előnyök leginkább az aggregált információkon, illetve a nyílt adatok használatára épülő mobil és webes alkalmazásokon keresztül válhatnak valóra. Ezek az előnyök a célközönség szerint csoportosítva a következők lehetnek. A hallgatókat igény szerint kifejlesztett alkalmazások tudják segíteni a tanulmányaikban. Már több egyetemen létezik ilyen mobil alkalmazás, amely az órarendben és az előadások helyszínében segít eligazodni. Ezen felül a tananyagok elérhetőségei, külső tananyagok kapcsolása is könnyen megvalósítható nyílt adatokon keresztül. Az elhelyezkedési lehetőségek, statisztikák, illetve a potenciális munkahelyek és elvárásaik közzététele mind segítik a hallgatókat abban, hogy minél eredményesebben ütemezzék tanulmányaikat és jól el tudjanak helyezkedni utána.

Az oktatók számára is fontos az oktatási tevékenységek áttekinthetősége, továbbá nekik a kutatási és együttműködési lehetőségek közötti tájékozódás is kiemelt. A nyílt adatokon keresztül értesülhetnek arról, hogy milyen szervezeti egységek milyen külső kapcsolatokat ápolnak, milyen kutatások folynak, milyen szaktudást keresnek az egyetemen.

Az egyetem egészére nézve is jótékony hatású a LOD alkalmazása. Első lépésként az, hogy az intézmény nyíltan publikál adatokat, azt jelenti, hogy felvállalja a közinformálást, és több esélye van egy jó hírnevet kiépíteni, illetve a médiában megjelenni. Ezáltal az intézmény kívülről is értékelhetővé válik, a trendek, a színvonal vizualizálható lesz. Alapja lehet ez az intézmények összehasonlításának is, és a végeredménytől függetlenül igaz lehet, hogy

bármilyen kevés adat is jobb a semmilyen adatnál. A felvételizők számára az egyetem nem csak a felvételi statisztikákat adhatja közre ily módon, hanem a várható költségeket, az átlagos képzési időre vonatkozó átlagokat is, amelyek zsebbevágó döntéseket alapozhatnak meg. Az innováció és a kutatások menedzseléséhez is jó alapot nyújt az adatok nyílt nyilvántartása. Végezetül általánosságban is várhatja az egyetem, hogy a LOD nyílt keretrendszerének alkalmazásával elősegíti a szolgáltatásainak javítását mind funkcionálisan, mind a minőség terén.

A fentiek alapján a közzétételre érdemes adatokat a következőképpen lehet csoportosítani:

- személyek (munkatársak, kutatók) adatai: elérhetőségek, oktatott tárgyak, egyéni érdeklődések, publikációk, kutatási tevékenységek, külső kapcsolatok, stb.
- szervezeti egységek adatai: vezetők, elérhetőség, szervezeti hierarchia, kutatások, stb.
- kurzusok, tanegységek adatai: kreditek, feltételek, oktatók, tananyagok, stb.
- képzések, szakirányok: célok, feltételek, felvételi, végzési és elhelyezkedési adatok, tárgyak, stb.
- dokumentumok: szabályzatok, tananyagok, publikációk elérhetőségei, metaadatai (dátuma, érvényessége, hatóköre, szerzői), stb.
- kutatási projektek: témák, résztvevők, publikációk, kooperációk, stb.
- egyetemmel összefüggő statisztikai adatok: hallgatói, oktatói, költségvetési statisztikák, várható költségek, képzési idők, elhelyezkedési arányok, stb.
- térinformatikai adatok: épületek, termek, szolgáltató egységek elhelyezkedése, navigáció, közlekedés, stb.
- időbeli adatok: eseménynaptár, órarend, előadások, stb.

Ilyen adatokra számos információs szolgáltatás és mobilalkalmazás építhető. Az ELTE-n készítették szemantikus web alapú beltéri navigációs alkalmazást [10]. Az Open University mobil alkalmazásai a képzések áttekintését segítik és a tárgyakhoz elérhető tananyagokat tesznek letölthetővé, míg a University of Southampton például a buszmenetrenddel kombinálva a hallgatók utazását igyekszik segíteni.

3. Adatmodell tervezése

Kapcsolt nyílt adatok publikálásának lépéseit több szerző is tárgyalta, például [1][2][3]. Mi a következő négy fő lépésből álló folyamatként határozzuk meg a kapcsolt nyílt adatok létrehozásának életciklusát:

1. Rendelkezésre álló, publikálásra kijelölendő nyílt adatok körének meghatározása
2. Adatmodell készítése
3. Adatok kapcsolt adatként való publikálása
4. Publikált adatokra épülő alkalmazások fejlesztése

Ezek a lépések összetett folyamatokat foglalnak magukban, publikációnkban a 2. lépés folyamatának részletes vizsgálatát végezzük el.

A kapcsolt adatok szabványosított adateleíró nyelve az RDF (Resource Description Framework), amellyel erőforrásokról szóló információkat ábrázolhatunk a weben (erőforrás fogalma alatt általánosan értünk bármit, ami azonosítható a weben). Az RDF adatok három részből felépülő kijelentésekből állnak, ezek az <alany-állítmány-tárgy> (angolul subject-predicate-object) hármasok (angolul triple). Az alany a leírandó erőforrás azonosítója, ami

egy feloldható HTTP URI, az állítmány (más néven predikátum) a leírandó erőforrás egy tulajdonsága, a tárgy pedig ennek a tulajdonságnak az értéke. Az alany és az állítmány helyén mindig egy feloldható HTTP URI áll, a tárgy pedig vagy egy feloldható HTTP URI vagy pedig egy szöveges leírás.

RDF segítségével két különböző weben lévő adatforrás könnyen összeköthető egymással. Az RDF hármassokat három attribútumú relációs adattáblában tárolják, amelyben a három attribútum rendre alany–állítmány–tárgy. Az RDF adatbázisokat triple store-nak is nevezik. A SPARQL az RDF lekérdező nyelve, mely RDF formátumú adatok elérését és manipulálását biztosítja.

Annak megadására, hogy egy adathalmazban milyen típusú adatok lehetnek és köztük milyen kapcsolatok állhatnak fenn, szükséges egy absztrakt szint, ami minden esetben érvényes és általánosságban jellemzi adatainkat. A Szemantikus Web az adatmodell formalizálására egy többrétegű, RDF alapú megoldást nyújt. A formalizálás első szintje az RDFS (RDF séma), amely önleíró, vagyis az RDF osztályok és kapcsolataik leírását is RDF-ben teszi lehetővé. A következő szinten az OWL helyezkedik el, amely a leíró logika alapjain fejlettebb következtetési konstrukciókat nyújt. Az RDFS és az OWL konkrét leíró nyelvűül szolgálhatnak az olyan absztrakt fogalmaknak, mint adatséma, ontológia vagy controlled vocabulary. Az elmúlt évtized tapasztalata, hogy egy sikeres, letisztult LOD szolgáltatáshoz nincs feltétlenül szükség az OWL használatára, hanem elegendő egy RDFS adatséma használata is. Így a legtöbb általános séma (pl. Dublin Core, FOAF) is RDFS-ben van formalizálva. Az RDFS használata nem zárja ki a szabályrendszerek vagy a leíró logika alkalmazását, de egyszerűbbé teszi az adatkezelést, és javítja az interoperabilitást.

Az adatmodell a konkrét környezethez vagy alkalmazáshoz írja le az ábrázolandó egyedek azonosítóinak felépítését, az egyedek osztályait, tulajdonságait és ezek kapcsolatait. Az Egyetemi Adatséma megadásával tehát egy általános szinten adjuk meg azt, hogy milyen egyed osztályok előfordulását feltételezzük egy egyetemi környezetben, ezeknek az osztályoknak mi egymással a kapcsolata és az osztály egyedei milyen tulajdonságokkal jellemezhetőek. Adatmodellt pedig egy konkrét egyetem adatainak ábrázolásához készítünk, így például beszélhetünk az Óbudai Egyetem kapcsolt nyílt adatmodelljéről.

