

GÉP

A GÉPIPARI TUDOMÁNYOS EGYESÜLET

műszaki, vállalkozási, befektetési, értékesítési, kutatás-fejlesztési, piaci információs folyóirata

SZERKESZTŐBIZOTTSÁG

Dr. Döbröczeni Ádám

elnök

Vesza József

főszerkesztő

Dr. Jármái Károly

Dr. Péter József

Dr. Szabó Szilárd

főszerkesztő-helyettesek

Dr. Barkóczy István

Bányai Zoltán

Dr. Beke János

Dr. Bercsey Tibor

Dr. Bukoveczky György

Dr. Czitán Gábor

Dr. Danyi József

Dr. Dudás Illés

Dr. Gáti József

Dr. Horváth Sándor

Dr. Illés Béla

Kármán Antal

Dr. Kulcsár Béla

Dr. Kalmár Ferenc

Dr. Orbán Ferenc

Dr. Pálincás István

Dr. Patkó Gyula

Dr. Péter László

Dr. Penninger Antal

Dr. Rittinger János

Dr. Szabó István

Dr. Szántó Jenő

Dr. Tímár Imre

Dr. Tóth László

Dr. Varga Emilné Dr. Szűcs Edit

KEDVES OLVASÓ!

A Gép folyóirat 2012. 05. számában a Mechatronikai és Logisztikai Kiválósági Központ kutatási eredményei kerülnek publikációk formájában ismertetésre. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával, valamint a Miskolci Egyetem kutatást végző szervezeti egységei anyagi hozzájárulásával valósul meg. A kiválósági központot négy tudományos műhely alkotja, melyek kutatásai a mechatronika és logisztika tématerületei köré csoportosulnak. A mechatronika és a logisztikai szakterületek komplex kutatása és az eredmények ipari alkalmazása elengedhetetlennek mondható, mivel a magyar gazdaság fejlődési tendenciái és a magyar kormány gazdaságpolitikája is ezt teszi indokolttá (pl. autópálya dinamikus bővülése, regionális logisztikai központok kialakítása, stb.). Az intelligens rendszerek csak megfelelő gazdaságosság, kommunikáció, illetve jogi szabályozás esetén piac-képesek, ami indokolja a mechatronikai és logisztikai rendszerek ilyen szempontú vizsgálatát is.

A Mechatronikai rendszerek elemeinek kutatása és fejlesztése, a Logisztikai rendszerek hatékonyságnövelési eljárásainak, módszereinek kutatása, a Vezetékes és vezeték nélküli kommunikációs rendszerek megbízhatóságának növelése és az Innovatív megoldások a szervezetek irányításában a versenyképesség fokozására tudományos műhelyek alkotják a Mechatronikai és Logisztikai Kiválósági Központot. Az elmúlt 1 éves kutatási időszakban számos oktató, kutató és hallgató kapott lehetőséget eredményeinek neves hazai és nemzetközi konferenciákon való bemutatására. A központ elsődleges céljainak tekinti a fiatal oktatók, kutatók helyben tartását az ipari vállalatokkal való kapcsolatok kiépítését és az azokkal való közös kutatások megvalósítását. A Vezetékes és vezeték nélküli kommunikációs rendszerek megbízhatóságának növelése a logisztikai és mechatronikai alkalmazásoknál tudományos műhelyben az Automatizálási és Kommunikáció-technológiai Tanszék, az Analízis Tanszék, az Általános Informatikai Tanszék és az Alkalmazott Informatikai Tanszék kutatói dolgoznak együtt. A műhelyben az alábbi területeken folynak kutatások:

A vezetékes és önszervező vezeték nélküli szenzorhálózatok kommunikációs rendszereinek vizsgálata, modellezése, különös tekintettel az energiaoptimalás, hibafelfedés és az időkritikus működésre. Az irányításméret és az automaták elmélete. Az ontológia alapú, tény és nyelvtan reprezentációs módszerek továbbfejlesztése. A statisztikai és lágy számítási elveken alapuló parancsfeldolgozó felületek kifejlesztése. Gyártásinformatikai, matematikai modellek továbbfejlesztése. Szuperszámítógépen implementált új ütemező algoritmusok. Gyártás- és logisztikai szimuláció ipari alkalmazása.

Ez a cikkgyűjtemény a kiválósági központ elmúlt 1 éves kutatási munkájának egy fontos részét ismerteti.

Prof. Dr. Illés Béla
egyetemi tanár,
központvezető

Dr. Czap László
egyetemi docens,
Tudományos Műhelyvezető

A szerkesztésért felelős: Vesza József. A szerkesztőség címe: 3534 Miskolc, Szervezet utca 67.

Telefon/fax: +36-46/379-530, +36-30/9-450-270 • e-mail: mail@gepujsag.hu

Kiadja a Gépipari Tudományos Egyesület, 1027 Budapest, Fő u. 68. Levélcím: 1371 Bp. Pf.: 433.

Telefon: 202-0656, fax: 202-0252, e-mail: a.gaby@gteportal.eu, internet: www.gte.mtesz.hu

A GÉP folyóirat internetcíme: <http://www.gepujsag.hu>

Kereskedelmi és Hitelbank: 10200830-32310236-00000000

Felelős kiadó: Dr. Igaz Jenő ügyvezető igazgató.

Gazdász Nyomda Kft. 3534 Miskolc, Szervezet u. 67. Tel.: (46) 379-530, e-mail: gazdasz@chello.hu.

Előfizetésben terjeszti a Magyar Posta Rt. Hírlap Üzletága 1008 Budapest, Orczy tér 1.

Előfizethető valamennyi postán, kézbesítőknél, e-mailen: hirlapelofizetes@posta.hu, faxon: 303-3440. További információ: 06 80/444-444

Egy szám ára: 1260 Ft. Dupla szám ára: 2520 Ft.

Külföldön terjeszti a Kultúra Könyv és Hírlap Külkereskedelmi Vállalat,
H-1389 Budapest, Pf. 149. és a Magyar Média, H-1392 Budapest, Pf. 272.

Előfizethető még közvetlenül a szerkesztőségben is.

INDEX: 25 343 ISSN 0016-8572

A megjelent cikkek lektoráltak.

A kiadvány a Nemzeti Kulturális Alap támogatásával jelenik meg.

TARTALOM

1. TÓTH T.: Új módszerek diszkrét technológiai folyamatok számítógépes tervezésében.....	3	16. SZILÁGYI SZ.: Rendezés-kongruenciák alkalmazása többgépes párhuzamos ütemezések esetén.....	67
2. TURÁN, L. OVSENIK, M. TURÁN, VÁSÁRHELYI J.: Légköri paraméterek minőségi monitorozási rendszerének tervezése és mérése.....	11	17. BARABÁS P.: Parancskinyerés magyar nyelvű szövegből.....	71
3. TROHÁK A.: Terheléselosztási algoritmusok tesztelésére alkalmas rendszer tervezése.....	15	18. KRIZSÁN Z.: Közös fejlesztői keretrendszer a Fuzzy Szabály Interpolációs módszerekhez.....	75
4. PINTÉR J., TROHÁK A.: Függősínpálya hangvezérlésének kifejlesztése.....	19	19. KOVÁCS L.: Fogalomháló alapú osztályozási módszerek.....	79
5. RÁDI P., TROHÁK A.: IWLAN ipari kommunikációs rendszerek tervezése, szimulációja, vizsgálata.....	23	20. BEDNARIK L., DR. KOVÁCS L.: Osztályozási feladatok a kérdésgenerálási mintarendszerben.....	83
6. MÉHES L., TROHÁK A.: Egyedi tervezésű telemetriai rendszer fejlesztése Bosch elektromobil járműhöz, ZigBee vezeték nélküli hálózaton.....	27	21. KOVÁCS SZ.: A „Double-Linear” Fuzzy Interpolációs Módszer kiterjesztése a fuzziság interpolációjával.....	87
7. BIRÓ Z. *, KOLOZSI-TÓTH M., TROHÁK A.: Autóbuszok távdiagnosztizálása gsm hálózaton keresztül.....	31	22. KOVÁCS SZ.: Fuzzy Szabály-interpolációs módszerek beágyazott irányítási alkalmazásokban.....	91
8. VARGA A. K.: Lokalizációs algoritmusok önszerveződő vezeték nélküli szeszor hálózatokban.....	35	23. MILEFF P., DUDRA J.: Hatékony technikák és megoldások a szoftveres raszterizációban.....	95
9. VARGA A. K.: A Miskolci Egyetem Programháló felületének fejlesztése.....	39	24. TÓTH ZS.: Formális nyelvtani modul a meta keretrendszer számára.....	99
10. TROHÁK A., HALUSKA ZS.: Vezeték nélküli ipari kommunikációs rendszerek RF spektrumának mérése.....	43	25. VINCZE D.: Egy újszerű hozzáférés szabályozási módszer a linux operációs rendszer kernelében.....	103
11. DR. CZAP L., PINTÉR J.: Hangvezérelt logisztikai rendszerek.....	47	26. WAGNER GY., TÓTH T.: (Helyi) hálózatok aktuális biztonsági problémái.....	107
12. FERENCZI I.: Beágyazott telegramokat tartalmazó ipari ethernet keretek ciklusideje.....	51	27. DR. KULCSÁR GY., DR. BIKFALVI P.: Termelésstervezési és -irányítási feladatok megoldása többcélú keresési módszer alkalmazásával.....	111
13. BOGNÁR G., ROZGONYI E.: Nemnewtoni folyadék kényszeráramlása síklapon porózus közegben.....	55	28. KÖREI A., TÓTH T.: Gyártócellák kialakítása metaheurisztikus módszerek alkalmazásával.....	115
14. VARGA P.: A Yang-Baxter egyenlet néhány megoldásának grafikus reprezentációja.....	59	29. HORNYÁK O., NEHÉZ K.: Felhő technológia a kis és közepes vállalatoknál.....	119
15. VARGA P.: Néhány általánosított Toda rács módosított Lax reprezentációja.....	63	30. ERDÉLYI F., TÓTH T.: A számítógépes folyamattervezés és a műhelyszintű gyártásirányítás egy integrációs problémája.....	123

ÚJ MÓDSZEREK DISZKRÉT TECHNOLÓGIAI FOLYAMATOK SZÁMÍTÓGÉPES TERVEZÉSÉBEN

NEW METHODS FOR COMPUTER AIDED PLANNING OF DISCRETE TECHNOLOGY PROCESSES

Tóth Tibor*

ABSTRACT

Computer Aided Process Planning (CAPP) is of growing importance, because it has central place and role between Computer Aided Design (CAD), Production Planning and Scheduling (PPS) and Computer Aided Manufacturing (CAM). The paper gives a brief summary about the historical background of CAPP developments and outlines the new trends too. Having summarized the traditional structure of CAPP systems, the paper proposes a new structure consisting of 3x3 subsystems for CAPP which gives a promising possibility for a well-established distribution of engineering knowledge and for solving optimization tasks at any hierarchy levels. The paper also emphasizes that the material removal rate (MRR) as a new variable is very advantageous from the point of view of optimization.

A diszkrét technológiai folyamatok számítógépes tervezése (CAPP) növekvő jelentőségű, mivel központi szerepe van a számítógépes konstrukciós tervezés (CAD), a termelés-tervezés- és ütemezés (PPS), valamint a számítógéppel segített gyártás (CAM) CIM rendszerbe való integrálásában. A cikk tömör összefoglalást ad a CAPP fejlesztések történelmi háttéréről és az új trendeket is körvonalazza. A CAPP rendszerek hagyományos struktúrájának ismertetését követően a cikk egy 3x3 alrendszerből felépülő új struktúrát javasol, amely lehetőséget nyújt a mérnöki tudásnak és az optimalizálási feladatoknak az egyes hierarchiai szintekhez való célszerű hozzárendeléséhez. A cikk hangsúlyozza az anyagleválasztási intenzitás (material removal rate, MRR) előnyeit a technológiai folyamatok optimalizálása szempontjából.

1. BEVEZETÉS

A gyártmánytervezés és a gyártás a termékfejlesztési folyamat kritikus fázisa. Az összekötő láncszem a gyártmánytervezés és a gyártás között a technológiai folyamat tervezése, röviden: a folyamattervezés.

*DSc, egyetemi tanár, Miskolci Egyetem Alkalmazott Informatikai Tanszék

A folyamattervezés során választják meg a szükséges gyártási folyamatokat és határozzák meg azok sorrendjét oly módon, hogy a gyártmánytervező elképzeléseit (a megtervezett alkatrészt) gazdaságosan és versenyképesen fizikai alkatrészre transzformálják. A forgácsoló megmunkálások területén a főbb folyamattervezési tevékenységek a következők: a gyártmánytervezési adatok értelmezése, a megmunkálási műveletek, szerzőgépek, felfogó készülékek, forgácsoló szerszámok, technológiai adatok megválasztása, valamint a költség és gyártási idő számítása. Nem kétséges, hogy ez nagyon komplex feladat.

A gyártási tevékenységet folytató vállalatoknál a folyamattervezési problémák hagyományos megközelítése az, hogy megoldásukat rábizzák a gyártási szakemberekre. Ezek a szakemberek tapasztalataikat és tudásukat felhasználva adnak utasításokat a termékek gyártásához a gyártmánytervezési specifikációk és a hozzáférhető erőforrások alapján. A különböző folyamattervezők gyakran eltérő terveket javasolnak ugyanarra a problémára, ezáltal a már addig is bonyolult probléma megoldását tovább nehezítik.

A nemzetközi szakirodalom tanúsága szerint a diszkrét technológiai folyamatok tervezésének számítógépes támogatására alkalmas első átfogó rendszer tervét és a megvalósításához szükséges algoritmusok néhány lehetséges változatát a volt Szovjetunióban *V.D. Cvetkov* és munkatársai dolgozták ki az 1960-as években [1]. Az első, bár lényegesen szerényebb nyugati kezdeményezés *Niebel*-től származik [2]. Az elmúlt közel 50 évre visszatekintve megállapítható, hogy a számítógéppel segített gyártmánytervezés (CAD) és a számítógéppel segített gyártás (CAM) fejlődéséhez képest a számítógéppel segített folyamattervezés állandó lemaradásban van a gyártóipar számára nyújtott megoldások gyakorlati hasznossága, érettsége, szakmai megalapozottsága és kereskedelmi forgalmazhatósága tekintetében. Mindez azonban nem a kutatási erőfeszítések hiányából adódik. Ellenkezőleg, a kutatások és a kapcsolódó publikációk jelentős ráfordításokat és folyamatos érdeklődést tükröznek.

Általában két fő megközelítést különböztetnek meg a CAPP területén, amelyeket a variáns-módszer, illetve a generatív módszer elnevezéssel tárgyal a szakirodalom. A

variáns-módszer azt az elvet követi, hogy hasonló alkatrészek hasonló terveket igényelnek. A módszer alapváltozata szerint egy technológus szakember a megfelelő osztályba (csoportba, típusba) sorolja az alkatrészt, megadja az alkatrész bemenő adatait, visszakeres az adatbázisból egy hasonló folyamattervet és végrehajtja a szükséges módosításokat. Ez a megközelítési mód olyan vállalatok igényeihez illeszkedik, amelyek stabil gyártási folyamatokkal és kis változatosságú termékkálával rendelkeznek. Előnye a kényelmes kezelhetőség, hátrányai: a manuális input nagy mennyiségű adat megadását teszi szükségessé a gyártási folyamatról; a folyamatterv minősége nagymértékben a folyamattervező szakmai felkészültségétől függ; nincs lehetőség a gyártási folyamattal kapcsolatos menet közbeni számításokra. A **generatív módszer** esetében a folyamatterveket kis mértékű emberi beavatkozással hozzák létre. Az új folyamattervek generálása a döntési logika és a folyamatra vonatkozó tudás alapján megy végbe. A korábbi fejlesztések speciális leíró nyelveket és a programlogikába épített technológiai tudás elvét alkalmazták. A későbbi fejlesztések (az 1980-as évek közepe óta) a sajátosságok (features) alkalmazását helyezik a középpontba, termékmodell definiálásának igényével. A módszer szűk keresztmetszete a használható sajátosságok kinyerésének nehézségében, valamint az emberi szakértelem reprezentálásának, kezelésének és kihasználásának nehézségében van. Ez a fő oka annak, hogy mind a sajátosság-alapú technológia, mind a tudás-alapú technikák nehezen épülnek be a CAPP-kutatásokba.

Egy hatékony CAPP-rendszertől elvárható néhány fontos követelmény:

- Legyen kölcsönkapcsolatban a felette és alatta elhelyezkedő tevékenység-szintekkel, vagyis a gyártmánytervezéssel és a gyártással, oly módon, hogy a CAPP-rendszer tudja fogadni a gyártmánytervezési adatokat változatlan formában, és olyan kimenetet generáljon, amely bevihető a CAM-rendszerbe, majd később egy CNC rendszerbe;
- Legyen kiterjeszhető, adaptálható és igény szerint illeszthető egyedi vállalatokra és új folyamatokra;
- Rendelkezzen a tudás kinyerésének, ábrázolásának és kezelésének hatékony mechanizmusaival, valamint a tudás teljességének és következetességének ellenőrzésére szolgáló eszközökkel;
- Vonja be a felhasználókat a döntéshozatali folyamat bizonyos részeibe, szolgáltatson heurisztikákat, ha szükséges, és egészítse ki a rendszer képességeit;
- Rendelkezzen egy felhasználó-barát interfésszel, amely hatékony interaktív kapcsolatot tesz lehetővé a rendszer és a felhasználó között (bemeneti, kimeneti és riport funkciók, valamint az eredmények grafikus megjelenítése).

A CAPP kutatásokról és fejlesztésekről az elmúlt évtizedekben nagyszámú publikáció és néhány átfogó, elemző tanulmány jelent meg a szakirodalomban. A következőkben kiemelünk néhány, a legfontosabb eredményeket ismertető forrásmunkát, amelyek alapján képet alkothatunk a CAPP fejlődéstörténetéről.

2. A CAPP FEJLŐDÉSTÖRTÉNETE

Az első, nemzetközi visszhangot keltő CAPP rendszert 1976-ban fejlesztették ki a Computer Aided Manufacturing International (CAM-I) támogatásával [3]. Azóta, tekintettel a CAPP területén folyó intenzív kutatásokra, több, a mindenkori helyzetképet tükröző, átfogó publikáció is született.

Az egyik első áttekintő publikációt *Stuedel* jelentette meg 1984-ben [4]. A cikkben a szerző a gyártási módszerek strukturálásához alkalmas megközelítéseket és stratégiákat, valamint egy generatív típusú automatizált folyamattervező rendszerhez szükséges adatokat tárgyalja. Ez a cikk körvonalaz egy korábbi fejlesztést is egy közös geometriai leíró nyelvre vonatkozóan, amely kapcsolatot létesít az alkatrész és a folyamat között és beszámol olyan CAD/CAM rendszerekről, amelyek CAPP funkciót is magukban foglalnak. A következő évben *Eversheim* és *Schulz* publikált egy elemző áttekintést azoknak a kérdőíveknek a felhasználásával, amelyeket 1983-ban és 1984-ben a CAPP fejlesztők és végfelhasználók részére küldtek ki Európában, Észak-Amerikában és néhány ázsiai országban [5]. Az áttekintésből nyilvánvaló, hogy a CAPP fejlesztések és alkalmazások akkor viszonylag újak voltak. 1988-ban *Ham* és *Lu* publikált egy értékelést a CAPP akkori helyzetéről, és helyesen állapította meg, hogy a jövőbeni kutatás főiránya a CAPP integrálása a gyártmánytervezés és a gyártás közé, továbbá a mesterséges intelligencia-módszerek (Artificial Intelligence Technologies) használata [6]. Ezt követően alighanem *Alting* és *Zhang* adta a legjelentősebb áttekintést 1989-ben. Tanulmányukban több mint 200 publikált munkát elemeztek és 14 jól ismert CAPP rendszert értékelték [7]. 156 létező rendszert soroltak fel táblázatos formában, közöttük a magyar fejlesztésű TAUPROG-T is szerepel [8, 9, 10]. Az áttekintés rámutat a CAD és a CAPP integrálásában megmutatkozó nehézségekre, amelyek fő oka a geometriai entitások reprezentálására alkalmas közös módszerek hiánya. A szerzők javasolták az interfész kérdések erőteljesebb kutatását is, egyrészt a CAPP és CAM között, másrészt más számítógépes rendszerekkel, mint NC szerszámpálya generáló rendszer, MRP (Manufacturing Resources Planning), termelés-szimuláció. Felismerték, hogy az AI technológiákat egy hatékony folyamattervező rendszer létrehozásakor kulcsfontosságú modellezési, algoritmizálási és kommunikációs feladatokra lehet felhasználni. Ezen túlmenően kiemelték a tanulórendszer fontosságát és egy ideális megközelítésként azt jelölték meg, hogy integrálni kell az adott alkatrész előállításához szükséges összes információt egyetlen (single) adatbázisban. Ugyanabban az évben *Gouda* és *Taraman*(1989) publikált egy összefoglaló áttekintést 128 CAPP rendszerről [11]. Munkájukban 4 CAPP rendszer típust emeltek ki részletesebben, ezek: a variáns elvű, a szemi-generatív elvű, a generatív elvű és a szakértő folyamattervező rendszer.

A következő, nagy lélegzetű áttekintést a CIRP nemzetközi szervezet CAPP Working Group nevű szakcsoportja állította össze 1993-ban [12] (*ElMaraghy*, et al.). Ez az

összefoglaló cikk a CAPP fejlesztési főirányával, a CAPP ipari perspektíváival, a fejlődési trendekkel és kihívásokkal, valamint a konstrukciós tervezés, a CAPP és a termelés-tervezés integrációjával foglalkozik. Kitérnek a minőség kérdéseire és a kapcsolódó szabványosítás igényeire. Ugyanebben az évben *Eversheim* és *Schneewind* megjelentetett egy széles területet átfogó, de nagyon tömör ismertetést a CAPP valódi ipari környezetben való alkalmazhatóságáról [13]. A szélesebb körű alkalmazás elősegítése végett javasolták, hogy a jövőbeni CAPP fejlesztések terjedjenek ki a szerelés tervezésére, az NC programozással való funkcionális integrációra, az AI eszközöknek a döntéshozatalban való felhasználására és az adatbázis megosztás használatára a CAD-vel való integrációhoz. 1995-ben *Kamrani et al.* bemutattak egy áttekintést az alkalmazható CAPP technikákról és a folyamat-tervezés szerepéről [14]. A cikkben tárgyalják egy CAPP rendszer kiválasztásához és értékeléséhez kapcsolódó kritikus kérdéseket és jellemzőket is. Ezek a kérdések a következők: a termék támogatásának mértéke, osztályozás /kódolás/grafikus képességek, munkautasítás létrehozása, folyamat-tervezési megközelítési módok, időelemzési képességek, megmunkálási paraméterek, megmunkálendő anyagokra és a szerkesztésre vonatkozó adatbázis, rendszer-technikai követelmények, költségek, kereskedelmi hozzáférhetőség és felhasználóbarát képességek, a vásárló/felhasználó minősítése és támogatása.

A következő átfogó ismertetés 1996-ban készült és szerzője *Leung* [15]. A szerző referáló ismertetést adott a CAPP területén 1989 és 1996 között megjelent 200 publikációról. *Leung* megfigyelte, hogy a testmodellelés (solid modelling) nem bizonyult olyan megfelelőnek, mint azt előre jelezték, ezért kapott új erőre a variáns elvű folyamat-tervező rendszerek fejlesztése és alkalmazása. *Leung* szerint logikailag teljesen indokolt, hogy a jövőbeni folyamat-tervező rendszerek intelligens rendszer architektúrára épüljenek AI technika felhasználásával. *Leung*-ot követően jelent meg *Cay* és *Chassapis* áttekintő cikke (1997), amely az 1990 és 1997 között folytatott kutatásokról számol be [16]. Összefoglaló értékelést ad a gyártási sajátosságokról és a sajátosság-felismerő technikákról a CAPP kutatások területén. *Cay* véleménye az, hogy egy teljesen automatizált környezet nem lenne messze a realitástól, ha elérhető lenne egy hatékony integráció a konstrukciós tervezés és a gyártás között. Az utolsó általános ismertetést a CAPP-ról az 1990-es években *Marri et al.* készítették [17] (1998). Ez az áttekintés az 1989 és 1996 között megjelent fontosabb publikációkat dolgozza fel. A CAPP rendszerek előnyeit és hátrányait az összefoglaló cikk elsősorban a generatív megközelítési mód szemszögéből tárgyalja.

Az általános célú CAPP áttekintések mellett egyéb, specifikus területen is készültek ilyen jellegű munkák, például a CAD-re és a gyártási sajátosság bázisú folyamat-tervezésre (*Shah*, 1991 [18]; *Shah et al.*, 1991 [19]), a neurális háló alapú folyamat-tervezésre (*Yue et al.*, 2002 [20]), szakértőrendszer-bázisú folyamat-tervezésre (*Gupta and Ghosh* 1989 [21], *Kiritsis* 1995 [22], *Metaxiotis et al.* 2002 [23], *Liao* 2005 [24]) és a virtuá-

lis valóság bázisú folyamat-tervezésre (*Peng et al.* 2000 [25]) vonatkozóan. Még újabban, *Shen et al.* (2006 [26, 27]) publikált egy átfogó (state-of-the-art típusú) cikket az ágens-alapú, megosztott folyamat-tervezésről és ütemezésről. *Shen* és szerzőtársai a gyártási folyamat-tervezési és ütemezési problémák komplexitását vizsgálták és áttekintették a folyamat-tervezés, az ütemezés és integrációjuk szakirodalmát az ágens-alapú megközelítésekre helyezve az elemzés súlypontját. *Zhang* és *Xie* 2007-ben publikált egy elemző-értékelő cikket az ún. együttműködő (collaborative) folyamat-tervezéshez kifejlesztett ágens-alapú technológiáról. Céljuk az volt, hogy megvizsgálják az ágens-alapú folyamat-tervező rendszerek fejlesztésének kulcs-kérdéseit, beleértve az ágens- és rendszer architektúrát, a kommunikációs szabványokat és protokollokat, valamint az alkalmazásokat.

3. JELENLEGI HELYZETKÉP A CAPP-RÓL

A legújabb és talán az eddigi legátfogóbb elemzést *Xun Xu*, *Likui Wang* és *Stephen T. Newman* munkája adja, amely az újabb fejlesztéseket kritikailag vizsgálja és megkísérli a jövőbeni trendek meghatározását [29]. A szokatlanul terjedelmes, (31 oldalas) folyóiratcikk a CAPP területén publikált többszáz publikáció áttekintésével és az „Elsevier Scopus absztrakt – és idézettségi adatbázisának felhasználásával meggyőző hitelességű képet ad a legnagyobb érdeklődésre számot tartó területekről és von le következtetéseket a várható fejlesztési trendekre vonatkozóan.

A *Xu*, *Wang* és *Newman* a jelenlegi CAPP fejlesztéseket az elméleti megközelítési módszerek alapján az alábbi kategóriákba sorolja [29]:

- sajátosság-alapú technológiák (feature-based technologies);
- tudásbázisú rendszerek (knowledge-based systems);
- mesterséges neurális hálók (artificial neural networks);
- genetikus algoritmusok (genetic algorithms);
- fuzzy halmazok elmélete és fuzzy logika (fuzzy set theory and fuzzy logic);
- Petri-hálók (Petri-nets);
- ágens-alapú technológia (agent-based technology);
- Internet-alapú technológia (Internet-based technology);
- STEP-támogatású CAPP (STEP-compliant CAPP);
- speciális (egyedi) technológiák.

A cikk a felsorolt kategóriákat követve tárgyalja az alkalmazott – számos tekintetben nagyon eltérő – módszereket. Aki ma bármilyen CAPP fejlesztést elindít, aligha nélkülözheti a kiváló szerző-hármas cikkének alapos tanulmányozását.

4. ÚJ HAZAI MÓDSZEREK A CAPP TERÜLETÉN

A következőkben röviden szólnunk a CAPP szakterületén elért néhány újabb hazai eredményről. Megjegyezzük, hogy a CAPP területén Magyarországon több mint

negyven éve folynak nemzetközi szintű K+F munkák, amelyekben a jogutód nélkül megszűnt Gépipari Technológiai Intézet (GTI), a Miskolci Egyetem (ME), a Budapesti Műszaki és Gazdaságtudományi Egyetem (BME) és az MTA Számítástechnikai és Automatizálási Kutatóintézet (MTA SZTAKI) jelentős eredményeket ért el. Ez a négy intézmény több évtizeden keresztül számos közös projektben dolgozott együtt és a jelenleg is működő három kutatóhely a termelésinformatika tudomány-területének részeként ma is intenzíven foglalkozik ezzel a szakterülettel. Terjedelmi okokból a [30] és [31] munkákra hivatkozunk, amelyek nagyon részletesen összefoglalják a CAPP hazai kutatásának és fejlesztésének legfontosabb eredményeit nagyszámú forrás elemző-értékelő felhasználásával.

Ebben a cikkben csak arra van mód, hogy a CAPP rendszerek más CAX rendszerekkel való integrálásáról és az ún. „intenzitás-bázisú” optimalizálásról röviden szóljunk. Mindkét részterületen, nemzetközi szinten elismert, több évtizedes kutatási-fejlesztési tapasztalattal és nagyszámú publikációval alátámasztott, sajátosan magyar megközelítésről van szó.

4.1. Egy új CAPP-struktúra CIM környezetben

Mint ismert, a számítógéppel integrált gyártás (*Computer Integrated Manufacturing*, CIM) egyike a legfontosabb termelési paradigmáknak és a múlt század hetvenes éveiben tűnt fel a szakirodalomban [31]. A CIM fókuszában az anyag- és adatfeldolgozás integrációja áll, amelyet számítógépes hálózati infra-struktúra támogat. Ez a paradigma radikális szakítást eredményezett a Taylor és Ford által lefektetett korábbi elvekkel és módszerekkel, valamint az izolált mérnöki funkciókkal. Összhangban a modern információ-elmélettel, az integrált rendszerben a mérnöki funkciók műszaki és szervezési szinten is összekapcsolódva megvalósítják az integráció *szemantikai* szintje lehetővé teszi, hogy a rendszerek ugyanazokat az absztrakt információ-osztályokat használják. Az integráció megteremti a közös cél elérése érdekében az üzleti és termelési folyamatok irányításának összhangját, amely megfelel az információ *pragmatikai* szintjének.

A technológiai tervezés mai módszertana – a számítógépes támogatás mértékétől függetlenül – egységesnek mutatkozik abban, hogy a gépgyártási folyamatok tervezését hierarchiai szintek szerint tagolva, felülről-lefelé haladva kell végrehajtani. A „felülről-lefelé” egyfelől azt jelenti, hogy az összetettebb, nagyobb tervezési egységektől az egyszerűbb, kisebb egységek irányába, másfelől a kevésbé részletezett folyamatszakaszoktól az egyre részletezettebb folyamatszakaszok irányába kell haladnia a tervezésnek. A tervezési folyamatra tehát mindvégig a fokozatos megközelítés jellemző, a tervezés eredménye (outputja) ugyanakkor rendkívül eltérő lehet a tervezés tárgyától, a gyártási körülményektől, a kitzűzött céloktól és egyéb tényezőktől függően.

Nincs teljesen egységes álláspont viszont a hierarchiai szintek célszerű számát és az egyes szintekhez hozzárendelhető feladatok tartalmát illetően.

