

AZ INFORMATIKA KORSZERŰ TECHNIKÁI

Cserny László–Hadaricsné Dudás Nóra–Nagy Bálint (Szerk.)

BUZÁNÉ KIS PIROSKA–TERÉNYI ANNA–KOVÁCS DAVID Kérdőíves felmérések megbízhatóságának növelése CZIRKOS ZOLTÁN Egyszerűen megvalósítható plágiumkeresés nagy létszámú kurzusok beadandó feladataihoz CSUTA GÁBOR–BERECZ ANTONIA Velünk élő történelem – A teáskamrától a Toy Story-ig FRIEDEL ATTILA–KACZUR SÁNDOR Naprendszer-szimuláció a *Virtuális valóság modellezése* tárgyban ADAM GRANICZ–ANTON TAYANOVSKIY–LOIC DENIZIERE Experience Report: Developing a Cloud- Based Integrated Development Environment for F# Web and Mobile Applications HADARICS KÁLMÁN–LEITOLD FERENC Evaluate security risk for cyber attacks on information systems HONFI VID–SEBESTYÉN–HORVÁTHNÉ KOVÁCS BERNADETT– NAGY MONIKA ZITA Környezetimodell-építés és adaptálásának aspektusai HORVÁTH BOTOND–HADARICS KÁLMÁN–KOVÁCS DAVID–LEITOLD FERENC Védelmi intelligenciával rendelkező ISR kialakítása HOSSZÚ GÁBOR Topológiai eltérések minimalizálására visszavezetett graféma leszármazási vizsgálatok KATONA JÓZSEF–UJBÁNYI TIBOR–KÖVÁRI ATTILA Agyszámítógép-interfészek kialakításának, tervezésének szempontjai KIRÁLY ZOLTÁN A körző a több száz éves számolóeszköz KOVÁCS CSABA ISTVÁN–SZABÓ JÓZSEF Active Directory alapú egységes informatikai azonosítási rendszer bevezetése

a Dunaújvárosi Főiskolán NAGY BÁLINT Hopf-bifurcation in higher dimensions NÉMETH GÁBOR–RÉTVÁRI GÁBOR Igényfüggetlen statikus elosztott útvonalválasztás – geometriai nézőpont ÖLLÖS GERGELY–VIDA ROLAND Lokális extrapolálás vezeték nélküli szenzorhálózatokban TÓTH CSABA–ÁGOSTON GYÖRGY Development opportunities of carpooling in Hungary TÓTH LORÁND LEHEL–RAYMOND PARDEDE–HOSSZÚ GÁBOR Grafémaanalízis kanonikus összetevők alapján UJBÁNYI TIBOR–KATONA JÓZSEF–KÖVÁRI ATTILA A non-invazív mobil EEG-regisztráló és-jelfeldolgozó eszközök összehasonlítása



Dunakavics
KÖNYVEK 2.

DUF PRESS–Ú M K

Cserny László–Hadaricsné Dudás Nóra–Nagy Bálint (szerk.)

AZ INFORMATIKA

KORSZERŰ TECHNIKÁI

A kötetet lektorálta Bánkuti Gyöngyi, Berke József, Buttyán Levente, Buza Antal,
Cserny László, Csiszár Csaba, Heszberger Zalán, Joós Antal, Pataricza András,
Póser Valéria, Seebauer Márta, Strauber Györgyi, Tóth János, Zachár András,
Illés Zoltán, Molnár György, Muha Lajos

© Cserny László, Hadaricsné Dudás Nóra, Nagy Bálint, editors 2014

© Ágoston György, Berecz Antónia, Buzáné Kis Piroska, Czirkos Zoltán,
Csuta Gábor, Friedel Attila, Hadarics Kálmán, Honfi Vid Sebestyén, Horváthné Kovács Bernadett,
Kaczur Sándor, Katona József, Király Zoltán, Kovács Dávid, Kővári Attila, Leitold Ferenc,
Mihalovicsné Kollár Anita, Nagy Bálint, Nagy Mónika Zita, Németh Gábor, Öllös Gergely,
Rádai Levente, Rétvári Gábor, Terényi Anna, Tóth Csaba, Ujbányi Tibor, Vida Rolland, 2014

DUNAÚJVÁROSI FŐISKOLA
www.duf.hu

ÚJ MANDÁTUM KÖNYVKIADÓ
www.ujmandatum.hu

Felelős kiadó András István, Németh István
Felelős szerkesztő Nemeskéry Artúr

Tördelés Duma Attila
Készült a HTSART nyomdában
Felelős vezető Halász Iván

ISBN 978 963 287 069 4
ISSN 2064-3837

AZ INFORMATIKA KORSZERŰ TECHNIKÁI

Cserny László–Hadaricsné Dudás Nóra–Nagy Bálint (Szerk.)



Dunakavics
KÖNYVEK 2.

DUF PRESS-Ú·M·K

Budapest, 2014

Tartalom

Előszó (Dr. Király Zoltán—Nagy Bálint)	7
<u>HORVÁTH BOTOND—HADARICS KÁLMÁN—KOVÁCS DÁVID—LEITOLD FERENC</u> Védelmi intelligenciával rendelkező ISR kialakítása	9
<u>ADAM GRANICZ—ANTON TAYANOVSKYY—LOIC DENUZIERE</u> Experience Report: Developing a Cloud-Based Integrated Development Environment for F# Web and Mobile Applications	21
<u>KOVÁCS CSABA ISTVÁN—SZABÓ JÓZSEF</u> Active Directory alapú egységes informatikai azonosítási-rendszer bevezetése a Dunaújvárosi Főiskolán	35
<u>UJBÁNYI TIBOR—KATONA JÓZSEF—KÖVÁRI ATTILA</u> A Non-Invazív Mobil EEG-regisztráló és -jelfeldolgozó eszközök összehasonlítása	46
<u>HOSSZÚ GÁBOR</u> Topológiai eltérések minimalizálására visszavezetett graféma-leszármazási vizsgálatok	60
<u>FRIEDEL ATTILA—KACZUR SÁNDOR</u> Naprendszer-szimuláció a <i>Virtuális valóság modellezése</i> tantárgyban	72
<u>CZIRKOS ZOLTÁN</u> Egyszerűen megvalósítható plágiumkeresés nagy létszámú kurzusok beadandó feladataihoz	85
<u>ÖLLÖS GERGELY—VIDA ROLLAND</u> Lokális extrapolálás vezeték nélküli szenzorhálózatokban	98

<u>KÁLMÁN HADARICS—FERENC LEITOLD</u> Evaluate security risk for cyber attacks on information systems	110
<u>TÓTH LORÁND LEHEL—RAYMOND PARDEDE—HOSSZÚ GÁBOR</u> Grafémaanalízis kanonikus összetevők alapján	122
<u>NÉMETH GÁBOR—RÉTVÁRI GÁBOR</u> Igényfüggetlen statikus elosztott útvonalválasztás – geometriai nézőpont	137
<u>CSUTA GÁBOR—BERECZ ANTÓNIA</u> Velünk élő történelem – A teáskannától a Toy Story-ig	148
<u>CSABA TÓTH—GYÖRGY ÁGOSTON</u> Development opportunities of carpooling in Hungary	160
<u>HONFI VID SEBESTYÉN—HORVÁTHNÉ KOVÁCS BERNADETT—NAGY MÓNICA ZITA</u> Környezetimodell-építés és adaptálásának aspektusai	171
<u>DR. KIRÁLY ZOLTÁN</u> A körző a több száz éves számolóeszköz	181
<u>BUZÁNÉ KIS PIROSKA—TERÉNYI ANNA—KOVÁCS DÁVID</u> Kérdőíves felmérések megbízhatóságának növelése	190
<u>KATONA JÓZSEF—UJBÁNYI TIBOR—KÖVÁRI ATTILA</u> Agyszámítógép–interfészek kialakításának, tervezésének szempontjai	198
<u>BÁLINT NAGY</u> Hopf bifurcation in higher dimensions	205
<u>RÁDAI LEVENTE—HONFI VID SEBESTYÉN—DR. KIRÁLY ZOLTÁN— MIHALOVICSNÉ KOLLÁR ANITA</u> ERP-rendszerek oktatása a Dunaújvárosi Főiskolán	211

Előszó

Az Informatika Korszerű Technikái című konferenciát két évente rendezi meg a Dunaújvárosi Főiskola Informatikai Intézete.

A rendezvény alkalmat teremt nemcsak a doktoranduszok, hanem a tapasztalt kutatók számára is eredményeik bemutatására. Teret nyit a fejlesztésben alkalmazott módszerek megvitatása mellett ipari és kutatói kapcsolatok kialakítására, melyek az oktatásban is új utakat nyitnak. Ennek eredményeként a Dunaújvárosi Főiskolán megjelentek az iparból érkezett oktatók, akik tudásukat közvetlenül adják át a hallgatóknak, részt vesznek a helyi kutatócsoportok munkájában, vagy éppen ők indítanak el egy kutatást az Informatikai Intézetben.

A jövőben induló duális képzésekhez is jó alapot nyújtanak a konferencián szerzett kapcsolatok. A képzések kialakítását segíti az a tudás, aminek átadását, bővítését az akadémiai és az ipari szektor közösen végezheti. A hallgatókat egyre nagyobb számban vonjuk be az alkalmazott kutatások körébe, amelyek keretén belül tapasztalatokat szerezhetnek, elmélyíthetik gyakorlati tudásukat, így javítva munkaerő-piaci esélyeiket.

Az utóbbi időben az előadásokban előtérbe helyeződött az alkalmazott kutatások és az ipari igények támasztotta informatikai problémák megoldása. A konferencián az alkalmazott kutatások eredményei mellett több olyan információ is megjelent, mely a közeljövő kutatási irányait határozza meg.

Jelen könyvbe azokat a dolgozatokat választottuk ki, melyek újszerűségükkel, eredeti gondolataikkal kiemelkedtek a számos előadás közül. A cikkek sokfélesége nemcsak hasznos, hanem érdekes olvasmány-nyá is teszi ezt a kötetet.

Köszönjük a lektorok munkáját, akik értő figyelemmel igyekeztek az írásokat javítani. Ezáltal a dolgozatok az olvasó számára érthetőbbek, a szakértő számára hasznosabbak lettek.

Dr. Király Zoltán—Nagy Bálint

Védelmi intelligenciával rendelkező ISR kialakítása

Összefoglalás: Napjaink hálózati támadásai során a támadók egyre bonyolultabb, és kifinomultabb módszereket használnak a célhálózatba való bejutásra. Mindenkinnek, aki használja az internetet, legyen akár otthoni felhasználó vagy vállalkozás, fontos, hogy adataira, informatikai rendszereire különös figyelmet fordítson. Sokszor a támadók az internet eléréshez használt router sérülékenységet kihasználva juthatnak be egy belső hálózatba, ezen eszközt felhasználva képesek lesznek a hálózati forgalom átirányítására, forgalom monitorozására, jelszavak és érzékeny adatok megszerzésére. Manapság léteznek olyan Linuxot futtató routerek, amelyek esetében lehetőség van saját operációs rendszer telepítésére. Ezáltal a router olyan funkciókat is elláthat, melyekre a gyári szoftverrel nem, vagy csak korlátozottan képes. Ezáltal a frissítés, továbbfejlesztés lehetősége bármikor adott. Amennyiben egy KKV hálózatában több telephely is van, vagy több szintű határvédelem van, akkor szükség lehet arra, hogy az eszközök kommunikáljanak egymással. A kommunikáció lehetővé teszi, hogy megosszák egymással „tapasztalataikat”, például egy betörési kísérletre vonatkozóan. Az egyes üzenetváltások kapcsán a routerek felkészülnek az egyes támadásokra, képesek lesznek egy-egy betörési kísérlet azonnali felismerésére, és megakadályozására. Ebben a cikkben rámutatunk arra, hogy milyen módon alakítható ki egy ilyen rendszer Linux alapú firmware használatával. Ismertetjük továbbá a kialakított rendszer működési modelljét.

Kulcsszavak: Router, firmware, kommunikáció, védelem, elosztott rendszer.

Abstract: Nowadays attackers use more and more sophisticated methods to gain access to a specified network. Everybody who use internet services as a home or a corporate user should be very careful to take care of personal data and applied information systems. Attackers frequently use vulnerabilities of routers that applied for internet access. Through the routers attackers are able login to a computer which connected to the intranet and catch all

*DF-hallgató
E-mail: botond@veszprog.hu

**DF Informatikai Intézet
E-mail: hadarics@mail.duf.hu

***DF-hallgató
E-mail: kdavid721@gmail.com

****DF Informatikai Intézet
E-mail: fleitold@veszprog.hu

[1] Az OpenWRT hivatalos weboldala: <http://openwrt.org>, 2012. 01. 10.

the traffic of the specified networks. It makes them possible to monitor and redirect the network traffic and catch passwords and sensitive data. These days there are Linux based routers in which everybody can change the operating system of the router. Using an own operating system the router can provide extra functions that are not implemented or usable limited in the original firmware. This modification makes possible further development. When a specified SME has two or more locations or has a border protection the applied devices should communicate with each other. The communication makes them possible to exchange their “experiences” for example the data of a break-in attempt. Using messages of the other routers make a router possible to detect or prevent an intrusion. In this paper we show who to develop a system previously mentioned using routers contain Linux based firmware. Moreover we will describe the working modell of the prepared system.

Keywords: Router, firmware, communication, protection, distributed system.

Bevezetés

Napjainkban rengeteg kisebb, nagyobb eszközt csatlakoztatunk saját belső hálózatunkra vagy az internetre. Az eszközök számának növekedésével új technológiák, operációs rendszerek jelennek meg a hálózatunkban. Mivel mindegyik eszköz csatlakozik a hálózatra, és különböző módszerek és protokollok segítségével kommunikálnak, ezért a hálózat egyre inkább heterogén lesz. Ennek következtében az egyes eszközök felépítésében, és működésében rejlő biztonsági problémák is előtérbe kerülnek. Éppen főként emiatt szükséges, hogy a hálózat egyes elemeit megerősítsük, képessé tegyük arra, hogy akár automatikusan felismerjenek adott biztonsági incidensekre utaló jeleket. Továbbá másik eszközöktől képesek legyenek speciális üzenetek fogadására. Ezen üzenetek elsődleges célja, hogy megosszák egymással „tapasztalataikat”, egy másik eszköz által már „átélt”, tapasztalt eseményt fel tudjanak ismerni, és ezáltal már például egy támadás kezdetekor a szükséges védelmi lépéseket meg tudja tenni az eszköz automatikusan. A továbbiakban elsősorban olyan ún. ISR-eszközök (Integrated Services Router) családjába tartozó eszközökkel foglalkozunk, amelyek többféle hálózati funkciót biztosítanak egy kompakt egységként. A CISCO, vagy egyéb gyártók eszközei saját szoftverrel (firmware-el) rendelkeznek. Ezen eszközök szoftverének bővítése általában nem lehetséges, éppen ezért valamilyen nyílt forráskódú firmware sokkal jobb választás. Ezeknek a kernelét, és a rajtuk futó szoftvereket könnyedén tesztre szabhatjuk, saját komponenseket, alkalmazásokat telepíthetünk és használhatunk. A következő fejezetben az általunk is használt OpenWRT-t [1] fogjuk bemutatni.

Az OpenWRT rövid története, jellemzői

A Linksys cég 2002 decemberében kiadta a WRT54G típusú routert, mely szoftvere nyílt forráskódú (GNU, GPL-licenc alá tartozó) komponenseket tartalmazott. Mivel a GPL-licenc megköveteli a forrás megnyitását, a közösség ezért nyomást gyakorolt a cégre, az pedig engedett, és nyilvánossá tette a WRT54G router firmware-ét. Ezzel a szoftverfejlesztők, rendszergazdák, kísérletezők, hobbi-Linuxosok hatalmas segítséget kaptak a beágyazott eszközökre való fejlesztéshez. A cég a WRT54G vonalon később Linux helyett VxWorks valós idejű operációs rendszerre váltott az 5.0-s változatban.

[2] Tomato firmware weboldala:
<http://www.polarcloud.com/tomato>,
2012. 05. 14.

AZ OPENWRT TÖRTÉNETE, ELŐZMÉNYEI

A **HyperWRT** projekt célja az volt, hogy bővítsék a WRT54G routerek képességeit, közben pedig minél közelebb maradjanak az eredeti, hivatalos forráshoz. A Linksys routerének későbbi változatai (2006-tól kezdve) már nem Linux alapon működtek, ezért ezekkel már nem kompatibilis. A Linksys a harmadik féltől származó firmware-ek futtatására kiadta a WRT54GL routert.

Később ez a projekt lett a Tomato. [2]

Az **OpenWRT** egy Linux alapú disztribúció beágyazott rendszerekre. A fejlesztők rengeteget tanultak a Linksys rendszerénél használt ötletekből. Ezt a firmware-t főleg routerekben alkalmazzák, de robotoknál, VoIP-rendszereknél, céges irányítási rendszereknél, „intelligens otthon” megoldásoknál is használható. Ennek oka, hogy a mérete kicsi, a rá telepíthető programokat úgy fordították és csomagolták, hogy minél kisebb helyet foglaljanak, és minél kevesebb felesleges funkciót tartalmazzanak. Az eredmény egy gyors, minimalista Linux-rendszer, amiben mégis megtalálható a legtöbb népszerű nyílt forrású szoftver.

A disztribúció két alapító tagja Gerry Rozema (groz) és Mike Baker (mbm/embeem). Az első verziója 2004 januárjában jelent meg, alapjául a Link-

sys WRT54G firmware szolgált, illetve az uClibc-projekt buildroot nevű patch gyűjteménye, amit azért készítettek, hogy könnyebb legyen létrehozni egy beágyazott rendszerekkel kompatibilis Linux firmware-t. Ezzel az eszközzel lefordíthatjuk a célrendszerre (x86, ARM, MIPS, PowerPC, stb) a kívánt szoftvereket (kernel, fájlrendszerek, bootloader). Az OpenWRT a hivatalos Linux-kernelt használja, és ezt patcheli. Verziókezelő rendszerként subversiont (svn) használnak.

AZ OPENWRT JELLEMZŐI, TELEPÍTÉSE

Ahhoz, hogy a saját megoldásainkat integrálhassuk a rendszerbe, tudnunk kell lefordítani az OpenWRT-re szoftvereket. Hasznos ez akkor is, ha nagy mennyiségű eszközre szeretnénk feltelepíteni ugyanazt a konfigurációt.

Ehhez szükségünk van egy Linux disztribúcióra, ez fogja biztosítani a fordítási környezetet. Bármelyik disztribúció használható, de ajánlott egy Debian, vagy valamelyik származéka (például Ubuntu). Telepítenünk kell a gcc/g++ fordítókat (C/C++ nyelvhez), illetve make, és egyéb fordításhoz szükséges szoftvereket. Mivel az OpenWRT az SVN verziókezelő rendszert használja, ezért erre is szükségünk van.

A backfire verzió tükrözése a következő paranccsal történik:
svn checkout svn://svn.openwrt.org/openwrt/branches/backfire
Ha a legújabb (béta) verziót szeretnénk, erre is van lehetőség:
svn checkout svn://svn.openwrt.org/openwrt/trunk/

A stabil ághoz tartozó forráskód 120–130MB helyet foglal, és körülbelül 17 ezer fájlból áll. Mivel ez csak maga a firmware, a hozzá tartozó programcsomagokat is le kell töltenünk. Az előbb létrejött egy backfire könyvtár, ebbe kell belépniünk, és az itt található feeds.conf.default fájlt át kell másolnunk a feeds könyvtárba feeds.conf néven. Ezután végrehajtjuk ezt a két parancsot:

```
./scripts/feeds update -a  
./scripts/feeds install -a
```

Alapértelmezetten ez letölti a backfire verzióhoz tartozó felhasználói programokat, az xwrt és luci webes felületeket. A feeds.conf-ot bővíthetjük a saját csomagforrásainkkal. A backfire könyvtár most már 300–310MB-ra nőtt, és 52 ezer fájl tartalmaz.

A konfigurálás és fordítás hasonló a Linux-kernel fordításának műveleteihez:

```
make defconfig  
make package/symlinks  
make menuconfig
```

Ekkor megjelenik egy menü ahol be lehet állítani, hogy miket fordítson le. Érdemes csak azokat beállítani, amik a router működéséhez feltétlenül szükségesek, és azokat, melyeket mindegyik routerünkön használni fogunk, mert a telepítésre később is lesz lehetőség. Beállíthatjuk például itt az OpenVPN-t, az SSH-elérést, ha nem feltétlenül szükséges a webes felület, akkor azt kihagyhatjuk.

Ami a TP-Link 1043nd típusú routerhez mindenképp kell:

Target System: Atheros AR71xx/AR7240/AR913x

Target Profile: TP-LINK TL-WR1043ND

Ha mindennel megvagyunk a végén elmentjük és kilépünk belőle. A firmware fordításához a következő parancs szükséges:

```
make V=99
```

illetve ha több szálon szeretnénk:

```
make -j4 V=99
```

Ezután elkezdődik a fordítás, ami a gép sebességétől függően tipikusan 30–60 percet vesz igénybe.

A végeredményt a bin/ar71xx könyvtárban fogjuk találni.

Ha végeztünk a fordítással, a router webes felületére belépve frissíthetjük a régi firmwaret (akár gyáriról, akár OpenWRT-ről frissítünk).

Ezután újraindul a router, és be tudunk lépni webes felületen (ha telepítettük), illetve SSH-n.

OpenWRT-t futtató eszközök használata kisvállalati környezetben

A cégeknél egy telephely esetén nincs probléma a kommunikációval, a csoportmunkával, mivel minden adat egy helyen, egy hálózatban van. Több telephelynél viszont felmerül néhány probléma:

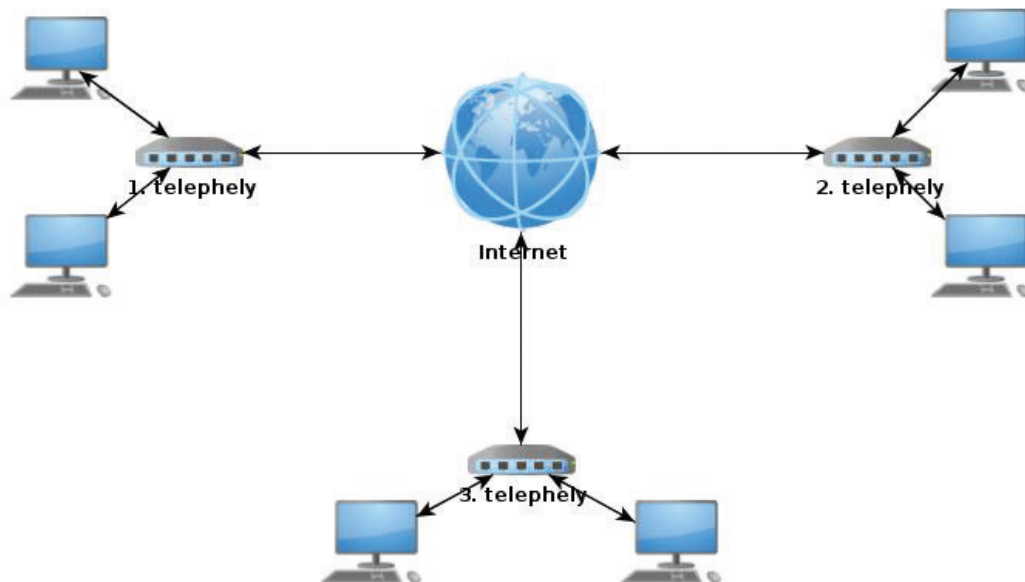
- Az adatok, dokumentumok verziói több helyen vannak szétszórva.
- A kommunikáció két telephely között nem feltétlenül zajlik biztonságosan.
- Ha egy telephelyet támadás ér (vírus, rosszindulatú támadók), akár elhárítjuk ezt a támadást, akár nem, a többi telephely még sérülékeny marad.

Nem csak több telephelyről beszélhetünk, hanem a távmunka lehetőségéről is. Ebben az esetben a dolgozók otthonról, kávézóból, szabadon elérhető WiFi-hálózatokból lépnek be a cégünk rendszerébe, hogy elvégezzék a napi feladataikat, megnézzék a leveleiket, letöltsenek egy-egy dokumentumot.

OPENWRT-T FUTTATÓ ESZKÖZÖK HASZNÁLATA TÖBB TELEPHELYES HÁLÓZATBAN

Az 1. ábrán látható egy példa több telephelyes hálózatra. A routerekhez csatlakoztathatóak akár további switchek.

1. ábra. Több telephelyes hálózat.



A feladat a biztonságos kommunikáció garantálása, a megoldás egy olyan rendszer, ami ezt egyszerűen, azonnal elvégzi a telephelyek között, vagy pedig a feladatra tökéletes egy olyan router, melyet szabadon módosíthatunk, beállíthatunk. OpenWRT-vel előre telepítve ezek az eszközök elláthatják a védelmi feladatokat (tűzfal, valamilyen formában vírusvédelem), a biztonságos kapcsolatokat (titkosított virtuális hálózat), szinkronizációra képesek (tűzfalbeállítások, vírusminták szinkronizálása a múltbéli támadások alapján).

AZ OPENWRT BIZTONSÁGI MEGOLDÁSAI

A tűzfal feladatait jól ellátja a Netfilter, a Linux-kernel csomagszűrője. Képességei között szerepel az IPv4/IPv6 (iptables/ip6tables parancsokkal), NAT-támogatás, bővíthető külső fejlesztők által készített kiegészítőkkel, illetve léteznek hozzá frontendek (pl. Shorewall).

Biztonságos hálózatok kialakítására például a VPN-technológiát használhatjuk. Ezen belül több típus áll rendelkezésünkre:

- IPSec (szerver és kliens)
- PPTP (szerver és kliens)
- Cisco Easy VPN (csak kliens)
- OpenVPN (szerver és kliens)

A Cisco Easy VPN csak kompromisszumokkal használható, kliensként kiválóan működik a router.

Ha szerverként szeretnénk használni, akkor a többi megoldáshoz kell folyamodnunk. Egyszerűsége miatt az OpenVPN-t választottuk, ehhez a router webes felületére is telepíthető konfigurációs oldal. Az OpenVPN széles körben támogatott, egyre több eszközbe építik be. Beállítani egyszerű, de bonyolultabb feladatokat is képes ellátni. Mivel nem igényli a kernel patchelését vagy módosítását, ideális olyan rendszerekre, ahol ez nem megoldható.

Titkosításra és azonosításra az OpenSSL-szoftvert használja, ismeri a TLS-t, használhatóak RSA-kulcsokkal. Képes UDP- és TCP-protokollon is kommunikálni, dinamikus, DHCP-vel működő hálózatokat is összekötni, a teljes forgalmat a VPN-interfészen továbbítani. Nem kompatibilis viszont az egyéb VPN-implementációkkal.

Telepítés OpenWRT-n:

```
opkg install openvpn openvpn-easy-rsa  
opkg install luci-app-openvpn
```

VÉDELMI MEGOLDÁSOK ÉS HONEYPOT-LEHETŐSÉG KIALAKÍTÁSA

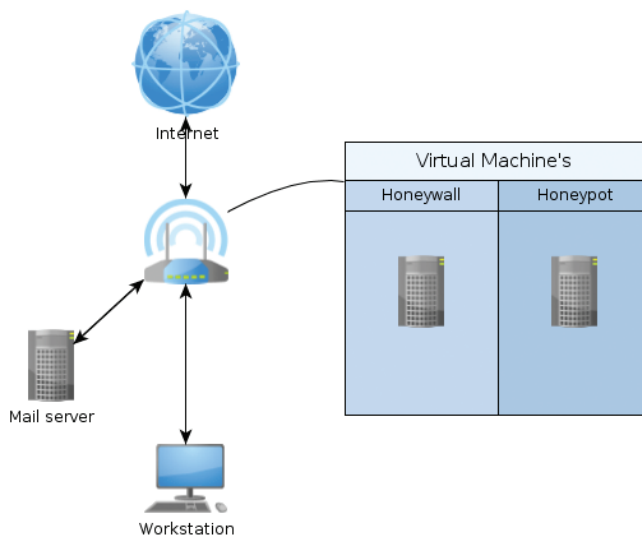
A rendszer fejlesztésének már kezdeti állapotában a biztonságot helyeztük előtérbe. A ma jelenleg forgalomban levő SOHO-megoldásoknál biztonságosabb, funkciógazdagabb megoldást tűztünk ki célul.

Kezdeti problémát a szűkös erőforrások jelentették, mivel a teszthez használt eszköz csupán 32MB memóriával rendelkezik. A firmware-be jelenleg két fontos védelmi funkciót implementáltunk.

Az egyik egy honeypot, mely a hálózaton történő támadás esetén álcélpontokat helyez el, ezáltal felkeltve a betörő figyelmét. A másik egy intelligens információtovábbító-rendszer mely segítségével a routerek megoszthatják egymással a tapasztalataikat a betörési kísérletekből.

A honeypotok alkalmazása a nagyvállalati környezetekben már régóta elfogadott, mint nagy pontosságú, korai észlelő rendszerek. Segítségével a biztonsági szakemberek tanulmányozhatják az egyes betöréseket, mellyel az esetleges biztonsági hiányosságokat pótolhatják. Mivel a honeypotokon nem található semmilyen éles rendszer vagy szolgáltatás, az ellenük irányuló összes aktivitást gyakorlatilag támadásnak vehetjük és megtehetjük a szükséges lépéseket, hogy a hálózatunkban található fontosabb szolgáltatásokat leállítsuk, vagy csak egyszerűen rögzítsük az eseményeket. Fontos megjegyezni, hogy a honeypotokat elsődlegesen az „általános” támadások kivédésére használhatjuk fel. Az egyik kiemelt szolgáltatás, ami beépítésre kerül az address harvesting, mely segítségével a támadó által használt robotok által gyűjtött email adatbázisokat megfertőzzük hamis email címekkel. (2. ábra)

2. ábra. Honeypot kialakítása.



A virtuális megoldás mellett szólt, hogy így a támadáshoz tudjuk szabni a csalirendszert, ezáltal az még hihetőbbé válik. A csalirendszer képes a hálózatban található szolgáltatások jellemzőit lemásolni, így képesek vagyunk MySQL-szervertől kezdve akár egy LDAP-kiszolgálót is szimulálni.

A rendszer alapja egy snort-variáns, mely egy magas interakciójú honeypot. Magas interakcióról akkor beszélünk, amikor csak azt figyeljük, hogy ki és hogyan akar a rendszer felhasználójává válni a rendszergazdai csoport engedélye nélkül.

EGY XML-ALAPÚ BIZTONSÁGI PROTOKOLL

A hálózati eszközök védelmi rendszerei úgy vannak megtervezve, hogy képes legyen az adott eszköz és egy bizonyos „belső hálózat” védelmére. Ez annyit jelent, hogy jelenleg hiányoznak azok a szabványosított megoldások, amelyek segítségével „védelmi információk” párbeszéde megoldott lenne.

Ahhoz, hogy az egyes hálózati eszközök biztonságos, és megbízható módon tudjanak üzeneteket cserélni, számos feltételnek kell teljesülnie:

- A felek közti kommunikációnak szabványosnak, lehetőség szerint platform-függetlennek kell lenni.
- Az egyes eszközök közti kommunikáció a normál hálózati forgalom részeként zajlik, éppen ezért lehetőség szerint biztosítani kell a megfelelő szolgáltatás minőségét (QoS).
- Bármilyen adatnak titkosított formában kell eljutni egyik végpontból a másikba.
- Az egyes eszközök csak megfelelő feltételek teljesülése esetében (hitelesítést követően) fogadhatnak adatokat más eszközöktől.
- A kommunikációs megoldásnak egyszerűnek, és gyorsnak kell lennie, továbbá kis mértékű sávszélességet emésztget fel.

Amennyiben egy router működését szeretnénk kiterjeszteni a korábbiakban felsorolt feltételek szerint, akkor mindegyikre kell megoldást találnunk. Mivel elsősorban nyílt forráskódú router firmware-ben gonolkodunk, ezért az ott elérhető megoldásokkal, protollokkal kell számolnunk.

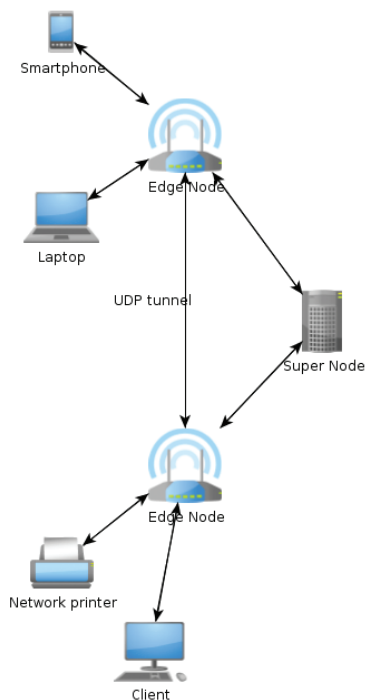
A probléma lehetséges megoldásai:

- A kommunikáció XML alapon történik, ezáltal az egyes üzenetek jól formázottsága, és megfelelősége ellenőrizhető.
- Titkosított alagutak (tunnels) kialakítása a kommunikációhoz; ezáltal a megfelelő eljárás biztosítja a kapcsolatot.
- A hitelesítés elvégezhető kulcs-párok alapján.

Az új rendszer tervezésekor kiemelt szempontnak tartottuk, hogy a rendszer a betörési kísérletek tapasztalatait egy biztonságos kapcsolaton keresztül meg tudja osztani több hasonló routerrel. Ez azért volt fontos, mert így a későbbi hasonló támadásokat sokkal könnyebb kezelnie a rendszernek és az esetleges adatszivárgás is megelőzhető.

Egy DDOS-támadás esetén nagyon könnyen elfogyhat a sávszélesség, így megnehezítve vagy lehetetlenné téve az információcserét. Ennek kiküszöbölésére a rendszerbe beépítettünk egy traffic control modult mely segítségével egy minimális sávszélességet „lecsippentünk”. Ezt a sávszélességet egy, az általunk használt speciális üzenetek továbbítására szolgáló hálózat felállítására használtuk fel annak érdekében, hogy az információ minél gyorsabban elterjedjen az útválasztókon.

3. ábra. A routerek közti üzenetváltások módszere.



Biztonsági megfontolások

Hiába alakítottunk ki egy kommunikációs csatornát az egyes részegységek között a támadások leírásának kommunikálására, ha az abban közölt információkat a támadó szabadon megváltoztathatja. Ezért ebben a fejezetben olyan eljárásokat illetve technológiákat mutatunk be, melyek a biztonság fokozását segítik elő.

A BIZTONSÁGOS KOMMUNIKÁCIÓS-CSATORNA KIALAKÍTÁSA

Első körben azt a problémát kell megoldani, hogy az egyes gépek kommunikációja rendben legyen. Ehhez egy olyan VPN-megoldást kell alkalmazni, ami egy külső titkosító modullal képes együttműködni. Habár a VPN-megoldások többsége képes titkosított csatornát létrehozni, ez csak egy kiegészítő funkció.

AZ EGYÉRTELMŰ AZONOSÍTÁS PROBLÉMÁJA ÉS MEGOLDÁSA

Egy biztonsági adatokat szállító hálózat esetében nem csak az adatok illetéktelenek által történő elolvasása a probléma, hanem, hogy egy támadó az adott kommunikációba ne legyen képes hamis adatokat elhelyezni. Ezért szükséges egy olyan egyedi hitelesítési eljárás bevezetése, mellyel egyértelműen hitelesíthető a forrás kiléte.

A nyilvános kulcsú infrastruktúra minden szereplőjének van nyilvános kulcsa és privát kulcsa. Minden szereplő titokban tartja a saját privát kulcsát (ezt jellemzően csak ő ismerheti), míg a nyilvános kulcsukat nem kell titokban tartaniuk, akár nyilvánosságra is hozhatják őket. Amikor nem biztonságos csatornán (pl. interneten) keresztül történik a két fél közti kommunikáció, e kulcsok segítségével írják alá az üzeneteiket.

XMPP-ALAPÚ HITELESÍTETT KOMMUNIKÁCIÓS-PROTOKOLL

Az Extensible Messaging and Presence Protocol (rövidebb nevén XMPP) egy 1999-ben indult és 2004-ben szabványosított XML-en alapuló kommunikációs-protokoll, ezt használja például a Jabber. A Jabber azért különleges, mert az e-mailhez hasonlóan hálózatszövetséget alkot, vagyis az egyes telephelyek független hálózatot alkotnak, de ha megbíznak egymásban, akkor a kapcsolat létrejöhet. Nagy előnye más protokollokkal szemben, hogy az üzenetek XML-alapúak. Fontos kiemelni, hogy decentralizált hálózatról beszélünk. Az XMPP-alap esetben TCP szállítási szolgáltatás segítségével juttatja célba az XML-adatfolyamot.

[3] A netfilter.org projekt
weboldala: <http://netfilter.org>,
2012. 01. 1.

[4] Northcutt, S.–Novak, J.
(2001): *Network Intrusion
Detection: An Analyst's Hand-
book*. 2nd edition. Indianapo-
lis: New Riders.

[5] Spitzner, L. (2003):
Honeypots: Tracking Hackers.
Addison. Boston: Wesley.

[6] Northcutt, S.–Zeltser,
L.–Winters, S.–Kent, F. K.–
Ritchey, R. W. (2003): *Inside
Network Perimeter Security*.
Indianapolis: New Riders.

Összegzés

Napjainkra belátható, hogy a technológia (a társadalom szélesebb körében való elterjedése miatt), az információtárolás a normál papír alapú (materia- lista) megoldások helyett egyértelműen a digitális világba költözik. A fel- használók joggal feltételezik, hogy adataik és végső soron az eszközük fölött ők rendelkeznek, mindezt úgy, hogy ne kelljen az eszközeik biztonságára feltétlenül ügyelniük, hanem ezt az erre a célra kialakított megoldások bizto- sítsák. Ebben a cikkben ezen probléma hatékony kezelésére kínálunk egy olyan módszert, mely a napjainkra egyértelműen elterjedt informatikai támadásokkal szemben képes megvédeni a felhasználókat. A cikkben is- mertetett módszer szerint egy újfajta online-hálózat segítségével az eszközök képesek megosztani a támadási kísérletek során szerzett tapasztalataikat, hogy egyre hatékonyabban legyenek képesek azokat elhárítani. Ehhez szük- ség van egy hatékony hitelesítésre, amivel az egyes eszközök egyértelműen azonosíthatóak.

Experience Report: Developing a Cloud-Based Integrated Development Environment for F# Web and Mobile Applications

Abstract: In this report, we present our ongoing experience developing a cloud-hosted Integrated Development Environment (IDE) for authoring F# web and mobile applications. Our IDE enables users to write F# code with modern features such as code completion and on-the-fly type checking, to execute exploratory, interactive F# code in the cloud, to work with distributed version control systems and projects hosted within, to compile F# code to JavaScript for execution in the browser and to instantly deploy web applications. The IDE does not impose any client side requirements beyond a modern web browser, and no plugin-based components (Flash or Silverlight) are used in its design. The IDE itself is written in F# and compiled to JavaScript with our WebSharper framework [1] to execute in the browser, and its server-side components run on Microsoft Azure.

Keywords: Integrated Development Environment, F#, cloud, Azure, JavaScript, WebSharper.

Összefoglalás: Ebben a tapasztalat jelentésben beszámolunk egy felhő-alapú, online Integrált Fejlesztői Környezet (IDE) kialakításáról, amely alkalmas F# webes és mobil alkalmazások fejlesztésére. Az IDE-nk lehetővé teszi hogy a fejlesztők a manapság divatos fejlesztői kényelmi szolgáltatásokkal, mint pl. kódkiegészítéssel és azonnali típusellenőrzéssel, fejlesszenek F# kódot, futtassák azt interaktívan, használjanak elosztott verziókövető rendszereket, és hogy a böngészőben futtassanak F# kódból generált JavaScript-kódot, vagy komplett webes alkalmazásokat. Az IDE futtatásához nincs másra szükségünk mint egy modern web-böngészőre, annak fejlesztéséhez pedig nem használtunk plugin-alapú komponenseket mint pl. Flash vagy Silverlight, mint az konkurens környezeteknél szokásos.

Az IDE-t teljes egészében F#-ban fejlesztettük, webes futtatásához a saját

* E-mail: granicz.adam@intellifactory.com

** E-mail: anton.tayanovskyy@intellifactory.com

*** E-mail: loic.denuziere@intellifactory.com

[1] The WebSharper website. <http://www.websharper.com>. (2012. 11. 05.)

webes keretrendszerünket, a WebSharper-t [1] használtuk, és annak szerveroldali komponensei Microsoft Azure környezetben futnak.

Kulcsszavak: Integrált Fejlesztői Környezet, F#, felhő, Azure, JavaScript, WebSharper.

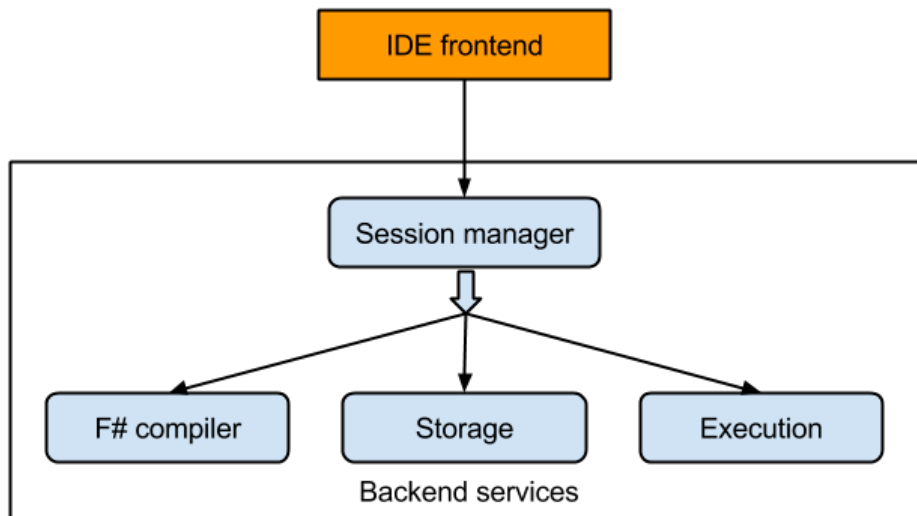
Introduction

Office, email and other business applications are continuously moving to the web, making their installation as simple as bookmarking a web page. Updates are seamlessly transferred to the user by a simple browser page refresh, offering significant savings on deployment and administering updates. This makes web-based applications an attractive choice for various businesses, especially those where organizational policies around security are stringent.

In this report, we present on our ongoing experience developing a cloud-hosted Integrated Development Environment (IDE) for authoring F# web and mobile applications. Our IDE enables users to write F# code with modern code assistance features such as code completion and on-the-fly type checking, to execute exploratory, interactive F# code in the cloud, to work with distributed version control systems and projects hosted within, and to compile F# code to JavaScript for execution in the browser. The IDE does not impose any client side requirements beyond a modern web browser, and no plugin-based components (Flash or Silverlight) are used in its design. The IDE itself is written in F# and compiled to JavaScript with our WebSharper framework [1] to execute in the browser, and its server-side components run on Microsoft Azure.

Architecture

The architecture of our IDE is depicted in Figure 1. It consists of a cloud-hosted backend that provides various services to clients connecting through its main IDE frontend. This backend utilizes and runs on Microsoft Azure, and enlists several Azure services in order to perform its various functions.

Figure 1. A simplified view of the IDE architecture.

Connecting clients are assigned into individual sessions by the main session manager. This component is designed to keep track of user sessions and all their accompanying server-side state, effectively isolating sessions from each other. When a session is established, a web socket connection is built between the connecting frontend and the supporting backend services. This web socket serves as a single channel to communicate between the client and the backend, and is used by both tiers to send messages to each other. Refer to Section 4.2 for more implementation details of this messaging interface.

One of the core backend functionalities is the F# compiler as a service. This service provides incremental syntax and semantic checking, and various code assistance features such as information about errors and code completion choices for a given code location. This service also provides means to interpret F# expressions in a stateful session, by keeping track of and executing the F# Interactive toolset in the background.

User sessions are created to manipulate workspaces. A workspace is a collection of files and containing folders. A workspace is created when initiated by the user, and can be opened on request. The storage service provides a file system abstraction around workspaces to the frontend. It is responsible for managing files and directories that belong to the current editing session. These are in turn stored in cloud storage and synchronized as user edits occur. Concurrent user sessions that try to manipulate the same files are

[2] The Try
F# website.
[http://tryf-
sharp.org](http://tryf-sharp.org).
(2012. 11. 05.)

left unprotected, and the latest edits win over previous ones without an attempt to provide synchronization. While this means no support for editing the same files from two or more concurrent sessions, or propagating changes to a given file opened in multiple sessions, it does provide for basic collaboration through recognizing certain file system operations such as file creation and deletion. When these file operations are initiated from one user session, they are propagated to the others with a reasonably small delay automatically.

The execution service manages the execution of external processes on behalf of the front-end user as part of their user session. These processes come about when the user requests well-defined operations such as running a source control command, or triggering a build of the current solution being edited, and usually exit shortly after they are invoked, producing a visual feedback in the IDE. However, they are terminated forcefully if they fail to exit by the time the user session is closed. This makes sure that there are no dangling processes left executing without a live parent session.

Basic bookkeeping of users, their profiles and workspaces are kept in Azure storage. This ensures that sessions can be constructed from any location, with the underlying workspace access rights enforced properly.

Features

We give a short overview of some of the key features of our IDE in the subsections below. A handful of features are only touched upon and explained briefly, and some will be expanded in separate papers and reports due to their more complex nature.

USER PROFILES AND WORKSPACES

One of the key design principles of the IDE enables anonymous users, e.g. users without an account, to arrive at the IDE site, either directly, or indirectly from various F# community sites, and to use the IDE to edit and experiment with code, similar in vein to exploratory F# programming sites such as Try F#. [2] In addition, each such user connecting to the IDE has the option of persisting their work, creating a new or reusing an existing user account, effectively joining the workspace currently in the IDE to the user.

In the hopes of providing better security and protection of user data, the IDE itself does not implement any custom access control functionality. Instead, it delegates this work to FPish [3] to authenticate users through its oauth [4] support. FPish enables users to sign up with further oauth-compatible login providers, including trusted and well-established ones such as Google, Twitter or Yahoo, enabling users to log in to the IDE with a popular community site account of their choice, without having to manage yet another set of user credentials.

From the point a user registers via FPish to the IDE, a user profile is created to hold various bits of data about that user and the files belonging to that user, collectively called workspaces. A workspace holds all files and folders of a particular solution: essentially all artifacts that with external assets, globally accessible via URLs, make up the application under development.

The IDE provides a rich set of functionality to manage workspaces. First, users can create empty workspaces and add files and folders to it at will, or remove workspaces from their account permanently. Workspaces can also be created from sources hosted in various external source control repositories, or from zip files uploaded by the user. At the time of writing, the IDE supports connecting to Git (and soon Mercurial) repositories, and updates can be committed to these from a given workspace. Users can also download the content of a given workspace as a zip file or import it into a new external repository.

[3] The FPish website.
<http://fpish.net>. (2012. 11. 05.)

[4] The Oauth website.
<http://oauth.net>. (2012. 11. 05.)

SOLUTION ORGANIZATION AND PROJECT TYPES

The basic unit of work the IDE fulfills is enabling editing F# and other source files. These can be added and removed from projects, a larger unit of work that encapsulates files that belong together. These projects can in turn make up entire solutions, e.g. a set of interdependent projects that are required to build and deploy a given application: the product of the solution. These closely mirror their .NET and Visual Studio counterparts, in fact such existing solutions can be imported into the IDE without any major surprises.

The IDE provides rich functionality for working with files and folders: users can create new files and folders, remove existing ones, move files between folders, and create entire structures of folders and files by adding or creating

new development projects based on various project and solution templates, similar to the same functionality in Visual Studio and other popular IDEs.

A workspace contains a single solution, which in turn can contain a number of interdependent projects of different kinds. At the time of writing, the IDE comes with project template support for most standard WebSharper project types: empty and sample web projects based on sitelets, ASP.NET, and ASP.NET MVC; WebSharper library projects, and empty C# web applications. This latter project type provides the bridge to legacy ASP.NET applications, enabling users to expand them with new, WebSharper-based web functionality and content.

While web projects and solutions are a clear priority for the IDE, we also are planning on adding various mobile application project templates. These enable users to develop mobile web applications for Android, Windows Phone, and iOS, browseable as web applications and with optional native packaging provided by the build automation found in these templates. With this option, upon successfully building projects of these types, users obtain a simple web page with links to download their mobile package binary. This is a handy way to install applications on actual mobile devices via URLs.

To further assist with working on mobile web projects, the IDE provides the ability to serve their web content in preset resolutions, conveniently picked from various resolutions of mobile device screens. With this feature, you can quickly test your mobile web applications as simple web applications running in fixed size containers within the IDE.

SYNTAX HIGHLIGHTING

Code files edited in the IDE come with standard syntax highlighting for the language involved. Syntax highlighting assists developers by marking the different types of tokens programs are made out of with different colors. This makes language keywords more apparent, among others, helping users to understand code quicker and easier. The IDE supports syntax highlighting of dozens of programming languages, and could in time enable the possibility to grow beyond .NET solutions and to expand developer coverage to more exotic development projects as well.

CODE ASSISTANCE

Using the F# compiler-as-a-service component of the IDE backend, the IDE can provide on-the-fly type information to the user as code is typed. This enables a development experience most developers are ac-

customized to using similar desktop IDEs such as Visual Studio, and includes features such as syntax and type checking, yielding red squiggles and error messages in a separate tool window that enables users to get a list of all issues with their application code.

The code service also exposes more detailed type information as well, and we are working on incorporating these into the IDE. Once this work is finished, users will enjoy full code completion and API browsing capabilities as well.

SOURCE CONTROL INTEGRATION

Distributed version control systems, such as Git and Mercurial, have become an essential tool in every developer's toolkit. The IDE supports Git, and Mercurial support is in the works. Users can connect their workspaces to Git repositories, and push their work back to these. The integrated Terminal facility (see below) provides a full suite of source control management (SCM) functionality, such as checking sources in and out, analyzing the current workspace for changes relative to the committed baseline, adding new tracked artifacts or removing existing ones, and splitting development work off to branches and shelves.

BUILD AUTOMATION

One of the main goals of the IDE is to provide off-the-client build automation, effectively freeing the developer to outsource the building activity to the backend. By scaling this functionality into the cloud, one hopes to get faster build (and deployment) times, effectively freeing the developer from having to wait around until compilation finishes locally.

The IDE can build just about any .NET project, provided that they fit the build scripts deployed in the backend. We are planning on providing various extensibility points and value-added build services as part of a larger IDE offering in the short future.

DEPLOYMENT

Next to building and producing binary files, the IDE currently supports deploying applications to AppHarbor, a service dedicated to deploying and running .NET-based web applications in the cloud. This gives a great way of deploying and serving WebSharper applications quickly without having to worry

[5] The AppHarbor website. <http://apphb.com>. (2012. 11. 05.)

about the details involved: all users need to do is to configure their AppHarbor account in their IDE profile. [5] While it is possible to deploy to AppHarbor manually using source control software and services, our IDE makes it even simpler to deploy directly by utilizing the AppHarbor web API. Next to AppHarbor, we are also planning on adding more deployment options in the near future.

TERMINAL

Developers often switch to various tools and utilities to perform different aspects of their coding activity, simply because these can't be easily catered to from the development environment they are using. Perhaps the most widely used such tool is a console or terminal window. A terminal window is a basic utility in any operating system, enabling the user to execute basic commands, often by chaining them into more complex tasks via so called scripts. These commands range from simple exploratory actions such as listing the files in the current working directory to more complex actions such as interacting with a source control management system that happens to host these files.

The IDE also features a terminal facility to enable precisely the same sort of tasks and actions. By allowing users to enter commands interactively and to immediately see their responses, users can develop a more refined sense of their development projects and their artifacts, and also “get things done faster” by removing the need to perform various user interface actions without having to use the mouse or having to navigate the menu system.

INTERACTIVE DEVELOPMENT

Similar to the terminal facility, the IDE also provides an interactive shell to enable users to explore their code as it is being developed. This feature closely mimics the functionality of the F# Interactive, a handy tool in the F# distribution that provides a Read-Evaluate-Print-Loop (REPL) interface for typing in and experimenting with F# code. As with the F# Interactive, users in the IDE can highlight code blocks from their code in the active editor tab, and send them to the interactive shell for evaluation. They can also type code directly in the interactive shell, which provides facilities for multi-line input and a language selector as well.

Interactive development is a great way to reduce the effort and time required to implement functionality by enabling developers to test their code on the fly. This results in changes, fixes applied to the code almost immediately, before proceeding onto other building blocks, thus establishing a higher fidelity and trust in the code being developed.

Interactive development also helps to enable Test-Driven Development (TDD), a popular agile methodology to document “tests first”, by putting testing at the micro level at the developers main focus.

Implementation

In this section we outline the main challenges and design decisions in four core areas: exposing the F# compiler as a service, enabling efficient client-cloud communication, integrating with source control systems, and providing a uniform user interface and experience.

EXPOSING THE F# COMPILER AS A SERVICE

The F# compiler sources, including essential Visual Studio integration such as interactive type checking and code completion, are available as open source code. We were able to reuse this code and adapt it to our needs to create an F# code service to perform various code-related chores, hosted in the cloud. The main challenge with this service included ensuring that code is run in a sandboxed environment, with limited privileges for dynamic user-generated code.

CLIENT-CLOUD COMMUNICATION

The choice of the HTML5 WebSockets technology for communicating between the browser and the Microsoft Azure cloud was motivated by a requirement to minimize latency for essential services such as code completion. We explicitly ruled out using software other than JavaScript in the browser in the interest of maintaining maximum portability. Our remaining options included using plain web services built on top of HTTP requests or using some form of long polling over HTTP.

We opted for WebSockets, in spite of the disadvantage of having no support in older browsers. One factor turned out to be critical in the cloud environment, namely, maintaining connection state. Once

[6] The HTML5 WebSockets API. <http://www.w3.org/TR/2009/WD-websockets-20091029>. (2012. 11. 05.)

established, a WebSocket connection acts as a bridge between a browser and a single worker machine selected by a load balancer. [6] The session is stateful on both ends, allowing maintaining context efficiently and locally to the worker machine. In our case, the context includes expensive to create in-memory F# compiler service state. In contrast, traditional web services and HTTP long polling workarounds periodically re-establish the connection. To work with these, we would need a custom routing mechanism to undo the effect of the Azure load-balancer and route client-server communication to the correct server within the cloud. While possible, we judged the additional latency and complexity of this solution not worth the portability cost of using WebSockets.

The WebSocket protocol is specified in terms of opaque binary or text messages. To make it easy to use in a typed setting, we developed a library of combinators for defining serializers and typed protocols layered on top of the untyped baseline protocol. With the help of this library, it is easy to construct bi-directional, fully asynchronous, typed services, working in both .NET and JavaScript thanks to WebSharper. Here is an example of an interface type and a matching contract defined using the combinators that simply follow the structure of the type:

```
/// Represents the client interface for the F# compiler service.
```

```
type FSharpClient =
```

```
{
```

```
    /// Handle FSI update.
```

```
    HandleInteractiveResponse : string -> unit
```

```
    /// Handle compiler message updates.
```

```
    UpdateCompilerMessages : CompilerMessageUpdate -> unit
```

```
}
```

```
[<JavaScript>]
```

```
let FSharpClient : C.Contract<FSharpClient> =
```

```
(
```

```
    C.Part (fun x -> x.HandleInteractiveResponse) (C.Send F.String) <<
```

```
    C.Part (fun x -> x.UpdateCompilerMessages) (C.Send CF.CompilerMessageUpdate)
```

```
)
```

```
|> C.Product (fun x y ->
```

```
{
```

```
HandleInteractiveResponse = x
UpdateCompilerMessages = y
})
```

With the contract definition in place, the system is able to build a proxy object of type `FSharpClient` to use on the client, and build a server given an implementation of `FSharpClient`. Contracts also compose, as is expected of combinator libraries.

[7] The
IKVM.NET
website.
<http://www.ikvm.net>.
(2012. 11. 05)

SOURCE CONTROL INTEGRATION

In order to integrate source control systems into the IDE, we experimented with running third-party Java code, most notably the JGit distributed version control system, in a managed .NET environment by cross-compiling it with IKVM. [7] This turned out to be a reasonably straightforward and robust approach, and we were quite happy with the results. Once the main Git hooks were established, we added the regular Git commands to the Terminal facility, and also hooked in some of the source control functionality into the workspace explorer facility.

UNIFORM USER INTERFACE

User experience is paramount in web applications. For our IDE, we used a number of Ext JS components to maintain a unified, efficient, and modern user interface. We also integrated a mature, capable source text editor component, CodeMirror, and adapted it to work with F# by creating a WebSharper extension for it.

Much of the IDE user interface has been designed with Ext JS Architect, a complementary tool to Ext JS that enables developers to design user interfaces using a drag-and-drop style WYSIWYG editor, automating the JavaScript code generation behind the scene. The resulting scripts were integrated into the main IDE application by ensuring that the appropriate scripts are loaded on demand. While these scripts dynamically create certain parts of the user interface, many of the generated elements must be accessible to the rest of the F# code base in order to bind event handlers to them (say, to implement the functional-

[8] The Sencha Ext JS website. <http://www.sencha.com/products/extjs>. (2012. 11. 05.)

[9] The CodeMirror website. <http://codemirror.net>. (2012. 11. 05.)

[10] The Sencha Architect website. <http://www.sencha.com/products/architect>. (2012. 11. 05.)

[11] The jQuery website. <http://jquery.com>. (2012. 11. 05.)

ity of various button clicks) [8] or to fill them up with actual pieces of data (say, to initialize a tree view control responsible to show the workspace hierarchy). [9] This binding is currently implemented in a loosely coupled manner by dynamically looking up the desired [10] DOM elements using primitive WebSharper calls to jQuery [11], and reinterpreting them as various typed values that have the desired signature to implement the necessary functionality, and doing the rest in type-safe F# code. We are also experimenting with providing a stronger form of type safety for working with these generated JavaScript fragments, similar to how we can bind to presentation layer HTML templates dynamically, yet with more coding guarantees.

Experience with Microsoft Azure

Overall, we had a positive experience with working with Microsoft Azure. Although designed primarily for C#, the cloud platform has no trouble executing F# and WebSharper projects, and Visual Studio integration with a cloud emulator makes it easy to get started. Azure comes with useful concepts of Web and Worker roles, a distributed file system with strong consistency guarantees and optimistic concurrency called the Blob Storage, a simple distributed database (Table Storage), and optional support for traditional SQL databases. Our F# compiler services run as a WebSocket server in a Worker role, and file system manipulation such as source control operations is performed locally to the instance associated with the user session, in the temporary file system, greatly reducing latency. We utilize the Blob service heavily for permanent storage of user workspaces and for coordinating between processes, and we use Table storage for searchable user and workspace metadata.

Our wish list for Azure includes faster deployment and better built-in support for diagnostics. Currently deployments of the IDE project to the cloud can take up to 10 minutes, and while there is a faster update mechanism for the Web role, we failed to find any quicker way to update a Worker role instance. This, combined with not-so-straightforward logging and diagnostics features contributed to long and painful debugging sessions when we experienced silent failure of an emulator-tested application deployed to the cloud. Thankfully, the issue turned out to be a fairly specific to working with the F# compiler. For general use, testing in the emulator should be sufficient to assure correctness.

Experience With F# and WebSharper

IntelliFactory has been the first commercial user of F#, and to this day we consistently use F# for most projects. We developed and are maintaining WebSharper specifically to be able continue using F# for web programming projects. Our commitment comes from a firm belief in the advantages of functional programming and the strongly typed discipline. While we are open to using other languages such as Haskell, Scala, and OCaml, F# continues to be our tool of choice. F# has a combination of features that work well at both small and larger-scale programming, and has the decisive advantage of excellent industrial support and backing by Microsoft.

Our use of F# and WebSharper in the IDE project has mostly followed the standard advice on avoiding mutation unless necessary, structuring code into independent components with clear-defined interfaces. There have been few surprises, but we did learn a few new lessons during the course of the project. We discuss here two observations that we intend to apply to similar projects in the future: one on techniques for reducing the editing cycle, and one the strategy for structuring code at a large scale.

The already suboptimal edit-compile-run-test F#/WebSharper cycle becomes even slower when using the Azure emulator. This led us to explore various techniques for reducing the need to recompile projects. In particular, we found a more suitable way to work with the presentation layer such as HTML templates and Ext JS definitions generated by Sencha Architect. For the HTML templates, we wired a simple HTML compiler that expands templates dynamically in the JavaScript environment, so that changes to the HTML can be immediately tested with a single page refresh, and not with a project recompilation. The Ext JS definitions share this property as they come as an external JavaScript file. A compilation pass is still necessary to maintain consistency guarantees with the typed F# code, however a key observation here is that a lot of small-scale presentation tweaks do not change the interface as seen by typed F# code and are therefore safe to bypass the recompilation pass. To ensure safety on recompilation, we wrote a small HTML compiler that exposes the HTML template interfaces as typed F#, and are in the process of making a similar compiler for the designer-generated Ext JS definitions. These mini-compilers are likely to make it into subsequent releases of WebSharper.

Our second observation involves ordering code at a large system level. A frequent complaint for new F# users is the F# compiler constraints on source code ordering. Programmers with an object-oriented programming background are used to structuring large projects as classes organized into namespaces, without any regard for declaration order. In F# this is not possible. While smaller F# projects have a natural ordering and this problem never arises, we did find while developing the IDE that sometimes having multiple co-recursive components without an a priori ordering can be an advantage.

This is primarily the case for large systems where relationships between the components are not well understood at the planning stage and are frequently changing during development. The freedom is easily recovered in F# by a simple trick of pre-declaring component interfaces, which can be done in a single F# file:

```
type IComponent1 =  
    abstract member Component2 : IComponent2 with get, set  
  
and IComponent2 =  
    abstract member Component1 : IComponent2 with get, set
```

Subsequent component implementations can be then declared independently and split into several files without any regard to ordering, and the application entry point is able to tie all components together.

Ongoing and Future Work

While the original motivation for a cloud IDE is providing the benefits of zero-install and instant deployment, a cloud environment opens up more interesting possibilities. One is instant developer collaboration based on the ability to simultaneously edit and interact with code. This feature is available in some of the competing products such as Cloud9, and we are working to add it to our IDE as well.

Another feature we are working on is extended language support. In particular, we plan to allow TypeScript editing in the IDE. TypeScript is a promising new language from Microsoft designed to be a superset of JavaScript supporting a form of type checking and code completion, and in this way better suited to large-scale application development. While we believe F# to be a better-designed general-purpose programming language that fits the same niche when you cross-compile F# code to JavaScript with WebSharper, TypeScript has some advantages. The TypeScript compiler and interactive code completion is able to run inside the browser JavaScript environment, offering outstanding latency for interactive editing. Also, with its Microsoft backing and language features such as a structural type system that makes it a close match to JavaScript, TypeScript has a potential to attract a large user base.

Active Directory alapú egységes informatikai azonosítási-rendszer bevezetése a Dunaújvárosi Főiskolán

Összefoglaló: Számos kutatás foglalkozik a heterogén informatikai környezetekben olyan autentikációs és autorizációs megoldások kialakításával, amely biztosítja a központi felhasználói adminisztrációt az informatikai infrastruktúra egészére, támogatja a számos alkalmazás jogosultságok szerinti használatát, valamint az infrastruktúra üzemeltetés speciális hozzáférési feltevéleit. A DF-en kialakított AAI-infrastruktúra az Active Directory-címtár funkcionalitására épülő olyan köztesréteg-alkalmazás kifejlesztését tartalmazza, amely magába foglalja a belső és külső azonosítást és jogosultságkezelést igénylő rendszerek kiszolgálását is. Illeszkedik az NIIF föderatív AAI-infrastruktúrájához [1], könnyen konfigurálható bármely IdP/SP-modell adatigénye szerint. Az adatkezelés szempontjából jól definiált eljárások mentén működik. A rendszerkonceptió arra épül, hogy az intézményben Microsoft Windows Server a legtöbb alkalmazás kiszolgálója. A címtár szigorúan a legmegbízhatóbb elsődleges adatforrás a Neptun-rendszer alapján épül fel. Az oktatók/dolgozók személyi és munkaügyi adatait kizárólag a HR és Jogi Osztály, a hallgatói adatokat a Tanulmányi Hivatal kezeli. A köztesréteg-alkalmazás (Middleware) integrálja a címtár adatait a Neptun tanulmányi rendszerben tárolt információkkal, így a kialakított rendszer nem csak felhasználók azonosításra, hanem egyes adatszolgáltatási funkciók ellátására is alkalmas (pl. telefonkönyv, órarendi információk, teremterhelés, stb.)