A kapcsolt nyílt adathalmaz számára előállítandó adatmodell párhuzamba állítható a relációs adatbázis-tervezés koncepcionális modelljének elkészítésével. LOD adathalmaz koncepcionális szintű leírására szükség van több ok miatt is [7]: információkinyerés elősegítése, különböző adathalmazok kombinált lekérdezése, SPARQL lekérdezések szerkezetének felépítése. A következő pontokban az adatmodell készítés fő lépéseit és feladatait vázoljuk fel:

1. Adatmodell osztályainak meghatározása
2. URI tervezés: alap URI, adathalmaz osztályainak URI-je (szakirodalomban TBox URI-nek hívják), entitások URI felépítése (szakirodalomban ABox URI-nek hívják). Módszertani ajánlások [5][6] tartalmazzák, hogy törekedni kell rövid és stabil (az idővel nem változó) URI-k használatára.
3. Létező ontológiák, adat szótárak meghatározása, amikben definiált tulajdonságokat és osztályokat az adatmodellben használni fogunk (általánosan megfogalmazott javaslat: a már definiált ontológiák és adat szótárak elemeit használjuk és ne magunk kreáljuk meg, ha nem szükséges). Itt célszerű az egyszerűségekre törekedni. Sok létező ontológia tartalmaz tartalmi átfedést egymással. Törekedni érdemes a lehető legkevesebb számú, illetve minél ismertebb és népszerűbb ontológia felhasználására. A következő kérdések jelenthetnek támpontot ennél a lépésnél: (1) milyen segítség létezik arra, hogy egy adott témakörhöz, problémához találjunk létező ontológiát, (2) hogyan lehet

információt szerezni egy adott ontológia felhasználásának gyakoriságáról? A lépés folyamán támaszkodni lehet például a következő alkalmazásokra:

- a. Linked Open Vocabularies (<http://lov.okfn.org/dataset/lov/>): melyben ontológiák metaadataira, más ontológiákhoz mutató kapcsolataira, használatának gyakoriságára tudunk keresni.
 - b. LODStats (<http://lodstats.aksw.org/>): a CKAN rendszerben regisztrált RDF adathalmazok tulajdonságainak leírása, beleértve az általuk használt ontológiákat is.
4. Adatmodellben használt névtér prefix-ek meghatározása: ez az adatmodell osztályainak URI-jeiből, illetve a használt ontológiák URI-jeiből áll össze.
 5. Minden osztály esetén meghatározzuk:
 - a. entitásainak javasolt URI felépítését,
 - b. az osztály URI elnevezését,
 - c. szülő osztályait (ha releváns),
 - d. vele ekvivalens osztályokat (ha releváns),
 - e. alosztályait (ha releváns),
 6. Táblázatot készítünk, mely az osztály egyedeihez, mint alanyokhoz tartozó állítmányokat (predikátumokat), ezek rövid leírását és az állítmányokhoz tartozó tárgyak osztályát (ha a tárgy szöveges leírás, akkor az `rdfs:Literal` az osztály) tartalmazza. Módszertani ajánlások [5][6] buzdítják az `owl:sameAs` használatát kapcsolatok más adathalmazokkal való létesítésére, illetve az `rdfs:label`, `foaf:depiction` használatát ember által olvasható metaadatok megadásához.
 7. A relációs adatbázis tervezésnél használt egyed-kapcsolat modellhez hasonlóan célszerű az adatmodellt grafikusán is megjeleníteni. Ez segítségül szolgál későbbi SPARQL lekérdezések megírásához, a LOD adatok hatékony felhasználásához, illetve különböző LOD adathalmazok egyidejű lekérdezéséhez. LOD adathalmaz grafikus ábrázolásának nincs az egyed-kapcsolat modellhez hasonló szabványosított leíró nyelve. Mi a következő jelölési módot javasoljuk. A dia grammnak négy alapvető eleme legyen:
 - a. osztályok (jelölése ovális),
 - b. osztályokhoz tartozó predikátumok (jelölése téglalap),
 - c. osztályokat összekötő predikátum kapcsolatok (jelölése címkézett nyíl),
 - d. osztályokat összekötő szülő-gyerek kapcsolatok (jelölése dupla keretes nyíl).Színek használatával vagy aláhúzással megkülönböztethetők azok a predikátumok, melyekhez egy adott entitás esetén csak egy érték rendelhető (funkcionális tulajdonság), illetve melyhez egy adott érték esetén csakis egy entitás tartozhat (inverz funkcionális tulajdonság).

4. Egyetemi LOD adatmodell

A fentiekben ismertetett LOD életciklus lépéseinek megfelelően az adatmodell tervezést meg kell előznie a publikálásra kijelölendő nyílt adatok körének meghatározása. Az adatok körének kiválasztásakor célunk volt egy kiindulásra alkalmas, későbbiekben könnyen bővíthető, jól körülhatárolható rendszer meghatározása. Az egyetemhez kötődő személyek (munkatársak és hallgatók), publikációk, az egyetemen oktatott tantárgyak és magának az egyetemnek a szervezeti felépítése mellett döntöttünk. Az adatmodell osztályai is ennek megfelelően álltak össze. Az eredeti négy osztály (Person, Publication, Course, Organization) mellé, – a tervezés rámutatott – szükségessé vált egy ötödik Szerepkörök osztály (Role) felvétele is.

URI azonosítók tervének a következőket választottuk:

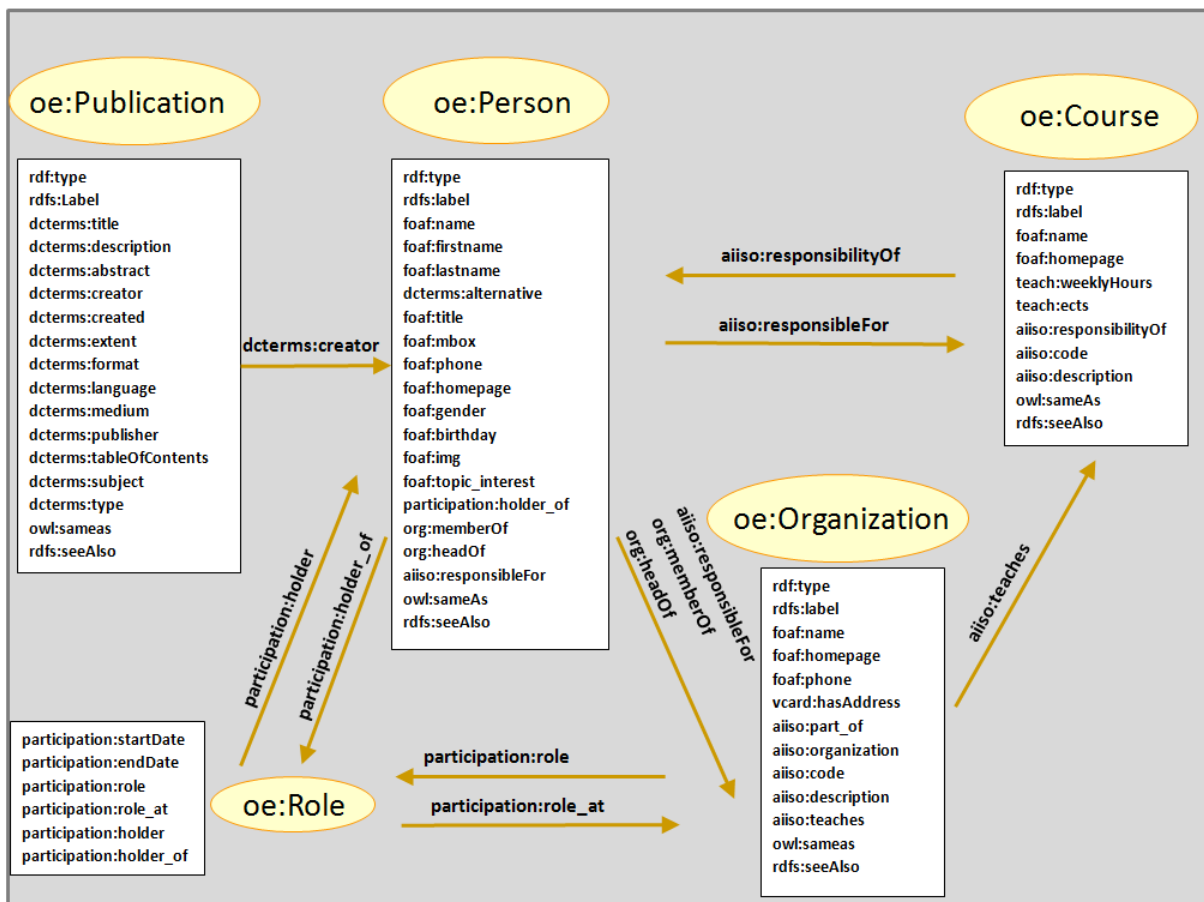
- alap URI: <http://lod.nik.uni-obuda.hu/>,
- adatséma osztály URI: http://lod.nik.uni-obuda.hu/schema#{osztály_neve},
- egyedek azonosítója: http://lod.nik.uni-obuda.hu/data/{osztály_azonosító}/egyed_ID.