Az egyik álláspont szerint a gépgyártástechnológiai folyamatok tervezésének legfelső hierarchiai szintje a technológiai előtervezés és a további tervezési fázisok három autonóm területe: a szerelés, az előgyártmány gyártás és az alkatrészgyártás technológiai folyamatának tervezése. Az alkatrészgyártás technológiai folyamatainak tervezését – felülről-lefelé haladva a hierarchiában – ez a koncepció öt szintre tagolja:

- előtervezés
- műveleti sorrendtervezés
- művelettervezés
- műveletelemek tervezése
- utófeldolgozás (posztprocesszálás).

A technológiai folyamat előtervezése ebben a hierarchiában különleges helyet foglal el. Az összes tervezési tevékenység közül ez a legnagyobb hatókörű, amely a gyártmány szerkezet lebontása (gyártmány → szerelési egység → részegység → szerelvény → alkatrészcsoport → egyedi alkatrész) során nyert valamennyi „helyben gyártandó” alkatrész potenciális gyártási lehetőségeit és környezetét feltérképezi, ezáltal a teljes gyártás koncepcióját, stratégiáját meghatározza. Ezen a szinten két alapvető döntéssorozat hajtódik végre. Az egyik döntéssorozat a gyártás főbb szakaszai – az előgyártmány gyártás, az alkatrészgyártás és a szerelés – közötti csatlakozási felületek meghatározására, a másik a felsorolt szakaszokat megvalósító gyáregységek, üzemek, gyártósorok, gyártórendszerek kijelölésére irányul. Az előtervezés során választják meg az előgyártmányok típusait, alakítják ki az egyes alkatrészek szerelés előtti állapotjellemzőit, itt történik meg a szerelési és gyártási méretláncok meghatározása, a gyártórendszerek „előválasztása” és ezzel együtt a gyártási folyamat elvi vázlatának (eljárások, berendezések) rögzítése. Az előtervezéshez csatlakoznak a szerelés, az előgyártmánygyártás és az alkatrészgyártás konkrét környezetre és feltételekre aktualizált tervezésének legfelső szintjei.

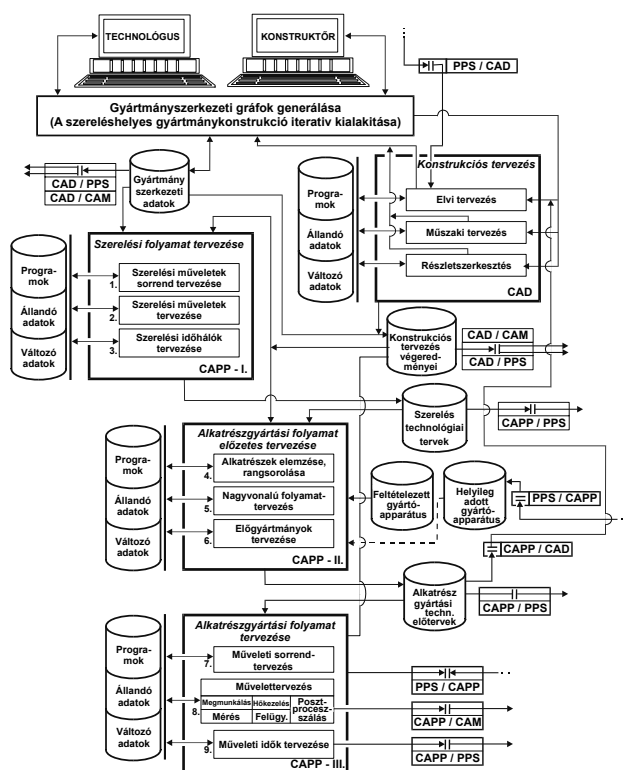
A vázolt *ötlépcsős* tervezési rendszer koncepciója – a legfelső, technológiai előtervezési szint kivételével – kiforrottnak és elméletileg jól megalapozottnak tekinthető, így a CAPP-rendszerek kifejlesztéséhez vagy utólagos elemzéséhez egyaránt jól felhasználható. A legfelső szint tartalma és a megoldandó feladatok megközelítési módja ma még több szempontból vitatott. Alapvető elvi különbség az alacsonyabb tervezési szintekhez képest egyfelől az, hogy az előtervezés nem csupán az adott termék egyetlen alkatrészére, hanem a terméknek az adott gyárban (üzemben, gyártórendszerben) gyártandó összes alkatrészére, az előgyártmányokra és a szerelésre is vonatkozik, másfelől egy magas szinten automatizált, igényes előtervező rendszernek közvetlen kapcsolatot kell teremtenie a konstrukciós tervezés, a technológiai tervezés és a termelésirányítás számítógépes rendszerei között.

Horváth Máttyás az előtervezés automatizálására olyan, vegyes felépítésű rendszert javasol [32], amelyben a variáns-, a generatív- és a vario-generatív módszerek, valamint a mesterséges intelligencia módszerei egyaránt helyet kapnának. A javaslat szerint szakértőrendszer szükséges a szerelési családfa, a szerelési vázlat és a méretlánc megoldások meghatározásához, a generatív módszer alkalmas a méretláncok elemzésére, az előgyártmányok meghatározására és az alkatrészek szerelés előtti állapotainak megtervezésére. Vario-generatív módszert ajánl a gyártórendszerek kiválasztásához, végül a variáns módszert javasolja az előgyártmányok típusának, valamint az előgyártási, alkatrészgyártási és a szerelési folyamat elvi vázlatának (a gyártási szakaszoknak és sorrendjüknek) meghatározására.

Ez a koncepció az egyes tervezési feladatokhoz elvben olyan megoldási eszközöket rendel hozzá, amelyek tervezésmethodikái szempontból jól megfelelnek azok sajátosságainak. A tényleges megvalósításhoz azonban ma még jórészt hiányoznak a feltételek, mindenképp a megfelelően általános, ugyanakkor kellően részletes gyármodellek és a szakértő rendszerek ún. meta-algoritmusai. Szerényebb átmeneti megoldásként Horváth Máttyás a variáns módszer kizárólagos alkalmazását ajánlja [32]. Ennek lényege az, hogy az aktuális alkatrészt egyszerű osztályozórendszerrel azonosítják, az ún. gyártábla segítségével ahhoz gyártórendszereket, valamint normatív költség- és időadatokat rendelnek, folyamatosan karbantartva a gyártórendszerek terhelési feltételeit és szabad kapacitásait. Az eljárás előnye a gyorsaság; az előkalkuláció és a középtávú tervezés segítése. Hátránya, hogy a megfelelő geometriai tervezés és a méretlánc elemzés hiányában nem létesít közvetlen kapcsolatot a konstrukció, a technológia és a termelésirányítás között és lényegében az alkatrészgyártás előtervezésére korlátozódik.

A technológiai előtervezésre a [32]-ben megfogalmazott elveket felhasználva és továbbfejlesztve célszerűnek látszik egy olyan, a mai üzemi gyakorlathoz jól illeszkedő rendszerfejlesztési koncepciót kidolgozni, amely az ajánlott magas teljesítőképességű, vegyes felépítésű rendszer és az alkatrészgyártásra szorítókozó variánselvű rendszer, mint két határeset között helyezkedik el. Eszerint a gépgyártási folyamatok tervezését következtetési sorrendben három nagyobb fázisra tagoljuk, amelyek mindegyike célszerűen további 3-3 tevékenység-szakaszra bontható (v.ö. 1. ábra):

- I. Gyártmányszerelési folyamat tervezése
 1. Szerelési műveletek sorrendtervezése
 2. Szerelési műveletek tervezése
 3. Szerelési időháló összeállítása
- II. Alkatrészgyártás előzetes tervezése
 1. Alkatrészek elemzése, rangsorolása
 2. Nagyvonalú folyamattervezés
 3. Előgyártmányok tervezése
- III. Alkatrészgyártási folyamat tervezése
 1. Műveleti sorrendtervezés
 2. Megmunkálási műveletek tervezése
 3. Műveleti idők meghatározása.



1. ábra Kilenc-szintű, korszerű struktúrájú CAPP rendszer a legfontosabb interfészek feltüntetésével.

Az új rendszerstruktúra előnye főként a CAD/CAPP/CAM, valamint a CAPP/PPS integráció kiépítésében mutatkoznak meg [30, 31].

4.2. A technológiai paraméterek szerepe a robusztus folyamattervezésben

Az optimális technológiai paraméterek meghatározása (pl. forgácsoló megmunkálások esetén az optimális forgácsolási feltételek rögzítése) a technológiai folyamatok tervezésének egyik klasszikus, rendszeresen visszatérő feladata. Az olyan forgácsoló műveletekre, amelyeknél a leválasztott pillanatnyi forgácskeresztmetszet első közelítésben konstans (vagyis ún. *kvázistacionárius* forgácsolási folyamat esetében), az optimalizálási modell hosszú idő óta ismert és számos megoldási módszert is publikáltak.

A feladat matematikai modellje a következőképpen fogalmazható meg [30, 31]:

$$U = \{u_i\}, \quad i = 1, \dots, I \quad (1)$$

$$S = \{s_j(\mathbf{u})\}, \quad j = 1, \dots, J, \quad \mathbf{u} \subset U \quad (2)$$

$$C = \{c_k(\mathbf{u}, \mathbf{s})\}, \quad k = 1, \dots, K, \quad \mathbf{s} \subset S \quad (3)$$

$$E = \{e_j(\mathbf{u})\} \quad (4)$$

$$\Phi = \{\varphi_m(\mathbf{u}, \mathbf{s})\}, \quad m = 1, \dots, M \quad (5)$$

ahol:

U - a technológiai paraméterek halmaza,
 S - az állapotváltozók halmaza,
 C - a termelési célfüggvények halmaza,
 E - az állapotegyenletek halmaza,
 Φ - a korlátozó relációk halmaza.

A korlátozó relációkat - attól függően, hogy csak egy, vagy egyszerre több technológiai paraméterre, mint optimalizálandó független változóra vonatkoznak-e - szokás *homogén*, illetve *inhomogén* korlátfeltételeknek is nevezni. Az (5) összefüggés részletesebben így is felírható:

$$\Phi = \left\{ \begin{array}{l} u_{i,\min} \leq u_i \leq u_{i,\max}, \quad i = 1, \dots, I; H_{j,\min} \leq \\ H_j(\mathbf{u}, \mathbf{s}) \leq H_{j,\max}, \quad j = 1, \dots, J \end{array} \right\} \quad (6)$$

Az (5) halmazban az u_i -re vonatkozó egyenlőtlenségek a homogén, a $H_j(\mathbf{u}, \mathbf{s})$ -ra vonatkozó egyenlőtlenségek az inhomogén korlátfeltételeket jelentik.

Leggyakrabban az alábbi két célfüggvény egyikét használják:

$$c_1(\mathbf{u}, \mathbf{s}) \rightarrow \text{minimum} \quad (7)$$

$$c_2(\mathbf{u}, \mathbf{s}) \rightarrow \text{maximum} \quad (8)$$

ahol c_1 költség típusú, c_2 termelékenység típusú célfüggvény.

Az (1) - (8) összefüggéseket konkretizáljuk a nagyoló esztergálás esetére. Ekkor

$u_1 \Rightarrow d[\text{mm}]$ - a fogásmélység,

$u_2 \Rightarrow f[\text{mm}/\text{ford}]$ - a főorsó-fordulatonkénti előtolás,

$u_3 \Rightarrow v[\text{m}/\text{min}]$ - a forgácsolási sebesség;

Φ : a korlátfüggvények halmaza (pl. az előtolás mértékadó felső korlátja: $f \leq f_{\max}$, a megengedhető forgácsolási teljesítmény: $P(d, f, v) \leq P_{\max}$);

C : az optimalizálandó paraméterektől és az állapotváltozóktól függő célfüggvény (pl. a művelet vagy művelelem költsége vagy termelékenysége);

S : az állapotváltozók halmaza (pl. $s_1 \Rightarrow T$: a szerszám éltartama, $s_2 \Rightarrow Q$: az anyagleválasztási folyamat intenzitása),

E : az állapotegyenletek halmaza (pl. az általános Taylor-féle szerzámméltartam-egyenlet, $T = T(\mathbf{u})$;

az anyagleválasztás intenzitása, $Q = Q(\mathbf{u})$).

Az itt bemutatott általános matematikai modellstruktúra bármilyen forgácsoló megmunkálás esetén alkalmazható a konkrét modell származtatására. Amennyiben vannak olyan paraméterkombinációk, amelyek az összes korlátfeltételt kielégítik, vagyis a megoldáshalmaz nem *üres* halmaz, az optimális paraméter-értékek és a célfüggvény extrémuma numerikus és/vagy grafikus eljárásokkal meghatározható [30, 31].

Meg kell azonban jegyezni, hogy a bemutatott matematikai modell alapján meghatározott u_i paraméterek használhatósága erősen korlátozott. Ennek okai a következők:

- Nem-stacionárius forgácsolási folyamat esetében a modell nagyon bonyolulttá válhat.
- A műveletelemenként külön-külön számított lokális optimumok nem robusztusak, ellenkezőleg, rendkívül érzékenyek a változásokra. Az optimalizálási feladat matematikai modellje csak technológiai és fizikai korlátozásokat tartalmaz. Az így meghatározott adatokat - NC szerszámgépek számítógépes programozása esetén - azonnal az alkatrészprogramba posztprocesszálják. Ez az *előidejű* tervezés nem számolhat a termelési folyamat olyan eseményeivel, amelyek a termelés tényleges megvalósítása, a *valós idejű* gyártásirányítás során merülnek fel (pl. kapacitások kiesése, szerszámihiány, rendelések prioritásának megváltozása, stb.). A korszerű gyártás nem az egyes műveletelemek, nem is az összetettebb műveletek költségoptimumát, hanem a *teljes gyártási folyamat* sok esetben összetett, többkomponenses *jó-sági fokának* (performance index) optimalizálását igényli. Mai tudásunk szerint ez igen nehéz feladat. Számítógéppel integrált rendszerekben a technológiai tervezés, a termelésstervezés és a műhelyszintű termelésirányítás (*Shop Floor Control*, SFC) integrált együttműködése hozhat megoldást.

A szerzőnek Detzky Ivánnal és Erdélyi Ferencsel közösen végzett sokéves kutatómunkája, amelybe 1993-tól Rayegani Farzad is bekapcsolódott, arra vezetett, hogy az előbbieken említett integrációt egy célszerűen megválasztott új állapotváltozó, a *technológiai intenzitás* (*material removal rate (MRR)* [cm^3/min] változó) jelentősen támogathatja [30, 31]. A kutatócsoport javaslata szerint a nagyoló műveletek tervezésénél *elegendő a technológiai intenzitást optimalizálni*. Ez egyszerűbben és gyorsabban megoldható, mint a fogásmélység, előtolás és forgácsolási sebesség háromváltozós optimalizálási feladata. Fontos felismerés, hogy csoporttechnológiai módszerek alkalmazása esetén a vezéralkatrésze számított optimális intenzitás az egész csoportra érvényes. Ha az alkatrészek nyers- és készdarabját CAD-rendszerrel tervezték, a műveletekben leválasztandó anyagterfogat (V_m [cm^3/min]) számítógéppel könnyen és gyorsan számítható. Ez pedig lehetővé teszi, hogy a termelésirányítás számára legfontosabb adatok, a megmunkálási idők és költségek számíthatók legyenek. A szükséges összefüggések a következők:

$$Q = d \cdot f \cdot v \quad (9)$$

$$t_g = \frac{V_m}{Q} \quad (10)$$

$$T = \left(\frac{C_v d^{1-x_v} f^{1-y_v}}{Q} \right)^q \quad (11)$$

$$\tau = \frac{K_\Sigma}{kV_m} = \frac{1}{Q} + \frac{Q^{q-1}}{R^q} \quad (12)$$

$$R = C_s \cdot f^{1-y_v} \quad (13)$$

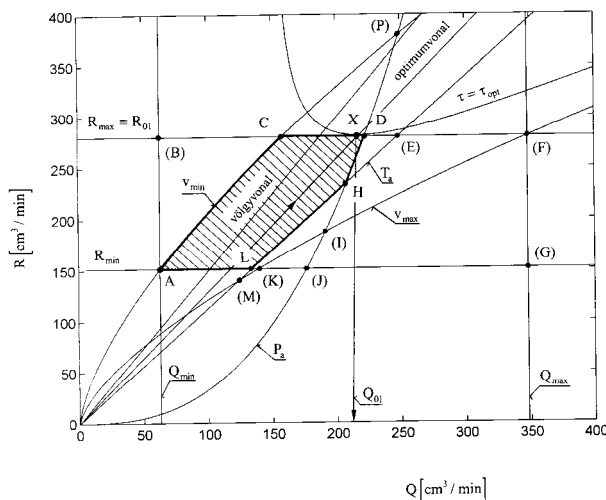
$$C_s = \frac{C_v d^{1-x_v}}{\left(\frac{K_{sb}}{kN_{\acute{e}l}} + t_{cs} \right)^m} \quad (14)$$

$$q = 1/m \quad (15)$$

ahol:

- Q - az anyagleválasztás intenzitása [cm^3/min],
- d - a fogásmélység [mm],
- f - az előtolás [mm/ford],
- v - a forgácsolási sebesség [m/min],
- t_g - a gépi főidő [min],
- V_m - a leválasztandó (működési) anyagterefogat [cm^3],
- T - a szerszám éltartama [min],
- C_v, x_v, y_v, m - empirikus kvázikonstansok, amelyek munkadarab- és szerszámél anyag párosításoként állandók,
- τ - a fajlagos anyagleválasztási idő [min/ cm^3],
- K_Σ - a művelet(-elem) összköltsége [Ft],
- k - a környezeti percköltség [Ft/min],
- $R=R(d,f)$ - egy új állapotváltozó, amely a d,f forgácsolási paraméterek monoton függvénye és dimenziója azonos a Q dimenziójával [cm^3/min],
- C_s - a szerszám komplex jellemzője egy adott anyagcsoport megmunkálásakor,
- K_{sb} - a szerszám alkalmazásával kapcsolatos teljes költség [Ft],
- $N_{\acute{e}l}$ - a szerszámlemez cserélhető éleinek száma,
- t_{cs} - az egy él cseréjéhez szükséges átlagos idő [min].

A módszer technikai részleteibe nyújt betekintést a 2. ábra. A teljes algoritmus –illusztratív példákkal – a [31] F1 Függelékében megtalálható.



2. ábra: Az optimumpont helyét R_{\max} és az optimumvonal metszéspontja jelöli ki: $X(Q_{01}, R_{\max})$.

Az ábrán zárójelbe írt betűkkel jeleztük azokat a pontokat, amelyek potenciálisan szóbajöhetnek a mindenkor keresési tartomány meghatározásakor, de a jelen esetben nem tartoznak az aktuális optimalizálási tartományhoz.

KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként, az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

IRODALOMJEGYZÉK

- [1] CVETKOV, V. D. 1965. Principi avtomatizacii projektirovanija optimalnih tehnologiceszkih processzov v masinosztroenii. *Masinosztroitel'*, No.6, pp. 7-10.
- [2] NIEBEL, B. W. 1965. Mechanized process selection for planning new designs. *ASME 33rd Annual Meeting collected papers*. V65. Book 4, Paper No.737.
- [3] CAM-I: Automated Process Planning System (CAPP). 1976. Systems Manual (Version 2, Release 1), Arlington, Texas
- [4] STEUDEL, H. J. 1984. Computer-aided process planning: past, present and future. *International Journal of Production Research*, 22 (2), 253-266.
- [5] EVERSHEIM, W. and SCHULZ, J. 1985. CIRP technical reports: survey of computer aided process planning systems. *CIRP Annals*, 34 (2), 607-613.
- [6] HAM, I. and LU, S.C.-Y. 1988. Computer-aided process planning: the present and the future. *CIRP Annals – Manufacturing Technology*, 37 (2), 663-675.
- [7] ALTING, L. and ZHANG, H. 1989. Computer aided process planning: the state-of-the-art survey. *International Journal of Production Research*, 27 (4), 553-585.
- [8] TÓTH, T. and VADÁSZ, D. 1977. TAUPROG-T: Programsystem for automatic planning of complete technological process of rotation symmetric parts. *COMPCONTROL '77. International Conference*, Proceedings of Section CAM, Warsaw, pp.194-202.
- [9] CSER, I. and TÓTH, T. 1986. Computer-aided technological process planning: methods, systems and application experiences. In: *Modelling and design of flexible manufacturing systems*, edited by Kusiak, A. Elsevier Science Publishers B. V., Amsterdam, pp.249-285.
- [10] TÓTH, T. and VADÁSZ, D. 1986. The TAUPROG system family: application experiences and new results of development. In: *Software for discrete manufacturing*, edited by CRESTIN, J. P. and McWATERS, J. F. (PRO-

- LAMAT'85, Paris), North-Holland, Amsterdam-New York-Oxford, IFIP, pp. 349-359.
- [11] GOUDA, S. and TARAMAN, K. 1989. CAPP: AAST, present and future. Technical Paper. Society of Manufacturing Engineers. MS.
- [12] ELMARAGHY, H. A. et al. 1993. Evolution and future perspectives of CAPP. *CIRP Annals-Manufacturing Technology*, 42 (2), 739-751.
- [13] EVERSHEIM, W. and SCHNEEWIND, J. 1993. Computer-aided process planning – state of the art and future development. *Robotics and Computer-Integrated Manufacturing*, 10 (1-2), 65-70.
- [14] KAMRANI, A. K., SFERRO, P. and HANDELMAN, J. 1995. Critical issues in design and evaluation of computer aided process planning systems. *Computers and Industrial Engineering*, 29 (1-4), 619-623.
- [15] LEUNG, H. C. 1996. Annotated bibliography on computer aided process planning. *International Journal of Advanced Manufacturing Technology*, 12 (5), 309-329.
- [16] CAY, F. and CHASSAPIS, C. 1997. An IT view on perspectives of computer aided process planning research. *Computers in Industry*, 34 (3), 307-337.
- [17] MARRI, H. B., GUNASEKARAN, A. and GRIEVE, R. J. 1998. Computer-aided process planning: a state of the art. *International Journal of Advanced Manufacturing Technology*, 14 (4), 261-268.
- [18] SHAH, J. J. 1991. Assessment of features technology. *Computer Aided Design*, 23 (5), 331-334.
- [19] SHAH, J., SREEVALSAN, P. and MATTHEW, A. 1991. Survey of CAD/ feature-based process planning and NC programming techniques. *Computer-Aided Engineering Journal*, 8 (1), 25-33.
- [20] YUE, Y. et al. 2002. Study on neural network techniques for computer integrated manufacturing. *Engineering Computations (Swansea, Wales)*, 19 (1-2), 136-157.
- [21] GUPTA, T. and GHOSH, B. K. 1989. A survey of expert systems in manufacturing and process planning. *Computers in Industry*, 11, 195-204.
- [22] KIRITSIS, D., NEUENDOR, K. P. and XIROUCHAKIS, P. 1999. Petri net techniques for process planning cost estimation. *Advances in Engineering Software*, 30 (6), 375-387.
- [23] METAXIOTIS, K. S., ASKOUNIS, A. and PSARRAS, J. 2002. Expert systems in production planning and scheduling: a state-of-the-art survey. *Journal of Intelligent Manufacturing*, 13 (4), 253-260.
- [24] LIAO, S. H. 2005. Expert system methodologies and applications – a decade review from 1995-2004. *Expert Systems with Applications*. 28 (1), 93-103.
- [25] PENG, Q., HALL, F. R. and LISTER, P. M. 2000. Application and evaluation of VR-based CAPP system. *Journal of Materials Processing Technology*, 107 (1-3), 153-159.
- [26] SHEN, W., WANG, L. and HAO, Q. 2006. Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey. *Systems, Man and Cybernetics, Part C: Applications and Review, IEEE Transactions*, 36 (4), 563-577.
- [27] SHEN, W. et al. 2006. Application of agent-based systems in intelligent manufacturing: an updated review. *Advanced Engineering Informatics*, 20 (4), 415-431.
- [28] ZHANG, W. J. and XIE, S. Q. 2007. Agent technology for collaborative process planning: a review. *International Journal of Advanced Manufacturing Technology*, 32 (3-4), 315-32.
- [29] XUN XU, LIHUI WANG and STEPHEN T. NEWMAN, 2011. Computer-aided process planning – A critical review of recent development and future trends. *International Journal of Computer Integrated Manufacturing*, 24(1), 1-31.
- [30] TÓTH TIBOR, 2004. Termelési rendszerek és folyamatok. A termelésinformatika alapjai. Miskolci Egyetemi Kiadó, ISBN 963 661 630 2.
- [31] TÓTH TIBOR, 2006. Tervezési elvek, modellek és módszerek a számítógéppel integrált gyártásban. Miskolci Egyetemi Kiadó, ISBN 963 661 339 7 (2. kiadás).
- [32] HORVÁTH MÁTYÁS, 1984. Alkatrészgyártási folyamatok automatizált tervezése. Akadémiai doktori értekezés, MTA Budapest.

LÉGKÖRI PARAMÉTEREK MINŐSÉGI MONITOROZÁSI RENDSZERÉNEK TERVEZÉSE ÉS MÉRÉSE

ATMOSPHERE QUALITY MONITORING SYSTEM DESIGN AND MEASUREMENT

Ján Turán*, Luboš Ovseník*, Michal Turán*, József Vásárhelyi**

ABSTRACT

Key elements and architecture of the developed optically powered atmosphere quality monitoring system are discussed. System design and some details of optically powered and optically controlled remote sensor module system design are presented. Finally some experimental results of system performance are presented.

Demand for information transmission can get from provider to user is rapidly growing. Establishing an optical connection has never been so easy as nowadays. Optical communication systems working in a free environment (FSO - Free Space Optics) becomes popular. Using FSO solutions of the so called last mile can be resolved within a few hours and without any legitimate self-administration, putting cabling in the ground or without infrastructure improvements.

Currently, lasers are so powerful that they allow transmitting signal from behind the glass window, thus increasing the possibility of in-door placement and security of FSO systems. But still the visibility of transmission and environmental impact of environment on the signal transmitted remains a big problem.

This paper deals in detail with a description of the instrument design and measuring the fog density, humidity and temperature of the environment. Long-term measurement of these parameters allows us to create models for optical signal transmission in FSO systems.

1. BEVEZETÉS

A korszerű ipari távfelügyeleti rendszerek minőségének alapját a szenzorhálózatok és hatékony beágyazott rendszerek alkotják [7].

A terepi rendszerek tápellátása megoldható optikai kábelek felhasználásával (powering trough optical fiber).

* Department of Electronics and Multimedia Communications University of Technology Košice, Szlovákia egyetemi tanár, egyetemi docens, PhD hallgató;

** Miskolci Egyetem, Automatizálási és Kommunikációtechnológiai tanszék, Miskolc, Magyarország egyetemi docens.

Az optikai kábelon keresztül történő tápellátás előnye az elektromágneses interferenciák (EMI) elleni immunitás, robbanásveszélyes környezetben (CH₄) történő biztonságos alkalmazás és különböző szenzorhálózatok illeszthetősége [2,3,7].

A terepi rendszerekben alkalmazott optikai kommunikációs rendszerek (Free Space Optics - FSO) látható vagy infra optikai sugarat alkalmaznak. [1-3]. Az FSO rendszer adó oldalán lézer diódák állítják elő a kis teljesítményű nyalábot, míg a vevő oldalon igen érzékeny fotódetektorok érzékelik az elküldött fénynyalábot. Az FSO rendszer hatótávolsága pár kilométer, amennyiben nem akadálymentes az adó-vevő közötti tér. Az ilyen rendszer legnagyobb hátránya az időjárási viszonyoktól való függőség. Az időjárási viszonyok legfőképpen a köd és a levegő páratartalma tartalma, azok a tényezők, amelyek zavarják az adatátvitel biztonságát [1, 3].

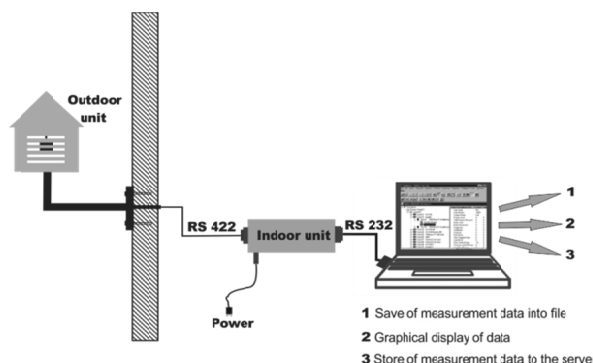
A köd pár száz mikronos átmérőjű vízcseppek összessége, de a fény szórását és a fényvisszaverődéseket nagymértékben befolyásoló tényező. Ezért az adatátvitel előtt szükséges az adat-átviteli közeg előzetes elemzése. Kódérzékelő szenzor segítségével megmérhetjük az adatátviteli rendszer minőségi paramétereit. Jelen cikk első része leírja az adatátvitel műszaki megvalósítását, majd bemutatja a kísérleti mérések eredményeit.

2. KÖD PARAMÉTEREINEK MÉRÉSE

A köd paramétereinek mérésére használt érzékelő (szenzor) méri a köd sűrűségét, hőmérsékletét és relatív páratartalmát. Az említett paraméterek mérése az adatátvitel minőségét befolyásoló tényezők kiértékelése szempontjából lényegesek [4]. A köd következtében fellépő jelesillapítást megbecsülni úgy lehet ha megmérjük a levegő páratartalmát (mh/m³). A feladat komplex és drága műszert igényel ezért a kísérlet során a BME Digitális és optikai Kommunikációs Csoport által fejlesztett műszert használtuk [4].

Az 1. ábra bemutatja a köd sűrűségét mérő rendszer összeállítását. A mérőrendszert a kísérletek alatt a

Kassai Egyetem Optoelektronikai laboratóriumában (KEMT FEI TUKE) helyeztük el. A beltéri egységet a rendszer, ahogyan már említettük méri a környezet hőmérsékletét, páratartalmát és a köd sűrűségét. A rendszer a következő modulokból épül fel: kültéri egység, beltéri egység.



1. ábra. Köd sűrűség mérő rendszer felépítése

A kültéri egység a beltéri egységgel RS422-es vonalon kommunikál, míg a beltéri egység és a központi számítógép között RS232-es soros kommunikációs vonalat építettünk ki. A kültéri egység által mért adatokat a számítógép dolgozza fel. Az adatok állományba elmenthetők, későbbi feldolgozások számára (diagramok és táblázatok) vagy egy központi szerveren eltárolhatók.

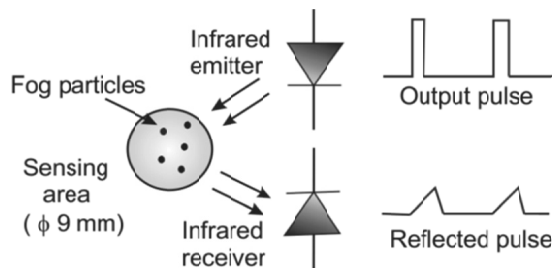
2.1 Módszer a köd sűrűségének mérésére

A kültéri egységben elhelyezett köd szenzor (2. ábra) a Kassai Egyetemen felszerelt kültéri egységet mutatja. A köd sűrűségének mérési elvét a 3. ábra mutatja. A sűrűség méréséhez az adó dióda (infra LED) impulzusokat küld, a vízcseppek által visszavert infra fény nyalábot a vevő LED vett impulzusok formájában közvetíti a beltéri egységnek. A visszavert jelek amplitúdójának nagyságát befolyásolja a ködben lévő vízcseppek száma.