Kulcsszavak: Felhasználói azonosítás, jogosultságkezelés, címtár, Active Directory, AAI, föderatív azonosítás kezelés, köztesréteg alkalmazás, metadirectory.

* Dunaújvárosi Főiskola,
Informatikai Szolgáltató
Központ
E-mail: cs.kovacs@mail.duf.hu

** Dunaújvárosi Főiskola,
Informatikai Szolgáltató
Központ
E-mail: szabo.jozsef@mail.duf.hu

[1] Nemzeti Információs Infrastruktúra Fejlesztési Intézet
AAI Wiki <https://wiki.aai.niif.hu/index.php?title=AAI>

Abstract: Several researches are being conducted in order to develop authentication and authorization solutions for heterogeneous IT environments. Solutions that would provide centralized user administration with privilege driven application running and special access features required by system management for the entire infrastructure. The so called AAI infrastructure developed in DF is a middleware application that builds on top of Active Directory functions and it serves such systems that require internal or external authentication along with privilege handling. It fits in the NIIF identity federation infrastructure perfectly furthermore it is easily tailorable to the data requirement of any IdP/SP model. From the data handling point of view the application operates by clearly defined procedures. The system concept is based on the fact that at DF most of the applications are being served by Microsoft Windows Server operating systems. The institute's Active Directory relies strictly on the most trusted primary data source, the NEPTUN system. Personal and work related information of teachers and other employees are handled solely by the Human Resources department while at the same time the Student Office is responsible for the accurate input of student information. The above mentioned middleware application integrates the active directory's data with information stored in the NEPTUN, thus creating a system which is no longer for user authentication only, but it also supports several data providing functions, like an electronic phonebook, schedule information, classroom usage and many more.

Keywords: Authentication, authorization, Active Directory, Federated Identity Management, middleware.

Előszó

Csak a legutóbbi időkben lett esélye jó néhány felsőoktatási intézménynek, hogy informatikai rendszereit professzionális technológiák és átgondolt folyamatok mentén szervezze. Ezt a lehetőséget főként az infrastruktúrafejlesztésekre rendelkezésre álló európai pályázati források biztosították, valamint az a tény, hogy az intézmények számára is egyre inkább nyilvánvalóvá vált az, hogy a működési folyamataikat korszerű, biztonságos és megbízható informatikai infrastruktúratámogatás nélkül képtelenek fenntartani. A Dunaújvárosi Főiskola (DF) számára az első komoly informatikai infrastruktúrafejlesztés 2004–2006-ban a HE-FOP pályázati körben valósult meg. Az elmúlt időszakban pedig az Új Magyarország Fejlesztési Terv TIOP és TÁMOP operatív programjai keretében kapott lehetőséget az infrastruktúrafejlesztésekre. A fejlesztés másik mozgatója az olyan intézményi szemlélet, amely az informatikai szolgáltatást központi erőforrásnak tekinti és más kritikus működési célokkal azonos szinten kezelve befogadja, sőt egyre inkább igényli is a korszerű technológiák alkalmazását. El kell mondani azonban, hogy míg a beruházások megvalósítására, korszerű eszközök és alkalmazói rendszerek bevezetésére még kerül megfelelő forrás, addig az üzemeltetésre, az üzemvitel feltételeinek maradéktalan biztosítására, fenntartására gyakran vagy nincs elegendő

fedezet, vagy a megfelelő szintű üzemviteli feltételek biztosítása jóval nagyobb energiát igényel, mint az elfogadható lenne.

A Dunaújvárosi Főiskola a TIOP 1.3.1 pályázati komponensben biztosított fejlesztési lehetőséget kihasználva egy korszerű európai campus informatikai igényeit minden tekintetben kielégítő, gazdaságos üzemviteli feltételeket igénylő ugyanakkor a korszerű infokommunikációs technológiák számára hosszabb távon is választ adó infrastruktúrát alakított ki, melynek egyik fontos elemét a felhasználói azonosítás és jogosultságkezelés egységes rendszerét kívánjuk bemutatni.

Kiinduló állapot

Az informatikai rendszerek evolúciós fejlődése hozza magával azokat a problémákat, amelyek a későbbi hatékony működés, működtetés visszatartói. A heterogén operációsrendszer-környezet, az eseti szoftverfejlesztések, az egymással integrált kapcsolatban nem lévő alkalmazások használatát külön-külön kell tanulni, a hozzáféréseket egyenként kell kezelni. Az egyes alkalmazások felhasználói adminisztrációja nehézkes, nem megbízható és nem koherens. A felsőoktatási térben való megjelenés által igényelt mobilitási szolgáltatások megbízható felhasználói azonosítást és egységes áttekinthető jogszabályi feltételeknek megfelelő adatkezelést igényelnek. A rendezetlen állapotok fennállása mellett ez utóbbi cél semmiképpen nem teljesíthető.

2006 és 2010 között a DF felhasználó-kezelése a központi irányítás alatt álló informatikai rendszerekben megoldottnak számított, ugyanakkor a legfontosabb alkalmazások egységes azonosítás-kezelésére koránt sem rendelkezünk megfelelő megoldással. A kulcsrendszerek esetében (hallgatói nyilvántartás, gazdasági rendszerek, iktató- iratkezelő rendszer, központi dokumentumtár, intranet, hálózati hozzáférések, stb.) egymással nem feltétlenül megbízható szinkronizálási mechanizmussal működő informatikai címtár állt rendelkezésre:

- Az Egységes Tanulmányi Rendszer (Neptun), amely elsősorban a hallgató entitások elsődleges felhasználói adatbázisaként funkcionált. Saját alkalmazásszintű azonosítási rendszere van.
- A Novell eDirectory LDAP-alapú címtárszolgáltatása, amely az intézményi oktató/dolgozók adatainak elsődleges címtára – offline szinkronizációval a hallgatókat is tartalmazta – a levelezési rendszerben, a központi fájlrendszer alapú dokumentumtárolásban és az intranet szolgáltatásokhoz való hozzáférésben volt elsődleges.
- Az NIIF által létrehozott X.500 alapú LDAP-címtárrendszer, melyhez a DF korán csatlakozott annak reményében, hogy az identitás-föderációs szolgáltatásokat minél szélesebb körben tudja igénybe venni. Az LDAP a névtárprojekt keretében elkészült Neptun-szinkron futtatással volt képes átvenni a hallgatói

adatokat a tanulmányi rendszerből, egy saját fejlesztésű alkalmazás segítségével pedig a dolgozói adatok szinkronizálása a Novell eDirectory-ből is megoldott volt.

- Active Directory (AD) a Windows operációs rendszerek hatékony üzemeltetése miatt szükségszerűen már létezett ebben az időszakban, de felhasználói azonosításra csak a rendszerfelügyeletet végző dolgozók számára és a kritikus rendszereket (tanulmányi rendszer, gazdasági rendszer) futtató munkaállomásokra való belépésre, a munkaállomások központi adminisztrációja érdekében került felhasználásra.
- Egyéb, nem címtár alapon működő felhasználó kezelés, pl. VPN, hálózatmenedzsment-eszközök és -alkalmazások kezelése – általában önálló felhasználói adatbázissal rendelkeztek.

Az 1. táblázat bemutatja azt a bonyolult belső azonosítási platformot, amely a négy címtárrendszer és a fontosabb alkalmazások felhasználó-kezelését biztosította.

1. táblázat.

Az informatikai rendszerek és az alkalmazott címtárak összefüggései a fejlesztés megkezdése előtt.

Rendszer / címtár-szolgáltató	NEPTUN	Novell eDirectory	NIIF központi címtár	Microsoft Active Directory	Egyéb
Oktatói/hallgatói Neptun-elérés	Neptun-rendszer felhasználó kezelése	-	-	-	-
Tanulmányi adminisztráció (Neptun)	Neptun-rendszer saját felhasználó kezelése	-	-	Domainbe léptetett munkahely (PC)	Nem domain felhasználóknak VPN-kapcsolat felépítése szükséges Cisco VPN-kliens, CS-ACS-authentikátor, saját felhasználó kezeléssel

Oktatói/dolgozó email-rendszer (POP, IMAP, Webmail)	-	Novell saját eDirectory alapú azonosítás	-	-	-
Hallgatói email-rendszer Webmail	-	-	-	-	Saját regisztrációra épülő felhasználó kezelés
Gazdasági rendszer (TÜSZ)	-	-	-	Domainbe léptetett munkahely (PC)	Nem domain felhasználóknak VPN-kapcsolat felépítése szükséges Cisco VPN-kliens, CS-ACS-authentikátor, saját felhasználó kezeléssel
Egységes tantárgyi adatbázis (ETA), e-learning-rendszer (MOODLE)	-	-	-		Sajátfelhasználó kezelés
Iktató rendszer (MonDoc)	-	-	-	Web autentikáció ADIR interfésszel, vagy Domainbe léptetett munkahely(PC)	-
Központi dokumentumtár	offline hallgatói adatszinkronizálás	Novell saját eDirectory alapú azonosítás	-	-	

Intranet/extranet	offline hallgatói adatszinkronizálás	Web autentikáció LDAP interfészen	-	-	-
Vezetéknélküli hálózat elérése (EDUROAM, saját)	-	Offline szinkron az eDirectory-val	Freeradius LDAP interfésszel	-	-
Desktop-adminisztráció	-	-	-	Domain-be léptetett munkahely (PC), vagy local felhasználó	
Rendszerfelügyelet (hálózat, szerverek, alkalmazások)	-	-	-	Domain-be léptetett munkahely (PC)	VPN-azonosítás szükséges Cisco VPN-kliens, CS-ACS-authentifikátor, saját felhasználó kezeléssel
Föderatív szolgáltatások (hálózatmenedzsment, VoIP, videokonferencia, stb.)	-	-	Offline szinkron az eDirectoryval	-	-

Egyértelmű volt, hogy a fenti rendszer hosszabb távon nem tartható fenn. Meg kell találni azt a központi elemet, amely mind az oktató/dolgozó, mind pedig a hallgatói entitások azonosításához megfelelő naprakészséggel és megbízható információ-tartalommal rendelkezik. Erre a célra a Windows Active Directory (AD) címtárszolgáltatását jelöltük ki.

A rendszertervezés fázisában derült ki, hogy a native AD-kapcsolat néhány alkalmazás esetében nehezen megoldható, illetve számos olyan entitás-attribútum kezelésére van szükség, amelyeket nem célszerű az AD-sémába bevonni. Emiatt egy köztes alkalmazás kifejlesztésére volt szükség, amely a címtárral meghatározható rövid időközönként kerül szinkronizálásra.

A köztesréteg feladata olyan specifikus szolgáltatások biztosítása, amellyel akár webalkalmazásokat (nyilvános intézményi telefonkönyv, webes órarend, Service Providerok speciális adatigénye, stb.) is ki lehet szolgálni.

A választás okai:

- Az intézményben működő kritikus alkalmazások döntő része Windows Server operációsrendszer-környezetben működik.
- Már előzőleg kialakításra került az Active Directory-rendszer, amelyet korlátozottan igénybe is vettünk.
- A főként Windows operációsrendszerekkel működő intézményi PC-park (~1200 db) központi desktop menedzsmentjének, vírusvédelmi rendszerének, operációsrendszer-frissítések központi kezelésének, egyes autorizációs feladatok ellátásához nélkülözhetetlen eszközének bizonyult.
- A Cisco technológiára épülő hálózatfelügyeleti és rendszeradminisztrációs azonosítási feladatok natív módon AD alapon működőképesek.
- Nem utolsó sorban, rendelkezünk olyan szakértelemmel, amely a Windows szerverkörnyezet üzemeltetését, biztonságos menedzselését, kellő szinten biztosította és a szolgáltatások kialakításának professzionális megoldására garancia volt.

Közben az NIIF intézet is elindította a felsőoktatási, kutatási kör számára AAI-projektjét, amely a központi címtár-rendszer korszerűsítését, IdP/SP szolgáltatási modellbe helyezését oldotta meg. A létrejött identitás-föderáció biztosítja a felsőoktatási térség szereplőinek bizalmi elven történő azonosítását oly módon, hogy az egyes intézmények saját dolgozóikat önmaguk azonosíthatják. A föderatív szolgáltatások DF-en működő Identity Providere (IdP) szintén a kifejlesztett köztesréteg-alkalmazás adatbázisából és az AD-ből nyeri az azonosítási – és ahol szükséges – jogosultságkezelési információkat.

A központi címtárkezeléssel az intézményre háruló adatkezelési feladatokat lényegesen rendezettebben az adatkezelési szabályoknak megfelelően tudjuk végezni.

Választott eszközök

Az előzőekben ismertetett problémák (feladatok) összetettsége miatt a megoldás nem (vagy csak nagyon nehézkesen) oldható meg „dobozos” termékekkel. A cél az volt, hogy olyan eszközt keressünk,

- amivel elérhetőek a kitűzött célok,

- lehetőleg nem igényli újabb licenszek beszerzését,
- a felhasznált technológiára vonatkozó ismeretek rendelkezésre állnak, vagy lehetőleg kis ráfordítással megszerezhetőek (a fejlesztők és üzemeltetők részéről),
- és a választott megoldás segítse elő az intézményi homogenitást.

A CÍMTÁR PROBLÉMA

A címtárak alkalmasak arra, hogy adott objektumokat adott attribútumokkal központilag tartsanak nyilván. Az objektumokat és azok szerkezetét a séma írja le. A séma igény szerint bővíthető, de ez a művelet nem mindig visszavonható. Ez azzal járhat, hogy a séma és ezzel együtt a címtár egy idő után olyan attribútumokat tartalmazhat, amelyekre már nincs szükség (garbage). Ezért egy olyan környezetben, ahol a séma „folyamatosan” változhat, célszerű más megoldást keresni.

Ezen felül a címtár nem relációs adatbázis. Az 1:N-kapcsolat (pl. szótár táblák) nem jól megoldott, pedig erre az adatok konzisztens állapotban tartása miatt nagy szükség lehet.

A fentiek alapján egy olyan megoldás rajzolódik ki előttünk, ahol a címtár funkcionalitását nem séma-bővítéssel, hanem egy relációs adatbázissal egészítjük ki. Ebben az esetben lehetőség lesz arra, hogy a címtárban alapértelmezésben is létező tulajdonságokat rögzítsük, kihasználva a szabványos címtár-funkciókat és sajátosságokat, a további adatokat pedig egy relációs adatbázisban tároljuk. Ez a kialakítás megoldja a séma-bővítéssel és a konzisztenciával kapcsolatos problémákat is.

A fenti megoldás viszont magával hoz egy újabb megoldandó feladatot: mivel két különálló rendszerünk van (címtár és adatbázis), így meg kell oldanunk a két rendszer összekapcsolását, így biztosítva a konzisztens állapotot. Továbbá a felhasználók (személyek, alkalmazások) számára el kell rejteni a rendszerben lévő kettősséget. Ezt úgy lehet megtenni, hogy olyan saját alkalmazást (alkalmazásokat) fejlesztünk, amelyek a felhasználók számára transzparensé teszik a rendszer használatát.

A fentiek figyelembe vételével az alábbi eszközök kerültek kiválasztásra:

- Címtár: Microsoft Active Directory.
- Adatbázis: Oracle RDB.
- Fejlesztői környezet: Visual Studio 2010 és .Net 4.0.

MIÉRT ACTIVE DIRECTORY?

Az intézményben már üzemelt Active Directory (AD) címtár. Ez a project kezdetén csak korlátozottan (csak bizonyos feladatokra) került felhasználásra. A EU-s pályázatokban több Microsoft technológia is

szerepelt (Exchange, SharePoint), amelyek kifejezetten igénylik az AD-t. Így az ezekkel megvalósított alkalmazások kiszolgálására az eDirectory már nem alkalmas. Az X.500 alapú LDAP-interfész alkalmas az NIIF központi címtár kiváltására. Ezzel együtt az erre a címtárra épülő szolgáltatások is tudják az AD-t használni.

Sokáig úgy tűnt, hogy a Neptun kivételével a többi három címtár integrálása megoldható az AD-ba. A Neptun eredetileg nem címtárból, hanem saját felhasználói adatbázisból végezte az azonosítását. A project elején egyeztetések történtek az SDA Stúdió Kft.-vel, aminek eredményeként az Neptun ETR fejlesztői elkészítettek egy olyan modult, ami a felhasználói azonosítást nem saját adatbázisból, hanem az AD-ból végzi. Így már nem volt akadálya annak, hogy a tanulmányi rendszer autentikációs funkcióit is az AD-ba integráljuk. Ezen felül az AD mellett szólt az is, hogy az intézményben használt kliensoperációs-rendszerek döntő része is valamilyen Microsoft termék (XP, Windows 7). Az AD nem csak címtárszolgáltatást nyújt, komoly központosított menedzsment feladatok ellátására is alkalmas (pl.: csoportházirendekeken keresztül). [2]

MIÉRT ORACLE DB, VISUAL STUDIO 2010 ÉS .NET 4.0?

Az intézmény rendelkezik Oracle DB szerver-licensszel. A Neptun DB ezen az Oracle szerveren fut, így a két rendszer közti kommunikáció könnyebben megoldható (gyakorlatilag „házon belül”).

Ezen felül az Oracle DB-vel kapcsolatban már voltak fejlesztési tapasztalataink. [3]

A Visual Studio 2010 és a .Net 4.0 egy számunkra jól használható fejlesztőrendszer, ami képes az AD-val és az Oracle DB-vel is kommunikálni. Továbbá ezzel az eszközzel a szükséges felhasználói felületeket (GUI) is relatíve könnyen el lehet készíteni. [4, 5]

[2] Active Directory Domain Services
[http://technet.microsoft.com/en-us/library/cc770946\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc770946(v=ws.10).aspx)

[3] Primary Author: Sumit Jeloka Oracle Data Provider for .NET Developer's Guide 11g Release 1 (11.1.0.7.20) E15167-01 Copyright © 2002, 2009. Oracle and/or its affiliates.

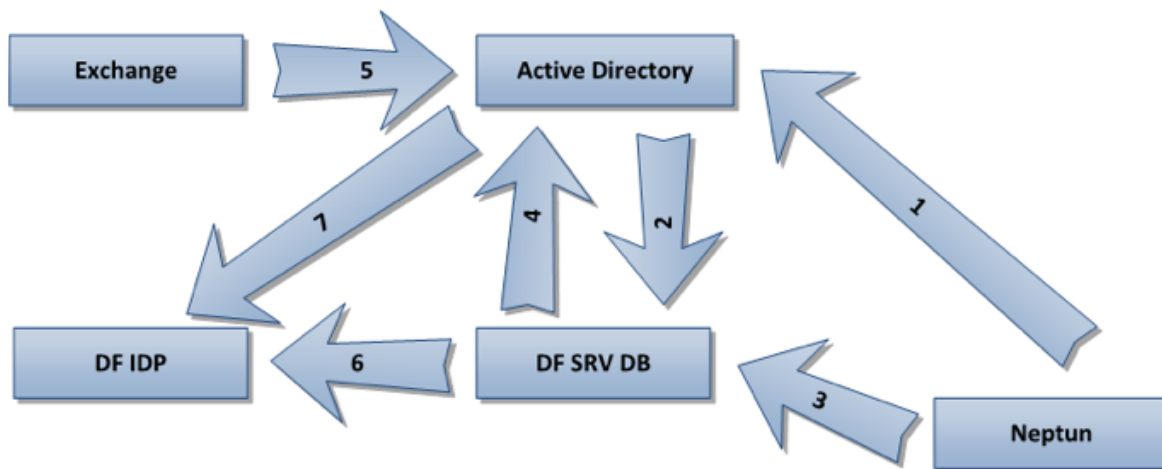
[4] P. Cibraro–K. Claeys–F. Cozzolino–J. Grabner: *Professional WCF 4: Windows Communication Foundation with .NET 4* Indianapolis: Wiley Publ.

[5] Árvai Z.–Csala P.–Fár A. G.–Kopacz B.–Reiter I.–Tóth L.: *Silverlight 4 A technológia, és ami mögötte van – fejlesztőknek*. HTML 5 ismertetővel bővített kiadás Készült a Devportal.hu közösség támogatásával <http://devportal.hu/TechCenters/TechCenterPage.aspx?id=2>

Megvalósítás

Az ismertetett eszközökkel az alábbi integrált rendszer került kialakításra, kiegészítve a közben lefutott mail-migrációs projekt eredményeként belépett Exchange-el.

1. ábra. Adatáramlás az integrált rendszerben.



A rendszerben az adatáramlást az alábbi folyamatok biztosítják:

- Új dolgozó és hallgató felvétele a rendszerbe csak a Neptunon keresztül történhet. Az így rögzített személyek a Neptun à AD szinkron-modulnak köszönhetően azonnal átkerülnek az AD-ba. A szinkronizáció a Neptunban végzett adatmódosítás esetén is végbemegy. A szinkron egyirányú (a Neptunba nem történik visszaírás).

Továbbá, csak az SDA-interfész által meghatározott attribútumokat tölti át a rendszer. (1)

- Az AD-ban lévő felhasználókat egy időzített futású (és kézzel is indítható) szerviz frissíti a köztesréteg adatbázisban (DF SRV DB). (2)

- Mivel a Neptun à AD-szinkron, csak az SDA által megadott attribútumokat tölti át, így a szükséges további attribútumokat a köztesréteg-alkalmazásnak kell áttölteni a saját adatbázisába. A szinkron alapján eldől, hogy a felhasználó hallgató és/vagy dolgozó, ennek megfelelő adatkezelés történik. (3)

- Szükség esetén bizonyos dolgozói adatok visszaírása kerülnek az AD-ba. Ezek tipikusan olyan adatok, amelyeket pl.: az Exchange használ. Ha a dolgozó több szervezeti egységben is megjelenik, akkor a preferált szervezeti egység adatai kerülnek át. (4)
- Amennyiben a dolgozó megkapta az intézményi email címét, úgy az ehhez kapcsolódó adatokat az Exchange tölti át az AD-ba. (5)
- A föderatív felhasználói azonosításrendszeréhez való kapcsolódás érdekében az Identity Provider a címtárból autentikál és a köztesréteg-alkalmazáson keresztül nyeri ki a Service Provider számára szükséges további adatokat. (6, 7)

Eddig megszerzett tapasztalataink szerint a fenti rendszer már alkalmas az intézményben felmerült komplex igények kielégítésére, továbbá képes adatkezelési szabályoknak megfelelő módon adatokat szolgáltatni a föderáció számára is.

A fejlesztés eredményeként valamennyi alkalmazói rendszerben a felhasználói belépés azonos felhasználónév/jelszó párossal biztosított. Ez azonban óvatosságra is int, mert az illetéktelen információszerzés nyomán szélesebb azon adatok köre, amelyhez a támadó hozzáférhet. A nagyobb biztonság elérése érdekében legalább az oktató/dolgozói felhasználók számára a kétfaktorú azonosítás rendszerét lenne ideális bevezetni.

A Non-Invazív Mobil EEG-regisztráló és -jelfeldolgozó eszközök összehasonlítása

*DF rendszeradminisztrátor
E-mail: ujbanyit@mail.duf.hu

**DF műszaki tanár
E-mail: katonaj@mail.duf.hu

***DF főiskolai adjunktus
E-mail: kovari@mail.duf.hu

Összefoglalás: Az elmúlt években az EEG-jelek mérését és feldolgozását végző eszközök igen nagy fejlődésen mentek keresztül, melynek köszönhetően az alkalmazási lehetőségeik egyre több területre terjednek ki. E cikk ismerteti és összehasonlít néhány ilyen EEG-eszközt.

Kulcsszavak: Agyhullámok, elektroencefalogram (EEG), non-invazív, EEG-headset.

Abstract: In recent years, the measurement and processing of EEG signals have been improved very significantly, which enables the application of these devices in more and more areas. This article describes and compares some EEG devices.

Keywords: Brain-waves, electroencephalogram (EEG), non-invasive, EEG headset.

Bevezetés

A tradicionális EEG-jelrögzítés és azok elemzése sokáig csak az agyi elektromos aktivitás időbeli és térbeli tulajdonságainak a grafikus illusztrálására szolgált. Igen hosszú ideig kizárólag a regisztrátumok által meghatározható jellemzőket használták a kiértékelés során, majd a 20. század második felében a digitális korszak kezdetével a jelek kiértékelése, feldolgozása igen nagy változáson ment keresztül. Az EEG-jelek összetett valósidejű matematikai elemzése a digitális technika fejlődésével vált lehetővé.

A digitális jelfeldolgozás megjelenése előtt nagyon körülményes volt a grafikus kinyomtatott regisztrátumok kezelése, kiértékelése.

Ez azonban digitális jelfeldolgozó processzorok (DSP) megjelenésével és a

jelfeldolgozó algoritmusok továbbfejlesztésével az agyjelek kiértékelésére az addig alkalmazott módszerektől eltérő, új algoritmusok segítségével teljesen új kiértékelési módszerek kerültek alkalmazásra, melyek új és pontosabb eredményeket adtak. Az egyik legáltalánosabban alkalmazott algoritmus az úgynevezett kvantitatív EEG (QEEG), melyen alapuló vizsgálatok létrejöttéhez a számítógépes jelfeldolgozás nagyban hozzájárult. QEEG során a digitalizált EEG-jelek Fourier-transzformáció segítségével kerülnek kiértékelésre, mely átalakítás alkalmazásával a jelfrekvencia-összetevői, vagyis a frekvencia-spektruma egzakt módon meghatározható. [1] A technológia fejlődése lehetővé tette olyan mobil mérő- és feldolgozó-eszközök megalkotását, amelyek az orvostudományon kívül számos más területen nyújthat új kutatási lehetőséget. Napjainkban egyre nagyobb teret hódítanak a műtéti beavatkozást nem igénylő, non-invazív eljárást alkalmazó EEG-eszközök. Az EEG-eszközök agyi elektromos jelek érzékelésén és feldolgozásán túl egyéb funkcionalitást is biztosíthatnak a felhasználók számára, mint egy mobiltelefon headset, amelyek még univerzálisabbá tehetik ezen eszközök felhasználását.

EEG-jelek keletkezése

EEG-jelek keletkezésének modellezésére készített legkorábbi modell a Hodgkin and Huxley's Nobel-díjat kapott modellje, mely 1952-ben került publikálásra. A testünk egyik legnagyobb szerkezete az idegrendszer, amely neuronhálózatok csoportja. Ez a kapcsolatrendszer az egész testünket átszövi. Az idegsejtek képesek egymással kommunikálni, így lehetővé válik az agyból érkező ingerek közvetítése. Azt a kapcsolódási helyet, amelyen keresztül az ingerület egyik sejtről a másikra áterjed, *szinapszis*nak nevezzük. A szinapszist nagyobb részben az egyik neuron sejtteste (axonja) hozza létre a másik neuron idegvégződésével (dendritjével). Az ingerületet hozó idegsejtet preszinaptikusnak a továbbvivőt posztzinaptikusnak nevezzük. (1. ábra) [2]

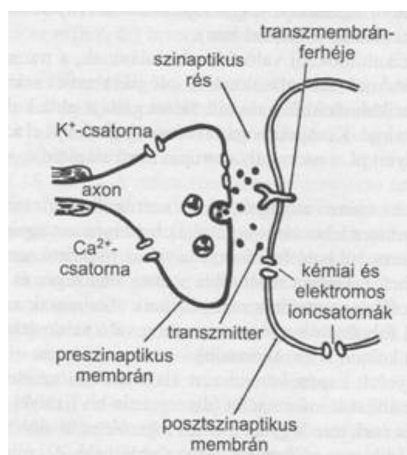
[1] Snyder, S. M.–Hall, J. R. (2006): A meta-analysis of quantitative EEG power associated with attention-deficit hyperactivity disorder. *Clin Neurophysiology*. Oct. Pp. 440–55.

[2] Saeid, S.–Chambers, J.A. (2007): *EEG Signal Processing*. Cardiff University UK: John Wiley & Sons.

[3] The FPish website.
<http://fpish.net>.
(2012. 11. 05.)

[4] The Oauth web-
site. <http://oauth.net>.
(2012. 11. 05.)

1. ábra. Szinapszis, ingerület-átvitel pillanata.



Forrás: <http://www.agr.unideb.hu/ebook/allatelettan/>

A neuronok elektromos potenciáljának összességét elektrofiziológiai aktivitásnak nevezzük. Amikor egy neurotranszmitter tüzelni kezd, a kémiai ingerületátvivő anyagok közvetítésével, a beáramló pozitív ionok depolarizálják az idegsejteket. Ezt a depolarizálást nevezzük posztoszínaptikus potenciálnak (PSP). A szinapszisok keletkezése akkor történik, amikor az egyik neuron axonja kapcsolatba kerül egy másik neuron dendritjével. A két transzmitter közötti részt szinaptikus résnek nevezzük. Az emberi agy közel 100 milliárd neuront tartalmaz. Amikor ezek közül a neuronok közül több ezer összehangoltan tüzelésbe kezd egy adott agyi területen, az mérhető feszültséget eredményez. Ezek a feszültségek az emberi fejbőrön mérhetőek. [3] Az EEG-elektrodákat megfelelően elhelyezve közvetlenül az emberi fejbőrön képesek vagyunk alacsony (5–100 μ V) elektromos potenciált mérni. Továbbá a váltakozó elektromos feszültségek és az általuk keltett áramingadozások következtében mágneses térerősség-változás is mérhető, amely szintén kimutatható az emberi fejbőrön.

Az elektrofiziológiai aktivitást két módon mérhetjük. Az első mérési módszer az úgynevezett invazív a második az úgynevezett non-invazív lehetőség. A non-invazív eljárás során egy EEG-berendezés segítségével a feszültség-változások vagy a mágneses tér változásai is rögzíthetőek. [4]

A legnagyobb hátránya ennek a módszernek, hogy a mért adatok zajosak. A zajok által keltett zavarjelek csökkenthetők invazív módszerrel, ahol az EEG-elektrodákat közvetlenül az agyba műtési eljárással ültetik be. Ennek következtében a mérés, és a mérés alapján kinyerhető információk pontossága nő [5]. A beültetéshez szükséges műtési eljárás kockázatai miatt ezt a módszert kevésbé alkalmazzák.

EEG-regisztráló és -jelfeldolgozó eszközök

Az agyjelek érzékelését, valamint továbbítását a jelek feldolgozását végző egység felé – esetleg azonnali feldolgozását – a fejre helyezhető EEG-jeleket mérő érzékelők, valamint ezen érzékelők jeleit feldolgozó, továbbító elektronikus egység segítségével valósítják meg.

Az elmúlt években bárki számára elérhetővé váltak olyan EEG-headsetek is, melyek az agyjelek érzékelésén túl tartalmazzák az érzékelt agyjeleket digitalizáló (esetleg továbbfeldolgozó, például: FFT meghatározását végző), valamint valamilyen digitális kommunikációs csatornán e jeleket továbbítani képes kommunikációs egységet. Ahhoz, hogy ezen eszközök minél mobilabban használhatók legyenek, ezért saját elemről vagy akkumulátorról történő energiaellátással rendelkeznek. A következőkben néhány non-invazív EEG-headset kerül ismertetésre, melyek egyedi fejlesztésű agyszámítógép-interfészek (BCI) megvalósítására is alkalmasak.

NEUROFOCUS

A NeuroFocus az első, több száraz típusú érzékelőt alkalmazó, vezeték nélküli EEG-eszköz (2. ábra). A fejlesztők a laboratóriumi alkalmazásokon túl már megkezdtek a kereskedelmi forgalomra szánt eszköz fejlesztését. A TOBI (The European Tools for Brain-Computer Interaction consortium) konzorcium a kutatásokhoz a NeuroFocus-t fogja használni EEG-regisztráló eszközként orvosi alkalmazási célú fejlesztésekre. [6]

A NeuroFocus EEG-szenzorai másodpercenként 2000-szer mérik az agyi aktivitást, majd az adatokat Bluetooth-kapcsolaton keresztül képes továbbítani. A vezeték-nélküliség nagy erőlelépést jelent a hagyományos EEG-eszközökhöz képest, melyek csak kórházi környezetben használhatóak, lehetővé téve ezen eszköz alkalmazását akár egyéb környezetben is. [7]

[5] Sawan, M. et al (2013): *Wireless Recording Systems: From Noninvasive EEG-NIRS to Invasive EEG Devices, Biomedical Circuits and Systems. IEEE Transactions.* Volume 7. Issue 2. Pp. 186–195.

[6] TOBI: *Tools for Brain-Computer Interaction:* www.tobi-project.org/ (2012. 06. 12.)

[7] Nielsen: *What People Watch:* www.nielsen.com/us/en/nielsen-solutions/nielsen-measurement/consumer-neuroscience.html (2013. 01. 24.)

[8] Jasper, H. H. (1958): The ten-twenty electrode system of the International Federation. *Electroencephalography Clin Neurophysiology*. 10: 371–375.

Az eszköz a nyers, mért adatokon túlmenően feldolgozott információkat is képes szolgáltatni.

Az eszköz fontosabb tulajdonságai:

- teljes agyi területre kiterjedő sokcsatornás EEG-eszköz;
- szabványos mérési pontokon történő mérés [8];
- adatokat vezetéknélküli átvitelen képes továbbítani;
- száraz elektródákat alkalmaz, így a szenzorok könnyen cserélhetőek;

Az EEG-headset hátránya, hogy egyrészt többcsatornás érzékelő kialakítású, amelynél az érzékelők a fej felső részén érintkeznek a fejbőrrel, ezért felhelyezése a megfelelő jelérzékelés biztosítása miatt körülményesebb.

2. ábra. NeuroFocus EEG-headset



Forrás: <http://www.neurofocus.com>

IMEC ÉS HOLST CENTRE ÁLTAL FEJLESZTETT EEG-HEADSET

A Holst Centre és IMEC 2009-ben mutatták be első EEG-fejhallgató prototípusukat, mellyel egy egyszerűen használható EEG-mérőeszközt alakítottak ki. (3. ábra) A headseten összesen 10 elektróda van a nemzetközi 10–20 szabvány szerint elhelyezve [8]. A Kereskedelmi EL120-as headsetben újrafelhasználható Ag/AgCl bevonattal ellátott elektródákat használtak, melyek speciális tervezésűek, ezzel biztosítva a megfelelő érintkezést hajon és szőrön keresztül is. Az elektródák speciális foglalatokban vannak rögzítve, melyek biztosítják a kellő mozgási szabadságot és a megfelelő jelátvitelt.

A technológia még fejlesztés alatt van, és a fejlesztés célja a séta közbeni EEG-adatok rögzítése. Ennek megvalósításához meg kell oldaniuk a mozgás és egyéb fiziológiai zavarok kompenzálását, a száraz elektródák és a hozzájuk tartozó áramkörök illetve az EEG-adatok feldolgozására alkalmas ultra-alacsony fogyasztású processzor kifejlesztését. [9]

Az EEG-rendszer egy az EEG-jelek mérésére alkalmas ultra-alacsony fogyasztású ASIC (Application Specific Integrated Circuit) áramkörre épül, amely nagy zajelnyomással és így alacsony zajszinttel jellemezhető. Az integrált EEG-mérő rendszer tartalmaz egy alacsony fogyasztású mikrokontrollert és vezeték nélküli átvitelt megvalósító egységet, lehetővé téve a helyi adatfeldolgozást és vezeték nélküli kommunikációt. [9]

3. ábra. IMEC és a Holst Centre által fejlesztett EEG-headset (2009).



Forrás: <http://www.imec.be/ScientificReport/SR2009/HTML/1213373.html>

2011-ben bemutatták a headset továbbfejlesztett változatát, melyet egy prototípus EEG-headsetbe integráltak (4. ábra). A prototípus a felhasználó fejére könnyen felhelyezhető. A rugós felfüggesztés javítja a stabilitást, valamint egy mágneses csuklós mechanizmus ad lehetőséget az egység finombeállítására. Az elektródák mágneses rögzítése a cserét gyorsá és egyszerűvé teszi. Egyes esetekben zselé használatára is lehetőség nyílik. A rendszer szintén Ag/AgCl elektródákat használ. A prototípus az EEG-jeleket 10 m-re lévő vevőegység felé tudja továbbítani. [10]

[9] Imec: *Imec and Holst Centre bring brain wave monitoring to your home*: http://www2.imec.be/be_en/press/imec-news/imecEEGMD-MWest.html (2011. 02. 08.)

[10] Imec: *Imec, Holst Centre and Panasonic Present Wireless Low-power Active-Electrode EEG Headset*: http://www2.imec.be/be_en/press/imec-news/imeceeg2012.html (2012. 11. 4.)

[10] Imec: *Imec, Holst Centre and Panasonic Present Wireless Low-power Active-Electrode EEG Headset*: http://www2.imec.be/be_en/press/imec-news/imeceeg2012.html (2012. 11. 4.)

4. ábra. IMEC és a Holst Centre által fejlesztett EEG-headset (2011).



Forrás: http://www2.imec.be/be_en/press/imec-news/imecEEGMDMWest.html

Az eszköz újabb továbbfejlesztését adták ki 2012-ben a Panasonic céggel közösen. Javították az eszköz által mért EEG-jelek regisztrálásának minőségét. Az alacsony fogyasztású ($750\mu\text{W}$) jelfeldolgozó egységnek köszönhetően akár 22 órán keresztül képes a 8 érzékelő jelének a rögzítésére, és vezeték nélküli kapcsolaton keresztül történő továbbítására. [10]

5. ábra. IMEC, Holst Centre és Panasonic által fejlesztett EEG-headset (2012).



Forrás: http://www2.imec.be/be_en/press/imec-news/imeceeg2012.html

HYDRODOT EEG-RENDSZER

A Hydrodot EEG-rendszer egy fejre illeszthető keretből, valamint a 10–20 szabvány szerint elhelyezett EEG-érzékelőkből áll (6. ábra). Az eszköz eldobható szenzorokat alkalmaz. A fejre illeszthető keret segítségével a szenzorok precízen pozícionálhatók. A rendszer zselés típusú Ag/AgCl szenzorokat alkalmaz. A mért elektromos jelek egy kábelén keresztül bármilyen szabványos EEG-eszközzel feldolgozhatóak. Az eszközt elsősorban kórházi sürgősségi és intenzív ellátás céljára fejlesztették ki. Az eszköz hátránya, hogy egyszer használatos és felhelyezése, viselése kevésbé komfortos. [11]

6. ábra. Hydrodot EEG-rendszer.



Forrás: <http://www.hydrodot.net/>

EMOTIV EPOC EEG-HEADSET

Az Emotiv System fejlesztette ki az Emotiv EPOC EEG-headsetet. Nem szabványos vezeték nélkül kapcsolaton csatlakoztatható számítógéphez (7. ábra). Az EmoKey software segítségével a parancsokhoz billentyű-kombinációk rendelhetők, így az eszköz egy számítógép vezérlésére könnyen alkalmassá tehető. Az EPOC 14 elektródát alkalmaz (szemben az általában 19 elektródát alkalmazó kórházi rendszerekkel). Fejforgást is képes mérni a beépített két-tengelyes gyorsulásmérőnek köszönhetően. Többféle alkalmazás-fejlesztői csomag is elérhető az eszközhöz. [12]

[11] Hydrodot:
*The hydrodot
EEG system:*
[http://www.
hydrodot.net/](http://www.hydrodot.net/)
(2010. 12. 03.)

[12] Emotiv:
*EEG System
– Electroen-
cephalography:*
[www.emotive.
com](http://www.emotive.com) (2012. 01.
27.)

[13]

NeuroSky:
*Brain Wave
Sensors
for Every
Body*: www.
neurosky.com
(2013. 10. 31.)

Az eszköz az alábbi információkat szolgáltatja:

- Kognitív: Elképzelt mozgás (bal, jobb, fel, le, előre és fordulás, forgás)
- Arkifejezések: egyenkénti szemhéj- és szemöldök-pozíció, szempozíció (mosoly, nevetés) azonosíthatók. Az arkifejezéseket az EEG-szenzorok által érzékelt arcon lévő izmok jelei alapján azonosítja az eszköz és nem agyi hullámok feldolgozása alapján.
- Fejforgás: beépített gyorsulásmérővel mérhető, melyek az EEG-feldolgozástól függetlenek.
- Egyéb: izgatottság, figyelem és ennek ellenkezője.

A headset tulajdonságai alapján a legalkalmasabbnak tűnő választás, akár általános irányítási és egyéb vezérlési feladatok végzésére.

7. ábra. Emotiv EPOC EEG-headset.



Forrás: <http://emotiv.com/>

NEUROSKY EEG-HEADSETEK

NeuroSky EEG-eszközeinek kifejlesztése során számos egyetem működött közre, mint például a Stanford, Carnegie Mellon, University of Washington, University of Wollongong, Trinity College. Az eszközben rozsdamentes ötvözetből készült száraz EEG-szenzort alkalmaznak az agyi elektromos jelek érzékeléséhez. A ThinkGear jelfeldolgozó egység végzi a mért agyjelek szűrését, kiértékelését és az agyhullám-információk meghatározását. Az EEG- eszközök egyszerűbb felépítésűek, viselésük komfortosabb, valamint relatív olcsók. [13]

MINDFLEX

A MindFlex szintén a NeuroSky által fejlesztett technológiára épülő eszköz (8. ábra). Az eszköz egy vezérelt egységből és egy vezeték nélküli EEG-headsetből áll. A készülék egy érzékelőt és két referencia pontot alkalmaz, melyek közül az érzékelő a homloklebeny részen helyezkedik el, a másik kettő referencia a fül alsó részére van felcsíptetve. A NeuroSky által kifejlesztett ThinkGear technológia segítségével a headsetben található jelfeldolgozó egység képes a koncentráció/figyelem értékét meghatározni. [13]

8. ábra. MindFlex.



Forrás: http://mindflexgames.com/how_does_it_work.php

MINDWAVE

A Mindwave EEG-headset a MindFlexhez hasonló eszköz, annak továbbfejlesztett változata, mely a pislogás detektálására is alkalmas. Az eszköz vezeték nélküli kapcsolatra is képes (9. ábra). [13]

[13]
NeuroSky:
*Brain Wave
Sensors
for Every
Body:* [www.
neurosky.com](http://www.neurosky.com)
(2013. 10. 31.)

[[14]

BrainAthlete:
*Mental Sports
Training using
Brain Wave:*
<http://brain-athletesports.com/> (2012.
03. 23.)

9. ábra. Neurosky Mindwave EEG-headset.



Forrás: <http://www.neurosky.com/>

BRAIN ATHLETE

Ezen EEG-sapkát elsősorban sportolók számára fejlesztették (baseball-játékosok, teniszezők, golfozók számára), akik információkat kaphatnak a sportolás közben előálló mentális állapotukról. Az eszköz beépített akkumulátort tartalmaz és Bluetooth kommunikációra képes. [14]

11. ábra. Neurosky Brain Athlete EEG-headset.



Forrás: <http://www.neurosky.com/>

MINDSET

A MindSet mind általános, mind kutatási célokra is használható EEG-headset (12. ábra). A headset beépített akkumulátort tartalmaz, valamint szabványos Bluetooth-kapcsolaton kommunikál, és hagyományos telefon headsetként is alkalmazható, sőt szabványos Bluetooth-fejhallgatóként is, valamint PC-vel történő kommunikáció esetén egy külön vezérlőgombbal egyéb funkciók is elérhetőek. Az eszköz mind a feldolgozott (agyhullám-információk a 3–50 Hz frekvenciatartományban), mind pedig a nyers mérési adatokat is szolgáltat, illetve figyelem/koncentráció, meditáció-értéket is meghatároz. [13]

12. ábra. Neurosky MindSet EEG-headset.



Forrás: <http://www.neurosky.com/>

EEG-headsetek összehasonlítása

Az alábbi táblázatban összefoglalóan látható az előzőekben ismertett EEG-headsetek néhány fontosabb paramétere (1. táblázat).

[13]
NeuroSky:
*Brain Wave
Sensors
for Every
Body*: [www.
neurosky.com](http://www.neurosky.com)
(2013. 10. 31.)

1. táblázat. EEG-headsetek összehasonlítása.

	Akkumulátoros táplálás	Érzékelők száma	Érzékelő típusa	Vezetéknélküli adatátvitel	Információ típusa
NeuroFocus	Igen	18	Száraz	Igen (Bluetooth)	Feldolgozott
IMEC és Holst Centre	Igen	10	Száraz/ Zselés	Igen (egyéb)	Feldolgozott
Hydrodot	Nem	20	Zselés	Nem (vezetékes)	Nyers adat
Emotiv EPOC EEG	Igen	14	Száraz	Igen (egyéb)	Feldolgozott
MindFlex	Igen	1	Száraz	Igen (egyéb)	Feldolgozott
Mindwave	Igen	1	Száraz	Igen (egyéb)	Feldolgozott /Nyers adat
Brain Athlete	Igen	1	Száraz	Igen (Bluetooth)	Feldolgozott
MindSet	Igen	1	Száraz	Igen (Bluetooth)	Feldolgozott/Nyers adat

Az összehasonlításból látható, hogy a vizsgált EEG-eszközök legtöbbje vezetéknélküli kapcsolattal rendelkezik és akkumulátoros táplálású. Ez lehetővé teszi, hogy az eszközök viselése minél komfortosabb, kevésbé zavaró legyen. A beépített alacsony fogyasztású jelfeldolgozó egységnek köszönhetően akár feldolgozott információk közvetlen szolgáltatására is alkalmasak, mely nagyban megkönnyíti alkalmazásukat és egyéb eszközökhöz történő illesztésüket. Kereskedelmi forgalomban elsősorban az

Emotiv és a Neurosky EEG eszközei elérhetőek, ezek a legolcsóbb és legkomfortosabb, legegyszerűbben felhelyezhető eszközök. A gyártó weboldalán található információk alapján a NeuroSky által gyártott Mindwave \$100, míg az Emotiv fejlesztési célokra is használható, nyers adatokat is szolgáltató headset \$750. A táblázat alapján az is látható, hogy míg a Neurosky egy érzékelőt alkalmaz, addig a többi eszköz több érzékelő segítségével gyűjt információt az agyi tevékenységről. Természetesen több – a fej eltérő részén elhelyezett – érzékelő felhasználásával több információ is kinyerhető, mint például az elképzelt mozgás.

Összegzés

Az ismertetett EEG-jelfeldolgozó eszközök a tradicionális EEG-regisztráló berendezésekhez képest igen nagy fejlődésen mentek keresztül, melynek köszönhetően az alkalmazási lehetőségeik kibővültek. Cikkünk összefoglaló képet adott az EEG-n alapuló non-invazív eljárást alkalmazó EEG-mérésadatgyűjtő és -feldolgozó eszközök tulajdonságairól. Az összefoglalás alapján megállapítható, hogy ma már több olyan EEG-mérésadatgyűjtő és -feldolgozó eszközök is elérhetőek, melyek nem igényelnek speciális alkalmazási ismereteket, használatuk egyszerűsödött. Ennek következtében már nem csak orvosi jellegű alkalmazásokban merülhet fel használatuk, hanem egyéb más területekkel is bővíthet, mint például a reakcióidő csökkentése, az éber állapot megfigyelése, az elképzelt mozgás detektálásán alapuló irányítások. A kereskedelmi illetve kutatási célra kifejlesztett, bárki számára elérhető EEG-regisztráló és -jelfeldolgozó eszközök folyamatosan fejlődnek, így a jövőben számos új, további funkciókkal bővített eszköz láthat napvilágot.

Topológiai eltérések minimalizálására visszavezetett graféma-leszármazási vizsgálatok

*BME Elektronikus Eszközök
Tanszéke
E-mail: hosszu@eet.bme.hu

Összefoglalás: A számítógépes paleográfia az alkalmazott informatika egyik ága, amely az írások, az egyes írásrendszerek és az írásokban található grafémák (betűk, képjelek és írásjelek) fejlődésének és egymásra hatásának vizsgálatával foglalkozik. Ennek során figyelembe veszi a társtudományok (nyelvészet, hangtan, történelem, földrajz) eredményeit is, azonban módszerei döntően a matematikai statisztika területére esnek. A cikk a grafémák leszármazási kapcsolatainak felderítésével foglalkozik. Két graféma rokonsági kapcsolatának következménye, hogy vagy azonos hangértékűek, vagy a hangértékbeli eltéréseik nyelvészeti vagy paleográfiai okokkal indokolhatók. Továbbá az egymással leszármazási viszonyban álló grafémák grafémaalakjai között mindig van valamilyen hasonlóság. Ez lehet teljes egybeesés, vagy csak módosulat. A grafémamódosulat jellemzően valamilyen topológiai transzformáció eredménye. Az írások fejlődésének vizsgálata alapján a szerző meghatározta a grafémák fejlődésénél szóba jöhető topológiai transzformációkat. Az egyes grafémák elhelyezése a grafémák leszármazási sorában régóta kutatott probléma. Rendszerint egy-egy írásból, mint a grafémák együtteséből kiindulva próbálják a rokonsági kapcsolatokat meghatározni. A szerző ezzel szemben az egyes grafémák közötti viszonyokat veszi figyelembe és ennek alapján kidolgozott egy grafémákra vonatkozó leszármazási láncot felépítő, matematikai optimalizálást tartalmazó algoritmust. Ennek lényege, hogy egy-egy graféma a már meglévő leszármazási láncok közül abba és a láncon belül oda kerül elhelyezésre, ahol az egymást követő grafémák topológiai eltéréseiből képzett hibafüggvény a legkisebb lesz. A szerző a kidolgozott leszármazási lánc építéséhez használható algoritmust alkalmazta különböző megfejtett írásmlékekben található jelek eredetének tisztázására.

Az egyes grafémákra létrejött leszármazási kapcsolatok összegzéséből a vizsgált grafémák által alkotott írás fejlődésének modellezése is lehetségessé válhat.

Kulcsszavak: Graféma, optimalizálás, számítógépes paleográfia.

Abstract: The computational paleography is a branch of applied informatics, which deals with the developments of scripts, script families, and graphemes (letters, ideograms, and punctuation marks of scripts. In order to achieve this goal some methods falling within the mathematical statistics are used. The article deals with exploring the genealogical relationships of the graphemes. The author takes into account the topological relationships between individual graphemes and builds the genealogical ties of the graphemes based on mathematical optimization, which minimizes the error function of the topological variations of the graphemes in a certain descent chain. The elaborated algorithm was applied for exploring the genealogical ties of graphemes. Based on the obtained results, exploring the development of a script could be also possible.

Keywords: Grapheme, optimization, computational paleography.

Bevezetés

Az írások fejlődése régóta tudományos kutatás tárgya. Ennek oka valószínűleg az, hogy az emberi gondolkodás fejlődése tükröződik a megmaradt írásemlékekben, amelyek jelentős része még ma is olvasatlan. A számítógépes paleográfia egy alkalmazott informatikai terület, amely érintkezik a nyelvtörténettel, a fonetikával, a történelemmel és a régészettel. A kifejlesztett grafémalánc-modellezési eljárás segítségével az egyes vizsgált grafémák adott valószínűséggel beilleszthetők egy leszármazási sorba. A módszer korábbi változatai már alkalmazásra kerültek a grafémaalakok közötti kapcsolatok feltárására vonatkozó vizsgálatokban [1], [2], [3]. A matematikai leírás mellett példákon keresztül bemutatásra kerül a kifejlesztett graféma leszármazás-vizsgálati modell hasznossága.

[1] Hosszú Gábor: *Az informatika írástörténeti alkalmazásai*. Informatika Korszerű Technikái Konferencia (IKT 2010), meghívott plenáris előadás, 2010. március 5–6, Dunaújvárosi Főiskola.

[2] Hosszú, Gábor: *Rovás paleográfia: Dinamikus vagy statikus? Nyelv és tudomány*. <http://www.nyest.hu/hirek/rovaspaleografia-dinamikus-vagy-statikus>, 2012.

[3] Tóth Loránd–Dian Szabolcs–Raymond Pardede–Hosszú Gábor–Kovács Ferenc: *Jelentésazonosító eljárás a 16–18. századi székely-magyar rovás emlékek értelmezésére*. IKT 2010, Informatika Korszerű Technikái Konferencia, 2010. március 5–6., Dunaújvárosi Főiskola

[4] Hosszú, G.: *Heritage of Scribes. The Relation of Rovas Scripts to Eurasian Writing Systems*. 2011: Első kiadás. 2012: Második, bővített kiadás. Budapest: Rovás Alapítvány. Olvasható a Google Books-on: <http://books.google.hu/>

[5] Rogers, H.: *Writing systems: A linguistic Approach*. Ser.: Blackwell Textbooks in Linguistics; 18. Blackwell Publishing, 2005.

[6] Malatesha Joshi, R.–Aaron, P. G. (Eds.): *Handbook of Orthography and Literacy*. London: Routledge, 2006

A cikk először az alapvető fogalmakat tisztázza, majd a paleográfiai szakirodalomból ismert néhány írásleszármazással kapcsolatos eredményt ismertet röviden, ezt követően pedig egy új eljárást ismertet az egymással valószínűleg rokon grafémák lehetséges leszármazási vonalainak minősítésére és ezzel a legvalószínűbb eset kiválasztására. Végül egyszerű példákon ismertetésre kerül a kidolgozott graféma-leszármazást meghatározó algoritmus.

Számítógépes paleográfiai alapok

A számítógépes paleográfiában a *graféma* kulcsfogalom, ennek és az ezzel összefüggő fogalmak meghatározását mutatja be az 1. táblázat. A cikkben a grafémákra a *grafémaalak*kal, az azt követő *grafémanév*vel és az utána feltüntetett *hangérték*kel (vagy hangértékekkel) hivatkozom, pl. a **Ű**BEZÁRT T /t/ [4].

1. táblázat. A grafémával összefüggő fogalmak meghatározása.

Fogalom	Leírás
betű	Olyan <i>graféma</i> , amely egy fonémát jelöl.
fonéma	Egy nyelv hangjainak megkülönböztető egysége [5].
graféma	A jel elvonatkoztatása, amely egy vagy több tipizált grafémaalakkal rendelkezik. A <i>graféma</i> megjelenési formája a <i>grafémaalak</i> . A graféma tulajdonságai: (i) az írás, amihez tartozik; (ii) grafémaalakok, (iii) hangértékek, (iv) használati időszak (előtte nem létezett, utána pedig eltűnt a gyakorlatból). Graféma lehet a <i>betű</i> , a <i>ligatúra</i> , a <i>szám</i> és az <i>írásjel</i> .
grafémaalak (glyph)	Egy graféma egyes feliratokban megjelenő, a tipizálttól esetleg eltérő topológiai tulajdonságú rajzolata.
írás (script)	A hangok egy bizonyos rendszerben történő rögzítését megjelenítő grafikus formátum. Az írásra példa a latin betűs írás (Roman script) vagy az arab írás (Arabic script) [6].

Fogalom	Leírás
írásbeliség (orthography)	Egy nyelv vizuális megjelenítése, amely meghatározott a nyelv fonológiai, szintaktikai, morfológiai és szemantikai jellemzői által. Az írásbeliségre példa a kínai írásbeliség vagy a latin betűs magyar írásbeliség [6].

Az írások leszármazása hagyományosan az egyes írások közötti szülő-gyermek kapcsolatok segítségével történik. Ez azon írások esetén elfogadható megoldás, amelyek ábécéje jól ismert és számos írásemlék maradt fenn [7], [8]. Azonban még ezen írások esetében is csak egy közelítés, hiszen egy-egy graféma bekerülhetett egy írásba olyan írásból is, amely eltér a szülő-írástól.

A különböző írások többszörösen hatottak egymásra és számos keverékük jött létre, amelyekben csak egy-egy betű került át egy másik írásból. Így az írásokat azok egészében kezelő hagyományos írásfejlődés-modellezési módszer alkalmatlannak bizonyul ezen írásbeliségek származásának leírására, ugyanis az egyes írásokat különböző történelmi korokban eltérő országokból érték kulturális hatások, így egy írásban sokféle ismert írás betűje megtalálható lehet. Vagyis egy írás fejlődésének leírása legpontosabban a grafémaszinten történő leszármazási viszonyok figyelembevételével történhet.

A különböző írásemlékeket össze lehet hasonlítani a bennük szereplő azonos alakú grafémák előfordulása alapján, amihez egy korábbi vizsgálatban klaszteranalízist [9] alkalmaztam. Mivel a grafémák kategória típusú adatpontok, ezek összehasonlítására a bináris értékű számszerűsítést [10], az egyes grafémaalakok különbözőségét leíró távolságmátrix létrehozásához pedig az **egyszerű egyezési együttható** (*Simple Matching Coefficient, SMC*) távolságmértéket használtam [11]. Az így kidolgozott eljárással különböző írásemlékekben használt grafémakészletek összehasonlításán keresztül az írásemlékekhez használt írásbeliségek különböző leszármazási kapcsolatairól tudok megállapításokat tenni.

A következőkben nem egész ábécék, hanem csak egyes grafémák leszármazási viszonyait fogjuk megvizsgálni. A grafémák alakzati (topológiai) tulajdonságait különböző jellemzőkkel le tudjuk írni, amelyeket a 2. táblázat mutat be.

[6] Malatesha Joshi, R.– Aaron, P. G. (Eds.): *Handbook of Orthography and Literacy*. London: Routledge, 2006.

[7] Faulmann, C.: *Das Buch der Schrift*. Enthaltend die Schriftzeichen und Alphabete aller Zeiten und aller Völker des Erdkreises. Wien 1880.

[8] Hackh, I. W. D.: The history of the alphabet. *The Scientific Monthly*. Vol. 25, No. 2, (Aug., 1927), American Association for the Advancement of Science, 97–118.

[9] Fogaras Dániel–Lukács András– Iványi Antal (Szerk.) (2005): *Klaszterezés*. In: Informatikai algoritmusok 2. Budapest: ELTE Eötvös Kiadó, 1397–1423.

[10] Pang-Ning Tan–Steinbach, M.–Kumar, V. (2005): *Introduction to Data Mining*. Cambridge: Pearson.

[11] Sokal, R. R.–Michener, C. D. (1958): *A Statistical Method for Evaluating Systematic Relationships*. The University of Kansas Scientific Bulletin, 38: 1409–1438.

2. táblázat. Az „a” írás egy grafémaalakjának alakzati jellemzőit leíró $\mathcal{G}_a$ vektor elemei.

Jellemző	Leírás
$\mathcal{G}_{a,1}$	végződés (egyfokú csomópontok) száma
$\mathcal{G}_{a,2}$	kétágú csúcsok (kétfokú csomópontok) száma
$\mathcal{G}_{a,3}$	háromágú csúcsok (háromfokú csomópontok) száma
$\mathcal{G}_{a,4}$	négy- v. többágú csúcsok (négy- v. magasabb fokú csomópontok) száma
$\mathcal{G}_{a,5}$	közel egyenes vonalak, azaz élek mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,6}$	jelentős görbületű vonalak, azaz ívek mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,7}$	hurkok száma
$\mathcal{G}_{a,8}$	(közel) függőleges vonalak mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,9}$	(közel) vízszintes vonalak mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,10}$	(közel) 45°-ban balra dőlő vonalak mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,11}$	(közel) 45°-ban jobbra dőlő vonalak mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,12}$	a grafémaalak (vízszintesen vagy függőlegesen) tengelyesen tükrözött (<i>mirrored</i>) egybe nem eső vonalainak mennyisége (negyed betűmagasságban kifejezve)
$\mathcal{G}_{a,13}$	a nem tükörképi helyzetben lévő, egymással párhuzamos (<i>parallel</i>) vonalak mennyisége (negyed betűmagasságban kifejezve)

A grafémaalakok származtatása a topológiai jellemzőik összevetését igényli. Ennek algoritmikus feldolgozásához szükséges a számszerűsítésük. Ezt két graféma g_1 és g_2 vektorral jellemezhető alakjának különbözőségét leíró $\delta(g_1, g_2)$ értékkel lehet kifejezni, lásd (1).

$$\delta(g_1, g_2) = \frac{\sqrt{\frac{1}{N_G} \sum_{i=1}^{N_G} w_i (g_2(i) - g_1(i))^2}}{\sqrt{\frac{1}{N_G} \sum_{i=1}^{N_G} w_i (\max(g_1(i), g_2(i)))^2}} \quad (1)$$

ahol a két – g_1 és g_2 – grafémaalak $\delta(g_1, g_2)$ különbözőségét úgy számítjuk ki, hogy az alakzati jellemzőnként számított topológiai eltérés súlyozott négyzetes átlagát normáljuk ugyanezen alakzati jellemzőknek a két vizsgált grafémaalaknál előforduló maximumának súlyozott négyzetes átlagával. Az (1)-ben N_G az alakzati jellemzők száma, a $w_1, w_2, \dots, w_i, \dots, w_{N_G}$ értékek pedig az egyes alakzati jellemzők viszonylagos fontosságát kifejező súlytényezők, amelyeket egyes írásoknak a nemzetközi paleográfiai szakirodalomból [7], [8] ismert leszármazási kapcsolatai alapján heurisztikus úton állapítottam meg. A súlyokra igaz (2).

$$\sum_{i=1}^{N_G} w_i = N_G \quad (2)$$

Az (1) egyszerűsítéséből adódik (3)

$$\delta(g_1, g_2) = \frac{\sqrt{\sum_{i=1}^{N_G} w_i (g_2(i) - g_1(i))^2}}{\sqrt{\sum_{i=1}^{N_G} w_i \max(g_1(i), g_2(i))^2}} \quad (3)$$

[7] Faulmann, C.: *Das Buch der Schrift*. Enthaltend die Schriftzeichen und Alpha-bete aller Zeiten und aller Völker des Erdkreises. Wien 1880.

[8] Hackh, I. W. D.: The history of the alphabet. *The Scientific Monthly*. Vol. 25, No. 2, (Aug., 1927), American Association for the Advancement of Science, 97–118.

A (3) figyelembe veszi, hogy bonyolultabb alakzatok esetén a kisebb eltérések nem számottevőek, míg kisebb összetettségű grafémaalakoknál a legkisebb különbség is jelentős lehet. A vizsgálatokban a $(w_1, w_2, \dots; w_i, \dots; w_{N_G})$ normált súlytényezők helyett a tapasztalati úton meghatározott

$$\tilde{W} = (2, 4, 9, 6, 1, 1, 4, 1, 1, 1), w_i = N_G \tilde{w}_i / \sum_{i=1}^{N_G} \tilde{w}_i, i = 1, 2, \dots, N_G$$

értékeket vettem figyelembe. A hurkok számára vonatkozó w_7 súlytényező értéke 4, ami viszonylag jelentős érték. Ennek az oka az, hogy míg a nem záruló egyenes és görbe vonalakat hat különböző jellemzőben is számításba vettem ($w_5, w_6, w_8, w_9, w_{10}, w_{11}$), addig a grafémaalakot jelentősen meghatározó hurkok száma csak egy jellemzővel (w_7) van képviselve. Hasonlóan a csomópontok számára vonatkozó súlytényezők is nagyobb értékűek (w_1, w_2, w_3, w_4), különösen a metszéspontokat jelző hármas és négyes csomópontok (w_3, w_4).