A létező ontológiák tanulmányozása után és az általunk meghatározott adatkör ismeretében a következők használata mellett döntöttünk:

- foaf: <http://xmlns.com/foaf/0.1/>
- dcterms: <http://purl.org/dc/terms/>
- aiiso: <http://purl.org/vocab/aiiso/schema#>
- aiiso-roles: <http://purl.org/vocab/aiiso-roles/schema#>
- participation: <http://purl.org/vocab/participation/schema#>
- teach: <http://linkedscience.org/teach/ns#> (kredit és heti óraszám megadása)
- org: <http://www.w3.org/ns/org#>
- vcard: <http://www.w3.org/2006/vcard/ns#> (postai cím megadása)

Az adatok tárolandó tulajdonságait és másik osztályba átvezető kapcsolataikat az 1. ábra segítségével szemléltetjük. A definiált saját osztályok között a következő ekvivalenciák állnak fenn:

- oe:Person = foaf:Person
- oe:Organization = org:Organization = foaf:Organization
- oe:Role = participation:Role
- oe:Course = aiiso:Course



1. ábra. Egyetemi LOD adatmodell

Az 1. ábrán szemléltetett adatmodell alapján létrehoztunk RDF adatokat, melyeket a következő Turtle formátumú részlettel szemléltetünk:

```

oeOrg:ObudaUniversity
  rdf:type          oe:Organization;
  rdfs:label        "Óbuda University"@en;
  foaf:homepage     <http://uni-obuda.hu/en/>;
  foaf:phone        <tel:+3616665603>;
  vcard:hasAddress [
    vcard:country-name    "Hungary";
    vcard:locality        "Budapest";
    vcard:postal-code     "1034";
    vcard:street-address  "Bécsi út 96/b."
  ];
  aiiso:organization  oeOrg:NIK;
  owl:sameAs        dbpedia:Óbuda_University.

oeOrg:AI
  rdf:type          oe:Organization;
  rdfs:label        "Alkalmazott Informatikai Intézet"@hu;
  aiiso:part_of     oeOrg:NIK.

oePerson:FleinerRita
  rdf:type          oe:Person;
  rdfs:label        "Fleiner Rita";
  foaf:firstname    "Rita";
  foaf:lastname     "Fleiner";
  foaf:mbox         <mailto:fleiner.rita@nik.uni-obuda.hu>;
  org:Memberof     oeOrg:NIK;
  participation:holder_of  oeRole:R1;
  aiiso:responsibleFor  oeKnowGroup:NSTAB0MAEM.

oeRole:R1
  rdf:type          oe:Role;
  participation:startDate  "2012-09-01";
  participation:role       participation:Senior_Lecturer;
  participation:role_at   oeOrg:NIK;
  participation:holder    oePerson:FleinerRita.

oeKnowGroup:NSTAB0SAND
  rdf:type          oe:KnowGroup;
  rdfs:label        "Adatbázisok";
  foaf:homepage     <http://users.nik.uni-obuda.hu/to/tanterv/kovetelmenyek/1097/adatbazisok>;
  teach:weeklyHours "2e-2gy";
  teach:ects        "4";
  aiiso:code        "NSTAB0SAND";
  aiiso:responsibilityOf  oePerson:FleinerRita.

oePub:N1
  rdf:type          oe:Publication;
  dcterms:title     "Közigazgatási adatbázisok összekapcsolásának
                    biztonsági kérdései";
  dcterms:creator   oePerson:FR;
  dcterms:creator   [
    rdf:type        foaf:Person;
    foaf:name       "Munk Sandor";
    foaf:mbox       mailto:munk@t-online.hu
  ];
  dcterms:created   "2012";
  dcterms:publisher [foaf:name "HADMÉRNÖK on-line folyóirat"].

```

5. Összegzés

Publikációnkban elemeztük a kapcsolt nyílt adatok használatának helyzetét felsőoktatási környezetben. Feltártuk a kapcsolt nyílt adatok vonatkozásában az adatmodell tervezés folyamatát és szükségességét. Megállapítottuk, hogy kapcsolt nyílt adatok publikálásakor szükséges az adatmodell specifikált megtervezése. A kellően részletes és szemléletes dokumentált adatmodell

- támogatást ad az adataink más adatforrásokkal való kombinálására külső felhasználó számára,
- segít annak az átgondolásában és rögzítésében, hogy milyen jellegű információkat, milyen eszközök (URI azonosítók, létező adatsémák) segítségével akarunk tárolni, habár ez nem ad olyan szigorú megszorítást az adatok felvitelére, mint például a tábla szerkezet meghatározása a relációs adatbázisok esetén.

Bemutatásra került a módszertanunk alapján elkészített, egyetemi kapcsolt nyílt adatmodell is.

Irodalomjegyzék

- [1] B. Hyland, D. Wood: The joy of data - a cookbook for publishing linked government data on the web, *Linking Government Data*, Springer New York, 2011. p. 3-26.
- [2] B. Hyland, G. Ateazing, B. Villazón-Terrazas: Best Practices for Publishing Linked Data, *W3C Working Group Note 09 January 2014*, <http://www.w3.org/TR/ld-bp/>
- [3] B. Villazón-Terrazas, et al.: Methodological guidelines for publishing government linked data. In: *Linking Government Data*. Springer New York, 2011. p. 27-49.
- [4] T. Berners-Lee: Linked data-design issues, 2006, <http://www.w3.org/DesignIssues/LinkedData.html>
- [5] Hogan, Aidan, Jürgen Umbrich, Andreas Harth, Richard Cyganiak, Axel Polleres, & Stefan Decker: An empirical survey of Linked Data conformance, *Web Semantics: Science, Services and Agents on the World Wide Web*, 14 (2012), <http://www.websemanticsjournal.org/index.php/ps/article/view/287>
- [6] Christian Bizer, Richard Cyganiak, and Tom Heath: How to Publish Linked Data on the Web. *linkeddata.org Tutorial*, July 2008. <http://linkeddata.org/docs/howtopublish>
- [7] Prateek Jain, Pascal Hitzler, Peter Z. Yeh, Kunal Verma, Amit P. Sheth: Linked Data Is Merely More Data, *Papers from the AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*, 82-86. <http://corescholar.libraries.wright.edu/knoesis/79>
- [8] Linked Universities, <http://linkeduniversities.org/>
- [9] Stefan Dietze, Honq Qing Yu, Daniela Giordano, Eleni Kaldoudi, Nikolas Dovrolis and Davide Taibi: *Linked Education: interlinking educational Resources and the Web of Data*, ACM Symposium On Applied Computing (SAC-2012), Special Track on Semantic Web and Applications, <http://oro.open.ac.uk/id/eprint/31077>
- [10] Matuszka T., Gombos G., Kiss A.: *A New Approach for Indoor Navigation Using Semantic Webtechnologies and Augmented Reality*, 15th International Conference on Human-Computer Interaction, VAMR/HCI 2013, LNCS 8021 Part I., pp. 202-210. Springer, DOI: 10.1007/978-3-642-39405-8_24

KOPI-Fotó: Plágiumkeresés egy lefotózott oldal alapján

KOPI-Photo: Searching for plagiarism via a photo

Pataki Máté^a, Micsik András^b, Kovács László^c, Szabó Mihály

MTA SZTAKI

^amate.pataki@sztaki.hu

^bandras.micsik@sztaki.hu

^claszlo.kovacs@sztaki.hu

Absztrakt: Az MTA SZTAKI közismert, számos felsőoktatási intézményben használt KOPI plágiumkeresője új szolgáltatással bővült: az iPhone alkalmazás képes egy lefotózott oldalról megmondani, hogy annak ki az eredeti szerzője. A KOPI Online Plágiumkereső Portál egy egyedülálló, nyílt szolgáltatás az internetező közönség számára, amely lehetővé teszi, hogy a felhasználók saját dokumentumaik valamint mások által feltöltött dokumentumok között azonos részeket, hasonlóságot, esetleg teljes egyezést keressenek. A nyelvfüggetlen algoritmusnak köszönhetően ez bármely európai nyelven megtehető. Ehhez a szolgáltatáshoz kapcsolódik a KOPI-Fotó alkalmazás, amely egy szöveges oldalról mobiltelefonnal készült kép alapján tud keresni a KOPI Plágiumkereső adatbázisában, a magyar weben és az angol Wikipédiában. Használata egyszerű, csak le kell fotózni egy jól olvasható részt egy szövegből, és a program megmondja, hogy az megtalálható-e valamelyik a KOPI által korábban eltárolt dokumentumban, cikkben, oldalon. Amennyiben van találat, a KOPI-Fotó az egyező részeket kijelzi, és lehetővé teszi, hogy a felhasználó a megfelelő oldalra navigáljon. Ha a szöveg vagy annak egyes részei több oldalon is megtalálhatóak, akkor mindegyiket felsorolja, a legtöbb egyezést tartalmazó találattól kezdve. Előadásunkban egy demó keretében is bemutatjuk az új szolgáltatás működését.