2. ábra. Kültéri egység – Köd érzékelő

A vett impulzus amplitúdójának nagysága függ a környezetben lévő vízcseppek mennyiségétől. Az érzékelő ezen kívül méri a levegő hőmérsékletét és páratartalmát. Ezen paraméterek mérésével fontos információkhoz jutunk az időjárási viszonyokról is. A kültéri egység esőtől és direkt napfénytől védett, de a ködcseppek behatolhatnak a egység belsejébe [4].



3. ábra. A köd sűrűség mérés elve.

2.2 A ködérzékelő rendszer felépítése

A ködérzékelő rendszer beltéri és kültéri egységét a 4. ábra és az 5. ábra mutatja be.

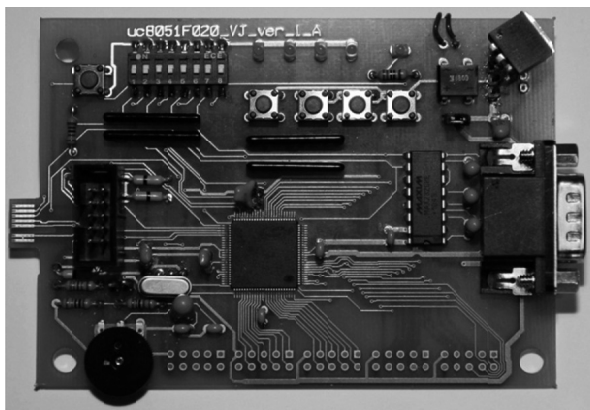


4. ábra. Kültéri egység

A kültéri egység mikrovezérlő egysége RS 422-es interfészen keresztül csatlakozik a beltéri egységhez. A beltéri egység fogadja a kültéri egység jeleit és konvertálja az RS-422-es szabványú jeleket RS-232-es szabványú jelekké, majd továbbítja a számítógép felé. A beltéri egység szolgáltatja a tápfeszültséget a kültéri egység felé, ezért a kültéri egységnek külön tápegységre nincs szüksége.



5. ábra. Beltéri egység

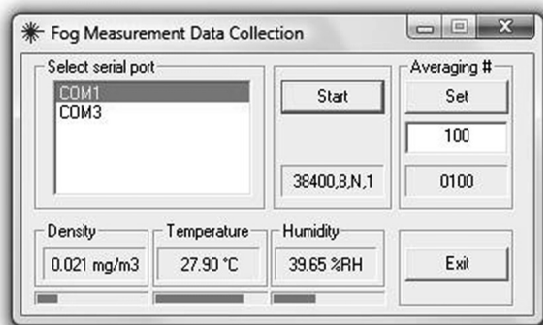


6. ábra. Mikrovezérlő egység

A Miskolci Egyetemen megtervezett mikrovezérlő egység használható mind beltéri, mind kültéri egységként (6. ábra). Az egység tartalmaz egy Silabs 8051F320 típusú mikrovezérlőt, RS-232-es illesztést és lehetőséget az RS422-es illesztés kiépítésére. Ugyanakkor rendelkezik analóg portokkal is. A Silabs F320-as kártya ezenkívül rendelkezik még nyomógombokkal, LED-ekkel és egy potenciométerrel. Az előbb említett elemek oktatási célokat szolgálnak.

3. A KÖRNYEZETI PARAMÉTEREKET MÉRŐ RENDSZER FELADATAI

A kültéri egység mikrovezérlőjének feladata az adatgyűjtés és továbbítás a beltéri egység felé. A beltéri egység átalakítja az adatokat és soros proton keresztül (RS-232) továbbítja az operációs rendszerrel rendelkező számítógép felé (Windows XP). A számítógépen az adatgyűjtés és az aktuális paraméterek kijelzése az adatgyűjtő programon keresztül történik. (lásd 7. ábra). Amennyiben a számítógép nem rendelkezik soros porttal, úgy a program tudja kezelni az USB portot is. A soros porton keresztül érkező adatokat, a program állományban tárolja.



7. ábra. Adatgyűjtő program kezelői felülete

A kültéri egységben a mért adatok mintavételezése 10 ms-os periódusokban történik. A mintavételezett adatokat 1 s-os intervallumban a kültéri egység átlagolja és az átlag értéket küldi el a beltéri egységnek. Az átlagolás értéke állítható (1-1000). A 8. ábrán ezt az értéket 100-ra állítottuk, mivel a mért értékeket szeretnénk másodpercenként ismerni.

4. MÉRÉSEK ÉS EREDMÉNYEK

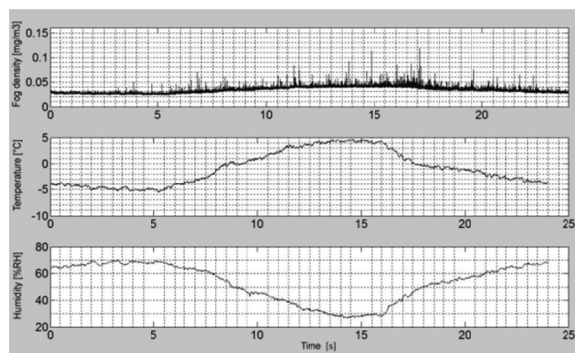
Általában a köd sűrűsége inhomogén közegként viselkedik, ezért az FSO rendszerekre gyakorolt csillapítás megállapításának érdekében az adatátviteli vezeték mentén kell elhelyeznünk a mérési egységeket. A mérési egységeket a Kassai Egyetem területén helyeztük el. A mérési eredményeket a továbbiakban ismertetjük.

Másodpercenként a mérésadatgyűjtő program a mért mintát elmenti egy szöveges állományba. Az alábbi táblázat egy ilyen mérésmintát tartalmaz. (8. ábra)

Density	Temperature	Humidity	Averaging	Relative Time
0168	3644	2110	100	0
0169	3644	2109	100	1
0169	3644	2109	100	2
0167	3644	2109	100	3
0168	3644	2109	100	4
0167	3644	2109	100	5
0167	3645	2108	100	6
0167	3645	2108	100	7
0167	3644	2109	100	8
0167	3644	2109	100	9
0169	3644	2108	100	10

8. ábra. Mérési adatok (köd sűrűsége, hőmérséklete, páratartalma)

A tárolt adatok további feldolgozása a MATLAB program segítségével lehetséges. A 9. ábrán a mérési eredményeket ábrázoltuk az idő függvényében. A „relatív idő” az aktuális napon mindig 0 –től kezdődik.



9. ábra. Mérési adatok grafikus ábrázolása (köd sűrűsége, hőmérséklete, páratartalma)

Egy hatékony és minőségi FSO rendszer tervezésekor fontos a környezeti paraméterek ismerete. Ezért a

felszerelt felügyeleti mérőrendszer képes rögzíteni a levegő páratartalmát, hőmérsékletét, sűrűségét (kód). A rendszer növeli az optikai kommunikációs rendszer hatékonyságát és megbízhatóságát.

5. KÖVETKEZTETÉSEK

Jelen cikkben bemutatunk egy a légköri paraméterek mérésére alkalmas rendszert, amely jelen állapotában a kód sűrűségét, hőmérsékletét és páratartalmát méri. A kísérleti méréseket a Kassai Egyetem területén elhelyezett kültéri egységek segítségével mértük.

A Miskolci Egyetemen kifejlesztett mikrovezérlős kártya alkalmas a kültéri és beltéri egységek megvalósítására.

A mért eredményeket felhasználtuk az optikai adatátviteli rendszer tervezésénél (FSO). A mért eredményeket MATLAB alkalmazás segítségével grafikusán is ábrázoltuk. Az eredmények kiértékelésekor meghatározott küszöbértékek segítségével meghatároztuk azokat a környezeti értékeket kód sűrűség, hőmérséklet, amelyeknél még az FSO rendszer működőképes.

A mérés adatgyűjtés még folyamatban van és az adatok feldolgozása is folyamatos.

AKNOWLEDGEMENT

This research was carried out as part of the TAMOP-4.2.1.B-10/2/KONV-2010-0001 project with support by the European Union, co-financed by the European Social Fund.

IRODALOM

- [1] BOUCHET O. et al. "Free Space Optics (Propagation and Communication)," ISTE Ltd., London, 2006.
- [2] KIM I., KOREVAR E., "Availability of Free Space Optics (FSO) and Hybrid FSO/RF Systems," Proc. SPIE, vol. 4530, 2001, 84-95.
- [3] NADEEM F., KVICERA V., SALEEM A. M., LEITGEB E., et al. "Weather Effects on Hybrid FSO/RF Communication Link," IEEE Journal on

- Selected Areas in Communications, vol. 27, no. 9, December 2009, pp. 1687-1697.
- [4] CSURGAI-HORVÁTH L., BITÓ J., "Fog Attenuation on V Band Terrestrial Radio and a Low Cost Measurement Setup," Future Network & Mobile Summit, Florence, Italy, Paper #47, June 2010.
- [5] TURÁN J., OVSENÍK L., VÁSÁRHELYI J., "Optically Powered Industrial Barometric System Design", Carpathian Journal of Electronic and Computer Engineering, vol. 3, no. 1, 2010, 131-136.
- [6] TURÁN J., OVSENÍK L., VÁSÁRHELYI J., TURÁN J. jr., "Optically Powered Industrial Barometric System", Proceedings of 19th International Conference Radioelektronika 2009, Bratislava, Slovak Republic, April 22-23, 2009, 165-167.
- [7] TURÁN J., OVSENÍK L., VÁSÁRHELYI J., "Optically Powered Industrial Barometric System Architecture", Proceedings of 11th International Carpathian Control Conference ICC2010, Eger, Hungary, May 26-28, 2010, 173-176.
- [8] PENA R., MATIAS I. R., ALGORA C., LOPEZ-AMO M., "Optical Fiber Based Power Delivery System for Optoelectronic Sensor Modules", LEOS'98, Piscataway, USA, vol. 2, 1998, 75-76.
- [9] BJORK P., LENZ J., FUJIWARA K., "Optically Powered Sensors. Optical Fiber Sensors", Washington, USA, vol. 2, 1988, 336-339.
- [10] BOTTER G., DRESCHMANN M., KLAMOURIS C., et al., "An Optically Powered Video Camera Link", IEEE Photon Technol. Lett., vol. 20, no. 1, 2008, 39-41.
- [11] HATMAN L. H., MUDD S. W., "SME Mining Engineering Handbook", SME Publ., New York, 1992.
- [12] WERTHEN J. G., ANDERSON A. G., WU T. C., "Optically Powered Sensors: Are they Really Fiber Optic Sensors", Proc. SPIE, vol. 2872, 1996, 131-138.
- [13] OVSENÍK L., TURÁN J., TATARKO M., MICHAL T., VÁSÁRHELYI J. "Fog Sensor System: Design and Measurement", proc of IEEE ICC2012, elfogadott cikk.

TERHELÉSELOSZTÁSI ALGORITMUSOK TESZTELÉSÉRE ALKALMAS RENDSZER TERVEZÉSE

DESIGN OF A TEST SYSTEM FOR LOAD DISTRIBUTION STRATEGIES

*Trohák Attila**

ABSTRACT

In my paper I introduce a part of my research results at the field of load distribution for power plants. I am developing different load distribution strategies to enhance the efficiency of power plants. To be able to validate my strategies I need a simulation environment and I introduce the details of the developed model system.

1. BEVEZETÉS

A kutatás során a célom egy olyan rendszer megtervezése és elkészítése volt, amely képes erőművi blokkok közötti terheléselosztásra. A terheléselosztó rendszer tervezése során két célt tűztem ki, ami alapján a terhelést eloszthatjuk. Az egyik ilyen cél a határfok növelése, a másik pedig a rugalmas terhelésváltoztatás lehetőségének megteremtése.

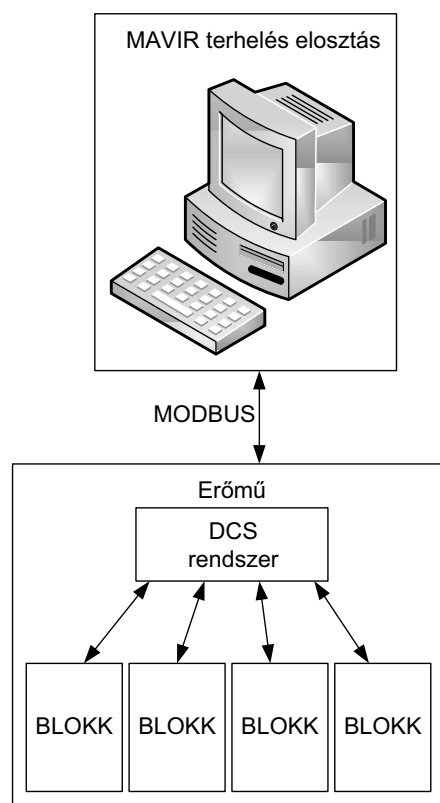
A kutatás során először el kellett készíteni egy olyan modell-rendszert, amin a kifejlesztendő algoritmusok tesztelhetők. A cikkben ennek a modell-rendszernek a részletei kerülnek bemutatásra. A modell-rendszer elkészülése után először a korábban alkalmazott terheléselosztási stratégiát valósítottam meg, majd a fenti két cél szerinti algoritmusokat fejlesztettem ki két illetve három blokkos esetekre.

Ahhoz, hogy a modell a valós rendszerhez minél jobban hasonlító környezetben tudjon üzemelni a kutatást ki kellett terjeszteni a modulok közötti kommunikációs lehetőségek feltárásával.

2. A MODELL-RENDSZER FELÉPÍTÉSE

A jelenlegi rendszer felépítését szemlélteti az 1. ábra. A MAVIR terheléselosztó rendszere meghatározza, hogy az irányítása alá tartozó egyes erőműveknek mekkora terheléssel kell üzemelniük. A szekunder szabályozásban résztvevő erőművek egy on-line kapcsolaton keresztül, esetünkben MODBUS-on keresztül kapják meg ezt a terhelés értéket. Ez az érték az adott erőmű karakterisztikáját figyelembe véve tetszőlegesen változhat. A korábban alkalmazott elosztási stratégia szerint a kívánt terhelési értéket az

erőmű irányító rendszere egyenlő arányban osztja el azt az üzemelő blokkok között.



1. ábra. A korábbi rendszer struktúrája

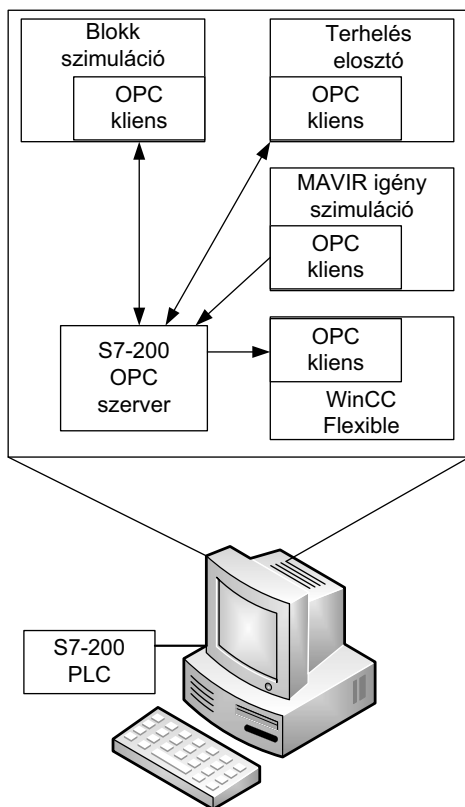
A 2. ábra az általam megtervezett modell-rendszert szemlélteti.

A modell fő moduljai:

- Blokk szimuláció;
- terhelés elosztó;
- MAVIR igény szimuláció;
- S7-200 OPC szerver;
- WinCC Flexible.

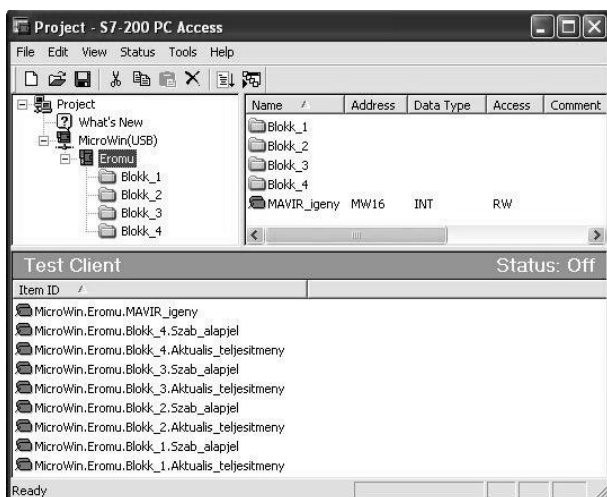
A továbbiakban bemutatásra kerülnek az egyes fő modulok, ismertetem a szerepüket, működésüket és használatukat.

* Egyetemi tanársegéd, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék



2. ábra. A modell-rendszer struktúrája

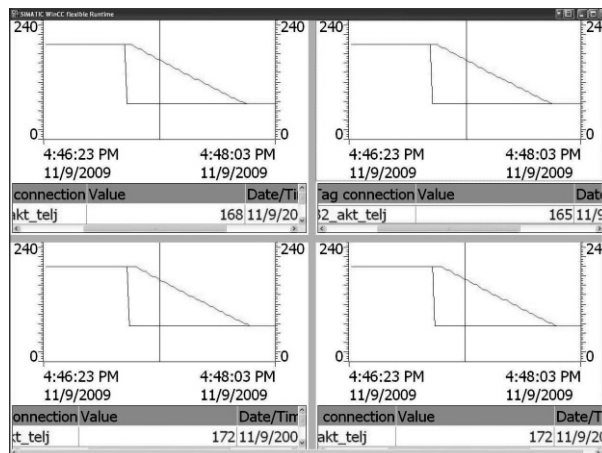
Az S7-200 OPC server feladata a DCS rendszer OPC serveri funkciójának ellátása a szimulációban. [4.] A másik négy modul ezen keresztül tud kommunikálni. A szerverben létrehoztam egy változót a MAVIR igény számára és blokkonként 1-1 változót a szabályozó alapjelének és az aktuális teljesítménynek.



3. ábra. Az S7-200 OPC server felülete

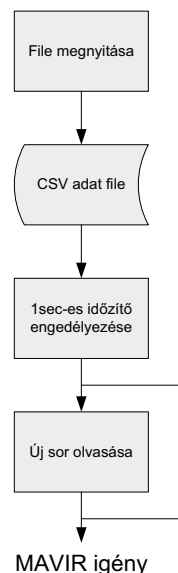
A WinCC Flexible egy HMI alkalmazás PLC rendszerekhez, mely képes OPC felületen keresztül érkező adatok segítségével is dinamikus megjelenítő felületeket kezelni. Feladata a blokkok üzemének

grafikus szemléltetése trendablakok segítségével. A 4. ábrán egy olyan helyzet látható, amikor 4 blokk esetén a csúcsterhelésről a minimális terhelésre adott parancsot az elosztó rendszer.



4. ábra. A megjelenítő felület működés közben

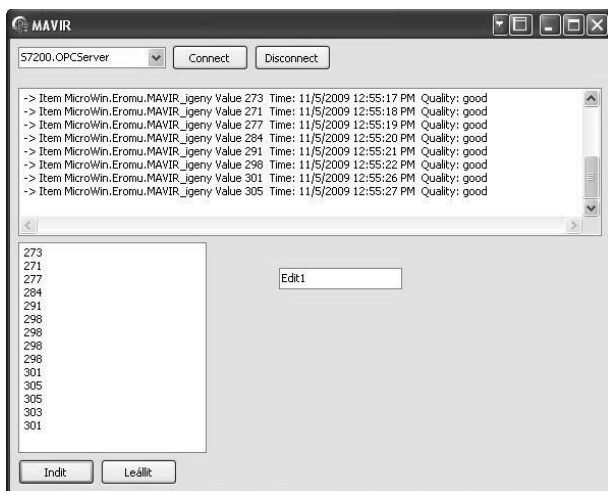
A MAVIR igény szimulátor feladata az, hogy a modellt ellássa olyan adatokkal, ami várhatóan a MAVIR-tól valós üzem közben érkezik. Ezt valós adatok felhasználásával oldottam meg úgy, hogy az egyes blokkok 1 perces mintavételezett adatait összegeztük. Az eredményt CSV formátumú file-ba tároltuk el. A szimulátor program a működése során ebből a file-ból olvassa ki másodpercenként a következő értéket és írja be azt az OPC server megfelelő változójába. A program működésének folyamatábráját szemlélteti az 5. ábra.



5. ábra. A MAVIR igény szimulátor folyamatábrája

A program használata során először ki kell választani az S7-200 OPC servert, majd a Connect gombra kell

kattintani a kapcsolódáshoz. Ezután az Indít gombra kattintással indíthatjuk el a beolvasási, OPC tag írási folyamatot. Ezt a folyamatot a Leállít gomb megnyomásával állíthatjuk le. Ilyenkor az Indít nyomógomb újbóli megnyomása a olvasási-írási folyamat folytatását jelenti attól a ponttól, ahol leállítottuk a folyamatot. Az OPC szerverről a Disconnect gomb segítségével lehet lecsatlakozni. A 6. ábra a működés közbeni programot szemlélteti.



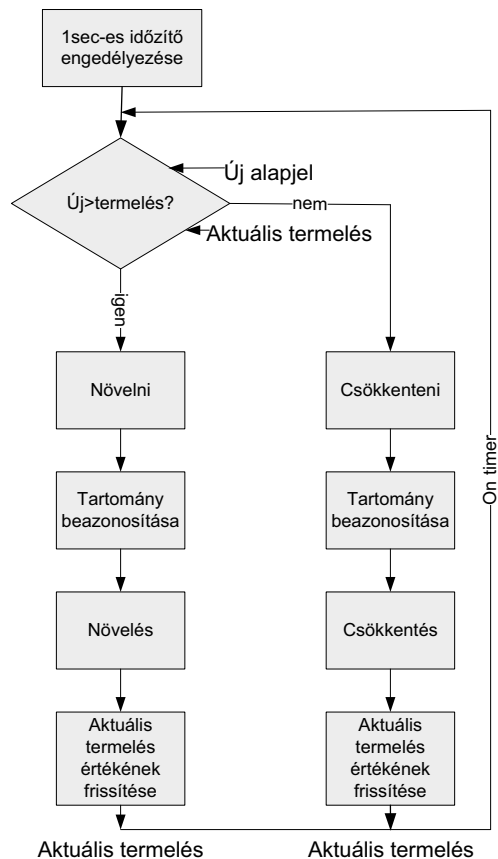
6. ábra. A MAVIR igény szimulátor kezelőfelülete

A blokk szimulátor feladata az, hogy az erőmű blokkjainak működését, a terhelés változását szimulálja. A modul az OPC szerverből olvassa a blokkok szabályozóinak alapjeleit és az aktuális termelési értékeket, elvégzi a szükséges számításokat, majd visszairja a szerverbe az új, aktuális termelési értékeket. A program működésének folyamatábráját szemlélteti a 7. ábra.

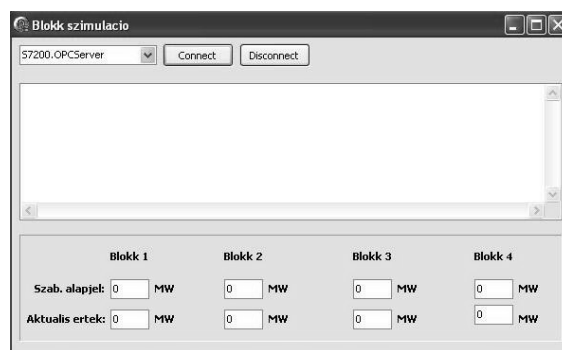
A program használata során először ki kell választani az S7-200 OPC szervert, majd a Connect gombra kell kattintani a kapcsolódáshoz. A program ezután automatikusan elkezd ellátni a feladatát. Az OPC szerverről a Disconnect gomb segítségével lehet lecsatlakozni. A 8. ábra a futó programot szemlélteti.

A 9. ábra egy olyan esetet szemléltet, ahol az 50MW-os minimális terhelésről adtunk ki parancsot a maximális 225MW-os terhelésre. Az ábrán jól láthatóak a különböző meredekségű szakaszok.

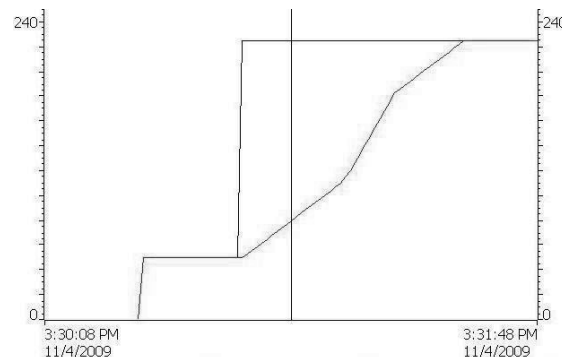
A terhelés elosztó feladata az, hogy az erőmű üzemelő blokkjai között a kiválasztott algoritmus szerint elossa a terhelést. A modul az OPC szerverből olvassa a blokkok aktuális termelési értékeit, a MAVIR igényt, elvégzi a szükséges számításokat, majd visszairja a szerverbe az új szabályozó alapjel értékeket. A program működésének folyamatábráját szemlélteti a 10. ábra.



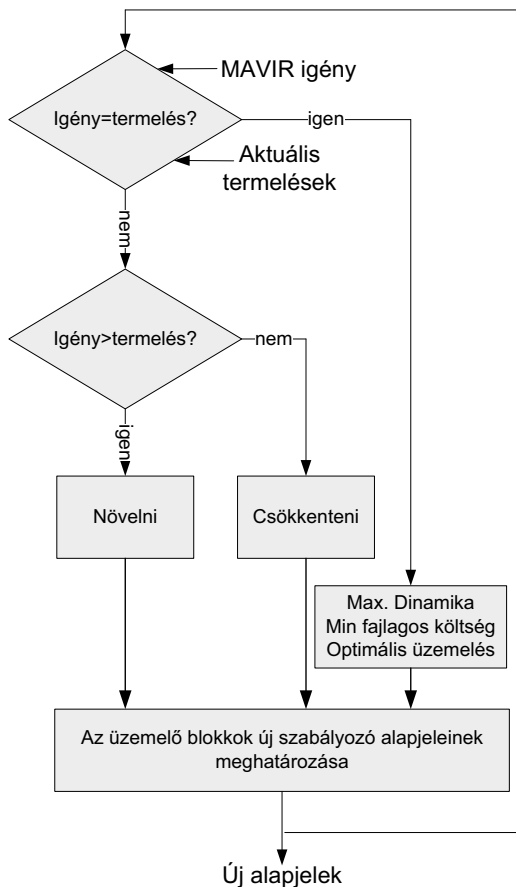
7. ábra. A blokk szimulátor folyamatábrája



8. ábra. A blokk szimulátor kezelőfelülete



9. ábra. A blokk szimulátor működésének eredménye



10. ábra. A terhelés elosztó folyamatábrája

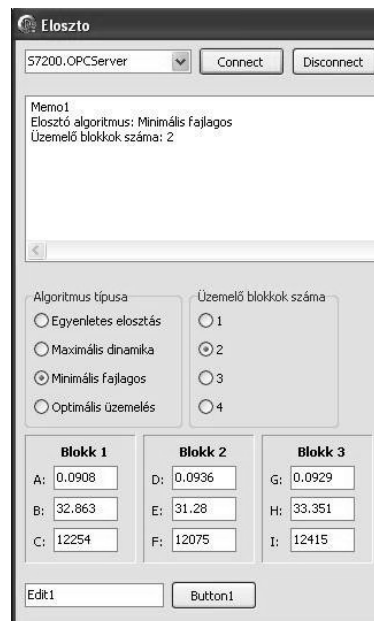
Az általam kidolgozott terheléelosztási stratégiák is ebben a modulban lettek leprogramozva, működési elvükkel korábbi publikációkban foglalkoztam. [1., 2., 3.] A program használata során először ki kell választani az S7-200 OPC szervert, az algoritmus típusát és az üzemelő blokkok számát, be kell állítani a blokkok karakterisztikáját leíró paramétereket, majd a Connect gombra kell kattintani a kapcsolódáshoz. A program ezután automatikusan elkezd ellátni a feladatát. Az OPC szerverről a Disconnect gomb segítségével lehet lecsatlakozni. A 11. ábra a futó programot szemlélteti.

3. ÖSSZEFOGLALÁS

A cikkben részletesen bemutatam az általam tervezett modell-rendszer felépítését, a komponensek szerepét, működésük folyamatábráit, működés közbeni kezelőfelületeiket. Az OPC kommunikációnak köszönhetően a rendszer részét képező, terheléelosztást megvalósító modul könnyedén integrálható valós ipari irányítórendszerekhez. A modell-rendszer alkalmas a kidolgozott terheléelosztási stratégiák szemléletes tesztelésére, kiértékelésére.

4. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az



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

11. ábra. A terhelés elosztó kezelőfelülete

5. IRODALOM

- [1.] TROHÁK A.: Terheléelosztási stratégiák kutatása erőművek hatékonyságnövelése céljából, *Fiatl Műszakiak Tudományos Ülésszaka XVII.: Nemzetközi Tudományos Konferencia*. Kolozsvár, Románia, 2012.03.22-2012.03.23., Kolozsvár: Erdélyi Múzeum-Egyesület, pp. 343-346.
- [2.] TROHÁK A.: *The Research and Modeling of Load Distribution Strategies in Power Plants*, ISW-DCS I konferencia, 2010.10.26., Lillafüred
- [3.] TROHÁK A.: *Erőművi terhelés-elosztási stratégiák és megoldások kutatása és modellezése a határfoknövelés és a rugalmas terhelésváltoztatás céljából*, *Doktoranduszok Fóruma 2009*, Gépészmérnöki és Informatikai Kar szekciókiadványa, pp. 176-181
- [4.] IWANITZ F., LANGE J.: *OPC Fundamentals, Implementation and Application*. Heidelberg, Hüthig GmbH, 2005.

FÜGGŐSÍNPÁLYA HANGVEZÉRLÉSÉNEK KIFEJLESZTÉSE

DEVELOPMENT OF A VOICE-COMMANDED INTERFACE FOR A MONORAIL

Pintér Judit Mária, Trohák Attila***

ABSTRACT

In our paper we introduce the steps and results of our research where our aim is to develop a system which is able to operate a monorail with voice commands.

1. BEVEZETÉS

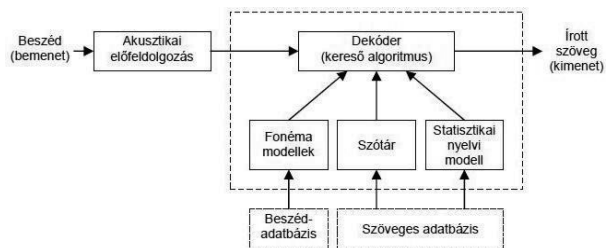
A cikkünkben bemutatásra kerülnek egy függősínpálya hangvezérlésének kifejlesztésére irányuló kutatás lépései, eredményei, melyben a célunk az, hogy szóbeli utasításokkal lehessen a rendszert működtetni. A cikkben bemutatásra kerülnek a számítógépes beszédfelismerők, azok általános felépítése és működése, majd a feladat ellátására alkalmasnak talált beszédfelismerő jellemzői, valamint a fejlesztés és a rendszerintegráció lépései.