A grafémaalak-összetettségi mutató

A különböző grafémák (betűk) alakja jellegzetesség szempontjából erősen eltérő lehet. Nyilvánvaló, hogy egy egyszerű függőleges vonalból álló betű szinte minden írásrendszerben előfordul, természetesen különböző jelentéssel: a latin betűs írásban ennek grafémaalaknak a jelentése /i/ vagy /l/, míg a rovásírásban mindig /sz/. Ezért a latin „l” vagy „l” és a rovás „l” SZ /sz/ lényegében azonos grafémaalakja semmiféle lezármazási kapcsolatra nem utal. Ezzel szemben, ha egy összetett grafikai alakzatú, jellegzetes alakú betű található több különböző írásrendszerben, akkor valószínűsíthető közöttük valamilyen származási kapcsolat. Erre példa a Φ BEZÁRT S /s/ és a vele rokon Φ US /s/ graféma. A különböző ábécékben előforduló grafémaalakok véletlenszerűen is lehetnek hasonlóak vagy azonosak. Az ebből adódó tévedés nyilvánvalóan elkerülendő. Ennek az esélye annál kisebb, minél összetettebb egy grafémaalak. A grafémaalak grafikai összetettségének mérésére vezessük be a **grafémaalak-összetettségi mutatót** (*Glyph Complexity Parameter*, **GCP**), amely azt adja meg, hogy egy grafémaalak topológiailag mennyire bonyolult. A GCP kiszámítása a heurisztikus (4) képlet alapján történik.

$$GCP = (\text{sgn } N) \left(\sum_{i=1}^N N_i^{(E)} - 1 \right) + (\text{sgn } L) \left(\sum_{j=1}^L (N_j^{(L)} + 1) \right) + A + M + P + (S - 1) \quad (4)$$

ahol N a csomópontok (*node*) száma; $N_i^{(E)}$ a vizsgált grafémaalakban az i -edik csomópontához tartozó élek (*edge*) száma; L a grafémaalakban található egymást át nem lapoló hurkok (*loop*) száma; $N_j^{(L)}$ a vizsgált grafémaalakbeli j -dik hurok csomópontjai közötti élek száma; A az egyik hurokba sem tartozó ívelt élek, azaz ívek (*arch*) száma; M a grafémaalakban vízszintesen vagy függőlegesen tükrözött (*mirrored*) egybe nem eső élek száma, P a nem tükröképi helyzetben lévő, egymással párhuzamos (*parallel*) élek száma, S pedig a grafémaalak topológiáját alkotó egymástól elkülönített (*separated*) alakzatok száma. Ha a vonalak ívesek, vagyis ívekről van szó, akkor a párhuzamossággént a koncentrikus tulajdonságot vesszük figyelembe. Az i és j futóindexek. Azokat a hurkokat, amelyeket átmenő vonalak többfelé osztanak, és ezzel formálisan több hurokra esnek szét, a (4) képlet második tagjában egyetlen hurokként veszem számításba, nem véve figyelembe a felosztó vonalakat (azok az első tagban már beszámításra kerültek) a hurok csomópontjai közötti élek számánál sem.

A GCP kiszámításánál az egy vonalba eső, és egymással érintkező párhuzamos éleket, ill. koncentrikus íveket csak egyszer számoljuk, néhány példát bemutat a 3. táblázat. A tapasztalatok alapján, ha két, nagy GCP-jű grafémaalak hasonló, akkor valószínű, hogy rokonai egymásnak.

3. táblázat. Különböző grafémák GCP értékei.

Graféma	GCP
$\supset N$	$(1+1-1)+0+1+0+0+(1-1)=2$
$\boxplus Z$	$(2+3+3+1+1+3+3+2-1)+(4+4+1)+0+2+3+(1-1)=31$
$\emptyset US$	$(3+3-1)+(2+1)+0+2+0+(1-1)=10$

A kidolgozott graféma-leszármazási algoritmus

Az egyes grafémaláncok úgy jöttek létre, hogy a különböző leszármazási lehetőségek közül azt választottam ki, amelyik a láncban lévő grafémák grafémaalakjainak legkisebb eltérését jelentette súlyozva az összehasonlításra kerülő grafémaalakok összegzett grafémaalak-összetettségi mutatójával. Más szóval az (5) képletben megadott hibafüggvényt minimalizáltam. Ezen alapul a grafémák leszármazásának a hangértékeket figyelembe vevő topológiai hasonlóságokon alapuló kifejlesztett modellezési módszere.

[4] Hosszú, G.: *Heritage of Scribes. The Relation of Rovas Scripts to Eurasian Writing Systems*. 2011: Első kiadás. 2012: Második, bővített kiadás. Budapest: Rovás Alapítvány. Olvasható a Google Books-on: <http://books.google.hu/>

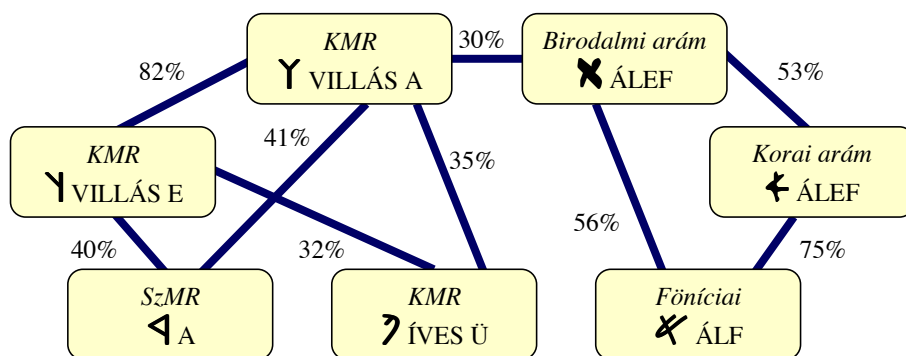
$$\sum_{i=1}^{N_H} [\delta(g_i, g_{i+1})]^2 \quad (5)$$

ahol g_i a vizsgált grafémaleszármazási lánc i -edik elemében szereplő grafémaalak, N_H a grafémaleszármazási lánc vizsgált részében lévő grafémák száma, $\delta(g_i, g_{i+1})$ az i -edik és az $(i+1)$ -edik grafémaalak eltérése. A grafémaalakok hasonlóságát a (6) egyenlettel lehet kiszámítani.

$$\sigma(g_1, g_2) = 1 - \delta(g_1, g_2) \quad (6)$$

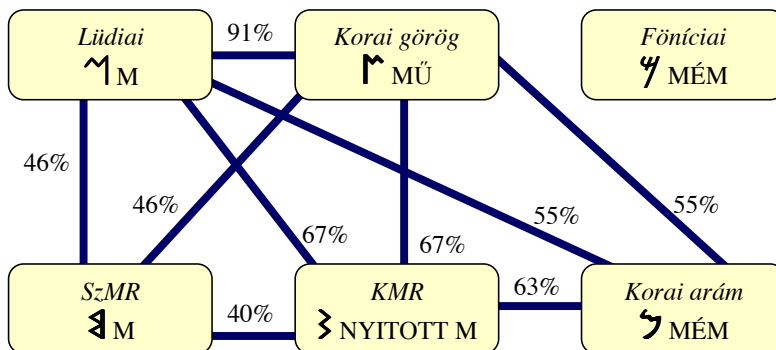
A grafémaalakok leszármazására kialakított hierarchikus leszármazási rend [4] néhány részlete az alábbiakban bemutatásra kerül. A főníciai ÁLF-grafémával rokonnak feltételezhető grafémák hangértékei egymással rokonságban állnak [4]. Az 1. ábra a főníciai ÁLF-grafémával rokon hangértékű grafémák rokonsági gráfját mutatja. A rovásírás két változatának jelölése: **KMR**: Kárpát-medencei rovás és **SzMR**: székely-magyar rovás.

1. ábra. A főníciai ÁLF-grafémával rokon hangértékű grafémák rokonsági gráfja a 26%-nál nagyobb mértékű hasonlósági viszonyokat ábrázolva.



A 2. ábra a föníciai MÉM-grafémával azonos, /m/ hangértékű grafémák rokonsági gráfját mutatja.

2. ábra. A föníciai MÉM-grafémával azonos hangértékű grafémák rokonsági gráfja a 39%-nál nagyobb mértékű hasonlósági viszonyokat ábrázolva.

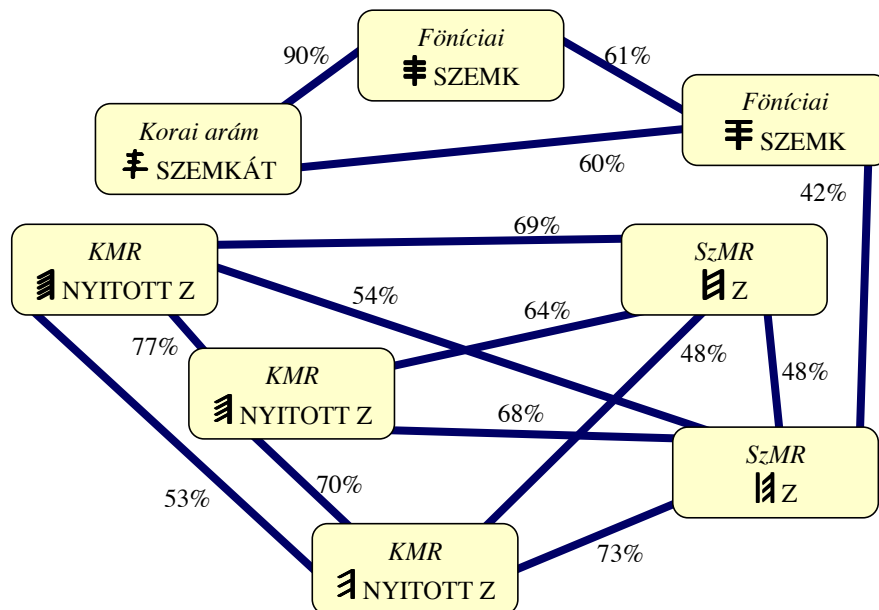


A föníciai SZEMK grafémával rokonnak feltételezhető grafémák hangértékei egymással rokonságban állók: föníciai 𐤌, 𐤍 SZEMK: /s/ zöngétlen alveoláris spiráns; korai arám 𐤌 SZEMKÁT: /sʰ/ nyomatékos zöngétlen alveoláris spiráns (nyomatékos sz) és KMR 𐤌, 𐤍, 𐤎 NYITOTT Z, ill. SzMR 𐤌, 𐤍 Z: /z/ zöngés alveoláris spiráns [4]. A 3. ábra a SZEMK grafémával rokon hangértékű grafémák rokonsági gráfját mutatja. Az ábrán több grafémaalak-változat is szerepel.

[4] Hosszú, G.: *Heritage of Scribes. The Relation of Rovas Scripts to Eurasian Writing Systems*. 2011: Első kiadás. 2012: Második, bővített kiadás. Budapest: Rovás Alapítvány. Olvasható a Google Books-on: <http://books.google.hu/>

[4] Hosszú, G.: *Heritage of Scribes. The Relation of Rovas Scripts to Eurasian Writing Systems*. 2011: Első kiadás. 2012: Második, bővített kiadás. Budapest: Rovás Alapítvány. Olvasható a Google Books-on: <http://books.google.hu/>

3. ábra. A főníciai SZEMK-grafémával rokon hangértékű grafémák rokonsági gráfja a 37%-nál nagyobb mértékű hasonlósági viszonyokat ábrázolva.



A fentiekben bemutatott, a grafémák hasonlósági kapcsolataira vonatkozó eredmények önmagukban nem elegendőek egy egyértelmű lezármazási vonal kialakításához, azonban felhasználhatók az (5) képletben megadott hibafüggvény minimalizálásához. Ez utóbbi optimalizálás eredményeképpen már kapható egy graféma-lezármazási adatbázis [4].

Következtetések, összefoglalás

A cikkben ismertetett új írásanalízis-módszer az egyes grafémaalakok fonetikai értékein, alakzati összehasonlításán, továbbá a leletek földrajzi és történelmi körülményein alapul. A feltárt kapcsolatokon alapulva lehetővé vált a különböző grafémaalakok módszeres leírása, amely alapján egy grafémaalak-szintű leszármazási adatbázis hozható létre, amely a hangtani és morfológiai jellemzőkön alapulva végigköveti az egyes grafémák fejlődését az elődjeiktől a leszármazottjaikig.

A bemutatott új grafémaleszármazás-vizsgálati eljárással lehetőség nyílik a különböző írásbeliségek között átörökített egyes grafémák fejlődésének nyomon követésére. A szükséges modellezési és minimalizáló eljárások ismertetése után példákon került bemutatásra a kifejlesztett módszer használhatósága.

Köszönetnyilvánítás

A munka szakmai tartalma kapcsolódik az *Új tehetséggondozó programok és kutatások a Műegyetem tudományos műhelyeiben* c. projekt szakmai célkitűzéseinek megvalósításához. A projekt megvalósítását a TÁMOP – 4.2.2.B-10/1-2010-0009 program támogatja.

Naprendszer-szimuláció a Virtuális valóság modellezése tantárgyban

*Gábor Dénes Főiskola
E-mail: friedel@gdf.hu

**Gábor Dénes Főiskola
E-mail: kaczur@gdf.hu

[1] *Virtuális valóság modellezése.*
Tantárgyi útmutató:
[http://ilias.gdf.hu/
repository.php?ref_
id=41199&cmd=sendfile](http://ilias.gdf.hu/repository.php?ref_id=41199&cmd=sendfile)
(bejelentkezve érhető el)

Összefoglalás: A cikk témája kapcsolódik a *Virtuális valóság modellezése* című tantárgyhoz, amely szabadon választható a Gábor Dénes Főiskola negyedéves mérnök-informatikus szakos hallgatói számára. A cikk ismerteti a feladat-specifikációt, ennek objektumorientált tervezését, a térben elhelyezkedő objektumok pozíciójának leképezését a síkra, a tömegvonzás közelítő kiszámítását a modelltérben, és a megjelenítést. A megvalósítás során különböző technológiákat hasonlítunk össze, például HTML5 canvas és JavaScript, Java2D.

Kulcsszavak: Felsőoktatás, programozás, virtuális valóság modellezése, szimuláció, tömegvonzás.

Abstract: The main topic of this article is about the facultative subject called Virtual Reality for the students studying IT Engineering at Dennis Gabor College. This article explains the specification with object-oriented design: how to map objects to the plane from the space, how to approximate calculation of the gravity model space, and visualization. During the implementation, authors compare different technologies, such as HTML5 Canvas and JavaScript, Java2D.

Keywords: Higher education, programming, virtual reality, simulation, gravity.

Bevezetés

A Gábor Dénes Főiskolán, a mérnök-informatikus szakos hallgatók számára a 7. félévben választható tantárgy a *Virtuális valóság modellezése* [1]. A tantárgy tematikájának elemei: a VR és a kapcsolódó fogalmak definíciói, a témakör áttekintése, a VR története; a VR felhasználása, előnyei és hátrányai, eszközei; 360°-os panorámás képek, 3D-s képek és filmek, 4D-s filmek,

holográfia, 3D modellező programok; Kiterjesztett valóság: az AR interfészel szemben támasztott követelmények, az AR típusai, felhasználása, jövője; a WebGL, az X3DOM és a Java grafikus lehetőségeinek áttekintése; a DirectX és az OpenGL nyelvek lehetőségeinek áttekintése; a VRML története és a nyelv áttekintése [1]. A gyakorlatokon a következő szoftverek, technológiákat, platformokat használtuk/használjuk: Blender, VRML, HTML5 canvas és JavaScript, Java2D, Java3D, C# és DirectX, C++ és OpenGL. A 2012–2013-as tanévben jelentősen frissítettük, bővítettük a példatárainkat, főként a JavaScript, Java, OpenGL területén. Tananyagaink az ILIAS LCMS rendszerben érhetők el [2]. A cikkhez tartozó programok szabadon letölthetők [3].

ELMÉLETI HÁTTÉR

A Naprendszer-szimulációhoz elengedhetetlen, hogy ismerjük a homogén koordinátákat, az elemi műveletek egységes megvalósításához szükséges transzformációs mátrixokat, a tömegvonzás elvét és az implementációhoz szükséges MVC-modellt.

HOMOGÉN KOORDINÁTÁK

Számítógépes algoritmusokkal egyszerű a térbeli transzformáció megvalósítása, ha homogén koordinátákat használunk. Segítségükkel az affin transzformációk egységesen kezelhetők. A cél egy egységes matematikai formalizmus alkalmazása [4, 5].

A pontok az égitestek középpontjait fogják jelölni. Legyen P pont 3D-beli 3 koordinátája:

$$P = (x, y, z) \quad (1)$$

[1] *Virtuális valóság modellezése*. Tantárgyi útmutató: http://ilias.gdf.hu/repository.php?ref_id=41199&cmd=sendfile (bejelentkezve érhető el)

[2] GDF ILIAS: <http://ilias.gdf.hu>

[3] A cikkhez tartozó programok: <http://kaczursandor.hu/VV3D>

[4] Homogén koordináták és transzformációk: <http://www.agt.bme.hu/szakm/szg/homogen.htm>

[5] Kaczur, S.–Kopácsi, S. (2008): Practical application of coordinate and dot transformations. A *GAMF Közleményei*, Kecskemét. XXIII. évf., Pp. 121–126.

[5] Kaczur, S.–Kopácsi, S. (2008): Practical application of coordinate and dot transformations. A *GAMF Közleményei*, Kecskemét. XXIII. évf., Pp. 121–126.

[6] Budai A.–Vári Kakas I. (2007): *Számítógépes grafika*. (3.1.3. A modelltér transzformációi, 88), Budapest: INOK.

[7] Berke J.–Hegedűs Gy. Cs.–Kelemen D.–Szabó J. (2002): 5.3. *Koordináta-transzformációk*, 78. Keszthely: Veszprémi Egyetem Georgikon Mezőgazdaságtudományi Kar.

[8] Hack F. (2002): *3D-grafika geometriai alapjai*. (2. Koordinátatranszformációk, 19) Budapest: ELTE.

Szükséges egy konstans érték. Ha $w \neq 0$, akkor P pont koordinátái:

$$P = (w \cdot x, w \cdot y, w \cdot z, w) \quad (2)$$

Ha $w = 1$, akkor a P pont normalizált homogén koordinátái:

$$P = (x, y, z, 1) \quad (3)$$

A pontnégyes kijelölése kölcsönösen egyértelmű [8]. Ha a P 3D-ben (1) szerinti, akkor a P homogén pontnégyese (3) és fordítva.

TRANSZFORMÁCIÓK

Koordináta-transzformáció során az ábrázolandó grafikus objektum pontjaihoz (tárgypontokhoz) új koordináta-rendszert rendelünk hozzá. Az objektum nem változik (nem torzul, nem változtatja meg az alakját), csupán a nézőpont változik meg [5, 6, 7]. Például: a koordináta-rendszer eltolása, elforgatása, a koordinátatengelyek felcserélése, tükrözése, és a léptékváltás (nagyítás, kicsinyítés, összenyomás, széthúzás), elforgatjuk az ekliptika síkját a szimulált Naprendszerben.

Ponttranszformáció esetén a tárgypontokhoz hozzárendeljük azok egy adott szempont szerinti hasonmását [6]. Például: 3D-s tárgyak leképezése 2D-s képre, objektumok eltolása, forgatása, mozgatása, bolygók mozgatása tömegvonzás alapján. Affin-transzformációk (egybevágósági és hasonlósági transzformációk) alkalmazása esetén pont képe pont, szakasz képe szakasz, felület képe felület, valamint metsző térelemek eredeti metszésvonala meg egyezik azok leképezett metszésvonalával [6, 8].

A számítógépes grafika területén az affin-transzformációk általános alakja (mátrixosan) [6]:

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} t_{11} & t_{12} & t_{13} & b_x \\ t_{21} & t_{22} & t_{23} & b_y \\ t_{31} & t_{32} & t_{33} & b_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} \quad (4)$$

A pont a $B = (b_x, b_y, b_z)$ vektorral eltolható. A pont – a $T = (t_{11}, t_{12}, \dots, t_{33})$ mátrixot használva – adott szöggel elforgatható, skálázható, tükrözhető. A számítógépes grafikában a (4) szerinti transzformációs mátrixot (3) szerinti homogén koordinátákkal alkalmazva, az összes geometriai transzformáció hatékonyan megvalósítható, visszavezethető mátrixok szorzására.

Mindezt saját magunk is implementálhatjuk, de része a DirectX és OpenGL rendering pipeline-jának is [6].

Más módon is lehetne: egyenes és ehhez tartozó szög párossal [9] is dolgozhatnánk.

A TÖMEGVONZÁS ELVE

A tömegvonzás (5) bármely két égitest között meghatározott, függ a gravitációs állandótól és az égitestek tömegétől egyenes arányban, és az égitestek (tömeg)középpontjainak távolságától fordított arányban. Ez a Newton szerinti értelmezés.

$$F_{grav} = G \frac{m_1 m_2}{d^2} \quad (5)$$

A hatás-ellenhatás törvénye miatt a vonzás – egymás felé való gyorsulás – kölcsönös, a gyorsulás az égitestek tömegével fordítottan arányos, sosem nulla. A Naprendszerben a bolygók a Nap körül keringenek, és a bolygóknak lehetnek holdjaik.

[6] Budai A.–Vári Kakas I. (2007): *Számítógépes grafika*. (3.1.3. A modelltér transzformációi, 88), Budapest: INOK.

[9] Hudson A.–Nelson R. (1994): *Útban a modern fizikához*. Budapest LSI Oktatóközpont.

[9] Hudson A.–Nelson R. (1994): *Útban a modern fizikához*. Budapest: LSI Oktatóközpont.

[10] A Lagrange-pontok: <http://astro.u-szeged.hu/ismeret/lagrange.html>

[11] Nemlineáris rendszerek stabilitása. Lyapunov-stabilitás.

[12] Kaczur S. (2010): *Programozási technológia*. Budapest.

[13] Kondorosi K.–László Z.–Szirmay-Kalos L. (1997): *Objektum-orientált szoftverfejlesztés*. Budapest: ComputerBooks.

A tömegvonzásnak más elméleti megközelítései is vannak: Einstein gödörmodellje, Kepler törvényei, illetve differenciál-egyenletrendszer, integrálszámítás is használható a közelítő képlet helyett (csak ideális modell esetén pontszerű a bolygó és gömbszimmetrikus azok tömegeloszlása) [9], illetve ismeretes többféle értelmezés a rendszer/modell stabilitására: Lagrange-pontok [10], Ljapunov-stabilitás [11].

Az MVC-MODELL

A klasszikus megközelítés szerint a szoftveres alkalmazások három, egymástól jól elkülöníthető szereppel rendelkező egységből állnak: modell (model), nézet (view), vezérlő (controller) [12, 13]. A Java nyelv Swing-komponensei az MVC-architektúra szerint működnek.

A vezérlő reagál az érkező eseményre, hozzáfér a modell adatszerkezeteihez, azaz igénybe veszi a modell szolgáltatásait, valamint frissítheti a nézetet. A nézet a vezérlő frissítési kérésére a közvetlenül megkapott adatok alapján, vagy a modelltől elkért adatok alapján frissíti saját magát. A vezérlő határozza meg az alkalmazás, komponens, program működését. Egy modellt több nézet is használhat. A modell közvetlenül is üzenheti a nézetnek, hogy megváltozott. A nézet adja a látványt, amelyet angolul *skin* vagy *look and feel*-nek neveznek.

Tervezés

A Naprendszer égitesteinek ábrázolása a valódi világban előforduló méretük és távolságuk szerint történik azért, hogy a szimuláció stabil legyen. A példában a Nap és a három belső bolygó szerepel, valamint a Hold. Utóbbi igazolja, hogy nem csak Nap-középpontú égitestekre működőképes a modell. A szimuláció diszkrét lépések véges sorozataként valósul meg, az egyes lépések között a bolygók a virtuális térben egyenes vonalú egyenletes mozgást végeznek. Olyan lépésközt kell választani, amely rövid idő alatt kellően nagy

változást képes bemutatni, ilyen például az 1 számítási ciklus / 1 nap érték. 10 képkocka / másodperces megjelenítést feltételezve – melyet egy időzítő biztosít – egy virtuális év kb. 37 másodperc alatt telik el, vagyis a Föld ennyi idő alatt tesz meg egy teljes fordulatot a Nap körül. Az égitestek kezdő pozíciója fiktív, nem függ konkrét dátumtól, együttállástól, méretük a jobb láthatóság érdekében torzított [14].

A program indításakor a szimuláció automatikusan indul, és nincs lehetőség a leállításra. Az alkalmazás felületének tetején foglalnak helyet a kezelő nyomógombok, a többi részt a megjelenítés/transzformált modell tér tölti ki. Futás közben – egyszerű eseménykezelést megvalósítva – lehet változtatni a méretarányt és a nézőpontot, így az ekliptika síkját felülről és elbillentve is ábrázolhatjuk.

Kivételkezelés nem szükséges a programba, mert ez egy önálló demonstrációs eszköz, nem épül rá több elem, nem érhetőek el a szolgáltatásai külső programok számára.

HTML 5 CANVAS ÉS JAVASCRIPT-MEGVALÓSÍTÁS TERVEZÉSE

Ebben a megvalósításban az a cél, hogy különösebb tervezés és elvek nélkül, működő és egyszerű ábrázolási technikát használva, gyors sikerélményt kapjunk. Legyen ez egy megvalósíthatósági vizsgálat, legyen alkalmas a stabil rendszer kikísérletezésére. A keletkező forráskód legyen ésszerűen átlátható. Az egyszerű matematikai modell alapvetően két ötletre épülhet: a modell térben ábrázolhatók az objektumok egy egyenes és ehhez tartozó szög párosával, illetve koordinátákkal. Előbbi esetet elvetjük, mert nehezebbnek találjuk az implementálását. Utóbbi esetben a vektorokkal leírt távolságok és erők sokkal egyszerűbben kezelhetők, a transzformációk mátrixokkal könnyen megvalósíthatók.

A program látvány részéért egy HTML 5 szabványú weboldal felel, ami egy canvas-elemet tartalmaz, mint megjelenítő [15]. Az oldal tetején található 5 gomb, amikkel a méretezést, és a forgatást lehet állítani, illetve a szimulált objektumok képernyő-koordinátáinak megjelenítését ki- és bekapcsolni. Ez utóbbi funkció nagyon hasznosnak bizonyul majd a program fejlesztési

[14] Horváth L.–Szlávi P.–Zsakó L. (2006): *Modellezés és szimuláció*. Budapest: ELTE.

[15] HTML5 Canvas tutorials: <http://www.html5canvastutorials.com>

[12] Kaczur S. (2010): *Programozási technológia*. Budapest.

[13] Kondorosi K.–László Z.–Szirmay-Kalos L. (1997): *Objektum-orientált szoftverfejlesztés*. Budapest: ComputerBooks.

fázisában, amikor nem biztos, hogy rögtön sikerül az égitesteket olyan paraméterekkel ellátni, ami stabil pályán tartja azokat. Ilyen esetekben bár az égitest nem látszik a képernyőn, de a koordinátái alapján mégis kiköveztethető lesz a helyzete a szimulált univerzumban.

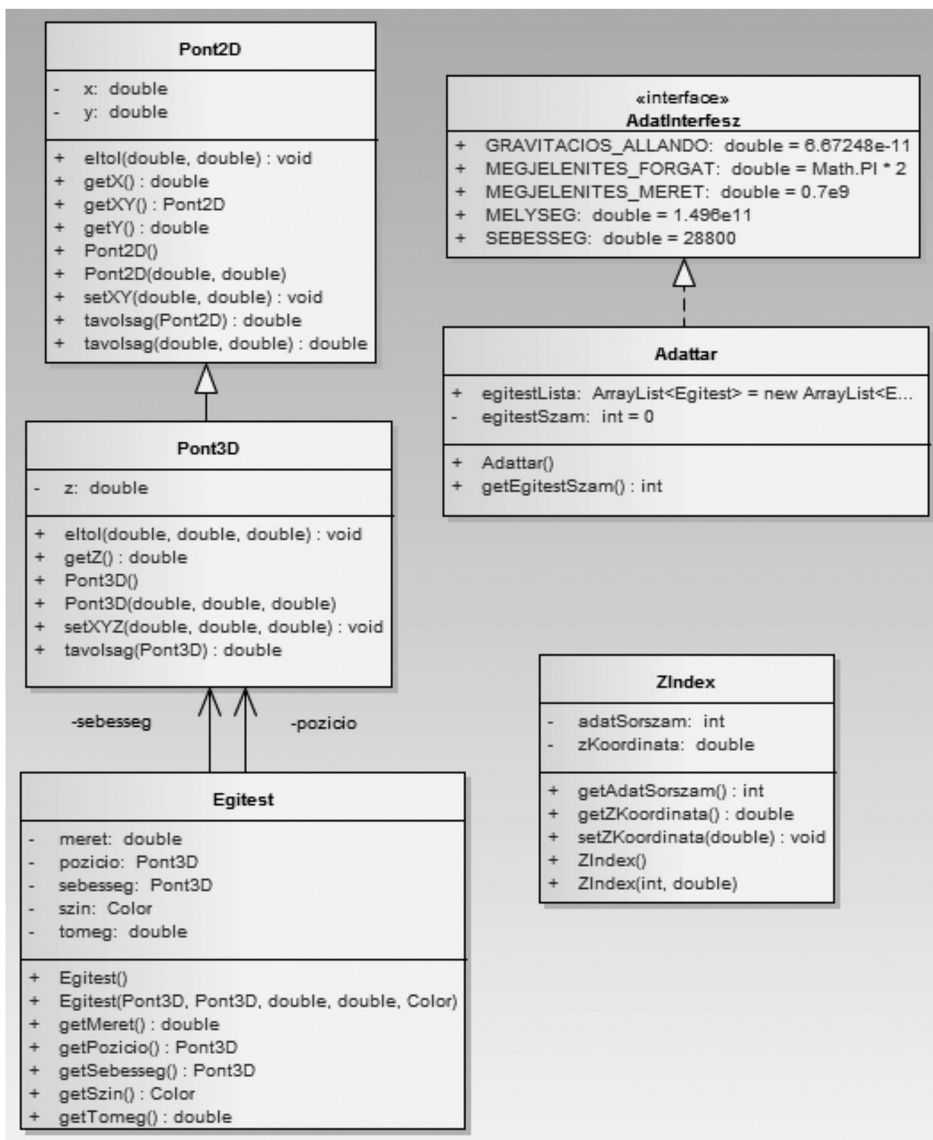
A HTML-kód definiálja a rajzoláshoz szükséges vásznat (canvas) és hozzárendeli a megjelenített nyomógombokhoz saját eseménykezelő metódusait. A JavaScript-rész megvalósítja az égitestek adatszerkezetét egy tömbbel. A vezérlés, számítások megvalósítását JavaScript-metódusok végzik el (főbb funkciók: tömegvonzás, gyorsulás, rajzolás, időzítő és eseménye, hibakeresés-támogatás).

Az alkalmazás futtatásához HTML 5-öt és JavaScriptet támogató böngésző szükséges.

JAVA NYELVŰ MEGVALÓSÍTÁS TERVEZÉSE

Most határozzuk meg másképpen a célt és a szempontokat: a Java-projektben a csomagokat az MVC szerint hozzuk létre, a funkciókat logikusan osszuk szét, csoportosítsuk [12, 13], tartsuk be az objektumorientált szemléletmód elveit, használjunk interfészt, biztosítsuk az egységbezárást, legyen öröklődés, alkalmazzuk a polimorfizmust, legyen szép és elegáns megoldás.

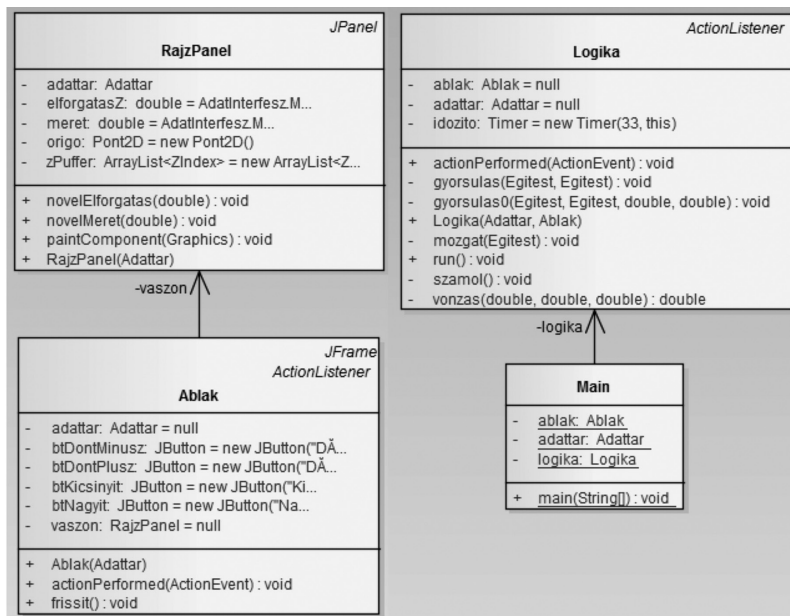
1. ábra. A modell csomag osztálydiagramja.



A modellhez 1 interfész és 5 osztály tartozik (1. ábra). Az *AdatInterfesz* tárolja a modell számításához és megjelenítéshez tartozó konstansait (ezek a szimuláció paraméterei), metódusfejet nem tartalmaz. A *Pont2D* osztály egy kétdimenziós pont sémája, valós x és y koordinátpárral, *eltol()* és *túlterhelt távolság()* metódusokkal. Ennek leszármazottja a *Pont3D* osztály, amely mindezt három dimenzióban biztosítja, valamint pozícióként és sebességvektorként is használható. Az *Egitest* osztályból létrehozott objektumnak van mérete, pozíciója, sebessége, színe és tömege. Az interfészt implementálja az *Adattar* osztály, amelynek *egitestLista* nevű generikus listája elérhetővé és egységesen kezelhetővé teszi a tervben felsorolt 5 égitestet. A *ZIndex* osztályú objektumok az égitestek kirajzolásakor szükséges mélységpufferbeli adatot képesek karbantartani.

A nézet 2 osztályból áll (2. ábra). Az *Ablak* osztály egy *javax.swing.JFrame* leszármazott, az alkalmazás teljes felületét biztosítja, valamint előkészíti az eseménykezelést. Tartalompanelje négy vezérlő nyomógombot tartalmaz és rajta található a rajzpanel, a vaszon. A *RajzPanel* egy *javax.swing.JPanel* leszármazott, amely kapcsolatban áll az adattárral, és kezeli a mélységpuffert. Ez felel a szimulált 3D térben lévő objektumok 2D-beli leképezéséért, figyelembe véve a nézőpont elmozdulását is. A rajzolást a felüldefiniált *paintComponent()* metódus irányítja.

2. ábra. A nézet és a vezérlő csomagok osztálydiagramja.



A vezérlőt 2 osztály valósítja meg (2. ábra). A Main osztály összefogja a projektet, ez a végrehajtás belépési pontja. Szükség szerint átadja az MVC szerinti objektumok referenciáit egymásnak, ezzel biztosítva a kommunikációt közöttük, valamint el is indítja a szimulációt. A Logika osztály képes az égitestek gyorsulásának és vonzásának kiszámítására, az égitestek mozgatására, továbbá a megjelenítésért felelős komponenst megfelelő időközönként értesíti a képernyő frissítésének szükségességéről, ami az alapbeállítás szerint 30 frissítés másodpercenként.

Mindez így egy Java asztali alkalmazás, a nézet módosításával könnyen átalakítható a böngészőben futó Java Appletté. Futtatásához JRE 1.7 szükséges.

[3] A cikkhez tartozó programok: <http://kaczursandor.hu/VV3D>

Megvalósítás

Az implementáció során jól összehasonlíthatók az alapvetően is különböző technológiák. Fejlesztőeszközként a HTML 5 canvas és JavaScript példához a Notepad++-t és a Firefoxot, a Java Swinges projektet pedig a JDK 1.7-re épülő NetBeans 7.x-et használtuk.

Hibakeresés során, a modell adatainak ellenőrzését és a működés helyességének egyszerű tesztelését, debugolást HTML 5 canvas és JavaScript esetén a rajzterületre kiírt szöveges adatokkal, Java Swinges projekt esetében pedig a konzolra történő kiírással oldottuk meg.

Ebben a fejezetben megmutatjuk a forráskódoknak azt a részét, ami megvalósítja a (4) transzformációs mátrix alkalmazását X tengely körüli elforgatásra, a nézőponttól való távolság függvényében az égitest látható méretének kiszámítását, valamint a 3D→2D leképezést.

A teljes és megjegyzésekkel ellátott forráskódok szabadon letölthetők [3], tesztelhetők.

HTML 5 CANVAS ÉS JAVASCRIPT MEGVALÓSÍTÁS

A megjelenítést a 3. ábrán látható HTML-kód végzi el. A felület tartalmazza a megtervezett 5 nyomógombot, valamint láthatók a hozzájuk rendelt eseménykezelő metódusok nevei. A modelltér leképezése a myCanvas nevű objektumra történik.

3. ábra. Forráskód-részlet – HTML 5 canvas- és JavaScript-implementáció – HTML-kód.

```
<div id="myTitle" style="text-align: center;">Tömegvonzás szimuláció (0,1 mp = 1 nap)</div>
<div>
  <span>Nagyítás:</span>
  <button onclick="zoomout()" ">-</button> <button onclick="zoomin()" ">+</button>&nbsp;
  <span>Döntés (0-90):</span>
  <button onclick="rotateout()" ">-</button> <button onclick="rotatein()" ">+</button>&nbsp;
  <button onclick="showCoordinates()" ">Koordináták be/ki</button>
</div>
<canvas id="myCanvas">A böngésző nem támogatja a HTML5 CANVAS-t !</canvas>
```

A 4. ábrán látható, hogy koordinátáinként külön-külön végezzük a transzformációs mátrixba történő behelyettesítést. Az ideális pont nézőponttól való távolságát (a tér mélységét) az égitest méretének kiszámítása során figyelembe kell venni. A 3D→2D leképezés perspektív módon történik, felhasználva a korábban már elforgatott koordinátákat.

4. ábra. Forráskód-részlet – HTML 5 canvas- és JavaScript-implementáció – JavaScript-kód.

```
// calculate rotate around X axis
var rotateX = planets[i].posX;
var rotateY = planets[i].posY * Math.cos(turn_rad) + planets[i].posZ * Math.sin(turn_rad);
var rotateZ = -1 * planets[i].posY * Math.sin(turn_rad) + planets[i].posZ * Math.cos(turn_rad);
// calculate planet's visible size
pSize = (DEPTH * planets[i].size) / (DEPTH - rotateZ);
// calculate perspective
screenX = rate((rotateX * DEPTH) / (DEPTH - rotateZ)) + origoX;
screenY = rate((rotateY * DEPTH) / (DEPTH - rotateZ)) + origoY;
```

JAVA NYELVŰ MEGVALÓSÍTÁS

A megvalósítás során az előre megtervezett osztálydiagram alapján készült el a Java nyelvű programkód. Az MVC-modell szerint elkülönített programrészek külön csomagokba kerültek, ezzel is kiemelve a funkciók szerinti szétválasztást – eleget téve a terv követelményeinek.

Az *Ablak* osztályú objektum elsődleges szerepet tölt be a megjelenítésben, keretbe foglalva a látható komponenseket, vagyis a kezelő nyomógombokat és a modellteret. Az objektum megvalósít egy ActionListener eseménykezelőt, így a program reagálni tud a felhasználó által kiváltott eseményekre.

Az ablak nagyítás és forgatás üzenetek küldésével saját vásznát – és csak azt – frissíti.

Az 5. ábrán megfigyelhető – összevetve a 4. ábrával –, hogy a 3D és 2D pontok objektumai okosabbak, valamint az interfész konstansainak használata is látható.

5. ábra. Forráskód-részlet – Java Swinges-implementáció – Java-kód.

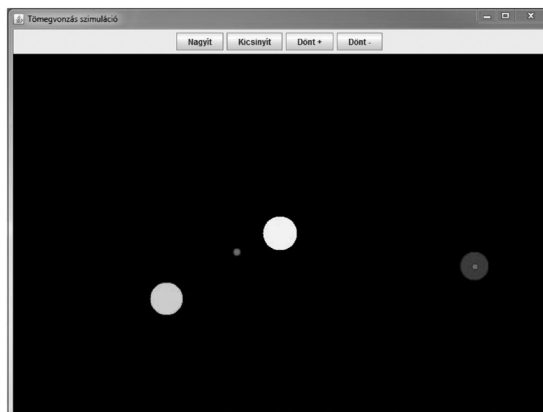
```

prog = new Pont3D(
    b.getPozíció().getX(),
    b.getPozíció().getY() * Math.cos(elforgatasZ) + b.getPozíció().getZ() * Math.sin(elforgatasZ),
    -1 * b.getPozíció().getY() * Math.sin(elforgatasZ) + b.getPozíció().getZ() * Math.cos(elforgatasZ)
);
bolygóMéret = (int) (
    (AdatInterfesz.MELYSEG * b.getMéret()) /
    (AdatInterfesz.MELYSEG - forog.getZ()));
pXY = new Pont2D(
    (forog.getX() * AdatInterfesz.MELYSEG / (AdatInterfesz.MELYSEG - forog.getZ()) /
    meret) + origo.getX() - bolygóMéret / 2,
    (forog.getY() * AdatInterfesz.MELYSEG / (AdatInterfesz.MELYSEG - forog.getZ()) /
    meret) + origo.getY() - bolygóMéret / 2);

```

Az elkészült Java Swinges alkalmazás felhasználói felülete a 6. ábrán látható, nagyon hasonlít erre a HTML 5 canvas és JavaScriptes megvalósítás GUI-ja.

6. ábra. A tömegvonzás-szimuláció Java nyelvű megvalósítása.



Tapasztalatok

A 10 kép/másodperces szimulációs sebesség a HTML canvas és JavaScript programban szerencsés választásnak tűnt, mert így kisebb teljesítményű eszközökön, például Android operációs rendszerű okostelefonokon is jól futott, és nem emésztette fel a processzoridőt.

A JavaScript nyelv gyengén típusos, minden számot dupla pontosságú lebegőpontos számként tárol és dolgoz fel. Ez lehet könnyebbség, mert sok típuskonverzióval nem kell foglalkozni, mivel automatikus. A Java nyelv erősen típusos, így a kötelező és sok lebegőpontos/egész átalakítás miatt észrevehető, hogy a legkisebb égitest (Hold) kissé ugrál.

A továbbfejleszthetőséget tekintve: az OO szempontból is szebb Java megvalósítás könnyebben módosítható és bővíthető, a funkciók jobban csoportosítottak, a felelősségi kör egyértelműen meghatározott. A Java-implementáció megtervezéséhez és elkészítéséhez lényegesen magasabb szintű absztrakciós készség szükséges.

A példaprogramok alkalmasak a különböző tantárgyak, témakörök (Matematika–Lineáris algebra, Fizika, Számítógépes grafika, Virtuális valóság modellezése) közötti kapcsolatok felismertetésére, megerősítésére, a (legalább részben) egymásra épületek felderítésére. A tervek átgondolásával, implementálásával gyors, látványos eredmény érhető el, a sikerélmény hamar jelentkezik.

Továbbfejlesztési lehetőségek

Célszerű ötlet a hardveres gyorsítás és 3D-megjelenítés megvalósítása. Felkínálható lenne a felhasználó számára több paraméter módosítása. Az égitestek lehetnének textúrázhatók is. Az égitestek pozíciója kiinduló helyzetben lehetne valós. A szimuláció szükség esetén lehetne elindítható, leállítható, újraindítható. A terv könnyen implementálható lehet Java3D technológia alkalmazásával, illetve DirectX és/vagy OpenGL támogatással is. Az égitestek pozíciója és mozgása demonstrálhatna/modellezhetne nevezetes együttállást is, mint egy esettanulmány. A program paraméterezhető lehetne konfigurációs fájlból (amelynek formátuma tetszőleges: ini, xml). Fejlettebb matematikai modell is alkalmazható lenne.

Egyszerűen megvalósítható plágiumkeresés nagy létszámú kurzusok beadandó feladataihoz

Összefoglalás: A gyakorlatias képzések fontos eleme az egyetemi, főiskolai hallgatók számára az egyénileg elkészített házi feladatok, beadandók, mérési jegyzőkönyvek kidolgozása. Ezek értékelése nem egyszerű feladat: egy nagyobb, több száz fős kurzus esetén ez a munka rengeteg időt igényel az oktatók részéről.

Sajnos gyakran előfordul az is, hogy a beadott művekről kiderül: plágium. Ez legtöbbször különböző név alatt beadott azonos dolgozatokat jelent. Az ilyesmit nagy létszám esetén emberi munkával szűrni már szinte lehetetlen, hiszen míg a javításhoz szükséges idő lineárisan nő a dolgozatok számával, az összehasonlításé már négyzetesen: mindegyik dolgozatot mindegyikkel össze kell hasonlítani.

A plagizált dolgozatok automatikus felderítését könnyíti a számítógépek teljesítménye. Jelentősen nehezíti azonban a probléma megoldását az a tény, hogy ezen írásművek között általában nincsen algoritmikus módszerrel könnyen vizsgálható tökéletes egyezés. A megírt szövegeket legtöbbször többé-kevésbé átfogalmazva, a programkódokban változók neveit, megjegyzéseket kicserélve próbálnak a plágiumot elkövetők egyedi, vagy egyedinek tűnő műveket előállítani. Az emberi gondolkodás ezt az akadályt könnyen veszi, és felismeri az azonos tartalmat, egy számítógép viszont annál kevésbé. Cikkemben egy egyszerű, de hatékony módszert írok le, amellyel szövegfájlok között lehet felkutatni az azonosakat, és ami fontosabb – a hasonlókat is. Az összehasonlítás azon alapszik, hogy a szövegeket szópárokra bontja, és azokat az így kapott kifejezések halmazaként értelmezi. A hasonlóság mértékének megállapítása ebben az esetben egy egyszerű halmazmetszés művelettel végezhető el. A fő célom a módszer egyszerűségének bemutatása: az algoritmus C++ implementációja alig több, mint 100 kódsorból áll. Ugyanakkor rendkívül gyors is: 250 darab, átlagosan 2–3 oldalnyi szöveget tartalmazó fájl összehasonlításához mindössze egy másodpercre van szüksége.

* BME

Elektronikus Eszközök

Tanszéke

E-mail: czirkos@eet.bme.hu

A bemutatott módszert sikeresen alkalmaztuk egy 250 fős elektronikai tárgyon, amelyen a hallgatóknak a félév során összesen hat mérési jegyzőkönyvet kellett leadni. Az 1500 szövegfájlból álló minta elegendő volt ahhoz, hogy az algoritmusban alkalmazott szópár-halmazok méretének összehasonlítását, mint metrikát értékelni lehessen, és a kvantitatív eredmények vonásait fel lehessen ismerni. A cikkben kitérek a jegyzőkönyvek szövegének néhány jellemző vonására, és azoknak az algoritmusra gyakorolt hatására is. A munka szakmai tartalma kapcsolódik a „Új tehetséggondozó programok és kutatások a Műegyetem tudományos műhelyeiben” c. projekt szakmai célkitűzéseinek megvalósításához. A projekt megvalósítását a TÁMOP - 4.2.2.B-10/1--2010-0009 program támogatja.

Kulcsszavak: Plágium, szöveg összehasonlítás.

Abstract: This article presents a simple method for plagiarism detection, which can be used on natural language documents. The method can find similar text segments, which may be paraphrased to some extent as well. The comparison of documents is based on fingerprinting: each document is represented by a set of word-pairs, which contains all words of the text. The comparison is carried out by examining the intersection of these sets, and relating the resulting sets to the original ones.

Keywords: Plagiarism, text comparison.

Bevezetés

A gyakorlatias képzések fontos eleme a felsőoktatás hallgatói számára az egyénileg elkészített házi feladatok, beadandók, mérési jegyzőkönyvek kidolgozása. Ezek értékelése nem egyszerű feladat: egy nagyobb, több száz fős kurzus esetén ez a munka rengeteg időt igényel az oktatók részéről. Sajnos gyakran előfordul az is, hogy a beadott művekről kiderül: plágium.

A plagizált dolgozatok automatikus felderítését megkönnyíti a nagy teljesítményű számítógépek használata. Jelentősen nehezíti azonban a probléma megoldását az a tény, hogy ezen írásművek között általában nincsen algoritmikus módszerrel könnyen vizsgálható tökéletes egyezés. A megírt szövegeket legtöbbször többé-kevésbé átfogalmazva, a programkódokban változók neveit, megjegyzéseket kicserélve próbálnak a plágiumot elkövetők egyedi, vagy egyedinek tűnő műveket előállítani.

1. ábra. Plágiumgyanús szövegrészletek.

1. A mérés célja egy minimális bonyolultságú, de működőképes áramkörü elem megvalósítása a Mentor Graphics ICStudio tervezőprogramjában, valamint szimulációval ellenőrizni a tervezett elem működőképességét, funkcióját. A feladat egy transzfer gate-es D tároló megvalósítása volt.
2. A mérés során egy nem túl bonyolult áramkört valósítottunk meg a Mentor Graphics ICStudio tervező programjában, valamint szimulációval ellenőriztük a tervezett elem működőképességét és funkcióját. A feladat egy D tároló elkészítése és szimulációja volt.

Az emberi gondolkodás ezt az akadályt könnyen veszi, és felismeri az azonos tartalmat, egy számítógép viszont annál kevésbé. A magyar nyelv változatossága tovább nehezíti a problémát: míg egy olyan nyelvénél, mint az angol, ahol a szavakat nem kell toldalékolni a mondatba illesztéshez, egyszerű megoldás lehet a szavak sztringként történő vizsgálata, a magyar nyelvben ez nehezebben megoldható.

A cikkben egy egyszerű, de hatékony módszert írok le, amellyel szövegfájlok között lehet felkutatni az azonosakat, és ami fontosabb – a hasonlókat is. Az összehasonlítás azon alapszik, hogy a szövegeket szó-párookra bontjuk, és azokat az így kapott kifejezések halmazaként értelmezzük. A hasonlóság mértékének megállapítása egy egyszerű metszés-művelettel végezhető el. A fő cél a módszer egyszerűségének bemutatása: az algoritmus C++ implementációja alig hosszabb 100 kódsornál.

A bemutatott módszert sikeresen alkalmaztuk egy 250 fős tárgyon, amelyen a hallgatónak a félév során összesen hat mérési jegyzőkönyvet kellett leadni. Az 1500 szövegfájlból álló minta elegendő volt ahhoz, hogy az algoritmusban alkalmazott szópár halmazok méretének összehasonlítását, mint metrikát értékelni lehessen, és a kvantitatív eredmények vonásait fel lehessen ismerni. A cikkben kitérek a jegyzőkönyvek szövegének néhány jellemző vonására, és azoknak az algoritmusra gyakorolt hatására is.

A plágiumkeresés módszerei

AZ EMBERI ÉS A GÉPI PLÁGIUMKERESÉS

Az emberi és az automatizált plágiumkeresés nagyon eltérő úton működik. Az ember érti az olvasott szöveget, ezért a tartalmára tud koncentrálni, míg a számítógép alapvetően csak a számára jelentés nélküli szöveggel tud dolgozni. De nem csak ez az egyetlen különbség, hanem a memória jellege is. Az emberi agy asszociatív működésű: egy adott szövegnél viszonylag könnyen felismeri azt, hogy találkozott-e már

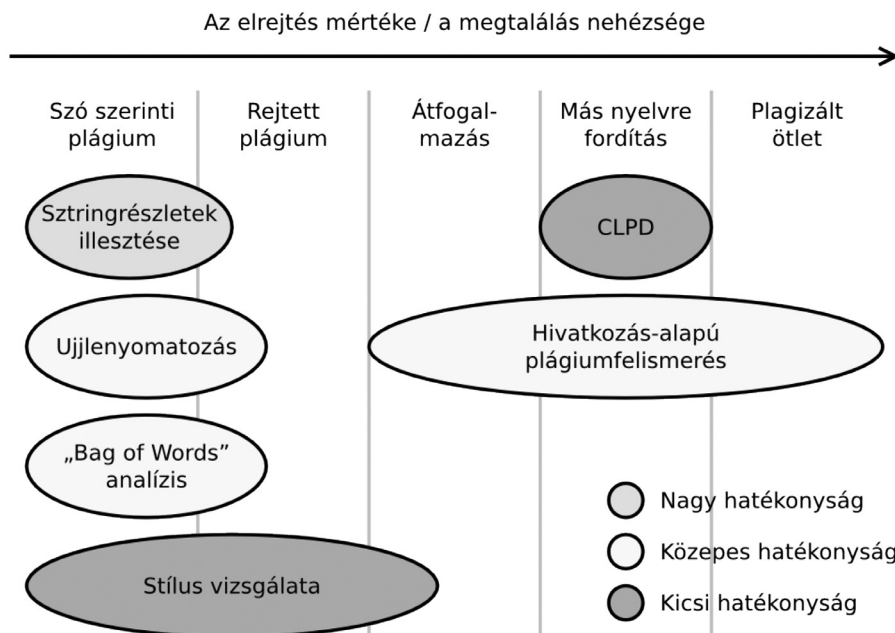
[1]A Wikipedia „Plagiarism detection” szócikke. Megtekintés: 2013. 11. 10. http://en.wikipedia.com/wiki/Plagiarism_detection.

az adott tartalommal vagy nem. Ezzel szemben a gépi keresés lineáris. A mai számítógépek számítási kapacitása azonban sok esetben ki tudja váltani az asszociatív, tartalomra koncentráló működést.

PLÁGIUMKERESŐ ALGORITMUSOK

A plágium detektálására a jellegétől függően eltérő algoritmusok alkalmazhatóak. Az egyes módszerek alkalmazhatóságát a 2. ábra foglalja össze.

2. ábra. A plágiumkereső algoritmusok csoportosítása [1] nyomán.



A plágium a legegyszerűbb esetben szövegrészletek egy az egyben történő átvételét jelenti. Ez algoritmikailag könnyen kezelhető, hiszen csak sztringeket kell összehasonlítani a programnak. A sztringrészletek keresése egyszerűnek tűnik, de igen számításigényes feladat [2].

A megtalált szövegrészletek hosszúsága egyre csökken, ahogy az összehasonlítandó szövegekben átfogalmazások jelennek meg. Ilyen esetben hatékonyabb lehet a keresés, ha nem hosszabb részleteket, hanem kifejezéseket vagy szavakat keresünk. Az ujjenyomatozás elve azon alapszik, hogy mindegyik dokumentumhoz egy kivonatot, ujjenyomatot rendelünk, amely annak szövegét jellemzi, és a keresésnél az ujjenyomatokat hasonlítjuk össze [3]. Hasonló elven működik a Bag of Words-algoritmus, amely szavak, előfordulását vizsgálja. Ennek eredeti célja egy szöveg témájának, kulcsszavainak automatikus meghatározása volt, de plágiumkeresésre is alkalmazható [4].

Ahogy haladunk a parafrázis, a szövegek átfogalmazása felé, úgy csökken ennek hatékonysága is. A legkifinomultabb plágiumdetektáló algoritmusok ezért nem is a szövegrészleteket ellenőrzik, hanem tudományos folyóiratokban gyakran a beküldött cikkek hivatkozásjegyzékét vizsgálják meg [5]. Ez abból a megfigyelésből fakad, hogy minden szerzőre jellemző egy hivatkozáshalmaz, hogy mely más folyóiratok cikkeit, konferenciaticikkeket idéz. Míg egy szöveget átfogalmazni viszonylag kis munka, egy szövegrészlethez új irodalmi hivatkozásokat találni annál nehezebb: ez kézzelfogható munkát, irodalomkutatót igénylő feladat, ezért a plágiumot elkövető szerzőnek ez sokkal kevésbé éri meg. Az ilyen módon történő keresés azonban csak tudományos szövegekre alkalmazható, vagy egyéb olyan helyzetekben, ahol a szövegek hivatkozásokat is tartalmaznak.

HASONLÓ FORRÁSKÓDOK ÉS KÉPEK FELISMERÉSE

Nem csak természetes nyelven írt szövegeket, hanem programnyelvi forráskódokat vagy képeket is felismerhetünk. Az ingyenesen elérhető SIM-program [6] célja az, hogy a bemenetként programkódok között a hasonlóakat megkeresse.

[2] Baker, B. S. (1993): *A theory of parameterized pattern matching: algorithms and applications*. In: Proceedings of the twenty-fifth annual ACM symposium on Theory of computing. ACM.

[3] Stein, B.–Zu E.–Meyer, S.–Potthast, M. (2007): *Strategies for retrieving plagiarized documents*. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM.

[4] Zechner, M. et al. (2009): *External and intrinsic plagiarism detection using vector space models*. In: Proc. of 3rd Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse and 1st International Competition on Plagiarism Detection.

[5] Gipp, B.–Beel, J. (2009): *Citation Proximity Analysis (CPA)-A new approach for identifying related work based on Co-Citation Analysis*. In: Proceedings of the 12th International Conference on Scientometrics and Informetrics.

[6] Grune, D.: *SIM similarity tester*. Megtekintés: 2013. november 10. http://dickgrune.com/Programs/similarity_tester/.

[7] *TinEye*.
Reverse Image Search.
Megtekintés:
2013. november 10. <http://tineye.com/>.

Ezt nem csak a szöveg alapján teszi, hanem a forrásszövegeket tokenizálja, és a már szintaktikailag feldolgozott programokat vizsgálja. Egy ilyen eljárásnak nem csak a programkódok lopásának felderítése lehet a célja, hanem egy nagyobb projektben a kód-duplikáció felderítése is. A közismert TinEye- képkereső [7] pedig az általa indexelt képeket ujjlenyomatokkal látja el, amely alapján az átméretezett, levágott képek megtalálhatóak.

A magyar nyelvű szövegekre alkalmazható algoritmus

Az algoritmus célja a hasonló szövegrészeket tartalmazó fájlok automatikus felderítése. A bemenet formátumával nem foglalkozunk, a feldolgozandó dolgozatokat szövegfájlunk tekintjük. Az algoritmus elvárt kimenete a párba állított dolgozatok hasonlósági mérőszáma, egészen pontosan az összehasonlított párok hasonlóság szerint csökkenő sorrendbe rendezett listája. A hasonlóságra nehéz formális definíciót adni, így azt csak köznapi értelemben definiáljuk: azoknak a szövegpároknak, amelyek valószínűsíthetően egy tőről fakadnak, magas pontszámot kell kapniuk. Mivel a felhasználó a megtalált pároknál a plágium valószínűsíthető tényét egy beleolvasással könnyedén ellenőrizheti, néhány fals pozitív találatot megengedünk.

AZ ALGORITMUS MŰKÖDÉSÉNEK ALAPJA

A megvalósított algoritmus az alábbi ötletre épül. Tekintsük a következő két szövegrészletet (3. ábra), amelyek áramkör-szimuláció laboratórium mérési jegyzőkönyveiből származnak.

3. ábra. Az összehasonlítás magyarázatához.

1. Inverter megvalósítása N-MOS és p-MOS-sal: [kép] Egy n és egy p típusú növekményes tranzisztorból áll. A 2 tranzisztort egyszerre vezéreljük. Állandósult állapotban a két tranzisztor közül mindig csak az egyik vezet, a másik lezárt.
2. A negált áramkör az alábbi képpel megvalósítható: [kép] Egy n-MOS és egy p-MOS-növekményes tranzisztort használunk. A 2 tranzisztort egyszerre vezéreljük. Állandósult állapotban mindig csak egy tranzisztor vezet, a másik pedig lezárt.

A két szövegrész érezhetően ugyanaz: nem csak jelentésükben egyeznek meg, hanem a megfogalmazás is hasonló. Különösen a két tranzisztorról szóló mondat árulkodó, ugyanis ez nemcsak hogy szóról szóra, de stílusban is megegyezik. A magyar nyelv helyesírási szabályai szerint a „2” számot betűvel ki kellett volna írni a szövegben.

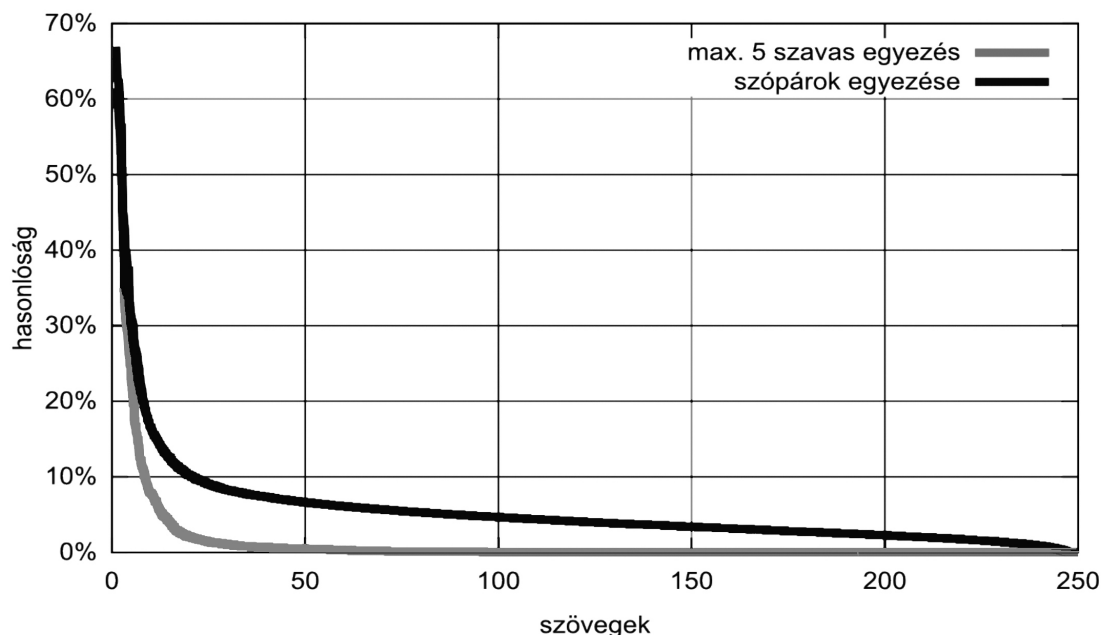
Ugyanazon témáról szóló szövegek plágiumkeresésére a Bag of Words-algoritmus nem alkalmazható közvetlenül. Nem tekinthetjük a szövegeket szavak halmazának, ugyanis bizonyos szavak (jelen példában: mérés, tranzistor, vezet, lezárt) jogosan szerepelhetnek az összes szövegben. A közös szópárok, -hármak, -négyesek azonban már árulkodóak, mert azok azt jelentik, hogy egyforma megfogalmazásról is van szó. Egy öt szó hosszúságú közös szekvencia egy teljes egyező mondatot jelenthet. Az összehasonlítások ezért a legalább két szó hosszúságú kifejezéseket vesszük alapul, és ezeket tekintjük egy szöveg ujjlenyomának. A közös kifejezéspárokhoz büntetőpontot rendelünk, az alábbi képlet szerint:

$$H_{i,j} = \sum_{n=2}^{\max} \frac{|K_{i,n} \cap K_{j,n}|}{|K_{i,n}|} f^{n-2} \quad (1)$$

ahol $H_{i,j}$ a két szöveg hasonlóságának mérőszáma, $K_{i,n}$ az i -edik szöveg n hosszúságú kifejezéseinek halmaza, f pedig egy hatványalap, amellyel az egyre hosszabb közös részkifejezéseket súlyozzuk. (Pl. $f=2$ esetén négyzetesen nő az egyre hosszabb közös kifejezések súlyozott pontszáma.) Tehát az i -edik szöveg lényegében annyira hasonlít a j -edikre, amennyi részkifejezése megtalálható a j -edik szövegben is. A halmazmetszés művelet határozza meg a közös részkifejezések halmazát, és a metszet halmaz méretét arányítja a képlet a szöveg teljes terjedelméhez.

A módszert és az (1). képletet alkalmazva az előbb említett szimulációs mérési jegyzőkönyvek szövegein, a 4. ábrán látható grafikont kaptuk a mérőszámok csökkenő sorrendbe rendezésével. A grafikon függőleges tengelye a hasonlóságot méri, a vízszintes tengely pedig az összehasonlított szövegeket mutatja. (Ez a tengely az eredeti eredményekhez képest négyzetgyök szerint skálázódik, mivel a közel 250 szöveg összesen 62500 szövegpárt eredményez. Itt arra voltunk kíváncsiak, hány olyan szöveget tartalmaz a halmaz, amely nem egyedülálló, vagyis nem önálló munka.) Az ábrázolt függvénynek egy határozott töréspontja van $x=20$ körül; ez alatt meredeken emelkedik, e fölött pedig majdnem konstans, és közelítőleg nulla. Ez a töréspont arra utal, hogy a szövegpárok jellege az adott pontban jelentősen megváltozik. A szövegekbe beleolvasás igazolja a sejtést, hogy az (1) képlet meg tudja találni a hasonló szövegrészeket. Az előbbi két példa is a megvalósított program által megtalált párokból származik.

4. ábra. Az összehasonlított szövegek hasonlósági mérőszámai csökkenő sorrendbe rendezve



AZ ALGORITMUS PARAMÉTEREZÉSE ÉS EGYSZERŰSÍTÉSE

Az algoritmusnak van két paramétere, ezek a leghosszabb vizsgált részkifejezést meghatározó max. és a kifejezések hossza szerinti súly, f . Ezek beállítása empirikus úton is történhet, de a tapasztalat azt mutatja, hogy a végeredménynek csak a kvantitatív tulajdonságaira vannak hatással, a kvalitatív tulajdonságait nem módosítják. Az ábra fekete görbéje (szópárok egyezése) a $\text{max}=2$ esetet mutatja, vagyis azt, amelynél az algoritmus a kifejezések hosszúságát egyáltalán nem veszi figyelembe. (Ha $\text{max}=2$, akkor a kifejezés f^{n-2} szorzója is kiesik, mivel ilyenkor csak az $n=2$ eset jelenik meg az összegzésben.) Látható, hogy ez a görbe ugyanolyan jellegű, csak a nem plagizált ($x>20$) szövegeknél lassabban tart a nullához.