Kulcsszavak: plágiumkereső, mobil alkalmazás, fotó, OCR, KOPI

Abstract: KOPI, the well-known Plagiarism Search Portal in Hungary, developed by the Institute for Computer Science and Control of the Hungarian Academy of Sciences (MTA SZTAKI) came out with a new service called KOPI-Photo: the iPhone app is capable to determine the original author of a text using a photo taken from a page of the document. The KOPI Online Plagiarism Search Portal is an open service for web users that enables them to check for identical or similar contents between their own documents and the files uploaded by other authors. This check function works in any European language, due to the language-independent algorithm. The new KOPI-Photo application enables the users of the service to search for the source of a text document by taking a picture of a page. The search is performed in the KOPI database, the English and the Hungarian Wikipedia and on the Hungarian Web. When a match is found the similar texts are displayed to the user so they can navigate to the source found by KOPI. If there are more sources, than all of them are listed, starting with the one that includes the most identical match. During the presentation we demonstrate how KOPI-Photo works.

Keywords: plagiarism search, mobile application, OCR, KOPI

1. Bevezetés

A KOPI Portál kifejlesztését a volt Informatikai és Hírközlési Minisztérium támogatásával az MTA SZTAKI Elosztott Rendszerek Osztálya (DSD), a Melbourne-i Monash Egyetemmel együtt, annak eredményeit felhasználva végezte. A Portál 2004-ben készült el, és azóta is szabadon hozzáférhető az érdeklődők számára.

A KOPI projekt célja elsősorban a tanárok, professzorok, konferenciaszervezők segítése a másolt művek eredetijének felkutatásában, a digitális könyvtárak védelme az illegális másolatoktól, a diákok tájékoztatása a plagizálásról és az idézés helyes módjáról, valamint a cikkek, dolgozatok, diplomamunkák értékének növelése az eredetiségük igazolásával. Érdemes kiemelni – ez az összes hasonló szolgáltatásra is igaz –, hogy a rendszer maga nem tudja megállapítani, hogy egy adott szövegrész idézet-e vagy plágium; az ilyen rendszer csak arra képes, hogy jelzi a felhasználónak, az adott dokumentumban mely más dokumentumból talált meg részeket, mekkora közöttük az átfedés vagy a hasonlóság. Annak a megállapítása, hogy ez szabályos módon történt idézés-e, és helyesen szerepel a forrás megjelölése, már a felhasználóra van bízva.

A KOPI Portál legfőbb célja a plágiumok illetve a plagizálás visszaszorítása, melynek elősegítésére az oldalon több szolgáltatás is található. A legfontosabb közülük az az információgyűjtemény, amely a plágiummal kapcsolatos tudnivalókat gyűjti össze. Mivel sokan nem tudják pontosan, mi számít plágiumnak vagy nem ismerik az idézés pontos szabályait, ezért a KOPI Portálon megtalálható a plágium definícióján kívül egy részletes leírás is arról, mi a plágium és milyen fokozatai vannak, valamint egy útmutató a helyes idézés módjáról. Az ide vonatkozó jogszabályok mellett az egyetemi szabályzatok is helyet kaptak a linkgyűjteményben, nem csak azért, hogy lássák a hallgatók, hogy milyen következménnyel jár a plagizálás, hanem azért is, mert még az idézésnek is pontos szabályai vannak. Egy diplomadolgozatban például nem lehet meghatározott mennyiségnél több idézet, hiába jelöljük meg a szerzőt, hiszen valami újat, valami sajátot is hozzá kell tenni a korábbi ismeretekhez, eredményekhez ahhoz, hogy a diplomamunka önálló, teljes munkának számítson. Hasonló módon, ha két diák közös témában ír diplomát, akkor is a diploma teljes terjedelmének csak megadott százaléka lehet közös, a többinek teljesen egyéni munkának kell lennie.

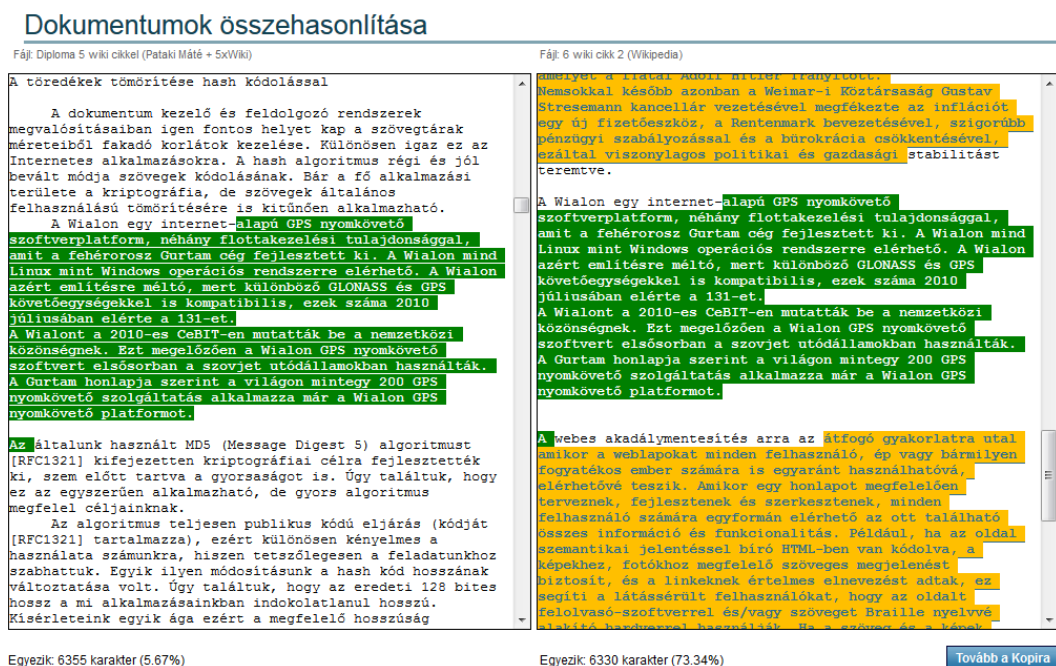
A rendszer lehetővé teszi azt is, hogy ha hasonlóságot vagy esetleg plágiumot talál egy felhasználó, akkor felvegye a kapcsolatot azzal, aki a másik művet feltöltötte, így meg lehet beszélni, hogy pontosan mi is az a mű, melyik az eredeti, ki kiről másolt. A portálnak magyar és angol felhasználói felülete is van, ezzel is elősegítve mind a felhasználói kör, mind az adatbázis bővülését. A KOPI Portál lényegét természetesen a plágiumkereső szolgáltatások adják. Azért, hogy később az adott felhasználó és a többiek által is azonosíthatóak legyenek a dokumentumok, a portálba feltöltött műveknek érdemes megadni a címét és a szerzőjét. Egyéb, részletesebb információk – mint például: kiadó, kiadás éve, kulcsszavak, vagy személyes megjegyzés – bevitelét is lehetővé teszi a rendszer.