2. A SZÁMÍTÓGÉPES BESZÉDFELISMERŐK ÁLTALÁNOS FELÉPÍTÉSE ÉS MŰKÖDÉSE

A számítógépes beszédfelismerők nem túl részletes működési vázlatát láthatjuk az 1. ábrán, amely fonémák modelljére épülő statisztikai alapú beszédfelismerő rendszert mutat be. A példa alkalmas annak bemutatására, hogy milyen működési elven alapszik a vezérléshez felhasznált beszédfelismerő. A bemenetre kapott beszédjelet előzetesen egy akusztikai feldolgozásnak vetjük alá, amelyet szokás lényegkiemelésnek is nevezni. A cél az, hogy olyan, az egyes beszédhangokat jól elkülönítő jellemzőket tartsunk meg, amelyek a beszédet a lehető legtömörebben reprezentálják lényegesebb információvesztés nélkül. A tömörítés szükségessége az eredendően hatalmas számítási igény kordában tartása végett is felmerül, a legtöbb alkalmazáshoz kívánatos ugyanis, hogy a beszédfelismerő valós időben is működőképes legyen (online üzemmód).

Ezután következhet a dekódolás folyamata. A művelet során a felismerés alapegységeinek modelljeit, a szótárt és a nyelv szintaktikai viszonyait statisztikai alapokon leíró nyelvi modellt használhatjuk fel. A Markov-modell, a szótár és a nyelvi modell együttesen egyfajta tudást visz a rendszerbe, melyet

számítógépes betanítás (training) során alapozhatunk meg. Ehhez beszéd, illetve szöveges adatbázis egyaránt szükséges. A beszédatadabázisnak a nyelvben előforduló valamennyi építőelemet és kapcsolatot is többszörösen tartalmaznia kell, hogy statisztikailag megfelelő lefedettséget adjon. Hasonlóképpen, a nyelvi modellnek is meg kell felelnie a beszédfelismerő használati területére jellemző szóhasználati szokásoknak.



1. ábra. Statisztikai számítógépes beszédfelismerő működése

3. A BESZÉDFELISMERŐ JELLEMZŐI

A függősínpálya kezelési feladatok megkönnyítésének és hatékonyabbá tételének egyik lehetősége a szóbeli utasításokkal való vezérlés. Bizonyos funkciók előhívása, vagy több ugyanazokból a lépésekből álló utasítások, folyamatok végrehajtása kiváltható lenne egészen rövid kifejezéseket tartalmazó szóbeli paranccsal. Ezeknek a kifejezéseknek természetesen egyértelműnek kell lenniük, azaz tartalmazniuk kell például egy rögzített kifejezést amivel "jelezzük" a rendszer felé, hogy egy utasítást fogunk közölni, nem pedig éppen csak beszélgetünk valaki mással. A "nyitó" kifejezést követően pedig következhet a fő utasítás, például adott funkció megnyitása, vagy értékek módosítása, lekérdezése stb., vagyis adottak lesznek kulcsszavak, amikkel egyértelműsíthető lesz a parancs. Bizonyos esetekben pedig biztonsági kérdésekkel (amiket a rendszer intéz a felhasználó felé pl. biztonsági kód kérése) láthatjuk el az egyes folyamatok elindítását vagy lezárását ezzel növelve a biztonságot.

* MSc. mérnök informatikus hallgató, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

** Egyetemi tanársegéd, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

Figyelembe kell venni azt is, hogy a felületet nem csak egy adott személy fogja nagy valószínűséggel használni, ezért a hangvezérlésnek ugyanolyan jó hatékonysággal kell majd működnie, minden egyes ember esetén. Ez két féle módon valósítható meg. Létrehozhatunk egy beszélőfüggetlen felismerőt, aminek minden személy esetén ugyanolyan magas hatékonysággal kell működnie, vagy minden felhasználó esetén külön tanítási folyamatot hajtunk végre és profilszerűen az adott személyhez kapcsolódó beszéd-modelleket (rejtett Markov-modellek) alkalmazunk a beszédfelismerőben és attól függően, hogy ki az aktuális felhasználó, úgy töltődik be az megfelelő konfiguráció. A beszélőfüggetlen megoldáshoz képest nagyobb biztonságot és könnyebb megvalósíthatóságot eredményez a felhasználónként történő betanítás. A munka során elsősorban beszélőfüggetlen rendszert szeretnénk létrehozni. Ehhez nagy számú tanítóanyagra van szükség.

Triviálisnak tűnő elvárás, de mégis meg kell jegyeznünk, hogy a hangvezérlés lehetőségének folyamatosan adottnak kell lennie és alkalmazkodnia kell a folyamatos emberi beszédhez.

Az igények és a kritériumok áttekintése után összegezhethetjük, hogy a navigáció megvalósításához pontosan milyen beszédfelismerőre is van szükség:

- kötött szótáras (kulcsszó alapú felismerés);
- mintafelismerő (rejtett Markov-modellek alkalmazása);
- kapcsolt szavak felismerésére alkalmas;
- beszélőfüggetlő;
- zajos környezetben alkalmazható;
- parancs üzemmódú;
- online üzemmódú.

4. A HANGVEZÉRLÉS FEJLESZTÉSÉNEK LÉPÉSEI

A projekt munka során, - amit az Anyagmozgatási és Logisztikai tanszékkel együtt valósítunk meg – az első lépés a kulcsszavak meghatározása volt. Mivel a függősínpályák olyan fejeletti, kötöttpályás szakaszos működésű anyagmozgató rendszerek, amelyek általában több, tipizált, önálló villamos hajtással rendelkező szállítóegységből (kocsiból), függősínekből és tetszőleges mértékig automatizált vezérlésből állnak, ezért azok koordinálása is nagy mértékben korlátozott. Használhatjuk például az előre/hátra és fel/le parancsokat kiegészítve, hogy milyen tempóban mozogjon, mivel a függősínpályán a kocsi két eltérő sebességgel is tud haladni. A pályán a kocsi két helyen tudjuk pozicionálni, a váltóban és az állomásnál, ezért ezeket a szavakat is belevettük a szókészletbe. Az Anyagmozgatási és Logisztikai Tanszék jövőbeli tervei közt szerepel, hogy koordináták megadásával is tudják pozicionálni a rendszert, ezért a számjegyek is a szókészlet részét képezik. Külön kérésre a vészleállítás

parancsot is felvettük a kulcsszavak közé. Így összesen 26 db kulcsszót határoztunk meg.

A következő lépés a hangminták begyűjtése volt. A hallgatók segítőkészségében bízva megpróbáltunk a Sphinx beszédfelismerő szoftver - amit majd a hangvezérlés megvalósításához fogunk használni – honlapján ajánlott betanító adatbázisához a szükséges mennyiségű hangmintára szert tenni. Az ajánlás szerint beszélőfüggetlen, vezérlő és irányító (command and control), kis szótáras beszédfelismerőkhöz összesen 5 órányi hanganyagot ír elő 200 beszélőtől lehetőleg egyenlő eloszlásban kell származnia nemekre és korcsoportokra nézve. Eddig összesen 56 embertől sikerült hangmintát összegyűjtenünk, durván a negyede az ajánlásnak de a tesztelesek megkezdéséhez elegendőnek bizonyult. A hangminták 22050 Hz-en lettek WAV formátumban (előjeles 16 bites PCM). rögzítve. (A hallás frekvenciatartományát 20 Hz és 20 kHz közé teszik a technológiai tervezés során. Például a legtöbb jó minőségű hangszóró ebben a sávban sugároz, a Hi-Fi minőségű hanganyagot 44100 Hz-en mintavételezik, ez a Shannon-féle mintavételezési törvény szerint 22050 Hz-es sávzélességet jelent. 20 kHz-nél nagyobb sávzélességet – pl. 96 kHz-es mintavételezés – általában akkor használnak, ha további jelfeldolgozási feladatokat kívánnak elvégezni.) [2] A hangmintákat a betanítás előtt rendszerezni és ellenőrizni kellett és a rögzítés során használt kérdőívek alapján pedig megállapítható, hogy a minták nemekre vonatkozó eloszlása megfelelő, korcsoport tekintetében (mivel hallgatóktól származnak a minták többségében) pedig szinte az összes minta 20 és 25 év közötti személyektől származik. A betanítás megkezdése előtt a hangmintákon lényegkiemelést (MFCC konverziót) kellett végrehajtani, amit a HTK toolkit szoftvercsomag HCopy programjával végeztük el.

A betanításhoz szükséges még:

- a szótár, ami a kulcsszavakat tartalmazza némileg módosult formában, melynek oka, hogy a HTK toolkit szoftver, (amin a betanítást és a teszteleket a jövőben végezzük) nem értelmezi az ékezetes karaktereket, ezért némely esetben a SAMPA* szabványt követtem,
- nyelvtan fájl, ami a parancsok szintaktikáját tartalmazza. A jellemzőkben megállapítottuk, hogy a biztonságosabb működéshez szükséges lenne egy "kezdő" kifejezés használata a parancsoknál. Így a nyelvtan fájlban meghatároztuk, hogy minden kifejezésnek a start kulcsszóval kell kezdődnie (kivéve stop parancs),
- a betanítandó rejtett Markov- modell prototípusa. A modellek különböző állapotszámúak lehetnek, hogy melyik lesz a leoptimálisabb, a tesztelesek során fog kiderülni.

A felsorolás csak a legfontosabb tanító elemeket tartalmazza, ezen felül számos más összetevő is szükséges.

* a nemzetközi fonetikai ábécé ASCII átírása a SAMPA

Miután ezek a tanításhoz szükséges fájlok rendelkezésre állnak, a továbbiakban elvégezzük a betanítást, tesztlesekkel megpróbáljuk a lehető legjobb hatékonyságot elérni, majd a „végleges” konfiguráción zajteszteket végrehajtani, hogy a rendszer mennyire érzékeny a különböző zajokra. Mivel logisztikai rendszerhez fejlesztünk hangvezérlést, ezért annak ipari környezetben is – ahol általában egyszerre több zajforrás is fellelhető - magas hatékonysággal kell működnie.

5. A FELISMERŐ INTEGRÁCIÓJA

A vezérlés megvalósításához a beszéd felismerő által felismert utasításokat valamilyen módon továbbítani kell a függősín-pálya vezérlőegysége felé, ahol adott folyamatok végrehajtását eredményezi. Ennek megfelelően egyértelműen meg kell határozni, hogy melyik kulcsszó melyik folyamat végrehajtásának felel meg és biztosítani kell a valós idejű végrehajtást.

Első lépésként a függősín-pálya jelenlegi vezérlését mértük fel. Megállapítottuk, hogy a jelenlegi vezérlés egy nagy, összetett rendszer vezérlő PLC-jén kapott helyet. Ez egy Mitsubishi PLC QX81 típusú digitális bemeneti és QY81P típusú digitális kimeneti modulokkal. Ezek a modulok 24 VDC jelszinten működnek. Mivel a beszéd felismerő egy számítógépen futó szoftver, ezért a felismerő által detektált parancsot be kell juttatni valamilyen módon a függősín-pálya vezérlő rendszerébe. A komplex irányító rendszernek ha a digitális bemeneti csatornáira juttatjuk a jelet, akkor a PLC programjában is módosításokat kell végrehajtani, ami egy több száz jelet kezelő, rendszeresen használt berendezés esetében nem kockázatmentes, ezért a vezérlő rendszer ideiglenes helyettesítése mellett döntöttünk.

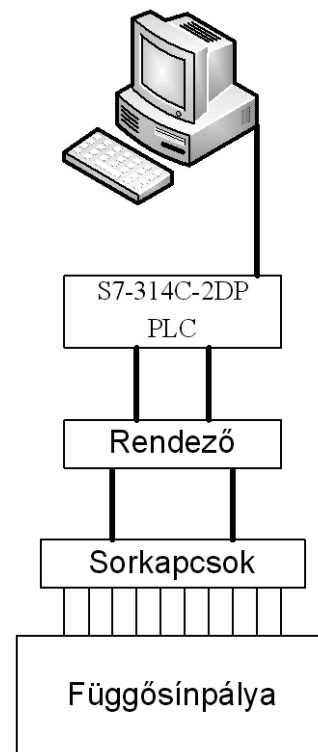
A technológia jelei egy sorkapcsos rendszert követően gyűjtőkábelek segítségével csatlakoznak a vezérlő rendszerre. Az IO modulok „apa” csatlakozóval vannak ellátva, míg a gyűjtőkábel végén „anya” csatlakozó található. A megoldásunk az, hogy egy új PLC-t csatlakoztatunk a technológiára úgy, hogy a PLC IO jeleinek vezetékét „apa” csatlakozó végével látjuk el és így a technológia jeleivel a gyűjtőkábel „anya” csatlakozójával szembe tudjuk kötni. Így a fejlesztés során a komplex technológiai rendszerről másodpercek alatt le lehet választani a függősín-pályát.

Az új PLC egy Siemens S7-314C-2DP típusú kompakt vezérlőberendezés, melyre az alábbi IO jelek kerülnek csatlakoztatásra:

- Bemenetek:
 - futómacska működik;
 - forgató végállás kapcsoló keresztben;
 - forgató végállás kapcsoló hosszában;
 - forgató, macska nincs bent;
 - forgató, ready lámpa;

- forgató, vészstop gomb benyomva.
- Kimenetek:
 - keresztbe forgatás Kone forgató;
 - hosszába forgatás Kone forgató;
 - fel;
 - le;
 - emelés gyors;
 - macska jobbra/előre;
 - macska balra/vissza;
 - futómacska gyors;
 - kürt.

A számítógépes szoftver kimenetének és a PLC-nek az összekötési problémája továbbra is megmaradt, de ezzel a rendszer kialakítással ez már könnyedén áthidalhatóvá válik. Amennyiben a PLC digitális bemeneteire szeretnénk csatlakozni egy számítógépről, akkor a számítógépet el kell látni egy olyan modullal, ami képes 24 VDC jelet küldeni a PLC irányába. Ezt a fajta bővítést a költséghatékony rendszer kifejlesztése érdekében elvetettük. Egy másik lehetséges megoldás a PLC bővítése kommunikációs kártyával, például egy soros kártyával és ekkor a számítógép soros portján küldjük ki a detektálás után az utasításokat. Valós üzemi körülmények között ez a megoldás is igen költséges lehet az irányítórendszerek bővítő moduljainak magas ára miatt.



2. ábra. A vezérlés kialakítása

A választott megoldásunk végül az lett, hogy kihasználjuk azt, hogy az ilyen irányítórendszerek operátori kezelőfelülettel ellátottak, azaz egy

számítógépen futó SCADA/HMI alkalmazás az operátoroknak lehetővé teszi a technológia állapotainak megjelenítését és beavatkozási, irányítási lehetőséget is biztosít számukra.

A választott PLC-hez egy PC alapú HMI felületet készítettünk WinCC Flexible Runtime segítségével és a klasszikus megjelenítő felületről működtethetővé tesszük a függősínpálya rendszert. Ebben az esetben a HMI felület képes a PLC változóit írni és olvasni, tehát valamilyen módon azt kell megoldanunk, hogy a PLC változójának írása ne egy kezelőfelületen elhelyezett gomb megnyomására történjen meg, hanem egy szóbeli utasítás hatására. A megoldás kulcsa az irányítástechnikai informatikai rendszerek esetében igen elterjedt OPC technológia lehet. Az OLE for Process Control (OPC) a windows alkalmazások és a termelési folyamatok irányítását végző hardvereszközök közti kapcsolatteremtés eszköze szerver/kliens formában. [5.] Nyitott szabvány, amely a termelés folyamatát irányító eszközök adatainak állandó, torzítatlan elérését célozza meg. Módszere független az adat típusától, illetve az adat forrásától. A HMI felületen nem csak a PLC irányába lehet adatcserére irányuló kapcsolatot konfigurálni, hanem egy OPC szerver felé is.

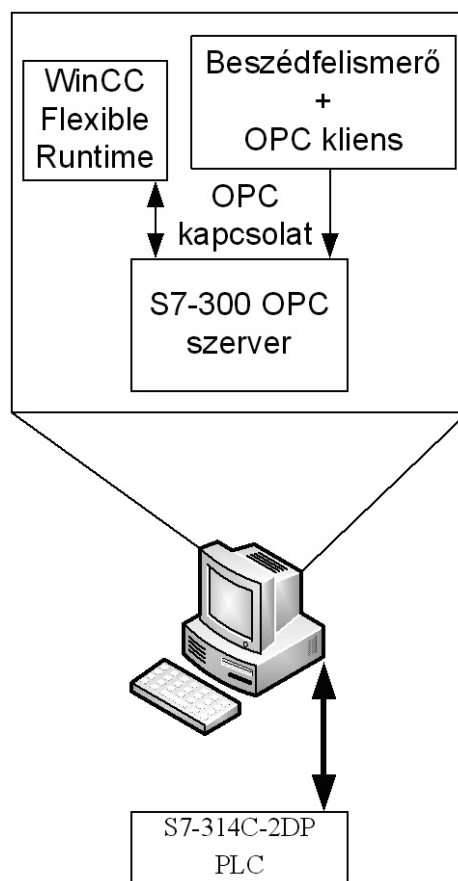
Tehát a cél az, hogy a beszéd felismerő szoftvert a HMI felületet futató PC-n futtatva egy OPC szerverbe juttatott adat segítségével a HMI felületen át hanggal vezérelhessük a függősínpályát. Ezt a legegyszerűbben úgy valósíthatjuk meg, ha veszünk egy adott változót és annak értékét annak megfelelően módosítjuk, hogy milyen kulcsszó hangzik el. Ez persze nem jelenthet sok értéket, de a mi esetünkben nincs is szükség rá (kötött szótár felismerő). Így az egyes kulcsszavakhoz egy adott értéket rendelünk, amit a változó vesz fel, ha elhangzik az adott utasítás. A HMI rendszernek ezt az értéket folyamatosan figyelnie kell és változása esetén az adott értéknek megfelelő kiváltott eseményt végrehajtani.

6. ÖSSZEFOGLALÁS

A cikkünkben bemutatásra kerültek egy függősínpálya hangvezérlésének kifejlesztésére irányuló kutatás lépései, eredményei, melyben a célunk az volt, hogy szóbeli utasításokkal lehessen a rendszert működtetni. A kutatás jelenleg a beszéd felismerő modul betanításánál tart és tesztelésénél tart, valamint az új PLC vezérlőprogram és a HMI felület készítése zajlik. Az OPC alapú egyedi kliensek és kapcsolatok kialakítása a korábbi tapasztalataink alapján nem fog problémát okozni. [3., 4.]

7. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az



3. ábra. Szoftveres integráció

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

8. IRODALOM

- [1.] VICSI K., NÉMETH G., SZASZÁK GY. A beszéd számítógépes feldolgozása
- [2.] NÉMETH G., OLASZY G.: A Magyar Beszéd, Akadémiai Kiadó, Budapest, 2010
- [3.] TROHÁK A.: Erőművi terhelés-elosztási stratégiák és megoldások kutatása és modellezése a hatásfoknövelés és a rugalmas terhelésváltoztatás céljából, Doktoranduszok Fóruma 2009, Gépészmérnöki és Informatikai Kar szekciókiadványa, pp. 176-181
- [4.] AJTONYI I., TROHÁK A.: Intelligens irányítórendszer tervezése és modellezése golyós őrlőmalomhoz, Doktoranduszok Fóruma 2008, Gépészmérnöki és Informatikai Kar szekciókiadványa, pp. 93-99
- [5.] IWANITZ F., LANGE J.: OPC Fundamentals, Implementation and Application. Heidelberg, Hüthig GmbH, 2005.

IWLAN IPARI KOMMUNIKÁCIÓS RENDSZEREK TERVEZÉSE, SZIMULÁCIÓJA, VIZSGÁLATA

DESIGN, SIMULATION AND EXAMINATION OF IWLAN INDUSTRIAL COMMUNICATION SYSTEMS

Rádi Péter Trohák Attila***

ABSTRACT

In the previous few years the wireless communication systems became popular in the industrial applications, because of the improvements in reliability and redundancy. This article will introduce the Siemens IWLAN system, and its capabilities.

I will show how to simulate, build and test an industrial wireless communication system.

1. BEVEZETÉS

A vezeték nélküli hálózatok ipari felhasználása sokáig váratott magára, mivel hiányosságai miatt még több területen fejlesztést igényelt.

A gyártók létrehozták a saját, ipari célokra használható vezeték nélküli adatátviteli rendszereiket, így a Siemens is az IWLAN (Industrial Wireless Local Area Network) technológiát. Nevéből is látható, hogy közel áll a civil felhasználású WLAN-hoz. A kettő kompatibilis egymással, viszont az IWLAN robosztus kialakítású és ki van egészítve ipari igényeket is kielégítő megoldásokkal. Ezek a megoldások megbízhatóvá és determinisztikussá teszik. A valós idejű, determinisztikus, illetve redundáns működés alapkövetelmény az ipari alkalmazásoknál, mivel egy későn érkező, vagy szélsőséges esetben elvesző adatsomag végzetes következményekkel járhat.

A telepítés sokkal gazdaságosabb és gyorsabb, mivel a kábelek mennyiségét nagymértékben lecsökkenti. Ennek következtében kisebb a kopásnak kitett alkatrészek száma is. Szabványosságának köszönhetően egyszerűen integrálható már meglévő rendszerekhez. 2,4GHz és 5GHz-es ISM sávokban is működik az IEEE802.11 kiterjesztéseként.

Cellás felépítésű rendszerek is kialakíthatók ilyen eszközökkel, ami kiküszöböli az interferenciát és lehetőséget biztosít a klienseknek, hogy válasszanak a



1. ábra IWLAN kompatibilis hardverek

csatornák közül az optimális adatkommunikációt szem előtt tartva. Hozzáférési pont (AP) centrikus, és ad-hoc típusú hálózatok építhetők ki. Az előbbi biztonságosabb, és lehetőség nyílik benne a kommunikációs folyamat nyilvántartására, az utóbbinál pedig a kliensek maguk közt építik ki a hálózatot koordinátor igénye nélkül. Az egységek nagyon részletesen paraméterezhetők a felhasználási igénynek megfelelően.

Rendkívül nagy segítséget jelent telepítésnél a szintén Siemens által készített Sinema E szoftver, mely lehetővé teszi a rendszerek megtervezését, beállítását és szimulációját. Létrehozhatjuk benne az elkészíteni kívánt rendszer, környezet virtuális másolatát, amit megfelelően bekonfigurálva (valóságnak megfelelő anyagok felvitelével) ellenőrizhetjük, hogy a terveink alapján működni fog-e, hogyan fog működni az összeállítás. A szoftver által számított értékek ellenőrzésére különböző eljárásokat használhatunk, mint például az IWLAN egységek jelerősséget megjelenítő menüpontja, vagy egy rádiófrekvenciás spektrumanalizátor.

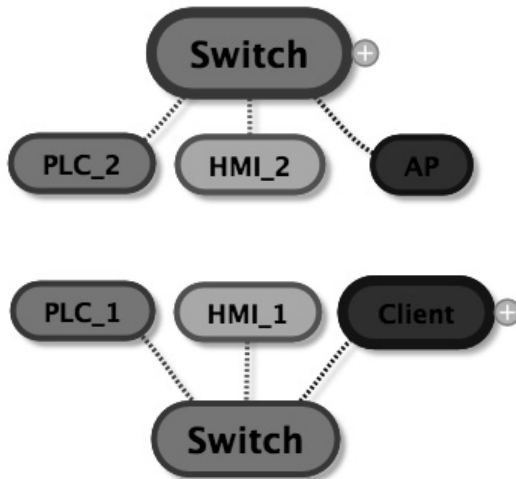
Az elvégzett és itt bemutatásra kerülő kutatómunka ezen rendszerek különböző beállítások melletti viselkedésének és megbízhatóságának részletekbe menő vizsgálata és feltérképezése.

* hallgató, Miskolci Egyetem, GÉIK

** tanársegéd, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék (társzerző)

2. A rendszer

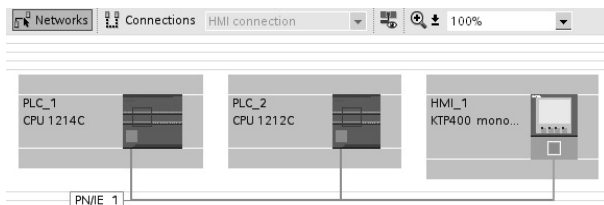
Egy célorientált rendszer került összeállításra, amin jól megfigyelhetők a vezeték nélküli hálózat átviteli képességei. A figyelt kommunikáció két Siemens S7-1200 PLC között zajlik. Egyikük fix helyen (Access Point oldal), a másik pedig mozgatható módon (Client Module oldal) van kialakítva. Mindkettő mellé csatlakoztatva van switch-en keresztül egy-egy HMI panel is, amiken nézhető az adatok küldése és fogadása.



2. ábra A rendszer sematikus rajza

A vizsgálat a következő képpen zajlik. Az Access Point oldali PLC bizonyos időközönként adatsomagokat küld a másik PLC-nek (értelmszerűen az IWLAN hálózaton keresztül) ami fogadja, majd visszaküldi azt. Az első PLC számolja a küldéstől a visszaérkezésig eltelt időt, és rögzíti. Ezzel a módszerrel ping szerűen tudjuk megfigyelni az átviteli időt az egyes helyeken. A kapott értékeket összehasonlíthatjuk a Sinema által szimulálttal, alátámasztva annak megbízhatóságát.

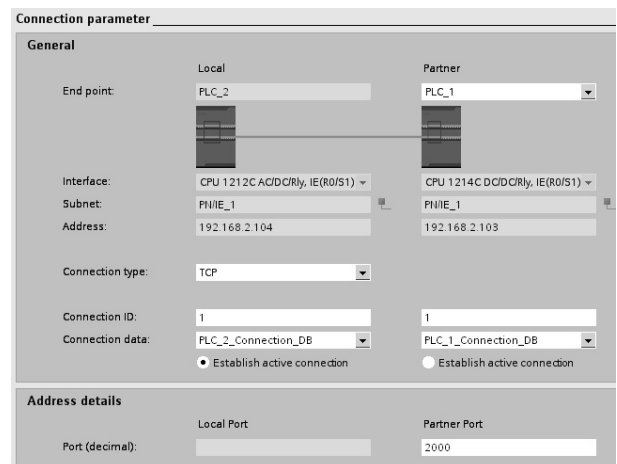
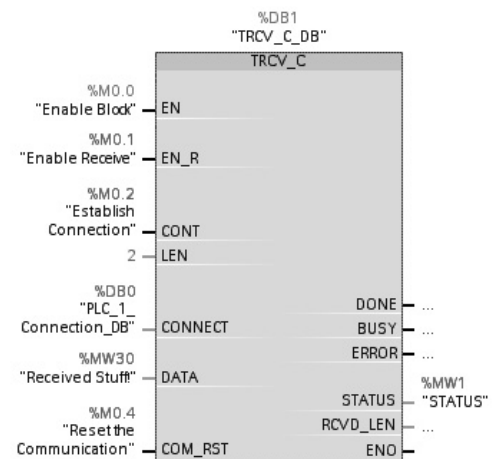
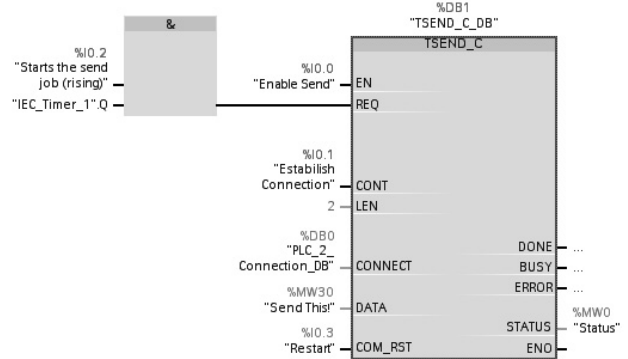
A rendszer kialakításánál nagy segítséget jelentett a Totally Integrated Automation (TIA) szofrver, ami egy egységes megoldást kínál az eszközök konfigurálására és programozására. Felvehetjük benne a PLC-ket, és HMI-eket, majd virtuálisan összevetékelhetjük őket a valóságnak megfelelően. IP címmel láthatjuk el őket, ami alapján elérhetővé válnak programozásra.



3. ábra Eszközök felvétele a TIA-ba

A rendszer bekonfigurálása után következett a programok megírása. A CPU-k PROFINET portján

keresztül történő kommunikáció a következő blokkokkal valósítható meg.



4. ábra A küldő és fogadó blokk, kapcsolatbeállítás

Ezek után a TIA felajánlja a kapcsolat beállításának lehetőségét. (IP-cím, port, stb). [1.]

3. IWLAN

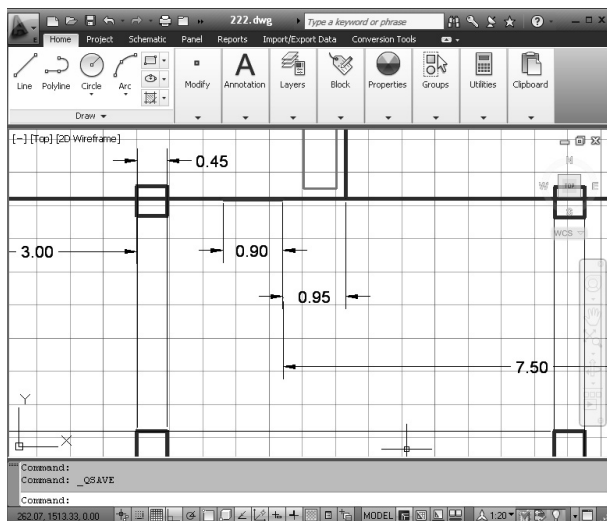
Access Point: SCALANCE W784-1
Client Module: SCALANCE W746-1
Antennák: ANT795-4MS
Szoftver: Siemens Sinema E 2006
Tervezés: AutoCAD 2012

Az Access Point az IWLAN hozzáférési pont, amihez a Client Module csatlakozik. A kettő vezeték nélküli hálózatot alkot, ami akár lehet a kiterjesztése egy már kiépített vezeték hálózatnak. Természetesen bővíthetjük a rendszert több klienssel, de jelenleg egy darab áll a rendelkezésünkre, így a cikk ez alapján készült.

A modulokra két-két, osztott küldés/fogadás-ra állított antenna van szerelve. A szoftver, amivel szimulálhatjuk a jelerősséget, és beállíthatjuk a modulokat a Siemens kifejezetten erre a célra készült Sinema E programja.

Mint minden mérnöki munkát ezt is meg kell tervezni előre. Szükségünk lesz a modellezni kívánt épület részletes méreteire. Ezt legegyszerűbben úgy tudjuk megoldani, hogyha megszerezzük az alaprajzot. Ennek híján saját magunknak kell megrajzolni azt a kívánt részletességgel. A kutatás során az AutoCAD-et használtuk erre a célra, de természetesen más tervezőprogramok is rendelkezésre állnak.

Az alaprajzra azért van szükség, mert a Sinema-ban arra húzzuk rá a szimulált falakat és egyéb objektumokat.