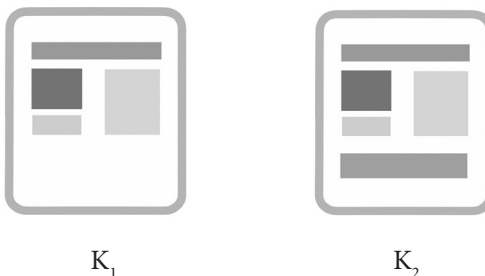
Mivel a cél csak a hasonló szövegek megtalálása, a két függvény határozott töréspontja pedig ugyanazon az x koordinátán van, az összehasonlítás képletét az alábbiak szerint egyszerűsíthetjük:

$$H_{i,j} = \frac{|K_i \cap K_j|}{|K_i|} \quad (2)$$

ahol K_i az i -edik szöveg szópárjainak halmaza, $H_{i,j}$ pedig a két szöveg hasonlóságának mérőszáma. Így az algoritmus jelentősen gyorsabb lehet, mivel jóval kevesebb halmazt kell eltárolni és metszeni. A két szónál hosszabb egyező kifejezések emellett az egyszerűsítés mellett is nagyobb pontszámot kapnak a kétszavas egyezéseknél, hiszen egy n szavas egyezés $n-1$ darab kétszavas egyezésként jelenik meg. Például a „2 tranzisztort egyszerre vezéreljük” mondat a program számára a „2 tranzisztort”, „tranzisztort egyszerre” és „egyszerre vezéreljük” szópárként jelenik meg a szövegek K halmazaiban.

Megfigyelhető, hogy a fenti módon definiált hasonlóság nem rendelkezik a köznapi értelemben vett hasonlóság szimmetria-tulajdonságával. Ott ugyanis azt szoktuk gondolni, hogy ha A hasonlít B -re, akkor B is hasonlít A -ra. Ez a szövegeknél is igaz, de különböző terjedelmek esetén a mérőszámok ennek ellenére nagyon eltérhetnek egymástól.

5. ábra. Különbőle terjedelmű szövegek hasonlósága.

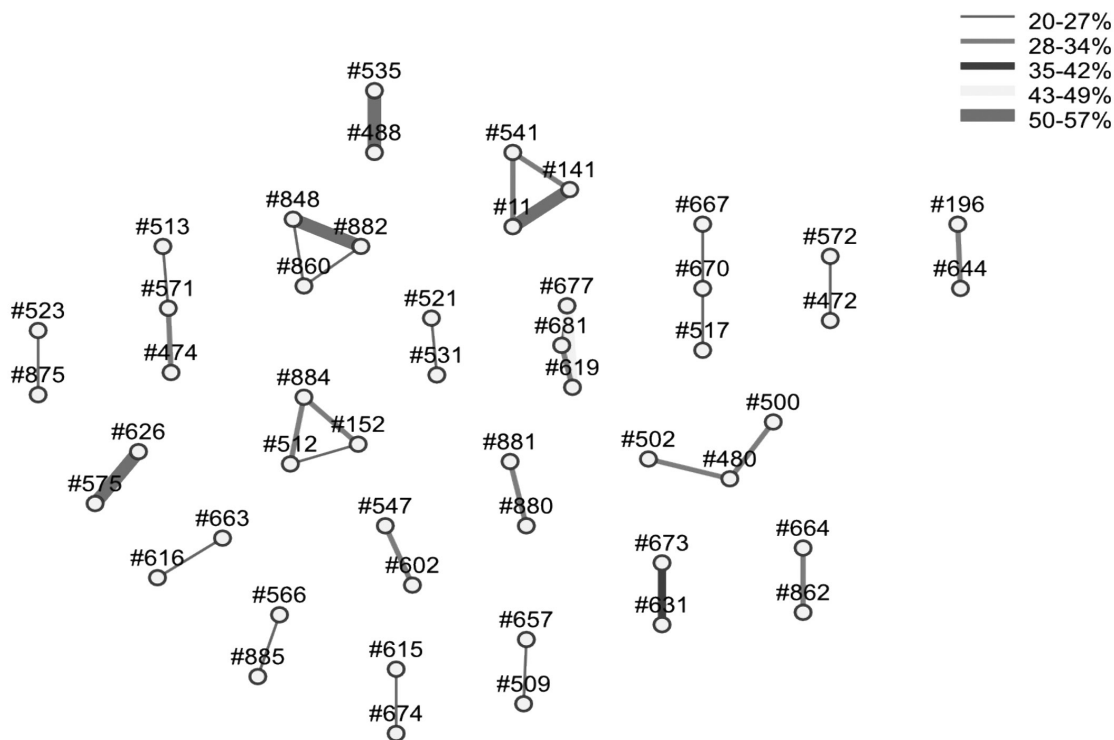


Az 5. ábrán két képzeletbeli szöveg látható, amelyek ugyanúgy kezdődnek, de a második szöveg egy bekezdéssel ki van egészítve az elsőhöz képest. Az algoritmus számára ezek K_1 és K_2 halmazként jelennek meg, amelyek metszete ugyanakkora számosságú, de az első szöveg vizsgálatakor a metszet számosságát az első szöveg méretéhez viszonyítjuk, a második szöveg vizsgálatakor pedig a második szövegehez.

Így azt mondhatjuk, hogy az első szöveg 100%-ban hasonlít a másodikra, mivel annak részei 100%-ban megtalálhatók a másodikban; míg a második szöveg terjedelmének csak 70%-a található meg az elsőben.

A hasonlóságfüggvény töréspontját vizsgálva könnyedén megbecsülhetjük azt a küszöbértéket, amely feletti hasonlóságot mutató szövegeket vizsgálnunk kell a további plágiumkeresés szempontjából. Az egymáshoz hasonló szövegek gráfba rendezésével további betekintést nyerhetünk a tapasztalt plágium jellegébe. A gráf csúcsai a vizsgált szövegek, az élek pedig a küszöbérték feletti hasonlóságot, esetleg annak súlyát is reprezentálhatják. A megjelenített gráf azt is megmutatja, hogy nem csak szövegpárokról van szó, hanem egyes esetekben akár hármasokról is: vagyis hogy kettőnél többen adták be ugyanazt a megoldást.

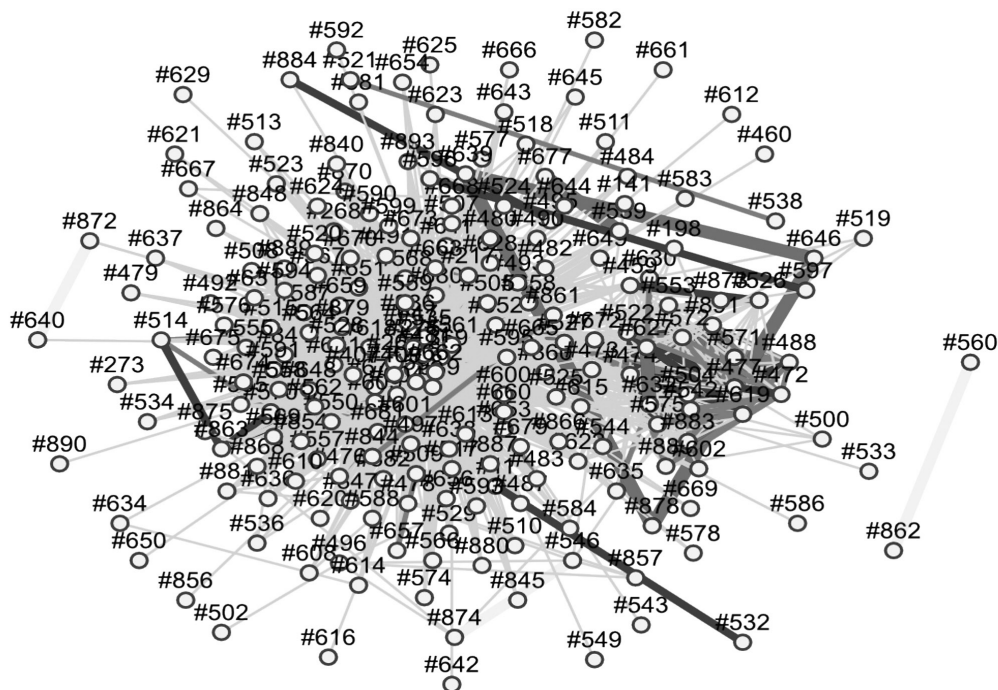
6. ábra. A hasonló szövegek egy gráfon



A KÖZÖS FORRÁSOK PROBLÉMÁJA

A fent bemutatott algoritmusnál gond lehet néhány olyan szöveggel, amely egy nagy évfolyamos házi feladat megoldásaként született. Egy másik laboratóriumi feladat mérési jegyzőkönyvein tesztelve a programot, a 7. ábrán látható hasonlósági grafikont kaptuk.

7. ábra. A hasonlósági viszonyok gráfja a mérési segédletből kimásolt szövegeknél.



A 8. ábra mutatja ezek hasonlósági görbéjét. Láthatóan a görbe nem tart olyan hirtelen a nulla hasonlósághoz, mint az előbb vizsgált szövegeknél. A gráf pedig azt mutatta, hogy a szövegek háromnegyede nem csak hasonló párokból áll, hanem ezek a párok egy nagy, kb. 200 csúcspontból álló klikket alkotnak a gráfon. A szövegek vizsgálata is ezt támasztotta alá. Kiderült, hogy a dolgozatot beadók tényleg egy közös forrásból másoltak ki szövegrészeket. Ez a közös forrás pedig a mérési segédlet volt, amelyet a tantárgy honlapjáról töltöttek le.

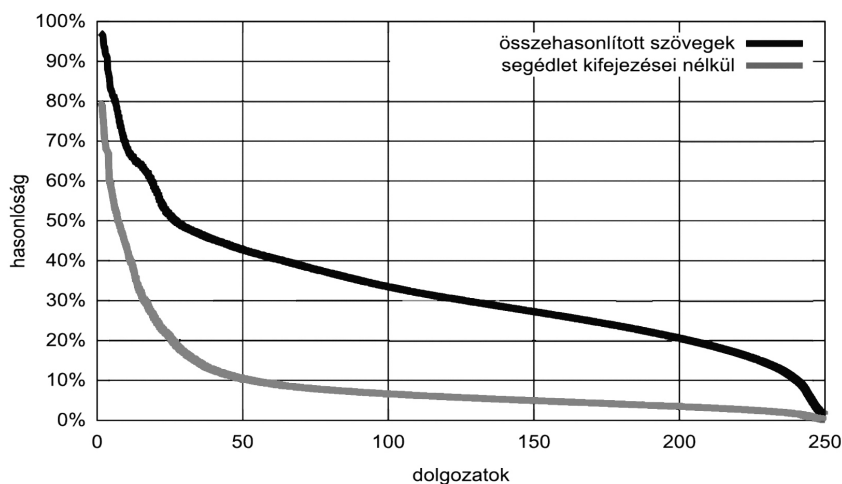
[8]
Stein, B.–
Koppel, M.–
Stamatatos,
E. (2007):
*Plagiarism
Analysis.*
Authorship
Identification
and Near-Du-
plicate Detec-
tion PAN'07,
SIGIR Fo-
rum 41 (2).

A hasonlóságot meghatározó mérőszám képletét ezért az alábbi módon módosítottuk:

$$H_{i,j} = \frac{|K_i \cap K_j - S|}{|K_i - S|} = \frac{|(K_i - S) \cap (K_j - S)|}{|K_i - S|} \quad (3)$$

ahol S a kiadott segédanyag szópárjainak halmazát jelenti. Tehát az összehasonlításakor nem vesszük figyelembe azokat a szópárokat, amelyek szerepelnek a mérési segédletben is; se a halmazok metszésekor (a számlálóban), se a szöveg terjedelmének meghatározásakor (a nevezőben). Ezzel a módosítással az eddigi intrinzik, a szövegeket csak egymás között összehasonlító algoritmusról áttértünk egy extrinzik megoldásra [8], mert egy külső, „eredeti” szöveget is vizsgálunk. A változtatás után a szövegek összevetésekor a várt eredményt kaptuk, a hasonlóság függvények ugyanolyan gyors lefutásúak voltak, mint az első esetben, és határozottan el lehetett különíteni az egyedi, illetve a plagizált szövegeket. Az eredeti, és a módosított képlet által adott eredmény közötti különbség látható a 8. ábrán. (Az algoritmus hatékonysága megmarad, ha az ekvivalens átalakítás alapján az implementációban eleve S szópárjai nélkül tároljuk a halmazokat.

8. ábra. Hasonlósági függvények, a mérési segédlet (közös forrás) szópárjainak szűrése nélkül, és azzal együtt



A bemutatott módszer gyakorlati alkalmazása

AZ ALGORITMUS HATÉKONYSÁGA

Az algoritmus lépésszáma naív megvalósítás esetén $O(n^5)$, amely az alábbi tényezőkből adódik:

- Minden dolgozatpár összehasonlítása: $O(n^2)$.
- A halmazok metszése: $O(n^2)$.
- A halmazok metszése közben a sztringek összehasonlítása: $O(n)$.

A viszonylag lassú működés azonban jelentősen gyorsítható, az alábbi észrevételek figyelembe vételével:

- A halmazok megvalósítását rendezett tárolóra érdemes építeni, mert akkor a metszés művelete összefésüléssel elvégezhető, és csak $O(n)$ lépés szükséges hozzá.
- Bár a sztringek összehasonlítása nem lassítja jelentősen a működést (mindegyik szópárról két-három betű összehasonlítása után kiderül, ha nem egyformák), teljesen elkerülhető ez a művelet. A hasonlóságok meghatározásához ugyanis nincs szükség a metszet-halmaz meghatározására, csak annak számossága szükséges a képletben. Ezért a szövegek beolvasásakor megismert szópárok egész számokkal helyettesíthetők, és a K halmazokban elegendő azokat tárolni. Az egész számok összehasonlítása pedig $O(1)$ időben történik. A hozzárendelő táblázat felépítése pedig csak dolgozatonként, nem dolgozatpáronként történik.

Így a lépésszám $O(n^3)$ -ra csökken. Egy hatékonyan kódolt, C nyelvű megvalósítással 250 szöveg összehasonlítása, amelyek egyenként átlagosan ezer szó terjedelműek, mindössze 1 másodpercet igényel egy Core2Duo processzorral rendelkező számítógépen.

ALKALMAZÁSI LEHETŐSÉGEK

A bemutatott algoritmus olyan szövegeken alkalmazható hatékonyan, amelyek célzottan egy adott témáról íródtak – ilyenek a példaként említett mérési jegyzőkönyvek, házi feladatok is. Ezek terjedelme néhány gépelt oldal, ezer-kétezer szó körül lehet. Nagyobb terjedelmű szövegek esetén figyelni kell arra a tényre, hogy a hasonlóságot a szövegek egész terjedelméhez képest viszonyítja a módszer – azaz egy szakdolgozatnyi, 30–50 oldalas írásba beillesztett néhány oldalas plágiumot alacsony pontszámmal fog csak tudni jelezni. Ilyen jellegű probléma esetén érdemes lehet a szöveget kisebb, néhány oldalas részekre bontani, és úgy végezni el a keresést.

Lokális extrapolálás vezeték nélküli szenzorhálózatokban

**BME Távközlési és Médiainformatikai Tanszék*

E-mail: ollos@tmit.bme.hu

***BME Távközlési és Médiainformatikai Tanszék*

E-mail: vida@tmit.bme.hu

Összefoglalás: Napjaink számítástechnikájának egyik legjelentősebb trendje a miniaturizáció: egyre kisebb méretű eszközökbe (laptop, okostelefon, szenzor) egyre nagyobb intelligenciát (számítási és tárolási kapacitást, kommunikációs képességeket) vagyunk képesek beépíteni. A rendkívül gyors technológiai fejlődéssel viszont nem minden téren tudjuk tartani az iramot. Jelentős lemaradásban van például az energiatárolás, ezért a különböző vezeték nélküli eszközök energia-hatékonysága kiemelt figyelmet érdemel. A többcellás telepek a leggyakrabban használt energiaforrás már a 18. század óta, amikor Alessandro Volta bemutatta az első telepét. Sajnos e klasszikus telepek kapacitása csak úgy növelhető, ha az elektrolit és elektróda anyagának mennyiségét növeljük, ami egyben növeli a telep fizikai méretét is. Mivel a vezeték nélküli szenzorok épp az ellenkező irányba, a miniaturizáció felé indultak, ráadásul az alkalmazott telepek általában nem újratölthetőek és nem cserélhetőek, az energia-hatékony működés kiemelten fontos feladattá vált. A szenzor csomópontok energiafelvétele az adott hardver valamint a futtatott szoftver komponensektől függ; ez utóbbira fókuszálunk ebben a cikkben. Több különböző eljárást dolgoztak ki az energia-hatékonyság növelése érdekében. Ilyenek az adattömörítő- és aggregációs módszerek, energiaérzékeny útválasztás, valamint alvásütemező eljárások. Az alvásütemező eljárások kivételesen hatékonynak bizonyultak, hiszen nagyságrendekkel képesek megnövelni a szenzorhálózatok élettartamát. Ebben a cikkben egy adaptív, elosztott és skálázható alvásütemező rendszert mutatunk be, mellyel jelentős energia megtakarítás érhető el a mintákban lévő mindenkorli redundanciára támaszkodva úgy, hogy a mintavételezési hiba szignifikánsan nem romlik. A cikkben javasolt architektúra adaptív, mivel képes dinamikusan kiaknázni a szenzorok mintáiban rejlő redundanciát annak érdekében, hogy minél nagyobb energia-hatékonyságot értsünk el, miközben a mintavételezés hibája jelentősen

nem növekszik. Az architektúra egy lineáris, véges impulzus-válaszú (FIR) szűrőt alkalmaz, melyet a rendszer folyamatosan követ, adaptálja súlyait, és a hatékonyság függvényében végez különböző méretű előrejelzéseket (lokális extrapolációt), majd az előrejelzés által lefedett időintervallumra a csomópont alvó üzemmódba vált. A modellt valós környezetben elemezzük, majd összehasonlítjuk egy nulladrendű tartóval (ZOH) ellátott extrapolátorral, valamint egy hasonló, korábban publikált megoldással (LE), mely konstans számú előrejelzést alkalmaz.

Kulcsszavak: Vezetéknélküli szenzorhálózatok, alvásütemezés, lokális extrapolálás, FIR-szűrő.

Abstract: Energy is the single most important resource and constraint in a wireless sensor network, since in most applications the lifespan of a sensor node ends when its energy source is depleted; therefore energy-efficient operation leads to an extended network lifetime. Sleep scheduling solutions proved to be exceptionally effective for achieving these goals. Numerous such algorithms have been proposed and examined, but virtually without any considerable support for dynamic event-driven systems. Dynamically occurring events, the constantly changing environment or the presence of mobile elements in the architecture make existing sleep scheduling techniques far too rigid. In such environments dynamic adaptation is desired through learning.

In this paper we address the problem of energy efficient sensing by adaptively predicting various number of samples within a user specified error bound where the node can enter sleep mode for the time covered by the prediction. This is done without the external assistance of the base station or other nodes in the vicinity. This approach enables scalable distributed sleep scheduling even in a dynamically changing environment. The scheme is based on an adaptive finite impulse response digital filter where there is no need for offline pre-computations or dedicated phases. We compare our technique to the well-known ZOH model in a real world application; provide system order sensitivity analysis and discuss simulation results.

Keywords: wireless sensor network, sleep scheduling, local extrapolation, adaptive FIR filter.

[1] Cao, Q.–Abdelzaher, T.–He, T.–Stankovic, J. (2005): *Towards Optimal Sleep Scheduling in Sensor Networks for Rare Event Detection*. In Proceedings of IPSN2005, The Fourth International Symposium on Information Processing in Sensor Networks, pages 20–27. Los Angeles.

[2] He, T.–Krishnamurthy, S.–Stankovic, J.–Abdelzaher, T.–Luo, L.–Stoleru, R.–Yan, T.–Gu, L.–Hui, J. (2004): *Energy-efficient Surveillance System Using Wireless Sensor Networks*. In Proceedings of ACM Mobisys, pages 270–283. Boston.

[3] Hsin, C.–Liu, M. (2004): *Network Coverage Using Low Duty-Cycled Sensors: Random and Coordinated Sleep Algorithms*. In Proceedings of IPSN, The Third International Symposium on Information Processing in Sensor Networks, pages 433–442. Berkeley.

[4] Wang, X.–Xing, G.–Zhang, Y.–Lu, C.–Ple, R. Gill, C. (2003): *Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks*. In Proceedings of ACM Sensys, pages 28–39. Los Angeles.

Bevezető

Az utóbbi néhány évben több különböző eljárást is kidolgoztak az energiahatékonyság növelésére. Ilyenek a tömörítő, és aggregációs módszerek, az energiaérzékeny útválasztás, vagy az alvásütemező eljárások, melyek nagyságrendekkel képesek növelni a szenzorhálózatok élettartamát [1, 2, 3, 4]. Az alvásütemezés módszere potenciálisan kimagasló energiamegtakarítást ígér.

Az energiahatékonysággal foglalkozó cikkek kezdetben a hibátűrésre [5] vagy az energiahatékony útválasztásra fókuszáltak [6], melynek lényege elkerülni azon utakat melyeken (relatívan) lemerült szenzorok találhatók. Az alvásütemező eljárások egy viszonylag új megközelítést jelentenek [3]. Az utóbbi években több cikk tárgyalt különböző megoldásokat. T. Yan lokalizált alvásütemezést javasol [7] elosztott entitásdetektálásra alapozva, mely szignifikáns csomópont-koordinációt valamint rádiós kommunikációt igényel. Hsin és Liu [3] rendszerelméletre valamint prototipizálásra épít, míg Cao [1] ritka események detektálásával foglalkozik. Az említett megoldások viszont statikus és nem adaptív módszerek, így nem támogatnak sem mobilitást sem dinamikus eseményvezérelt hálózatokat.

Cikkünkben egy adaptív, elosztott és skálázható alvásütemező rendszert mutatunk be, ami egy korábban már általunk publikált eljárás [8] továbbfejlesztése. A rendszer célja, hogy olyan alvásütemezést valósítson meg, mellyel a mintavételezési hiba ne romoljon, de ugyanakkor szignifikáns energia megtakarítást érjen el. A korábban publikált rendszerre LE (lokális extrapoláció) rövidítéssel hivatkozunk majd [8], az ebben a cikkben közölt rendszerre pedig LE2 rövidítéssel. Mindkét eljárás adaptívan koordinálja az alvási fázisokat külső információ nélkül (bázis állomás függetlenek) a mintákban lévő mindenkorai redundanciára támaszkodva.

1.ábra. A lokális extrapolációs eljárás algoritmus.

Require: $SysOrder, \mu, U_{err}, \{At_p, Bt_p\}$

```

1: init  $\underline{e} \leftarrow ones(SysOrder, 1)$ ;
2: init  $\underline{b} \leftarrow [0, 1, 0, \dots, 0]^T$ ;  $\underline{x} \leftarrow [1, 0, \dots, 0]^T$ ;
3: while (true) do
4:    $y \leftarrow \underline{b}^T \underline{x}$ ; %előrejelzés
5:    $l \leftarrow sample()$ ; %környezet mintavételezése
6:    $x(3 : end) \leftarrow x(2 : end-1)$ ;  $x(2) \leftarrow l$ ;
7:    $e(2 : end) \leftarrow e(1 : end-1)$ ;  $e(1) \leftarrow (l - y)^2$ ;
8:    $\underline{b}^{new} \leftarrow \underline{b}^{old} + 2\mu \underline{e}$ ; %szűrő adaptálás
9:   if ( $mean(\underline{e}) \leq U_{err}$ ) then
10:     $t_p = round((U_{err} - mean(\underline{e})) * At_p + Bt_p)$ ;
11:    for  $i = 1$  to  $t_p$  do
12:       $y_p(i) \leftarrow \underline{b}^T \underline{x}$ ; %előrejelzés
13:       $x(3 : end) \leftarrow x(2 : end-1)$ ;  $x(2) \leftarrow y_p(i)$ ;
14:    end for
15:     $send(to\ bts, y_p)$ ; %többszörös előrejelzést BTS-nek
16:     $GoToSleepMode(t_p)$ ; %alvás  $t_p$ -ig
17:  end if
18: end while

```

A javasolt eljárás

Cikkünkben bemutatjuk a javasolt lokális extrapolációs eljárást. A cél tehát, hogy egy adott szenzor mérési eredményeinek redundanciájából kiindulva minél hosszabb időre alvó állapotba küldhessük a szenzort úgy, hogy erre az időszakra a kimaradt méréseket a korábbi eredményekből extrapoláljuk. Az algoritmus három valódi paraméterrel ($SysOrder, \mu, U_{err}$) és két becsülhető paraméterrel (At_p, Bt_p) rendelkezik. Ebben a kontextusban valódi paraméter alatt azokat a paramétereket értjük, melyekre nem adtunk becslési eljárást. Empirikusan fogalmazva a $SysOrder$ paraméter az adaptív szűrő fokszámát határozza meg, más szóval mennyi múltbeli minta alapján becsljük meg az aktuális mintát. A μ az adaptációs lépések nagyságát, avagy bátorságát határozza meg mivel nem biztos, hogy jó irányba adaptálunk (az irány becslt érték).

[5] Ye, F.–Zhong, G.–Cheng, J.– Lu, S.– Zhang, L. (2003): *Peas: A Robust Energy Conserving Protocol for Long-Lived Sensor Networks*. In Proceedings of ICDCS, 23rd IEEE International Conference on Distributed Computing Systems, pages 28–37. Providence.

[6] Ganesan, D.–Govindan, R.–Shenker, S.– Estrin, D. (2001): *Highly Resilient Rnergy-Rfficient Multipath Routing in Wireless Sensor Networks*. In ACM SIGMOBILE Mobile Computing and Communications Review, pages 11–25.

[7] Yan, T.– He, T.–Stankovic, J. (2003): *Differential Surveillance for Sensor Networks*. In Proceedings of ACM Sensys, pages 51–62. Los Angeles.

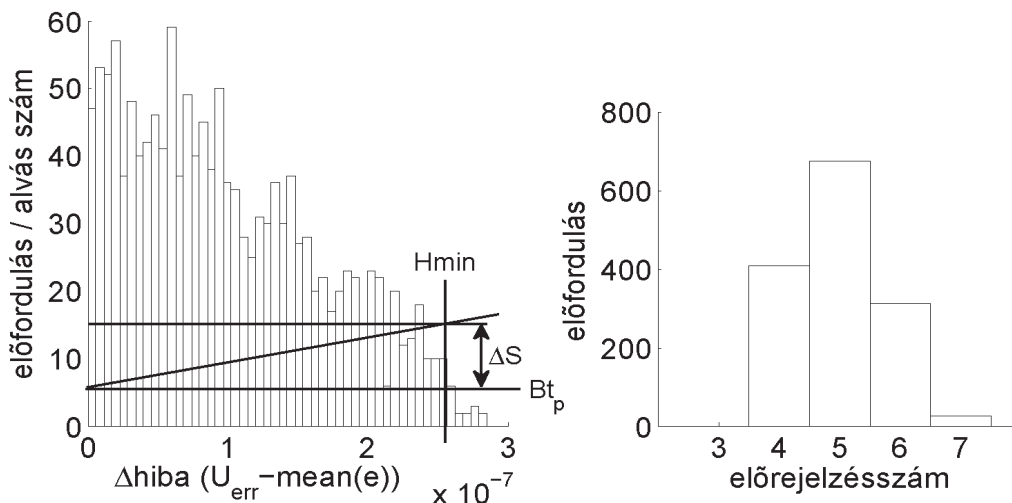
[8] Öllös, G.– Vida, R. (2009): *Adaptive Local Extrapolation in Event Driven Wireless Sensor Networks*. In Proceedings of BioSense09, Novi Sad, Serbia.

Az U_{err} pedig a felhasználó által specifikált maximális várható mintavételezési hibát definiálja, vagyis mennyi hibát engedünk meg a minták approximációja során.

Mint látni fogjuk az At_p és Bt_p paraméterek (melyek lineáris kapcsolatot teremtenek a hiba és az alvászám között) viszonylag könnyen becsülhetőek (a következő fejezetben erre fókuszálunk) azonban a μ batorsági faktor és a *SysOrder* rendszer fokszám már nehezebben, vagy csak konvergencia korlátok adhatók meg. A felhasználó által specifikált hiba U_{err} pedig attól függ milyen várható hibát hajlandó a felhasználó elfogadni a megspórolt energiáért cserébe, így ez a paraméter szubjektív.

Az 1. ábra specifikálja az LE2-eljárást. Első lépésben létrehozuk hibavektort egyesekkel kitöltve (kezdetben nagy hibát feltételezünk), ennek mérete megegyezik a szűrő fokszámával. Ez a hibavektor egy FIFO-ként működik és folyamatosan csúsztatjuk benne az adaptálás során elkövetett négyzetes hibákat (1. sor). Ezt követi a szűrő kezdeti paramétereinek beállítása, ami azt a célt szolgálja, hogy az első előrejelzés megegyezzen az aktuális mintát megelőző értékkel, vagyis kezdetben nullával (2. sor). Ezt követi a legutóbbi minták FIFO vektorának létrehozása mely egy egyest (bias szorzó mindig egyes) majd ezt követően (múltbeli minták) csupa nullát takarnak (2. sor).

2.ábra. Adaptációs hiba és előrejelzésszám-histogram.



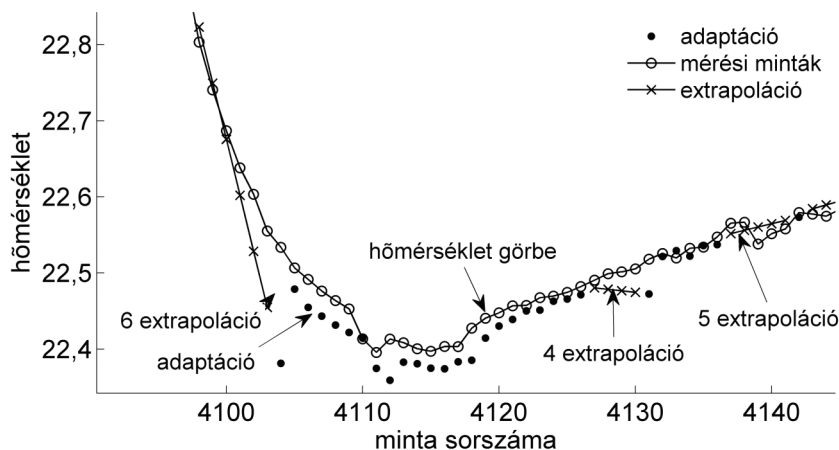
A 3. és 18. sor közt találhatjuk a végtelen (de bármikor megszakítható) ciklust, mely a környezet folyamatos mintavételezését írja le. Minden periódus a következő műveleteket tartalmazza. Első lépésben teszünk egy próba-előrejelzést, hogy megbecsüljük a modell helyességét (4. sor), majd mintavételezzük a környezetet (5. sor). A mintavételt becslésszámításunk legutóbbi mintákat tartalmazó FIFO-vektorba (6. sor) ahol az első elem mindig 1-es a bias miatt (később ezt részletesebben kifejthetjük). Ezek után kiszámítjuk a négyzetes hibát és becslésszámításunk legutóbbi hibák FIFO-vektorába (7. sor), majd adaptáljuk a szűrő súlyait az LMS (Least Mean Square) eljárással (8. sor). Ha a legutóbbi adaptációs hibák négyzetes átlaga (MSE) kisebb vagy egyenlő, mint a felhasználó által specifikált U_{err} hiba (9. sor) akkor biztos, hogy a szenzort alvó állapotba küldhetjük, de a következő kérdés az, hogy mennyi ideig.

A 10. sorban meghatározzuk a t_p alvási időt, ami egy adott ciklusban a megszakítás nélkül alvással töltött időegységek számát jelenti, és a legutóbbi elkövetett hibák négyzetes átlagától függ (a következő szekcióban részletesen tárgyaljuk). Rekurzívan előrejelzünk t_p mintát (11–14. sor), elküldjük azokat a bázisállomásnak (15. sor) majd t_p ideig alvó állapotba váltunk (16. sor). Miután felébredt a csomópont egy újabb mintavételezési ciklust kezdünk (4–17. sor). A következő fejezetben megtárgyaljuk az alvási idő meghatározását (10. sor), az At_p és Bt_p paraméterek becslését, valamint az adaptív szűrő modelljét a teljesség kedvéért.

Adaptáció

Elsőként az At_p és Bt_p paraméterek becslésével, a dinamikusan változó alvási idő meghatározásával, majd a véges impulzus-válaszú szűrő LMS-adaptációjával foglalkozunk. A 2. ábrán látható hisztogramok valamint minden további szimulációs eredmény valós adatokból származik. A minták egy kollégiumban kihelyezett szenzorhálózatból származnak, amellyel egy héten át percenként mintavételeztünk úgy, hogy a bentlakók folyamatosan a szobában tartózkodtak. A környezet nem kontrollált, nem volt megtervezve, a rendszer függetlenül, karbantartás és szakmai felügyelet nélkül működött. A szenzorok semmiféle különleges eljárásban nem részesültek. A szimulátor bemeneteként hőmérséklet- és fényintenzitás-mintákat használtunk. Minden alvás előtt (az algoritmus 9. és 10. sora közé ékelve) logoltuk, hogy mennyivel kisebb az átlagos négyzetes hiba a felhasználó által specifikált hibánál ($U_{err} - mean(e)$). A H_{min} -paraméter a tartomány felső 90%-át jelzi a zajos szélső értékek miatt.

3. ábra. Az adaptáció és az előrejelzések számának változása.



Ezek után azt javasoltuk, hogy ha a hibakülönbség nulla, akkor legyen az alvási idő a minimális Bt_p , és ha a hiba H_{min} -nel kisebb, nyújtuk meg az alvási időt ΔS -sel. Ezzel definiáltuk a lineáris kapcsolatot a hiba és az alvási idő között. A mi esetünkben a minimális alvási idő $Bt_p = 4$, és a kinyújtás $\Delta S = 2$, a felső hibapont $H_{min} = 2,5 \cdot 10^{-7}$ volt. Az $At_p = \Delta S / H_{min} = 8 \cdot 10^6$ az említett egyenes meredeksége. Ez a két paraméter (At_p és Bt_p) egy lehetséges konfigurációja melyet könnyedén becsülhetünk a mintákból. Ezen konfiguráció mellett a 2. ábra jobb oldalán látható alvásiidő-eloszlást kaptuk. Mint látható, legtöbbször 5 egységnyi hosszúságú alvások voltak és legritkábban 7 egységnyi alvások, ami csak akkor fordult elő, ha a hibakülönbség H_{min} -nél kisebb volt. A jobb oldali hisztogram segítségével a statikus alvási időket használó LE-eljárás alvási idejét 5 egységre állítottuk, az objektív összehasonlíthatóság érdekében. Elmondható tehát, hogy ha jól alakultak a hibák, akkor az LE2-algoritmusban az alvási időt egy egységgel csökkentette, ha rosszul, akkor egy egységgel növelte, míg az LE-rendszer statikusan 5 időegységnyi alvást végzett. A 3. ábrán látható az adaptáció és az előrejelzések számának változása, ahol egy-egy 6-, 4-, illetve 5 időegységnyi alvás fázis látható.

Véges impulzus válaszu szűrő LMS adaptálása

Mivel az adaptív szűrők elmélete jól ismert, csak célirányosan és összefoglaló jelleggel vázoljuk működésüket. Mint ahogy azt említettük, minden alvási fázis előtt adott számú előrejelzést végzünk.

Az előrejelzés nem más, mint a legutóbbi N minta súlyozott összege, ami a következő egyenlettel leírható:

$$y = b_0 + b_1 x_{n-1} + \dots + b_N x_{n-N}$$

ahol y a kimenete a modellnek, x_n az aktuális minta, az öt megelőző minták $x_{n-1}, \dots, x_{n-N}$ továbbá $b_0, \dots, b_N$ a szűrő súlyai. A mi esetünkben az aktuális minta x_n nem szerepelhet a jobb oldalon, mivel azt akarjuk extrapolálni, tehát:

$$y = b_0 + b_1 x_{n-1} + \dots + b_N x_{n-N}$$

ahol az y egy becslése az x_n -nek az öt megelőző minták alapján. Látható, hogy meghagytuk a b_0 súlyt, amit torzításnak (bias) nevezünk. Ez a lépés azért szükséges, mert így általánosabb a modell. A $\underline{b} = [b_0, \dots, b_N]^T$ paraméter-vektor egy hipersíkot feszít ki, ahol kétdimenziós esetben b_0, b_1, b_2 paraméterek mellett a b_0 torzítás segítségével vertikálisan is elmozgatható a sík (enélkül nem lenne lehetséges affin-viselkedés modellezése). Az $\underline{x} = [1, x_{n-1}, \dots, x_{n-N}]^T$ vektor az x_n becslendő minta előtti N mintát tartalmazza, ahol az $y = \underline{b}^T \underline{x}$ skalár az x_n becslése. Az U hibát úgy definiáljuk, hogy $U = l - y$, ahol l a tanuló minta, vagy más szavakkal a helyes válasz, amit a becslő extrapolált. Ez a paraméter rendelkezésre áll, ha a rendszer tanuló állapotban van (x_n). Mielőtt mintavételeznénk a környezetet, mindig teszünk egy előrejelzést (1. ábra, 4. sor) a legutóbb mért minták alapján, majd mintavételezünk (1. ábra, 5. sor) és meghatározzuk a hibát (1. ábra, 7. sor). Ezt követően az elkövetett U hiba alapján adaptáljuk a modell paramétereit. Mivel E hibát reprezentál, nem lenne szabad, hogy negatív értéket vegyen fel, ezért legyen $E^2 = (l - y)^2$. Nem használjuk az abszolútérték-funkciót, mivel problémás differenciálni. Eddig definiáltuk a hibát, most nincs más dolgunk csak megtalálni a minimális hibát. Az E^2 hiba kibontott állapotban a következő:

$$E^2 = (l - \underline{b}^T \underline{x})^2 = l^2 - 2l \underline{b}^T \underline{x} + \underline{b}^T \underline{x} \underline{x}^T \underline{b}$$

Jól látható, hogy a hiba a négyzetes függvénye a $\underline{b}$ paraméter-vektornak. Mivel nem használtuk az abszolútérték-függvényt a hibadefiníciónál, nem lesz nehéz dolgunk differenciálni. Egy második fontos megjegyzés, hogy létezik egy abszolút minimum pont, mivel a hibafelület egy sima hiperkvadrátikus felület. Mivel nem azt akarjuk, hogy a modell egy adott mintát becsljön minimális hibával, hanem azt szeretnénk, hogy átlagosan minden becslés a lehető legjobb legyen, a globális $\xi = E(E^2)$ várható hibát kell minimalizálni és nem az E^2 -et, tehát:

$$\xi = E(E^2) = E(l^2) - 2l^T E(\underline{x}l) + \underline{b}^T E(\underline{x} \underline{x}^T) \underline{b}$$

ahol $\underline{P} = E(\underline{x}l)$ a keresztkorrelációs vektor, valamint $\underline{R} = E(\underline{x} \underline{x}^T)$ az autokovariancia-mátrixa a mintáknak.

Ebből következik az új hibadefiníció:

$$\xi = E(l^2) - 2\mathbf{b}^T \mathbf{P} + \mathbf{b}^T \mathbf{P} + \mathbf{b}^T \mathbf{R} \mathbf{b}$$

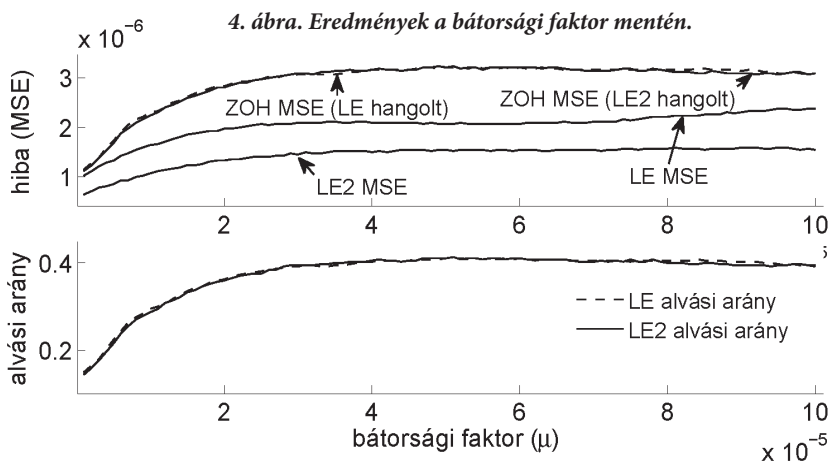
Mivel tudjuk, hogy a hibafelület hiperkvadrátikus, ezért tudjuk, hogy a minimum pont ott található, ahol a gradiens zérus, tehát:

$$\frac{-\partial \xi}{\partial \mathbf{b}} = -2 \mathbf{P} + 2 \mathbf{R} \mathbf{b} = \mathbf{0}$$

Kiemelve az optimumot adó $\mathbf{b}^*$ paraméter-vektort, mely garantálja a rendszer optimalitását:

$$\mathbf{b}^* = \mathbf{R}^{-1} \mathbf{P}$$

Ez az egyenlet a jól ismert Wiener–Hopf-egyenlet. Más szóval, megtalálható az optimális paraméter-vektor, de ismernünk kell az $\mathbf{R}$ autokovariancia-mátrixot és a $\mathbf{P}$ keresztkorrelációs-vektort. Sajnos az esetünkben ezek ismeretlenek, hisz nem tudhatjuk a mintákat előre. A második probléma az, hogy nem végezhetünk mátrix-inverziót a szenzorcsomópontokon ilyen óriási méretben, ahol az összes múltbeli és jövőbeli minta szorzata megtalálható egy mátrixban, még ha lenne is megfelelő méretű memóriánk. Ebből következik, hogy eliminálni kell mind a mátrix inverziót, mind a jövőbeli (ismeretlen) mintáktól való függést. Hogy ezt elérhessük, az aktuális iteráción belüli E^2 hibával fogunk dolgozni és nem az optimalitást biztosító ξ -vel. Ahhoz, hogy megbecsüljük a $\mathbf{b}^*$ vektort, a gradiens-eljárást fogjuk alkalmazni.



A gradiens-eljárás a mindenkor hiba felületen az adott pontban a negatív gradiens irányába tesz egy nagyságú lépést, ezért a minimum pont irányába fogunk haladni, tehát a $\underline{b}$ paraméter-vektor adaptálása a következő:

$$\underline{b}^{uj} = \underline{b}^{regi} + \mu \left(\frac{-\partial E^2}{\partial \underline{b}} \right)$$

ahol μ a bátorsági faktor. Más formában $\underline{b}^{uj} = \underline{b}^{regi} + \Delta \underline{b}$, ahol:

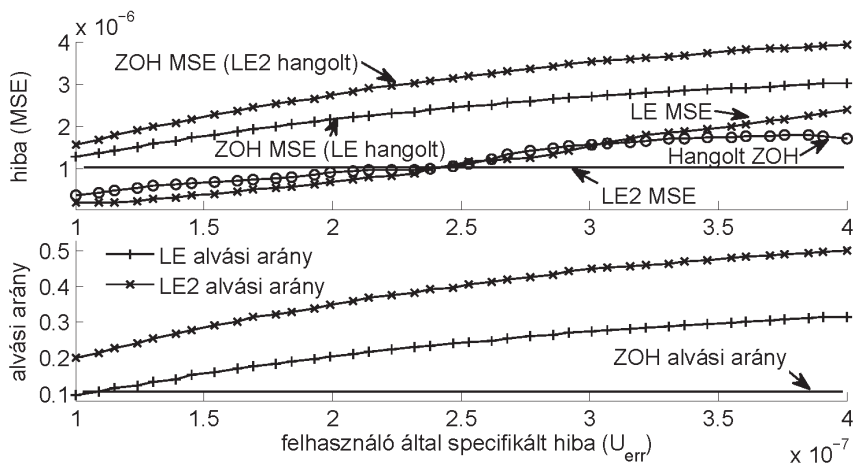
$$\Delta \underline{b} = -\mu \frac{-\partial E^2}{\partial \underline{b}} = -2\mu E \frac{-\partial E}{\partial \underline{b}} = 2\mu E x$$

Az adaptációs-egyenlet tehát bármely iterációban $\underline{b}^{uj} = \underline{b}^{regi} + 2\mu E x$. Ez egy iteratív LMS-megoldás, a szűrők paramétereinek adaptálására. Meg kell jegyezzük, hogy ez az LMS-algoritmus a mi esetünkben konvergál a $\underline{b}^*$ optimális paraméter-vektorhoz, vagyis $E(\lim_{k \rightarrow \infty} \underline{b}(k)) = \underline{b}^*$, ahol $\underline{b}(k)$ a k . adaptált vektor. Ha a μ bátorsági faktor pozitív, valamint nem nagyobb az $\underline{R}$ autokovariancia-mátrix legnagyobb sajátértékének reciprokától, akkor a módszer konvergens. Nem tudunk azonban semmit az $\underline{R}$ autokovariancia-mátrixról, mivel jövőbeli mintákat tartalmaz, ezért a μ bátorsági faktort empirikusan kell beállítani. Egy adaptív eljárásban lehet, hogy először kisebb bátorsági faktort kezdünk, majd megpróbáljuk azt növelni, vagy egyszerűen beállítunk egy kisebb értéket statikusan. A következő szekcióban kiértékeljük az eljárást, valamint összehasonlítjuk az LE-metódussal, illetve a nulladrendű tartóval.

Kiértékelés

Ebben a szekcióban megvizsgáljuk a javasolt architektúra teljesítményét. Két paraméter-érzékenységi elemzést (felhasználó által specifikált hiba valamint bátorsági faktor) és két összehasonlítást mutatunk be. A szimulátor bemeneteként felhasználtuk minden szenzor minden mérését ($5 \cdot 2 \cdot 600000 = 6 \cdot 10^6$ minta). A módszer paramétereit (kivéve azokat, melyeket változtattunk az érzékenységi vizsgálat során) a következők voltak: $SysOrder = 5$, $\mu = 2 \cdot 10^{-5}$, $U_{err} = 3 \cdot 10^{-7}$, $Bt_p = 4$, $At_p = 8 \cdot 10^6$. A 4. ábrán látható a bátorsági faktor mentén végzett érzékenységi vizsgálat valamint a lokális extrapolációval (LE) illetve a nulladrendű tartóval (ZOH, Zero Order Hold) való összehasonlítás. Az összehasonlíthatóság kedvéért az LE és LE2 alvási arányokat egyeztetettük, valamint a nulladrendű tartó mintavételezési számát is úgy állítottuk be, hogy a megspórolt energia (kihagyott minták száma) közelítőleg egyezzen az LE és LE2 által spórolt energiával. Ezek után objektíven megvizsgálható a különböző eljárások által elkövetett hiba, ahol a spórolt energia mindhárom (LE, LE2, ZOH) eljárásnál megegyezik.

5.ábra. Eredmények a felhasználó által specifikált hiba mentén.



Látható, hogy ugyanannyi energiabefektetés mellett mindkét eljárás jobban teljesít, mint egy megspórolt energiában ekvivalens ZOH alul-mintavételezés. Az is jól megfigyelhető, hogy a teljes paraméter-tartományban az LE2 eljárás szignifikánsan kisebb hibával mintavételez, mint az LE statikus eljárás. Az 5. ábrán látható a felhasználó által specifikált hiba mentén végzett érzékenységi vizsgálat és összehasonlítás.

Ellentétben a 4. ábrán látható vizsgálattal, itt az elkövetett hibát próbáltuk egyeztetni az LE- és LE2-eljárások között, hogy objektíven tudjuk összehasonlítani az eljárások által megspórolt energiát. A nulladrendű tartót egyszer úgy állítottuk be, hogy a spórolt energia megegyezzen az LE- illetve az LE2-eljárásával. Másodszor pedig ráhangoltuk az LE és LE2 által elkövetett átlagos hibára (közelítőleg $1 \cdot 10^{-6}$) és megvizsgáltuk, hogy ezt milyen alvási aránynál tudja elérni.

Szeretnénk megjegyezni, hogy a ZOH által elkövetett hiba azon a tartományon a legkisebb, ahol az alvási arány kevesebb mint 50% (a kihagyott mintákat az őket közvetlenül megelőző mintavétellel közelítettük). Ez azért van így, mert a mért értékek többnyire folytonosak, és a közvetlen szomszédos minták közötti eltérés hasonló (a változás sebessége hasonló). Addig, amíg a minták kevesebb, mint 50%-át dobjuk el (egyenletesen), addig bármely ZOH által kihagyott mintát a közvetlenül öt megelőző minta extrapolálja, ami hasonló átlagos hibához vezet. Amint azonban az energiaspórolást 50% felé kényszerítjük, a hiba drasztikusan elkezd nőni a ZOH esetében, hisz minél több mintát hagyunk el 50% felett, annál több mintát extrapolálunk távolabbi szomszédokkal (kénytelenek vagyunk egymást követően több mintát is elhagyni).

Az érvelhetőség kedvéért a ZOH-eljárást a legjobban teljesítő tartományában hasonlítottuk össze mind az LE- mind az LE2-eljárással.

Az 5. ábra alsó részén jól látható, hogy hasonló elkövetett hiba mellett (felső ábra LE MSE és LE2 MSE) az LE2-eljárás ~60%-kal több energiát spórolt, mint a korábban publikált [5] LE-eljárás. Az is látható, hogy a ZOH által elkövetett hiba szignifikánsan nagyobb az LE illetve LE2 által elkövetett hibánál (felső ábra, ZOH MSE), ahol a ZOH alvási arányát ráhangoltuk az LE illetve LE2 alvási arányára. Azért, hogy a megspórolt energia objektíven érzékelhető legyen, további két konstans egyenes is látható az ábrán (lent: ZOH alvási arány, fent: Hangolt ZOH). Itt a ZOH által elkövetett hibát hangoltuk az LE és LE2 által elkövetett átlagos hibára és megfigyeltük, milyen alvási aránynál tudná ezt a célt teljesíteni a ZOH (alsó ábra). Látható, hogy kb. 12%-os alvási aránynál teljesül ez, ami szignifikánsan kisebb, mint az LE vagy az LE2 által teljesített alvási arány.

Konklúzió

Az itt bemutatott munka a korábbi lokális extrapolációs eljárásunk (LE) [8] továbbfejlesztése dinamikus előrejelzésekkel. A szimulációs eredmények megmutatták, hogy az LE2-eljárás jobban teljesít mind a ZOH-, mind az LE-eljárásnál, és így mintegy 60%-kal képes több energiát megspórolni.

[8] Öllös, G.– Vida, R. (2009): *Adaptive Local Extrapolation in Event Driven Wireless Sensor Networks*. In Proceedings of BioSense09, Novi Sad, Serbia.

Evaluate security risk for cyber attacks on information systems

*College of Dunaújváros
E-mail: hadarics@mail.duf.hu

**Veszprog Ltd., College of
Dunaújváros
E-mail: fleitold@veszprog.hu

Abstract: Nowadays, the security became more and more complex and important for an information system. Computer networks are enticing targets of cyber attacks. There are threats using manual or automatic (purpose-designed) tools as well. Attacks on computer networks generally use communication between computers and computer users as well. For example, worms are spreading different ways like e-mail, malware using botnet networks and attacks based on personal communication (social engineering). In this paper, a new mathematical model for measuring or estimating the risk related to threats is described. The primary objective of this research is to propose a risk-assessment model to evaluate the security risk of information systems. This model can help to measure the security risk of individual entities of an information system, and using this information the overall security risk of the whole infrastructure can be calculated. This model can help to find the most vulnerable points of IT infrastructure as well. This paper is based on the ongoing research project of Veszprog Ltd. and the College of Dunaújváros.

Keywords: Security, risk, threat, attack, model.

Összefoglalás: Az informatikai rendszerek biztonsága napjainkban egyre összetettebbé és fontosabbá vált, a számítógép-hálózatok kiemelt célpontjai lettek a kibertámadásoknak. A számítógépek fenyegetései manuálisan vagy akár automatikusan (célprogramok használatával) terjedhetnek. A hálózati támadások általában a számítógépek és a felhasználók közti kommunikációt használják. A férgek (worms) terjedhetnek különböző módon, például e-mailben, a rosszindulatú programkódok botnet hálózatokat hozhatnak létre, és a támadás alapulhat a személyes kommunikáción (social engineering) is. Ebben a cikkben egy a fenyegetések kockázatának mérésére és becslésére

használható matematikai modellt fogunk bemutatni. A kutatás elsődleges célja egy kockázatelemzési modell megalkotása, melynek segítségével az informatikai rendszerekben rejlő biztonsági kockázatok számszerűsíthetők. Ez a modell segíthet abban, hogy meghatározzuk egy informatikai rendszer független entitásainak biztonsági kockázatát, és ezt felhasználva a teljes infrastruktúra biztonsági kockázatát is becsülhessük. A modell használható az IT infrastruktúra gyenge pontjainak meghatározásában is. A cikk egy jelenleg folyó kutatási projekten alapul, amely a Veszprog Kft. és a Dunaújvárosi Főiskola közreműködésével valósul meg.

Kulcsszavak: Biztonság, kockázat, fenyegetés, támadás, modell.

Introduction

The security of computer networks is becoming more and more a serious problem. In the field of network security, automatically spreading malware also mean a significant source of danger besides attacks carried out manually or with the help of purpose-designed programs. Attackers frequently exploit the effect of malware, and occasionally launch malware in order to use the remote controlled (botnet) network of infected computers for future attacks. Malware basically base their spreading on two factors: they can exploit the credulity or lack of expertise of users and persuade them to run the malicious code hidden in the object believed to be safe. On the other hand, malware can exploit the security gap of operation systems and applications of computers and can even gain control automatically, without the knowledge and permission of the user.

A further problem originates from the contact between persons. By using the methods of social engineering, an attacker can persuade the user of a computer not available to the attacker to perform some operations on their computer such as visiting a webpage, where a code had been placed earlier that enables the attacker to gain control over the computer.

Hereafter we would like to examine how dangerous can be a threat in a specific network environment. As long as we know the prevalence of threats and the working method of spreading, and also details of operation, we can tell the risk of a threat in a specific environment. Since all of the threats and their prevalence are changing dynamically in time so that the risk is also constantly changing. All components of an IT system similarly continuously changing, therefore they can all be taken into account when assessing the risk. [1]

[1] Leitold, F.–Hadarics, K. (2012): *Measuring security risk in the cloud-enabled enterprise*. 19th International Conference on Malicious and Unwanted Software, Fajardo, Puerto Rico.

[2] Somayaji, A.– Li, Y.– Inoue, H.–Fernandez, J.– Ford, R. (2009): *Evaluating security products with clinical trials*. In Proceedings of the 2nd conference on Cyber security experimentation and test, pages 3–3. USENIX Association.

[3] Microsoft Security Advisory (912840) <http://www.microsoft.com/technet/security/advisory/912840.mspx>, 2006.

[4] Microsoft Security Bulletin MS02-072 <http://www.microsoft.com/technet/security/Bulletin/MS02-072.mspx>, 2002.

Elements of a security model

With the help of the security model discussed in this article, we would like to model computers, with the help of the processes running on them, the users (be it non-professional users or knowledgeable attackers) and the relationships between them. Applications, processes run on computers. Every process that is able to establish communication with other processes either online or offline, is defined as entities.

In the case of offline communication, an entity loads and interprets the data file placed in the backing store of the computer containing it. In this case, the other process, with which communication takes place, is the process that had created the certain data file. This process can even be on another computer and/or it can pass the data file to the other computer through data media or an online communication channel. In this case, the rules of the communication channel are provided by the format description of the data file.

Apart from the processes running on the computer, the persons using the computer are also included in the circle of entities as they are also able to communicate with other entities. Communication with the processes can typically occur through user's input or the messages of applications, processes, but it is also possible that they establish a connection with another user and communicate with that user (e.g.: personally or by phone). [2]

The threats and their properties are the further parameters of security model. The behavior of a threat can be various. They use completely different methods for spreading. The threats are able to exploit an operating system and a software bug various ways. The prevalence of threats mainly influences the chance of admission into a target network. Of course this is only one aspect of the problem. The success of admission depends on state of software used for defense, the spreading is influenced by access rules and controls used in a computer network. In a security model we should respect the prevalence statistics of threats. Statistics contain instantaneous values and it can change very quickly. Using the internet a new threat can spread very fast, such way prevalence can increase every second. This property of threats extremely changes the risk of a computer system. Therefore we consider only momentary values in a risk model. This value may change extremely caused by a typically new threat. [3, 4]

SIMPLIFIED MODEL

Thereafter a simplified example will be showed. This example illustrates the working of the simplified model. Let us consider the network of a small company. This network contains three computers, where a person does his/her every day job. All computers have internet connection, through a switch and a router.

In the model we consider the following five threats (A to C) with appropriate percentage of prevalence showed in the *Figure 1*.

Then we can define attack vectors that can describe the spreading methods of a threat. Each threat may use a method (attack vector) or not for its spreading, and we can measure the percentage of this usage. In this example there are four methods of spreading called attack vectors 1 to 4.

Besides that threats can use devices in a computer environment and vulnerabilities of operating systems and programs, they can do several tricks to persuade users to do specific operations. For example click on a specific link and visit a web page. This can help attacker to gain control over a computer. In the security model we should use such social engineering tricks. In this example we consider three social engineering tricks.

In the security model the resistances of a user to social engineering tricks are also important. It gives the percentage of abuse can do using a social engineering method.

The three users in this example have the following features:

- Kim: a managing director, who can be deceived easily, he will do almost everything.
- Susan: a secretary, who is cautious, but she can be deceived with specific tricks.
- Peter: a system administrator, who knows many aspect of security, and recognizes many tricks, and he can be deceived with difficulty.

Using values in *Figure 1*, we can set the probability of admittance. These values depend on the social engineering tricks. In the situation, that a thread does not depend on a social engineering trick, we put 100% to the appropriate cell. In this way, the user does not affect the spreading of the threat.

When we do risk assessment, we will use the maximum value of a column. This means, it is the maximum, which users can affect to system security (affection rate).

Then we can use prevalence values in the model. The multiplication of prevalence and affection rate gives us the risk of one threat. As we sum all the risk of threats, it will give us the risk estimation (vulnerability metric) of the whole system.

Figure 1. summarizes the entire table used by simplified model.

Figure 1. A simplified model example.

Threats	A	B	C	D	E
Prevalence	35%	24%	18%	15%	8%
social engineering trick 1	100%	0%	0%	0%	0%
social engineering trick 2	0%	0%	50%	0%	0%
social engineering trick 3	0%	0%	0%	0%	100%
attack vector 1	0%	5%	7%	0%	60%
attack vector 2	0%	4%	0%	25%	0%
attack vector 3	0%	0%	11%	0%	0%
attack vector 4	0%	0%	0%	16%	30%

Kim	Susan	Peter
50%	12%	0%
43%	8%	0%
35%	7%	1%

Kim	50%	100%	72%	100%	35%		VULNERABILITY METRIC
Susan	12%	100%	54%	100%	7%		
Peter	0%	100%	50%	100%	1%		
	0%	2%	2%	6%	3%		13%

As a specific parameters change occurs, it mainly affects the whole system security.

Figure 2. Parameters changed in simplified model example.

Threats	A	B	C	D	E
Prevalence	35%	24%	18%	15%	8%
social engineering trick 1	100%	0%	0%	0%	0%
social engineering trick 2	0%	0%	50%	0%	0%
social engineering trick 3	0%	0%	0%	0%	100%
attack vector 1	30%	5%	7%	0%	60%
attack vector 2	0%	4%	0%	25%	0%
attack vector 3	0%	80%	11%	0%	0%
attack vector 4	50%	0%	0%	16%	30%

Kim	Susan	Peter
50%	12%	0%
83%	8%	0%
35%	7%	1%

Kim	50%	100%	92%	100%	35%		VULNERABILITY METRIC 47%
Susan	12%	100%	54%	100%	7%		
Peter	0%	100%	50%	100%	1%		
	14%	21%	3%	6%	3%		

MODEL BASED ON MARKOV CHAIN

Thereafter a Markov chain based model will be showed. Previously we defined the notion entity. The entities located within one computer are – as a cluster – defined as belonging together. It is presumed that if an attacker has successfully attacked an entity, the attacker is able to control or influence the other entities belonging to that computer, too. [5]

The elements of the security model can be represented as a graph, where the individual entities – representing the processes running on the computer and the users themselves – are the nodes. The edges between the nodes represent the communication channels between the entities. Communication between two processes means some kind of online or offline data transfer. Naturally, there can also be a connection between persons; since any person can contact any other person (e.g. can call the other by phone).

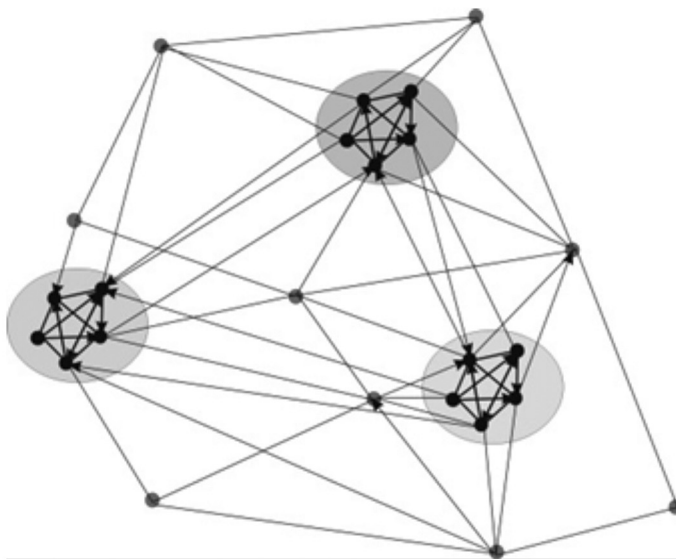
[5] Leitold, F. (2010): *Security Risk analysis using Markov Chain Model*. 19th Annual EICAR Conference, Paris, France– Puerto Rico.

In the case of connections between processes and persons, it is necessary to differentiate between the two directions of communication. While computer users can shape the operation of certain processes with the help of the appropriate input fields, processes can also send messages to users.

By differentiating between the directions of the communication channel we can represent the model with a directed graph. This displays real circumstances in a much more accurate way, since the two directions cannot usually be considered as the same in the case of protocols. This is especially so in the case of server-client-based communication.

The grey points represent the users; the black points mean the processes within the computer. The grey ellipses display the processes belonging together within a computer. In this way, the model also displays which entities are connected to each other. However, weighting can also be assigned to the individual directed graphs, depending on whether the appropriate direction of the given communication channel is suitable for enabling an attacker to attack another entity. If this value is 0, then it is not possible, and the higher the value, the more easily the channel can be exploited. If the value is 0, the directed edge does not figure in the directed graph.

Figure 3. A simple graph model.



With the help of the security model discussed in this article, we would like to model computers, with the help of the processes running on them, the users (be it non-professional users or knowledgeable attackers) and the relationships between them. Applications, processes run on computers. Every process that is able to establish communication with other processes either online or offline, is defined as entities.

Based on the graph model, the matrix representation of the model can also be prepared. Here an entity corresponds to every row and column. Being the values assigned to the directed edges in the graph, the numbers in the graph mean the value assigned to the value of the appropriate direction of the communication channel between the two nodes.

Based on all these, if the value in the matrix is chosen in a way that the sum of the values in each row is exactly 1, we arrive at the Markov chain known in the random walk examples of the theory of probability. Let us consider the state vector of entities that contains the value of each entity describing to what extent each entity is vulnerable. It can be assumed that the attacker as a person is also present among the entities and, in the beginning, only the attacker is considered as dangerous. Consequently, in the initial state vector the value of the attacker is 1, the other values are 0.

Then, with the help of the initial state vector and the matrix meaning state transition (by multiplication) we can find out what other entities can be controlled by the attacker and also how difficult or easy it is to do so.

The a_{ij} value in the state transition matrix is then 0, if there is no possibility for the attacker disposing of control over entity i . to gain control over entity j ., otherwise $a_{ij} > 0$. Value a_{ij} is characteristic of the communication channel, the protocol and of entities i . and j . Its value can be influenced by several factors:

- The reliability and vulnerability of the communication channel and the protocol describing the rules of communication.
- The reliability and security gaps and the corrections referring to it of entity j . representing the process. It is important, for example, whether we use the many-year-old version 5 of Microsoft Outlook Express or the much more secure Bat mail.
- If entity j . is a person, this person's gullibility also influences the a_{ij} value.
- Another influencing factor can be time itself since vulnerability also increases if a security gap becomes known.