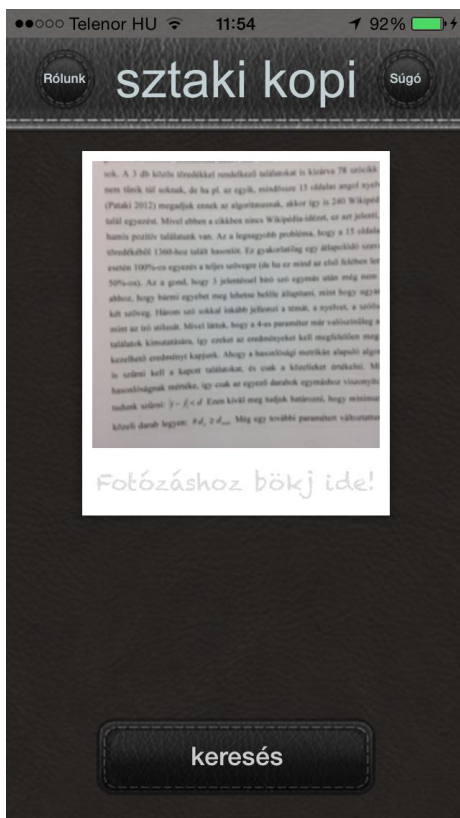
2. Folyamatos fejlődés

Az informatika rohamos fejlődésével a KOPI Portálnak is lépést kell tartania, egyfelől hogy minél jobban kielégítse a felhasználók igényeit, emellett kellő visszatartó erő legyen a plágiumok visszaszorításában. Az algoritmusok fejlesztése 2001-ben kezdődött, a KOPI szolgáltatás 2004-ben indult. 2007-ben az első három év üzemeltetési tapasztalatai alapján továbbfejlesztettük a rendszert. 2011-ben a világon elsőként beépítettünk a KOPI Plágiumkeresőbe egy fordítási plágiumok megtalálására képes, saját fejlesztésű algoritmust, amely a teljes angol Wikipédiában keres.

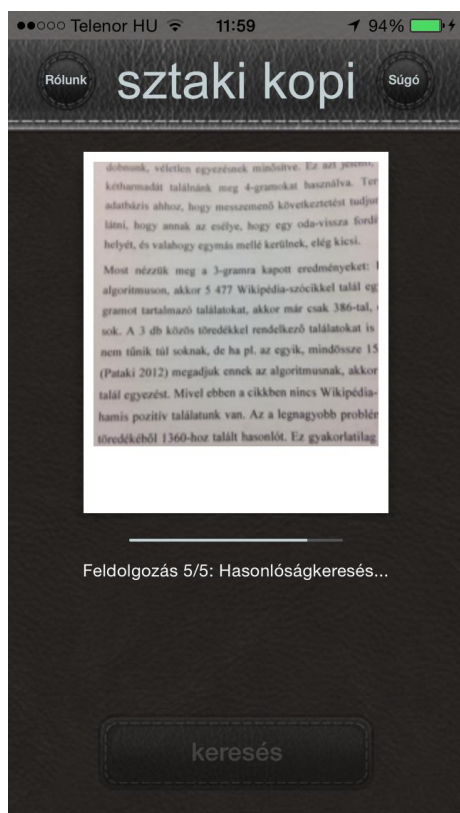
2014-ben újabb szolgáltatásokkal bővült a közkedvelt plágiumkereső. Az eddig elfogadott fájlformátumokon (doc, docx, pdf, rtf, html) felül már a nyílt odt formátumot is támogatja a rendszer, valamint megbirkózik a szkennelt dokumentumot tartalmazó pdf fájlokkal is, amelyre erős igény mutatkozott a régebbi, csak papír formátumban meglévő diplomák, dokumentumok nagy száma miatt.

A hasonlóságok megtekintését, a másolások és idézetek jobb elkülönítését segíti az ugyancsak idén debütált új szolgáltatásunk, a dokumentum-összehasonlító. Ennek segítségével bármely két, egymással közös részt tartalmazó dokumentum megjeleníthető egymás mellett, az egyező részeket a rendszer színes háttérrel kiemelve mutatja.



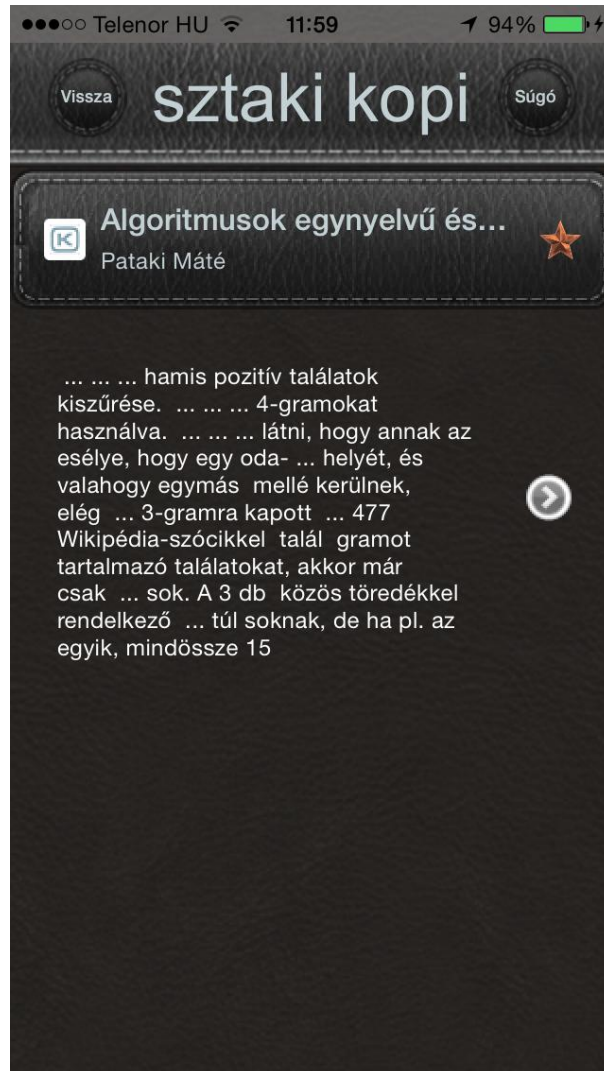


2. ábra. Fénykép készítés



3. ábra. Fut a keresés

Az eredmény a szöveg mennyiségétől függően pár másodpercen belül megjelenik. A találatokra kattintva meg lehet nézni az egyező részeket, valamint a Wikipédia esetén lehetőség van a megfelelő Wikipédia oldalra navigálni.



3. ábra. Keresés eredménye

Az applikáció a KOPI-ba feltöltött fájlok között, az angol és magyar Wikipédiában, valamint egy internetes adatbázisban keres. Utóbbiban több tíz millió magyar oldal található, főleg olyan honlapokról, amelyeken nagyobb mennyiségű szöveges tartalom van, illetve amelyek témában illeszkednek a KOPI-ba jelenleg feltöltött dolgozatokhoz.

4. Összefoglaló

A KOPI-Fotó rendszer jól kiegészíti az online plágiumkeresőt, és alkalmas arra, hogy ha valaki kétes eredetű szöveget találkozik, akár bírálatra beadott diplomadolgozatban, akár egy

újságcikkben, akkor megnézhesse, hogy a KOPI adatbázisában megtalálható-e, vagy nem a Wikipédiából, esetleg az internetről származik a kérdéses szövegrész.

Terveink között szerepel, hogy a másik két közkedvelt mobil platformra, Android-ra és Windows-ra is elkészítsük az applikációt, minden felhasználó számára elérhetővé téve a KOPI-Fotó szolgáltatást.

Irodalomjegyzék

- [1] Micsik András, Pataki Máté és Garzó András: A KOPI Plágiumkereső terhelésének elosztása cloud környezetben, *Networkshop 2013 Sopron*,
<http://nws.niif.hu/ncd2013/docs/ehu/061.pdf>
- [2] Pataki Máté és Marosi Attila: Searching for Translated Plagiarism with the Help of Desktop Grids, *Journal of Grid Computing* Vol 11, Issue 1, pp. 149-166, DOI: 10.1007/s10723-012-9224-5
- [3] Pataki Máté: Algoritmusok egynyelvű és különböző nyelvek közötti fordítások és plágiumok megtalálására, *Pázmány Péter Katolikus Egyetem, PhD disszertáció*, 2013
- [4] Pataki Máté, Vajna Miklós és Marosi Attila: Wikipedia as Text, *ERCIM News* 89, April 2012, <http://ercim-news.ercim.eu/en89/ri/wikipedia-as-text>

Infografika - rétegszemléletben

Infographics with layer approach

Takács Péter ^a

^a Debreceni Egyetem, Egészségügyi Kar, Egészségügyi Informatika Tanszék
takacs.peter@foh.unideb.hu