6. ábra AutoCAD tervrajz

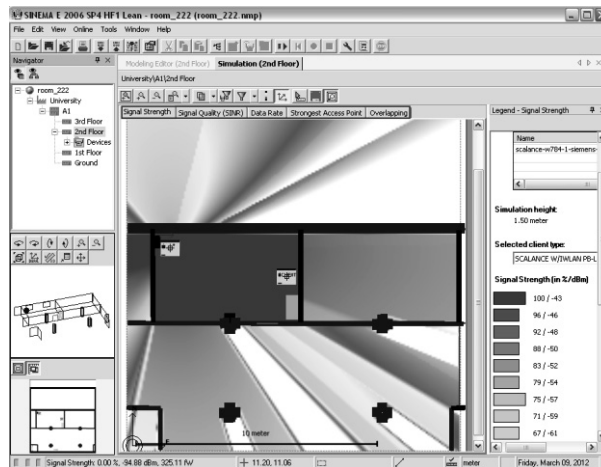
A mérést egy ultrahangos távolságmérővel végeztük. Az eredményeket felírtuk, majd bevittük az AutoCAD-be. Rétegeket is használtunk, így a végén el lehetett tüntetni egy kattintással a méretvonalakat, amik csak zavarták volna a további munkát.

A szimulációs környezet felépítése a Sinema E programban történik. Új projekt létrehozása után lépésről lépésre történik az emeletekkel kezdve. Az

alaprajzokat célszerű beimportálni háttérnek, és utána arra ráhúzni a szimulálandó anyagokat.

Az IWLAN modulokkal három féle módon léphetünk kapcsolatba.

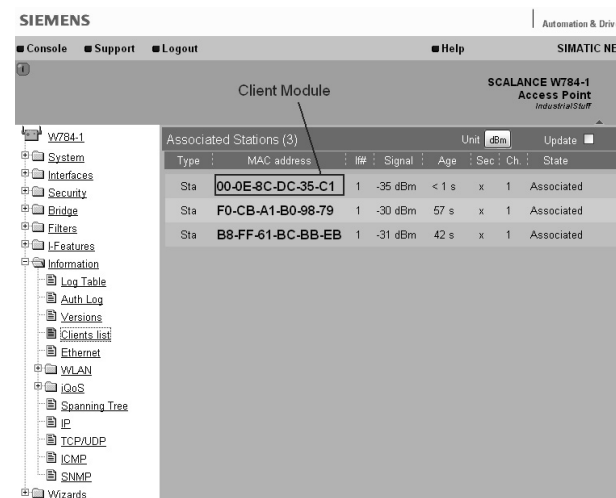
1. beállítások letöltése a Sinema-ból
2. beállítások feltöltése a Sinema-ba
3. firmware frissítés



7. ábra Jelerősség-szimuláció

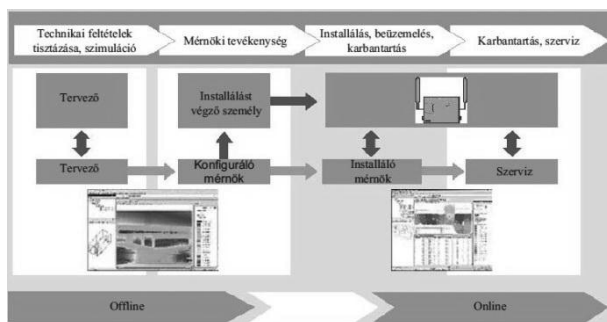
A program kirajzolja az általa számított jelerősségeket a szimulált területen. [2.]

Egy internetböngészőbe beírva az eszközök IP címét elérhetjük a webes beállítási felületet. Ez részletesebb, mint a Sinema által kínált beállításlista. Megnézhetjük az Access Point (192.168.2.100) klienseinek listáját. Itt láthatjuk a rá csatlakozó eszközök fizikai címét. (Kiemelve a Scalance Client Module)



8. ábra WEB interface

4. ÖSSZEKÉPÉS



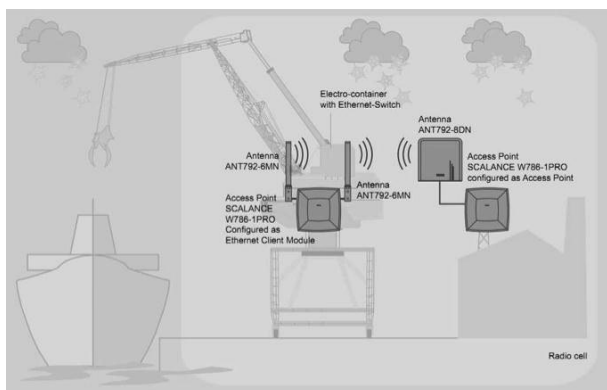
9. ábra IWLAN tervezési lépések

A kutatás során kidolgozott mérési, vizsgálati módszerek jól alkalmazhatók az ilyen rendszerek javasolt tervezési lépéseihez [3.]:

- A technikai kérdések tisztázása,
- a környezet feltérképezése,
- topológia, adatátvitel meghatározása,
- terjedési viszonyok meghatározása szimulációval,
- elemek kiválasztása, elhelyezése a szimulációs eredményekre alapozva,
- konfigurálás,
- üzembehelyezés,
- üzemeltetés, karbantartás.

A szimuláció eredményeinek összevetése a PLC-k közötti átvitel késési idejével még várat magára, ugyanis a küldő-fogadó program még fejlesztés alatt áll. Az alaprajzon be fogjuk jelölni a jellezgetes pontokat, és a szimuláció által generált hőterkép értékei mellett ábrázoljuk majd táblázatban. Ezeket görbével ábrázolva egymás mellett szemléletesen megjelennek majd az esetleges különbségek.

A 10. ábrán egy az időjárás viszontagságoknak kitett alkalmazási példa látható.



10. ábra Egy ipari alkalmazás az IWLAN-ra

A routing és az Ad-hoc megoldásokat nem teszteltük, mivel csak két IWLAN eszköz állt rendelkezésre, de a jelerősség vizsgálatához tökéletesen elegendő volt ez is.

Egy másik vizsgálati terület, amivel a jövőben foglalkozni fogunk, az a szimulációs értékek meghatározott pontokban végzett RF spektrumanalizátoros méréseiből adódó validációs módszer kidolgozása, valamint a nem várt zavaró körülményekre szimulációs módszer kidolgozása.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg

6. IRODALOM

- [1.] Siemens S7-1200 kézikönyv
- [2.] Sinema E kézikönyv
- [3.] Ajtonyi István: Ipari kommunikációs rendszerek IV., AUT-INFO Kft., Miskolc, 2011. június, 467 p., ISBN szám: 978-963-08-1516-1., ISSN szám: 1789-5456.

EGYEDI TERVEZÉSŰ TELEMETRIAI RENDSZER FEJLESZTÉSE BOSCH ELEKTROMOBIL JÁRMŰHÖZ, ZIGBEE VEZETÉK NÉLKÜLI HÁLÓZATON

UNIQUELY DESIGNED TELEMETRY SYSTEM DEVELOPMENT TO A BOSCH ELEKTROMOBIL VEHICLE ON ZIGBEE WIRELESS NETWORK

Méhes László, Trohák Attila***

ABSTRACT

This paper describes a telemetry system for an Elektromobil vehicle. The Bosch Elektromobil race is car race for engineer students. The mechanical design and construction are very important and also important the optimization of the complete car. This telemetry system is developed to be able to monitoring the live behavior of the Elektromobil vehicle. The system use a Microchip PIC microcontroller to handle the velocity, the RPM and the G force sensors and the on board dashboard. Furthermore the microcontroller sends these values on ZigBee wireless network to the Java data manager and dashboard application on the host computer. With the test results we could refine the vehicle behavior.

1. BEVEZETÉS

A Bosch Elektromobil verseny 2012-ben 4. alkalommal kerül megrendezésre. A verseny célja, hogy 4-5 főből álló főként egyetemista csapatok egy olyan járművet építsenek, amelyet a Bosch által gyártott akkumulátoros kéziszerszámok hoznak mozgásba. A versenyen több kategóriában indulnak a járművek, többek között egy körversenyen, egy gyorsasági versenyen, valamint a műszaki megoldásokat is díjazták. A 8 darab akkumulátoros kéziszerszám, mint meghajtó motor nem lebecsülendő, ugyanis a 2011-es versenyen a döntőbe került járművek sebessége elérte a 35-40 km/h-t is, és a 2012-es versenyen részt vevő járművek verseny sebessége előre láthatólag akár az 50 km/h-t is eléri!

Ilyen sebesség és a kiélezett verseny mellett nem csupán a gépészeti megoldások fontosak a jobb eredmények elérése érdekében. A korábbi évben is alkalmazott csapatunk egy fedélzeti fordulatszám-mérő és sebességmérő, amely nagymértékben hozzájárult a verseny eredményes teljesítéséhez, mivel hasznos segítséget nyújtott a pilóta számára. Ennek

továbbfejlesztéseként a 2012-es évben kidolgozásra kerül egy olyan ZigBee alapú vezeték nélküli szenzor hálózatot, amely egy telemetriai rendszer alapjaként szolgál és közel valós idejű adatokat szolgáltat a jármű pillanatnyi állapotáról.

2. HARDVER FELÉPÍTÉS

A járművön elhelyezett elektronika megtartotta eddigi műszerfal jellegét és továbbra is sebesség, illetve fordulatszám adatokat közöl a pilótával. Az aktuális sebesség kijelzésére két darab hét szegmenses kijelző került felhasználásra, illetve kilenc pár nagy látószögű LED-et a fordulatszám megjelenítésére a jobb láthatóság miatt. A kijelző szerkezetet egy MAX6957 típusú port bővítő és LED meghajtó áramkör vezérli, melyhez SPI porton keresztül lehet hozzáférni. Ezen LED meghajtó segítségével a kijelzésre használt eszközök eltérő fényerősségét is kompenzálni lehet, mivel minden egyes kimenetén áramgenerátor segítségével vezérelhető a LED-ek fényereje. Valamint a korábbi években a pilóta számára zavaróan nagy fényerőt sötétben, illetve az alig láthatóan kis fényerőt napsütésben is kompenzálni tudja, mivel minden kimenetére egyszerre alkalmazva is lehet áramerősséget állítani analóg módon. Amennyiben elég idő nyílik a verseny előtti felkészülésre, úgy ezen funkció is megvalósításra kerül.

A fedélzeti egység lelke egy PIC24HJ128GP204 típusú Microchip PIC mikrovezérlő. Ezen mikrovezérlő 16 bites működésű és 128 KB program memóriával bír, valamint 80 MHz-es működési frekvencia mellett 40 MIPS-es adatfeldolgozásra képes. Ez az egység fogadja a sebesség és fordulatszám szenzoroktól érkező jeleket, továbbá feldolgozza, majd megjeleníti azokat a fedélzeti műszerfalon a pilóta számára.

A fordulatszám, illetve sebességmérő szenzorokat egy-egy reflexiós optocsatoló alkotja. A sebesség mérése az első kerékre szerelt féktárcsák felületén történik, mivel ennek kialakítása éppen megfelelő ezen funkcióra.

* MSc. mérnök informatikus hallgató, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

** Egyetemi tanársegéd, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

A féktárcsán található lyukak szakítják meg a fény visszaturkózásához szükséges felületet, ennek köszönhetően generálódik egy külső megszakítás a mikrovezérlő számára. Az aktuális fordulatszám mérése is hasonló elveket követ. Itt a meghajtó eszközök behajtási pontján kerül sor a mérésre a tengelyen elhelyezett fényvisszaverő tükör segítségével.

A rendszer része még két darab háromtengelyű gyorsulásmérő szenzor is, melyeknek a vízszintes síkkal párhuzamos két tengelye került felhasználásra. Ezen szenzorok a járművet érő erőhatásokkal arányos analóg jelet szolgáltatnak, továbbá az autó orrában és a hátsó lökhárítóján kerültek elhelyezésre. A mikrovezérlő terhelésének csökkentése érdekében a gyorsulásmérő eszközök az XBee gyártmányú ZigBee rádiós chiphez csatlakoznak, amely képes négy darab analóg jel fogadására és automatikus továbbítására előre meghatározott időközönként. Továbbá a mikrovezérlőtől érkező sebesség és fordulatszám érzékelőkkel mért értékek továbbítására is ezen XBee modul szolgál. Az XBee modul és a mikrovezérlő közötti kapcsolat UART-on keresztül valósul meg.

Továbbá a fedélzeti egység rendelkezik egy GPS modullal is, mely a megfelelő jelszint illesztés után UART-on keresztül kapcsolódik a mikrovezérlőhöz. Majd a koordináta adatok beérkezése után a mikrovezérlő tovább küldi azokat az XBee modulon keresztül a számítógép számára. Ennek segítségével a bejárt pálya vonala és pályán belüli pozíciója is meghatározható a járműnek, azonban ennek pontosságát a sebesség és a műholdakra történő rálátás is nagymértékben befolyásolja.

A rendszer részét képezi egy számítógépi egység is, amely a számítógép USB portjára csatlakozik. Az egységen található USB-soros átalakító chip segítségével a számítógép számára egy soros portot emulál, melyre a számítógépi programból könnyedén kapcsolódni lehet. Ez az egység a fedélzeti párjához hasonlóan, szintén rendelkezik egy XBee vezetékek nélküli modullal, amelynek segítségével fogadja a járművön elhelyezett modultól érkező adatokat. Ennek köszönhetően a tesztelési és felkészülési időszakban a jármű működés közbeni paraméterei monitorozhatóvá válnak. Így a különböző gépészeti beállítások járműre gyakorolt hatásai összehasonlíthatóak. Ennek köszönhetően olyan részletekre és beállítási lehetőségekre derül fény, amelyekre eddig talán nem is gondoltak a mechanikai beállítások végzői. Ennek eredményeként jobban a versenykörülményekhez alkalmazkodó beállításokat lehet alkalmazni a járművön, hogy sikeresebb eredménnyel záruljon a verseny!

3. SZOFTVER FELÉPÍTÉS

A telemetriai rendszer részét képezi a számítógépes szoftver is, amely Java nyelven került implementálásra. Ezen alkalmazás felhasználja a JAVA XBEE API-t,

amely lehetővé teszi, hogy az USB-re csatlakozó XBee modul az emulált soros porton keresztül elérje a Java alkalmazás. Továbbá ennek segítségével az XBee modul nem csupán AT parancsokkal vezérelhető, hanem API parancsok segítségével is, valamint így a távoli modulok is vezérelhetővé, konfigurálhatóvá válnak.

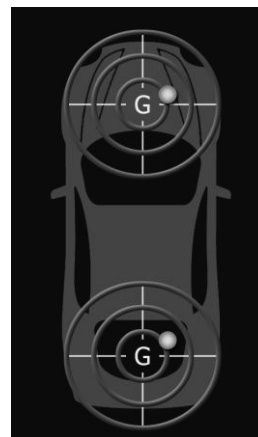
Továbbá ezen alkalmazás fogadja és feldolgozza a járműtől érkező sebesség, fordulatszám, G erő és GPS adatokat. A sebesség és fordulatszám megjelenítéséről két darab, a közúti forgalomban résztvevő autókól ismert megjelenésű analóg kijelző gondoskodik. Ezek élő képet adnak a jármű paramétereiről. Ezt az 1. ábrán lehet megtekinteni.



1. ábra. A fordulatszám és sebesség kijelző

Látható még az 1. ábrán, hogy visszajelzők is alkalmazásra kerültek, melyek a járművel való kapcsolatról adnak információt, illetve a jármű néhány további paraméteréről is, mint a kézifék állapota és az autó karosszériájának orrában elhelyezett LED logo működési állapota.

A G erő megjelenítése a 2. ábrán látható grafikus ábra segítségével történik. A célkeresztben mozgó narancs színű golyók az erőhatás irányát mutatják, elmozdulásuk ezzel az erővel arányos. Az autó orrában és hátulján elhelyezett gyorsulásmérők azt a célt szolgálják, hogy a jármű kanyar közbeni viselkedés is megfigyelhető legyen, így, ha eltérő értéket mutat a két érzékelő, abból következtetni lehet a jármű kicsúszására, alul-, vagy túlkormányozottságára is.



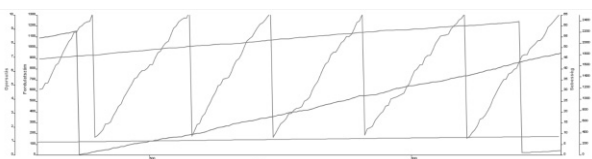
2. ábra. G erő kijelző

A GPS koordináták alapján számított pozíció adatok egy statikus térképen kerül megjelenítésre, mivel a tesztek és a verseny helyszíne is előre ismert, továbbá ezen helyszíneken kizárólag vezeték nélküli internetkapcsolat rendelkezésre állása esetén lehetne online térkép szolgáltatást elérni. A statikus műholdképek a Google Maps-ról kerültek felhasználásra az alkalmazásban. A térkép működés közben mutatott képét a 3. ábra szemlélteti.



3. ábra. Jármű pozíció a pályán.

Továbbá az adatok rögzítésre kerülnek a mérés során, melyet később vissza lehet nézni és utólag kielemezni. De nem csupán a fenti adatok monitorozhatóak, hanem az ezekből származtatható és becsülhető további adatok is. Ilyen például a gyorsulás, a megtett út hossza, egy kör ideje, a teljes táv hossz és ideje, a nyomaték, az akkumulátorok aktuális töltöttségi szintje, illetve további adatok is szükség szerint. Ezek egy kéttengelyes diagramon kerülnek megjelenítésre, melynek x tengelye az idő, y tengelye pedig színek szerint a kijelzendő érték. Erre egy teszt példát a 4. ábrán lehet megtekinteni.



4. ábra. Generált próbaadatokat a kijelzőn.

A túl sok adat által okozott zavaró hatás miatt, a megjeleníteni kívánt görbék egyesével kiválaszthatóak, továbbá a görbéhez tartozó segédvonalak is aktiválhatóak külön-külön. Ennek köszönhetően mindig pontosan annyi információ áll a felhasználó rendelkezésére, amennyire éppen szüksége van.

A tesztelés közben nyert adatokat és görbéket le is lehet menteni Excel formátumú táblázatokba, melyeknek a későbbi feldolgozása a bemutatott program alkalmazásától függetlenné válik. Ennek köszönhetően akár más gépészeti és matematikai számításokat végző

szoftverek számára importálható adatok kerülnek rögzítésre, melyekkel a beállításokra javaslatot tevő szakemberek könnyebben tudnak a továbbiakban dolgozni. Ezen opció az alkalmazás rugalmasságát nagyban növeli. Az Excel formátum kezeléséhez az alkalmazás a Apache POI-t használja, amely lehetővé teszi a Microsoft formátumok rugalmas kezelését Java nyelvről.

4. KEZDETI TESZT EREDMÉNYEK

Jelenleg még egyetlen kezdeti tesztet sikerült elvégezni. A kapott eredmények biztatóak. A tesztek a Miskolci Egyetem egyik parkolójában lettek elvégezve. A parkolóban egy kör megtételéhez csaknem 250 méter szükséges. A jármű a számítógéptől 120 méter távolságig távolodott el. Ez a távolság az rádiós kommunikáció legnagyobb áthidalandó távolságát is jelenti. Azonban a versenypálya kialakításának ismeretében ettől nagyobb távolságra és rosszabb optikai rálátásra kell számítani.

A teszt során, a járművön elhelyezett elektronikában egy 2,4 GHz-es XBee modul foglalt helyet, amely XBP24-ZB firmware-el volt ellátva. Ezen modul bot antennával volt felszerelve.

A számítógép oldali eszközben egy szintén 2,4 GHz-en üzemelő XBee modul volt megtalálható, azonban ez XB24-ZB típusú firmware-el került alkalmazásra és drót antennával volt felszerelve. A teszt közben a jelerősség értékek a fent leírt modul páros mellett jó értéket mutattak, még a legtávolabbi ponton is!

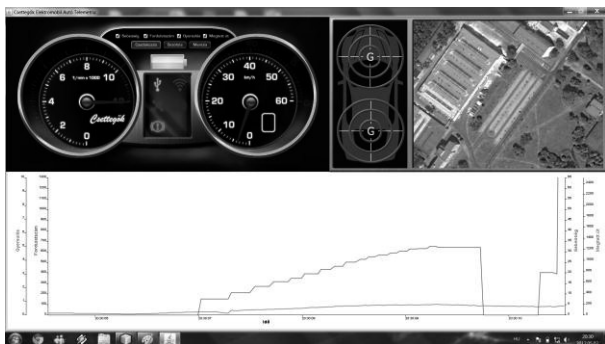
Az adatok beérkezésének sebessége is kiváló. A számítógépen időbélyeggel vannak ellátva a beérkező fordulatszám és sebesség értékek, ennek köszönhetően megállapítható, hogy a mikrovezérlő programjában beállított frissítési időt a számítógép oldalán megérkező adatok követik, az adatforgalom zavaroktól mentes. A mikrovezérlő által 0,03 másodpercenként küldött adatokat a számítógép oldalán legfeljebb 0,004 másodperces késéssel megérkeztek.

A későbbiekben sor kerül más modul típusokkal is a rendszer kipróbálására, habár az eddigi eredmények nem indokolják azt, hogy ezen a téren bármely változtatás is szükségszerű lenne.

A mérés során fény derült hibákra is. Mindegyik egyformán jelentkezett és következetesen. Minden egyes bukkánónál a teszt ideje alatt a sebesség- és főként a fordulatszám-mérő kijelzők értéke hirtelen megugrik. Erre a kivitelezés során elkövetett rossz kontaktok kialakítása adhat választ. Ennek orvoslása hamarosan megtörténik és remélhetőleg a hiba megszűnik.

Továbbá a sebességmérő kalibrálásából adódó helytelen megjelenítésre is fény derült, amit a következőkben szintén javítani fogok.

A teszt alatt készített egyik pillanatkép az 5. ábrán látható:



5. ábra. Első tesztkép az elindulás pillanatában.

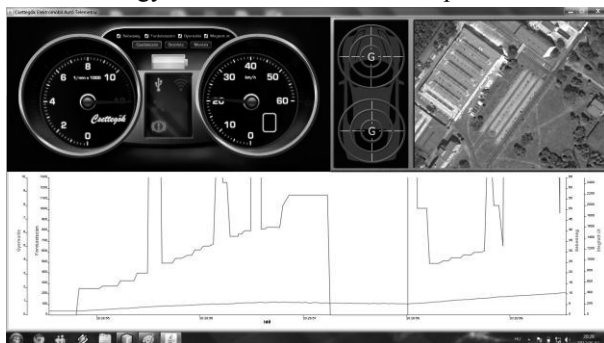
Az 5. ábrán jelzett diagramon a kék szín a fordulatszámot, a halványpiros szín a sebességet jelzi. A fentebbi analóg típusú kijelzők közül a bal oldali mutatja a fordulatszámot és jobb oldali a sebességet.

A további számított adatok nem kerültek megjelenítésre, hogy tisztább képet kaphassunk a működésről.

Az ábrán látható, hogy elindulás után a fordulatszám gyorsan emelkedik, majd a sebességváltás ideje alatt visszaesik. A későbbiekben pedig alacsonyabb fordulatszámmal újra indul és folytatódik a hajtás a második sebességfokozatban. Végül a fordulatszám értéke hirtelen megugrik. Ezen megugrás a korábban már említett üttesten elhelyezkedő egyenetlenségnek, valamint a hibás kontaktusoknak.

A sebesség eközben folyamatosan emelkedik, majd a fordulatszám nullára csökkenése alatt, vagyis a sebességváltás idejére elkezdi lecsökkenni. Majd mikor a következő fokozatban ismét elindul a hajtás, akkor a sebesség nagysága is fokozódni kezd. Az 5. ábrán látható értékek a vártak megfelelő eredményeket mutatják.

A 6. ábra egy hibás értéket mutató képet ábrázol:



6. ábra. Hibás értékek a kijelzőn.

A 6. ábráról leolvasható a korábban említett bukkanók által okozott fordulatszám érték kiugrások. Az eredmények az 5. ábrával hasonlatosak, a sebességváltás folyamatát mutatják második fokozatból harmadik fokozatba.

4. ÖSSZEZGÉS

A rendszer jelenleg tesztelés alatt áll, azonban a tesztelés közbeni használatra eddig csupán egyetlen lehetőség nyílt a jármű mechanikai és szerkezeti kivitelezésének késése miatt, így még nem kiforrott a rendszer működése.

Ezen cikk írásáig még a versenyre nem sor, így a verseny körülmények közötti használatról nem sikerült tapasztalatokat szerezni. A továbbiakban a verseny követelményeihez való szorosabb hozzáigazítása következik a telemetriai rendszernek és a jármű mechanikai beállításainak is.

A rendszer továbbfejlesztési lehetőségei adottak és egy későbbi projekt keretein belül sor kerülhet rá. Mind a grafikus felület sokoldalúbbá tétele, mind a mérhető értékek számának további szenzorokkal történő bővítése megoldást jelenthet egy eredményesebb verseny felkészüléshez a jövőben.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg a Miskolci Egyetem Automatizálási és Kommunikáció-technológiai Tanszékén.

6. IRODALOM

- [1.] MÉHES L.: Logisztikai rendszerek vezeték nélküli hálózatának GPS-alapú jelerősség mérése
- [2.] FALUDI R.: Building Wireless Sensor Networks, ISBN: 978-0-596-80773-3
- [3.] <http://poi.apache.org/>
- [4.] <http://code.google.com/p/xbec-api/>

AUTÓBUSZOK TÁVDIAGNOSZTIZÁLÁSA GSM HÁLÓZATON KERESZTÜL

REMOTE DIAGNOSTICS OF VEHICLES VIA GSM NETWORK

Biró Zoltán*, Kolozsi-Tóth Máté*, Trohák Attila**

ABSTRACT

In the paper, we introduce the results of our research in GSM-based remote diagnostics systems. The slow bitrate of the GSM network was the most problematic part. We had to decrease the amount of data. That has been solved by implementing some type of filter algorithms in a programmable controller.

1. BEVEZETÉS

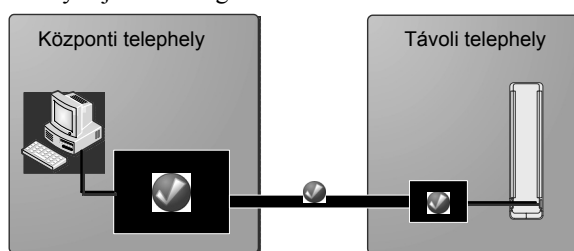
Jelen téma a Miskolci Egyetem és a Borsod Volán Zrt. együttműködése által 2010 folyamán kezdett kutatás folytatása. Első lépésben megvalósítottuk az Etherneten keresztüli távdiagnosztikát, mely eredményeként lehetővé vált a távoli telephelyeken levő autóbuszok CAN hálózata csatlakozni, motorparamétereket futásidőben megfigyelni, hibakódokat kiolvasni, törölni. A megoldás nagy előnye, hogy ezentúl az esetek nagy részében a járműdiagnosztikai szoftver kezeléséhez értő szakembereknek nem szükséges minden egyes hibaüzenet kiolvasáshoz elutazni az adott telephelyre. A miskolci központi irodából mindez gyorsan, egy telefonos egyeztetés után megtehető. Ennek köszönhetően jelentős idő és üzemanyagköltség megtakarítás érhető el. A megoldásunk részletei már korábbi cikkekben publikálásra kerültek. [4., 5.]

A kutatás következő lépésében egy olyan távdiagnosztikai megoldás kidolgozása a cél, mellyel folyamatosan nyomon követhetőek a járművek műszaki jellemzői egy központi számítógépen, melyen akár egy intelligens, prediktív elemző program karbantartási javaslatokat adhat a meghibásodások megelőzése céljából. Amennyiben mégis lerobban egy busz útközben, akkor annak hibakódjait, motorparamétereit ki tudnánk olvasni távolról, megkönnyítve ezzel a helyszínrre érkező szerelők munkáját.

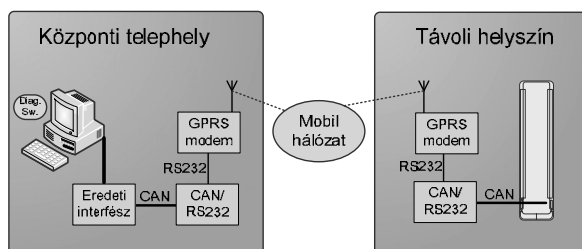
* MSc hallgató, Miskolci Egyetem

** egyetemi tanársegéd, Miskolci Egyetem Automatizálási és Kommunikáció-technológiai Tanszék

Például a hiba tudatában lehetőségük lenne eldönteni, hogy egy egyszerűbb szervizeléssel a jármű újra üzembe hozható-e, illetve milyen alkatrészt, szerszámot vigyenek magukkal, vagy esetleg vontatás szükséges-e a hiba körülményes javíthatósága miatt.



1. ábra Távdiagnosztika vezetékes hálózat segítségével



2. ábra Távdiagnosztika mobil hálózat segítségével

Az országúton levő autóbuszok és a telephely közötti adatkapcsolat létrehozására egy olyan rádiókommunikáció alapuló megoldást kellett találnunk, mellyel

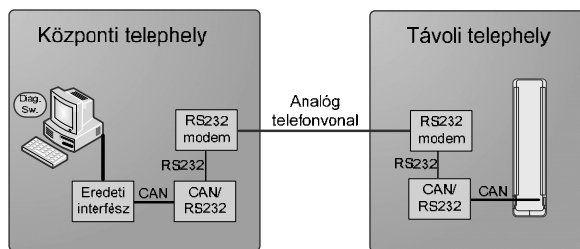
- áthidalhatók a nagy távolságok, illetve
- garantálja a megfelelő adatátviteli sebességet.

A távolságból eredő probléma nagyobb teljesítményű rádió adó-vevőkkel, műholdas adatkapcsolattal, vagy az országos lefedettséggel rendelkező GSM hálózatok alkalmazásával küzdhető le. Ha saját rádiót alkalmaznánk a megfelelő adóteljesítménnyel, az nagy valószínűséggel megsértené az erre vonatkozó törvényeket, előírásokat. A mikrohullámú műholdas kapcsolatok kialakítása igen körülményes a parabola antennák pozicionálása miatt, ráadásul igen költségesek is az ilyen típusú adatkapcsolatok. A

GSM hálózat alkalmazása viszont kézenfekvő megoldás a nagy lefedettség, illetve az egyre gyorsuló adatátviteli sebessége miatt, így ennek alkalmazása mellett döntötünk.

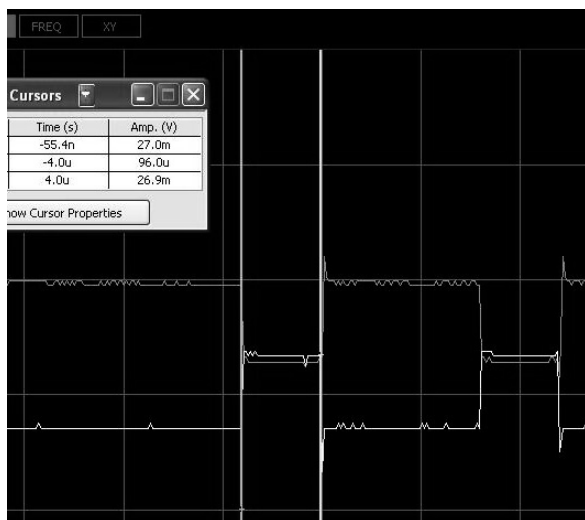
2. MÉRÉSEK

A megfelelő adatátviteli sebesség-igény meghatározása már nehezebb feladatnak bizonyult. Először megpróbáltunk egy transzparens átalakítást, mellyel ha működött volna a diagnosztizálás, akkor könnyebben megvalósítható lett volna a távoli helyről is. Két CAN/RS232 átjárót használtunk az átalakításhoz, azonban a jármű CAN hálózatához csatlakoztatott eszköz belső puffere nagyon hamar megtelt és hibát jelzett.