Knowing the transition matrix, the following transition vectors can be determined on the basis of the initial state vector. Mathematically, the entities that can be attacked by an attacker can be defined as the limit values of the series of state vectors. In reality, however, a final value can be determined depending on the number of entities. This final value reveals how many transitions are necessary to run out of devices over which the attacker can take control.

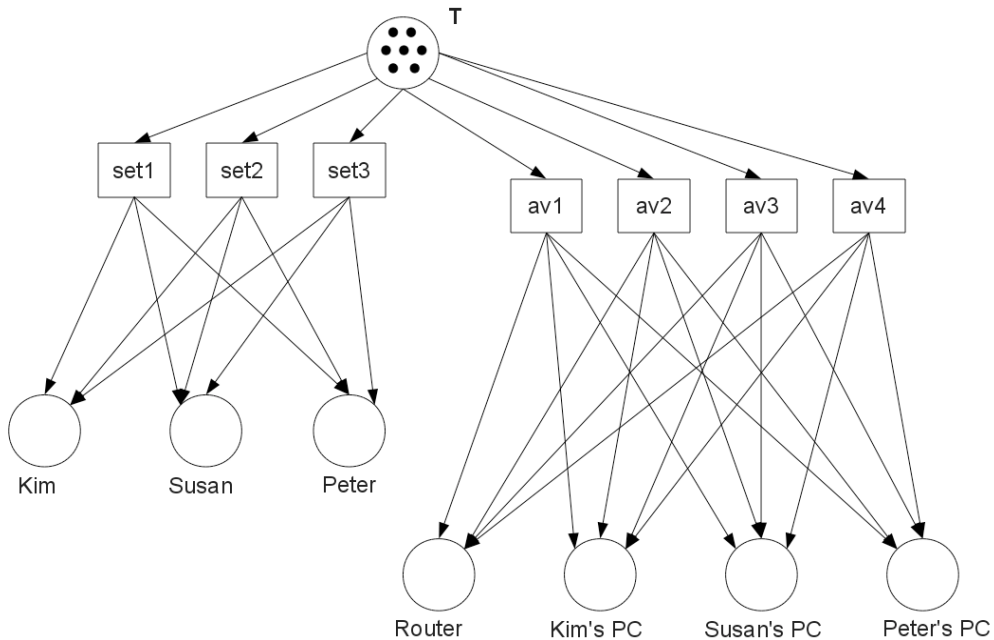
The mathematical model described here applies the theory of Markov chains to define a simulation instrument that is suitable to make the risk of malware measurable in the case of enterprises.

The model does not only take IT devices into consideration, but it is also able to handle users as entities that are just as attackable (influenceable). The risk caused by malicious codes keeps changing. The major reason for this is that the range of malicious codes as well as their degree of prevalence is also changing.

MODEL BASED ON PETRI NETS

Let us consider we want to examine one threat. In the model we use a given number of social engineering tricks and attack vectors. The next Petri-graph illustrates the initial state of the model:

Figure 4. A Petri-graph example.



Notations on *Figure 4*:

- Circles represent places, rectangles represent transitions.
- Entities presented in the model mean places, social engineering tricks, and attack types mean transitions.
- Circle T represents the attacker.
- In initial state T containing the same number of token as the circles in the model.
- Other circles represent persons, computers and network devices.
- Set1, set2, set3 mean social engineering tricks; av1, av2, av3, av4 mean attack types.

In this example we consider the followings:

Figure 5. Used parameters in Petri graph model.

Threats	C	Kim	Susan	Peter
Prevalence	18%	50%	12%	0%
social engineering trick 1	0%	43%	8%	0%
social engineering trick 2	50%	35%	7%	1%
social engineering trick 3	0%			
attack vector 1	7%			
attack vector 2	0%			
attack vector 3	11%			
attack vector 4	0%			

If the above conditions met, the model can be simplified. Where attack vector is zero, the arcs are unnecessary, so we can delete them.

This Petri-graph has a problem as it does not use influence between entities. We should supplement the model with the communication between entities.

The next four influences are possible:

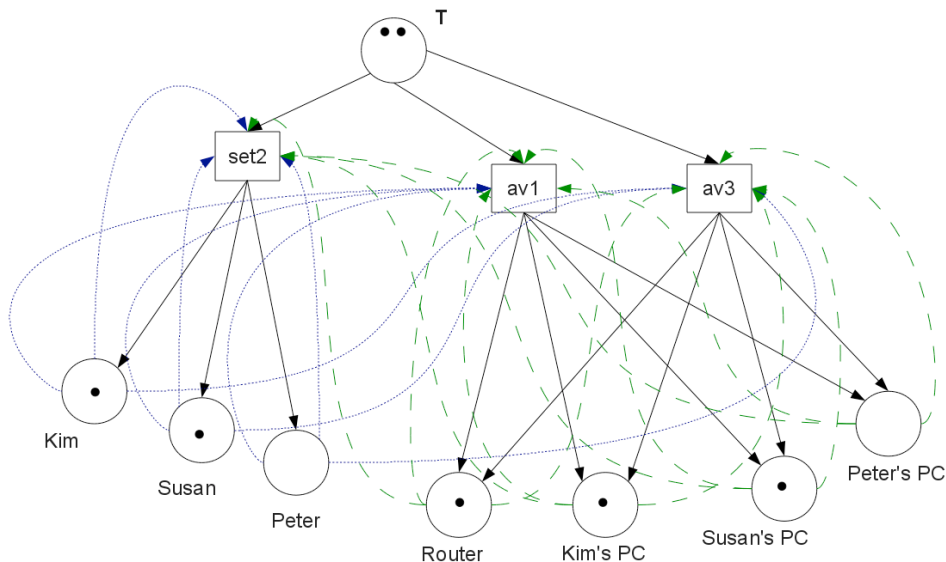
- Device – device (D-D) connection
- Device – person (D-P) connection
- Person – device (P-D) connection
- Person – person (P-P) connection

If we consider all of influence, get a sophisticated Petri-graph. The dotted arrows mean influences caused by persons; the dashed arrows mean influences caused by devices. The next step is to start tokens from attacker. A condition located in a transition depends on:

- Prevalence of threats.
- Network topology, connections, and firewalls and control list configurations.
- The version of used operation systems and software.
- The behaviors of persons working with computers.
- The used protocols and operating systems of network devices
- Direct or indirect connections between persons and devices.

The most unpredictable things are behaviors of persons. Sometimes they apply security regulation, other time they do not. Pseudo random numbers help to illustrate this behavior. When a threat successfully admitted to a place, we sign it with token.

Figure 6. An extended Petri-graph example.



If we execute the simulation many times, after a number of step we can see the states of entities. Based on the number of a possible state, we can define vulnerability metric.

Conclusion

The vulnerability of an information system is influenced by many factors. When we create a security risk model, all things should use or forget. As factors are connected to each other, they can affect differently with each other. In many situations we can have practical feeling about security. We have an idea, but not sure its reality. Mathematical models help to check the validity of the idea. The models do not only take IT devices into consideration, but it is also able to handle users as entities that are just as attackable (influenceable). The risk caused by threats keeps changing. The major reason for this is that the range of threats as their degree of prevalence is also changing.

Grafémaanalízis kanonikus összetevők alapján

*BME Elektronikus Eszközök
Tanszéke
E-mail: tothlori@gmail.com

** BME Elektronikus Eszközök
Tanszéke
E-mail: pardede@eet.bme.hu

***BME Elektronikus Esz-
közök Tanszéke
E-mail: hosszu@eet.bme.hu

Összefoglalás: A cikk az ismeretlen eredetű székely–magyar rovásírással készített feliratok korának és stílusának meghatározására kidolgozott új eljárást mutatja be. A szerzők különböző székely–magyar rovásábécékben található grafémákat kétdimenziós részalakzatokra, úgynevezett kanonikus alakokra bontották. Ezek a kanonikus alakzatok lehetővé teszik ezeknek a grafémáknak a kanonikus alakokból történő felépítését, mivel a vizsgált grafémák mind szétbonthatók e kanonikus alakzatokból álló elemi részekre. A kidolgozott kanonikus alakra bontás tulajdonképpen a Fourier-analízisre emlékeztető módszer, amely az egyes grafémák közös részeit kezelhetővé teszi. Kanonikus alakzat lehet például egy körszerű hurok, ferde szakasz, függőleges szakasz, kereszteződés. A kanonikus alakzatokra épülő analízis során grafémánként csupán a kanonikus alakok létezését jelöljük, és nem vesszük figyelembe egy adott feliratban az egyes grafémák előfordulásának számát.

A vizsgált különböző korú és stílusú rovásábécékben szereplő grafémák kanonikus felbontásában szereplő együtthatókat az egyes ábécékre összegezzük és egy vektort kapunk. Ezt a vektort az ujjlenyomathoz hasonlóan az ábécék kanonikus lenyomatának nevezhetjük, mivel egy adott korban és stílusban készített ábécében lévő grafémák együttesére jellemző.

A szerzők egy ismeretlen eredetű, de jó állapotban levő feliratot alkotó grafémákat felbontották kanonikus alakjaira, és elkészítették a rájuk jellemző lenyomatokat. A korábban nem meghatározott korú és stílusú feliratok lenyomatát korrelációanalízis segítségével összehasonlították a már ismert ábécék kanonikus lenyomataival. A korrelációs együtthatók elemzésével sikerült a vizsgált felirat korára egy becslést adni.

Kulcsszavak: Grafémaanalízis, feliratok azonosítása, számítógépes paleográfia.

Abstract: The article presents a new method for determining the ages and the styles of the Szekely-Hungarian Rovas inscriptions with unknown origin. The authors decomposed graphemes of various Szekely-Hungarian Rovas alphabets into two-dimensional primitive shapes, so-called canonical forms. The various graphemes can be built from these canonical forms. The developed canonical decomposition is analogue with the Fourier analysis. Examples of the canonical forms are the circular loop, slating section, vertical section, and intersection. The parameters of the canonical decompositions of graphemes of each alphabet are summarized into one vector. This vector can be called the canonical print of the alphabets similarly to the fingerprint. The authors produced the canonical prints of the well-known alphabets and of the previously unidentified inscriptions. Correlation analysis was applied to compare the canonical prints of the unidentified inscriptions to the prints of the well determined alphabets. By analyzing the correlation coefficients an estimate of the ages of the unidentified inscriptions became possible.

Keywords: Grapheme analysis, identification of inscriptions, computational paleography.

Bevezetés, irodalmi áttekintés

A korábbi időkben készített írások feltárása után számos nehézséget okoz a kutatók számára azok azonosítása, stílus és kor szerinti besorolása. Ennek oka az írást hordozó anyag (fa, kő, tégl, papír, stb.) romlásától eltekintve elsősorban az, hogy az írásokban használatos betűk alakjai idővel változtak [1].

Számos kutatás látott napvilágot, amelyekben az írásfelismerést, jelalak-felismerést, nehezen olvasható szövegek jelentésazonosítására dolgoztak ki számos algoritmust a kutatók [2]. A nemzetközi szakirodalomban található cikkek, folyóiratok, konferenciák sokszínűsége alátámasztja e téma hasznosságát, aktualitását. Többek között értékes eredményeket értek el az Indiában a kannada nyelvet beszélők között az 5. század óta használatos saját kannada írás emlékei korának automatikus meghatározásában [3]. A kannada írás fejlődését az 1. ábra mutatja.

[1] Dian, Sz.:
*Jelentésazonosító
szoftver fejlesztése.*
Tudományos
Diákköri
Konferencia,
Budapest, 2009.

[2] Tik, D. (2007):
Szövegbányászat.
Budapest: TypoTeX

[3] Harish-Kashyap-
Bansil-Arun-
Koushik: *Hybrid
neural network
architecture for age
identification of
ancient Kannada
scripts.* Proceedings
of the 2003 IEEE
International
Symposium on
Circuits and Systems
(ISCAS), Vol. 5
(25–28 May 2003) Pp.
661–664.

[4] Dinesh, A. U.–Subba Reddy N. V.–Makkithaya, K.: *Multilevel classifiers in recognition of handwritten Kannada numerals*. Proceedings of World Academy of Science, Engineering and Technology (WASET); Vol. 32 (Aug. 2009), pp. 278–283.

1. ábra. A kannada írás fejlődése.

८	↓	†	Σ	Σ	८	3rd BC -2nd AD
W	↓	†	Σ	Σ	८	2nd -5th
८	८	८	Σ	Σ	८	5th - 6th
८	८	८	—	Σ	८	6th - 8th
८	८	—	Σ	Σ	८	8th - 9th
—	८	८	Σ	Σ	८	9th - 11th
८	८	—	—	Σ	—	11th - 13th
८	८	८	Σ	Σ	८	13th - 15th
८	८	८	Σ	Σ	८	15th -18th
८	८	८	Σ	Σ	८	18th onwards

Forrás: Harish–Kashyap–Bansilal–Arun–Koushik: *Hybrid neural network architecture for age identification of ancient Kannada scripts*. 2003 IEEE International Symposium on Circuits and Systems (ISCAS), Volume 5. 25–28. May 2003. Pp. V-661–V-664.

A cikkben közölt eljárás neurális hálózatok alkalmazásával kísérli meg a különböző történelmi korokban használatos betűalakokra jellemző azonosítók felismerését egy tetszőleges szövegben [4].

Egy másik kutatócsoport a régi perzsa írások felismerési eljárásának kidolgozásában hasonlóan ígéretes eredményeket ért el. A perzsa kézírás felismerésének folyamata a következő terminológiát követi: előfeldolgozás; részegységekre bontás; jelfelismerés; ellenőrzés. Az előfeldolgozás magában foglalja a kép optikai feldolgozását, képhibák minimalizálását. A részegységekre bontás feladata a kézírásra jellemző érintkező betűalakok felismerése és ezek elválasztása. Ezt követően a felismert betűalakokat osztályba sorolják, és ennek alapján azonosítják.

Az algoritmusban felhasználják a betűk alakjának szerkezeti felépítését is. Ezt az a váz adja meg, amellyel a betűket korábban körüljárták a betűalakok elválasztásánál alkalmazott módszer alapján. Ezt illusztrálja a 2. ábra.

2. ábra. betűk szerkezetének tulajdonságai.



Forrás: Doermann, D.–Jaeger, S.: *Arabic and Chinese Handwriting Recognition*. SACH 2006 Summit College Park, MD, USA. September 27—28. 2006. P. 29.

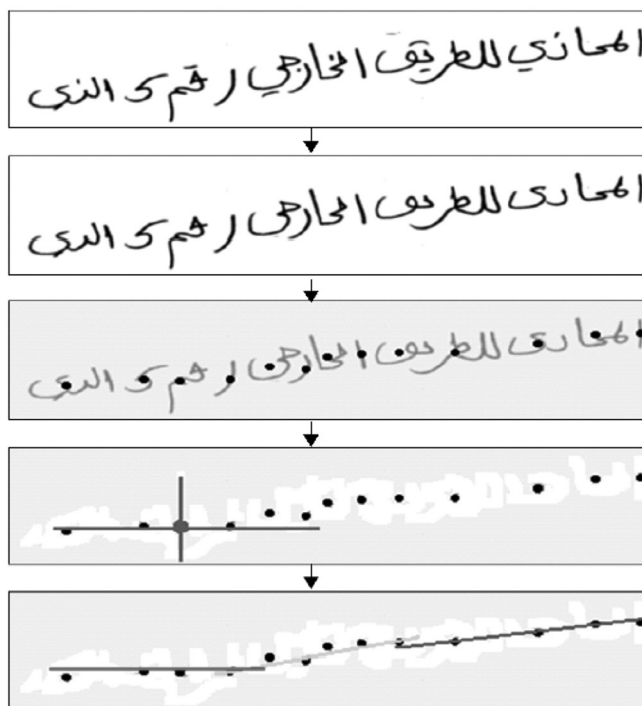
A cikkben közölt betűalak-elválasztás és jelfelismerő módszerek együttes használatával megbízhatóan szét lehet választani a perzsa írásjeleket egymástól [5].

Az optikai karakterfelismerő szoftvereknél gondot okoz a ferde írás felismerése egy dokumentum képéről, amikor a szöveg sorai nem vízszintesen íródnak, egymásba csúsznak. A ferdén írt arab szövegek korrigálásához készítettek egy algoritmust a kutatók, amelynek egy részlete a 3. ábrán látható.

[5] Izadi, S.–Sadri, J.–Solimanpour, F.–Suen, Ch.Y.: *A review on Persian script and recognition techniques*. In: *Arabic and Chinese Handwriting Recognition. Lecture Notes in Computer Science*. Springer Berlin, Heidelberg, March 13, 2008.

[6] Kumar J., Abd-
Almageed W.,
Le Kang and
Doermann D.:
"Handwrit-
ten Arabic
Text Line
Segmentation
using Affinity
Propagation."
Document
Analysis
Systems, pp.
135-142, 2010.

3. ábra: egy ferdén írt arab szöveg azonosításának folyamata.



Forrás: Kumar, J.–Abd-Almageed, W.–Kang, L.–Doermann, D. (2010): Handwritten Arabic Text Line Segmentation using Affinity Propagation. *Document Analysis Systems*. Pp. 135–142.

Meghatározzák a betűalakok orientációját, az egyes betűalakok geometriai közép-pontjának kiszámolásával, és ezen pontok távolsága és iránya ismeretében határozzák meg az egy sorhoz tartozó betűalakokat, melyeket további geometriai lépéseket végrehajtó algoritmusokkal helyükre rendeznek. A pontok távolságának számításához a „legrövidebb út” algoritmust, a betűalakok egy sorba tartozásának vizsgálatához az „affinity propagation” klaszterező módszert alkalmazzák [6].

A székely–magyar rovás (SzMR) a magyar írások közé, másrészt a rovás íráscsaládba tartozik. A rovásírás alfabetikus elveken jelöli a hangokat, minden hanghoz egy betűt rendel.

Az írásirány a legtöbb esetben jobbról balra halad, de lehet látni balról jobbra haladó emlékeket is. Az utóbbi esetben az írásjegyek függőlegesen tükrözendők.

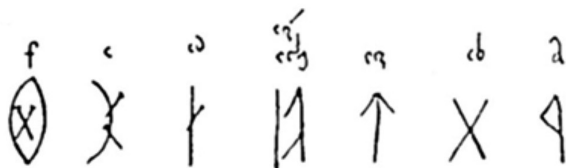
Kis- és nagybetűk közötti különbségtétel nincs, noha néha a tulajdonnevek kezdőbetűjét szokás volt kicsit nagyobbak írni.

Ezt az írást a székelyek őrizték meg. A feliratokat leggyakrabban különböző anyagokba, például fába és kőbe karcolva lehet megtalálni. Eredete az utóbbi évtizedekben tisztázódott. A korábban általánosan elfogadott, de sohasem igazolt álláspont szerint a türk írásból fejlődött ki. Ezzel szemben a legfrissebb kutatások szerint a SzMR részben kisázsiai, részben birodalmi arám eredetű. Nem azonos sem a nagyszentmiklósi kincsen is olvasható kárpát-medencei rovással, sem a homokmégy-halomi emléken található korai sztyeppei rovással [7].

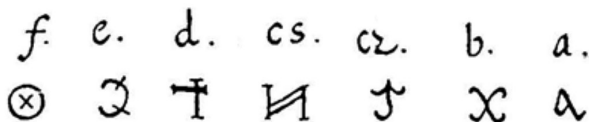
Egymástól stílusban eltérő SzMR-ábécék jöttek létre különböző korokban, és különböző földrajzi területeken, továbbá a különböző írásemlékeket eltérő kézírással és ismeretekkel rendelkező emberek hozták létre. Az ábécé grafémáinak topológiai tulajdonságaira jellemző stílusjegyek ábécéről ábécére változnak.

4. ábra. Székely-magyar rovásábécé részletek.

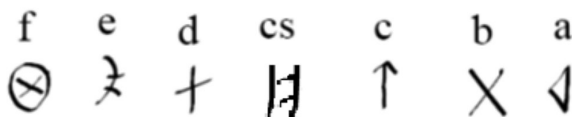
Nikolsburgi ábécé



Bél Mátyás féle ábécé



Patakfalvi ábécé



Forrás: Rovás kincsek CD-tár.

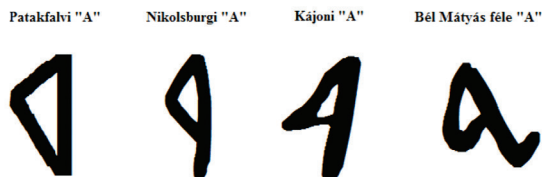
[7] Hosszú, G.: *Heritage of Scribes. The Relation of Rovas Scripts to Eurasian Writing Systems*. 2011: Első kiadás. 2012: Második, bővített kiadás. Budapest: Rovás Alapítvány. Olvasható a Google Books-on: <http://books.google.hu/>

[1] Dian Sz.:
Jelentésazono-
sító szoftver
fejlesztése,
Tudományos
Diákköri
Konferencia,
Budapest,
2009.

[7] Hosszú,
G.: *Heritage
of Scribes.
The Relation
of Rovas
Scripts to
Eurasian
Writing
Systems.*
2011: Első
kiadás. 2012:
Második,
bővített
kiadás.
Budapest:
Rovas
Alapítvány.
Olvasható
a Google
Books-on:
[http://books.
google.hu/](http://books.google.hu/)

A 4. ábrán a Nikolsburgi, Bél Mátyás-féle és Patakfalvi SzMR-ábécék részleteit láthatjuk. Amíg a Nikolsburgi és Patakfalvi ábécék betűalakjai egyenes szakaszokból épülnek fel, addig Bél Mátyás ábécéjét jellemzően íves primitívek alkotják. Vannak grafémák, amelyek a legtöbb ábécében hasonló topológiai tulajdonságokkal bírnak, ugyanakkor létezik ezek ellentéte is, amikor az adott hangot jelölő betűalak egymástól eltérő primitívekből épül fel. Erre mutat példát az 5. ábra, ahol az A betűt emeltük ki négy különböző SzMR ábécéből.

5. ábra. Székely–magyar rovás A betűk.



Jelen cikkben olyan hazai kutatási eredményekről számolunk be, amely ismeretlen eredetű SzMR-írással [5] évszázadokkal korábban készített szövegek stílusának és korának beazonosításához kíván a kutatóknak segítséget nyújtani azzal, hogy egy egzakt minősítő algoritmus segítségével a lehetőségeken belül teljes körűen meghatározza egy adott felirat lehetséges eredetét és azt kategóriákba sorolja matematikai statisztikai eszközök segítségével. A kidolgozott eljárás az egyes grafémák topológiai felbontására [1], a vizsgált felíraton azonosítható stílusjegyekre és azok kiértékelésére kifejlesztett algoritmusra épül, továbbá felhasználja az eddig ismert SzMR-ábécék betűalak tárát kanonikus összetevőikkel.

A kifejlesztett eljárás

GRAFÉMÁK KANONIKUS ALAKJA

Grafémának nevezzük az írás alapegységét, *karakternek* pedig azt a grafémát nevezzük, amelyhez valamilyen kódot rendeltek. A graféma különböző típusú lehet: van, amelyik egy betű, egy másik egy írásjel, és vannak kiegészítő jelek is, ilyen pl. az ékezet. További attribútumai az írás, amin belül használták, grafémanév, grafémaalak (glyph), hangérték,

használati időszakai, használati területei stb. Betűalakból – amely egy graféma alakzati tulajdonságait tartalmazó kép – és hangértékből – amely egy adott nyelven egy adott korban a grafémához tartozó hangot jelöli – lehet több.

Kanonikus alak alatt a grafémák kétdimenziós topológiai részalakzatát, úgynevezett primitívjét értjük. Jelen kutatásunkban a következő primitíveket különböztetjük meg egymástól:

- háromszögszerű alakzat,
- körszerű hurok,
- négyszög,
- függőleges szakasz,
- vízszintes szakasz,
- ferde szakasz,
- ív,
- keresztezés,
- pont.

A primitíveket úgy választottuk meg, hogy azok páronként függetlenek legyenek, vagyis nem fedhetnek át egymással. Létrehoztuk a grafémák primitívjeinek mátrix reprezentációját, amelyet egy adott SzMR-ábécéhez a 6. ábra mutat.

6. ábra. SzMR-ábécé grafémaalakjainak kanonikus alakra bontása.

glyph	abc	háromszög- szerű alak	körszerű hurok	négyszög	függőleges szakasz	vízszintes szakasz	ferde szakasz	ív	x (keresztelés)	pont
q	a	1			1					
X	b						1		1	
↑	c				1		1			
H	cs			1	1					
f	d				1				1	
3	e							1	1	
2	é						1	1		
0	f		1						1	
Λ	g	1					1			

[4] Dinesh, A. U.–Subba Reddy N. V.–Makkithaya, K.: *Multi-level classifiers in recognition of handwritten Kannada numerals*. Proceedings of World Academy of Science, Engineering and Technology (WASET); Vol. 32 (Aug. 2009), pp. 278–283.

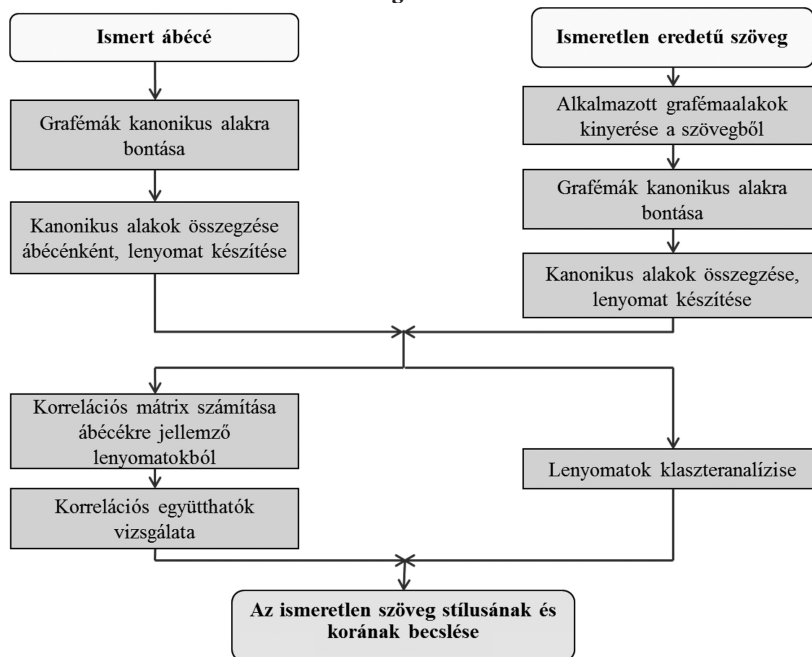
[8] Doermann, D.–Jaeger S. (2006): *Arabic and Chinese Handwriting Recognition*. Berlin: Springer. SACH Summit, pages: 278. Heidelberg.

Az analízis során grafémánként csupán a kanonikus alakok létezését jelöljük, és nem vesszük figyelembe egy adott feliratban az egyes grafémák előfordulásának számát. A 6. ábra oszlopaiba kerülnek a kanonikus alak elnevezései a soraiba pedig az ábécé grafémái.

A KIFEJLESZTETT ALGORITMUS LEÍRÁSA

Meg kell jegyezni, hogy algoritmusunkban nem cél a grafémák pl. bitkép alapján való felismerése. Ezzel a témával (nem az SzMR-írás esetére) több nemzetközi publikáció foglalkozik [4], [8]. Az általunk választott primitívek az eddig analizált rovás ábécé grafémáit teljes egészében fel lehet bontani, amit a jövőben csak abban az esetben kell bővíteni, ha olyan ábécét, illetve ismeretlen eredetű szöveget bontunk kanonikus alakjaira, amely igényli a kanonikus alaktár bővítését újabb primitívekkel. Az algoritmus blokkvázlatát a 7. ábra szemlélteti.

7. ábra. Az algoritmus blokkvázlat



Az algoritmusnak két bemenete van. Az egyik bemenet rendre megkapja az ismert SzMR-ábécék betűalakját, a másik az ismeretlen eredetű szövegből kinyert ábécé betűalakjait.

Az algoritmus lépései:

– Meghatározzuk az általunk definiált kanonikus alakokat grafémánként.

Ez intuitív módon történik, a már korábban definiált kanonikus alakok felhasználásával.

– A kanonikus alakokból úgynevezett lenyomatot készítünk, amely egy adott ábécére jellemző, akár egy újlenyomat. A lenyomat az 6. ábrán szereplő táblázat oszlopainak összegzéséből tevődik össze. Az elkészített lenyomatokat a 8. ábra mutatja.

– Az ismeretlen eredetű feliratból ábécét készítünk.

– Meghatározzuk az általunk definiált kanonikus alakokat grafémánként.

– A kanonikus alakokból lenyomatot készítünk, ami az ismeretlen eredetű feliratra jellemző. Az ily módon elkészített lenyomatokat vizsgálatoknak vetjük alá:

– Az egyik vizsgálati módszer az ábécé lenyomatok korrelációs mátrixának elemzése.

– A másik általunk alkalmazott módszer, az ábécé lenyomatok klaszteranalízise.

8. ábra. Vizsgált székelymagyar (1–10) rovásábécék lenyomatai

ID	ABC	háromszög- szerű alak	körszerű hurok	négyszög	függőleges szakasz	vízszintes szakasz	ferde szakasz	ív	x (keresztelés)	pont
1	Szabványosításra előterjesztett SzMR	7	4	5	20	1	24	6	9	0
2	Patakfalvi SzMR	9	5	3	16	2	22	8	7	0
3	Nikolsburgi SzMR	6	5	3	17	0	22	4	8	0
4	Bél Mátyás féle SzMR	5	7	3	9	10	6	19	10	2
5	Magyar Adorján féle SzMR	7	3	4	16	0	20	5	11	0
6	Rettegi István féle SzMR	3	6	2	7	6	23	8	10	0
7	Kájoni János (rég)i féle SzMR	4	4	2	10	2	7	6	8	1
8	Gyulafehérvári /portsalmi/ SzMR	6	7	3	15	4	14	7	9	1
9	Hickes - Harsányi SzMR	6	8	3	8	3	10	22	12	0
10	"Magyar ABC" / Gönczi György SzMR	5	5	4	11	5	9	13	11	2
11	Steppean Rovás	6	12	4	31	1	39	23	9	0
12	Középkori utánzat	7	23	2	21	18	10	26	19	0

[9] Kovács, P. (2007):
Elméleti összefoglalók – Korreláció- és regresszió-számítás.
Szeged: SZTE Gazdaságtudományi Kar, Statisztika.

Eredmények

A SZÉKELY–MAGYAR ROVÁSÁBÉCÉ-LENYOMATOK KORRELÁCIÓS ELEMZÉSE

Korrelációs számítás alatt azt vizsgáljuk, hogy az elemzésbe vont metrikus változók között mennyire szoros a kapcsolat. Két metrikus változó (x és y) közötti kapcsolat vizsgálatánál pontdiagramot készíthetünk az (x, y) változópár alapján. A pontdiagram alapján megállapíthatjuk a változópár közötti kapcsolat típusát: lineáris, vagy nem lineáris a kapcsolat. Lineáris kapcsolat esetén a pontok egy képzeletbeli egyenes, nem lineáris kapcsolat esetén egy szabályos görbe körül szóródnak. Lineáris kapcsolatnál a pontoknak a képzeletbeli egyenes körüli szóródásából következtethetünk arra, hogy milyen szoros kapcsolat van a két változó között. Az egyenes meredekségéből pedig következtethetünk a kapcsolat irányára, ami pozitív, vagy negatív lehet. A pozitív irányú kapcsolat azt jelenti, hogy a két változó azonos irányba változik. A kapcsolat erősségének jellemzésére lineáris kapcsolat esetén az úgynevezett lineáris korrelációs-együtthatót használjuk. Az r lineáris korrelációs-együttható értéke $[-1; +1]$ tartományba esik. Előjele megadja a két változó közötti kapcsolat irányát, míg abszolút értéke a kapcsolat erősségét. A nullához közeli érték gyenge, az egyhez közeli érték erős kapcsolatot jelent [9].

Az (x, y) változópár esetünkben egy-egy vektor, amelyek közül az egyik az ismeretlen eredetű ábécé lenyomata míg a másik rendre az ismert ábécék lenyomatai. A két vektorra számolhatunk korrelációt, amely a pozitív egyhez közeli, ha az ismeretlen eredetű ábécé lenyomata nagy hasonlóságot mutat egy adott ismert ábécé lenyomatával, ellenkező esetben nulla, vagy negatív előjelű értéket ad.

Elkészítettük tíz darab ismert SzMR-ábécé és két másik rovás-ábécé a lenyomatát, és rendre kiszámoltuk a korrelációs-együttható értékét mindegyik ábécére, ezzel megkaptuk a 9. ábrán található korrelációs mátrixot, melynek főátlójában természetesen csupa egy szerepel, mivel az adott ábécék lenyomatának korrelációs-együtthatóját önmagukkal számoltuk. Sötét háttérrel a 0,9-nél nagyobb korrelációs-együttható értékeket emeltük ki. Megállapítható, hogy például a Nikolsburgi -ábécé lenyomata erős korrelációban áll a Patakfalvi-ábécé lenyomatával, ugyanis stílusukban kicsi az eltérés, amint azt korábban meg is mutattuk.

A szabványosításra előterjesztett SzMR-rovásábécé nagyrészt egyenes vonalvezetésű kanonikus alakokból tevődik össze, ezért nem meglepő, hogy lenyomata erősen korrelál a Patakfalvi illetve Nikolsburgi ábécék lenyomataival. Ugyanakkor Bél Mátyás-féle SzMR ábécéje, amely nagyrészt íves kanonikus alakokból tevődik össze, nullához közeli korrelációs együtthatót ad a Patakfalvi-, Nikolsburgi- illetve szabványosításra előterjesztett SzMR-ábécékkel való vizsgálatnál.

9. ábra. Vizsgált rovás ábécék korrelációs mátrixa.

ABC	Szabványosításra előterjesztett SzMR	Patakfalvi SzMR	Nikolsburg i SzMR	Bél Mátyás féle SzMR	Magyar Adorján féle SzMR	Rettegi István féle SzMR	Kájoni János (rég)i féle SzMR	Gyulafehérvári /portsalmi/ SzMR	Hickes - Harsányi SzMR	"Magyar ABC" / Gönczi György SzMR	Steppean Rovás	Középkori utánzat
Szabványosításra előterjesztett SzMR	1,000	0,973	0,992	0,049	0,982	0,777	0,796	0,936	0,291	0,571	0,905	0,147
Patakfalvi SzMR		1,000	0,972	0,137	0,952	0,816	0,767	0,923	0,388	0,589	0,930	0,204
Nikolsburgi SzMR			1,000	0,007	0,978	0,798	0,783	0,937	0,270	0,531	0,899	0,150
Bél Mátyás féle SzMR				1,000	0,058	0,234	0,443	0,269	0,868	0,794	0,337	0,836
Magyar Adorján féle SzMR					1,000	0,781	0,837	0,927	0,332	0,612	0,855	0,145
Rettegi István féle SzMR						1,000	0,570	0,739	0,440	0,544	0,791	0,268
Kájoni János (rég)i féle SzMR							1,000	0,921	0,594	0,866	0,774	0,568
Gyulafehérvári /portsalmi/ SzMR								1,000	0,439	0,714	0,898	0,461
Hickes - Harsányi SzMR									1,000	0,873	0,554	0,706
"Magyar ABC" / Gönczi György SzMR										1,000	0,698	0,708
Steppean Rovás											1,000	0,382
Középkori utánzat												1,000

[10] The MathWorks. Inc.: MATLAB, The Language of Technical Computing Version 7, September 2006. www.mathworks.com

[11] Füstös L. (2009): *A sokváltozós adatelemzés módszerei*. Budapest: MTA SZKI.

A SZÉKELY–MAGYAR ROVÁSÁBÉCÉ LENYOMATOK KLASZTERANALÍZISE MATLAB-PROGRAMMAL

Az ábécé lenyomatok klaszteranalízisét MATLAB R2009A program alkalmazásával végeztük el [10].

Az általános klaszterezési eljárás a következő:

1. Kiindul n db egyelemű csoportból (klasztérből), lényegkiemelés.
2. Megkeresi a hasonlósági (távolság-) mátrix maximális (minimális) elemét, vagyis a két leghasonlóbb klasztért.
3. A két klasztért összevonja, ezzel a klaszterek számát eggyel csökkenti. Az új klaszter többletől mért távolságát (hasonlóságát) újrászámítja.
4. A 2. és 3. lépést $(n-1)$ -szer elvégezve minden egyed egy klaszterbe kerül.

A módszerek abban különböznek, hogy hogyan definiáljuk a csoportok hasonlóságát (távolságát). Az összevonó eljárások eredménye a klaszterek hierarchikus elrendezését tükrözö kétdimenziós ábrán is megjelenik. Ha nem engedünk meg átfedö klasztereket, akkor *dendrogram*nak nevezzük ezt az ábrát [11].

A dendrogram vízszintes tengelyén az egyedek sorszámait, függöleges tengelyén pedig a klaszterek összevonásának szintjeit tüntetjük fel. A függöleges tengelyen hasonlósági vagy távolságmértékek találhatók az input adatoknak megfelelően. A vizsgálatunk során hierarchikus klaszterelemzési módszert alkalmaztunk.

A lényegkiemelés az egyedek, vagyis az ábécék leképezését jelenti valamilyen jól kezelhető kódba, esetünkben a primitívekkel elkészített lenyomat-vektorokba. A hasonlóság mérésének feltételei a következők: tekintsük az $X(n \times p)$ -s adatmátrix sorait (ábécé lenyomat), mint kanonikus alakok egy-egy pontnak a p dimenziós térben. E pontok között értelmezhetők metrikus tulajdonsággal rendelkező távolságmérö függvények. Jelöljük az (x,y) pontpár távolságát $d(x,y)$ -nal, amely minden $x, y, z \in R_p$ esetén az (1), (2), (3) és (4) tulajdonsággal rendelkezik:

$$d(x,y) = d(y,x) \tag{1}$$

$$d(x,x) = 0 \tag{2}$$

$$d(x, y) > 0, \quad \text{ha } x \neq y \quad (3)$$

$$d(x, y) \leq d(x, z) + d(z, y) \quad (4)$$

Ezek a metrikus tér általános tulajdonságai, és az ezeket kielégítő $d(x, y)$ függvényt metrikus függvénynek vagy röviden metrikának nevezzük [12]. Az euklideszi távolságmétrikát bemutatja az (5) képlet.

$$d_E(x, y) = \sqrt{\sum_{i=1}^p (x_i - y_i)^2} \quad (5)$$

ahol az x_i és y_i változók két vizsgált ábécé lenyomat-vektorának (x és y) megfelelő tagját jelölik. Az euklideszi távolságon kívül a hasonlóságot lehetséges a Manhattan-távolsággal (további elnevezései: *city block distance*, *taxicab distance*, *rectilinear distance*) is megadni, lásd a (6) képletet [12]. A kétféle metrika közül a vizsgálatainkban az euklideszi metrikát találtuk alkalmasabbnak az ábécék közötti különbség kifejezésére.

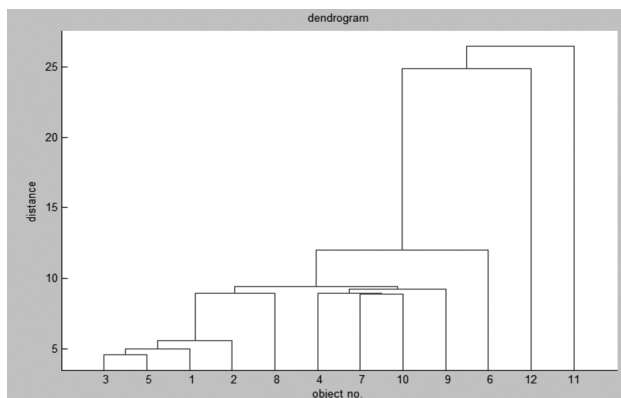
$$d_M(x_i, x_j) = \sum_{k=1}^n |x_{ik} - x_{jk}| \quad (6)$$

A MATLAB futásának eredményeként rajzoltatott dendrogramot a 10. ábra mutatja. A dendrogram elemzéséből arra a következtetésre jutottunk, hogy a vizsgált rovásábécék klaszterezése is az általunk várt eredményt hozta. Vagyis az egyenes vonalakat tartalmazó betűalakok által felépített SzMR-ábécék egy csoportba kerültek, míg az íves vonalakból felépülő betűalakokat tartalmazó SzMR-ábécék egy másik csoportba kerültek. Ugyanakkor a vizsgált ábécék tartalmaztak két olyan rovásábécét, melyek betűalak-tára eltér az SzMR-betűalakoktól, ezek külön-külön csoportokba kerültek. Az azonos csoportba került SzMR-ábécék között kicsi a MATLAB által számolt euklideszi távolság, míg a csoportok között relevánsan nagyobb.

A csoportokat úgy határozzuk meg, hogy a dendrogram y tengelyén kijelölünk egy távolságértéket, amin keresztül húzunk az x tengellyel párhuzamosan egy egyenest. Az egyenes és a klaszterek vonalának metszéspontja határozza meg a vizsgált csoportok, klaszterek számát.

[12] Krause, E. F. (1986): *Taxicab Geometry: An Adventure in Non-Euclidean Geometry*. New York: Dover.

10. ábra. MATLAB-program futásának eredménye.



Összefoglalás, következtetések

A cikkben bemutatásra került egy újonnan kidolgozott ismeretlen eredetű székely–magyar rovásírással készített feliratok stílusának és korának meghatározásához készített eljárás.

A kialakított eljárás az egyes grafémák topológiai tulajdonságainak modellezésén alapul. A grafémák kanonikus alakra bontásával lehetővé vált áttranszformálni a problémát egy absztrakt térben elvégezhető optimalizálási problémára. Ezzel a módszerrel lehetőségünk nyílt a matematikai statisztika módszereivel vizsgálni a rovásírással írt szövegeket. A kidolgozott módszer szükségességét számos jól olvasható, ugyanakkor ismeretlen eredetű szöveg létezése igazolja.

A cikkben ismert rovásábécékre készítettük el a lenyomatokat, és azok egymás közti kapcsolatát vizsgáltuk statisztikai módszerekkel. Az ezzel végzett kezdeti vizsgálatok alapján algoritmusunk pozitív eredményeket szolgáltatott. Ezen módszerünkre szeretnénk a továbbiakban kiterjeszteni további matematikai statisztikai módszereket, illetve beépíteni a szintén általunk kidolgozott székely–magyar rovásírás jelentésazonosító algoritmusunkba, amelyet nehezen olvasható feliratok, szövegek megfejtésére tervezünk alkalmazni.

Köszönetnyilvánítás

A munka szakmai tartalma kapcsolódik az *Új tehetséggondozó programok és kutatások a Műegyetem tudományos műhelyeiben* c. projekt szakmai célkitűzéseinek megvalósításához. A projekt megvalósítását a TÁMOP – 4.2.2.B-10/1-2010-0009 program támogatja.

Igényfüggetlen statikus elosztott útvonalválasztás – geometriai nézőpont

Összefoglalás: Az igényfüggetlen statikus elosztott útvonalválasztás (oblivious routing) alapötlete az, hogy egyetlen statikus útvonalválasztó függvény felelős a hálózaton megjelenő összes igény elvezetéséért. Az útvonalválasztó függvény jóságát a relatív jósági tényező (oblivious ratio) méri, amely az adott útvonalválasztás és az optimális útvonalválasztás relatív élterhelésének a hányadosát maximalizálja az összes lehetséges forgalmi mátrix felett. Irányítatlan hálózatok relatív jósági tényezője felülről korlátos a hálózatban levő csomópontok számának logaritmusában, és értéke általában kettő alatt marad irányított hálózatok esetében is. Ebben a cikkben felírjuk az igényfüggetlen statikus elosztott útvonalválasztás ekvivalens geometriai megfogalmazását, majd a geometriai képből kiindulva egy új mérőszámot, a *torlódás valószínűségét*, rendeljük hozzá. Megmutatjuk, hogy létezik a hálózatoknak egy olyan növekvő sorozata, amelyekre a relatív jósági tényező nem éri el a kettőt, de amelyekre belátható, hogy a torlódás valószínűsége minden határon túl közelít az egyhez. A mintahálózatokon tapasztalható viselkedést a valós hálózatokon végzett numerikus eredményekkel is alátámasztjuk.

Kulcsszavak: Igényfüggetlen statikus elosztott útvonalválasztás, útvonalválasztó függvény, konvex halmazok, poliéderek.

Abstract: The main idea of the demand-oblivious routing is that one single static routing function is used to make all the routing decisions in the network. Here, the goodness of the routing function, the routing performance is usually measured in terms of the competitive ratio, the proportion of the maximum congestion caused by oblivious routing to the best possible congestion. This competitive ratio is called sometimes oblivious ratio. Oblivious routing turned out to be surprisingly efficient in this regard: in undirected graphs we

* BME Távközlési és
Médiainformatikai Tanszék
E-mail: nemethgab@tmit.
bme.hu

** BME Távközlési és
Médiainformatikai Tanszék
E-mail: retvari@tmit.bme.hu

[1] Gunnar, S.–Johansson, M.–Telkamp, T. (2004): *Traffic matrix estimation on a large IP backbone: a comparison on real data*. ACM IMC, 2004. Pp. 149–160.

[2] Racke, H.: *Minimizing Congestion in General Networks*. FOCS '02. Pp. 43–52.

[3] Racke, H.: *Optimal hierarchical decompositions for congestion minimization in networks*. STOC '08. Pp. 255–264.

[4] Applegate, D.–Cohen E.: *Making intra-domain routing robust to changing and uncertain traffic demands: understanding fundamental tradeoffs*. ACM SIGCOMM 2003. Pp. 313–324.

[5] Azar, Y.–Cohen, E.–Fiat, A.–Kaplan, H.–Racke, H.: *Optimal oblivious routing in polynomial time*. ACM STOC '03. Pp. 383–388.

only pay a logarithmic performance penalty, and empirical studies indicate that the competitive ratio usually remains under 2 in directed graphs as well. In this paper, we sketch the geometrical framework of the static routing problem. Then we argue that the competitive ratio, being a worst case metric, hides crucial details on how oblivious routing performs when subjected to a broad range of diverse traffic matrices. We define a new performance metric, the probability of congestion, and we argue that these give a different, significantly less attractive view on the competitiveness of oblivious routing. Finally, by numerical evaluations we show that this adverse behavior is common in real networks.

Keywords: Demand-oblivious routing, static distributed routing, routing function, convex set, polytope.

Bevezetés

A modern hálózatok komplex és dinamikus rendszerek, amelyekben nehéz pontosan mérni és megbízhatóan megjósolni a felhasználók igényeit [1]. Ez problémát jelenthet a hálózatok adminisztrátorainak, akik általában az útvonalválasztást a forgalmi igények egy feltételezett halmaza mellett optimalizálják.

Az igényfüggetlen statikus elosztott útvonalválasztás („hanyag” útvonalválasztás, oblivious routing) egy lehetséges mód ennek a problémának a feloldására. Ekkor az útvonalválasztó függvény a forgalmi igények ismerete nélkül kerül optimalizálásra [2]. Alapvető ötlete, hogy egyetlen statikus útvonalválasztó függvény felelős a hálózaton megjelenő összes igény elvezetéséért. Az útvonalválasztás jóságát a relatív jósági tényező (oblivious ratio) méri, amely az igényfüggetlen statikus elosztott útvonalválasztás és az optimális útvonalválasztás relatív életterhelésének hányadosát maximalizálja az összes lehetséges forgalmi mátrix felett. Irányítatlan hálózatok relatív jósági tényezője felülről korlátos a hálózatban levő csomópontok számának logaritmusában [3], és értéke általában kettő alatt marad irányított hálózatok esetében is [4, 5].

A relatív jósági tényező egy worst-case mérőszám, hiszen az összes

lehetséges igény esetére a maximális torlódást méri. A következő fejezetekben megmutatjuk, hogyan lehet a hanyag útvonalválasztást geometriai szemszögből felírni, majd a következőkben numerikus eredmények bemutatásával megnézzük a hanyag útvonalválasztás korlátait is. A módszer előnyeit és hátrányait egy-egy esettanulmány is szemlélteti.

Geometria és hanyag útvonalválasztás

Legyen adva a hálózat a szokásos módon egy irányított $G(V, E)$ gráf alakjában, és legyenek adva az élek $c = [c_e : e \in E]$ kapacitásai. Azonosítsunk minden hálózati felhasználót egy egyedi (s_k, d_k) forrás-cél párral és a P_k statikus útvonalak halmazával, ahol $k \in \{1, 2, \dots, K\}$ a felhasználó sorszáma. Legyen P_k a k -dik felhasználó útvonalmátrixa, és jelölje minden k -adik felhasználó s_k csomópontnál jelentkező igényét θ_k . Egy routing, azaz hálózatban levő összes útvonalválasztó együttes konfigurációja, a hálózatban levő folyamok vektoraként írható fel, azaz $u = [u_k : k \in \{1, 2, \dots, K\}] \in \mathbf{R}^{p_1} \times \mathbf{R}^{p_2} \times \dots \times \mathbf{R}^{p_K}$, ahol P_k a k -adik felhasználó útvonalainak a száma és p az összes útvonal száma.

1. Definíció: Az
$$M = \left\{ u : \sum P_k u_k \leq c, u \geq 0 \right\} \subset \mathbf{R}^p$$

poliédert folyampoliédernek nevezzük [6, 7].

Legyen $\mathbf{T}$ egy affin transzformáció, amely egy adott u routinghoz megadja a hozzá tartozó forgalmi mátrixot:

$$\mathbf{T} : \mathbf{R}^p \rightarrow \mathbf{R}^K : u \rightarrow \mathbf{T}(u).$$

2. Definíció: Az M folyampoliéder $\mathbf{T}$ -vel vett leképezése a T megengedett igények tartományát adja [6, 7].

A megengedett igények tartománya az összes olyan forgalmi mátrix halmaza, amely valamely módon az élek túlterhelése nélkül elvezethető a hálózatban, azaz

$$T = \{ \theta : \exists u \in M, \mathbf{T}(u) = \theta \}.$$

Belátható, hogy T konvex és alulról monoton poliéder [6].

[6] Rétvári Gábor–Bíró József–Cinkler Tibor (2007): *Fairness in Capacitated Networks: A Polyhedral Approach*. IEEE INFOCOM. Pp. 1604–1612.

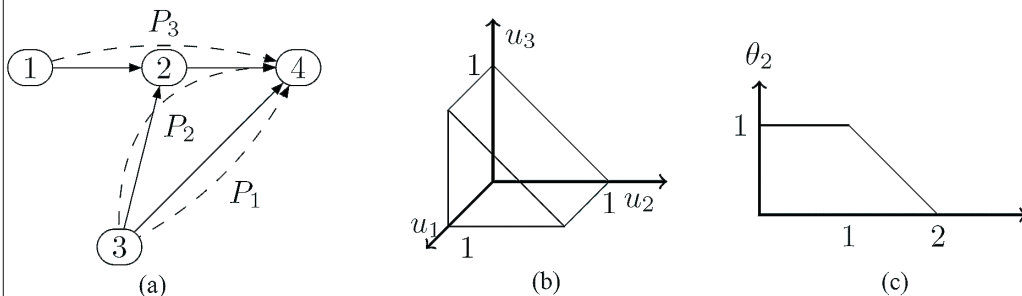
[7] Rétvári Gábor–Németh Gábor: *Demand-oblivious routing: distributed vs. centralized approaches*. IEEE INFOCOM '10. Pp. 1217–1225.

[7] Rétvári
Gábor–
Németh Gábor:
*Demand-
oblivious
routing:
distributed vs.
centralized
approaches.*
IEEE
INFOCOM
'10. Pp. 1217–
1225.

Vessünk egy pillantás az 1. ábrán látható hálózatra. Legyen két forrás-cél párunk, a (3, 4) és az (1, 4), rendre. Legyen továbbá minden él kapacitása 1. Ekkor a hálózat megengedett igényeinek a tartománya a következő módon írható le:

$$T = \{[\theta_1, \theta_2] : \begin{aligned} &\theta_1 + \theta_2 \leq 2; \\ &\theta_2 \leq 1; \\ &\theta_1, \theta_2 \geq 0 \end{aligned}\}$$

1. ábra. (a) Mintahálózat és a hozzá tartozó (b) folyampoliéder és (c) a megengedett igények halmaza.



Az útvonalválasztás geometriai leírásának egyik központi eleme maga az S útvonalválasztó függvény, amely megmondja, hogy az igények milyen módon vannak leképezve az útvonalakra: $u = S(\theta)$. Az útvonalválasztó függvényre teljesülnie kell a $\forall \theta \in \mathbf{R}^K : T(S(\theta)) = \theta$ összefüggésnek, azaz $S(\theta)$ -nak pontosan θ -t kell megvalósítania. A továbbiakban csak az $S : \mathbf{R}^K \rightarrow \mathbf{R}^p : \theta \mapsto S(\theta)$ alakú, lineáris útvonalválasztó függvényekkel foglalkozunk.

3. Definíció: Az $R(S) = \{\theta \in T : S(\theta) \in M\}$ halmazt az S útvonalválasztó függvény érvényességi régiójának nevezzük [7].

Egy útvonalválasztó érvényességi tartománya azon igények összessége, amelyek az adott útvonalválasztó függvénnyel a hálózat éleinek túlterhelése nélkül elvezethető a hálózaton. Az 1. ábrán vázolt hálózat érvényességi régiója az

$$S : \theta \mapsto \begin{bmatrix} 2/3 & 0 \\ 1/3 & 0 \\ 0 & 1 \end{bmatrix} \theta$$

útvonalválasztó függvény esetén a 2. b. ábrán (143. oldal) látható.

Igényfüggetlen statikus elosztott útvonalválasztás

A hálózatok klasszikus leírása geometriai fogalmak helyett általában folyamatokkal, folyamszintű leírókkal dolgozik. Épp ezért az igényfüggetlen statikus elosztott útvonalválasztás eredeti leírása is ilyen mennyiségekkel dolgozik. Jelölje $\kappa_S(\theta)$ az S útvonalválasztó függvény alkalmazása esetén a hálózatban a θ igény esetén kialakuló maximális relatív életterhelést, azaz legyen

$$\kappa_S(\theta) = \max_{(i,j) \in E} \frac{1}{c_{ij}} P_k^{ij} S_k(\theta)$$

Jelölje OPT_θ azt az útvonalválasztó függvényt, amely egy adott θ igény esetében minimalizálja κ_S -t.

4. Definíció: Egy tetszőleges S útvonalválasztó függvény relatív jósági tényezőjét a

$$\alpha_S = \max_{\theta \in T} \frac{\kappa_S(\theta)}{\kappa_{opt_\theta}(\theta)} \quad (1)$$

kifejezés adja meg [2, 3, 4, 8].

5. Definíció: Az

$$S^* = \operatorname{argmin}_S \alpha_S \quad (2)$$

optimalizációs feladat megoldását igényfüggetlen statikus elosztott útvonalválasztó függvénynek nevezzük [2, 3, 4, 8].

AZ IGÉNYFÜGGETLEN STATIKUS ELOSZTOTT ÚTVONALVÁLASZTÁS EKVIVALENS GEOMETRIAI FELÍRÁSA

Az előbb bemutatott geometriai keretrendszer segítségével az igényfüggetlen statikus elosztott útvonalválasztás egy ekvivalens felírása a következőképp adható meg:

$$\alpha_S = \min\{\lambda \in \mathbf{R}_+ : \forall \theta \in T, S(\theta) \subseteq \lambda M\}. \quad (3)$$

[2] Racke, H.: *Minimizing Congestion in General Networks*. FOCS '02. Pp. 43–52.

[3] Racke, H.: *Optimal hierarchical decompositions for congestion minimization in networks*. STOC '08. Pp. 255–264.

[4] Applegate, D.–Cohen E.: *Making intra-domain routing robust to changing and uncertain traffic demands: understanding fundamental tradeoffs*. ACM SIGCOMM 2003. Pp. 313–324.

[8] Németh Gábor–Rétvári Gábor: *Towards a Statistical Characterization of the Competitiveness of Oblivious Routing*. ACM SIGMETRICS 2012. Poszter.

Tétel: A relatív jósági tényező (1)-es és (3)-es megfogalmazása ekvivalens.

Bizonyítás: Vegyük észre, hogy az igényfüggetlen statikus elosztott útvonalválasztás esetében az α_S szférikus függvény, így elég csak a megengedett igények tartományának a felületén levő forgalmi mátrixokat vizsgálni. Tudjuk továbbá, hogy a megengedett igények tartományának felületi pontjai telítik a hálózat valamely élet, ezekre $\kappa_{OPT_\theta}(\theta) = 1$. Így az (1)-es optimalizálási feladat visszavezethető annak a kérdésnek a megválaszolására, hogy az adott útvonalválasztó függvény hányszorosan terheli túl az éleket, amely feladat pedig a (3)-as felírássra vezet.

Ha S lineáris leképezés, azaz $S : \mathbf{R}^K \rightarrow \mathbf{R}^P : \theta \mapsto F\theta$, akkor

$$\alpha_S = \min\{\lambda \in \mathbf{R}_+ : \lambda R(S) \supseteq T\}. \quad (4)$$

Tétel: Az igényfüggetlen statikus elosztott útvonalválasztás (3)-as és (4)-es megfogalmazása ekvivalens.

Bizonyítás:

$$(4) \Rightarrow (3): T \subseteq \lambda M \Rightarrow S(T) \subseteq S(\lambda R(S)) \stackrel{S \text{ lineáris}}{=} \lambda S(R(S)) \subseteq \lambda M, \text{ és}$$

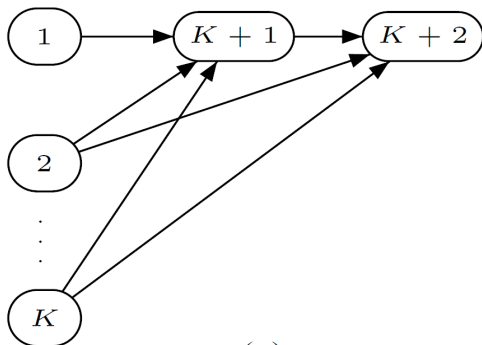
$$(3) \Rightarrow (4): S(T) \subseteq \lambda M \Rightarrow \mathbf{T}(S(T)) \subseteq \mathbf{T}(\lambda M) = \lambda T$$

1. ESETTANULMÁNY – A RELATÍV JÓSÁGI TÉNYEZŐ

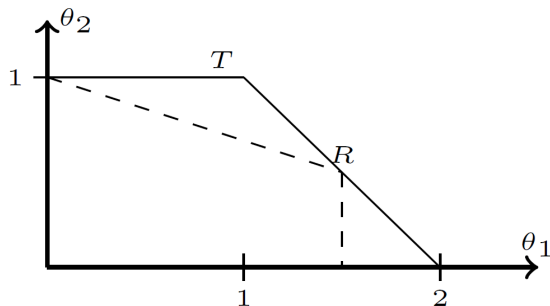
Határozzuk meg az 2. ábrán vázolt irányított hálózat igényfüggetlen statikus elosztott útvonalválasztás útvonalválasztó függvényét és a relatív jósági tényezőjét.

Az első forrás–cél-pár csak egy útvonallal rendelkezik, így az összes forgalma ezen az útvonalon kerül továbbításra. A többi felhasználónak két-két útvonala van. A hálózat magas szimmetriája miatt nyugodtan leszűkíthetjük a vizsgálódásainkat egy általános $k \in \{2, \dots, K\}$ felhasználóra.

2. ábra. (a) Mintahálózat a relatív jósági tényező és a torlódás valószínűségének a meghatározásához. (b) A megengedett igények tartománya és az érvényességi régió $K=2$ esetében.



(a)



(b)

Legyen β a k -dik felhasználó $K+1$ -es jelű csomóponton keresztül küldött forgalmának a részaránya. Két kritikus forgalmi szituációt vizsgálunk, amelyek a legnagyobb terhelést okozzák a hálózatban. Az egyik az $[1, \dots, 1] \in T$, amelyre a maximum terhelés a $(K+1, K+2)$ élen lép fel, és amelynek értéke $1 + \sum_2^K \beta$. A másik kritikus forgalmat egy általános $k \in \{2, \dots, K\}$ esetében a $[0, \dots, 0, 2, 0, \dots, 0] \in T$ igény okozza az $(s_k, K+1)$ élen. A maximális terhelés értéke ekkor $2(1 - \beta)$. A relatív jósági tényező meghatározásához a fellépő terheléseket szeretnénk minimalizálni, ehhez az $1 + \sum_2^K \beta = 2(1 - \beta)$ egyenletet kell megoldani. Az egyenlőségakkor áll fenn, amikor $\beta = \frac{1}{K+1}$, és ekkor a relatív jósági tényező értéke $\frac{2K}{K+1} \leq 2$.

[7] Rétvári Gábor–Németh Gábor: *Demand-oblivious routing: distributed vs. centralized approaches*. IEEE INFOCOM '10. Pp. 1217–1225.

[8] Németh Gábor–Rétvári Gábor: *Towards a Statistical Characterization of the Competitiveness of Oblivious Routing*. ACM SIGMETRICS 2012. Poszter.

Szép új világ?

Felmerül a kérdés: Megtaláltuk a Szent Grált, minden útvonalválasztás legjobblikát? Hisz' numerikus eredmények mutatják, hogy valós hálózatokban a relatív jósági tényező nem lépi túl a 2-t, ami azt jelenti, hogy az élek maximum csak kétszeresen lesznek túlterhelve. Ez pedig egy egyszerű tervezési receptet ad a hálózati adminisztrátorok kezébe. Nem beszélve a módszer könnyű implementálhatóságáról, amely elosztott tulajdonságából következik. Mindezek ellenére a kérdésre adandó válasz komoly megfontolást igényel. Az eddigiekben, amikor az igényfüggetlen statikus elosztott útvonalválasztást vizsgáltuk, akkor mindig a relatív jósági tényezőt mint mérőszámot rendeltük hozzá. Ez a mérőszám tényleg a módszer jóságát domborítja ki, de nem beszél az igényfüggetlen statikus elosztott útvonalválasztás statisztikus tulajdonságairól. A hálózati adminisztrátorokat nem csak a hálózatban fellépő maximális túlterhelés érdekli, hanem az is, hogy milyen gyakran terhelődnek túl bizonyos élek a hálózatban. Sajnos erre a kérdésre nem ad választ a relatív jósági tényező.

Ezért szükségünk van egy új mérőszámra, amely ezt a statisztikai tulajdonságot, a torlódás valószínűségét, adja meg. Legyen a torlódás valószínűsége egyenlő azzal a valószínűséggel, hogy a megengedett igények tartományából véletlenszerűen választott forgalmi mátrix nem vezethető el a megadott útvonalválasztó függvénnyel a hálózatban [7, 8]. Felhasználva a geometriai keretrendszer elemeit a torlódás valószínűsége felírható geometriai valószínűségként, mégpedig a következő alakban:

$$\eta_s = \frac{\text{vol}(R(S))}{\text{vol}(T)}$$

ahol $\text{vol}(\cdot)$ a K -dimenziós térfogatot jelöli.

2. ESETTANULMÁNY – A TORLÓDÁS VALÓSZÍNŰSÉGE

Becsüljük meg a 2. ábrán látható irányított hálózat torlódási valószínűségét az igényfüggetlen statikus elosztott útvonalválasztó függvény esetére.

A pontos érték meghatározásához mind a megengedett igények tartományát, mind az érvényességi régió pontos K -dimenziós térfogatát meg kellene határozni, de céljainknak megfelel a térfogatok könnyebben számolható becslült értéke is. Legyen $T' \subseteq T$ és $R' \supseteq R(S)$. Ekkor a torlódás valószínűsége a következőképp becsülhető:

$$\eta_s = 1 - \frac{\text{vol}(R(S))}{\text{vol}(T)} \geq 1 - \frac{\text{vol}(R')}{\text{vol}(T')}.$$

T' megkonstruálásához T monoton tulajdonságát felhasználhatjuk fel. Belátható, hogy a K -dimenziós 1 oldalhosszú hiperkocka teljes egészében T része. További $K+1$ darab fél hiperkocka is elfér még T -ben, amelyek a következőképp találhatók meg: helyezzük az 1 oldalhosszú hiperkockát a $[0, \dots, 0, 1, 0, \dots, 0]$ pontba és vegyük metszetét a $\theta_1 + \theta_2 \leq 2$ síkkal.