Absztrakt: A képzőművészet, azon belül a grafika egy alkalmazott ága az infografika. Jellemzője, hogy ötvözi a rajzi, a fényképi, a szöveges és sok más ábrázolási technika elemeit. Ezt minden esetben a hatékony, ugyanakkor látványos információközlés érdekében teszi. Sok esetben szorosan kötődik a tudomány-kommunikáció problémaköréhez, amely azzal foglalkozik, hogy egy kutatás eredményeit hogyan lehet megjeleníteni, érthetővé tenni azok számára, akik nem szakértői az adott területnek. Infografika megjelenik hagyományos nyomtatott felületeken (újságok, hirdetések, plakátok), de legjellemzőbben internetes kommunikációs oldalakon fordul elő. Egy színvonalas, tartalommal bíró, ugyanakkor figyelemfelkeltő infografika elkészítése nem könnyű feladat. A létrehozás lépései leírhatók és megközelíthetők általános módon, ugyanakkor a részletes kidolgozás már az egyén, vagy az alkotói csoport feladata. Sok nézőpontból közelíthető meg ez a folyamat. Egy lehet az a szemléletmód, ami rétegekre osztja a létrehozandó grafikát. Egy első réteg a közlendő információ tartalom szerkezeti, szervezeti nézete. Itt a szereplőket, azt a struktúrát ábrázolhatjuk, amely egy következő, „történeti” réteg alapja. Ebben a második rétegben a szereplők tevékenysége, a szerkezeti elemek egymásra hatása, a „történet” jelenítendő meg. A harmadik réteg az első kettő adattartalmát hordozhatja; azokat a számokban, grafikonokon megjeleníthető adatokat tartalmazza, amelyeket a közlő továbbítani szeretne a néző felé. Minden réteg az eltérő megvalósítandó részcélokból fakadóan természetesen más és más megjelenítési megoldásokat alkalmaz. Nyilvánvaló, hogy ez a megközelítés az infografikák csak egy bizonyos körére alkalmazható.

Kulcsszavak: infografika, rétegszemlélet, tudományos eredmények közzététele

Abstract: The infographics is an applied area of the graphics and the visual art. An infographics combines the graphical, photographical, textual and so many others charting elements. Its main purpose is not only the efficient but the spectacular communication of information in every case. It is often tightly bound to the scientific part of the disclosure of information, which tries to visualize and make the results of a research understandable for those who are not professionals of that field. The infographics usually appear in well-known typed themes such as newspapers, advertisements, posters etc.; but mostly can be found at online communicational pages. It is not an easy job to make good infographics. The making procedure of it can be described and approached in general way, but the detailed elaboration is the task of individual person or team. The procedure can be approached in many points of view. It can be an outlook which distribute the created infographics into layers. The first layer is the structural and constitutional point of view. In this case characters and structures are represented, which are the next „historical” layer’s base. In the second layer the acts of characters, the interference of structural elements, the action are visualized. The third layer may contain essence of first two layers, but the data are now in graphs and summarized in numbers, which are passed to the listeners from the announcer. The information of every layer is shown in different ways and tries to explain the purpose in other ways, but manage the same goal. Obviously this approach can be used only for the certain part of the infographics.

Keywords: infographics, layer approach, publication of scientific results

1. Bevezetés - az infografika fogalmáról

Az infografika (information graphics, infographics) emberi tudás – adatok, információk, történések, kapcsolatok, összefüggések – képi megjelenítése, bemutatása. Lényege összetett, komplex ismerethalmaz közvetítése az olvasó/néző felé minél érthetőbben, gyorsan felfogható formában. Sok esetben a létrejött alkotás túllép a hétköznapi sablonos ábrázolásokon; ötleteivel és látványos megoldásaival művész értékeket is felmutat. Infografikai megoldásokkal sok helyen találkozhatunk: újságokban, könyvekben, plakátokon, a szociális média területein (Facebook, Twitter), stb.

A fenti, pontosnak nem nevezhető meghatározáson kívül még számos más megközelítése is lehetséges az infografika fogalmának. Például a Wikipedia szerint az infografika „Több mint egy fénykép, összetettebb, mint egy rajz és sokkal informatívabb, mint az önmagában álló írott szöveg. Célja minden esetben az információközlés.” [1].

N. Iliinsky és J. Steele álláspontja az, hogy az infografika és az adat-vizualizáció közötti különbség feltárásával határozhatók meg pontosan a fogalmak körvonalai. Szerintük az adat-vizualizáció inkább az oszlop- és kördiagramok világa. A számítógépes programokkal készített ábrázolási módszerek összessége, ahol könnyű újra generálni az eredményt más adathalmazokkal. Viszonylag adat-gazdag ez az ábrázolási mód; az esztétikai szempontok nem alapvető fontosságúak. Cél az adatok közlése – sokszor az olvasóra bízva a tartalom értékelését. Ezzel szemben az infografika inkább a művészi ábrázolás felé hajló világ. Jellemző az egyedileg készített, erős vizuális tartalom – kisebb a közölt adatok mennyisége. Az esztétikai szempontok alapvető fontosságúak. Cél az adatok közlése mellett az olvasó figyelmének felkeltése, bizonyos összefüggésekre történő rávilágítás, azok kiemelése [2].

David McCandless szerint egy infografikának három komponense van. Ezek a vizuális-, a tartalmi- és a tudás komponensek, amelyek szoros kapcsolatban együtt alkotják az infografikát. A vizuális rész a grafikus elemeket, a forma és színsémákat foglalják magukba. A statisztikai adatok és tények a tartalmi részt képezik. Mindezek felül a legfontosabbnak tekinthető aspektusa az ilyen formájú információközvetítésnek az, hogy rávilágít az adattartalom mögötti tényekre és összefüggésekre – ez a momentum képezi a tudás komponensét [3].

A Bubik Veronika által szerkesztett egyetemi jegyzetben Péntek Csilla az infografikát a tudománykommunikáció szemszögéből közelíti meg, szintén az információ közvetítésére, annak átadására helyezve a hangsúlyt: „Az infografikának az a célja, hogy valamilyen információt a vizualitás nyelvére fordítson le és így segítse annak befogadását. ... Jól előkészített, érthető információt ad át az olvasónak, amit folyószövegben talán sokkal bonyolultabb lenne megérteni ...” [4].

A különböző források más és más egyéni tapasztalatból, eltérő nézőpontból kiindulva próbálnak definíciót alkotni az infografikáról, egy olyan fogalomról, ami napjainkban is időben és térben folyamatosan változik – ahogyan folyamatosan átalakuló képi megoldásokkal találkozunk a mindennapi élet egyre több területén. Az infografika újabb és újabb területeket hódít meg – például megjelent az önéletrajzok területén (infografikus önéletrajz) [5] [6] [7], és betört a mozgóképek világába (infovideó) is [8] [9].

Sokan úgy vélik, hogy ezek a kép alapú információközlések napjaink termékei, alkotásai. Azonban lényegesen korábról is tudunk infografikai példákat felsorolni. A szakirodalom az 1620-as évekre teszi az első olyan grafikai megoldásokat, amelyek már tényleges infografikáknak tekinthetők a korábbi kezdetlegesebbnek tekinthető rajzos ábrázolások után (1626; Christoph Scheiner; Rosa Ursina sive Sol; a nap forgását vizsgáló tanulmányok) [10]. A fejlődés korszakait és időbeli alakulását sok forrás tárgyalja [11] [12] [13].

A fentiekből látható, hogy a különböző nézőpontok eltérő definíciókat, meghatározásokat eredményeznek; más és más kulcselemeket emelnek előtérbe. A szerző további célja ebben az írásban a többféle megközelítés mellé egy újabb szemléletmód felsorakoztatása. A következő fejezetben egy infografika létrehozása **projekt feladat**ként kerül értelmezésre. Ez nem új a szakirodalomban, magyar nyelven is elérhető olyan forrás [4], amely hasonló megközelítést követ. Ehhez a nézőponthoz kerül itt csatolásra néhány kiegészítő adalék. Jelen leírás igazodik a projekt elméletben jól ismert életciklus szerinti megközelítéshez. Egy kisebb megbízás esetén talán túlzónak érezhetjük felvonultatni a nagyobb projektek esetén alkalmazott eszközöket. Azonban a szerző véleménye, hogy a kisebb (egy-két személy munkájára korlátozódó) feladatoknál is célszerű ez a szemléletmód. A harmadik fejezetben a projekt felfogás további eredményeként (szisztematikus folytatás) egy infografika elkészítése három rétegre bomlik (**objektum-réteg, esemény-réteg, adat-réteg**). Az objektum rétegre szorosan építve kerül kialakításra sorban az esemény-réteg és az adat-réteg.

2. Infografika létrehozása, mint projekt feladat

Egy tetszetős, figyelemfelkeltő infografika vagy infovideó elkészítése összetett, komplex feladat. Jellemzője, hogy egy konkrét, jól körülhatárolt cél érdekében készítik. Általában valamilyen megrendelés áll az elkészítés mögött, határidővel és véges forrásokkal. A véges források például jelenhetnek az emberi, az anyagi korlátokat. A létrejövő alkotás egyedi, eddig még nem létező dolog létrehozása.