3. ábra Transzparens CAN/RS232-es átalakítás

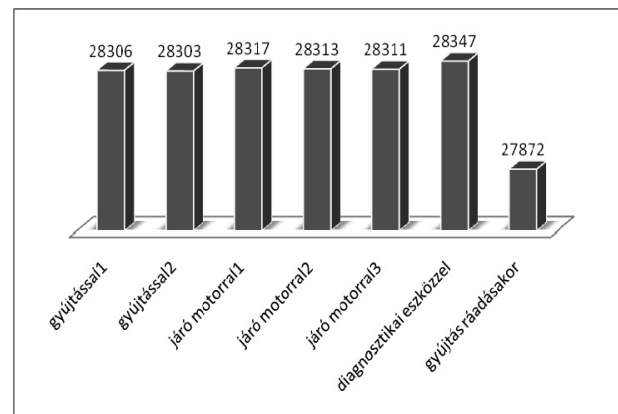
Ezután oszcilloszkóp segítségével meghatároztuk járművek CAN buszának adatátviteli sebességét. A mérésből megállapítható, hogy a névleges bit idő $4 \mu\text{s}$, ebből kiszámolhatjuk, hogy a közepes és nagy teljesítményű járművek belső buszrendszere 250 kbit/s-os adatátviteli sebességet használ. [1.]



4. ábra Oszcilloszkópos mérés

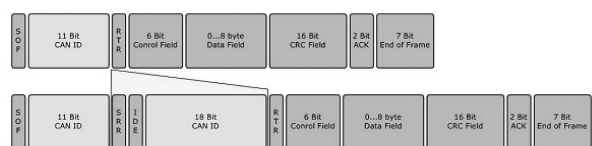
Ezen jellemző ismeretében arra következtethetnénk először, hogy a jármű CAN adatai teljes egészében csak a nagyobb sávszélességet biztosító távközlési technológiákkal lenne átvihető, mint például az EDGE, vagy 3G. Ezekkel viszont az a probléma, hogy a megfelelő lefedettséggel főleg a városok, települések területén vannak jelen, így az eredeti célra, mely az út-közbeni kapcsolat biztosítása, nem, vagy csak korlátozottan, egyes területeken lennének alkalmasak.

Ugyanakkor másik fontos jellemző a CAN busz átlagos üzenetsűrűsége. Ennek mérésére egy programozható, CAN porttal rendelkező eszközt alkalmaztunk, mellyel megszámláltuk az egységnyi idő alatt, a buszon levő üzenetek számát a jármű különböző állapotjaiban.



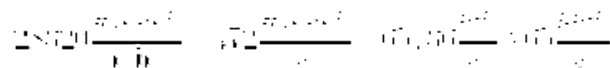
5. ábra Egyes állapotokban az üzenetszám egy perc alatt

A járművek belső buszrendszere a CAN 2.0B szabványt támogatja, ez annyiban különbözik a CAN 2.0A szabványtól, hogy a 11 bites azonosítók helyett 29 bites azonosítót tartalmaz egy üzenetkeret, ezáltal többfajta azonosítót tartalmazó üzenetek lehetnek a hálózaton. Egy normál CAN 2.0B-s üzenet 138 bitből áll, ezt mutatja a következő ábra is. [3.]



6. ábra A CAN 2.0A és 2.0B üzenetek

Ez alapján meghatároztuk azt a minimális átlagos átviteli sebességet, mellyel a CAN adatok átvihetőek lennének.

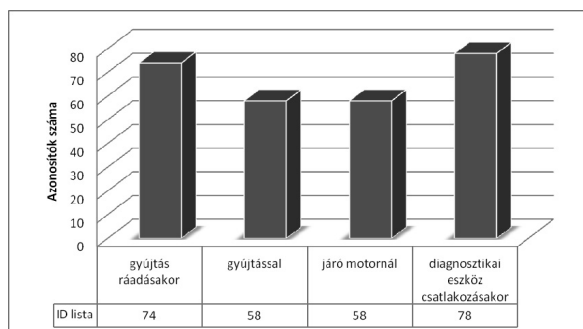


Ez még mindig többnek bizonyult, mint a hagyományos GPRS kapcsolat, mely rosszabb esetekben 20-30 kbps adatátviteli sebességet tesz lehetővé. [2.]

	azonosítók száma	üzenetszám
gyújtás ráadásakor	74	27872
gyújtással1	58	28306
gyújtással2	58	28303
diagnosztikával	67	28347
járó motor közben1	58	28317
járó motor közben2	58	28313
járó motor közben3	58	28311

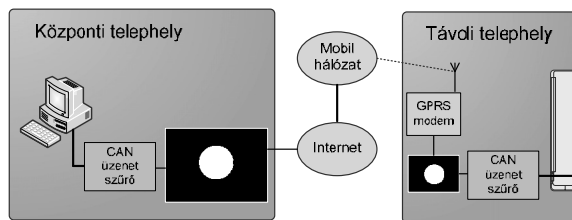
1. táblázat Az azonosítók és az üzenetek száma az egyes állapotokban

Azonban a CAN üzenetek elemzése után arra a következtetésre jutottunk, hogy a percnkénti ~28000 üzenet mindössze 60-70 különböző azonosítót tartalmazó üzenetből tevődik össze. Azaz például a motorvezérlő elektronika percnként több ezerszer elküldi néhány adatát a buszra (pl. fordulatszám), hogy más vezérlőegységek a megfelelő reakcióidővel fel tudják dolgozni azt, illetve reagálnak rá. Nekünk viszont a megfogalmazott céljaink eléréséhez, azaz a távmonitorozáshoz, illetve távdiagnosztikához felesleges ekkora adatsűrűség, így kidolgoztunk olyan eljárásokat, mellyel lecsökkenthető az átviteli sebesség a kívánt szintre.



7. ábra Az azonosítók száma az egyes állapotokban

3. SZŰRÉSI ELJÁRÁSOK



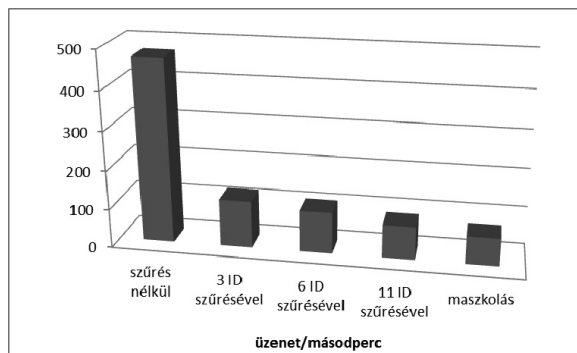
8. ábra Távdiagnosztika szűréssel

3.1. Leggyakoribb azonosítók szűrése

A leggyakoribb azonosítók az online paramétereket közvetítik, így ha csak a hibakódokat akarjuk átvinni, akkor ezeknek az üzeneteknek a kiszűrése egy lehetséges megoldás. A szűrést megvalósító kód minden egyes új üzenet beérkezésekor fut le, megvizsgálja, hogy az üzenet azonosítója benne van-e a nem kívánatos ID-k között. Ha igen, akkor nem továbbítja. Ha nincs benne, azaz az azonosítót át akarjuk engedni, akkor az üzenetet továbbítja.

3.2. Maszkolás

Emellett korábbi mérésekre alapozva a diagnosztikai eszközzel való kommunikációért felelős üzenetek azonosítóját vizsgáltuk meg. Az ilyen üzenetek kiszűréséhez maszkolást használunk, azaz az üzenet azonosítójának csak egy részét vizsgáljuk meg, és ha az benne van a megadott listában, akkor továbbítja az üzenetet.



9. ábra A szűrés és maszkolás hatása

A 3, 6 és 11 leggyakoribb azonosító kiszűrése és a maszkolás közben is megpróbáltunk csatlakozni egy diagnosztikai eszközzel a jármű hálózatára. A csatlakozás nehézkes volt, több időt vett igénybe, mint normál esetben, azaz amikor a diagnosztikai eszköz közvetlenül csatlakozik a jármű belső hálózatára. A csatlakozás után a hibakód olvasást el tudtuk végezni, de a kapcsolat instabillá vált,

többször megszakadt. Ennek pontos okait egyelőre még nem sikerült meghatározni, ez további mérések elvégzése után derülne ki.

	idő (sec)	üzenet szám	üzenet/sec
szűrés nélkül			472
3 ID szűrés	82,107	10000	121,7911
6 ID szűrés	93,262	10000	107,2242
11 ID szűrés	116,83	10000	85,59211
maszkolás	69,948	5000	71,48157

2. táblázat Az üzenetszám csökkenése

3.3. Minden azonosítót átengedünk, csak részarányosan megritkítjuk

Az előző két megoldás kizárólag a diagnosztikai üzenetek kiolvasására alkalmas. Egy következő még kifejlesztendő módszer az lesz, hogy minden egyes azonosítót átengedünk, azonban az igen gyakoriakat, amikből például több ezer darab is jön percenként, ritkítanánk, és csak bizonyos időközönként, vagy gyakoriságuk arányában engednénk át egyet-egyét. Ezzel még a maximális adatátviteli sebesség alatt lehetne maradni, viszont minden adatot megkaphatnánk a járműről. Ez egy lehetséges megoldás lenne az adatkapcsolat stabilizálására, illetve a távmonitorozás üzembiztos megvalósítására.

4. ÖSSZEFOGLALÁS

A kutatás kezdeti szakaszában jól működő megoldást találtunk a vezetékes, Ethernet hálózaton keresztüli távdiagnosztizálásra. A vezeték nélküli megoldás kialakításához a GSM hálózatot választottuk annak országos lefedettsége miatt. Ugyanakkor a korlátozott adatátviteli sebesség miatt akadályokba ütköztünk, melyek leküzdésére különböző módszereket dolgoztunk ki, melyekkel részleges sikereket értünk el. Ezek bizonyítják az elképzeléseink helyességét, ugyanakkor egy teljes értékű megoldáshoz még további mérésekre, illetve technikai problémák leküzdésére van szükség.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

6. IRODALOM

- [1.] AJTONYI I.: Ipari kommunikációs rendszerek IV., AUT-INFO Kft., Miskolc, 2011. június, 467 p., ISBN szám: 978-963-08-1516-1., ISSN szám: 1789-5456.
- [2.] AJTONYI I.: Ipari kommunikációs rendszerek I., AUT-INFO Kft., Miskolc, 2008. december, 431 p., ISBN szám: 978-963-06-5813-3, ISSN szám: 1789-5456
- [3.] J1939 szabvány
- [4.] TROHÁK, A., KOLOZSI-TÓTH, M., RÁDI, P., MÉHES, L., BIRÓ, Z.: The development of a remote diagnostic system for vehicles, Advanced Logistic Systems Volume 5., HU ISSN 1789-2198, 2011, pp. 216-220.
- [5.] BIRÓ Z., TROHÁK A.: Szűrési eljárások kutatása járművek GSM alapú távdiagnosztikai rendszerének kifejlesztése céljából, FMTÜ XVII. nemzetközi tudományos konferencia, ISSN 2067-6808, 2012.03.21., pp. 47-50.

LOKALIZÁCIÓS ALGORITMUSOK ÖNSZERVEZŐDŐ VEZETÉK NÉLKÜLI SZEZOR HÁLÓZATOKBAN

LOCATION DETERMINATION ALGORITHMS IN SELF-ORGANIZING WIRELESS SENSOR NETWORKS

Varga Attila K.*

ABSTRACT

Localization in wireless sensor networks is a main issue, i.e. to determine the position of a given device in the network. To give the position of the target node reference points have to be used. The node can calculate its distance and/or angle between itself and the reference points. In the 2D space, if a node knows its distance from three reference points its position can also be determined. One more reference point is needed in the 3D space to determine the current position of the target device. The positioning method should not increase the cost and complexity of a sensor because an application may require a lot of sensors. Communication and collaboration between nodes should be minimized for achieving energy saving. The paper shows well-known techniques used for localization in wireless sensor networks.

1. BEVEZETÉS

A vezetékes hálózatokat már jó harminc éve használjuk, a "vezeték nélkülség" viszont csak néhány éve került be a köztudatba, kecsegtető lehetőségei azonban rendkívüli ütemben növelik ismertségét és alkalmazását. Ma már rengeteg olyan termék van, amellyel olcsón kiépíthetjük akár otthoni vezeték nélküli hálózatunkat is, tehát ez a technológia már nem csak a nagy cégek, kormányzati és oktatási intézmények sajátja, hanem bárki számára elérhető.

A vezeték nélküli hálózatok az évek során több szabvánnyal is gazdagodtak. Napjainkban több vezeték nélküli kommunikációs szabvány áll rendelkezésre közepes és nagysebességű adatátvitelhez, hang, kép, videó továbbító, és számítógépes hálózatok kiszolgálásához, a kínálatból azonban nem rég még hiányoztak a szenzor és vezérlő egységek speciális igényeit kielégítő vezeték nélküli kommunikációs

szabványok. Ezek a rendszerek nem igényelnek nagy sávszélességet, de szükséges a rövid várakozási idő, látencia, az alacsony energiafelhasználás és a biztonságos kommunikáció. Természetesen az alacsony költségek is fontos paramétert jelentenek egy széles körben felhasznált vagy elterjeszteni kívánt szabványnál. szempontból is igen fontos, így ezt a területet kormányrendelet szabályozza. Hasonlóan kezelhetőek a különböző szolgáltató hálózatok, például villamos energia-, gáz-, hő-, vízellátás biztosítására szolgáló olyan objektumok, biztonsági berendezések, irányító alközpontok, ellenőrző egységek, kritikus hálózati elemek, amelyek időszakos felülvizsgálata, helyszíni ellenőrzése, karbantartása szükséges. [1]

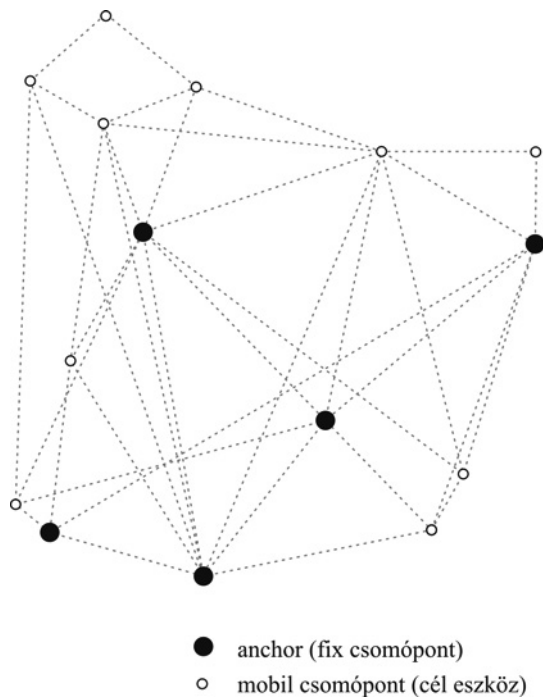
2. A LOKALIZÁCIÓ ALAPJAI

A szenzor csomópont (más néven mote – az USA-ban elterjedt elnevezés) gyakorlatilag egy olyan eszköz a vezeték nélküli hálózatban, amely képes adatfeldolgozásra, információ begyűjtésre, illetve kommunikációra a hálózatba kapcsolódó többi csomóponttal. A lokalizáció valamennyi vezeték nélküli szenzor hálózatra épülő alkalmazásban nagy jelentőséggel bír. A vezeték nélküli lokalizációs technikákat a mobil szenzorok nem ismert helyzetének meghatározására használják, felhasználva a rendszer ismert priori helyzeti információit. Ezek a technikák nem tekinthetők a hagyományos lokalizációs eljárások (GPS, radar) triviális kiterjesztésének. [2]

Ahhoz, hogy egy eszköz helyzetét kiderítsük, referencia pontok használatára van szükség. Az eszköz meghatározza a távolságot, a szöveget vagy mindkettőt önmaga és a referencia pont között, tehát a 2D-s síkban, ha egy eszköz ismeri a távolságát három referencia ponttól (melyek helyzete ismert), akkor meghatározható az eszköz helyzete. Ezzel szemben a 3D-s térben már 4 referencia pontra van szükség. Ugyanakkor, ha egy

* adjunktus, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

eszköz ismeri a helyzetét és a szöget egy referencia ponthoz képest, akkor szintén meghatározható a helyzete.

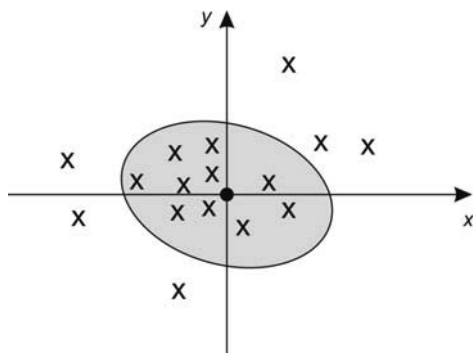


1. ábra. Önszervező vezeték nélküli szenzor hálózat elemei

A legtöbb ismert forgatókönyv szerint a felhasználó magával visz egy egyszerű szenzort, melyet mobil jeladónak, vagy cél eszközhöz is nevezünk, és ennek a jeladónak a helyzetét fix, ismert pozícióval rendelkező hálózati érzékelőkhez képest következteti ki. Ezeket a fixen telepített érzékelőket egyszerűen csomópontnak, horgonynak (anchor-nak) is nevezik.[3]

3. LOKALIZÁCIÓS ALGORITMUSOK

A lokalizációs technika kiválasztását sokszor az ár, a méret és az elvárt lokalizációs pontosság determinálja.



2. ábra. Lokalizációs pontosság szemléltetése

Az önszervező vezeték nélküli szenzor hálózatoknál alkalmazott lokalizációs technikák [4] vonatkozásában még nem történt olyan átfogó hatékonyság elemzés, mely révén a helyzetbecslés pontossága tovább fokozható a hagyományos számítási modellek javításával az idő és energiafelhasználás minimalizálása mellett, így a vezeték nélküli megoldás számos problémát vet fel a helyzetbecsléssel kapcsolatban:

- számos mérést kell elvégezni a helyzeti információk meghatározására,
- a lokalizációs technika megválasztása nagyban függ az adott környezeti viszonyoktól,
- a vezeték nélküli szenzorok általában olcsók, de korlátozott számítási képességekkel bírnak,
- a lokalizációs technikák az adott mérési lehetőségekhez kapcsolódó implementációk igényelnek, minimális hardver beruházással,
- a szenzor hálózatokat sok esetben nagyfokú multi-hop hálózatokban való használhatóságra kell tervezni.

A fenti problémák aktualitása miatt kiemelt fontosságúnak tartom a meglévő lokalizációs technikák hatékonyság elemzését, mely új modellek és algoritmusok alapjait teremtheti meg. A lokalizációs algoritmusokat három fő csoportba sorolhatjuk:

- tartomány-független,
- tartomány-függő, illetve
- csomópont független algoritmusok.

A tartomány-független algoritmusok esetén a csomópontok a szomszédjaik ismert helyzeti információi alapján becsülik meg saját helyzetüket. Itt feltételezzük, hogy nem minden csomópont rendelkezik távolság, szög vagy egyéb metrikus információval. Ez esetben a célcsomópont helyzete becsülhető a szomszédos fix csomópontok súlypontjával vagy a fennálló geometriai összefüggésekből is levezethető. [5]

A tartomány-függő algoritmusok távolságbecslést végeznek a fix és a mobil csomópontok között. A hálózati csomópontok egy részének ismert a helyzete, valamilyen helyi mérés vagy beépített GPS vevő révén. A cél csomópontok relatív módon próbálják megbecsülni helyzetüket a fix csomópontokhoz képest. Olyan technikákat alkalmaznak, mint például a vett jelerősség alapján (Received Signal Strength), a jel beérkezésének időtartama alapján (Time of Arrival of Signals) vagy a beérkező jel szöge alapján (Angle of Arrival of the Signals) becslik a távolságot, ehhez viszont a fix csomópontok helyzetét nagy pontossággal kell ismerni. [6][7]

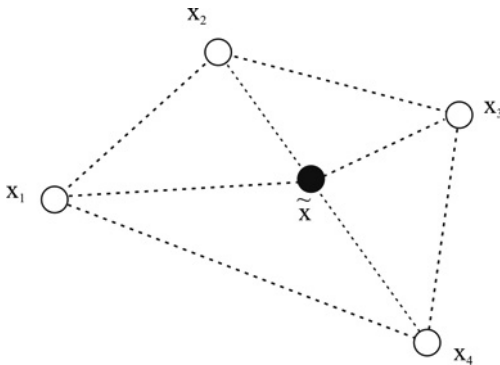
3.1. Súlyozott súlyponti lokalizáció

A súlyozott súlypont ötlete, hogy a cél eszköz helyzetét az átviteli zónában lévő fix csomópontok ismeret helyzetéből kalkulált átlag alapján becslik. Egyszerűsége és hatékonysága miatt előszeretettel alkalmazzák sűrű szenzor hálózatokban, ahol

feltételezzük a fix csomópontok jelenlétét – így ezek pozíciója ismert –, valamint egymást átlapoló átviteli zónákból épül fel. A fix csomópontok helyzete ezen felül súlyozható. Az ismeretlen pozíció a következőképpen számítható:

$$\tilde{x} = \frac{\sum_{i=1}^N \omega_i x_i}{\sum_{i=1}^N \omega_i}, \quad (1)$$

ahol N a fix csomópontok száma az átviteli zónában, x_i az i -edik fix csomópont helyzete, ω_i a súlyokat jelöli. Egy alkalmasan megválasztott súlyozási módszerrel finomítani lehet a pozícióbecslést, mely pontosabb helyzet meghatározó információkat eredményez. A leggyakoribban alkalmazott technika, hogy a mobil csomópontok broadcast jelleggel kiküldenek egy üzenetet.



3. ábra. Súlyponti lokalizáció

A fix csomópontok folyamatosan figyelik a jeladókat, és a beérkező üzenet alapján egy kapcsolódási metrikát alakítanak ki. Ez esetben a súlyozás a következőképpen számítható ki:

$$\omega_i(t) = \frac{n_{vett}^{(i)}(t)}{n_{küldött}^{(i)}(t)}, \quad (2)$$

ahol t a broadcast üzenetek fogadásának időtartama, $n_{vett}(t)$ és $n_{küldött}(t)$ pedig a kapott és küldött üzenetek száma t idő alatt. Ezt követően pedig, egy meghatározott arány - általában 90 % - felett produkáló csomópontokat veszik figyelembe a számításánál.

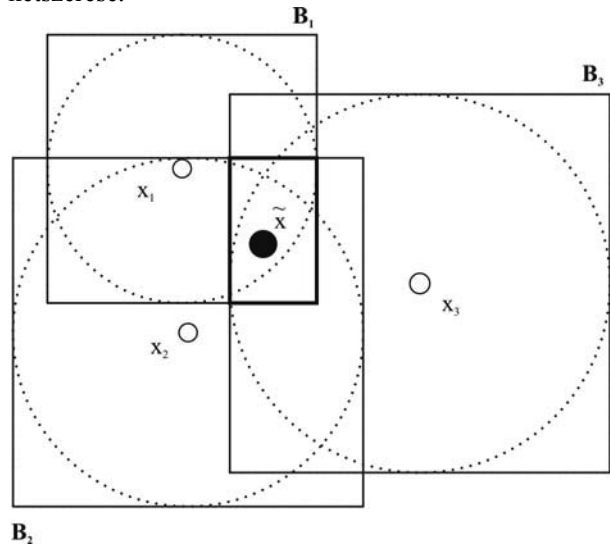
3.2. Befoglaló doboz

A befoglaló doboz egy egyszerű, számítás hatékony lokalizációs technika, ahol a fő koncepció, hogy minden egyes fix csomópont köré konstruált befoglaló doboz által meghatározott metszetben helyezkedik el a cél eszköz. Az i -edik burkoló doboz oldalának hosszával

megközelítőleg becsülhető, hogy milyen távolságra helyezkedik el a cél eszköz. A helyzetbecslés eredménye tehát a metszetként kialakuló határoló doboz, vagy ennek a közepe, azaz

$$\tilde{x} \in \left\{ \bigcap_i B_i \right\}, \quad (3)$$

ahol B_i az x_i csomópont köré létrehozott négyzet alakú doboz, melynek az oldala a hatókör sugarának a kétszerese.



4. ábra. Befoglaló doboz

A keresett burkoló doboz

$$B = \left\{ \bigcap_i B_i \right\}$$

geometriája, azaz alakja a csomópontok topológiájától függ. Sűrű szenzorhálózatban a burkoló doboz mérete lekorlátozódik, és a becsült pozíció jól konvergál a valódi pozícióhoz. Bár ezen technika pontossága jellegénél fogva korlátozott, mégis nagyon egyszerű és gyorsan implementálható és futtatható a csomópontokon. Ugyanakkor ezzel a technikával kalkulált pozíció jól használható egy kiinduló becslésként, melyet rekurzív módon más technikákkal lehet finomítani.

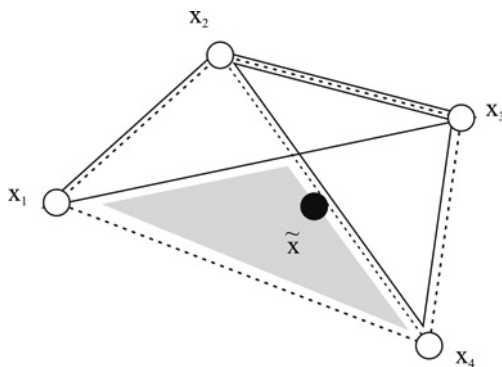
3.3. Pont a háromszögben (PiT)

A PiT algoritmus is egy tartomány független lokalizációs séma. Ebben a megközelítésben a cél eszköz küld egy jeladó üzenetet, amit a fix csomópontok vesznek. Ezt követően háromszög alapú kommunikációs tartományokat alakítanak ki, azaz a csomópontok által meghatározott összes lehetséges 3-as alhalmazt hoznak létre:

$$\begin{aligned}\tilde{x} \in T_1 &= \{x_1, x_2, x_3\} \\ \tilde{x} \in T_2 &= \{x_1, x_2, x_4\} \\ \tilde{x} \in T_3 &= \{x_1, x_3, x_4\} \\ \tilde{x} \in T_4 &= \{x_2, x_3, x_4\}\end{aligned}\quad (4)$$

ahol T_i az i -edik alhalmaz, mely egy háromszöget határoz meg. Minden egyes háromszög esetén meg történik a PiT teszt azon célból, hogy kiderüljön, melyik tartományban található a céleszköz. Végül a cél eszköz pozícióját az összes olyan háromszög alakú terület metszete fogja adni, melyek tartalmazzák az eszközt:

$$T_{cél} = \bigcap_{i=1}^4 T_i. \quad (5)$$



5. ábra. Pont a háromszögben alapú lokalizáció

A legfőbb probléma ezzel a technikával, hogy meg kell vizsgálni, egy adott csomópont a vizsgált háromszögben helyezkedik el vagy sem. A PiT teszt gyakorlatilag geometria alapokon nyugszik.

4. ÖSSZEFOGLALÁS

Manapság az intelligens szenzorok kínálja kommunikációs lehetőségek széles skálájával találkozhatunk az élet bármely területén. Alapvető probléma a vezeték nélküli szenzor hálózatban a lokalizáció, azaz egy eszköz helyzetének felderítése, meghatározása. Egy alkalmasan megválasztott súlyozási módszerrel finomítani lehet a pozícióbecslést [9], pontosabb helyzet meghatározó információk révén pedig gyorsabban megy végbe az önszerveződés a szenzor hálózatban belül, ami energia megtakarításhoz vezet. Ezzel a telepélettartam növelhető, tehát az adott rendelkezésre álló energiakapacitás hatékonyabban használható ki, ezért gazdaságosabbá válhat az önszervező szenzor hálózat üzemeltetése.

KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

IRODALOM

- [1] T.S.Rappaport, Wireless Communications: Principles and Practice. Prentice Hall: New Jersey, 1996.
- [2] P. Bahl and V. Padmanabhan, "RADAR: An in-building Rfbased user location and tracking system," in *Proc. Of Infocom' 2000*, Tel Aviv, Israel, Mar. 2000, vol. 2, pp. 775–584.
- [3] J. Blumenthal, F. Reichenbach, and D. Timmermann, "Position estimation in ad hoc wireless sensor networks with low complexity," in *Joint 2nd Workshop on Positioning, Navigation and Communication 2005 and 1st Ultra-Wideband Expert Talk*. Shaker Verlag, March 2005, Workshop, pp. 41–49, Hannover, Germany.
- [4] Bischoff, U., Strohbach, M., Hazas, M., Kortuem, G.: Constraint-based distance estimation in ad-hoc wireless sensor networks. In: *Proceedings of the Third European Workshop on Wireless*
- [5] He, T., Huang, C., Blum, B.M., Stankovic, J.A., Abdelzaher, T.: Range-free localization schemes for large scale sensor networks. In: *MobiCom '03: Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, pp. 81–95. ACM, New York,
- [6] Niculescu, D., Nath, B.: Ad hoc positioning system (APS) using AoA. In: *Proceedings of INFOCOM (2003)*
- [7] X. Shen, Z. Wang, P. Jiang, R. Lin, and Y. Sun. Connectivity and RSSI based localization scheme for wireless sensor networks. In *Advances in Intelligent Computing*, pages 578–587, 2005.
- [8] Nakamura, E.F., Loureiro, A.A.F.: Information fusion in wireless sensor networks. In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 1365–1372. ACM, New York, USA (2008)
- [9] A. Savvides, C.-C. Han, and M. B. Strivastava. Dynamic fine-grained localization in Ad-Hoc networks of sensors. In *Proc. of the Int. Conference on Mobile Computing and Networking (MOBICOM)*, pages 166–179, 2001.

A MISKOLCI EGYETEM PROGRAM HÁLÓ FELÜLETÉNEK FEJLESZTÉSE

DEVELOPMENT OF THE PROGRAM NETWORK INTERFACE AT THE UNIVERSITY OF MISKOLC

*Varga Attila K.**

ABSTRACT

Based on existing bilateral contracts and having experience in European Community programmes, the University of Miskolc in 1999 submitted its first Leonardo mobility project in order to provide its students and young graduates with practical placement opportunities in the EU. The project was completed successfully and has been followed by several others. Not only the number of mobilities grows year by year, but also the number of partner host organisations, that provide the beneficiaries with placement offers. In the recent years the University of Miskolc has established strong relations in the frame of the Leonardo Programme in Germany, Austria, Belgium, Italy, UK and Spain, and has also arranged mobilities in Sweden, Portugal, the Netherlands and Finland. In 2012, an online interface was developed in order to modernize the process of registration as well as to assist the administration and monitoring tasks. The aim of the paper is to introduce the mobility programs and the recent developed online Program Network Interface.