Legyen T' térfogata egyenlő az előbb meghatározott egész és fél hiperkockák térfogatának az összegével:

$$\text{vol}(T') = 1 + \frac{1}{2}(K-1)$$

Másfelől $R(S)$ térfogatát felülről szeretnénk becsülni egy jól megválasztott $R' \supseteq R(S)$ halmaz segítségével. Válasszuk R' -nek az R -t magába foglaló hiper-téglatest, amelynek a „bal alsó” sarka az origóban, míg a „jobb felső” sarka az $[1, 1 + 1/K, \dots, 1 + 1/K]$ pontban van.

Ekkor

$$\text{vol}(R') = \left(1 + \frac{1}{K}\right)^{K-1}$$

Végezetül

$$\eta_s = 1 - \frac{\text{vol}(R(S))}{\text{vol}(T)} \geq 1 - \frac{\text{vol}(R')}{\text{vol}(T')} \geq 1 - \frac{6}{K+2}$$

Az esettanulmányokat összehasonlítva egy érdekes megfigyelést tehetünk. Találtunk egy olyan irányított hálózatot, amelyben a relatív jósági tényező értéke kettő alatt van, de a torlódás valószínűsége a hálózat méretének növelésével egybe tart.

NUMERIKUS EREDMÉNYEK

[9] Mahajan, R.–Spring, N.–Wetherall, D.–Anderson, T. (2002): *Inferring link weights using end-to-end measurements*. ACM IMC. Pp. 231–236.

[10] Chinoy, B.–Braun, H. W. (1992): *The National Science Foundation Network, Tech. Rep.* CAIDA, available online: <http://www.caida.org/outreach/papers/1992/nsfn/nsfn-t1-technology.pdf>.

[11] Bollobás Béla (1997): *Volume Estimates and Rapid Mixing, Flavors of geometry* 31. Cambridge University Press. Pp. 151–182.

[12] Lovász László–Simonovits Miklós (1993): *Random walks in a convex body and an improved volume algorithm. Random Structures & Algorithms* 4, Wiley Subscription Services. Pp. 359–412.

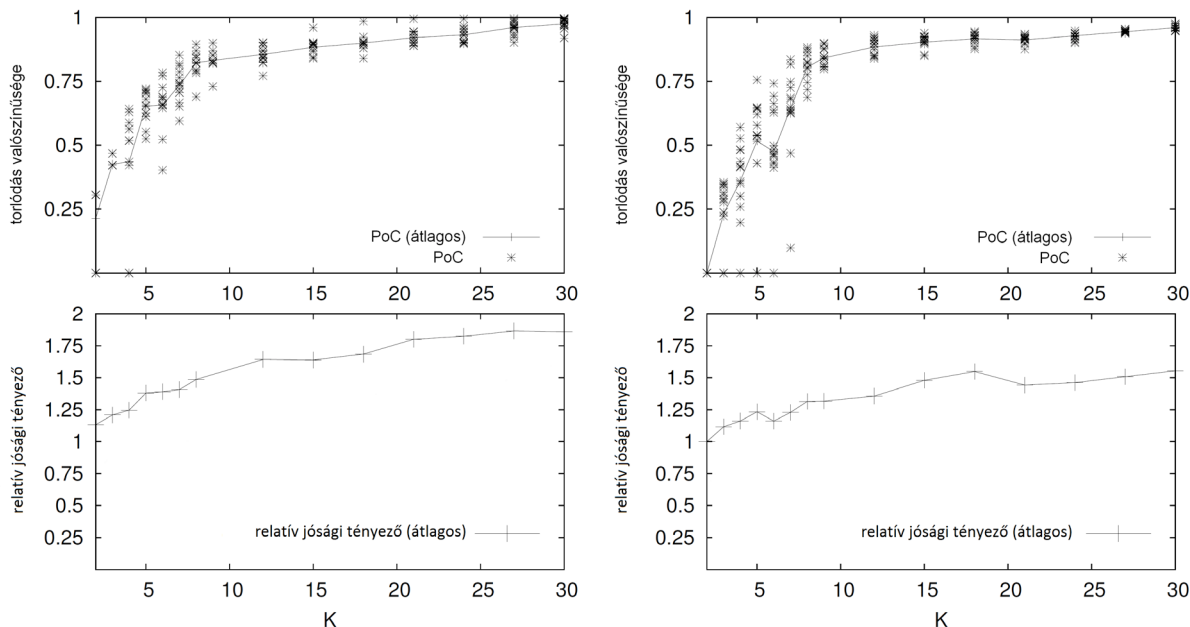
[13] Dyer, M.–Frieze, A.–Kannan, R. (1991): *A random polynomial time algorithm for approximating the volume of convex bodies*. J. ACM 38. Pp. 1–17.

Az előző esettanulmány eredményein felbuzdulva, felmerül a kérdés, hogy valós hálózatok esetében tapasztalható-e a fenti jelenség, azaz alacsony relatív jósági tényező mellett a torlódás valószínűsége egybe tart-e. Ennek a kiderítésére szimulációkat futtattunk különböző valós topológiákon. A szimulációk során használt ISP-hálózatokat a Rocketfuel-adatbázisból vettük [9]. Itt csak az AS3257-topológiára kapott eredmények kerülnek bemutatásra. További szimulációkat futtattunk az NSFNET Phase II hálózaton [10] is.

A fenti hálózatokból egyre bonyolultabb, egyre több forrás-cél-párt tartalmazó hálózatokat generáltunk. Minden egyes forrás-cél-pár szám esetében 15 különböző módon, véletlenszerűen módon adtuk a forrásokat és a nyelőket a hálózathoz. A forgalmat a bimodális eloszlás szerint határoztuk meg. A torlódás valószínűségének meghatározásához a minden alkalommal a megengedett igények tartományának és az érvényességi régió térfogatának a meghatározására lett volna szükség. Ez a feladat azonban nagy dimenzióban bonyolult [11, 12], így azokban az esetekben, amikor a forrás-cél-párok száma meghaladta a kilencet, a térfogatokat Monte-Carlo-módszerrel határoztuk meg [11, 12, 13]. Az approximációt úgy állítottuk be, hogy annak a valószínűsége, hogy a becslés hibája nagyobb, mint 10% legyen kevesebb, mint 10%. A relatív jósági tényezőt minden esetben pontosan számoltuk ki. Az eredmények a 4. ábrán láthatóak.

A numerikus eredmények azt mutatják, hogy a hálózatok az idő nagy részében túlterhelt állapotban vannak. Irányított hálózatok esetében a torlódás valószínűsége már 10 felhasználó esetében is eléri a 80%-t, és a felhasználó számát növelve ez az érték felkúszik 90% fölé. Mindeközben a relatív jósági tényező 1,5–1,8 körüli értéket mutat. Ezek az eredmények azt is mutatják, hogy az igényfüggetlen statikus útvonalválasztás pozitív tulajdonságai, amiket a relatív jósági tényező jelképezett, eltűnnek az útvonalválasztás statisztikus elemzése során.

3. ábra. Numerikus eredmények a NSF és az AS3257 hálózatok esetében.



(a) NSF (dir)

(b) AS3257 (dir)

Összefoglalás

A cikkünk röviden bemutatta az igényfüggetlen statikus elosztott útvonalválasztást, valamint annak egy lehetséges geometriai interpretációját. Két esettanulmányon szemléltettük, hogy habár az igényfüggetlen statikus elosztott útvonalválasztás jó működést jelez előre, statisztikai szempontból a hálózat az idő nagy részében túlterhelt állapotban van. Numerikus eredményekkel megmutattuk, hogy ez a jelenség valós hálózatokban is fennáll.

Velünk élő történelem (A teáskannától a Toy Story-ig)

* GlaxoSmithKline Kft.
E-mail: csutagabi@gmail.com;

** Gábor Dénes Főiskola
E-mail: berecz@gdf.hu

Összefoglalás: Írásunk egyéni nézőpontból ad áttekintést a számítógépes grafika fejlődéséről a kezdetektől a jelenig. Olyan úttörő személyeket mutat be, akik nevéhez a számítógépes grafika számos lényeges lépése kötődik, és akik nélkül ma nem lenne ennyire látványos a film és a szórakoztatóipar. Érdekes összefüggéseket felfedezhetünk fel a fejlődés mögött, és meglepetésszerűen bukkannak fel sokak által ismert, nagy nevek. Több tényt megismerve benyomásunk lehet arról, hogy mi kellett a számítógépes grafika „lavinájának” megindulásához, és az merre vitte a szereplők útját. A történet a korai 1960-as években kezdődött, amikor a számítástechnika csak gyerekcipőben járt, és a számítógépek szobaméretű bonyolult berendezéseként voltak jelen. Az idő haladtával láthatjuk az első számítógépes grafikai próbálkozásokat és azt a rögs utat, amelyet csak a legkitartóbbak és legtehetségesebbek jártak végig. A technikai ismertető, nevek, találmányok, eljárások, cégóriások sűrű kavalkádjában, a rohamos fejlődés viharában egy ember életének körvonalait rajzoljuk ki most élesen. Egy álmot, amelyet fiatalként megálmodott, és érett férfiként valóra váltott. Mi történt ez idő alatt? Történelmet írt... Vajon ki Ő? A kérdésre és sok egyéb másra is választ ad összefoglalásunk.

Kulcsszavak: Számítógépes grafika, számítógépes grafika úttörői, számítógépes grafikai találmányok, számítógépes grafikai eljárások, Edwin Earl Catmull.

Abstract: The presentation shows the history and evolution of computer graphics from the beginning to the present day from a unique point of view. It presents the pioneers of computer generated images, who took important steps in computer imagery and without whom today's movies and media show-business would be less attractive. We can find interesting connections behind this development and might be surprised by the well-known, great names we come across. After discovering a lot of interesting facts, we will

realize what was required to start the avalanche of CGI, and how it shaped the actors' own careers. The story begins in the early 60's when computer science was in its infancy and the machines were room-sized, bulky calculators. As time went by we witnessed the first attempts on this bumpy road the end of which only the most tenacious and most talented were able to reach. In the twisting, swirling mass of technical terms, famous people, inventions, methods and corporate giants, we can see a life's work of a man. The dream he had as a young person he realized as a mature man. What happened during this time? He wrote history ... Who is he? Our presentation will answer these and other questions.

Keywords: Computer Generated Images, pioneers of computer generated imagery, inventions of computer generated imagery, computer graphics methods, Edwin Earl Catmull.

Bevezetés

A 3D számítógépes grafikai programok nap mint nap használt eljárásai, parancsai közül sok (például antialiasing, light mapping, subdivision surface, texture mapping, z-buffering) a tudományterület kialakulásakor keletkezett. Jelenleg mégis olyan magyar nyelvű weboldal vagy elektronikus kiadvány, ahol részletesen foglalkoznának a számítógépes grafika történetével. A külföldi weboldalakon egy szemszögből (például egy cég vagy iskola története, személy vagy film kapcsán), vagy az eseményeket vázlatosan ismertetve és nem értelmezve írnak. Pedig több szálon futottak és futnak az események. Minél többet kutat valaki, annál érdekesebb összefüggések és információk bukkannak fel.

Dolgozatunkban először elhelyezzük a számítógépes grafikát a tudományok között, és megadjuk felhasználási területeit. Ez után az ún. utahi iskolában tevékenykedő tudósokról lesz szó, majd a New York Institute of Technology megalakulásáról, ahol többen is dolgoztak az utahiak közül. Végül Edwin Earl Catmull munkásságának néhány lényeges momentumát és eredményét foglaljuk össze. Igyekszünk felvázolni személyén keresztül is azt az ívet, amely az utahi teáskanna megalkotása és a Pixar Toy Story filmjének létrejötte között volt.

A számítógépes grafika helye a tudományokban, felhasználási területei

A számítógépes grafika a számítógép-tudomány egyik szakterülete. A 2D-s és 3D-s vizuális tartalom digitális szintetizálására és a geometriai adatok kezelésre használható módszereket tanulmányozza számítási technikák használatával. A számítógépes grafika a képgenerálás és -előállítás matematikai és számítási alapjaira koncentrál, nem pusztán esztétikai kérdésekre.

[1]
Sutherland,
Ivan Edward:
*Sketchpad: A
man-machine
graphical
communi-
cation system.*
Technical
Report,
UCAM-CL-
TR-574 ISSN
1476-2986
Number
574, 2003.
szeptember,
[http://www.
cl.cam.ac.uk/
techreports/
UCAM-CL-
TR-574.pdf](http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-574.pdf)

Sokak szerint a játék- és szórakoztatóiparnak, valamint a filmiparnak köszönhető a PC-k ugrásszerű fejlődése az 1990-es évek elejétől. A számítógépekre és a játékkonzolokra fejlesztett játékok ma már szinte mind valamilyen 3D motort használnak. A számítógépes grafika tudományterülete egyre szélesebb körben felhasznált, alkalmazási területei között van a digitális művészet (digital art), a grafikai tervezés (graphic design), az információ megjelenítés (information visualization), az információ-grafika (infographics), az oktatás (education), a racionális gyógyszertervezés (rational drug design), a számítógépes biológia (computational biology), a számítógépes fizika (computational physics), a számítógépes szimuláció (computer simulation), a számítógéppel segített tervezés (computer-aided design), a tudományos vizualizáció (scientific visualization), a virtuális valóság (virtual reality) és a webtervezés (web design) is.

Az utahi iskola úttörői

A számítógépes grafika alapjainak lerakásában kiemelkedő jelentősége van az 1965-ben David Evans által alapított, és 2000-ig működő Utah Egyetem Computer Science Departmenten kutató tanároknak és Ph.D. -hallgatóknak. Ezen úttörő tudósok többsége ma is tevékeny, számos esetben művészi vénával rendelkezik, sokuk jó üzletember.

IVAN EDWARD SUTHERLAND

A történet az 1960-as években kezdődik, amikor Ivan Edward Sutherland megindította a lavinát. Bár többen foglalkoztak számítógép segítségével történő grafikai megjelenítéssel, de az ő személyisége és munkája idézte elő a robbanásszerű fejlődést. Munkásságát mi sem igazolhatja jobban, mint a 2012. október 10-én neki kiosztott előkelő Kyoto-díj az információs tudomány kategóriában.

Sutherlandnek a Massachusetts Institute of Technology-n (MIT) 1961-ben elkészített *Sketchpad* nevű találmánya kiemelkedő jelentőségű. Ez volt 1963-as Ph.D.-dolgozatának (*Sketchpad, A Man-Machine Graphical Communication System* [1]) témája is, amelyet az ember-számítógép interakciók, a grafikus felhasználói felületek, a számítógéppel támogatott tervezés, valamint az objektumorientált programozás kezdetének tekintenek. A mai grafikai

interfészek sok szabványának elindulása kötődik hozzá – egy ezek közül például a rajzi megszorítások (kocka megrajzolásához például elég megadni, hogy kockát akarunk rajzolni, majd helyét és méretét).

1968–74 között Sutherland az Utah Egyetem professzora volt. Tanítványairól, akikkel számos kiemelkedő felfedezést tett, a következő alfejezetben szólunk.

Sutherland David C. Evans-szel 1968-ban megalapította az Ivans & Sutherland céget (lásd [2] History lapján), amely úttörő szerepet játszott a nyomtatók és a számítógépes grafikához felhasznált berendezések gyártásában és fejlesztésében, így mind szoftveres mind hardveres oldalról jól ismerte a grafikához szükséges eszközöket.

Ez a vállalatkészítette például a Picture System rendszert, amelyet kifejezetten 3D grafikai munkákhoz lehetett használni. A gépet monitorral, operációs rendszerrel és speciális szoftverekkel szállították, amelyen számos műveletet végezhetett a felhasználó (modelleket készíthetett, 3D perspektivikus nézetet használhatott, forgathatott, mozgathatott, méretezhetett, manipulálhatott). Az egyik leglényegesebb része az ún. frame buffer, amely gyakorlatilag egy hatalmas memória volt, itt tárolhatták átmenetileg az elkészült (renderelt) képeket, amely a mozgókép vagy animáció elengedhetetlen előfeltétele [3].

Az Ivans & Sutherland cég VAX számítógépekkel is foglalkozott, ezek az egyfelhasználós gépektől a mainframe óriás (szerver) számítógépekig terjedtek. Az 1980-as években, Ronald Reagan elnöksége alatt rengeteg berendezést állítottak elő az amerikai hadseregnek. Ezen kívül készítettek/készítenek például ESIG (Evans & Sutherland Image Generator) gépeket, amelyek szimulátorok, és nem csupán a vezetők, de a személyzet is használhatja polgári és katonai területeken. A sok lényeges tevékenység közül kiemelhetjük még a digitális színházi termékeiket, amelyek planetáriumokban és tudományos központokban világszerte megtalálhatók. [2]

[2] Evans & Sutherland Computer Corporation honlapja, <http://www.es.com>

[3] Evans & Sutherland Computer Corporation: The Evans & Sutherland Picture System, termékismertető, 1974,

KIEMELKEDŐ PH.D-HALLGATÓK

Sutherland az Utah Egyetemen a Computer Science Departmentben 1968–1974 között dolgozott. Itt is kiemelkedő munkát végzett, számos hallgatójával folytatta kutatásait a számítógépes grafika területén.

[4]University of Utah: U School of Computing, University of Utah: History <http://www.cs.utah.edu/school/history/>

[5] Carlson, Wayne E.: An Historical Timeline of Computer Graphics and Animation, <http://design.osu.edu/carlson/history/timeline.html>

Néhány ismert tanítványának munkásságából emelünk ki alább eljárásokat és mozzanatok. (A tudósokat vezetéknevük ábécérendje szerint említjük.)

Feltétlenül szólni kell *James Blinn*ről, aki a NASA sugárhajtómű laboratóriumában dolgozott az 1970-es években, és „mellesleg” komoly munkákat végzett „nyári gyakornokként” a 3D grafika területén. Hobbiját becsempézve a NASA-hoz, ő készítette a Voyager űrszondát prezentáló animációt. Phonggal együtt megalkotta a módosított Phong visszaverődési modellt, amely a Direct3D 10 és az OpenGL 3.1 előtt az alapértelmezett árnyalási modell volt a rögzített funkció pipeline-ban. [4]

Edwin Earl Catmull igen termékeny tudós, és mivel számunkra kiemelt jelentősége, külön fejezetben szólnunk róla.

James H. Clark több szilikonvölgybeli technológiai céget alapított (Silicon Graphics Inc., Netscape Communications Co., myCFO, Healtheon). Kutatómunkája során a 3D-s számítógépes képek gyors rendereléséhez fejlesztett ki rendszereket. A népszerű subdivision sémát Catmullal dolgozta ki. [5]

Danny Cohen diákként 1967-ben Sutherland vezetésével fejlesztette ki Cohen–Sutherland számítógépes grafika vonalkivágó algoritmust. Ezzel az egyszerű módszerrel a figyelembe veendő látótér meghatározása után, a kivágás utolsó lépéseként meghatározzuk, hogy a modelltől objektumai közül melyek esnek bele a látótérbe.

Franklin C. Crow továbbment az antialiasing módszerek fejlesztésében, jelentős munkája az első gyakorlati térbeli antialiasing technikák terén. Ő javasolta a shadow volume technikát a geometriailag pontos árnyékok előállításához. [5]

Henri Gouraud Ph.D-disszertációján, amelynek címe „Íves felületek számítógépes megjelenítése” (Computer Display of Curved Surfaces) 1971-ben Evans-szel és Sutherlanddel együtt dolgozott. Emberi arcképeken mutatta be árnyalásának effektjét, amelyhez felesége, Sylvie Guaraud volt a modell, akinek arcát digitalizálta be. A Gouraud árnyalási technikát róla nevezték el. Munkája nagy előrelépés volt a karaktermodellezés felé is. [4]

Alan Kay a Smalltalk nyelv feltalálója. Ezen kívül jelentős az objektumorientált programozás és az ablakozó grafikus felhasználói felület tervezése terén a korai időkben végzett úttörőmunkája is. Az ő nevéhez kötődik 1971-ből „A legjobb módja annak, hogy megjósoljuk a jövőt, ha

kitaláljuk.” („The best way to predict the future is to invent it.” [6])

Martin Newell azt az „átkozott” teáskannát készítette, amely a mai napig alapobjektuma a 3D grafikai innovációknak. A modellre azon probléma kapcsán talált, hogy nem volt elég érdekes modellje Ph.D-kutatásaihoz az algoritmusok tesztelésére. Sandra, a felesége javasolta egy komplett teáskészlet ötletét. 1975-ben először papíron megrajzolta a teáskannát, majd Tektronixba kézzel írta be a felületeket leíró Beziér-modell kontrollpontjait. Összetettsége miatt megfelelő volt minden algoritmus (például bump map, environment map) tesztelésére. Diplomája megszerzése után ott maradt az egyetemen 1979-ig, majd később a Xeroxnál dolgozott. 1988-ban megalapította az Ashlar CAD (Computer Aided Design) szoftvercéget. [7]

1. ábra. Az eredeti utahi teáskészlet huzalvázmodellje [8]



Utah olyannyira vonzó volt, hogy még például a vietnámi születésű Bui Tuong Phong is az Egyesült Államokba utazott, hogy az intézményben tanulhasson. Ő többek között a róla elnevezett fényvisszaverődési modell megalkotója. Ez az ún. specular fényvisszaverődés az a „csillanás”, amely a tárgyak felületén látható, amikor egy erős fényforrás fénye közvetlenül a szembe vagy a kamerába „pattan”. Sokkal valóságosabbá teszi a modelleket. Emellett például a róla elnevezett árnyalási (shading) interpolációs módszert is feltalálta. [4]

[4] University of Utah: U School of Computing, University of Utah: History <http://www.cs.utah.edu/school/history/>

[6] Kay, Alan C.: *Predicting The Future*. Stanford Engineering, Volume 1, Number 1, Autumn 1989. Pp. 1–6, <http://www.ecotopia.com/webpress/futures.htm>

[7] Vass Gergely: *A Cornell box és a Utah Teapot története*. Vass Gergely honlapja. http://www.vassg.hu/cornell_teapot.htm

[8] Glydeck: *Who remembers the Utah Teapot?* 2011. augusztus 11., <http://glydeck.blogspot.hu/2011/08/who-remembers-utah-teapot.html>

[4] University of Utah: U School of Computing, University of Utah: History <http://www.cs.utah.edu/school/history/>

[9] McDermott, R.: Robert Remembers: The VW Bug. *The Utah Teapot*. 2003/ősz. <http://www.cs.utah.edu/news/teapot/archives/fall03.pdf>

[10] Smith, A. R.–Heckbert, P.: *Brief History of the New York Institute of Technology Computer Graphics Lab*. Szemelvény Terrence Masson számítógépes grafika története könyvéből, <http://www.cs.cmu.edu/~ph/nyit/masson/nyit.html>

Phong olyan sikeres ember, akinek rövid életet szánt a sors, viszont annál lelkesebben és eredményesen élte azt. Például benne volt Sutherland, Robert McDermott, Clark és Raphael Rom mellett abban a csapatban, amely Sutherland Volkswagen Bogarát 1970-ben bedigitalizálta, és elkészítette az első olyan realisztikus, számítógéppel generált képet, amely megegyezett a fizikai másával. A modell sokáig a grafika egyik tisztelt és elismert jelképe volt. [9]

Bob Sproull 1968-ban Sutherlanddel, annak diákjaként létrehozta az első virtuális valóság és kiterjesztett valóság fejeire illeszthető kijelző rendszert (head-mounted display system), amely Damoklész kardjaként („The Sword of Damocles”) ismert.

Végül a diákok közül *John Warnock*ot említjük meg, aki részt vett a NASA űrrepülőszimulátorának és az Evans & Sutherland cégnél repülőgépszimulátorok kifejlesztésében. A róla elnevezett rekurzív felosztó algoritmus kidolgozója rejtett felületek eltávolításához. Az Adobe Systems Inc. társalapítója és a PostScript nyelv egyik kifejlesztője, a PDF fájlformátum alapjainak kidolgozója. [4]

A New York Institute of Technology elindulása

A New York Institute of Technology intézményt (NYIT) Alex Schure alapította 1974-ben. Célja az volt, hogy a kor elérhető legjobb technikáival készítsenek animációs filmeket. Első lépésként felvásárolta gyakorlatilag a teljes utahi számítógépparkot (amely nagy része Evans & Sutherland termék volt), és elszállította New Yorkba. Továbbá kereste azokat a tehetségeket, akikkel együtt tudott dolgozni, így felkérte Catmullt és Malcolm Blanchardot, hogy vezessék az intézményt, akikhez hamarosan csatlakozott Fred Parke, Garland Stern, Alvy Ray Smith és mások Utahból. [10]

A társaság fő tevékenysége kezdetben az volt, hogy létrehozzák azokat a szoftvereket, amelyekben később dolgozhatnak. Így jött létre többek között az animáláshoz a Tween, a rajzoláshoz a Paint és az animációhoz a SoftCel. Hozzájárultak a képi technológiák fejlődéséhez animáció, fraktálok, morphing, kompozitálás, texture mapping és sok más területen. A társaság tagja volt még: Gene Miller, Lance Williams és Michael Chou, akik továbbfejlesztették a

reflection mapping renderelési technikát, amely arra szolgált, hogy a tárgyak csillogása élethű legyen a televízióban és mozikban (például a *Flight of the Navigator*, a *Terminator*, az *Abyss* filmekben). [10]

A Paint programot később eladták az Ampex cégnek, ahol az Ampex AVA-rendszernek lett a fő komponense. Ezt többek között LeRoy Niemann használta a Super Bowl '78 alatt az interaktív rajzokhoz.

A képek minősége és a NYIT munkája felkeltette George Lucas figyelmét is, aki érdeklődött a számítógépen létrehozott kép (Computer-Generated Imagery, CGI), a speciális effektek fejlődése iránt. Megalapította saját cégét, a Lucasfilmet, megkereste a legjobb tehetségeket a NYIT-nél, és felvette Catmullt, Smith-t és Guggenheimet a csapatába.

[10] Smith, A. R.–Heckbert, P.: *Brief History of the New York Institute of Technology Computer Graphics Lab*. Szemelvény Terrence Masson számítógépes grafika története c. könyvéből. <http://www.cs.cmu.edu/~ph/nyit/masson/nyit.html>

Edwin Earl Catmull

Az utahi számítógépes grafika úttörői közül egy valaki neve fényesen ragyog a grafika történelmének egén, aki nem más, Edwin Earl Catmull. Gyerekként rajongott a Walt Disney rajzfilmekért, és nagyon szeretett volna azok közé tartozni, akik ezeket a mesés rajzfilmeket készítették. Álmát akkor sem adta fel, amikor közölték vele, hogy kézügyessége nem elég jó. Matematikát, fizikát és számítógép-tudományokat tanult.

Élete akkor kezdett a grafika ösvényére kanyarodni, amikor működés közben látta Sutherland sketchpadjét. Új célt talált egy egész estés, teljes egészében számítógéppel készített animációs filmben. Belevetette magát a számítógépes grafika tudományába. Hamar felfigyeltek kivételes tudására és kitartó munkájára, amivel gyorsan az élbolyban találta magát először a legendás Utah Egyetemen, majd később a NYIT falai között Schure oldalán. Catmull neve mögött eljárások, algoritmusok, találmányok hosszú sora húzódik, amelyekkel kivívta magának a tiszteletet, és vezető szerepet szerzett tudománya területén. Csak néhány fontosabb ezek közül:

- *Z-buffering*: A z-buffer a képek mélységadatát tárolja. A grafikus kártya ez alapján számolja ki, hogy ha egy pixelre például két objektum vertexei kerülnek, melyiket számítsa ki és jelenítse meg.

[5]
Carlson,
Wayne E.:
*An Historical
Timeline of
Computer
Graphics and
Animation.*
[http://design.
osu.edu/
carlson/
history/
timeline.html](http://design.osu.edu/carlson/history/timeline.html)

– *Subdivision surface*: A subdivision surface-t, azaz a felületfelosztást Catmull és Clark együttesen dolgozta ki 1978-ban. Az eljárás a nyers modellt többfelé osztja, ezáltal simább felületet eredményez. Az algoritmus rekurzív, meg kell adni a meghívások határát, ez befolyásolja a felület simaságát. Az új vertexek és felületek pozíciója a korábbi élek és felületek arányából számított.

– *Texture mapping*: A texture mapping a tárgyak textúrázására szolgáló eljárás, amikor külön eltárolt, mintázatot ábrázoló bittérképes képet vagy procedurális mintázatot viszünk fel 3D-s objektum felületére. Az objektum vertexei hozzárendelésre kerülnek a textúra koordinátaíhoz. Multitexture esetén egy vertexhez vagy objektumhoz több textúra is hozzá van rendelve.

2. ábra. Példák texture mappingre [5].

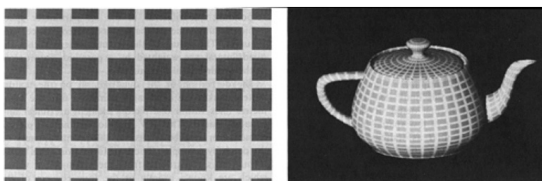
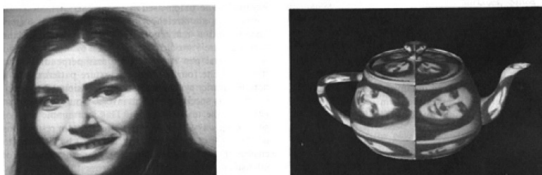


Fig. 3. Hand sketched texture pattern: left-hand side shows texture pattern; right-hand side shows textured object.



Fig. 4. Photographic texture pattern: left-hand side shows texture pattern; right-hand side shows textured object.



– *Light mapping*: Light mapping esetén a fénytöréseknek külön textúrát adunk meg. Előre kiszámolt adatstruktúra, amely a felületek világosságát tárolja.

– *Antialiasing*: Az antialiasing vagy más néven vonalsimítás-eljárás arra a problémára nyújt megoldást, amikor nagyfelbontású képet kisebb felbontáson próbálunk megjeleníteni. Ekkor nem várt torzítások és zajok keletkeznek a képben. Az eljárás végigmegy a kép pixelein, majd egy közelítő értékkel újra kiszámítja a pixeleket. A legmemóriakímélőbb eljárás, ha minden pixel köré 4x4-es rács kerül, és ebben a környezetben kerül átlagolásra az adott pont. Ezáltal a vonalak széle nem lesz „fűrész”, hanem kicsit elmosott, viszont az emberi látáshoz ez közelebb áll, ezért a képet élesebbnek érzékeljük.

Az első animációt, amelyet Catmull látott, 1972-ben a saját kezéről készítette [11]. A film ma már az Egyesült Államok filmarchívumának része. Négy évvel később, 1976-ban újra felfedezték Hollywoodban, és felhasználták a *Futureworld* című sci-fi filmben [12].

A számítógépes grafika nagy támogatói, vezetői cserélődtek, mint például George Lucas, akinek hite, elhivatottsága és pénztárcája óriásit lendített a számítógépes grafikán, és aki népszerűsítette a televízión és a mozin keresztül. Őt követte Steve Jobs, aki szintén nagy fantáziát látott a területben, és akinek a Pixar Studiót köszönhetjük, amely annyi szórakoztató, számítógéppel készített produkciót készített már. A tulajdonosok és szponzorok változtak, de egy valamiben mindenki egyet értett: Catmullnak kell vezetnie a csapatot. [13]

Miközben találmányok és sikerek születtek, elért egy újabb mérföldkövet, 1995-ben a Pixarnál megszületett az első egész estés, teljesen számítógéppel készített animációs film, a *Toy Story*.

[11] Catmull, E. E.–Parke, F.: *A Computer Animated Hand*. University of Utah, 1972. <http://www.youtube.com/watch?v=QfRAfsK5cvU>

[12] Heffron, R. T.: *Futureworld* (detail). The Aubrey Company, 1976. <http://www.youtube.com/watch?v=QfRAfsK5cvU>

[13] Price, D. A.: Book Review: The Pixar Touch: The Making of a Company Author, Alfred A. Knopf, 2008. május, <http://www.renderosity.com/news.php?viewStory=14010>

[14] Ndalianis, A.:
Traditional
animation, the
computer finish.
RealTime issue #81
2007. október–
november. [http://
realtimearts.net/
article/81/8711](http://realtimearts.net/article/81/8711)

3. ábra. A Toy Story storyboardjának egy képe [14]



Majd érett férfiként, 2006-ban felajánlották neki a legelőkelőbb helyet, amiről gyermekként csak álmodhatott, a Walt Disney elnöki székét.

Catmull munkásságát, amelyet tudósként, valamint a NYIT, a Pixar, a Walt Disney vezetőjeként végzett, több kiemelkedő szakmai díjjal ismerték már el. A 2009 februárjában megrendezett 81. Oscar-díj átadáson a Gordon E. Sawyer-díjjal (hivatalos angol megnevezése Academy Award) tüntették ki, amelyet azon személyek kapnak, akik technológiai munkássága kiemelkedő a mozgóképipar számára. A legutóbbit, a Museum Fellow címet, 2013-ban kapta a Computer History Museumtól a számítógépes grafika, animáció és filmkészítés terén végzett úttörő munkájáért.

Összefoglalás

A számítógépes grafika területén végzett munkájuk során az úttörők, akik közül sokat megemlégtünk az előző fejezetekben, kitalálták a szükséges elveket, eljárásokat, elkészítették a hardvereszközöket, szoftvereket, tanítottak, cégeket alapítottak, filmek készítését tették lehetővé. Az 1960-as évektől napjainkig Ivan Sutherland, mint „a számítógépes grafika atyja”, egy egész generációt ösztönzött és segített munkájával. A Utah Egyetem a 3D grafika bölcsőjévé vált, a tehetsős befektető, Alex Schure által alapított NYIT pedig az iskolájává.

Az általa megálmodott Computer Graphics Laboratory-t pont a hatalmas Rockefeller birtok mellé építette, és többek között az Evans & Sutherland cég berendezéseit vásárolták meg munkájukhoz. George Lucas is kivette a maga részét a tortából az általa alapított Lucasfilmmel, ahol Edwin Catmull bebizonyította, hogy az álmok valóra válthatók. Az Industrial Light and Magic (ILM) azóta készített CGI képeket, trükköket, mióta felbérelték Edwin Catmullt a NYIT-től 1979-ben. A grafikai részleget később eladták Steve Jobs-nak, aki elnevezte Pixarnak, és megalkották az első, egész estés, teljes egészében számítógéppel készített filmet, a Toy Story-t – amelyben Martin Newell akkor 20 éves teáskannája is megjelent egy rövid teaparti jelenetben.

Development opportunities of carpooling in Hungary

* *Central European University*
E-mail: toth_csaba_02@ceu-budapest.edu

** *College of Dunaujváros*
E-mail: agoston@mail.duf.hu

Abstract: This paper aims to ascertain the current and to predict the future position of carpooling in Hungary by analyzing two surveys in order to assess its potential development opportunities in Hungary. The surveys, which targeted the general public and the users of a carpooling scheme in Hungary, were performed to identify travel patterns, the willingness to carpool, as well as the barriers and possibilities of carpooling. The analysis of those surveys revealed that although carpooling has been gaining significant popularity for the last five years there are still huge reserves among the Hungarian population. On the other hand, the need of adapting to others signifies a considerable barrier to the further spread of carpooling. Nevertheless, this barrier can be overcome by smartphone applications which allow real-time carpooling. The suitable features of such an application as well as other development opportunities are also described in this paper.

Keywords: Carpooling, real-time carpooling, smartphone application.

Összefoglalás: A cikk célja meghatározni a telekocsi-(carpooling) rendszer jelenlegi elterjedését Magyarországon, és megjósolni a jövőbeni helyzetét. Ehhez két kutatás felhasználása, elemzése történik. A kutatások egyrészt mindenki felé, másrészt a telekocsit már használó személyek felé irányultak, felmérték az utazási szokásokat, a telekocsival történő utazási hajlandóságot, valamint a korlátokat és lehetőségeket. A felmérések elemzése során kiderült, hogy bár a telekocsi egyre népszerűbb lett az elmúlt öt évben, még mindig hatalmas tartalékok vannak a magyar lakosság körében. Másrészt, a másokhoz történő alkalmazkodás jelentős akadályt jelent a további terjedéshez. Mindazonáltal ezt az akadályt le lehet küzdeni okostelefon-alkalmazásokkal, amelyek lehetővé teszik a valós idejű telekocsizást. Egy ilyen alkalmazás szükséges funkciói, valamint egyéb fejlesztési lehetőségek is leírásra kerülnek.

Kulcsszavak: Telekocsi, carpooling, valós idejű telekocsi, okostelefon-alkalmazás.

Introduction

Road transportation methods are in transition during the recent decades all over Europe and in Hungary. Flexible forms appear both in private and public transport. New services in private transport are carsharing and carpooling, whilst in public transport the demand responsive transport and community bike system. This article deals with the carpooling.

Carpooling

DEFINITION, CHARACTERISTICS

There are several definitions of carpooling. The broadest version is the collaborative use of a car by at least two persons for a certain ride determined by the driver's will, without the primary purpose of gaining profit (though the travel cost may be shared among the carpoolers). An important feature is that these cars are privately owned. This definition might be narrowed by excluding some categories of users: while the broader conception of carpooling (somewhat more common in the U.S.) allows even household members as passengers (though this is usually distinguished by the term of “fampooling” or internal carpooling) [1], the narrower one, usually applied in Europe, excludes household members as well as persons with accompanying purposes (external carpooling) [2]. The narrower conception can be further narrowed down according to the way of arranging the ride:

- Hitchhiking is a completely random form of carpooling when the ride (destination, etc.) is arranged anywhere on the road where the driver and passenger(s) meet, the passengers cannot predict the waiting time until they are picked up if it takes place at all, and the ride is usually free.
- In the case of ‘slugging’, drivers and passengers meet at defined locations, the routes are set, the passengers can expect to be picked up in a relatively short time, and the ride is always free (but sharing rides allows drivers to go faster by using designated lanes).

[1] Li, J. et al.: Who Chooses to Carpool and Why?: Examination of Texas Carpoolers. *Transportation Research Record*. 2007/2021: Pp. 110–117.

[2] Vanoutrive, T. et al.: What determines carpooling to workplaces in Belgium: location, organisation or promotion? *Journal of Transport Geography*. 2012/22: Pp. 77–86.

[3] Certu. 2008. L'autopartage en France et en Europe. État des lieux et perspectives [Carsharing in France and in Europe. Situation and perspectives]. Lyon. http://www.communauto.com/images/03.coupures_de_presse/CERTU_Autopartage_France_Europe.pdf (2012. 11. 29.)

[4] Collaborative Consumption: Who are the users? <http://www.carpooling.com/press/companypressnews/press-releases/carpooling-user-profile/> (2012. 07. 03.)

[5] Avego. <http://www.avego.com> (2012. 10. 03.)

[6] Kenguru Lift Centre. http://www.kenguru.hu/?p=nyito&l=_en (2012. 10. 04.)

[7] Cs. Köbli, A. Operator of Kenguru Lift Centre. Email communication. (2012. 06. 13.)

– In the case of 'modern' carpooling all details (place and time of departure and arrival, amount of financial contribution, etc.) are set in advance through a medium (usually the internet), and passengers usually contribute to the travel cost. Real-time carpooling is the state-of-the-art form of this version, when setting the detail can take place as little as a few minutes in advance of the meeting through e.g. smartphone applications and the passenger is picked up in her/his current location.

In this paper we use the term 'carpooling' for the modern (3rd) version.

THE HISTORY OF CARPOOLING

The modern form of carpooling has gained popularity since the 1980s though there were some informal initiatives in the 1960s and 1970s, such as Allostop Association in France [3]. Non-operation of public transportation during general strikes gave a temporary boost to carpooling in several places, but it was the spread of the internet, which triggered a "mass" carpooling, as it made arranging rides much more easy and effective. Today, the European-level carpooling.com scheme alone has 3.6 million users (more than 1% of the population of the operating area) and moves around one million people per month [4].

Most recently the real-time applications have started to appear, too. One of these schemes is the Avego Iphone application in the U.S. which can make a match between drivers and passengers even immediately after setting the destinations, and if everybody agrees, it directs the parties to a suitable place close to the passenger. In the end, it automatically transfers the cost of carpooling, which is set in advance on the base of distance, from the passenger(s) account to the driver account [5].

In Hungary, organized carpooling has been present at least since 1991, when the Kenguru Lift Centre [6] started its operation, though this centre has been organizing only international trips [7]. Later, as the internet became more and more available and globalized, more and more Hungarian departure and destination points were added to Western European or international

carpooling systems, such as www.mitfahrgelegenheit.de or <http://hitchhikers.org>; the latter was even translated into Hungarian. A real boost for a national system came in 2007, when www.oszkar.com started operating. It was soon followed by utazzunkegyutt.hu, telekocsi.eu and most recently by collectivetraffic.net. At present kenguru.hu, oszkar.com and utazzunkegyutt.hu, which are the most popular schemes, have approximately 30,000 [7], 31,000 [8] and 27,000 [9] users respectively, though the user communities of the two latter schemes might significantly overlap (at the end of 2012 kenguru.hu seems to be closed). The scheme of oszkar.com allows arranging carpooling through Android smartphone application since July, 2012, which is a significant step towards real-time carpooling.

[7] Cs. Köbli, A. Operator of Kenguru Lift Centre. Email communication. (2012. 06. 13.)

[8] Prácser, A.–Gyűrűs, M.: Operators of oszkar.com carpooling scheme. Informal interview. (2012. 05. 12.)

[9] [Utazzunkegyutt.hu](http://utazzunkegyutt.hu) telekocsi rendszer. <http://www.utazzunkegyutt.hu/> (2012. 07. 13.)

Carpooling researches in Hungary

Two surveys were performed about carpooling in Hungary in 2012. A national survey on a representative sample of 1,200 persons over 18 in Hungary in 18–22 May, 2012 (hereinafter referred to as ‘national survey’) was performed by Policy Solutions with help of Median Public Opinion and Marketing Research Institute on behalf of the Clean Air Group (*Levegő Munkacsoport*). Apart from demographic data, the level of willingness to use carpooling to work and arguments for and against it were gathered. Nevertheless, the sample size of one of the most important groups, those who travel to work by car is very small (153 persons), which entails a rather large margin of error (3.95%) at a 95% confidence level. The data of this survey were analyzed in order to get an insight about who are the potential carpoolers as well as what are the barriers and potential drivers of carpooling in the case of the general public.

The other survey was performed by Csaba Tóth in the frame of the Master’s in Environmental Studies and Policy of the Central European University. In this survey the opinions of carpoolers were gathered by the researcher through an online questionnaire (hereinafter referred to as ‘carpooler survey’) between 1st and 24th of June 2012. The users of the oszkar.com scheme, the most popular carpooling site with 31,220 registered members on 19 May, 2012 participated in the survey, which included 24 questions

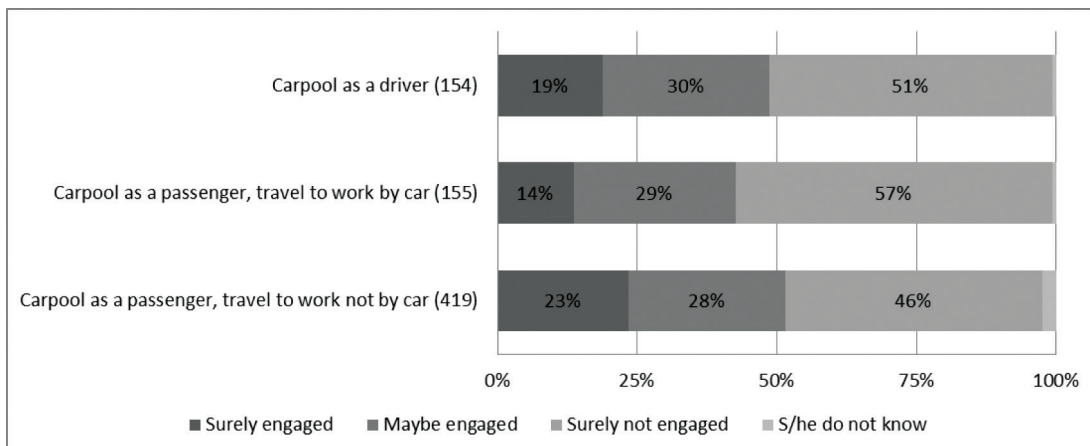
regarding the demography, the travel purposes, the travel patterns and the experiences of carpoolers as well as the barriers which discourage them from carpooling more intensively and the reasons why they carpool. The latter addressed not only the motivations for carpooling, but those possibilities as well which would encourage users to carpool more frequently. As some questions do not make sense for both driver and passenger carpoolers (e.g. drivers might be not able to answer a question regarding their reasons to favour carpooling over public transportation as they might not use the latter), two different questionnaires were designed for the two groups. The links to the online questionnaires originally were sent by email to 2,074 randomly selected passengers as well as 1,981 randomly selected drivers but as drivers' response rate was low (235 of them responded within two weeks) 1,000 additional drivers were emailed about the driver questionnaire. In order to raise the response rate, USB memory sticks were offered for 3 randomly selected passengers and drivers in each group among those who filled in the questionnaires. The questionnaires were filled in by 348 passengers and 468 drivers (17% and 15.6% response rate, respectively). In addition, the database of offered and booked rides in 2011 (hereinafter referred to as 'route database'), which were provided by the oszkar.com operators, was analyzed, too. The databases contained the number of rides offered as well as booked for each route in 2011, and the number of rides offered in the different days of the week.

This paper aims to review the current situation and the development opportunities of carpooling in Hungary.

The results of the researches

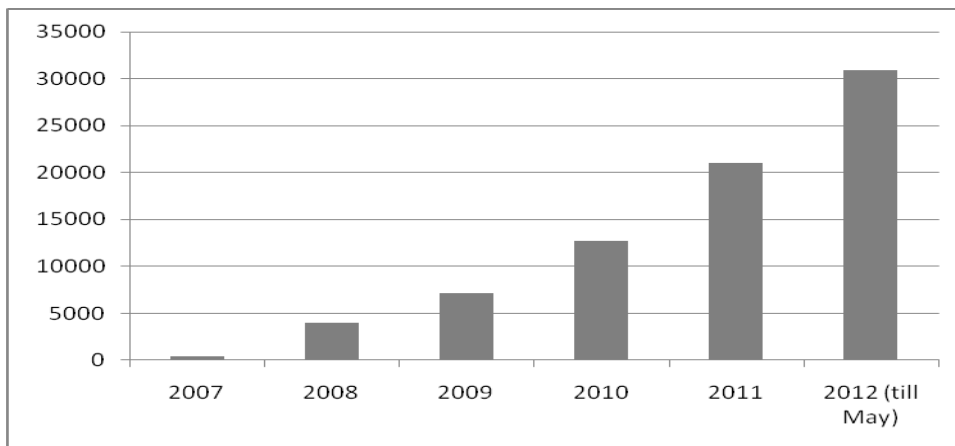
The demographical data of the carpooling and national survey suggest that *there is a considerable potential for carpooling in Hungary*. While based upon the national survey the 19–23% of the approximately four million working people (that is approximately 800 thousand people, *Figure 1.*) would be surely willing to carpool (as a driver or as a passenger), there were only 31 thousands registered users in the oszkar.com scheme in May, 2012, though this number has been dynamically growing since 2007 when the oszkar.com started its operation (*Figure 2.*). The aggregated number of all users of different carpooler scheme is unknown, but as the oszkar.com is the most popular scheme, the whole carpooler population is probably well below 100 thousand people.

Figure 1. Level of interest in carpooling (number of respondents).



Data source: Policy Solutions 2012.

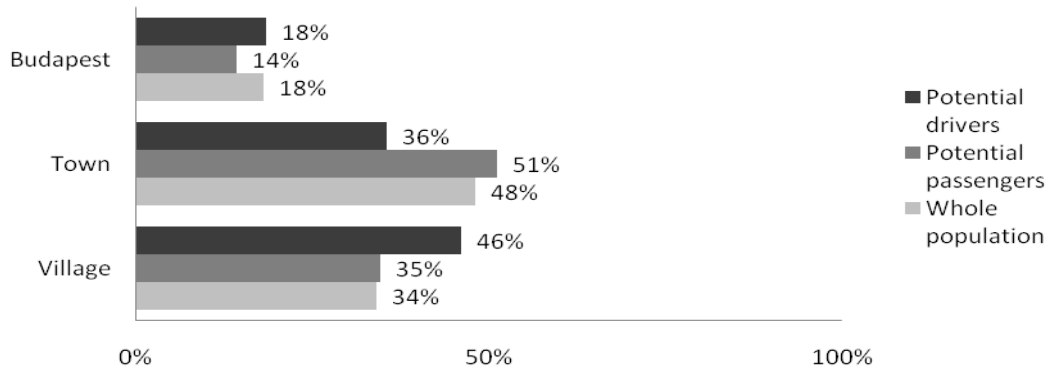
Figure 2. Number of registered users at oszkar.com scheme.



Data source: Attila Prácsér

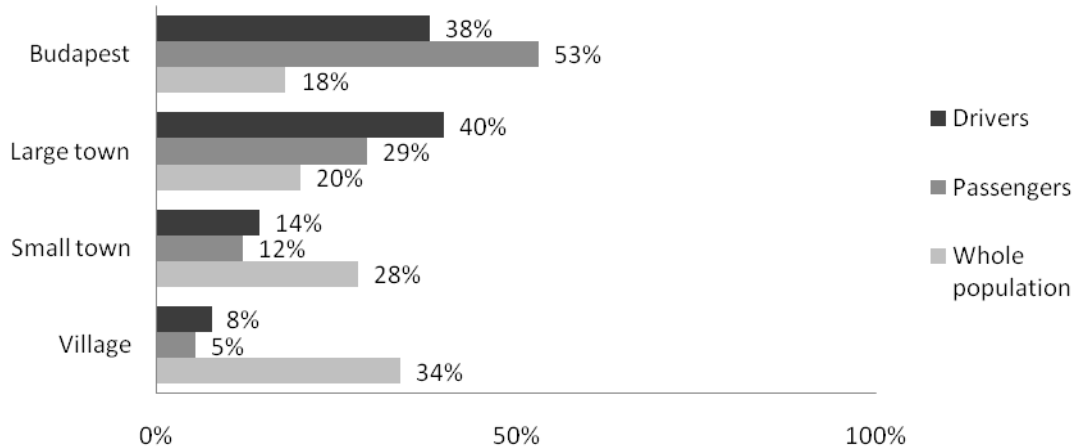
The largest reserves of potential carpoolers are among the inhabitants of villages, who, though equally or even more strongly willing to carpool (*Figure 3.*), are strongly underrepresented among the carpoolers (*Figure 4.*). It can be assumed that lower internet penetration or the lower awareness of new concepts, such as carpooling, are the main reasons for this underrepresentation.

Figure 3. Potential carpoolers' place of residence compared to the whole population of Hungary.



Data source: Policy Solutions 2012.

Figure 4. Oszkar.com users' place of residence compared to the whole population of Hungary.



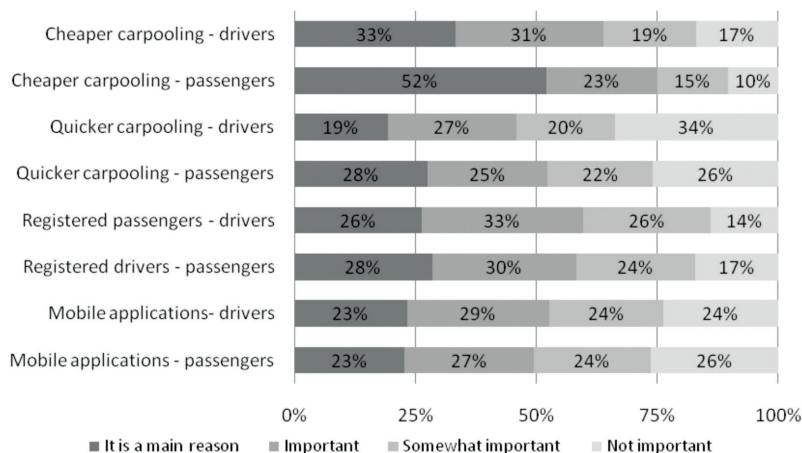
Data source: Policy Solutions 2012.

The analysis of the route database also indicates significant untapped possibilities in carpooling as it suggests that present passengers primarily *consider carpooling as a cheap and convenient alternative of public transportation for long-distance trips* between the working place (usually Budapest) and the place where carpoolers spend the weekends (hometown, parents' place, relatives' or friends' place, etc.): carpooling is mainly practiced on Friday and Sunday (62% of rides were booked on these two days in 2011), on routes where the trains tend to be overbooked or overused, i.e. between Budapest and the five largest cities of Hungary (70% of rides were booked on these routes in 2011). In addition, according to the carpooler survey people use carpooling more often for weekend trips home. On the other hand, 71% of commuting passengers and 64% of commuting drivers in the oszkar.com scheme consider carpooling as an option for commuting and half of those who travel to work would be maybe or surely engaged in carpooling. Considering that the need to get to the workplace is the most important reason why people travel, generating approximately 33% of all travel (KTI research 2012), *carpooling to work could significantly increase the prevalence of carpooling*.

Finally the positive experiences gained during carpooling forecasts a further spread of carpooling. Both the drivers and passengers gave typically positive feedbacks about carpooling, 86 and 93% of them gained primarily positive experiences during carpooling, respectively.

Based upon the above carpooling is likely to undergo a significant development. The carpooling survey aimed to investigate the different possibilities which can facilitate this process. The answers of the respondents suggest that a greater financial benefit would be the most important possibility (e.g. lower toll or congestion fee for drivers and public or workplace subsidies for passengers) – it was considered as very important by 33% of drivers and 52% of passengers. The other possibilities – quicker travel (e.g. through high-occupancy vehicles / HOV lanes reserved for carpoolers), especially in the case of passengers, officially registered users in order to reduce safety concerns, and mobile applications for arranging rides more easily – would be less important, though all of these options were still indicated as 'a main reason' by a significant minority of both groups (Figure 5.). The low interest towards smartphone applications probably reflects the low share of such phones in the Hungarian population (13% of mobile phone users owned smartphones in November 2011 [10]).

[10] Magyarországon közel egy millió okostelefon-használó van [There are almost one million smartphone users in Hungary]
http://www.gfk.hu/pressreleases/press_releases/articles/008984/index.hu.html (2012. 10. 08.)

Figure 5. The importance of different possibilities encouraging respondents to carpool more often.

The most important factor discouraging passengers from carpooling more often is the lack (or rarity) of rides offered (54% of them considered it as very important) – in this area significant improvement can be expected as the number of users (and so the number of rides offered) increase considerably. *The drivers considered the uncertainty of the start of their journey as most important barrier* (37% of them considered it as very important) while in the case of national survey the difficulties of adapting to others was mentioned most often (by 40% of those who travel to work by car) as an arguments against carpooling. It is important to note that real-time carpooling can eliminate the latter barriers to a great extent as it allows drivers to offer rides at the time of departure.

IT and other development possibilities to facilitate carpooling

The creation of a smartphone application *which allows real-time carpooling* would be the most important IT development possibility. The current smartphone application of the oszkar.com scheme allows only the offering and booking of rides. Through a real-time application both the driver and the passenger can specify the destination at the time of departure (if they specify them before the departure they shall provide the time and location of the departure, too). The smartphones detect the locations of the driver and the passengers by the help of the built-in GPS and make the best matches so that the drivers should not make too large detours and the passengers should not wait for too long. Finally the application directs the driver and passenger to

place nearby the passenger and suitable for getting into the car while indicating the time of arrival. A real-time application allows multi-loop carpooling, too, when a passenger use more than one car to get to the destination. As the smartphone continuously detect the location of the passenger, it can predict the time of arrival to the point where the ways of the driver and passenger are separated and by doing so can make another match for the further section of the passenger's trip. The application can allow the identification of users which can reduce safety concerns. In addition, the users can evaluate each other, which allows the application not to make pairs of users who do not want to travel together any more. Finally a smartphone application can allow arranging the payments online if the per km prices are set. This option is especially important on short distance when dealing with small amounts can be inconvenient. In addition, environmental considerations can be implemented by setting different prices for different periods. For example, setting lower prices for rush-hours can encourage drivers to leave their cars at home (and travel by carpooling). Moreover, different prices can be set for drivers and passengers for certain periods: e.g. when public transportation run quickly and frequently, higher prices for passengers and lower benefits for drivers can encourage the use of public transportation which is more effective and environmentally-friendly than cars. And the difference between the prices paid by passengers and the benefits received by drivers can be used to encourage drivers to carpool when public transportation is not effective, for example when there are only few passengers or public transportation occurs in a sparsely inhabited area.

There are other useful IT development possibilities beside a real-time application. For example, IT developments could also support the possibility to filter passengers or drivers based on some personal parameters (such as non-smoking driver). Or the arrangements of rides can be facilitated by a map-based approach, such as the one which collectivetraffic.net [11] use. This approach eases making the match in those cases when the routes of the driver and the passenger are only partially overlap. There is a possibility to specify intermediate stops on the route without a map-based approach but specifying the route on a map can significantly increase the number of potential matches. Adapting carpooling to Hungarian conditions could facilitate the proliferation of carpooling, too.

The separated HOV lanes, which allow a considerable faster progression for carpoolers in the highways of the U.S compared to the crowded normal lanes, seem to be less beneficial in the less crowded Hungarian highways. On the other hand, the roads in Budapest are notoriously overcrowded and there are many bus lanes which often seem underutilized. These lanes, which have been significantly extended in recent years, can be used now only by buses, taxis and vehicles bearing a distinctive mark. *Allowing carpoolers to use these bus lanes might significantly raise the level of carpooling in Budapest.*

[11]
Telekocsi
[Carpooling],
Collective!Traffic.
<http://collectivetraffic.net>
(2012. 10. 08.)

[12]
Webpage of
Esztergom
town, local
public
transport
timetable.
[http://
esztergom.hu/
wps/portal/
erkezoek/](http://esztergom.hu/wps/portal/erkezoek/)

[13]
Webpage
of Vértess
Volán Ltd.,
interurban
bus service
timetable
around
Esztergom.
[http://
vertesvolan.
hu/index](http://vertesvolan.hu/index).

As the bus lanes in Budapest have been already being checked by traffic surveillance cameras for years – and unauthorized users are fined through an automatic number plate recognition system –, checking carpoolers on bus lanes would require only enabling this system to identify the number of passengers in cars. Nevertheless, as the application of bus lanes is a sensible issue and is often seen as an unnecessary exclusion by many, allowing carpoolers to use them might induce strong negative emotions in non-carpoolers. There are also further research opportunities in the field of carpooling. For example, a special research opportunity has emerged in Esztergom and Esztergom-Kertváros recently. The local public bus transportation which included a bus line to Štúrovo [12] run by the Vértess Volán Zrt. ceased in 3rd of July, 2011 due to financial problems [13]. Since then either the interurban bus services or the free bus service of the Tesco hypermarket are used as local public transport. In both cases the lines were extended and getting on and off the bus became possible in each stop unlike during the previous practice. It could be investigated if the cessation of local bus transportation has resulted in an increase of carpooling in and around the town or in any other changes in the travel patterns. In general, additional researches could also be launched by investigating the carpooling possibility in urban areas as a urban mobility alternative.

Finally, the possibility of freight transport by carpooling could be an interesting research area, too. For the time being this is not a real option in carpooling schemes, probably due to the high risk associated with it. Nevertheless, the drivers of interurban bus services can be asked unofficially to deliver smaller packages and this “service” could be adapted to carpooling at one point in the future.

Conclusion

As flexible road transportation methods bloom in Europe, the same trend can be observed in Hungary. Public transportation companies plan to introduce new demand responsive transport lines. Some cities intend to introduce the community bike system. A carsharing company is also under development in Budapest. Carpooling fits into this scheme. It seems that there is a huge potential for carpooling in Hungary, but certain IT developments (such as real-time arrangement of trips or automatic arrangement of payments) are required to achieve that potential. In addition, opening the bus lanes for carpooling vehicles can be considered to encourage carpooling in urban areas though this policy should be applied carefully due to the drivers’ negative attitudes towards bus lanes.

Környezetimodell-építés és adaptálásának aspektusai

Összefoglalás: A tanulmány „fokozott környezeti érzékenységgű terület” konfliktusvizsgálatához hasznosítani kívánt matematikai-statisztikai modell megalkotásának lehetőségeivel foglalkozik. Az elsődleges és másodlagos kutatások alapján azonosítjuk azokat a konfliktusteret befolyásoló szereplőket és ezeket leíró jellemzőket, amelyek a matematikai-statisztikai rendszert jellemzik. Jelen tanulmányban megalapozzuk a környezeti rendszerek vizsgálatára felírható modell lehetséges alrendszereit és a változók körét, valamint kitérünk egy lehetséges elméleti modell megadására.

Kulcsszavak: Konfliktusvizsgálat, környezeti-modell, statisztikai elemzés.

Abstract: This study deals with a potential mathematical-statistical model for the use of conflict analysis of “sensitive environmental areas”. On the basis of results gained by primary and secondary surveys actors of “conflict space” and behaviours will be defined, which characterise the mathematical-statistical model. This study grounds potential sub-systems of model suitable for analysis of environmental conflict systems and gives a potential theoretical model of the variables to be involved.

Keywords: Conflict analysis, environmental model, statistical analysis.

A modell

A matematikai modellek egyszerű, ugyanakkor átfogó definiálása nem könnyű feladat [1]. Praktikusabb a modell funkciója szerint a modellt, mint információt adó rendszert értelmezni.

A modell tehát rendszer:

– célja az emberi megismerési folyamat elősegítése, újabb ismeretek szerzése;

* DUE, Informatikai Intézet
E-mail: honfivid@mail.duf.hu

** Kaposvári Egyetem,
Gazdaságtudományi Kar
E-mail: kovacs.bernadett@ke.hu

*** Kaposvári Egyetem,
Gazdaságtudományi Kar
E-mail: nagy.monika@ke.hu

[1] Koncsos L et al. (2011):
*Környezeti rendszerek
modellezése*. Budapest: BME.

[2] Stoff, V. (1973): *Modell és Filozófia*. Budapest: Kossuth.

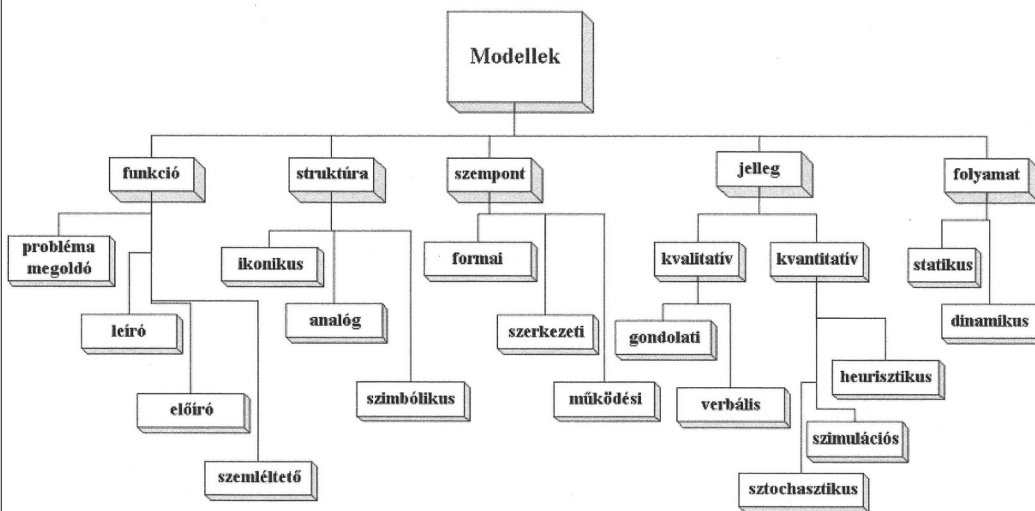
[3] Szücs Ervin (1997): *Rendszer és modell II*. Budapest: Nemzeti Tankönyvkiadó. Online: < http://www.geo.u-szeged.hu/~joe/modell/pdf/Tech_model.pdf>

– egymással kölcsönhatásban lévő részekből (a modell elemeiből) összeálló (összeállított) szerves egész;
– meghatározott „környezetével” (az ún. modellezzettel) hasonlósági összefüggésben van, nélküle nem is értelmezhető.

Stoff [2] a modellt egyenesen úgy definiálja, mint olyan „eszmeileg elképzelt vagy anyagilag realizált rendszer, amely visszatükrözve vagy reprodukálva a kutatás objektumát képes helyettesíteni” [3].