A fentiek mind olyan kérdéseket érintenek, amelyekre adott válaszok egy feladat tervezése során segítenek eldönteni, hogy projekttel álunk szemben, vagy sem. A leírtak alapján kétségtelen, hogy egy-egy infografika elkészítése általában projekt feladatként fogható fel. Ebben az esetben viszont célszerű alkalmazni a projektek esetén már kidolgozott módszertani elemeket, könnyítve így az alkotói folyamatot. A projekteknél követett szigorúbb kötöttségek segítenek a korlátok között tartani a folyamatokat.

Az infografikai projektek kiindulópontja általában egy megbízáshoz vagy önálló kezdeményezéshez kötődő elképzelés vagy ötlet. Cél a megrendelő, a vevő igényeinek és elképzeléseinek kielégítése. A projektek záró eredménye az a produktum (maga az infografika, vagy az infovideó), amely a projekt végrehajtása során létrejön.

Egy infografikai projekt menedzseléséhez szükséges alkotói technikákat és módszereket is célszerű a projektek időbeli lefolyása szerint rendezni, projekt fázisokat kialakítani. Ezt nevezzük az általános projekt elmélet szerint életciklus szerinti megközelítésnek:

- A. Projekt definiáló fázisa
- B. Tervezési fázis
- C. Tervlezárás, kockázatelemzés
- D. Megvalósítási fázis
- E. Zárás, értékelés

A. Célmeghatározás vagy definiálás

Ebben a fázisban kerülnek tisztázásra az infografika elkészítésével kapcsolatos pontos elvárások; a sikeres projekt zárás kritériumai. Elemzésre kerülnek a felelősségi körök, a kommunikációs csatornák, a rendelkezésre álló erőforrások és a kapcsolódó korlátozó feltételek. Sok esetben írásban is rögzítésre kerül (projekt definiáló dokumentum) a

- projekt célkitűzése,
- vevő személye,
- nagyvonalú szervezeti környezet,
- megvalósíthatósági elemzések,
- gazdasági számítások,
- fontosabb időbeli határpontok.

Fontos rögzíteni, hogy ki a projekt vevője; és személye nem keverendő össze a végfelhasználóval, a szponzorral. Szintén figyelemmel kell lenni a projekt külső és belső szereplőire. A belső személyi körben a projekt fejlődése során kialakulnak, vagy már kezdetkor készen állnak azok a szerepkörök, amelyeket a projekt munkatársai képviselnek [14]. A külső partnerek összességének vizsgálatát a terminológia „stakeholder analysis”-nek nevezi. Magyarul talán a projekt vonatkozásában „érintettek köre” fordítás megfelelő.

További érintendő területek:

- a jelenlegi helyzethez kapcsolódó háttér információk gyűjtése és feldolgozása,
- minőségi, mennyiségi elvárások a termékkel kapcsolatban
- szabályozóknak (törvények, rendeletek, stb.) való megfelelés,
- felhasznált/ható anyagok
- környezeti elvárások,
- vevő kommunikációs igényei,
- funkcionális tartalommal kapcsolatos alternatívák, feltételezések, kizáró tényezők,
- döntési prioritások,
- stb.

B. Tervezés

A definiálási fázisban megalkotott célkitűzéseket a tervezés során érdemes lebontani elemi lépésekre, úgynevezett munkacsomagokra. Az így végrehajtott felülről lefelé történő tervezés biztosítja azokat a tevékenységeket, amelyeket felelősségi szinteken számon lehet kérni, amelyekhez erőforrások és költségek rendelkeznek, amelyekhez rendelt időbeli ütemezés áttekinthetővé válik. Néhány technikai elem a részletek nélkül:

- Tevékenység vagy munkacsomag hierarchia tervezése - Work Breakdown Structure (WBS).
- Logikai tevékenység diagram vagy hálóterv - Precedence Diagram.
- Felelős hozzárendelési táblázat.
- Végrehajtási idő és munkaráfordítás becslési táblázatok.
- Kritikus út elemzés, tartalékidő becslések.
- Naptári ütemterv vagy Gantt diagram.
- Erőforrás terhelési diagramok.
- Költségterv.

C. Tervlezárás és kockázatelemzés

Az elkészült terv és annak tartalma egy kiinduló állapotként jelenik meg az infografikai projekt indításakor (alapterv). Érdemes ezt az állapotot a tervlezárással rögzíteni. Ez egy referenciapontot jelöl ki, amely összehasonlítási alapot képez a projekt előrehaladása során. Bármilyen

Ilyen külső, vagy belső változás összemérhetővé válik a referenciaponttal. A rögzített terv azt is lehetővé teszi, hogy a projekt vezetése elemezze a projektet veszélyeztető elemeket, azok bekövetkezési esélyét, hatását, a kivédésükre fordítandó erőforrások jellegét és erőforrásigényét.

D. Megvalósítás

Ez a projekt azon fázisa, ahol a tervet végre kell hajtani. A végrehajtás során helyzetjelentéseket kell készíteni, reagálni kell a külső és belső változásokra. Néhány technikai elem a részletek nélkül:

- Monitoring: a projekt helyzetének figyelemmel kísérése - információ gyűjtés.
- Nyom(on)követés: az alapterv és az aktuális helyzet összehasonlítása.
- Eltéréselemzés: az alapterv és az aktuális helyzet közötti különbség analizálása.
- Beavatkozások meghatározása: az eltérések csökkentésére irányuló szükséges intézkedések meghozatala.
- Alapterv revíziója: szükség esetén az alapterv módosítása; a külső és belső hatások és az azokat kezelő beavatkozások okán létrejövő új állapot rögzítése; új referenciapont kialakítása.
- Mindezen állandó kommunikációja az érintettek felé.

E. Projekt lezárás

A projekt zárásának lényege a visszacsatolás a kiindulópontához. Így értékelhető az elvégzett munka, levonhatók a tanúságok a következő projekt számára. Egy cég, vagy vállalat esetén a projektzárások a szervezeti fejlődés fontos mozzanatait képezik.

A fenti pontok csak rövid összefoglalói egy projekt elkészítéséhez összegyűlt ismereteknek és megoldási technikáknak. Ennél sokkal bővebb leírások találhatóak a projekt-szakirodalomban [15] [16].

A fejezet összefoglalása egy mondatba tömöríthető: ***projekt ismeretek és tapasztalat szükséges a művészi képességek, tudás mellé egy infografika sikeres megvalósításához.***

3. Infografika, mint egy projekt eredményének megjelenítése - rétegek

Az előző fejezet azt tárgyalta, hogy egy infografika megalkotása projekt munka, és a résztvevők számára célszerű használni a projektekkel kapcsolatos felhalmozott általános tapasztalatokat. Ugyanakkor egy kutatás infografikai megjelenítése annak a projektnek a bemutatása, amely elérte a kutatási eredményt. Egy más nézőpontból, de ismét projekttel kerülünk szembe. Ekkor az infografika megvalósításakor figyelmünket irányíthatjuk:

A, a kutatási folyamatra; az eredményeket elérő személyekre, csoportra – ***ki/kik végezték a vizsgálatot;*** az eredményekre – ***mit vizsgáltak;*** az eredmények eléréséhez szükséges részlépésekre – ***hogyan történt a vizsgálat, módszer; mikor és hol*** történt a vizsgálat.

B, A kutatás tárgyára és eredményére; kikkel, mivel kapcsolatos az eredmény – *kit/mit vizsgáltak; mi a vizsgálati eredmény*; az eredménynek milyen térbeli – *hol*; és *idő*beli vonatkozásai vannak.

A két nézőpontot összeköti maga a vizsgálat. A térbeliség és időbeliség megjelenítése általában speciálisabb formában, térképi vonatkozásokkal, időszalaggal (balról jobbra, fentről le) történik.

Egy *infografika* esetében *általában a B*, nézőpont kerül előtérbe, az A, rész kiegészítő információként jelenik meg. Ezzel szemben egy tudományos közleményben elvárt az A, nézőpont hangsúlyosabb leírása.