1. BEVEZETÉS

Az 1996-ban létrehozott Tempus Közalapítvány a Nemzeti Erőforrás Minisztérium felügyelete alatt működő, kiemelten közhasznú szervezet, amely az általa kezelt programokon keresztül a legnagyobb mértékű mobilitást bonyolítja le Magyarországon. A Tempus Közalapítvány évtizedes szakmai múlttal rendelkező, dinamikus és munkavégzésében igényes, kiemelten közhasznú szervezet, amelynek célja a közös európai értékek és célok képviselése és közvetítése az oktatás, képzés és K+F területén, valamint a magyar oktatási, képzési és K+F rendszer megismertetése és érdekeinek képviselése nemzetközi környezetben. A Tempus Közalapítvány végzi az Európai Bizottság Egész életen át

tartó tanulás programjának és a közép-európai CEEPUS programnak a magyarországi koordinációját. A Tempus Közalapítvány igen összetett, nemzeti és nemzetközi oktatási-képzési pályázati programokat kezel, különböző képzéseket nyújt az EU támogatások felhasználása témakörében, illetve tudásközpont szerepet tölt be. [1]

Magyarország 1997 óta vesz részt a Leonardo da Vinci programban, azóta támogat mobilitási programokat is, amelyek középiskolások, felsőfokon tanulók, fiatal dolgozók külföldi szakmai gyakorlatát, valamint a szakképzésben részt vevő oktatók külföldi tanulmányait segíti.

A Miskolci Egyetem által benyújtott Erasmus University Charter-t az Európai Bizottság elfogadta, így egyetemünk részt vehet az Integrated Life-Long Learning Programme/Erasmus programban. A Miskolci Egyetem 1999-től vesz részt a programban, elsősorban mobilitási projektekkel. Az oktatói, személyzeti és hallgatói cserék olyan egyetemek között jöhetnek létre, amelyek kétoldalú keretszerződést kötöttek egymással. Az Erasmus program a felsőoktatásnak teremt pályázati lehetőségeket. Az Erasmus az Európai Bizottság egyik legsikeresebb és legismertebb programja, mely mostanáig több mint 2,3 millió egyetemi hallgató mobilitását segítette elő Európában. Az egykori Erasmus hallgatók szerint az Erasmus tapasztalat a végzés utáni munkakeresést is megkönnyítette számukra.

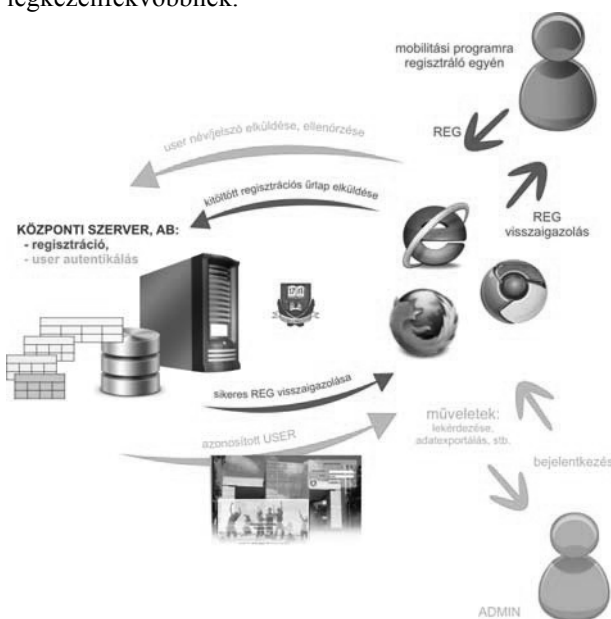
A Leonardo da Vinci program az Európai Bizottság 1994-ben indult együttműködési programja a szakképzés területén. 2007-től a Leonardo da Vinci program az Egész életen át tartó tanulás Program alprogramjaként működik tovább. A Leonardo mobilitások elsődleges célja az, hogy a leendő kedvezményezettek számára magas színvonalú külföldi szakmai gyakorlat lehetőségét biztosítsa. Ennek érdekében a Leonardo program segít az ilyen jellegű szakmai gyakorlat megszervezésében és finanszírozásában. Kedvezményezettek lehetnek többek között a felsőoktatásban végzett frissdiplomások is, mint fiatal munkavállalók.[2]

* adjunktus, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

2. A MISKOLCI EGYETEM PROGRAM HÁLÓ FELÜLETÉNEK FEJLESZTÉSI KONCEPCIÓJA

A Miskolci Egyetemen futó mobilitási programokra való jelentkezéshez, az online nyilvántartáshoz, az egyszerűbb nyomon követhetőség és az adminisztratív munka megkönnyítése érdekében egy elektronikus nyilvántartó rendszer kialakítása volt a cél, mely kapcsán elsődleges szempont volt, hogy online módon bárhol elérhető legyen.

A mai programozási trendeket, illetve az ergonomikus ember-gép interfész kapcsán felmerülő felhasználói igényeket figyelembe véve, ezen kritérium kielégítésére a webes fejlesztés tűnt a legkézenfekvőbbnek.



1. ábra. Alapkonceptió

A Miskolci Egyetem Program Háló rendszerének (továbbiakban: **MEPH**) legfőbb feladata:

- a mobilitási programokat bemutató, aktuális információkat tartalmazó, egységes, fiatalos arculatú webes felület kialakítása,
- a programokra való online jelentkezés lehetőségének biztosítása, felváltva ezzel az eddigi papír alapú regisztrációs folyamatot,
- a jelentkezettek nyilvántartása, az ehhez kötődő járulékos adminisztratív teendők, későbbi nyomon követésük és archiválási lehetőségeket biztosító adminisztrációs felület fejlesztése.

A MEPH fejlesztésének legfőbb célja tehát a papír alapú regisztráció és adminisztráció kiváltása, az elektronikus formátumban történő archiválás elősegítése, valamint az, hogy a regisztrált adatok

adatbázis alapon is tárolásra kerüljenek, biztosítva az alábbiakat az adminisztratív ügyintézők számára:

- online módon elérhető rendszer (autentikált),
- adatok archiválása, tárolt adatok exportálása,
- adatmódosítás,
- adattörlés,
- regisztráció során feltöltött dokumentum menedzselése,
- tárolt adatok adott kritérium(ok) szerinti szűrése, keresése,
- szűrt adatok exportálása,
- járulékos funkciók:
 - o userek, jogkörök menedzselése,
 - o felhasználói jelszó változtatási lehetősége,
 - o felhasználói műveletek naplózása.



2. ábra. Bejelentkezés a MEPH adminisztrációs felületére

3. A MEPH FELÜLETÉNEK FEJLESZTÉSE

3.1. A fejlesztőkörnyezet bemutatása

A nyilvántartó rendszer fejlesztése PHP alapon történt MySQL támogatással, mely révén az adatok adatbázistáblákban kerülnek letárolásra a Miskolci Egyetem központi szerverén (a működés elvét az 1. ábra mutatja be). A MEPH működéséhez szükséges infrastruktúrát tehát egy egyetemi kiszolgáló szerver biztosítja az alábbiak szerint:

- PHP modul: a szerver oldali PHP alapú programkódok futtatásához,
- MySQL modul: centralizált adattárolás, szűrési, keresési funkciók megvalósítása.

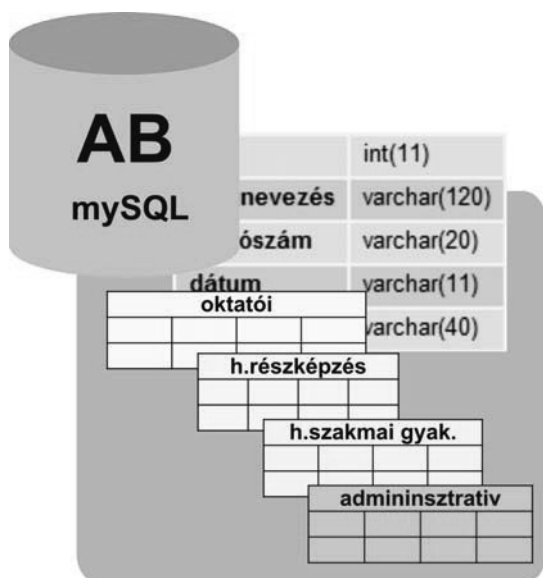
A PHP (Hypertext Preprocessor) egy olyan nyílt forráskódú számítógépes szkriptnyelv, mely szerver oldalon az operációs rendszerek bármelyikén képes

futni, a legtöbb kiszolgálóprogrammal együttműködve. Legfőbb felhasználási területe a dinamikus weboldalak készítése. [3]

A MySQL egy nagyon elterjedt adatbázis kezelő rendszer, amely egyszerűségéről és hatékonyságáról híres. Az egyszerűsége többek közt annak is köszönhető – ahogyan azt a neve is jelzi –, hogy SQL parancsokon alapszik. Az SQL (Structured Query Language) szabványosított nyelvezete segítségével a különböző típusú adatbázisokat azonos módon kezelhetjük. [4]

3.2. Archiválásra kialakított AB struktúra

Az archiválás online módon elérhető központi adatbázisban történik, melyben a regisztráció során megadott adatok kerülnek letárolásra adatbázis táblákban. A regisztrált jelentkezők adatainak adatbázisban való tárolásához egyértelműen definiálni kellett, hogy melyek azok az információk, amelyeket le kell tárolni az általános keresési és összetettebb szűrési funkciók biztosításához.



3. ábra. Adattároláshoz használt AB struktúra

Az adatmezőkkel kapcsolatban az alábbi járulékos paraméterek is rögzítésre kerültek:

- **adatmező típusa:** szám, szöveg, logikai, dátum, stb.
- **adatmező hossz:** hány karakternyi hossz szükséges a tároláshoz,
- **adatmezőhöz kötődő integritási feltétel:** belső szabályrendszer, a letárolt adatok helyessége (lehet-e például üresen hagyni az adott adatmezőt a rögzítés során, elsődleges kulcsról van-e szó, stb.).

A táblázat szerkezete, sémája tehát a táblákhoz tartozó mezőkkel lettek leírva. Azaz, definiálva lettek, hogy milyen mezőkből épülnek fel a táblázatok, ezzel egyértelműen megadva az AB táblák szerkezetét is. A mezők megadása a mező névének és a mező adattípusának a kijelölésével történt. Mivel a szerkezetek megadása nem azonosít egyértelműen egy táblázatot, hiszen több táblázat is létezhet ugyanazzal a szerkezettel, másrésztől a szerkezet leírása igen hosszadalmas, ezért minden táblázat kapott egy egyedi azonosító nevet az adatbázison belül. Ezzel a névvel egyértelműen lehet azonosítani a táblázatokat a műveletek során. A tábla névnek tehát az adatbázison belül, a mezőnévnek pedig a táblán belül kell egyedinek lennie a különböző struktúrát igénylő adathalmazok nyilvántartásához. A mobilitási programokra négyféle kategóriában lehetséges jelentkezni, regisztrálni:

- hallgatói részképzés,
- hallgatói szakmai gyakorlat,
- oktatói mobilitási programok,
- adminisztratív személyzet.

Tekintettel arra, hogy négyféle regisztrációhoz kötődő tárolást kell megoldani az elektronikus nyilvántartó rendszerben, így a tárolás és keresés funkciók meggyorsítása érdekében célravezető volt a különböző regisztrációs típusokat külön táblákban tárolni. Ezzel a keresés gyakorlatilag gyorsítható, mivel a lefutó adatbázis-kezelő parancsban a keresési kritériumokkal definiált lekérdezés csak az adott típusú regisztrációkat tartalmazó táblán fog végrehajtódni, és nem a teljes adatbázison. Tehát egy ömlesztett adathalmaz helyett, egy típus szerint szétválogatott adatokat tartalmazó táblákban történik a keresés.

Az évenkénti adatbázistáblákban való tárolás – azaz a regisztrációs típusok továbbbontása évenként strukturált AB táblákban – megvalósítását a MEPH-et kezelő adminisztratív jogkörrel rendelkező felhasználókkal egyelőre nem láttuk indokoltnak. Ezen irányú fejlesztést az egy év alatt felhalmozódó adatmennyiség ismeretében kívánjuk megfontolni. Nagy adatmennyiség esetén már tárgyév alapú tárolásában is érdemes gondolkodni, mivel így a keresési funkciók végrehajtása még hatékonyabbá tehető. Ezzel gyakorlatilag a tárgyév szerinti keresés/rendezés kérdése is megoldódik, mivel év szerinti lekérdezés esetén csak az adott évnek megfelelő táblá(k)ban kell az adott szelekciós utasítást lefuttatni.

Ömlesztett adathalmaz – ugyanazon tábla használata valamennyi év vonatkozásában – használata nagymértékben megnövelné a háttérben futó szelekciós lépések számát, mely a keresési idő megnövekedését eredményezné.

4. MEPH FELHASZNÁLÓ FELÜLET

4.1. MEPH információs felület

A Miskolci Egyetemen futó mobilitási programokra vonatkozó általános információkat, regisztrációkra vonatkozó hasznos tudnivalókat a MEPH honlapja szolgáltat (www.meph.uni-miskolc.hu). A jelenleg futó mobilitási programok vonatkozásában egy-egy figyelemfelkeltő link található a kezdőoldalon, mely az egyszerű kezelhetőséget és elérhetőséget célozza.



4. ábra. MEPH információs felület

4.2. MEPH regisztrációs felület

A mobilitási programokra való online jelentkezéshez kifejlesztett regisztrációs felület az alábbi linken érhető el: <http://www.meph.uni-miskolc.hu/regisztracio/{rt}.php>.

Az {rt} kifejezés a regisztrációs típust jelöli:

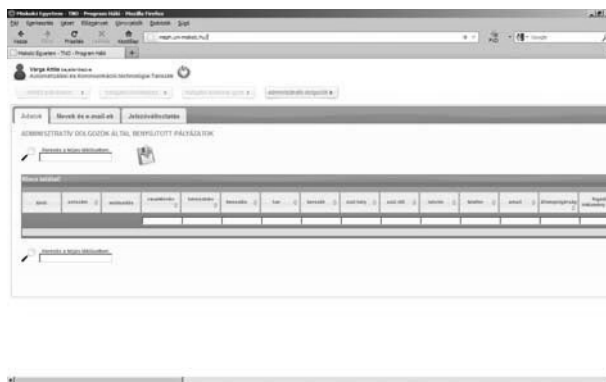
- oktatói:
{rt} = erasmus_oktatoi,
- hallgatói részképzés:
{rt} = erasmus_rk,
- hallgatói külföldi szakmai gyakorlat:
{rt} = erasmus_kszgy,
- adminisztratív személyzet mobilitása:
{rt} = erasmus_admin.

4.3. MEPH adminisztrációs felület

A felhasználói profil a felhasználói jogkörök szabályozása céljából kétféle lehet, mely a user regisztrálása során rögzítésre kerül:

- ADMIN jogosultságú user (teljes jogkör),
- korlátozott jogosultságú user (csak adatlekérdezésre jogosult az adott hatáskör alapján).

A bejelentkezést követően a négyféle regisztrációs típusnak kialakított adattábla közül kezelni kívánt kiválasztását követően nyílik lehetőség az adatok menedzselésére (jogkörnek megfelelően).



5. ábra. MEPH adminisztrációs felület

5. ÖSSZEFOGLALÁS

A felhasználóbarát felület fontossága a fejlesztés során kiemelt prioritást kapott, mivel a külakak és a user-ek elvárásainak összekapcsolása egy kényelmes, egyszerűen kezelhető rendszert eredményezett, megkönnyítve ezáltal a felhasználók munkáját, illetve csökkentve az adatbeviteli és adatkezelési hibalehetőséget. A fejlesztést indukáló célok (online elérés, több évre visszamenőlegesen nyomon követhető nyilvántartás, papír alapú archiválásról elektronikusra való áttérés) maradéktalanul teljesültek a felület kifejlesztésével, mely mind a mobilitási programokra jelentkezők, mind pedig az adminisztratív teendőket végzők számára egy könnyen kezelhető rendszert biztosít. A továbbiakban a felhasználói igényeknek megfelelő módosítások, további kényelmi funkciók, valamint a még gyorsabb és hatékonyabb működésre irányuló fejlesztések lettek megcélözva.

KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

IRODALOM

- [1] TEMPUS Közalapítvány honlapja: www.tpf.hu
- [2] A Miskolci Egyetem PROGRAM HÁLÓ felülete: www.meph.uni-miskolc.hu
- [3] P. MOULDING: PHP Haladóknak, 2002., Perfact-Pro Kft., ISBN 9789630095587
- [4] SÁGI G.: Webes adatbázis-kezelés MYSQL és PHP használatával, 2005., BBS-INFO Könyvk. És Inform. Kft., ISBN 978 963 942 504 0

VEZETÉK NÉLKÜLI IPARI KOMMUNIKÁCIÓS RENDSZEREK RF SPEKTRUMÁNAK MÉRÉSE

THE MEASUREMENT OF RF SPECTRUMS OF WIRELESS INDUSTRIAL COMMUNICATION SYSTEMS

Trohák Attila , Haluska Zsolt***

ABSTRACT

The spectrum analyzer plays an important role in the examination of industrial wireless communication systems. The spectrum analyzer can help us to install a wireless communication system, and to choose the correct channel and the right operation place. It can also play an important role in diagnostics to determine the interference sources.

1. BEVEZETÉS

Miután az IEEE megalkotta a 802.11-es szabványcsaládját, a vezeték nélküli kommunikáció szabványosítása következtében a vezeték nélküli kommunikáció egyre jobban terjedni kezdett. A mobil eszközök, laptopok terjedésével a vezeték nélküli kommunikáció használata gyorsan elterjedt, mivel nem szükségesek vezetékek, kényelmes a használata. Nemcsak az irodai és otthoni számítógépes hálózatok kialakításában terjedt el, hanem különböző ipari alkalmazási területeken is egyre gyakrabban alkalmazzák. Az ipar számára az egyes gyártók saját implementációikkal egészítették ki az IEEE által megalkotott szabványt, a különböző ipari alkalmazási területek igényeinek megfelelő vezeték nélküli kommunikációs rendszert dolgoztak ki. Ezek a célorientált rendszerek előszeretettel kerültek alkalmazásra az ipar egyes területein, mivel jelentős költségek takaríthatóak meg azáltal, hogy nem szükséges a vezetékes hálózat kialakítása és a csúszóérintkezők karbantartási költségei is megtakaríthatóak. A vezeték nélküli kommunikáció elterjedése annak is köszönhető, hogy bizonyos teljesítmény szint alatt nincs engedélyhez kötve a frekvenciasáv használata.

2. A VEZETÉK NÉLKÜLI IPARI KOMMUNIKÁCIÓS RENDSZEREK ÁTVITELI MÓDJAI

A szórt spektrumú rádiófrekvenciás átvitelt először a II. világháborúban alkalmazták, majd 1980-tól egyre nagyobb számban alkalmazzák az interferencia csökkentése céljából, mivel két egymáshoz közel elhelyezkedő, azonos frekvenciát használó vezeték nélküli eszköz zavarná egymást, ill. az erősebb elnyomná a gyengébbet. A vezeték nélküli ipari kommunikációs rendszerek általában szórt spektrumú átviteli módot használnak. Szórt spektrumot az átviteli biztonság növelése céljából alkalmaznak, az ipari kommunikációs rendszerek esetében egyes gyártók a szórt spektrumú átviteli módot kombinálják más átviteli módszerrel is. [3.]

2.1. Szórt spektrumú átviteli módok (DSSS, FHSS)

A DSSS technikánál a frekvencia szórását egy kódszó és a bemeneti bitsorozat XOR kapcsolatából generálják. Ezt a jelet használják az RF vivő modulálására. A DSSS technika az általa használt csatornán szétteríti a jelét, a bemeneti bitsorozat és egy daraboló kód XOR kapcsolatából generálják a frekvencia szórását. A bitmintákat csak az adó és a vevő ismeri, egy kívülállónak értelmetlen zajnak tűnik a kommunikáció. A frekvenciaszórás funkcióhoz egy úgynevezett daraboló kódot alkalmaznak. Ez a kód a véletlen zajhoz hasonló tulajdonságokkal bír. Ilyen a Barker kód és a CCK (Complementary Code Keying). [1.]

Az FHSS átvitelnél az adat közvetlenül modulálja az egyetlen vivőfrekvenciát, de a vivőfrekvencia nem állandó, hanem a csatorna RF sávján belül ugrásonként változik az adóban és vevőben generált véletlen számminta alapján. A 2,4 GHz-es sávban a csatorna maximális sáv szélessége 1 MHz FHSS esetében így 79 csatorna, vagyis ugrási frekvencia érhető el. FHSS

* Egyetemi tanárségéd, Miskolci Egyetem, Automatizálási és Kommunikáció-technológiai Tanszék

**MSc mérnök informatikus hallgató

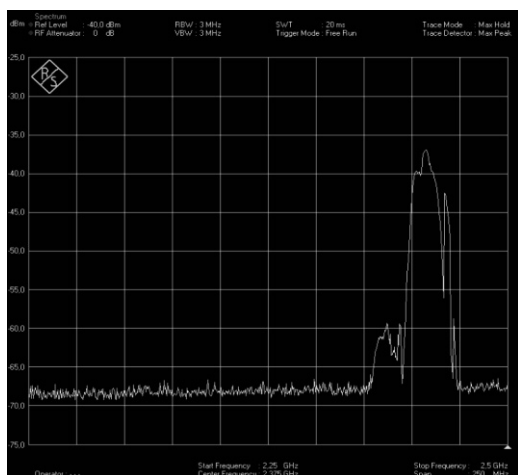
esetében az átvitelnél pillanatnyilag használt csatornák, ill. ugrási frekvenciák a teljes 2,4 GHz-es spektrumon szóródnak. Az adó szabályos időközönként kapcsolja a következő vivőfrekvenciát. A következő vivőfrekvenciát csak a vevő ismeri. Több FHSS csatorna működtetése esetében, a teljes átviteli csatornát a frekvenciák ugrási sorrendje határozza meg. A vevőben egy PN (Pseudo Noise) generátor előállítja ugyanazt az ugrási mintát, mint az adóban. Ugyanabban az időben két különböző hálózat ugrási mintájának megegyezésének a valószínűsége kicsi, ezért több FHSS hálózatot lehet működtetni átlapoló területen interferencia nélkül. [1.]

3. A VEZETÉK NÉLKÜLI IPARI KOMMUNIKÁCIÓS RENDSZEREK SPEKTRUMKÉPEINEK VIZSGÁLATA

A mérések során, a spektrumképek rögzítésére a tanszéki R&S® (Rohde & Schwarz) FSH8 mobil spektrumanalizátort használtuk. A mérések során vizsgáltuk az IWLAN, a WirelessHART és a ZigBee spektrumképét és az IWLAN egy olyan működési esetét, amelyben csatornája átlapolásban van a ZigBee csatornájával.

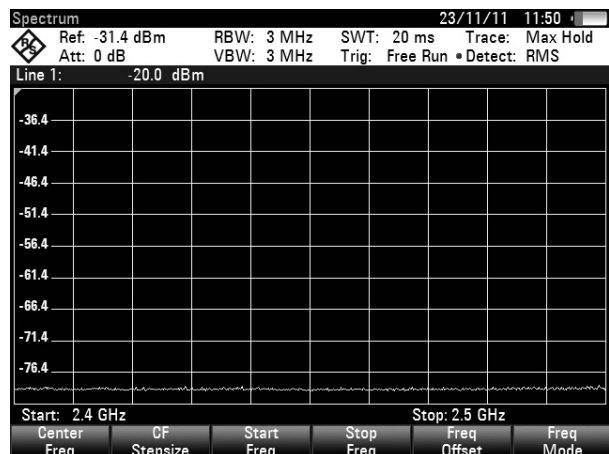
3.1. Az IWLAN spektrumképe

Az 1. ábrán az IWLAN kommunikációja (SIEMENS SCALANCE hozzáférési pont és kliens) során rögzített spektrumkép látható. A frekvenciatartomány végén lévő kb. a 9-es és 11-es csatornákon direkt sorrendű szórt spektrumú jelek találhatók, az is látható, hogy kissé átfedik egymást. A mérés eredményéből látható, hogy a 11-es csatornákon üzemelő eszközök nagyobb jelerősséggel rendelkeznek, mind a 9-es csatornán üzemelő rádiós hozzáférési pont. A mérés során látható, hogy a kisebb frekvenciájú csatornák szabadak, ill. mely csatorna a kevésbé zajos.



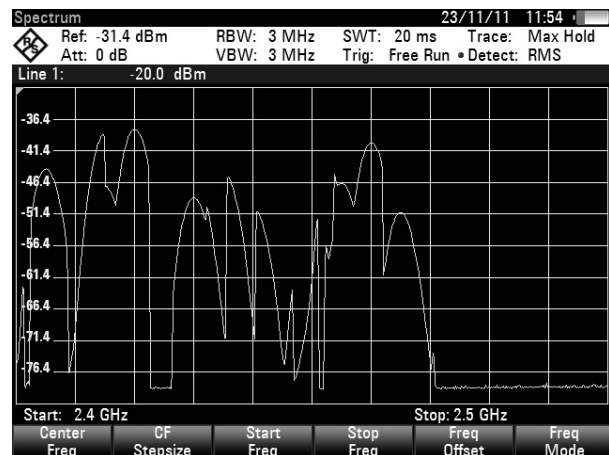
1. ábra: Az IWLAN spektrumképe

3.2. A Wireless HART spektrumképe



2. ábra: Zavarmentes mérési hely

A 2. ábrán látható, hogy a mérési hely zavarforrástól mentes a 2,4 GHz-es ISM sávban.



3. ábra: A WiHART spektrumképe

A WiHART gateway és a WiHART terepi távadó üzembe helyezése után a 3. ábrán lévő spektrumképet kaptam, amely a távadó és a gateway kommunikációjából eredő mikrohullámú jel spektrumát ábrázolja. A 2. ábrán látható, hogy a WiHART üzembe helyezése előtt mikrohullámú jel nem volt érzékelhető a mérési helyen, a spektrumképből az a következtetés vonható le, hogy a WiHART eszközök DSSS átvitelt használnak, amelyet bizonyos idő után csatornaváltással kombinálnak.

Egy másik oka is lehet a frekvenciatartomány telítődésének, mivel a WiHART távadók csak a gateway hatótávolságában bocsátanak ki mikrohullámú jeleket, a gateway kikapcsolásakor a távadók nem küldenek rádiófrekvenciás jeleket. Ezek szerint a gateway szólítja meg a terepi távadókat, hogy mérési adatokat küldjenek. Ezek alapján, a spektrumképen látható DSSS csatornák adódhatnak abból is, hogy a WiHART gateway

bizonyos időközönként megpróbálja felébreszteni, ill. mérési értékeket kérni a különböző csatornáin lévő esetleges távadóktól.

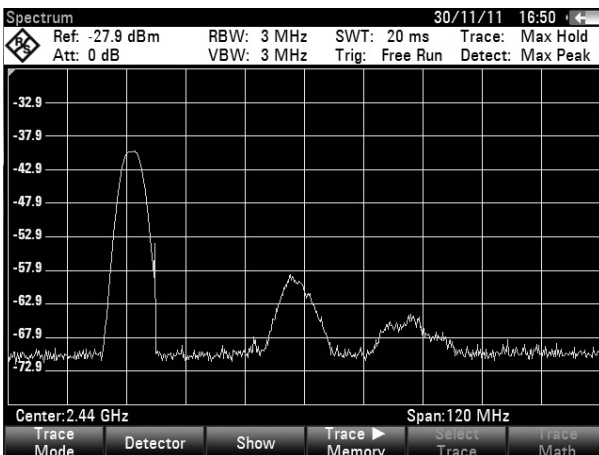
3.3. A ZigBee és az IWLAN együttes vizsgálata

A 4. ábrán az IWLAN kommunikációjából adódó spektrumkép, ill. a környező hozzáférési pontok rádiós jelei és az egyéb zavarok láthatóak. Az IWLAN a frekvenciatartomány elején sugároz, a frekvenciatartomány felén túl az egyéb, a környezetben üzemelő hozzáférési pontok spektrumképe látható.



4. ábra: A mérési helyen lévő IWLAN ill. egyéb WLAN hozzáférési pontok jele

Az IWLAN hozzáférési pont és kliens kikapcsolása után a ZigBee kommunikáció spektrumképe került feltérképezésre, ami a frekvenciatartomány elején jelenik meg (5. ábra), és a környező hozzáférési pontok is érzékelhetőek a frekvenciatartomány közepén, ill. azon túl.

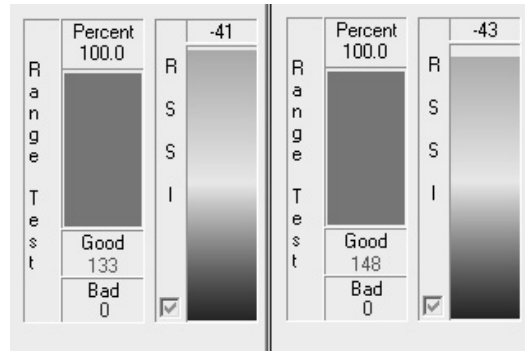


5. ábra: DigiMesh node-ok kommunikációjának spektrumképe

A következőkben mindkét Xbee modul konfigurációjában beállításra került a legelső csatorna

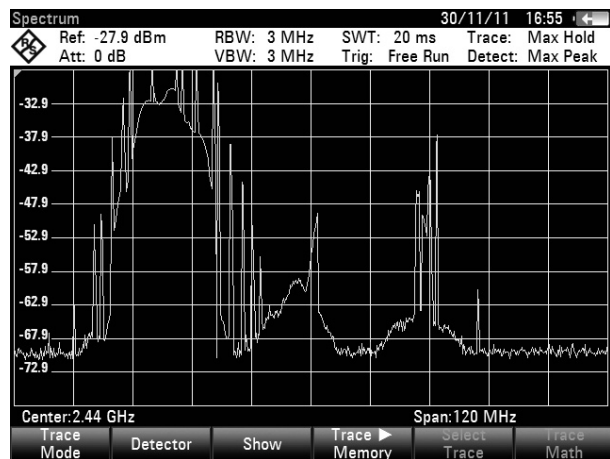
(0x0B), ami átlapolásban van az IWLAN csatornájával, majd bekapcsolásra került az IWLAN hozzáférési pont és az IWLAN kliens.

A 6. Ábra mutatja az átlapolás esetén elindított „range test” eredményét, amin az látható, hogy hibátlan csomagok érkeznek, de a jelerősség kissé lecsökkent, ill. ingadozik. A hibátlan csomagok annak is köszönhetőek, hogy a két Xbee modul 1 m távolságban helyezkedett el egymástól.



6. ábra: X-CTU range test eredménye átlapolás esetén

A 7. ábrán látható, hogy az IWLAN és Xbee modemek csatornáit egybeesnek. A frekvenciatartomány közepe táján ugyancsak az egyéb hozzáférési pontok spektrumképei figyelhetőek meg. Látható, hogy a ZigBee és IWLAN jele fedésben van, a spektrumkép szerint az IWLAN jele elfedi a ZigBee jelét.



7. ábra: Az IWLAN és ZigBee spektrumképe átlapolás esetén

Láthatjuk a spektrumképen, hogy más csatornákon is megjelennek rádiófrekvenciás jelek, amelyet nagy valószínűséggel az IWLAN-tól származnak, mivel próbál csatornát váltani az interferencia miatt.

A mérés eredményei alapján belátható volt, hogy a vizsgált kommunikációs rendszerek még interferencia

esetében is képesek a megfelelő kommunikációra, de ezzel egyben jelentősen csökken a hatótávolságuk is, mivel a mérés során a ZigBee kommunikációs eszközök 1 m távolságban helyezkedtek el és ekkora távolságban is érzékelhető volt a jelerősség csökkenése. [5., 7., 8.]