A modellek csoportosításának kiterjedt irodalma van még csupán a technikai rendszereken belül is. Az áttekintésünkhöz megfelelő alapot adhat a modellek csoportosíthatóságának megismerése. A csoportosítások lényegében a modellezett, illetve a modellt képviselő objektum jellege vagy a modellezési – pontosabban: a hasonlósági – szempont szerinti megkülönböztetésre épülnek. Bármelyik csoportosítási mód megengedett, de célszerű a viták és félreértések előtt tisztázni, hogy melyiket használtuk. Például a „fizikai modell” elnevezés egyaránt utalhat arra, hogy mit, vagy arra, hogy mivel modellezzünk, illetve arra, hogy milyen szempont szerint hasonló a modell a modellezzetthez. A célul kitűzött modell elemei predesztinálhatják a szóhajóhető modellezési eljárásokat, valamint a részterületek egymásba kapcsolásának lehetőségeit.

1. ábra. A modellek típusai.



Forrás: Szücs, 2000.

A környezeti rendszerek vizsgálatára felírható matematikai modell szimbolikusan az alábbi kifejezéssel adható meg:

$$\underline{y} = f(\underline{x}, \underline{p}) \quad (1)$$

ahol $\underline{y}$ a rendszer jellemzésére használt változók vektora, $\underline{x}$ az anyagi rendszert kívülről érő hatásokat leíró változók vektora. Az $f(\)$ azoknak a függvényeknek az együttese, amik az $\underline{x}$ és $\underline{y}$ között fennálló kapcsolatot számszerűsítik; $\underline{p}$ pedig az ezekben szereplő állandók, együtthatók, vagyis az ún. paraméterek vektora. A célul kitűzött környezeti állapotot leíró modell eredményváltozója a „fokozott környezeti érzékenységgű terület” konfliktus-állapota, amely egy területi szinten összevethető, illetve idősorosan vizsgálható mutatószám. A rendszerre hatással levő változók köre olyan alapkészlet-mutatókból, primer kutatási eredményekből és származtatott mutatószámokból kerül kiválasztásra, amelyek a környezeti konfliktusteret a vizsgálat céljainak alárendelve lefedik.

A változók kiválasztásához szükséges áttekintenünk a környezeti-modellek irodalmát.

A környezeti-modellek

Az ENSZ, az OECD, valamint az Eurostat által alkalmazott modellek főbb vonásaikban megegyeznek. [4] A 2002-ben, Johannesburgban tartott Fenntartható Fejlődés Föld-csúcson a Johannesburgi Deklaráció és a megvalósítási terv újra megerősítette az 1992-es *Riói Állásfoglalást*, aláhúзва a szegénység és a környezetvédelem kérdését. A kidolgozott indikátorkészlet 155 mutatót tartalmaz, melyek tíz témakörben; gazdasági fejlődés; szegénység és társadalmi kirekesztettség; öregedő társadalom; egészség; éghajlatváltozás és energia; termelési és fogyasztási szokások; természeti erőforrások; közlekedés; jó kormányzás; globális partnerség mutatják be a fenntartható fejlődés megvalósulásának állapotát. [5]

Az európai indikátorkészlet [6] mutatói hierarchikus rendszerben, három szinten helyezkednek el. Az első szinten lévő 12 indikátor (fő indikátorok) átfogó képet nyújtanak az egyes területeken végbemenő főbb tendenciákról.

[4] Katonáné Kovács Judit (2004): Környezeti mutatók – Agrár-környezeti mutatók. *Agrártudományi Közlemények*. Debrecen, különszám.: Pp. 248–255.

[5] Központi Statisztikai Hivatal (2007): *A fenntartható fejlődés indikátorai Magyarországon*. Budapest: KSH.

[6] Központi Statisztikai Hivatal (2005): *Magyarország környezetterhelési mutatói*. Budapest: KSH.

[7] Pomázi István (2003): *Környezeti mutatók fejlesztésének és alkalmazásának tapasztalatai a Gazdasági Együttműködési és Fejlesztési Szervezetben.* Budapest: OECD Közgazdaságtani munkacsoport. Online < www.kep.taki.iif.hu/file/Pomazi_oecd.doc >

A második szinten található mutatók – az első szint mutatóival együtt – a fő célkitűzések megvalósulását mérik. A harmadik szint mutatói (elemző mutatók) egy-egy altéma mélyebb elemzésére adnak lehetőséget.

Az OECD [7] környezeti mutatókkal kapcsolatos tevékenysége a következő területekre irányul:

- Az egyes tagországok környezeti mutatókkal kapcsolatos munkálatainak összehangolása közös megközelítések és koncepcionális keretek kialakításával.
- A tagállamok segítése a környezeti mutatók használatának továbbfejlesztésében.
- Az OECD-országokban felhalmozott tapasztalatok átadása a nem-tagállamok és más nemzetközi szervezetek számára.
- Az OECD politikai elemző és értékelő munkájának támogatása megbízható, mérhető és politikailag releváns mutatók kidolgozásával annak érdekében, hogy mérni lehessen a környezeti haladást és teljesítményt, a politikák integrációját és a hatékony nemzetközi összehasonlítást.
- Az OECD tevékenysége alapvetően a nemzeti, nemzetközi és globális szintű döntéshozatalt segítő mutatók használatára összpontosít, ugyanakkor ez a megközelítés alkalmazható szubnacionális (regionális) szinteken, ökoszisztémák vagy vízgyűjtőterületek szintjén is.

A környezeti haladás és a teljesítmény mérésére kifejlesztett környezeti mutatók alapkészlete a PSR-modell szerint osztályozhatók: a környezetterhelési mutatók, a környezetalapot mutatói és a politikai válaszok (intézkedések) mutatói.

Az ágazati és a környezeti elszámolásból származtatott mutatók a környezeti szempontból jelentős ágazati trendeket, azok pozitív vagy negatív kölcsönhatásait a környezettel, valamint a kapcsolódó gazdasági és politikai szempontokat mérik. A környezeti elszámolásból származtatott mutatók elősegítik a környezeti szempontok beépítését a gazdaságpolitikába és az erőforrás-gazdálkodási politikákba. A fő figyelem ezek kidolgozásánál a következő területekre irányul: környezeti kiadások számbavétele, a természeti erőforrások elszámolási rendje, beleértve a természeti erőforrások fenntartható használatát, az anyagáram-elszámolást, amely kapcsolódik az erőforrás-használat hatékonyságához és termelékenységéhez.

A környezetminőséget középpontba helyező indikátorkészlet mellett a környezet erőforrás-funkcióját hangsúlyozó, a természeti erőforrások mennyiségi kérdéseire összpontosító mutatókat alkalmaz.

A jelen tanulmány célját szolgáló modell esetében fontos kiemelni, hogy az alapkészlet mutatóit térségi szinteken is lehet alkalmazni a környezetgazdálkodás és teljesítmény területi dimenzióinak elemzésére, a főbb eloszlási kérdések azonosítására és azon regionális különbségek bemutatására, amelyek rejtve maradtak az országos szintű mutatók használatakor.

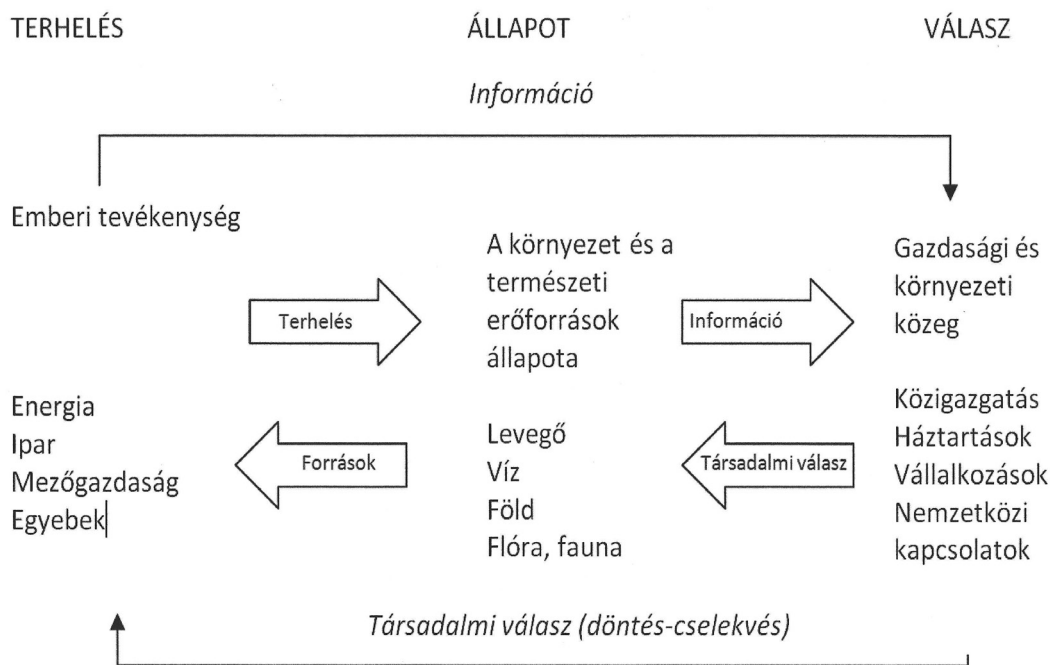
A környezeti kulcsmutatók kialakításának legfőbb célja volt, hogy az alkészlet indikátorai által lefedett információtartalmat a politikai döntéshozás számára sűrítse. A környezeti kulcsmutatók is a PSR-moddal megfelelően osztályozhatók, hangsúlyt helyezve a szennyezésekre, az erőforrások használatára és környezetterhelésekre.

A környezeti kulcsmutatók osztályozása

Szennyezési kérdések	Rendelkezésre álló mutatók	Középtávú mutatók
Éghajlatváltozás	Szén-dioxid intenzitások	ÜHG-kibocsátások indexe
Ózonréteg	OKA-felhasználás indexei	Ugyanaz
Levegőtisztaság	SOx- és NOx-kibocsátás-intenzitás	A légszennyezésnek kitett lakosság
Hulladékkeletkezés	Települési hulladékkeletkezés intenzitása	Teljes hulladékkeletkezés intenzitása, anyagáram-mutatók
Édesvizek minősége	Szennyvíztisztítási csatlakozási arányok	Víztestek terhelése
Természeti erőforrások		
Édesvízkészletek	Vízkészletek használatának intenzitása	Ugyanaz + regionális megoszlás
Erdővagyon	Erdővagyon használatának intenzitása	Ugyanaz
Halállomány	Halfogás intenzitása	Ugyanaz + teljes halállomány
Energiaforrások	Energiaintenzitás	Energiahatékonysági index
Biológiai sokféleség	Fenyegetett fajok	Faj-, élőhely-, vagy ökoszisztéma-sokféleség, kulcs-ökoszisztémák területe

Sok mutató és mutatókészlet az OECD terhelés–állapot–válasz PSR-modelljén vagy ennek egy változatán alapul. A környezet állapotát az emberi tevékenység oldaláról érkező terhelés, illetve az erőforrások állapota által generált válaszok együttes hatása befolyásolja.

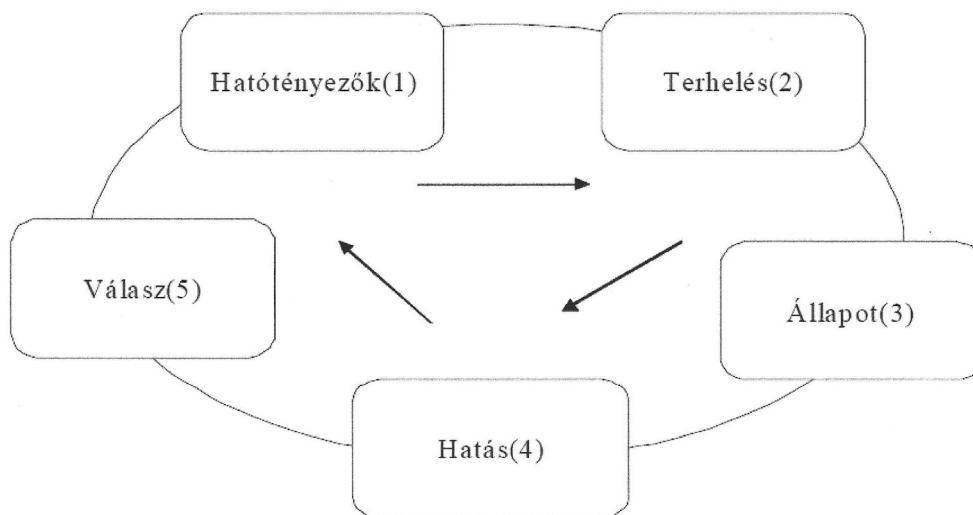
2. ábra. Terhelés–állapot–válasz (PSR) OECD-modell.



Forrás: Saját szerkesztés, OECD alapján.

Az EUROSTAT DPSIR-modellje egy kibővített hatásrendszerben beilleszti a hatótényezőket, valamint az állapot közvetlen hatásait, amelyek a gazdasági, politikai döntéshozók választ indukálják. Ekképpen szintén egy, a döntéshozás, politikai döntések intézkedések hatásának vizsgálatára alkalmas a rendszer.

3. ábra. A DPSIR-modell.



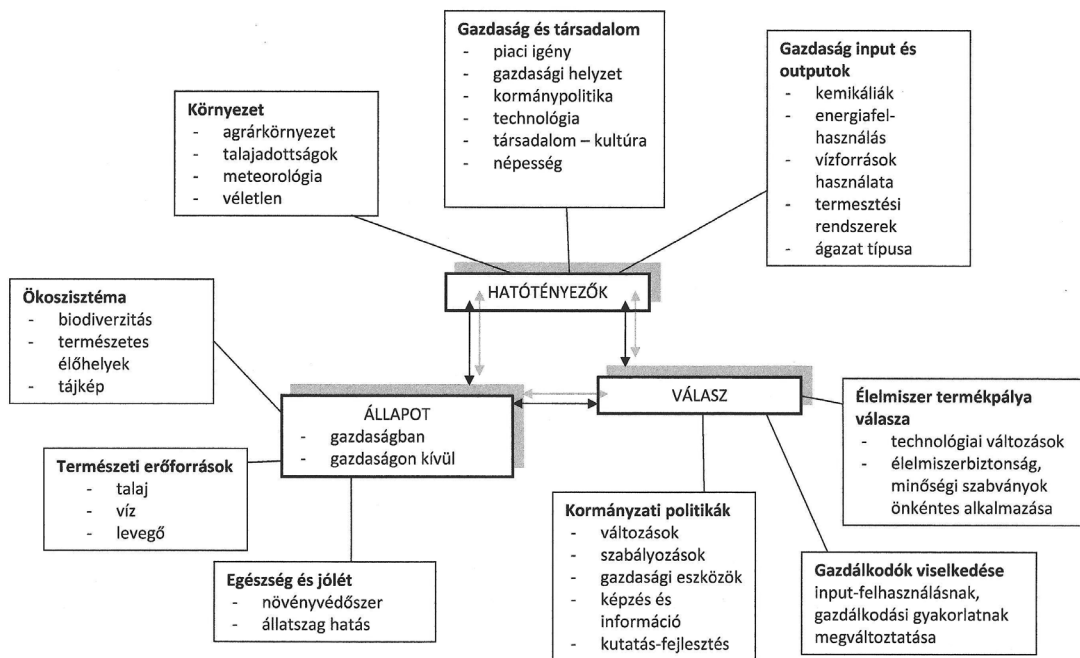
Forrás: Jesinghaus, 1999.

A fenntartható mezőgazdaságra történő adaptációban az OECD kutatói által bevezetett modell mérhető (származtatható) indikátorokkal, mutatókkal „tölti fel” a modell keretrendszerét.

A kialakítani kívánt konfliktusmodell útmutatóul szolgálhat az aktorok és tevékenységük hatásának, valamint a környezet állapota érdekében végzett tevékenységek számbavételéhez.

[8] Horváthné Kovács B.–Nagy M. Z. (2013): Az attitűd-cselekvés-környezeti állapotmodell kialakításának lehetősége – a Balaton déli vízgyűjtőjének területén. In: *Acta Scientiarum Socialium* 39. Pp. 111–120. Kaposvár: Kaposvári Egyetem.

4. ábra. A hatótényezők–állapot–válasz-modellszerkezet alkalmazása a fenntartható mezőgazdaságra.



Forrás: Saját szerkesztés, Dalal-Clayton-Bass, 2002 alapján.

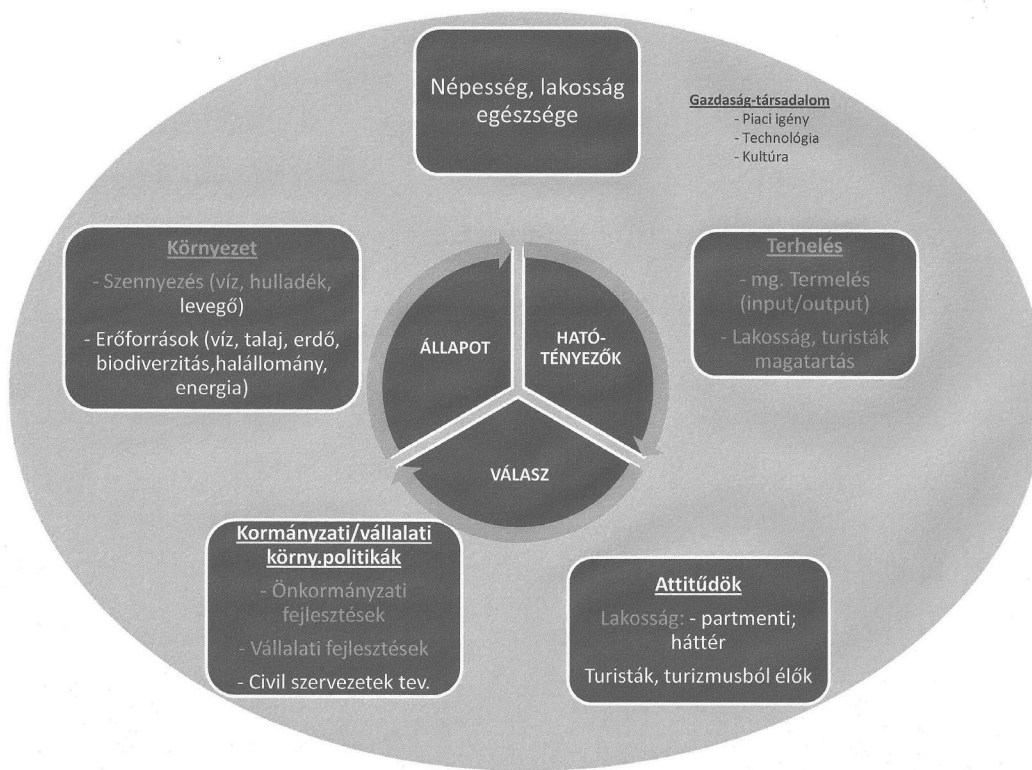
A fentiek alapján és a rész kutatás célkitűzéseinek megfelelően a „fokozott környezeti érzékenységgű” területen megállapíthatjuk a folyamatban résztvevő aktorokat, ezek kölcsönhatásának területeit, mint ún. „konfliktustereket” ill. behatárolhatjuk a specifikus területi szinten mérhető környezeti statisztikai mutatókat [8]. A környezeti folyamat szereplőinek vélekedése, viselkedése és tevékenysége alapján a kutatás későbbi fázisa elkülöníti a szereplők eltérő jellemzőkkel leírható csoportjait (a környezetvédelemhez történő hozzáállásuk és cselekvésük szerint), illetve a mérhető környezeti állapot alapján igyekszik a felelős környezeti magatartás modelljéhez hozzájárulni.

A környezeti modellrendszerben figyelembe vett alrendszerek

Az „egy fokozott környezeti érzékenységu terület” vizsgálni kívánt környezetének aktoraira olyan egymásra ható alrendszerekből álló modell építhetőségét vizsgáljuk, amelynek elemei leírják egy, a hatótényező-állapot-válasz-folyamat alkotóinak kulcselemeit.

Az alábbi modellterképen egy lehetséges összefüggésrendszert mutatunk be, amelyben már elhelyezhetők a közvetlenül vizsgálendő aktorok, ezek tevékenysége és hatásukra módosuló/beálló környezeti állapot.

5. ábra. Hatótényező-állapot-válasz-modell lehetséges modellterképe.



Forrás: Saját szerkesztés.

Természetesen a környezetiállapot-modell modellalkotóinak „viselkedését” a gazdasági-társadalmi környezet adott idejű jellemzői paraméterezik, de ezen tényezők vizsgálatától jelen tanulmányban eltekintünk. A kutatás további területe a modellváltozók és állandók között fennálló kapcsolatrendszer alapján a konkrét modell felállítása, tesztelése és érzékenységvizsgálata.

Hatás–állapot–válasz-modell matematika megfogalmazása

A modell lehetséges felépítése a modell szerkezetét valamint az egyes elemeit leíró almodellek tekintetében is komplex.

Figyelembe véve az általános modellt leíró összefüggést (1), az eredményváltozó, mint konfliktusállapot a fent leírt alrendszerekkel az alábbi összefüggésben áll:

$$\textit{konfl. állapot} = f(\textit{attitűd}) + f(\textit{cselekvés}) + f(\textit{konfliktus}) + f(\textit{körny.áll}) + \varepsilon \quad (2)$$

A kutatás további feladata az alrendszereket leíró változók megadása után, azoknak a modell számára felhasználható inputváltozókká történő manipulációja, majd a statisztikai modell megalkotása, a változók hozzájárulásának értékelése és érzékenységvizsgálatok elvégzése.

A körző, a több százéves számolóeszköz

Összefoglalás: A számítástechnika történetében több olyan eszköz is található, melyet első hallásra nem is úgy azonosítunk, mint számoló eszközt. Ilyen berendezés a körző is. Több válfajával hihetetlenül sok művelet végezhető. Talán az a legmeglepőbb, hogy ezeknek a műveleteknek csak kisebb része geometriai számítás, nagyobb fele inkább az aritmetikához köthető. A mai használt rajzeszköz az ókortól kezdve mérésre, számolásra is alkalmas volt. A sok módosítást megélt eszközt ma is használják, bár számolási, számítási eljárásokra kisebb mértékben.

Kulcsszavak: Körző, aránykörző, Alder-hárfa, osztóköörző, Galilei-körző, informatika történet.

Abstract: There are several instruments in the history of computational studies that we do not immediately identify as an arithmetic tool. The compass is such an instrument. With numerous sorts of compasses we may execute a great amount of operations. Surprisingly, the majority of these operations are arithmetical calculations, not geometric constructions. The compasses of today have been suitable for calculations and measurements since the ancient times. This instrument has been modified several times, and is still used today.

Keywords: compass, sector, Alder-harp, dividers, Galileo's compass, history of computational studies,

**DUF Informatikai Intézet*
E-mail: kiru@mail.duf.hu

[1] *Dinglers polytechnisches journal* 69. Stuttgart, 1938 .

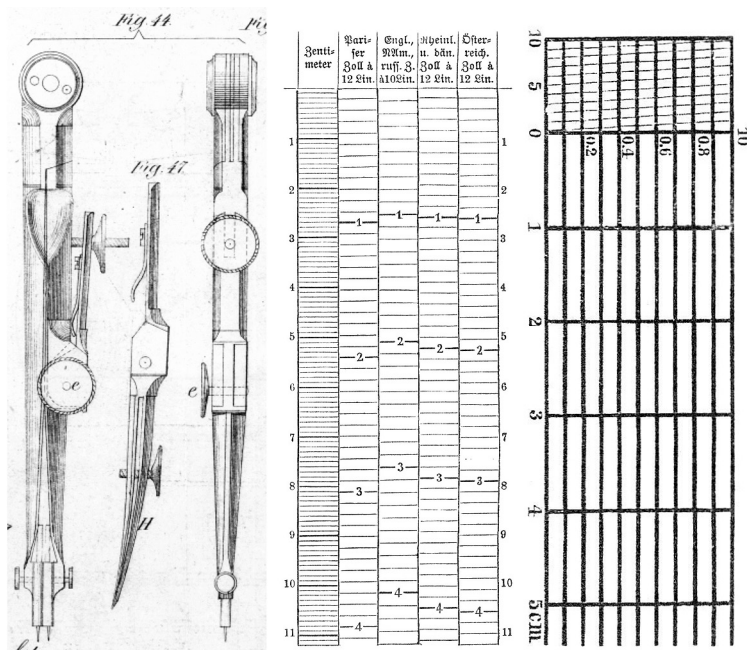
Bevezetés

A körzőt, mint rajzolóeszköz használatát minden kisiskolás gyermek tanulja. Az ókor óta használatban lévő rajzeszközt, számolóeszközként azonban kevesen ismerik. Cikkemben azoknak a műveleteknek a sokaságára világítok rá, amelyet ezzel az eszközzel el lehet végezni. A körzők több speciális fajtájának segítségével építették fel a ma is ismert történelmi épületeink nagyrészét, navigáltak hajósaink, terveztek mérnökeink. A körzők felhasználhatóságának sokféleségét csak a számítógépek számolási kapacitása múlhatta felül. A számítástechnika történetében több szoftver a vonalzó és a körző „kiváltására” készült.

OSZTÓKÖRZŐK

A mindenki által ismert kétszárú eszköz, két végén hegyes, vagy egyiket valamilyen rajzolásra alkalmas eszköz (ón, grafithegy, tuskihúzó, stb.) alkotja. Ezzel az eszközzel adott szakaszt, egy másik (rövidebb) szakasszal oszthatunk fel, de használhatjuk arra is, hogy járművünk egységnyi idő alatt megtett útját ismerve, kiszámolhatjuk (lemérhetjük mennyi idő alatt tudunk egy útvonalat megtenni). Az osztókörzőket földek kimérésére is gyakran használták, a száraikat rögzítették így az átforgatáskor mindig az adott egységgel lépve haladhattak. (Maximum 2 m-re beállítható földmérő körzőt ma is lehet kapni.) Az osztókörzőkkel a térképek, kicsinyített rajzok, eredeti méreteit is meg tudjuk állapítani, a térképre rajzra helyezett ún. vonalas léptékkel. Nagyobb pontosságot tudunk elérni a méretek átváltásánál, ha ún. átlós léptéket használunk, melyet gyakran külön fa, réz vagy vas vonalzókra rajzoltak. Ebben az esetben $\pm 0,1$ – $\pm 0,3$ mm középhibájú adat-levételt ill. -felrakást tudunk elérni. A nagyobb pontosság végett használtak ún. hajszálkörzőket, vagy ahogy Bácsatyai László nevezi: ütközőszálas körzőket, melyeknél egy beállítótárcsa segítségével a milliméter tört részét is állítani lehetett a körző nyílásán. A hajszálkörzők különféle variációkban készültek, az egyik ilyenről a *Dinglers Polytechnisches Journal*, 69. kötete is beszámol képet is közölve róla. [1]

1. ábra. Hajszálkörző, vonalas és átlós léptékek.



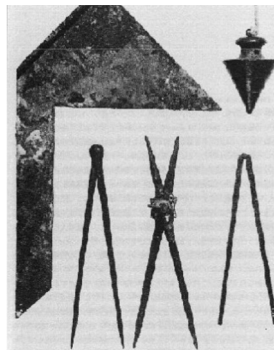
Az osztókörzők egyszerű összeadásra és kivonásra is képesek, egy nem egyenes vonalban haladó útvonal esetén pl.: célszerű a rövidebb szakaszok összegzése a körző továbbnyításának segítségével. A görbe útvonalak szakaszokra oszthatók fel, így végezhető el ezek hosszának összegzésével a teljes görbe hosszának közelítő megállapítása.

REDUKCIÓS KÖRZŐK

Az ókorban is már használt számoló- és szerkesztőeszköz, mely az ókori mérnök készletéhez tartozott. A többségében rögzített csuklóponttal rendelkező eszköz a hármasszabály ismeretlen tagjának kiszámítására, valamint geometriai nagyításra és kicsinyítésre, átváltásra is szolgáló eszköz a római birodalom teljes területén fellelhető volt. A legtöbb ismert példányt Anglia mai területén találták. A több anyagból és több méretben is készült eszközt más mérnöki berendezésekkel együtt ásták ki a földből.

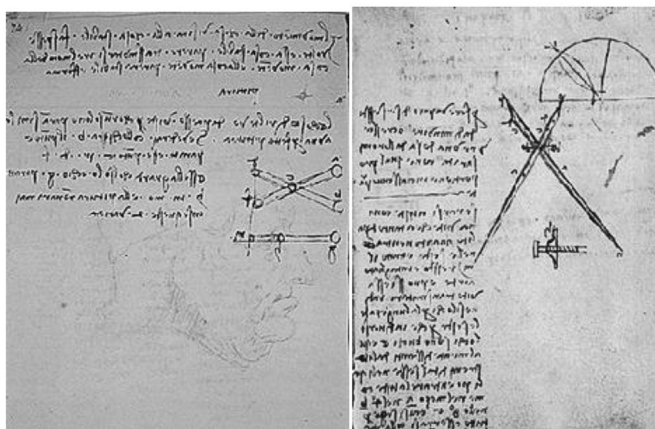
[2] Istituto e Museo di Storia della Scienza: Galileo's compass, 2004.
Online: http://brunelleschi.imss.fi.it/esplora/compasso/dswmedia/risorse/testi_completi_eng.pdf

2. ábra. Mérnöki készlet a Hadriánusz-fal tövéből.



A középkorban is sűrűn használt redukciós körzők használatáról, az eszköz formájáról Leonardo Da Vinci is írt. Az egyszerű aránykörzőkről Fabrizio Mordente-től (1532–?1608) 1567 körül hallhatunk újra, azonban „Mordente” nevű készülékének más a felépítése, mint Leonardóénak és a később használtakénak. [2] A 15. századtól ismert aránykörzők szárai már hornyoltak, így a csuklópont beállítására alkalmas csavar helyzete változtatható, ezzel az arányok is változtathatóak lesznek.

3. ábra. Leonardo Da Vinci jegyzetei az áttételi- (arány-) körzőről.



Forrás: Istituto e Museo di Storia Della Scienza gyűjteményéből <http://www.imss.fi.it/2005/>

GALILEI GEOMETRIAI ÉS HADI KÖRZŐJE

Galileo Galilei (1564–1642) 1597-ben készített egy olyan kétszárú eszközt, mely főleg a földmérők, valamint a tüzérek, hadrendező, hadmérnökök elvégzendő számításait egyszerűsítették le. 1606-ban „Le operazioni del compasso geometrico et militare” címmel használati utasítást is mellékelte ehhez az eszközhöz. [2] *A geometrikus és hadi körző használata* című művét Pádovában adatta ki.

Műveletek:

- Összeadás, kivonás, osztás, szorzás.
- Szakasz felosztása egyenlő részekre.
- Egy szakasz újrafelosztása, amennyiben a szakasz n egyenlő részre van osztva x hosszúságú szakaszokra, m egyenlő részre osztása az így keletkező y hosszúságú szakasz hosszának meghatározása.
- Tetszőleges egyenes vonalakkal határolt alakzatot nagyíthatunk vagy kicsinyíthetünk kívánt mértékben.
- Tetszőleges sokszög szerkesztése, oldalhosszúságának meghatározása a befoglaló kör sugarának függvényében.
- Pénz váltásra, átváltásra használhatjuk.
- Négyzetgyököt, köbgyököt vonhatunk.
- Aránypár ismeretlen negyedik tagját határozhatjuk meg, (ez az ún. hármasszabály használata), tetszőleges hosszúságú vonalak közötti arányszámot állapíthatunk meg.
- Kamatos kamatszámításra használhatjuk.
- Testek „összeadása”: kockák és téglatestek összegzése oly módon, hogy két azonos típusú idomból a kettő összegét adó idom oldalhosszúságait meghatározzuk.
- Magasság meghatározása dőlésszög és alap segítségével.
- Tetszőleges sokszög területének meghatározása.
- Ágyúgolyó átmérője és anyaga alapján a lőpor mennyiségének meghatározása.
- Asztronómiai számítások a navigációhoz. (Pozíció-meghatározás a csillagok állása alapján.)
- Magasságszámítás távolság és szög ismeretében.
- Két egyenmű síkbeli alakzat egymáshoz viszonyított aránya határozható meg.
- Síkbeli alakzatok *összeadásával* újabb azonos nemű alakzat állítható elő.
- Két egyenmű síkbeli alakzat különbségeként állíthatunk elő egy újabb alakzatot. (Geometrikus kivonás.)
- Parallelepipedont kockává transzformálhatunk.

[2] Istituto e Museo di Storia della Scienza: Galileo's compass, 2004. Online: http://brunelleschi.imss.fi.it/esplora/compasso/dswmedia/risorse/testi_completi_eng.pdf

[2] Istituto e Museo di Storia della Scienza: Galileo's compass, 2004.

Online: http://brunelleschi.imss.fi.it/esplora/compasso/dswmedia/risorse/testi_completi_eng.pdf

[3] Galileo Galilei (1606): Operations of the Geometric and Military Compass.

Padua.

Online: <http://brunelleschi.imss.fi.it/esplora/compasso/dswmedia/risorse/eleoperazioni.pdf>

[4] Brachner, A. (Hrsg.) (1983): *G. F. Brander (1713–1783) wissenschaftliche*. München: Instrumente aus seiner Werkstatt.

[5] Hanftmann, B. (1905): Ein Meßinstrument Balthasar Neumanns. *Deutsche Bauhütte* 32.

– Fémek térfogatának kiszámításában segít. Azonos súlyú, de más-más fajtájú fémek térfogatát.

– Számíthatjuk ki. [2, 3]

Az aránykörző igazán sokféle továbbfejlesztést élt meg. Anyagában, skáláiban való változtatások kiterjesztették az elvégezhető műveletek számát: pl.: Georg Friedrich Brander 1768-ban fából és papírból készült perspektív aránykörzője (Kremsmünsteri Csillagvizsgáló múzeumának tulajdonában) [4] a szögfüggvények számítására is alkalmas volt. Balthasar Neumann (1687–1753) 1713-ban készített építészeti aránykörzője, a különböző típusú (dór, jón, korintoszi, toszkán) oszlopok magasság átmérő díszítési arányainak meghatározására volt alkalmas. Mindezek mellett ezzel a körzővel is lehetett összeadást, kivonást, osztást, szorzást, gyökvonást, szakasz felosztását, ugyanolyan területi szabványos síkidomok átszerkesztését is elvégezni. [5]

Van a ki a száraz hosszúságát növelte meg pl.: J. Rabone Birminghamben (1900 környékén), így téve lehetővé az értékhatárok kiterjesztését.

A legnagyobb aránykörző gyűjteménnyel a Firenzei Történeti és Tudományos Múzeum rendelkezik, melyben nemcsak a feltalálónak Galileo Galilei eszközének prototípusát és másolatait találjuk, hanem a későbbiekben az ötletet továbbfejlesztő, vagy éppen egyszerűbb aránykörzőket készítő feltalálók iparosok műveit is. A főleg bronzból, de később vasból is készült eszközöket a 20. század elejéig használták a hadmérnökök, földmérők. Gyakran egészítették ki a körzőket talpakkal függőónnal és szögmérővel, mely mérések, függőlegesség (merőlegesség) ellenőrzésére is alkalmassá tette az eszközt vagy éppen iránytűvel (tájolóval) melynek segítségével a terepen még jobban használható volt az aránykörző.

Valószínűleg Edmund Gunther (1581–1626) logarlécének előde is egy aránykörző volt, hiszen ő maga is készített ilyen eszközt Sector néven. Eszköze annyira elterjedt lett, hogy sok helyen ma is Sector névvel illetik az aránykörzőket.

KALIPER-KÖRZŐ

Több tucat fajtája ismert, némelyik csak a külső vagy a belső átmérő meghatározására képes, de olyan is akad, mely a belső és/vagy külső átmérő alapján területet vagy kerületet, esetleg gömb esetén térfogatot számol. Különleges Kaliper-körzők az ellipszoidok térfogatának kiszámítására is alkalmasak voltak. A kaliperkörzők ma is általánosan használt eszközök. Az orvostudományban több tucat fajtájukat használják, hiszen műtét esetén a pontos, gyors mérés, esetleg számítás fontos lehet, ugyanakkor egyszerűbb egy kézbe vehető eszközzel ezt megoldani, mint az adott szervet esetleg digitalizálni és így megadni az eredményeket. Az ékszerészek szintén használják drágakövek értékének felbecslésénél.

4. ábra. Kaliper-körző.



ALDER-HÁRFA ÉS A MÉRŐKÖRZŐ

A földmérők, bányamérnökök terület-meghatározó eszközei között az 1850-es évektől az ún sávmódszerrel végzett terület-meghatározás is szerepelt. A térképen már megrajzolt objektumot (földterületet) egyenlő távolságra lévő vonalakkal felosztották. A körvonal mentén a vonalakra merőlegeseket húzva téglalapokat alkottak, amelyek területét már könnyedén ki lehetett számítani:

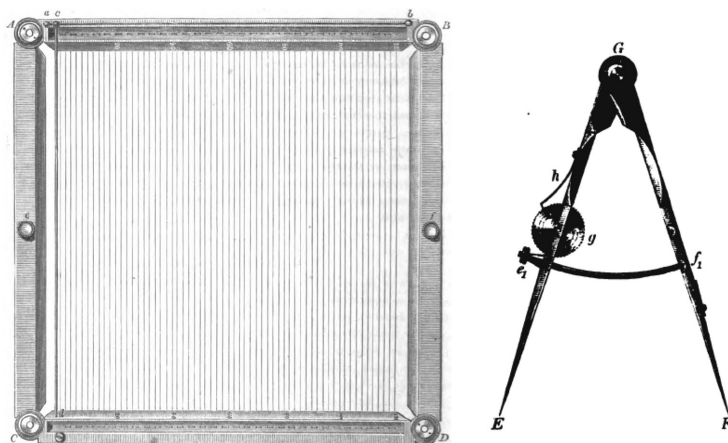
A számítást úgy tették könnyebbé, hogy egy ún.: Alder-hárfa-t használtak (5. ábra). Ez a fahérből és lószőr húrokból álló kis készülék megkönnyítette a vonalak felrajzolását. A téglalapok hosszabbik oldalának lemérésére osztókörzőket használtak. Ami még gyorsabbá tette a munkát az ún. mérőkörző (Hunderterzirkel) alkalmazása volt. A körzőt az első téglalap hosszára nyitották, majd a következő téglalapot úgy adták hozzá, hogy a kinyitott körzőt a következő téglalapon kívül, de a számlálós szárával a téglalap kezdőpontjához illesztették, így a körzőt addig tágtították, míg az a beállított határértéket el nem érte, ekkor a rajta lévő számláló kör egyet léptetett.

[6] Bácsatyai László (2002): *Geodézia II.* Sopron: Nyugat-Magyarországi Egyetem.

[7] Georg Christian Conrad Hunäus, G. Ch. C. (1864): *Die Geometrischen Instrumente der gesamten praktische. Geometrie deren Theorie Beschreibung und Gebraucht.* Hannover.

A körzőt újra összezsukva lehetett folytatni a mérést (számolást) a végén a számláló körről leolvasva és a maradék szártávolságot (nyílást) lemérve lehetett az Alder-hárfa rögzített húr hosszúságával a téglalapok területét kiszámítani. [6] A módszert Eduard Oldendorp fejlesztette tovább aki az Oldendorp-planimétert fejlesztett ki, ami tulajdonképpen egy csuklókkal ellátott Alder-hárfa volt, mellyel már nemcsak téglalapok, hanem trapézok közé lehetett a mérendő területet szorítani. A számoláshoz ő is mérőkörzőt használt. [7]

5. ábra. Alder-hárfa és mérőkörző.



TENGERÉSZETI KÖRZÖK

Gyakran egyszerű osztókörzőket vagy kaliperkörzőket kiegészítettek, többskálás szögmérőkkel, így tengeri utak hosszát a térképek alapján könnyebben kiszámolhatták. A szárazra vésott skálák segítségével bonyolultabb számításokat is el lehetett végezni (lásd: aránykörzők). Valószínűleg a tengerészeti szextáns is a tengerészeti körzőkből alakult ki.

GÖMBI KÖRZŐK

Feladatuk gömb vagy hengeres, kúpos testek mérése számítása. Különleges alakjuknak köszönhetően ezeknek az eszközöknek a használatával egy golyó (ágyúgolyó) súlyát anyagának ismeretében egy egyszerű leolvasással megállapíthatjuk. Furat-átmérőt mérhetünk, vagy a furat aljáig lenyomva a körző „hegyét” leolvasható a furat térfogata. Építészek oszlopok mérésére, súlyának, térfogatának számítására, hadmérnökök fegyverek űrtartalmának, súlyának, golyók súlyának megállapítására is használták.

FELMÉRŐ KÖRZŐK

Hasonlóan az aránykörzőkhöz, kétszárú csuklós szerkezetek. A szárazon azonban a tájolást, mérést, pontkitűzést segítő skálák és eszközök (iránytű, függőőn, szögmérő, stb.) kerültek elhelyezésre. Gyakran számoló skálák is megtalálhatóak az eszközökön. (Lásd: Aránykörzők) A felmérő körzők egyik Magyarországon is elterjedt verziója az ún. Kirchner-féle térképkörző, mely két szára és az azokon látható széles skálák segítségével térképek méretarányos kicsinyítését, nagyítását, vagy több különböző méretarányú térkép, műszaki rajz migrálását, kompozícióját teszi lehetővé.

Zárszó

A fenti eszközöket vonalzókkal, karctűkkel, rajzszénnel együtt gyakran mérnöki csomagokban kínálták már a 17. században. Sejtethető, bár kevésbé bizonyítható, hogy ilyen mérnöki készleteket (szetteket) az ókorban is készíthettek. A sejtést erősíti az, hogy gyakran egy helyen találtak több mérőeszközt. (Hadriánus-fal, Aquincum, Alba Regia, stb.) A mérőeszközöket számos alkalommal egymásba is építették, így növelték hordozhatóságukat és az elvégzett funkciók (műveletek) számát. Magyarországon szinte mindenféle az imént leírt körzővel lehet elszórtan –múzeumokban vagy magángyűjteményekben– találkozni.

A fentiekben jópár többkarú körzóról, pantográfíró, rajzolásra, geometriai műveletekre szolgáló körzóról nem ejtettünk szót, de talán ez a rövid cikk is rávilágított arra, hogy milyen széles mérési, rajzadási, számítási lehetőséget nyújtottak a körzők. A digitális korszakban is megtalálták a helyüket: a digitális osztókörzők nemcsak a leolvasást teszik pontosabbá, hanem a felvett értékeket saját memóriájukban tárolják, vagy ezekkel egyszerűbb műveleteket végeznek. A digitális körzők adatai számítógépre vihetők, így ezekkel az eszközökkel folytatott mérnöki munka megfelel a 21. század követelményeinek.

Kérdőíves felmérések megbízhatóságának növelése

* Dunaújvárosi Főiskola
E-mail: buzanekispiroska@yahoo.com

** Dunaújvárosi Főiskola
E-mail: panny07@freemail.hu

*** Dunaújvárosi Főiskola
E-mail: david.kovacs91@gmail.com

Összefoglalás: Bizonyos kérdőíves felmérések során a feltett kérdésekre őszintén válaszolni vonakodnak az érintettek. Félrevezető válaszokból helyes következtetés aligha vonható le. A munka az őszinte válaszok megszerzésének egy lehetséges, valószínűségelméleten alapuló módszerét mutatja be. Az elméleti eredmények gyakorlati alkalmazására a Dunaújvárosi Főiskola diákjai körében végeztünk felmérést. Ennek eredményét és a levonható következtetéseket is ismertetjük.

Kulcsszavak: Kérdőív, kérdéspár, feltételes valószínűség, valószínűség.

Abstract: People hardly answer quite openly some types of questionnaires. However, insincerity leads up to false inferences. The aim of the paper is to introduce a method based on probability theory for getting of the straightforward answers. In order to apply the theoretical results in the practice, a group of students at College of Dunaújváros were involved into filling in of questionnaires. The paper contains the assessment of the survey and also the suitable inferences.

Keywords: Questionnaire, pair of questions, conditional probability, probability.

Bevezetés

A Dunaújvárosi Főiskolán a TÁMOP 4.2.2.B tehetséggondozásra irányuló pályázat keretében lehetőség nyílt a szakkollégiumi keretek között már korábban is folyó tehetséggondozás további népszerűsítésére és kiterjesztésére. A pályázat nyújtotta lehetőségek figyelemfelkeltőnek bizonyultak és elősegítették a hallgatók bekapcsolódását különféle kutatómunkákba. A Dunaújvárosi Főiskolán a matematikai tárgyak többségének oktatása alapozó tárgyként a főiskolai tanulmányok kezdetén történik.

Ebben az időszakban a szaktárgyak esetén szintén az alapozás folyik, így a matematika tananyag kevésbé kapcsolható össze a szaktárgyakkal. A tanulmányokban előre haladván azonban szükséges több-kevesebb matematikai háttér a szaktárgyak sikeres elsajátításához. Ezért a matematika oktatása során törekedni kell arra, hogy minél maradandóbb ismereteket szerezzenek a hallgatók, minél jobban megragadja a figyelmüket a tananyag.

A Kerpely Antal Computers (KAC) Szakkollégiumhoz kapcsolódó tehetséggondozás keretében többek között alkalmazott matematika irányú kutatásba is bekapcsolódtak hallgatók. Többen foglalkoztak azzal, hogyan lehetne a matematika oktatását minél emlékezetesebbé tenni és ezzel egyidejűleg az ismeretek maradandóságát is növelni. Ezt a cél szolgálta az a szakkollégiumi munka is, amelyet az alábbiakban bemutatunk. Főként valószínűségelméleti és matematikai statisztikai háttérre alapozott kutatást végeztek tehetséggondozásba bevont hallgatók.

Kényes kérdések

Kaphatunk-e őszinte választ „kényes” kérdésekre? Bizonyára találkoztunk már olyan kérdésekkel, amelyek esetén jogosan feltételeztük, hogy őszinte választ nem kapunk rájuk. Ilyen kérdések különböző felmérésekben is előfordulhatnak. Például, ha egy közvélemény-kutatásban arra vagyunk kíváncsiak, követnek-e el a gépjárművezetők kisebb-nagyobb szabálysértéseket – anélkül, hogy ezt a rendőr észrevenné –, erre a kérdésre a megkérdezettek elég őszintén válaszolnak még a kérdezőbiztos jelenlétében is. Ha azt kérdeznénk, hogy „Fogyaszt-e Ön alkoholt a munkahelyén?” vagy „Elmegy-e Ön betegállományba, amikor az orvos ezt javasolja?” vagy „Bevallja-e Ön minden jövedelmét?”, vajon nem kételkednénk a válaszok őszinteségében?

A kérdőívek összeállítása

Egy példán keresztül mutatjuk be kényes kérdés esetén a kérdőív összeállítását az őszinte válasz megszerzésére. Például legyen az a feladatunk, hogy egy felméréssel derítsünk ki, hogy az orvos javaslatára a betegállományt milyen mértékben veszik igénybe a munkaviszonyban állók. Ez esetben a megkérdezettekől az „Elmegy-e Ön betegállományba, amikor az orvos ezt javasolja?” kérdésre szeretnénk őszinte választ kapni. Ha azt gyanítanánk, hogy erre a kérdésre sokan nem válaszolnak őszintén, akkor jobb, ha ezt a kérdést így fel sem tesszük. Hiszen félrevezető válaszokból helyes következtetések aligha vonhatók le. A kérdőíves felmérés kiértékelésével kapott eredmények nem tekinthetők pontosnak, ezek a valóságos helyzetet közelítő eredmények.

Célunk a jó közelítő eredmény megszerzése. Ennek érdekében használhatjuk például az itt ismertetésre kerülő módszert, amely valószínűségelméleten alapul. A módszer lényege az alábbi:

A kényes kérdést párba állítjuk egy semleges kérdéssel. A semleges kérdést a felmérésbe bevont személyek figyelembevételével választjuk. Ügyelünk arra, hogy ez egyszerű kérdés legyen és „igen”-nel vagy „nem”-mel könnyen megválaszolható legyen. Továbbá az „igen” és a „nem” válasz egyaránt semleges legyen a válaszadó számára, egyikhez se fűződjön valamilyen elvárás, pozitív vagy negatív kicsengés. Semleges kérdések lehetnek például: „Szereti-e Ön a reklámokat?”, „Szereti-e Ön a spenótot?”, „Rendszeres hallgatója-e Ön a szombat délelőtti rádióműsoroknak?”, „Sportol-e Ön rendszeresen?”

A kényes kérdésből és a semleges kérdésből egy párt alkotunk, például:

- A. Elmegy-e Ön betegállományba, amikor az orvos ezt javasolja?
- B. Szereti-e Ön a spenótot?
- C. Válasz Igen ☐ Nem ☐

A felmérésbe bevont személynek azt a feladatot adjuk, hogy tetszőlegesen válassza ki az egyik kérdést, de ne mondja meg, ne tüntesse fel sehol sem, melyik az, és válaszolja meg csak azt a kérdést, amit kiválasztott.

Tegyük fel, hogy a fenti kérdéspárra „Nem” válasz érkezett. Ilyenkor nem tudjuk, hogy a megkérdezett azért válaszolt „Nem”-mel mert nem megy el betegállományba, ha beteg, vagy azért mert a spenótot nem szereti.

Azt szeretnénk elérni, hogy a megkérdezettek még csak ne is sejtsek, hogy a felmérésben mire vagyunk igazából kíváncsiak. Így még kevésbé tudnak félrevezető válaszokat adni. Ezért további kérdéspárokat készítünk. Ez utóbbiak a felmérés szempontjából teljesen lényegtelenek, a kiértékelésükkel sem foglalkozunk. Ezeknek csupán annyi szerepük van, hogy feltűnésmentessé tegyék a kényes kérdésünket tartalmazó kérdéspárt.

A teljes kérdőívet kérdéspárokból állítjuk össze. A felesleges kérdéspárok közé beágyazzuk azt, amelyikre valóban kíváncsiak vagyunk. Ügyelünk arra, hogy a kényes kérdés a kérdőívek felénél az „A” kérdés, felénél a „B” kérdés legyen. Az egyes kérdőíveken a kérdéspárok sorrendjét is változtatjuk. Példánkban a betegállományra vonatkozó kérdést a kérdőívek csak egy részénél tesszük a 3. helyre. Elkészíthetjük például az 1. ábrán látható kérdőívet.

1. ábra. Kérdéspáros kérdőív.

Általános felmérés

Kérem, mindegyik pontban a két kérdés közül válassza ki tetszőlegesen csak az egyik kérdést és válaszolja meg a kiválasztott kérdést. Kérem, ne tüntesse fel, melyik kérdésre ad választ!

1. A. Sportol-e Ön rendszeresen?
B. Ment-e Ön át piros jelzésnél útkereszteződésen a múlt héten?
Válasz Igen ☐ Nem ☐
2. A. Előfordul-e, hogy Ön alkoholt fogyaszt munkahelyén?
B. Előfizetője-e Ön valamelyik országos napilapnak?
Válasz Igen ☐ Nem ☐
3. A. Elmegy-e Ön betegállományba, amikor az orvos ezt javasolja?
B. Szereti-e Ön a spenótot?
Válasz Igen ☐ Nem ☐
4. A. Hallgatja-e Ön rendszeresen a rádió reggeli zenés műsorát?
B. Elégedett-e Ön mostanában a tejtermékek minőségével?
Válasz Igen ☐ Nem ☐
- .
.
.
- n. A. Mozgó járműre felszáll-e Ön?
B. Dohányzik-e Ön?
Válasz Igen ☐ Nem ☐

A kérdéspáros kérdőívek kiértékeléséhez még egy felmérést kell végeznünk. E felmérésnek az a lényege, hogy az eredeti kényes kérdésünk semleges kérdéspárjára kapjunk választ. Esetünkben a „Szereti-e Ön a spenótot?” kérdésre keressük a választ. Feltételezhetjük, hogy erre a kérdésre a megkérdezettek őszintén válaszolnak.

A felmérésbe bevont személyek, valamint a válaszok kiértékelése

A HOMOGÉN CSOPORTOK KÉRDŐÍVEI

A kényes kérdést tartalmazó felmérést az alábbiak szerint végezzük el:

A felmérésbe bevont személyeket úgy választjuk ki, hogy belőlük két, jó közelítéssel azonos összetételűnek, egyformának tekinthető csoportot képezhessünk. A két csoport legyen I. és II.

Az I. csoportban elvégezzük a felmérést a kérdéspárokat tartalmazó kérdőívekkel. Az értékelésnél csak azt a kérdést tekintjük, amelyre kíváncsiak voltunk, a példánkban ez a betegállományra vonatkozó kérdés. Tegyük fel például, hogy az alábbi eredményt kaptuk: 56% „Igen” válasz, 44% „Nem” válasz.

A II. csoportban a semleges kérdésre vonatkozóan végzünk felmérést. Valójában az a célunk, hogy a kérdéspár nem kényes kérdésre megkapjuk a választ. Peldánkban azt vizsgáljuk, hogy a megkérdezettek hány százaléka szereti a spenótot. Ezt a felmérést akár úgy is elvégezhetjük, hogy a csoportban csak ezt az egy kérdést tesszük fel, a válasz az „Igen”, „Nem” lehet. A II. csoportban is szerepelhet egy kérdőíven több kérdés is, amelyek mindegyikére „Igen”-nel vagy „Nem”-mel kell válaszolni. A kiértékelésnél csak az egyetlen, számunkra érdekes kérdéssel foglalkozunk. Tegyük fel, hogy a II. csoportban a megkérdezettek 37%-a szereti a spenótot, 63% nem szereti.

Feltehetjük– a kérdőívek szerkezete miatt– hogy az I. csoportba tartozó megkérdezettek közül fele-fele arányban válaszolták meg a kényes kérdést és a semleges kérdést. Azaz feltehetjük, hogy a megkérdezetteknek körülbelül a fele a betegállományra vonatkozó kérdésre válaszolt, fele a spenóra vonatkozóra. Azt is feltehetjük– mivel egyformának tekinthető csoportokat képeztünk–, hogy az I. és a II. csoportba tartozók körülbelül egyforma mértékben szeretik a spenótot.

A FELMÉRÉS KIÉRTÉKELÉSE

Vezessük be az alábbi jelöléseket a rövidebb írásmód kedvéért. Jelölje $P(A)$ a kérdéspáros kérdőíven egy kiszemelt kérdésre az igen válaszok százalékát a kérdésre adott összes válaszhoz viszonyítva. Ez a példánkban 56%, azaz 0,56. Ez az „Igen” válaszok valószínűsége a kérdéspáros kérdőíven.

Jelölje $P(A|B_1)$ annak a valószínűségét, hogy a megkérdezett igennel válaszolt a kiszemelt kérdésre, feltéve, hogy a betegállományra vonatkozó kérdést válaszolta meg.

Felmérésünk célja ennek a számnak a kiderítése. Ilyen valószínűséggel veszik igénybe a betegállományt a beteg dolgozók. Ahány század ez a szám, a munkavállalók annyi százaléka megy betegállományba, ha beteg.

Jelölje $P(A|B_2)$ annak a valószínűségét, hogy a megkérdezett igennel válaszolt a kiszemelt kérdésre, feltéve, hogy a spenótra vonatkozó kérdésre válaszolt. Ezt a számot a II. csoportban végzett felmérésből kapjuk. Ilyen mértékben szeretik a spenót. Példánkban ez 37%, azaz $P(A|B_2) = 0,37$.

Jelölje $P(B_1)$ annak a valószínűségét, hogy a megkérdezett a betegállományra vonatkozó kérdésre adott választ.

Jelölje $P(B_2)$ annak a valószínűségét, hogy a megkérdezett a spenótra vonatkozó kérdésre adott választ.

Feltehetjük, hogy $P(B_1) = P(B_2) = 0,50$, ami azt fejezi ki, hogy a megkérdezettek körülbelül fele válaszolt a betegállományra vonatkozó kérdésre, fele a spenótra vonatkozóra.

Feladatunkat megoldva a teljes valószínűség tételének [1, 2, 3] alkalmazásával:

$$P(A) = P(A|B_1) \times P(B_1) + P(A|B_2) \times P(B_2)$$

egyenlőség megoldásával kapjuk:

$$P(A|B_1) = 0,75.$$

Felmérésünkben a számadatainkkal azt a következtetést vonjuk le, hogy a megkérdezetteknek csak a 75%-a fogadja el a betegállományt, negyedrészüket akkor is dolgozik, ha beteg.

A hallgatók körében végzett felmérés

A fenti módszer gyakorlati alkalmazására a Dunaújvárosi Főiskola diákjai körében felmérést végeztünk. Ismertetjük néhány kérdésre adott válasz kiértékelését. A néhány kényes kérdésnek gondolt kérdésen kívül számos egyszerű, nem kényes kérdést is vizsgáltunk.

[1] Rényi Alfréd (1973): *Valószínűesszámitás*. Budapest: Tankönyvkiadó.

[2] Reimann József–Tóth Julianna (1989): *Valószínűesszámitás és matematikai statisztika*. Budapest: Tankönyvkiadó.

[3] Csernyák László (Szerk.) (1990): *Valószínűesszámitás*. Budapest: Nemzeti Tankönyvkiadó.

Egyszerű kérdések voltak például az alábbiak:

„Kellemesnek találja-e a főiskola oktató helyiségeit?”

„Volt-e kedvenc tanára a középiskolában?”

Az első kérdésre 88% volt az „Igen” válasz, az utóbbira 90%.

Kiderült az is, hogy a megkérdezettek 61%-a 40 km-nél távolabb lakik Dunaújvárostól, valamint 65%-uk jobban érzi magát a főiskolán, mint a középiskolában.

A „Szeret-e főiskolás lenni?” kérdésre csak igen válasz érkezett.

Egyszerű kérdésként is szerepelt és kérdéspár kérdéseként is az alábbi:

„Jónak tartja-e jelenlegi formájában a selmeci hagyományok őrzését?”

Egyszerű kérdésként kérdezve 70% „Igen” válasz érkezett, míg a páros kérdőíven 69%.

A kétféle úton kapott válaszból az derült ki, hogy ez a kérdés nem volt kényes kérdés, s a megkérdezett hallgatónak körülbelül 70%-a jónak tartja a hagyományörzés jelenlegi formáját.

Több kérdés összevetésével a felmérésből kiderült, van egy olyan hallgatói réteg is, amely a tudását a kötelező oktatási kereteken túl is szeretné növelni.

A felmérésből levont következtetések összhangban vannak-e a tapasztalattal?

A felmérésünk vajon helyes eredményre vezetett-e? Mennyire fogadhatók el az eredmények? Az eredmények csak arra a populációra érvényesek, amelyen a felmérést végeztük, illetve ahonnan a minta származik, ha a felmérés résztvevői mintaelemeknek tekinthetők.

Bizonyos kérdésre kapott válaszok igen-nem arányát helyesnek kell elfogadnunk. Így például a hallgató lakóhelyének/állandó tartózkodási helyének Dunaújvárostól mért távolsága objektív érték, feltételezzük, hogy a hallgatók tudják ezt az értéket. A tapasztalatok megerősítik, hogy a hallgatók szeretnek főiskolások lenni és a hagyományokat is kedvelik.

A hallgatók bekapcsolódása az egyes intézetek munkájába/kutatásokba, demonstrátori feladatok vállalása, versenyeken való részvételek, TDK-munkák igazolják azt a felméréssel kapott eredményt, hogy vállalnak/vállalnának többlet munkát.

Összefoglalás

A bemutatott munkánknak több célja is volt. Ezek egyike a matematikai ismeretek szilárdítása és a tananyag felhasználhatóságának demonstrálása egy gyakorlati alkalmazáson keresztül. Másik cél a hallgatók bevonása az alkalmazott matematika irányú tehetséggondozásba oly módon, hogy a legaktívabb hallgatók – a mentoráltak – tanári irányítással elvégzik az egyes feladatokat. A felmérésben résztvevő további hallgatók is megismerik magát a kérdőíves felmérési módszert és a hozzá kapcsolódó elméletet. A mentorált hallgatók a kötelező tananyagon felül a szakirodalom [4, 5, 6] és a saját tapasztalatuk alapján is gazdagítják tudásukat. Kiegészül ez azzal, hogy a kísérleti eredmények feldolgozásának korszerű, számítógépes módját megismerik. Módjuk nyílik betekinteni tudományos publikációk tartalmi, formai követelményeibe, lehetőséget kapnak prezentációk készítésére és tartására.

Végül, de nem utolsó sorban, a kérdésekre adott válaszok megismerése is cél volt. Több kérdésre kapott válaszok alapján kiderült, hogy valóban van egy olyan hallgatói réteg a főiskolán, aki a tudása gyarapítását fontosnak tartja és a kötelezőnél többet is tesz ezen a téren. Az ő igényeik kielégítéséhez hozzájárulnak a Dunaújvárosi Főiskolán megalakult szakkollégiumok is, amelyek keretében főként szakterületi ismeretek fejlesztése valósul meg, valamint személyiségfejlesztésre, készségfejlesztésre, kulturális programokon való részvételre is nyílik lehetőség.

[4] Hajdu Ottó–Pintér József–Rappai Gábor–Rédey Katalin (1994): *Statisztika I.* Pécs: Carbocomp.

[5] Herman Sándor–Pintér József–Rappai Gábor–Rédey Katalin (1994): *Statisztika II.* Pécs: Carbocomp.

[6] Falus Iván–Ollé János (2000): *Statisztikai módszerek pedagógusok számára.* Budapest: OKKER.

Agyszámítógép–interfészek kialakításának, tervezésének szempontjai

* Dunaújvárosi Főiskola
E-mail: katonaj@mail.duf.hu

** Dunaújvárosi Főiskola
E-mail: ujbanyit@mail.duf.hu

*** Dunaújvárosi Főiskola
E-mail: kovari@mail.duf.hu

[1] Wolpaw, J. R.,–Wolpaw, E. W. (2012): Brain-Computer Interfaces: Something new under the sun, in Brain-Computer Interfaces: Principles and Practice. J. R. Wolpaw –E. W. Wolpaw (Eds.): *Principles and Practice*. New York: Oxford University Press 3–12.

Összefoglalás: Az agyszámítógép–interfész (BCI) egy olyan komplex rendszer, amely különféle utasításokat és parancsokat képes közvetíteni egy számítógép számára az emberi agytól jövő jelek feldolgozása útján. A BCI-rendszerek többsége az elektroencefalogram (EEG) típusú eszközök működésén alapul. Ezen rendszerek tervezésénél és kialakításánál számos szempontot kell figyelembe venni. A cikk e szempontokról ad egy összefoglalót.

Kulcsszavak: EEG, BCI, agyszámítógép–interfész, agyjel.

Abstract: The brain-computer interface (BCI) is a complex system that is able to transmit various intructions and commands from the humain brain to a computer. The majority of BCI systems are based on devices which use the electroencefalogram (EEG) technology. A number of aspects have to be involved to planning and design these systems. The article provides a summary of these aspects.

Keywords: EEG, BCI, brain-computer interface, brain signal.

Bevezetés

Az elmúlt 20 évben az agy mérhető jeleinek feldolgozását és továbbítását végző agyszámítógép–interfész (BCI) technológiát alkalmazó kutatások száma jelentősen megnőtt. A kutatások eredményeképpen jelentősen javultak az agyi jelek mérését, rögzítését és feldolgozását biztosító eszközök, valamint a jelfeldolgozási algoritmusok, melyek az agyjelek értelmezését végzik.

A kilencvenes évek elején még csak néhány cikket találhattunk az agyszámítógép–interfészekről, mára ezzel a technológiával foglalkozó publikációk száma közel exponenciálisan megnőtt. [1]

Az elmúlt évek kutatásainak eredményeként a technológia megbízhatósága nagymértékben javult. Az agyszámítógép-interfész lényegében egy alternatív kommunikációs csatornát biztosít a technológiát használó személy és egy számítógépes alkalmazás között. A technológia lehetővé teszi a felhasználók számára, hogy gondolataikkal ezen a csatornán keresztül adatokat küldjenek. A BCI elsődleges célja az volt, hogy a súlyos neuromuscularis fogyatékoságban szenvedő betegeken segítsen. [2]

Azonban a multidiszciplináris kutatások és a fejlődésben lévő kognitív idegtudomány arra inspirálták a kutatókat, hogy új célokat tűzzenek ki a BCI területe előtt. A BCI-k az agyjeleket felhasználva információkat gyűjtenek a felhasználók szándékairól, majd ezeket lefordítja értelmezhető utasítások vagy parancsok összességére. [3]

Az agy-számítógép alapú interfészek és ezek jelen és jövőbeli továbbfejlesztése számos felhasználási területen nyújthat új alkalmazási lehetőségeket. A BCI-technológiának köszönhetően az elmúlt években számos szórakoztató eszköz került a piacra, azonban e technológia területeken is forradalmian új lehetőségeket nyújt, mint például olyan alkalmazások fejlesztése, amelyek képesek segíteni azoknak az embereknek, akik súlyos mozgásszervi rendellenességben szenvednek. Továbbá megemlíthetők az emberi figyelem, éber állapot (koncentráció vizsgálata), esetleg a lankadó figyelem, vagy elalvasközel állapot megfigyelése is, amely segítségével az elalvasközel állapot detektálható. Azon alkalmazások is nagy jelentőségűek, amelyek a reakcióidő csökkentésére irányulnak, mint például egy hirtelen vészhelyzet esetén a minél rövidebb időn belül történő beavatkozás, amely előállhat egy autó vészhelyzet esetén történő hirtelen lefékezésekor.