A **B**, nézőpontra, az eredményekre koncentrálva, a megjelenítés rétegekre bontható¹

- I. a kutatás tárgya (kit/mit vizsgáltak) – *objektum-réteg*
- II. az objektumok kapcsolata, történések (vizsgálati eredmény) – *esemény-réteg*
- III. az előző két réteg adat szintű jellemzése – *adat-réteg*

Amennyiben követjük a rétegekre bontás szerinti megközelítést, úgy *első lépésben* össze kell gyűjteni az infografikán megjelenítendő *személyek, objektumok* körét. Ezt követően rögzíteni kell, hogy milyen formában jelenjenek meg az említett elemek. Sokszor célszerű metafora rendszereket kialakítva igazodni az olvasó/néző feltételezett ismereteihez. Ezen a szinten történik a színsémák és más objektumokhoz kötődő grafikus elemek meghatározása is. Egy csoportmunkát vezető menedzsernek ekkor fokozottan figyelnie kell, hiszen ez az a pont ahol a szigorú projekt tervezés vissza kell vonuljon és egyre nagyobb teret kell engedjen a művészi alkotóerőnek. A rétegek szerinti munka *következő lépése* a már rögzített szereplők és/vagy objektumok közötti *kapcsolatok, viszonyok* ábrázolása. Ez jelentheti egy feltárt természeti folyamat egymást követő lépéseinek ábrázolását, ahol például nyilakkal jelölhetjük az események sorrendjét (a már rögzített személyek, objektumok között); vagy lehet egy emberek közötti szituáció ábrázolása, ahol a folyamatok értelmezését az olvasó/néző előzetes ismereteire alapozzuk – a felhasználható ötletek száma az alkotó képzelőerőtől függ. Az *utolsó lépés* a harmadik réteg megalkotása az előző két réteghez kapcsolva. Itt a szereplők, az objektumok és a kapcsolatok, viszonyok *jellemzőit szöveges és/vagy grafikus* elemekkel írjuk le. Rövid értelmező leírások, adattáblák, vagy azok diagram formájú ábrázolásai kerülnek ekkor az infografikába.

Az infografikai munka részeredményeinek elérése, azok érintettek felé történő kommunikálása ismét a projekt szemlélet erősödését kell maga után vonja; minél előrehaladottabb az infografika annál inkább vissza kell térniük a projekt résztvevőinek a szigorúbb projekt szemlélethez, az időbeli (határidők, mérföldkövek) és erőforrásbeli (költségek) korlátok betartásához.

Érdekes a bemutatott projekt szemléletmód szerint is elemezni néhány infografikát. Vannak olyan megoldások, ahol a második réteg sokkal kisebb szerepet játszik. Például ilyen megoldások szerepelnek a világ legmagasabb épületeit (Tallest buildings of the world) [18] [19], vagy a naprendszer (Solar system) [20] [21] bemutató infografikák között. Arra is találunk példákat, amikor a harmadik réteg megjelenése sokkal mérsékeltebb. Ilyenek például a víz (water cycle) [22] [23], vagy a szén (carbon cycle) [24] [25] természetbeli körforgását bemutató infografikák.

¹ A bemutatott megközelítés párhuzamba állítható az objektumorientált tervezési módszertani egy részével. Bővebben: [17] 4. fejezet.

4. Összefoglalás

Egy infografika komplex módon, egyes rész-elemeket hangsúlyozva tárja elénk a világ összefüggéseit, a szerzők által lényeginek vélt vonásait. Az infografikák létrehozása összetett feladat, projektek megvalósítását jelentik. Ugyanakkor az infografika sok esetben maga is egy projekt eredményeit jeleníti meg. A bemutatott gondolatok arra hívják fel a figyelmet, hogy számos létező más nézőpont mellett kialakítható egy projekt szemlélet is, amelynek elsajátítása, figyelembe vétele megkönnyítheti és elősegítheti még színvonalasabb munkák létrejöttét.

Irodalomjegyzék

- [1] Wikipedia: Infografika. <http://hu.wikipedia.org/wiki/Infografika>
Letöltés: 2014.07.04.
- [2] Noah Iliinsky, Julie Steele: Designing Data Visualizations. O'Reilly Media Inc., Sebastopol 2011.
- [3] David McCandless: The beauty of data visualization; TED előadás, 2010 -
http://www.ted.com/talks/david_mccandless_the_beauty_of_data_visualization
Letöltés: 2014.07.04.
- [4] Balázs Barbara, BubikVeronika (szerk.), Hadabás Gitta, Hegyi Béla, Kárpáti Andrea, Király Andrea, Péntek Csilla, Váradi Judit, Zsupponits Anett: Vizualizáció a tudomány-kommunikációban. Egyetemi jegyzet, ELTE, TTK, 2013.
- [5] Infografika mint önéletrajz. http://infographics.blog.hu/2011/02/08/infografika_mint_oneletrajz
Letöltés: 2014.07.04.
- [6] Három látványos infografika CV. <http://szuperinfografika.blogspot.hu/2013/06/harom-latvanyos-infografika-cv.html>
Letöltés: 2014.07.04.
- [7] Infographic Visual Resumes. <http://www.pinterest.com/rtkrum/infographic-visual-resumes/>
Letöltés: 2014.07.04.
- [8] Értéktér – Infovideó. „A gazdasági felemelkedés társadalmi-kulturális feltételei” TÁRKI kutatás (2009) alapján. Forgatókönyvet írta: Babarczy Eszter; forgatókönyv alapján a szövegírási feladatokat végezte: Tóth Károly (Carnation); vizuális tervezés: Varga Zsolt (Carnation); animáció: Tarján Zoltán (Carnation); producer: Parrag Krisztina és Fábián Péter (Carnation). <http://www.youtube.com/watch?v=E01M-YDG-6Q>
Letöltés: 2014.07.04.
- [9] Media Toolkit - animirana infografika.
<http://www.youtube.com/watch?v=oXefMNFHVow>
Letöltés: 2014.07.04.

- [10] Michael Friendly: A Brief History of Data Visualization. In: Handbook of Computational Statistics: Data Visualization, Vol. III. Editors: C. Chen and W. Härdle and A. Unwin. Springer-Verlag, 2006.
- [11] Edward R. Tufte: Envisioning Information. Graphics Press, Cheshire, Connecticut, USA, 1998.
- [12] Friendly, M. & Denis, D. J. (2001). Milestones in the history of thematic cartography, statistical graphics, and data visualization. Web document, <http://www.math.yorku.ca/SCS/Gallery/milestone/>
Letöltés: 2014.07.04.
- [13] Mark Smiciklas: thePowerof Infographics - Using Pictures to Communicate and Connect with Your Audiences. Pearson Education Inc. 2012.
- [14] Szabó Gabriella, Csanádi Péter: „... és akkor jött a Tenkes kapitánya! ...” - avagy mivel javíthatjuk a projektmenedzsment sikerességét? Híradástechnika, LXV. Évfolyam 2010/1-2. p. 40-43.
- [15] Project Management Institute: Projektmenedzsment útmutató (PMBOK® Guide) 5. kiadás. Akadémiai Kiadó, Budapest, 2013.
- [16] Görög Mihály: A projektvezetés mestersége. Aula Kiadó, Budapest, 2003.
- [17] Kondorosi Károly, Szirmay-Kalos László, László Zoltán: Objektum orientált szoftverfejlesztés. ComputerBooks Kft., Budapest, 2007.
<http://www.tankonyvtar.hu/hu/tartalom/tkt/objektum-orientalt/index.html>
Letöltés: 2014.07.04.
- [18] The World's Tallest Buildings [Infographic]. <http://laurencebradford.com/tallest-buildings-east-asia/>
Letöltés: 2014.07.04.
- [19] The World's Tallest Buildings [Infographic]. <http://blog.maxwellsystems.com/building-to-the-sky-infographic/>
Letöltés: 2014.07.04.
- [20] Scott McNeill: The Solar-system [Infographic]. <http://www.scottmcneill.ca/Solar-System-Infographic>
Letöltés: 2014.07.04.
- [21] Michael Babwahsingh: The Solar system [Infographic]. <http://michaelbabwahsingh.com/fullsolarsystem/>
Letöltés: 2014.07.04.
- [22] Water cycle [Infographic]. <https://www.behance.net/gallery/12047479/Water-Cycle-Infographic>
Letöltés: 2014.07.04.
- [23] Water Infographic: The Water Cycle [Infographic]. <http://www.thegreenmarketoracle.com/2013/09/world-water-day-poster-water->

cycle.html

Letöltés: 2014.07.04.

[24] Ken Marshall: The carbon cycle: Nature in balance [Infographic].

<http://kenmarshallgraphics.com/infographics11.html>

Letöltés: 2014.07.04.

[25] Infographics Carbon Wallpaper [Infographic].

<http://wallpaper.com/wallpaper/infographics-carbon-272054>

Letöltés: 2014.07.04.