A BCI-rendszerek többsége az elektroencefalogram (EEG) típusú eszközök működésén alapul, mivel ezek a berendezések relatív egyszerű kialakításúak, hordozhatóak, biztonságosak és kezelésük is viszonylag egyszerű. Az EEG-eszközök által szolgáltatott jelek feldolgozásával a BCI-technológia lehetővé teszi az agyi elektromos aktivitás figyelemmel kísérését a különböző EEG-jeltípusok jellegzetességeinek megfigyelése útján. Az elmúlt években az EEG-jelek mérését és feldolgozását végző eszközök igen nagy fejlődésen mentek keresztül, melynek köszönhetően az alkalmazási lehetőségeik egyre több területre terjednek ki. Ezen rendszerek tervezésénél és kialakításánál számos szempontot kell figyelembe venni. Cikkünk e szempontokról ad összefoglaló képet.

[2] Wolpaw, J. R.–Birbaumer, N.–McFarland, D. J.–Pfurtscheller, G.–Vaughan, T. M. (2002): Brain–computer interfaces for communication and control. *Clinical Neurophysiology*. Vol. 113. No. 6. Jun. Pp. 767–791.

[3] Zander, T. O. –Kothe, C. (2011): Towards passive brain-computer interfaces: applying brain-computer interface technology to human-machine systems in general. *Journal of neural engineering*. Vol. 8, No. 2. Apr.

[4] Lebedev, M. A.–Nicolelis, M. A. (2006): Brain–machine interfaces: past, present and future. *Trends Neurosci.* 29. Pp. 536–546.

[5] Millett, D.–Berger, H. (2001): From Psychic Energy to the EEG. *Perspectives in Biology and Medicine.* Vol. 44. No. 4. Pp. 522–542. New Jersey: Johns Hopkins University Press.

[6] Kéri Sz.–Gulyás B. (2003): Elektrofiziológiai módszerek a kognitív idegtudományban. In: Pléh Cs.–Kovács Gy.–Gulyás B. (Szerk.): *Kognitív idegtudomány.* Budapest: Osiris. Pp. 81–98.

Agyszámítógép-interfészek fejlődése

Az agyszámítógép-interfészt (BCI) néha úgy is hívják, hogy közvetlen neurális interfész vagy agygép-interfész (BMI), amely egy közvetlen kommunikációs csatorna az agy és egy külső eszköz – általában számítógép – között. [4] Az agyszámítógép-interfészek történelme egészen a '60-as évekig nyúlik vissza. Ebben az évtizedben még az volt a köztudatban, hogy különböző eszközök irányítása csupán agyhullámok segítségével megvalósíthatatlan. Ennek ellenére már az első agyhullámokkal történő kutatások is figyelemfelkeltőek voltak, amikor is Hans Berger német tudós 1929-ben az emberi fejbőrön keresztül képes volt az agyi elektromos tevékenységeket rögzíteni. Azonban a 20. század elején a technológia még nem volt elég fejlett ahhoz, hogy ezeket az agyi jeleket feldolgozzák és kiértékeljék. Ennek a korlátnak következtében Berger nem tudta egyértelműen bizonyítani az agyi jelek jelentőségét. [5]

Az első jelentős BCI-vel kapcsolatos kutatási eredmények Grey Walter nevéhez fűződnek, aki 1964-ben publikálta vizsgálatai eredményeit. Kutatásában közvetlenül egy páciens agyának azon részéhez csatlakoztatott elektródákat, amelyek a motoros funkciókért felelősek.

Walter arra kérte a beteget, hogy kapcsolja be a tévét. A kísérlete alatt a teljes agyi aktivitást figyelemmel kísérte és egy EEG-berendezés segítségével rögzítette azokat. Az EEG-készülék eredményei alapján Walter felfigyelt arra, hogy minden egyes bekapcsolás előtt kis feszültségű agyjelek mérhetőek. Ezeket az agyjeleket Grey Walter készenléti hullámoknak nevezte el. Következő kísérletében a készenléti hullámokat elektromos úton felerősítette, majd ezzel a tévékészüléket vezérelte. Ezek után arra kérte a páciens, hogy gomb használata nélkül csak gondoljon arra, hogy be szeretné kapcsolni a tévékészüléket. Walter felfedezte, hogy a páciens pusztán a gomb megnyomásának gondolatával is képes a tévékészüléket bekapcsolni. Arra a következtetésre jutott, hogy az úgynevezett készenléti hullám volt a bekapcsolást kiváltó tényező. Walter úgy találta, hogy némi késedelem mutatkozik az észlelés és az agyi aktivitás között, de ezek agyi jelek hamarabb jelentkeznek, mint ahogy a beteg megnyomta volna a gombot. Az agy hamarabb képes volt motoros funkciókat irányítani, még azelőtt, mielőtt a tényleges mozgás létrejött volna. Ez azt jelenti, hogy tényleges mozgás nélkül is képes az agy motoros funkciókra, így jött létre az első BCI-rendszer. [6]

A BCI-kutatások az 1970-es években tovább folytatódtak a kaliforniai egyetemen Los Angelesben a National Science Foundation támogatásával. A kutatások következményeként létrejött egy szerződés a DARPA-val (Defense Advanced Research Projects Agency, azaz Fejlett Védelmi Kutatási Projektek Ügynöksége). A későbbiekben a tudományos irodalomban megjelent az új agy-számítógép kifejezés. [7] A sikeres kísérleteket követően több éven keresztül kevés előrelépés történt a BCI-kutatások területén, azonban az utóbbi években ez a helyzet jelentősen megváltozott. Az elmúlt évtizedekben végzett neurológiai kutatások nagy segítséget nyújtottak abban, hogy jobban megértsük a keletkező agyi jeleket.

Agyszámítógép-interfészek típusai

A BCI-rendszerek legtöbbször a nem-invazív EEG, vagyis a fejbőrön elhelyezett elektródák által mért agyjelek módszerét alkalmazza, mivel ez relatíve könnyen megvalósítható. Ráadásul a legújabb előrelépéseknek köszönhetően megjelentek olyan EEG-berendezések, amelyek vezeték nélküli rendszerek, továbbá ezek az eszközök viszonylag olcsón elérhetőek. Ennek következtében manapság a legtöbb BCI-kutatás az EEG-re alapozott. A BCI-rendszerek különböző kategóriákba oszthatóak, ezek az aktív, reaktív és passzív rendszerek a következők szerint: [8]

– *Aktív BCI-rendszerek:* Egy aktív BCI-rendszerben a felhasználó önkéntes alapon és külső eseményektől függetlenül olyan utasításokat generálnak, amelyeket egy külső alkalmazásnak továbbítanak. A legjobb példa az aktív BCI-rendszerekre az agy motoros funkcióinak, az elképzelt mozgás során mért jelek megjelenítése.

– *Reaktív BCI-rendszerek:* Egy reaktív BCI-rendszerben az agy reakcióját egy külső inger váltja ki. Ezek a rendszereket a felhasználóik közvetett módon modulálják, mivel a felhasználó képes irányítani egy alkalmazást. Ezen BCI-alkalmazások legjobb példája a P300-nak nevezett módszer vagy az egyéb fényingerek.

– *Passzív BCI-rendszerek:* A passzív BCI-rendszerek kimeneteik eredményei egy kognitív visszacsatolás a felhasználó önkényes agyi tevékenységéről, ahol az alanyra semmilyen hatás nincs. Ezekre a rendszerekre a legjobb példa az izgalomra vagy terhelésre adott visszacsatolás.

[7] Wolpaw, J. R. –Birbaumer, N.–McFarland, D. J.–Pfurtscheller, G.–Vaughan, T. M. (2002): Brain-computer interfaces for communication and control. *Clinical Neurophysiology*. 113. Pp. 767–791.

[8] Zander, T. O.–Kothe, C. –Jatzev, S.–Gaertner, M. (2010): Enhancing Human-Computer Interaction with Input from Active and Passive Brain-Computer Interfaces. In: Tan, D. S. –Nijholt, A. (Eds.): *Brain-Computer Interfaces Applying our Minds to Human-Computer Interaction*. London: Springer London. Pp. 181–199.

[9] Kübler, A.–Kotchoubey, B. (2007): Brain-computer interfaces in the continuum of consciousness. *Current Opinion in Neurology*. Pp. 643–649.

[10] Zickler, C. et al. (2011): A brain-computer interface as input channel for a standard assistive technology software. *Clinical EEG and Neuroscience*. Pp. 236–244.

Agyszámítógép-interfészek kialakításánál, tervezésénél figyelembe veendő szempontok

A BCI-rendszerek legtöbbje az EEG-módszerrel mérhető agyjelek, vagyis az agy elektromos aktivitása következtében létrejövő feszültségváltozások, mérése, digitalizálása, szűrése, valamint részleges kiértékelése útján feldolgozott információt hoznak létre. Az információ felhasználásával egyéb alkalmazásokat vagy eszközöket működtetnek. Az alkalmazásokban a felhasználó visszajelzést kap az általa kiváltott cselekvésekről, vagyis a BCI-visszacsatolást valósít meg a felhasználó és a rendszer között.

A BCI-alapú irányítások megvalósításakor figyelembe veendő négy fő szempont: [9]

- A BCI-eszközt használók eltérő jellege, egyedi jellemzői.
- Az alkalmazott BCI-eszköz jelfeldolgozásának és kiértékelésének módja.
- A visszacsatolás módja és a kiadható utasítások.
- A BCI-vel irányított alkalmazás.

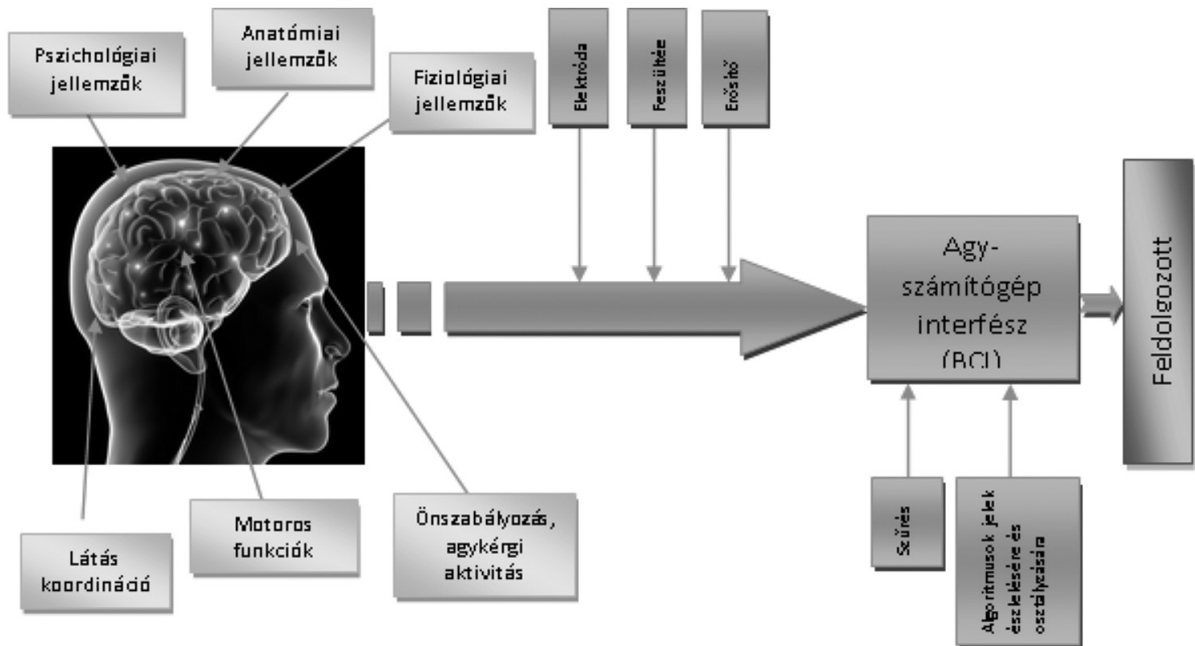
A BCI-eszközök tervezésénél a nemzetközi ISO 9241–210 (ember-gép kapcsolat ergonómiája, 210. rész: Interaktív rendszerek emberközpontú tervezése) szabvány figyelembevétele ajánlott. A szabvány alapján három megközelítést kell figyelembe venni:

- Üzleti követelmények (kereslet, ár).
- Felhasználói követelmények, működési leírás (felhasználó szemszögéből meghatározott igények, használhatóság, használat célja).
- Műszaki követelmények (adott funkciók megvalósítása, adat-struktúra, belső feldolgozás, adatkommunikáció protokollja, sebessége, megfelelő EEG-mérés és komfort közötti kompromisszum).

Az ISO 9241–201 (2010) definiálja a termékek használhatóságát, mint a termék használatával meghatározott célok elérésének eredményesség, hatékonyság és a felhasználói elégedettség együttesét. Eredményesség arra utal, hogy a felhasználók milyen pontosan és teljes körűen tudják elvégezni használatukkal a kitűzött feladatot. Hatékonyságot az eredményesség elérése érdekében befektetett felhasználói erőfeszítés és az idő határozza meg. A felhasználói elégedettséget a felhasználó által elvárt komfort, hatékonyság és elégedettség együttese befolyásolja. [10]

Ezen a BCI-eszközökre is vonatkozó általános előírások határozzák meg a BCI-eszközök tervezésének és fejlesztésének irányait. A BCI-eszközök fejlesztésében igazi nagy áttörését a már említett digitálisteknikakorszak, a digitális technika fejlődésével az egyre pontosabb feldolgozást és kiértékelést lehetővé tevő jelfeldolgozó algoritmusok megvalósításához és futtatáshoz szükséges megfelelő számítási teljesítmény adta. Ezek odáig vezettek, hogy képesek vagyunk az agyi jeleken bonyolult, valós idejű jelfeldolgozást végezni, amely nem igényel drága és méreteiben nagy berendezéseket. A jelfeldolgozó algoritmusok csak akkor szolgáltatnak megfelelő információt, ha az agyjelek alacsony zajszint mellett mérhetőek. Ez akkor valósítható meg, ha μV nagyságrendű agyjelek jó jel-zaj viszony mellett mérhetőek. Ennek biztosítása céljából a mérés helyéhez lehető legközelebb kell a jel digitalizálását alacsony zajszintű A/D-átalakítóval kivitelezni. Évről évre több BCI-vel kapcsolatos tanulmány jelenik meg, illetve számos termék érhető el erre a technológiára alapozva. Egy BCI-eszköz felépítését mutatja az 1. ábra.

1. ábra. BCI-eszköz felépítése.



Összefoglalás

Összefoglalva, egy BCI-eszköz tervezésekor és kialakításakor elsősorban a következő szempontokat kell szem előtt tartani:

– *Megfelelő és kényelmes használat:* A jelenlegi EEG-rendszereknél szabvány, elektrolitikus géllal bekent elektródákat használnak, hogy csökkentsék az elektróda és fejbőr felszíni impedanciáját. Az elektrolitikus gél használata kényelmetlen és nem megfelelő, főleg amikor nagyszámú elektródát használnak. Egy elektróda sapka sok elektródával kényelmetlen viselet és ezért nem alkalmas hosszú idejű felvételekhez. Továbbá, egy sokcsatornás BCI elég költséges. Ezen okokból kifolyólag, a BCI-rendszerekben az elektródák számának minimalizálása, száraz típusú érzékelők alkalmazása az eszköz bonyolultsága, ára és kényelmes használata szempontjából előnyös.

– *Stabil rendszerteljesítmény:* Figyelembe véve, hogy az adatrögzítés nem földelt környezetben történik, ahol erős elektromágneses zavaró hatás is jelen van, egy aktív elektróda (érzékelő és erősítő/feldolgozó egyben) használata sokkal jobb, mint egy passzív elektródaé. Ez biztosíthatja azt, hogy a felvett és feldolgozott jelet kevésbé befolyásolják a zavaró jelek. Másrészt a jelfeldolgozásnak képesnek kell lennie időben észlelni a rossz elektróda-kapcsolatot, és ezt automatikusan jelezni, esetleg a feldolgozó algoritmust módosítani, hogy hibatűrőbb legyen.

– *Alacsony költségű hardver:* BCI-alapú eszközök általános elterjedésének feltétele a minél alacsonyabb költségű hardver. Ez a digitális elektronika, és a jelfeldolgozó processzorok fejlődésével, azok árának csökkenésével napjainkra részben megvalósult.

A cikkben meghatározott szempontok figyelembevételével olyan BCI-eszközök fejleszthetők, melyek megfelelnek mindazon elvárásoknak, amelyek ezen eszközök elterjedését és mindennapi használatát megalapozzák. Ezen a területen jelenleg is folyó kutatások következtében feltehetően a jövőben egyre több BCI-alapú eszközzel találkozhatunk majd, melyek számos alkalmazási területen nyújthatnak új lehetőségeket.

Hopf bifurcation in higher dimensions

Abstract: In this work we investigate a problem within bifurcation theory, an important chapter of the qualitative theory of differential equations. Hopf bifurcation is studied in a biological model. We give an overview of the theorems determining Hopf bifurcation for two-, three- and four variable systems. These theorems are illustrated by some examples using XPPAut.

Keywords: Ordinary differential equation, Bifurcation, Hopf bifurcation.

Összefoglalás: Jelen dolgozatban a differenciálegyenletek kvalitatív vizsgálatának egy jelentős fejezetéhez, a bifurkációelmélethez kapcsolódó témával foglalkozunk. Röviden bemutatjuk a Hopf-bifurkációt, majd áttekintjük a meghatározásához ismert tételeket két-, három- és négydimenziós rendszerek esetén. Ezen tételeket korábbi kutatásainkból származó gyakorlati példákkal illusztráljuk az XPPAut alkalmazásával.

Kulcsszavak: Közönséges differenciálegyenlet, Bifurkáció, Hopf-bifurkáció.

Introduction

Investigating ordinary differential equations is an elementary problem in natural sciences. Ordinary differential equations often involve a number of parameters, which determine the qualitative properties. Symbolic solutions can be rarely found if the equations are nonlinear. In this article a method (parametric representation method, PRM, [1]) and a computer program (AUTO [2, 3] XPPAut [4, 5]) are used to study the qualitative changes of a differential equation, which is applied to investigate a problem of mathematical biology.

*College of Dunajváros
E-mail: nagy.balint@mail.duf.hu

[1] Simon, P. L.–Farkas, H.–Wittmann, M. (1999): Constructing global bifurcation diagrams by the parametric representation method. *Journal of Computational and Applied Mathematics*. 108. Pp. 157–176.

[2] Doedel, E. J.–Champneys, A. R.–Fairgrieve, T. F.–Kuznetsov, Y. A.–Sandstede, B. Wang, X.-J. (1997): *AUTO97: Continuation And Bifurcation Software For Ordinary Differential Equations* (with HomCont). <http://cmvl.cs.concordia.ca/publications.html>

[3] AUTO homepage: <http://indy.cs.concordia.ca/auto/> (2012. 11. 29.)

[4] Ermentrout, B. (2002): *Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAut for Researchers and Students*.

[5] XPPAut homepage: <http://www.math.pitt.edu/~bard/xpp/xpp.html> (2012. 11. 29.)

[6] Hopf, E.: *Abzweigung einer periodischen Lösung von einer stationären Lösung eines Differentialsystems*. Berichten der Mathematisch-Physischen Klasse der Sächsischen Akademie der Wissenschaften zu Leipzig. XCIV. Band Sitzung vom 19. Januar 1942.

[7] Kuznetsov, Y. A. (2004): *Elements of Applied Bifurcation Theory*. New York: Springer.

[8] Guckenheimer, J.–Myers, M.–Sturmfels, B. (1997): Computing Hopf Bifurcations I. *SIAM Journal on Numerical Analysis*. 34. Pp. 1–21.

[9] Guckenheimer, J.–Myers, M. (1996): Computing Hopf Bifurcations II. *SIAM Journal on Scientific Computing*. 17. Pp. 1275–1301.

[1] previous page

[10] Simon, P. L.–Hild, E.–Farkas, H. (2001): Relationships between the discriminant curve and other bifurcation diagrams. *Journal of Mathematical Chemistry*. 29. Pp. 245–265.

Hopf bifurcation

In this section we give the definition of the Hopf bifurcation [6, 7] in one of the most simple forms using [8, 9].

Let $\dot{x} = f(x, \mu)$, $f: \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^n$ be a smooth n -dimensional vector field depending upon parameters with the property that (x_0, μ_0) is an equilibrium point at which the Jacobian matrix $D_x f$ has no zero eigenvalues and a single, simple pair of pure imaginary eigenvalues $\lambda, \bar{\lambda}$. Assume further that $\lambda, \bar{\lambda}$ cross the imaginary axis transversally as the parameters are varied. Then there is a smooth submanifold P of dimension $k+1$ containing (x_0, μ_0) which is a union of periodic orbits and equilibrium points of f . We are interested in the symbolic and numerical detection of Hopf bifurcation points. First the PRM method investigating the symbolic case is studied.

Parametric representation method. In this section we summarize the main theorems of PRM. These theorems are stated and proved in a more general form in [1, 10].

Let us consider the equation $\dot{X}(t) = H(X(t), u)$ where $H: \mathbb{R}^n \times \mathbb{R}^2 \rightarrow \mathbb{R}^n$ is a differentiable function, $X \in \mathbb{R}^n$, $X(i)$ is the vector of state variables at time t , and $u \in \mathbb{R}^2$ is the vector of parameters. Let us suppose that the system of equations $H(X, u) = 0$ giving the stationary points can be reduced to a single equation and two control parameters, u_1 and u_2 , involved linearly in the right-hand side of the reduced equation. Now we can write the reduced equation into the form

$$h(x, u_1, u_2) = h_0(x) + h_1(x)u_1 + h_2(x)u_2 = 0, \quad (1)$$

where prime denotes differentiation with respect to x . We introduce the singularity (or saddle-node bifurcation) set

$$S = \{(u_1, u_2) \in \mathbb{R}^2: \text{exists } x \in \mathbb{R}, h(x, u_1, u_2) = h'(x, u_1, u_2) = 0\},$$

which can be given by the PRM as a curve parameterized by x . Hence S can easily be constructed, and the solutions belonging to a given parameter pair can be determined by a simple geometric algorithm.

Let us solve equations $h(x, u_1, u_2) = 0$ and $h'(x, u_1, u_2) = 0$ for (u_1, u_2) . This solution is the parametric form of the D -curve. Using this curve we can determine the number and the value of the solutions of (1), because the following lemmas hold [10].

Lemma 1 (Tangential Property). $x \in \mathbb{R}^n$ is a solution of (1) for the parameter values u_1 and u_2 if and only if a tangent line can be drawn from the point (u_1, u_2) to the D -curve at the point $D(x)$.

Lemma 2 (Convexity Property). The D -curve consists of convex arcs that join with a common tangent or asymptote.

The convexity of the separate arcs means that they lie on one side of the tangent line that belong to any point of the arc. The D -curve can be plotted in the plane of (u_1, u_2) .

Let (u_1^*, u_2^*) be a parameter pair that is moved in the parameter plane. If (u_1^*, u_2^*) crosses the D -curve, the number of stationary points of (1) changes by two.

Lemma 3. Let J be the Jacobian of $\dot{X}(t) = H(X(t), u)$ where $H: \mathbb{R}^2 \times \mathbb{R}^k \rightarrow \mathbb{R}^2$ is a differentiable function, $X \in \mathbb{R}^2$, $X(i)$ is the vector of state variables at time t , and $u \in \mathbb{R}^k$ is the vector of parameters. Let $g = \text{Tr } J$ denote the trace of J . If J has two pure imaginary eigenvalues, then $g=0$.

Lemma 4. Let J be the Jacobian of $\dot{X}(t) = H(X(t), u)$, where $H: \mathbb{R}^3 \times \mathbb{R}^k \rightarrow \mathbb{R}^3$ is a differentiable function, $X \in \mathbb{R}^3$, $X(i)$ is the vector of state variables at time t , and $u \in \mathbb{R}^k$ is the vector of parameters. $\text{Det } J$ and $\text{Tr } J$ denote the trace and the determinant of J , and J_{11}, J_{22}, J_{33} are the corresponding minors of J . We introduce a new notation $g = \text{Tr } J(J_{11} + J_{22} + J_{33}) - \text{Det } J$. If J has two pure imaginary eigenvalues, then $g=0$.

Lemma 5. Let J be the Jacobian of $\dot{X}(t) = H(X(t), u)$, where $H: \mathbb{R}^4 \times \mathbb{R}^k \rightarrow \mathbb{R}^4$ is a differentiable function, $X \in \mathbb{R}^4$, $X(i)$ is the vector of state variables at time t , $u \in \mathbb{R}^k$ is the vector of parameters. Let $P(\cdot)$ denote the characteristic polynomial of J , $P(\lambda) = c_0 + c_1\lambda + c_2\lambda^2 + c_3\lambda^3 + \lambda^4$ and let $g = c_0c_3^2 - c_1c_2c_3 + c_1^2$. The matrix J has two pure imaginary eigenvalues if and only if $g=0$ and $c_1c_3 > 0$.

Using Lemma 3–5 we can investigate the stability of the stationary points by *g*. AUTO – a bifurcation tool. AUTO is a popular, publicly available software for continuation and bifurcation problems in ordinary differential equations originally written in 1980, and widely used in the dynamical systems community. AUTO can do a limited bifurcation analysis of systems of ordinary differential equations of the form $\dot{x} = f(x, u)$. Here u denotes one or more free parameters [2].

[10]
previous page

[2]
first page

[5]
first page

[11] Tyson, J. J.
–Hong, C. I.–
Thron, C. D.–
Novak, B.(1999):
A simple model
of circadian
rhythms based
on dimerization
and proteolysis
of PER and TIM,
*Biophysical
Journal*. 77.
Pp. 2411–2417.

[12] Nagy, B.
(2008):
Comparison of
the bifurcation
curves of a two-
variable and a
three-variable
circadian
rhythm
model. *Applied
Mathematical
Modelling -
Journal*. Pp.
1587–1598.

To use AUTO, some knowledge in FORTRAN is necessary. If the user is not familiar with FORTRAN a nice program named XPPAut (X-Windows Phase Plane plus AUTO) [5] can be applied. XPPAut is a tool not only for solving and analyzing ordinary differential equations, but for delay equations and functional equations as well. (XPP has been successfully compiled on a number of systems, for example a PC using Linux. (It also runs under Win95/NT/98 if you have an X-Server.) In this work Ubuntu Linux was used. An animation package allows the user to create animated versions of simulations is available. It is possible to automatically generate "movies" of three-dimensional views of attractors or parametric changes in the attractor as some parameters vary. Fortunately, AUTO is built in XPPAut, and it has a nice, friendly interface as well. XPPAut has a useful online documentation.

Investigating bifurcations with XPPAut

In this section we introduce how to investigate the Hopf bifurcation of an ODE using XPPAut. First, an .ode file is constructed to study the Drosophila biological clock introduced in [11]. The system is as follows:

$$\dot{M} = \frac{v_m}{1 + (P_2 / P_c)^2} - k_m M$$

$$\dot{P}_1 = v_p M - \frac{k_1 P_1}{J_p + P_1 + r P_2} - k_3 P_1 - 2k_a P_1^2 + 2k_d P_2$$

$$\dot{P}_2 = k_a P_1^2 - (k_d + k_3) P_2 - \frac{k_2 P_2}{J_p + P_1 + r P_2}$$

This system is nonlinear, hence finding symbolic solution is hardly possible, which is why we use numerical methods by XPPAut. (Although a lot of information can be found about the behavior of the system [12].)

The first line of the .ode file includes the name of the model in a comment line starting with a #.

```
# Drosophila Tyson-Hong-Novak
```

In the next three lines the ODE is given. Here x' denotes the derivative of x(t) with respect to t.

```
M' = num/(1+(P2/pc)^2) - km*M
P1' = nup*M-k1*P1/(jp+P1+r*P2)-k3*P1-2*ka*P1^2+2*kd*P2
P2' = ka*P1^2-kd*P2-k2*P2/(jp+P1+r*P2)-k3*P2
```

The parameters are set to the values suggested by Tyson *et al.* [11] in a line starting with "p" or "param". (In this line we mustn't use a space next to the "=".)

```
param k1=1, k2=0.03, k3=0.1, km=0.1, num=1, pc=0.01, jp=0.05, ka=200,
kd=1, r=2, nup=1
```

Initial conditions are given.

```
M(0) = 0.1341778
P2(0) = 0.08574842
P1(0) = 0.02289641
```

This system is stiff, so in the the method.

```
@ METHOD=stiff
```

The last line of the file is „d” or „done”
done

After starting XPPAut we determine the equilibrium point of the system with the parameter values in the .ode file hitting I (Initial conditions) G (Go) and I (Initial conditions) L (Last) a few times, because AUTO should start from a known solution, usually a stable stationary point or a stable periodic orbit. If we start AUTO next to a stationary point, no bifurcations can be found. In this model a stationary point exists with the above settings, as the graph suggests, or can be seen in the Data viewer. After finding this equilibrium, we can start AUTO by the buttons F (File) and then A (Auto).

The default settings give us the first parameter of the .ode file as the bifurcation parameter. Now we may change these settings in the Parameters menu. We would like to investigate bifurcation varying k1, so we don't change this setting. By clicking Axes and Hi-lo, we plot the minima and maxima of one of the coordinates (M) of the solution depending on one of the parameters, here k1.

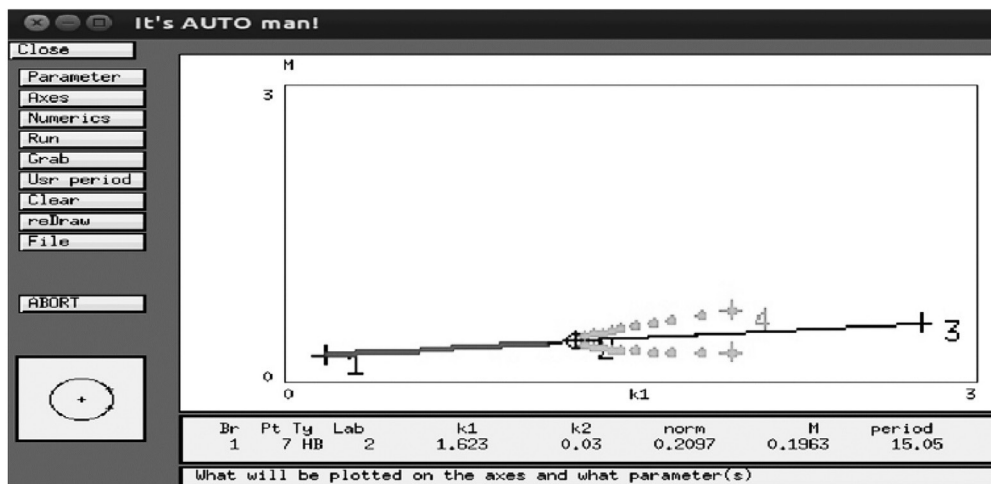
Results

In one of our earlier work Hopf bifurcation is found if . In what follows we give this bifurcation numerically. Hence the k1 axis is set to Xmin=0 and Xmax=3. Hitting N (Numerics) the computation can be set. Par max is the maximum value of the parameter, Ds is a suggested value for the stepsize, because AUTO automatically finds the stepsize between Dsmin and Dsmax. Clicking Run and Steady states starts the computation. Two kinds of output are given.

[12] Nagy, B. Comparison of the bifurcation curves of a two-variable and a three-variable circadian rhythm model, Applied Mathematical Modelling, (2008) 1587–1598.

The points of the bifurcation diagram are plotted in the $k1$ – M parameter-variable coordinate system (Figure 1.), and some information about the computed point is listed below the graph. The third column is TY, this is the type of the bifurcation point. Here HB means, that point label (LAB) 2 is a Hopf bifurcation point with parameter $k1 \approx 1.623$.

Figure 1. The bifurcation curve of the Tyson circadian rhythm model in the $k1$ – M plane.



Summary

In this work a biological model is investigated from the bifurcation point of view. A numerical computational tool is applied to compute the Hopf bifurcation point of the model. The results confirm our findings in a previous article [12].

ERP-rendszerek oktatása a Dunaújvárosi Főiskolán

Összefoglalás: A cikk célja, hogy bemutassuk, miként és milyen irányba fejlődött a Dunaújvárosi Főiskola integrált vállalatirányítási rendszerekkel kapcsolatos oktatási tevékenysége az elmúlt közel másfél évtizedben, mely során az SAP különböző nagy- és kisvállalati verzióin keresztül eljutott a 2013 őszén oktatásba kerülő Microsoft Dynamics NAV alkalmazásáig. A cikkben összevetjük egymással az egyes rendszerek oktatásbeli jellemzőit, és kitérünk az oktatásban alkalmazott gyakorlati módszerekre és feladatokra, ezek fejlődésére. Továbbá elemezzük a vállalatirányítási rendszerekkel foglalkozó tantárgyak esetében mérhető oktatási eredményességet a tantárgyi eredmények hallgatói tanulmányi átlagokkal történő összevetésével.

Kulcsszavak: ERP-oktatás, SAP R/3, SAP Business One, SAP ERP, MS Dynamics NAV, oktatási hatékonyság, teljes logisztikai vevőkiszolgálási folyamat.

Abstract: The goal of this article is to present the development of the teaching activity of ERP systems at the College of Dunaujváros in the last half and other decade through using SAP products for small, medium and large enterprises till starting using Microsoft Dynamics NAV in September of 2013. The article compares the features of these ERP systems from the viewpoint of teaching and summarises the practical methods and its development used in the education. Moreover it try to measure the efficiency of teaching through the student results related to their academic average.

Keywords: teaching ERP systems, SAP R/3, SAP Business One, SAP ERP, MS Dynamics NAV, teaching efficiency, full logistic process of customer service.

* Dunaújvárosi Főiskola
Társadalomtudományi Intézet
Vezetési és Vállalkozástudományi Tanszék
E-mail: radail@mail.duf.hu

** Dunaújvárosi Főiskola
Informatikai Intézet Szoftverfejlesztési és Alkalmazási Tanszék
E-mail: honfivid@mail.duf.hu

*** Dunaújvárosi Főiskola
Informatikai Intézet Szoftverfejlesztési és Alkalmazási Tanszék
E-mail: kiru@mail.duf.hu

**** Dunaújvárosi Főiskola
Informatikai Intézet Szoftverfejlesztési és Alkalmazási Tanszék
E-mail: mkollar@mail.duf.hu

[1] Mészáros K. (2005): *SAP R/3 ismeretek*. Dunaújváros: DUF Press.

[2] Mészáros K. (2005): *Logisztikai folyamatok az SAP R/3-ban – MM*. Dunaújváros: DUF Press.

[3] Mészáros K. (2006): *Logisztikai folyamatok az SAP R/3-ban – SD*. Dunaújváros: DUF Press.

[4] Mészáros K. (2005): *Operatív vállalatirányítási rendszerek bevezetése*. Dunaújváros: DUF Press.

[5] Mészáros K. (2005): *Pénzügyi folyamatok a SAP-ban: Pénzügyszámvitel – tárgyi eszközgazdálkodás - kontrolling*. Dunaújváros: DUF Press.

Integrált vállalatirányítási rendszerek az oktatásban

Az ERP oktatási rendszerek a műszaki informatikus, a gazdasági informatikus, a közgazdász és a műszaki menedzser alapszakok (utóbbin angol nyelvű képzésben is), valamint a logisztikai műszaki menedzser-asszisztens felsőfokú szakképzési szak tantárgyai keretében folytak.

Az elsők között a közgazdász szak alkalmazta 2001. tavaszától az SAP R/3 4.6C rendszerét *Üzleti informatika* szakirányán, mely keretében az alap kezelési ismereteket [1], a pénzügyi és a kontrolling [5] modult, valamint a logisztikához kapcsolódó anyaggyártási [2] és értékesítési [3] modulokat ismerték meg a hallgatók, illetve az ASAP bevezetési módszertant [4] tekintették át. Ezen témához kapcsolódó kérdések gyakoriak voltak a záróvizsgákon is. A rendszer és elméleti hátterének megismeréséhez 5 jegyzet született Mészáros Károly és szerzőtársai tollából. [1, 2, 3, 4, 5] Ezek a jegyzetek kiváló elméleti hátteret adtak, azonban a gyakorlati – a rendszer használatával kapcsolatos – fejezetek kissé elnagyoltak voltak és komoly hiányosságuk, hogy nem készültek hozzá konkrét gyakorlati példák, illetve megoldható feladatsorok. A sorozat negyedik kötete az ASAP-módszertant bemutató *Operatív vállalatirányítási rendszerek bevezetése* című jegyzet, melyet szintén szerencsés lett volna konkrét esettanulmányokkal kiegészíteni, így kevésbé „izzasztották” volna meg a közgazdász-jelölt hallgatókat a záróvizsgán e témában elhangzó kérdések.

Ezután a logisztikai műszaki menedzser-asszisztens felsőfokú szakképzési szak *Logisztikai információ-rendszerek* tantárgyába, ahol a Vértess Edit-féle *Logisztikai informatika* tankönyv elméletére alapozva a 2006–2007. II. félévében a levelezős, 2007–2008. II. félévében a nappali hallgatók ismerkedtek meg az SAP R/3 kezelési alapokkal és az anyaggyártási modullal. Már ezekben a félévekben megjelent a főiskolán a Microsoft az Dynamics Axapta rendszerrel, de csak demó erejéig, egyéni gyakorlatra a hallgatóknak nem volt lehetőségük.

A 2006-ban akkreditált műszaki menedzser szak első évfolyama a 2008–2009. tanévben ért a szakirányos képzéshez, ennek keretében a logisztikai szakirányon az SAP R/3 anyaggyártási és értékesítési moduljával, a termelésirányítási szakirányon pedig az termelésirányítási

moduljával ismerkedtek. Az idősödő R/3 mellett az akkor már piacon lévő SAP ERP-vel és Business Suite-tal csak elméletben találkoztak Herger Tamás előadásában. Viszont ezzel párhuzamosan a főiskola oktatásában bevezetésre került az SAP Business One kis és középállati vállalatirányítási rendszer, mely a vállalatirányítási rendszerek oktatásának alapozó tantárgyában került alkalmazásra az informatikai és műszaki menedzser alapképzésekben, valamint kiváltotta a logisztikai menedzser-asszisztenseknél az SAP R/3 alkalmazását a *Logisztikai információrendszerek* tárgyban. A szoftver előnyei között megemlítendő, hogy kis- és középállati szintű komplexitásával lehetővé tette teljes logisztikai-vevőkiszolgálási folyamat önálló gyakorlását saját esettanulmányok kidolgozásával. Ez végigvezette a hallgatót a cikktörzs- és darabjegyzék, valamint partnertörzs és üzleti előrejelzések rögzítésétől, az anyagszükséglet-tervezéssel támogatott beszerzésen, gyártáson és raktári folyamatokon keresztül a komissiózásig, vevő felé történő kiszállításig, számlázásig és számlafogadásig, a banki tranzakciók rögzítéséig, továbbá lehetővé tette a „vállalat” „félèves” eredményeinek elemzését a hallgatók és oktatók számára egyaránt. Ezen gyakorlati feladattípus a főiskola R/3 rendszerében és az SAP ERP ELTE által szolgáltatott IDES rendszerében is alkalmazásra került, de a magasabb komplexitás miatt csak szűkebb terjedelemben, elsősorban a beszerzési, anyaggazdálkodási és értékesítési folyamatokra szorítkozva, kihagyva a pénzügyi és kontrolling folyamatokat.

A 2007–08. tanév 1. félévében a műszaki informatikus és a gazdasági informatikus szakokon elindultak a VIR A és VIR B tantárgyak, melyek az üzemeltetési és vállalatmodellezési kérdésekkel foglalkoznak. Emellett 2011–12. és 2012–13. tanévekben az SAP alapjai, SAP üzemeltetése és SAP programozása tantárgyak voltak jelen egy-egy félévig. A fent említett SAP ERP többszöri próbálkozás után a 2011–2012. tanév I. félévétől 3. féléven át került alkalmazásra, kiváltva a máig működő, eredeti 2001. évjáratú hardverén futó DFI SAP R/3 4.6C rendszert. Azonban a 2012-es év végén az SAP visszavonta az ELTE szolgáltatási jogosultságát, ezért a már csak jóval drágább SAP University Alliance program keretében van lehetőség hasonló szolgáltatás igénybevételével, egy nagyságrenddel magasabb áron. Mivel ezt a főiskola nem tudja finanszírozni, az SAP Business One és Business by Design komoly versenytársaként egyre terjedő Microsoft Dynamics NAV iránt kezdett érdeklődni, mely magasabb funkcionalitással és kedvezőbb áron érhető el, egyszerre váltva ki a főiskola SAP kiöregedő 4.6C és a relatív drágán licenszelhető kisvállalati rendszerét.

Az egyes rendszerek oktatásban történő alkalmazhatóságának összehasonlítása

SAP R/3 4.6C

Az SAP R/3 rendszert érintő tantárgyakat mindig nagy várakozás övezi a hallgatók körében, mivel közismertek az igen magas SAP szakértői rapidíjak. Azonban a rendszer használata igen komoly felkészültséget igényel, felhasználói felülete jelentősen eltér a szokványos irodai alkalmazásokétól. Emellett a nagyvállalati rendszer magas komplexitása szintén igényli, hogy a hallgató kellően elmélyüljön egy folyamat illetve az adott szervezet felépítésének részleteivel. Ez az DFI-rendszert tekintve túlzónak is mondható, amely rontja a rendszer átláthatóságát, ezzel nehezíti a vállalatirányítási rendszerek lényegének megértését, és így nehezen jön létre újrahasznosítható tudás. A túlzott komplexitás miatt egy-egy félévben a komplex feladatok végzése nehezen ütemezhető, így a tantárgy tematikájába nem fér bele a komplex hallgatói önálló munka. A gyakorlásban komoly hangsúlyt kell fektetni a rendezésekre, szűrésekre, keresési lehetőségekre és többértékű kiválasztásra, valamint ezek során a különböző szervezeti kódok alkalmazására.

SAP BUSINESS ONE 8.751-8.8

A legkönnyebb dolga mind az oktatóknak, mind a hallgatóknak az SAP Business One rendszerrel van. A telepítés, a karbantartás, az egyedi jelentések készítése és a programozás is rendkívül könnyű, az alkalmazás elsajátításáról nem is beszélve.

Az oktatórendszer telepítéséhez egy MS SQLServer Express is elegendő, mely kompakt méretével idő- és helytakarékos módon szinte felszalad egy átlagos színvonalú kisszerverre is. Ezután alig egy óra alatt elvégezhető a szervizmenedzser és a szerver telepítése, a kövér kliens pedig nem áll ellen a klónozásnak a kliens oldali telepítésnél. Probléma egyedül az Office integrációval és az alap Add-On-on telepítésénél lehet, de ezek hiánya nem okoz problémát az oktatásnál. A Mosaic Business System Kft. a PlasticWare Kft.-mintavállalatott bocsájtotta rendelkezésünkre, de a gyakorláshoz rendelkezésre áll a gyártó által konfigurált és mintaadatokkal feltöltött számítástechnikai kisvállalkozás is amerikai és magyar lokalizációval. A kezelőfelület nagyon hasonló az R/3 GUI-jához, de sokkal letisztultabb és könnyebb a hallgatónak az egyes kezelőelemeket a funkciójukhoz kötni. A kezelőfelület rendkívül barátságos és szinte mindegyik tranzakcióban egyformán néz ki. A könnyű kezelést támogatja, hogy gombnyomással átkapcsolható a lekérdező (kereső), illetve beszűrő üzemmód között. Szintén hasznos funkció az egy

adott folyamatelem rekordjának létrehozása forrásdokumentumból, illetve meglévő rekord adatainak felhasználása, másolása a folyamatelem következő lépésének dokumentálásához. Az „alapbizonylat” és „célbizonylat” gombok vizualizálják a betöltött tranzakciós bizonylat alapbizonylatának, illetve célbizonylatának meglétét és az aktuális űrlapba történő betöltését is el tudják végezni. További nagyon kedves barátunk a sárga nyilacska – a Navigátor –, amellyel az adott mezőben megjelenített azonosítóhoz kapcsolódó rekordhoz tudunk lefújni. Ezek a funkciók természetesen minden vállalatirányítási rendszerben megtalálhatók, de a SAP Business One rendszerben vannak a legkönnyebben és legkövetkezetesebben elhelyezve, illetve alkalmazva, lehetővé téve a villámgyors navigációt a táblák között és az egymáshoz kapcsolódó rekordok visszakeresést, és ezzel a hallgatók számára a vállalatirányítási rendszerek lényegének megértését. A rendezések, szűrések, keresési módszerek és többértékű kijelölések kissé sajátosak, de könnyedén elsajátíthatók. Az online help a help.sap.com-on pedig kiváló segítséget tud nyújtani az önálló tanulásban mind magyar, mind külföldi hallgatók számára, ugyanis számos nyelven elérhető az online help, pl. kínai, orosz, török, holland és finn nyelven is.

[6] Navision Support Online Kft. (2013): *Micro-soft Dyna-mics NAV*. Oktatási anyag. Budapest–Dunaújváros: DUF Press.

SAP ERP ECC6.00

A szolgáltatási jogosultság visszavonása miatt már csak ebben félévben tudjuk használni SAP egyik legmodernebb IDES mintavállalatot modellező oktatási rendszerét mind alkalmazás, üzemeltetés, testreszabás és fejlesztés oktatásának területén.

A tananyagok és a feladatsorok mySAP ERP Business Processes in Financial Accounting and Management Accounting, a Business Intelligence és az ERP Financials eredeti angol nyelvű tananyagok felhasználásával a Dunaújvárosi Főiskola Informatikai Intézetében készültek. Emellett az oktatáshoz rendelkezésre állt az All-in-One Best Practises csomag angol nyelven. A rendszer korlátozott angol nyelvű helppel segítette a munkát, mely kiegészült a help.sap.com-on elérhető angol és német nyelvű anyagokkal.

MS DYNAMICS NAV

A Microsoft rendszerével való ismerkedéskor már felhívták figyelmünket, hogy ez a rendszer alapvető sajátosságai miatt más rendszerektől eltérően nem könyveli le az egyes tételeket a számlatükör főkönyveire. Ezért ezt kézzel el kell végezni. [6]

[7] Louis Columbus: 2013 ERP Market Share Update: SAP Solidifies Market Leadership. A Passion for Research – Focusing on the intersection of technology and trust – Blog at WordPress.

Ezen nehézség rendkívül hasznos a könyvelés és a kontrolling megértéséhez. Ezért a Dynamics NAV oktatásában az első leckék között fog szerepelni a dimenzionálási és könyvelési kérdések tisztázása. A kezelőfelület a Microsoft alkalmazások képét viseli, mely kissé zavaró, ha valaki korábban más vállalatirányítási rendszerekkel és Microsoft Office alkalmazásokkal egyszerre dolgozott. De ugyanazon funkciók itt is megtalálhatók, tehát könnyedén lehet lefűjni, célbizonylatot létrehozni.

A folyamathoz tartozó bizonylatokon viszont nehezebb navigálni. Emellett teljeskörű termelésirányítási funkcionalitás is rendelkezésre áll, mely az SAP Business One-ból hiányzott. Így teljesen ki tudja váltani az SAP R/3-as rendszerét. A rendszerhez több adatbázis is létrehozható, így fejlesztő, teszt és éles oktató környezet, illetve több minta- és gyakorló vállalat is kialakítható. A programfejlesztés rendkívül egyszerű, viszont emiatt hosszadalmas is, mert csak alapszintű technikák állnak rendelkezésre az adatbázis eléréshez.

ZÁRÓ KÉRDÉS

Záró kérdésként tegyük fel, hogy visszalépésnek számít-e nagyvállalati ERP-rendszerről áttérni az MS Dynamics NAV-ra. A rendszer architektúrájának, a szervezeti struktúra leképezésének és a folyamat komplexitásának szempontjából mindenképp. Azonban tekintettel a főiskola anyagi helyzetére a költséghatékonyság és az oktatási cél és alkalmazhatóság szempontjából jó választásnak tűnik. Bár a Microsoft hírneve az ERP-k piacán jóval kisebb, mint a piacvezető SAP-é, és piaci részesedése 2012-ben az ötöde volt az SAP 25%-os részesedésének [7], Magyarországon jelenleg rohamosan terjed, elsősorban a kis- és középvállalkozások körében. Így talán szerencsés választásnak bizonyulhat hallgatóink számára.

A vállalkozó kedvű hallgatók számára pedig rendelkezésre áll a jó öreg DFI SAP R/3 4.6C. Eddigi tapasztalataink azt mutatták, hogy ha egy hallgató elsajátítja egy ERP-rendszer használatát, és megérti annak célját és felépítését, akkor könnyedén átáll bármilyen más típusú vagy méretű rendszer használatára. Nagyvállalatoknál többnyire a képzés során megismert funkcionális területeknél jóval szűkebb tartományban kell egy-egy adott munkakörben dolgozni és csak kis- és középvállalatoknál fordul elő, hogy több területtel foglalkozik ugyanazon dolgozó.

Apró megjegyzés: Mindegyik a rendszerben könnyen és gyorsan, rövid idő alatt átállítható a GUI angol nyelvre, így az angol nyelvű képzéseinkben résztvevő külföldi hallgatóink is oktathatók ezen rendszerek használatára.

Az ERP-oktatás hatékonyságának vizsgálata

Az elmúlt több mint egy évtizedben összesen mintegy 1359 külföldi és magyar hallgató ismerkedett vállalatirányítási rendszerekkel a Dunaújvárosi Főiskolán. Három (napjainkban kettő) intézete gondozásában mintegy 10 tantárgy valamelyikének keretében, melyeken résztvevők közelítő létszámát az 1. táblázat mutatja be. (A táblázat tartalmazza az elégtelen eredményű pótvizsgákat is.) A korábbi Közgazdaságtudományi és Vezetés- és Vállalkozástudományi Intézetek elsősorban üzleti és logisztikai, termelésirányítási alkalmazás oktatása céljából alakították ki a tantárgyi struktúrát, 2008-tól figyelembe véve a kis- és középvállalati igényeket is, az Informatikai Intézet pedig az üzemeltetési és fejlesztői területeket célozta meg.

1. táblázat. ERP Integrált vállalati rendszerekkel kapcsolatos tantárgyakat hallgatók létszáma(2000–01. – 2012–13. tanév).

Gondozó intézet/tanszék	Gazdálkodás- tudományi tanszék (korábban Közgazdaság- tudományi Intézet)		Vezetés- és Vállalkozás-tudományi Tanszék (korábban önálló intézet)				Informatikai Intézet				ÖSSZESEN
	SAP alkalmazás	SAP bázis ismeretek	Logisztikai információsrendszerek	SAP Business ONE	SAP logisztikai irányítási modul alkalmazási projekt	SAP termelésirányítás alkalmazási projekt	SAP alapjai	SAP üzemeltetése	Vállalatirányítási rendszerek A	Vállalatirányítási rendszerek B	
Gazdálkodási	285	285		1							571
Gazdálkodási és menedzsment				7							7
Gazdaságinformatikus							85	31	324		440
Logisztikai műszaki menedzserasszisztens			151	128							128
Mérnök-informatikus							67		549	317	933
Mérnök-tanár - Mérnök-informatikus									3		3
Műszaki informatikai mérnökasszisztens							26				26
Műszaki menedzser				93	163	26			7		289
ÖSSZESEN	285	285	151	229	163	26	178	31	883	317	2548

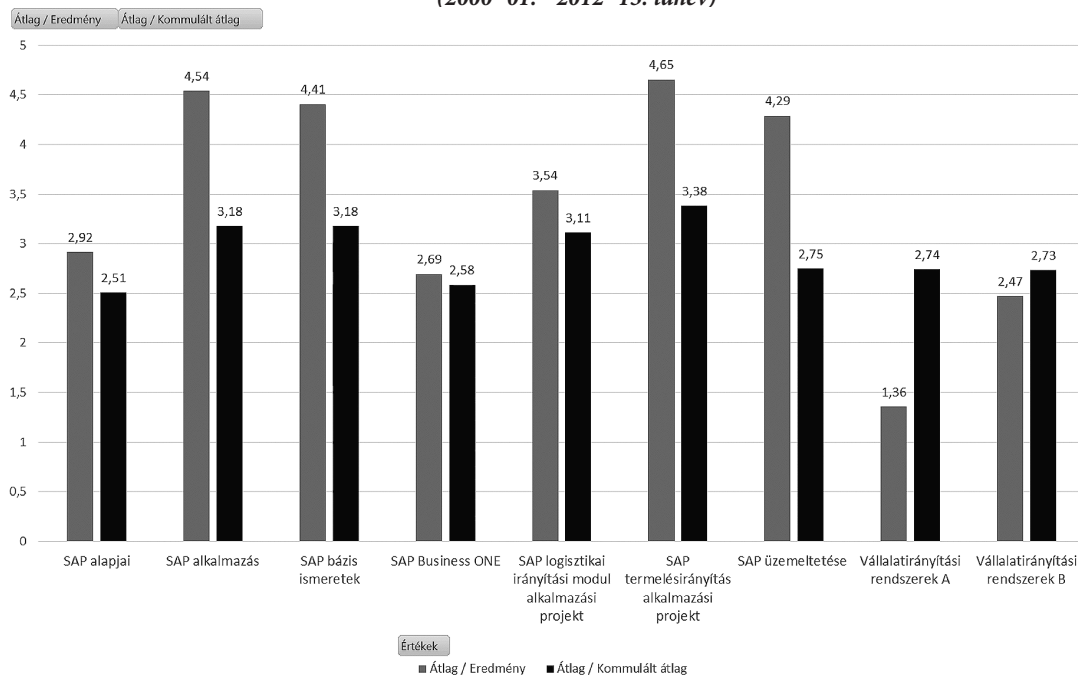
Forrás: DF Neptun Tanulmányi Rendszer adatai alapján saját szerkesztés

Az elemzéshez az integrált vállalatirányítási rendszerekkel foglalkozó tantárgyak a Neptun Tanulmányi Rendszerben elérhető hallgatói eredményeit és az érintett hallgatók teljes tanulmányi átlagát használtuk fel különböző bontásban. Utóbbival vizsgálni lehet, hogy melyik tantárgynál illetve oktatási rendszernél milyen volt a hallgatói eredmény a hallgató teljes tanulmányi átlagához képest, amellyel kiküszöböljük a hallgatói képességek és az aktuális és átlagos szorgalom eredményt befolyásoló hatását.

Megjegyzés: A *Logisztikai információrendszerek* tantárgy csak 30–40%-ában foglalkozott ERP-rendszerekkel, ezért ezen kurzusok eredményeit kihagytuk az elemzésből.

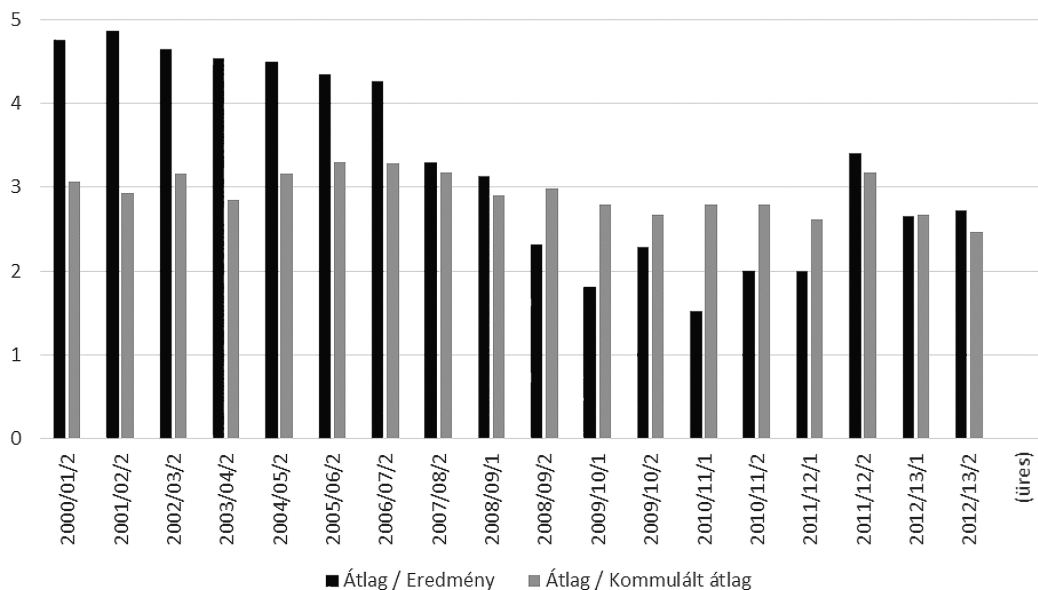
Ha áttekintjük 2001-ig visszamenőleg az érintett tárgyakat és eredményeit az 1. ábrán, akkor megfigyelhetjük, hogy az össz-tanulmányi átlaghoz képest általában jobban teljesítenek a hallgatók, annak ellenére, hogy a tárgyak a komplex szemléletmód miatt a nehezebben teljesíthetők közé tartoznak. Mindez köszönhető a tantárgyak közkedveltségénél és vállalati körökben való elismertségének.

1.ábra. Tantárgyi átlageredmények és a tanulmányi átlag alakulása az egyes ERP-tantárgyak esetében (2000–01. – 2012–13. tanév)



Forrás: DF Neptun Tanulmányi Rendszer adatai alapján saját szerkesztés

2. ábra. Féléves átlageredmények és a tanulmányi átlag alakulása ERP-tantárgyak esetében (2000–01.–2012–13. tanév)



Forrás: DF Neptun Tanulmányi Rendszer adatai alapján saját szerkesztés

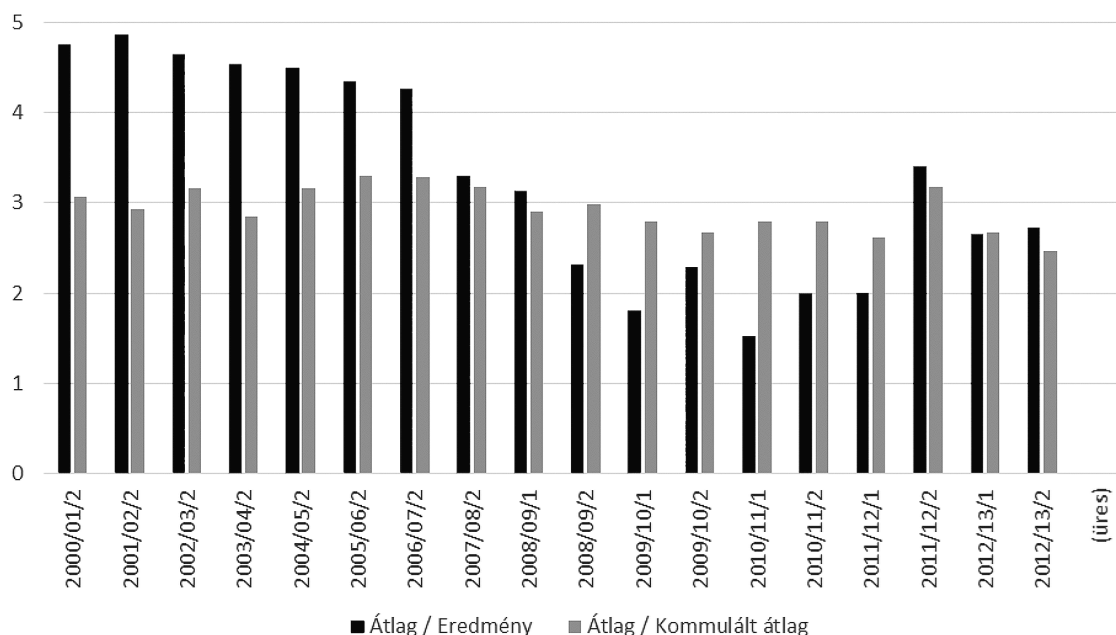
[8] Mosaic Business System Kft. (2010): SAP Business One. Oktatási anyag. Budapest.

Azonban ha a 2. ábra segítségével félévek szerint együttesen vizsgáljuk az eredmények alakulását, akkor csökkenő átlagot tapasztalhatunk a hallgatói eredményekben. Ennek okát kereshetnénk az egyes rendszerek közötti átállásokban, de a nagyobb változások elsősorban szakok indításához illetve átalakításához köthetők. Az első nagy ugrás (esés) a 2007–08. tanév II. félévében történt, ekkor indult el az ERP-oktatás a mérnök informatikus és a gazdaságinformatikus alapszakokon. Az esés újult erővel folytatódott egy évvel később. Ekkor indult el az ERP-képzés a Műszaki Menedzser alapszakon. Majd az utolsó nagy esés a 2010–11. tanév 1. félévében történt, amikortól a *Hatásfok 2010* program miatt a *Logisztikai műszaki menedzser* felsőfokú szakképzési szakon 1. féléves tantárgyként szerepelt az SAP Business One tantárgy 2012–13. tanév 1. félévig. Ezt ellensúlyozta az oktatási segédletek lecserélése, a Mosaic oktatási anyag helyett a képzéshez jobban illeszkedő előadásanyagok kidolgozása és bevezetése. [8]

Továbbá ha a 3. ábrán áttekintjük a tantárgyankénti féléves átlagok alakulását tantárgyak szerint csoportosítva, akkor megfigyelhetjük, hogy az informatikus képzésben szereplő *Vállalatirányítási rendszerek A és B* (VIR A és VIR B), valamint SAP-üzemeltetés tantárgyak rontják az átlagot, amely ezen tantárgyak komplexitását tekintve érthető. Ennek alapján jövőbeni fejlesztési irány lehet több tantárgyra történő felosztásuk, gyakorlati foglalkozásaik arányának növelése, mely jelentősen hozzájárulna az ismeretek mélyebb elsajátításához. Tanácsos lenne ezt megtenni a vállalati igények miatt is, hisz a kis- és középvállalatok körében mind Magyarországon, mind Európában rohamosan terjed a vállalatirányítási rendszerek alkalmazása. Magyarországon Uniós társfinanszírozású forrásra is lehet pályázni.

Ezzel tehát hallgatóink munkaerő-piaci versenyképessége jelentősen javulna, mivel költséghatékonyabb a vállalatok számára bizonyos fejlesztések, karbantartások belső munkatársakkal történő megvalósítása.

3. ábra. Tantárgyi átlageredmények és a tanulmányi átlag félévenkénti az egyes ERP-tantárgyak esetében (2000–01.–2013–14. tanév)



Forrás: DF Neptun Tanulmányi Rendszer adatai alapján saját szerkesztés

Jövőbeni fejlesztési célok

Az ERP-rendszerek integrált funkcionalitásának köszönhetően nemcsak a vállalatirányítási rendszerekkel foglalkozó tárgyaknál alkalmazható, hanem a pénzügyi, adózási, kontrolling, menedzsment és emberi erőforrás-gazdálkodás tantárgyaknál is. Az általános ERP-tantárgyaknál ezen funkcionális területek részleteire nem tudunk kitérni, azonban az említett tantárgyaknál az elméleti és gyakorlati oktatást a hallgatók számára is vonzó eszköz megismerésével és alkalmazásának elsajátításával egészítheti ki, teheti tartalmasabbá, segítve az elmélet rögzülését. Hosszú távon a Microsoft Dynamics NAV mellett az SAP oktatását is folytatni szeretnénk, de ehhez szükséges az SAP Hungary Kft licensz-politikájának rugalmasabbá és a magyar felsőoktatási intézmények anyagi lehetőségeihez jobban illeszkedővé tételére.

Összefoglalás

Összegzésül megállapíthatjuk, hogy a Dunaújvárosi Főiskola az elmúlt több mint tíz évben sok más felsőoktatási intézményhez hasonlóan élen járt az integrált vállaltirányítási rendszerek oktatásában, de anyagi lehetőségei jelentősen korlátozzák a megfelelő színvonalú nagyvállalati rendszer oktatásban történő alkalmazásában. Ezt a szintet a jövőben is meg kívánja tartani, és további szolgáltatói együttműködésekkel és tantárgyi fejlesztésekkel meg kívánja erősíteni.