4. SPEKTRUMKÉPEK VIZSGÁLATÁNAK JELENTŐSÉGE

Egy ipari üzemben, csarnokban, nyílt terepen, ahol vezeték nélküli kommunikációs rendszert szeretnénk üzembe helyezni, az egyéb már működő vezeték nélküli kommunikációs rendszerek jelerősségét megvizsgálhatjuk, megvizsgálható, hogy a telepítési kommunikációs rendszer által használt frekvenciatartományban milyen zavarforrások vannak. [6.] Kimérhető, hogy alkalmas-e az adott terület a kommunikációs rendszer telepítésére, ha igen, akkor mely csatornák szabadak a telepítési kívánt kommunikációs rendszer számára. Egy hordozható kivitelű rádiófrekvenciás spektrumanalizátor és egy iránykereső antenna segítségével könnyedén feltérképezhetők a vezeték nélküli kommunikációs rendszerek jelerősségei, megkereshetők a zavarforrástól mentes helyek, a szabad csatornák.



8. ábra: Iránykereső antenna

5. TOVÁBBI VIZSGÁLATOK

A további vizsgálataink során, olyan ipari körülmények között tervezünk méréseket végezni, ahol szélsőségesek a működési körülmények, illetve olyanok, amik a legrosszabb esetben állnak elő és megvizsgáljuk, hogy ez miként befolyásolja a kommunikációs rendszerek közötti kapcsolatot, adatforgalmat, válaszidőt. [4.] Szélsőséges körülmény lehet egy olyan ipari terület, ahol hegesztenek, vagy esetlegesen más vezeték nélküli kommunikációs rendszer valamilyen okból átlapolásban van a vizsgált rendszerrel. Egy további ilyen, a valóságban könnyen előforduló eset az, amikor egy területen a telepített rendszer üzemeltetése során a tereptárgyakban változás következik be, amit okozhat egy fa növekedése, szállítójármű, daruzás,

állványzat szerelése. A vizsgáltokhoz egy olyan kommunikációs rendszert fogunk használni, mely két Xbee node-ból álló ZigBee hálózat [2.], melynek vizsgálatához egy segédprogram készül, ami segítségével monitorozhatjuk az Xbee node-ok közötti válaszidőt, jelerősséget és a másodpercenkénti válaszcsoomagok számát. Ezzel a rendszerrel különböző adatterhelési szintek, távolságok, zavaró tereptárgyak, bekapcsolt átlapoló kommunikációs rendszerek esetében fogunk méréseket végezni.

6. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

7. IRODALOMJEGYZÉK

- [1] Dr. Ajtonyi I.: Ipari kommunikációs rendszerek IV., Aut-Info Kft., Miskolc, 2011. Június
- [2] FALUDI R.: Building Wireless Sensor Networks, ISBN: 978-0-596-80773-3
- [3] T.S.Rappaport, Wireless Communications: Principles and Practice. Prentice Hall: New Jersey, 1996.
- [4] Rackley, S.: Wireless Networking Technology, Newnes, 2007
- [5] Digi International: Demystifying 802.15.4 and ZigBee, 2008
- [6] Kolozsi-Tóth, M.: Final Thesis - Developing a self-organizing, wireless remote monitoring system for conveyors. - University of Miskolc, 2011.
- [7] <http://www.zigbee.org/>
- [8] <http://www.digi.com/>

HANGVEZÉRELT LOGISZTIKAI RENDSZEREK

VOICE-CONTROLLED LOGISTICS SYSTEMS

Dr. Czap László^{}, Pintér Judit^{**}*

ABSTRACT

The goal of the logistics is the process optimization, and the logistical information systems are helping us in this. The Department of Materials Handling and Logistics created a voice system for the University of Miskolc, that can demonstrate the collecting process in a warehouse like it would have a WMS system in the background, where the commands and the voice terminal tasks are coming from. The article will explain the unique properties of the system, its role in the education, and the following research possibilities.

1. BEVEZETÉS

A technika folyamatosan törekszik az ember és gép közötti kommunikáció megkönnyítésére és gyorsítására. Mivel az emberek között a beszéd a legtermészetesebb információátviteli forma ezért a gépi beszédfeldolgozás az elmúlt évtizedekben a technikai eredményeknek köszönhetően dinamikus fejlődésnek indult. Egyre több alkalmazás jelenik meg ahol a hangvezérlés már részét képezi a rendszernek. A logisztika területén sincs ez másképp. A logisztikát jellemezhetjük úgy is, mint anyagok, információk, személyek, energia rendszereken belüli és közötti áramlásának tervezése, szervezése, irányítása, ellenőrzése. Más nézetek szerint viszont a logisztika nem csak az anyagáramlások optimalizálása, illetve az ehhez szükséges erőforrások biztosítása, koordinálása, hanem tudomány is. Eszerint a nézet szerint a logisztika olyan interdiszciplináris tudomány, amely a logisztika területét érintő valamennyi társtudomány, szakma eddigi és legaktuálisabb ismereteit szintetizálja, azokat saját céljai érdekében egy sajátos rendszerbe rendezi és ezen tudományok és szakmák már meglévő eljárásait, módszereit alkalmazza folyamatai optimalizálására.[1]

Az Anyagmozgatási és Logisztikai Tanszék által megvalósított Voice rendszer raktári folyamatok - elsősorban a szedési feladatok - hangvezérléssel való támogatását teszi lehetővé a beépített TTS (Text-to-

Speech) engine segítségével. A rendszert kiegészítő WMS rész nem került megvalósításra helyette szimulációt valósítottak meg.

2. VOICE RENDSZER

2.1 A rendszer részei [2]

A rendszer célja elsősorban az Egyetem hallgatóit megismertetni a voice technológiával, a hang technológia működésével és gyakorlatot is szerezhettek a rendszer használatához. A rendszerben a voice terminálok valósítják meg a központi rendszer (WMS) felől érkező parancsok hanggá és a kezelő által bemondott információk parancssá alakítását a már említett TTS (Text-to-Speech) engine segítségével. A Vocollect Voice terminálok WLAN-on keresztül csatlakoznak a hálózathoz és a Voice szerverhez. A VoiceConsole a Vocollect Voice terminálok felügyeletére szolgáló alkalmazás. A telepítést követően a VoiceConsole webes felületen érhető el.



1. ábra Vocollect Talkman® Terminál

A betanításhoz -és ami a felismerő rész alapját is képezi - egy eszköz is hozzá tartozik a VoiceClient. A TTS motorok segítségével a Vocollect VoiceClient a "szabad szem és szabad kéz" elvű hangvezérelt munka

^{*}Automatizálási és Kommunikáció-technológiai Tanszék vezetője

^{**}MSc Mérnök informatikus hallgató

lehetőségét teremti meg, így növelve a munkafolyamatok pontosságát, a dolgozók produktivitását és biztonságát.[3] A teljes működés demonstrálásához elengedhetetlen WMS szimulátor fő feladata a voice termináloktól küldött üzenetek fogadása TCP/IP socket-en keresztül, és válaszüzenetek küldése, illetve a kapott üzenetek szerint az adatok adatbázisba írása. A megvalósítandó voice rendszer a WMS „szimulációja” érdekében az alábbi fő funkciókat tartalmazza:

- Szekvencia szerkesztő – Ezzel a modullal állítható össze egy elképzelt eladási rendelés, amit soronként kap meg a voice alapú kiszedő raktári munkás. Egy raktárban is kötött pl. a tárhelyek elnevezése így a szekvencia szerkesztőben összeállítható a voice rendszerben definiált üzenetek formátuma kötött^{***}, a paraméterek száma és mérete is megadott értékeket vehet fel. A megkötéseket fontos betartani mivel a rendszer nem ellenőrzi a logikai szekvenciákat, ezért lehetőség van érvénytelen üzenet beillesztésére is.
- WMS szimulátor – Alapvető feladata a voice terminálok felé a WMS rendszer szimulálása. Socket-en keresztül fogadja a voice terminálok üzeneteit és a kapott üzenetnek megfelelően a Szekvencia szerkesztőben létrehozott szekvenciák szerinti választ küldi el a terminálnak. A kapott üzenetek alapján a szimulált WMS adatbázisban elvégzi a változások beírását. A kapott és küldött üzeneteket egy LOG adatbázisban tárolja, az utólagos elemzés lehetőségének érdekében.
- Eredmények megjelenítése, szerkesztése – Az oktatásban történő szemléltetéshez a létrehozott szimulált adatbázisban a WMS szimulátor által beírt adatok változásait szemlélteti. A funkció feladata az adatbázisban a voice terminálok által végzett műveletek szerinti mennyiségek megjelenítése, az adatbázis szerkesztése. A funkció megvalósítható a Szekvencia szerkesztőben is, mint annak egy része. A megjelenítés a tárhelyek listáját táblázatszerűen (grid) jeleníti meg.

2.2 A rendszer használata[2]

A megfelelő működéshez, használathoz először létre kell hoznunk felhasználókat a VoiceConsole alatt. A felhasználók tetszőlegesen hozzárendelhetők különböző kezelőcsoportokhoz, ezáltal eltérő jogosultsági köröket oszthatunk ki. A felhasználóknak, hogy használni tudják a rendszert el kell végezniük a tanítási folyamatot. A tanítás során a 2. ábrán is szereplő navigációhoz szükséges szavakat kell felmondaniuk a

felhasználóknak többször. A tanítási folyamatot a VoiceClient segítségével végezzük el. A létrejövő minták és az általuk létrejövő TTS folyamathoz szükséges modellek az adott felhasználóhoz rendelődnek.

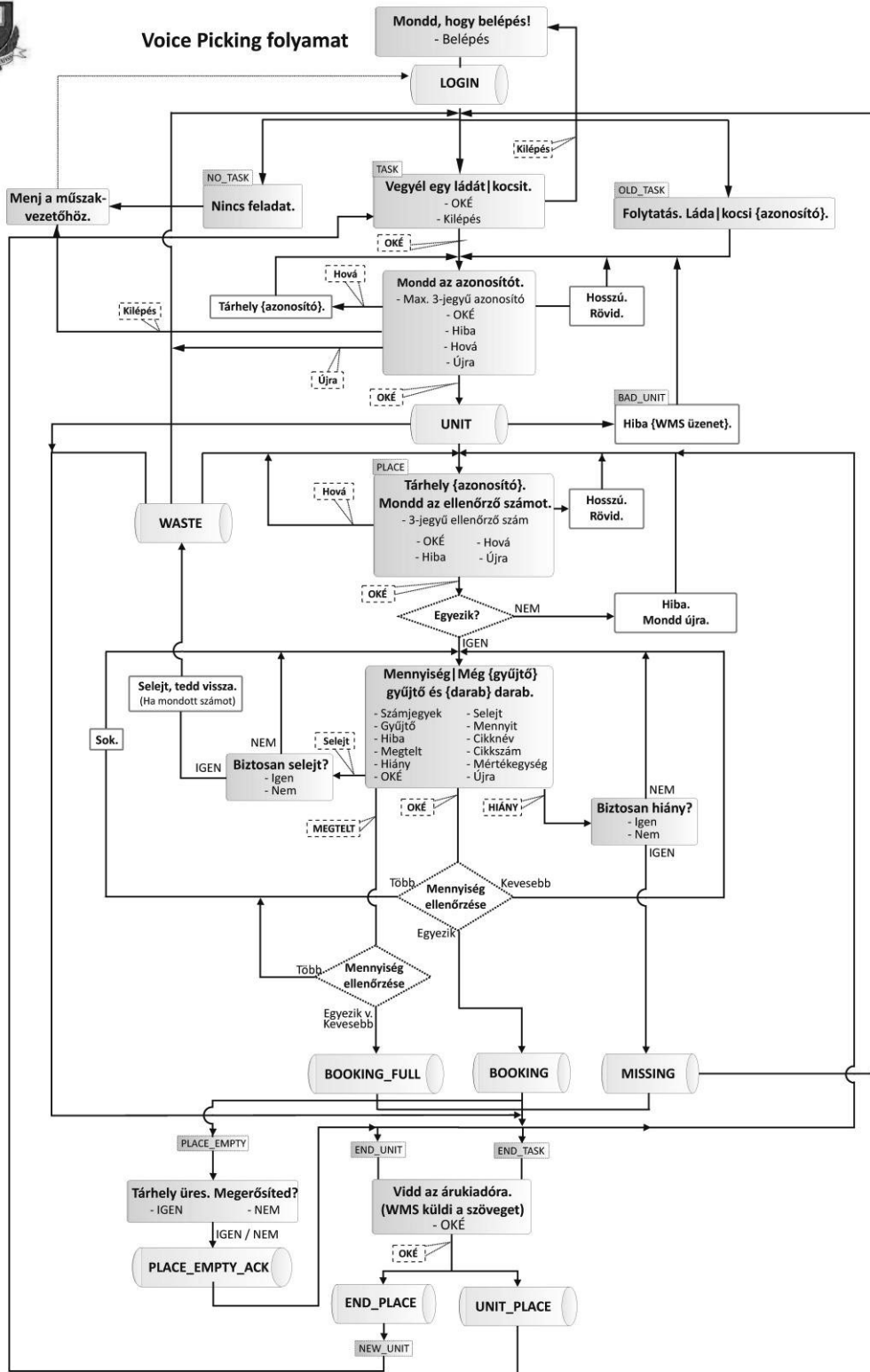
A rendszer működéséhez szintén hozzátartoznak a feladatcsomagok. A Vocollect Voice rendszerben a feladatcsomag jelenti egy adott üzleti logika megvalósítását. Hagyományos számítógépen ez a kiválasztott programmal analóg. A feladatcsomag tartalmazza a folyamatot, a fonetikus tartalmat (szövegeket), és a beállításokat. A BCS Hungary által kifejlesztett feladatot (taszkot) alkalmazzuk a demonstrációhoz.

A VoiceConsole segítségével tudjuk konfigurálni a Vocollect T5 terminálokat is. Az említett beállításokon felül további lehetőségeket is nyújt a VoiceConsole:

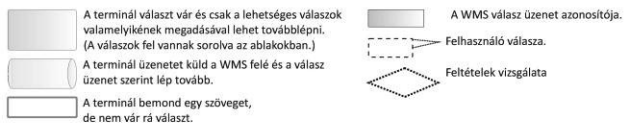
- Új VoiceClient importálása
- Eszköz működési LOG-ok tárolása
- Üzenet küldése kezelőnek
- Hangsablon lementése
- Hibásan betanított szavak törlése

A rendszer konfigurálása és a felhasználók rögzítése után megkezdődhet a munkavégzés. A 2. ábra szemlélteti a folyamatot. Először a rendszerbe kell bejelentkeznie a felhasználónak. A felhasználót a headset azonosítja, a felhasználó azonosítót a Voice Console felületen kell megadni úgy, hogy a WMS rendszerben tárolt azonosítóval egyezzen. Így a felhasználó adatai töltődnek be a megfelelő hangmodellekkel együtt. A terminál a feladat indításakor a WMS felé egy üzenettel indít, ami a WMS felé bejelentkezést jelent, és a WMS-től visszakapja az elvégzendő feladatot. A feladat lehet új feladat vagy korábbi feladat (már van megkezdett szedés, pl. részben feltöltött egységképző), de itt dől el az is, hogyha nincs elvégezhető szedési feladat. A terminál az indítás után a WMS-től lekér egy feladatot. Ha van szedési feladat, akkor a rendszer tovább lép, ha nincs, közli a felhasználóval, hogy „Nincs feladat. Menj a műszakvezetőhöz.” A szedési feladat az egységképző azonosításával kezdődik. A 2. ábra szemlélteti az azonosítás részleteit és a hibakezelést is, mivel a terminál ellenőrzést is végez. Az egységképző kiválasztása után a WMS rendszertől a terminál megkapja a tárhely adatokat. Először a terminál a tárhely azonosítót mondja be, ahova a kezelőnek kell mennie. Ha a tárhely egy szedési folyamatban, ugyanabban a sorban van, mint az előző szedés, akkor a terminál (a WMS-től kapott módon) rövidített tárhelyet mond be, nem mondja a sort, csak az oszlop, emelet és ajtó számát mondja be. A 2. ábra alapján látható, hogy a terminál csak akkor hagyja továbblépni a kezelőt a mennyiség meghatározására, ha bemondta a helyes ellenőrző számot. A mennyiség két egységből állhat össze (gyűjtő és darab, a gyűjtő mértékegysége többféle lehet). A felhasználónak a mennyiséget számjegyenként külön-külön kell bemondani.

^{***} Az üzenetekre vonatkozó formai követelményekről és megkötésekről részletesen a rendszer specifikációjában olvashatunk



Jelmagyarázat:



© BCS Hungary Kft. 2011



2. ábra Voice Picking folyamat[2]

A mennyiség megadása után a terminál ellenőrzi, hogy a kiszedett mennyiség egyezik-e a feladatban megadott mennyiséggel. A feladatban meghatározott mennyiségnél többet nem lehet szedni. A terminál nem engedi tovább a kezelőt, amíg a helyes vagy kevesebb értéket nem ad meg. A 2. ábrán látható, hogy a folyamat ezen lépésénél mely parancsok értelmezettek (számok, hiba, gyűjtő, megtelt, hiány stb.). A mennyiség ellenőrzésénél az alábbi esetek fordulhatnak elő:

- Helyes mennyiséget szedett: Helyes mennyiség szedése esetén a terminál nem mondja vissza a mennyiséget, hanem a következő feladatra lép.
- Selejtet talál a polcon: Ebben az esetben a tárhelyről nem szabad kiszedni árut. A terminál üzen a WMS-nek, hogy selejt van a megadott tárhelyen, és az ábra szerinti szálacon futhat tovább a folyamat.
- Egységképző megtelt: a kezelő jelzi ezt a terminálnak, majd a WMS rendszer felől érkezik egy parancs, hogy az egységképzőt vigye az árukiadó területre egy megadott tárhelyre, majd az alábbi két lehetőségek fordulhatnak elő.
 - Új egységképzőt kell felvenni
 - A feladat végére ért
- Kevesebbet szedett: Ha kevesebbet szedett, a terminál bemondja a hiányzó mennyiséget (részszedés). A kezelőnek ekkor a mennyiség bemondásánál az így megadott hiányzó mennyiséget kell bemondania. A kevesebb mennyiség esetén is előfordulhat, hogy a tárhely kiürül. Ebben az esetben ugyanaz a folyamat zajlik, mint a helyes mennyiség esetén.

Az egységképző megtelte esetén vagy a feladat végén az egységképzőt az árukiadó területre kell vinni. Ezt a parancsot a mennyiség bemondása után kaphatja a felhasználó a WMS rendszer felől. Az egységképző árukiadón történő letétele után új feladatot kezdhet a felhasználó vagy kiléphet a rendszerből, és így a folyamat újra kezdődik.

3. A HANGVEZÉRLŐ RENDSZER JELLEMZÉSE

A rendszerről beszéd felismerés szempontjából vizsgálva, az alábbi jellemzőket állapíthatjuk meg. A rendszer a TTS engine-re épül ezért a "beszéd szöveggé alakítása" beszéd felismerési feladatot látja el.

Mivel minden új felhasználó felvétele esetén az adott személynek végre kell hajtania a tanítási folyamatot a rendszer mindig az aktuális felhasználóhoz tartozó modelleket tölti be, így a beszéd felismerő modulja a rendszernek beszélő adaptáltság szempontjából beszélőfüggő [4]. Biztonsági szempontból helyes döntésnek nevezhetjük, hogy a rendszert

beszélőfüggőnek tervezték, mivel így nem fordulhat elő, hogy a rendszer akár a kezelő mellett álló éppen csak beszélgető személy hangjára is reagáljon. Viszont hátránynak tekinthetjük, hogy egy adott személy esetén lehet, hogy a tanítást újra el kell végezni, ha a felhasználó hangja valamilyen oknál fogva jelentősen megváltozik (pl. megfázás).

A 2. ábra magába foglalja az összes előforduló parancsszavakat, amelyek száma nem túl sok így ha szótár nagysága szerint osztályozzuk, akkor a rendszert a kis (kötött, zárt) szótárú rendszernek tekintjük.

Beszédmód szempontjából a rendszer demonstrációja arra enged következtetni, hogy a hangvezérlés a kapcsolt szavas felismerésére alkalmas [4].

Üzem mód szempontjából a rendszer parancsmódúnak nevezhető, hiszen valós időben használható (online üzem módú) és vezérlés valósítható meg vele [4].

4. KITEKINTÉS

Akusztikai környezet szempontjából vizsgálva a rendszert a zajos környezetben használt architektúrák közé sorolhatjuk. A Voice rendszer céljából adódik, hogy ipari környezetben használják a rendszert, ami kicsit sem mondható csendesnek. Ilyen esetben valamilyen zajszűrő algoritmus, eljárás használata elengedhetetlen. A bemutatott rendszer a zajszűrést úgy oldja meg, hogy a Vocollect Voice terminálok két mikrofonnal rendelkeznek, így a nem a kezelő felé irányuló mikrofonnal folyamatosan mintavételezhetik a külső zajokat hanghatásokat és egy algoritmus segítségével a kezelő mikrofonjából érkező hangot megszüri.

A jövőben terveink közt szerepel a rendszer tesztelése különböző zajok alkalmazásával illetve a rendszer jeleinek megfigyelése és elemzése.

5. KÖSZÖNETNYILVÁNÍTÁS

"A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg."

6. IRODALOM

- [1] http://hu.wikipedia.org/wiki/Logisztika#cite_note-0
- [2] Miskolci Egyetem - Voice rendszer dokumentációk
- [3] http://www.bcs.hu/hu/vocollect_termek/voicelient
- [4] NÉMETH G., OLASZY G.: A Magyar Beszéd, Akadémiai Kiadó, Budapest, 2010

BEÁGYAZOTT TELEGRAMOKAT TARTALMAZÓ IPARI ETHERNET KERETEK CIKLUSIDEJE

CYCLE TIME OF EMBEDDED DATAGRAM USED INDUSTRIAL ETHERNET FRAMES

Ferenczi István*

ABSTRACT

There are two ways to forward control data in Hard Real-Time industrial Ethernet communication systems. The first approach is that the control device (master) sends separate frames containing the control data to IO devices (slave) that are connected to it via Ethernet. The second approach, one tries to utilize the maximal payload that amounts to 1500 bytes according to the IEEE 802.3 standard. In this case, one attempts to place the control data (telegrams) so as to fully utilize frames. The EtherCAT system developed by Backhoff is based on this principle, which is more advantageous when using small sized (16-64 bytes) telegrams.

1. BEVEZETÉS

Az elsősorban motorvezérléseknél használt, szigorúan valós idejű ipari Ethernet kommunikációs rendszereknél a vezérlési adatok továbbítása két módon valósítható meg. Az egyik megoldás, amikor a kontroller mindegyik, vele Ethernet kommunikációs kapcsolatban lévő IO eszköznek külön-külön címezi a kereteket, amelyek a megfelelő vezérlési adatokat tartalmazzák. Ezt a módszert alkalmazza a Profinet IRT rendszer. Ebben az esetben 36 bájt nál kisebb vezérlési adatok gyakorlatilag nem befolyásolják a továbbításhoz szükséges időt, mert így a teljes keretméret nem haladja meg a minimális, 64 bájt os keretet. Hátránya ennek a megoldásnak, hogy az előbb említett esetben a keretterhelési tényező a legjobb esetben is csak 56% lehet. Igazi előnye az 1000Mbps-os gigabit Ethernet hálózaton mutatkozik meg, amikor az adattovábbításhoz szükséges idő gyakorlatilag már nem függ a keret méretétől. [1]

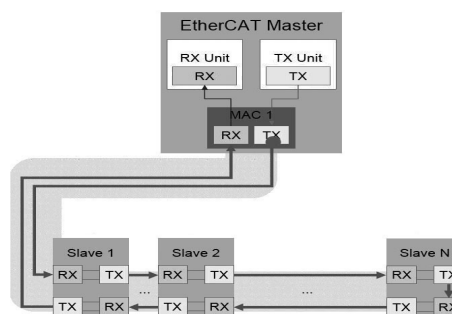
A másik megoldás, amikor arra törekszünk, hogy kihasználjuk a maximális keretterhelést, amely az IEEE 802.3 szerint 1500 bájt lehet. Ebben az esetben az IO eszközök számára továbbítandó vezérlési adatokat (telegramokat) igyekszünk egy vagy több keretben belül úgy elhelyezni, hogy minél nagyobb legyen a keretek kihasználtsága. Ez a módszer kétségtelenül előnyt jelent kisméretű (36 bájt nál kisebb) telegramok továbbítása esetén.

* főiskolai adjunktus, Nyíregyházi Főiskola, MMK

Ebben a cikkben egy ilyen megoldást és egy ehhez tartozó ciklusidő számítási algoritmust fogok bemutatni.

2. AZ EtherCAT RENDSZER FONTOSABB JELLEMZŐI

Az EtherCAT-et igen rövid ciklusidő, zavarmentes kommunikáció és nagy pontosságú szinkronizálás jellemzi. 100 Mbps-os full duplex Ethernet hálózaton, szegmensenként gyakorlatilag szinte korlátlan számú (max. 65 535) IO eszköz kapcsolható össze egymással. A gyártó adatai szerint 100 IO eszköz, összesen 1000 digitális IO csatorna esetében a frissítési idő mindösszesen 30 μ s, de 100 darab szervo tengely, egyenként 8 bájt os IO adatokkal történő hajtásakor sem sokkal haladja meg a 100 μ s-ot [3]. E rendkívüli gyorsaságot annak köszönheti, hogy a csomópontokban lévő IO eszközök „menet közben”, közvetlen memória hozzáféréssel (DMA) veszik le a nekik címzett, illetve teszik fel a továbbítani kívánt adatokat (1. ábra). Ennek következtében a csomópontokon belüli késleltetési idő is rendkívül rövid, mindösszesen 1,35 μ s full duplex hálózaton.



1. ábra. EtherCAT adattovábbítás

A vezérlési adatok nagysága sem mondható korlátozottnak. Noha a motorvezérléseknél nincs szükség nagyméretű telegramok továbbítására, a gyártó 60 kilobájtban határozta meg a maximális adatmennyiséget, amely továbbítható a csomópontoknak. (Ebben az esetben természetesen több keretre van szükség.)

A rendszer elsősorban a busz topológiát támogatja, de jól használható, csillag, fa és vegyes topológiájú hálózatokban is. Vegyes topológiában a ciklusidő azért lényegesen nagyobb, mint például a busz topológiában.

3. AZ EtherCAT ESZKÖZ-PROTOKOLL

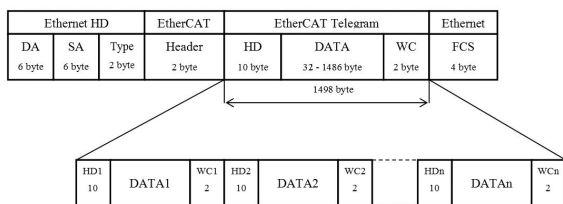
Egy teljes EtherCAT rendszer alapvetően két féle kommunikációs protokollt használ, amelyek a szabványos IEEE 802.3 Ethernetre épülnek:

- Nem szinkronizált folyamatokhoz, mint például konfigurációs beállításokhoz, folyamatvizualizáláshoz valamint lekérdezésekhez a TCP/IP vagy az UDP/IP-t,
- Szinkronizált vezérlési adatok továbbítására az EtherCAT eszköz-protokollt.

Ebben a cikkben csak az utóbbi kerül bemutatásra.

Ahhoz, hogy a busz ciklusidő meghatározásához szükséges algoritmust felépíthessük, vizsgáljuk meg először, hogy hogyan épülnek be a telegramok az Ethernet keretbe. Az IEEE 802.3 szabvány szerinti Ethernet keret annyiban módosul, hogy kiegészül egy 2 bájtos EtherCAT fejléccel, amely főleg a beágyazott adatmennyiség hosszára és típusára vonatkozóan tartalmaz információt. Így az adatok számára egy kereten belül már csak 1498 bájtnyi hely marad, ahová az EtherCAT telegramokat helyezhetjük. A keret azonosítására az ipari Ethernet rendszereknél használatos Ether Type-al történik, amely ebben az esetben 0x88A4. [2]

Azonosítás céljából mindegyik beágyazott telegramot is fejléccel (Telegram Header), illetve ellenőrző résszel, un. működés számlálóval (Working Counter) kell ellátni (2. ábra).



2. ábra. Ethernet keretbe ágyazott telegramok

A keret maximális mérete 1518 bájt, amely még kiegészül 8 bájtnyi előtaggal (7 bájt Preambulum + 1 bájt SFD), valamint 12 bájt keretek közötti résszel (IFG), így a teljes keretméret, összesen 1538 bájt lesz.

4. A CIKLUSIDŐ MEGHATÁROZÁSA

A busz ciklusidő meghatározása meglehetősen összetett feladat, de alapvetően két tényezőtől függ: a csomópontok számától (N) és a keretterheléstől ($Data$). Először – a keretterhelés függvényében – meg

kell határoznunk, hogy hány telegram helyezhető el egy keretben. Figyelembe véve, hogy mindegyik telegramhoz tartozik egy 10 bájtos fejléc (HD) és egy 2 bájtos ellenőrző rész (WC), következik, hogy egy kereten belül elhelyezhető telegramok számát (k) a következő összefüggéssel határozhatjuk meg:

$$k = INT\left[\frac{Max_eCAT_Data_Size}{HD + Data + WC}\right] = INT\left[\frac{1498}{12 + Data}\right] \quad (1)$$

ahol $INT[]$ a tört egész részét jelenti.

A továbbiakban bemutatott modellben feltételezzük, hogy egyszerre, egy időben ugyanakkora adatmennyiséget továbbítunk minden csomópontbeli eszköznek. Ez nem egy rendkívüli dolog, mivel a valóságban is, például tengelyvezérléseknél, ha az eszközök azonosak, akkor a vezérlésükhöz szükséges adatmennyiség is azonos nagyságú, csak a tartalmuk más.

Ha már ismerjük az egy keretben elhelyezhető telegramok számát (k) és ez a szám kisebb, mint az eszközök száma, akkor meghatározhatjuk, hogy a teljes adatmennyiség továbbításához hány keretre (n_F) van szükség.

$$n_F = INT\left[\frac{N}{k}\right] + 1; \quad \text{ha } N > k \quad (2)$$

Ellenkező esetben a teljes adatmennyiség elfér egyetlen egy keretben és k egyenlő lesz N -el. (k a telegramok, N pedig a csomópontok száma.)

Figyelembe véve az előbbi kritériumokat és azt a tényt, hogy a minimális Ethernet keretméret nem lehet kisebb, mint 64 bájt, 3 esetet különböztethetünk meg:

- 1) $N = k$ és a teljes EtherCAT telegramok mérete (fejléccel és működés számlálóval együtt) nem haladja meg a 44 bájtot. Ez a legegyszerűbb eset, ilyenkor a teljes keretméret 64 bájt, az előtaggal pedig 84 bájt lesz. A ciklusidő (T_c) meghatározásához figyelembe kell vegyünk még a csomóponti eszközök, valamint a közeg késleltetési idejét ($t_D = 1,35\mu s$ és $t_M \approx 0,5ns/m$; maximum 100m-es szegmenseknél $t_M = 0,5\mu s$). Következik:

$$T_c = \frac{8 \cdot 84}{b} + N \cdot (t_D + t_M) \quad (3)$$

Az előbbi összefüggésben b a bitebességet jelöli megabit/szekundumban (Mbps), a ciklusidőt pedig mikroszekundumban kapjuk.

Megjegyzés: A fenti eset a valóságban gyakorlatilag nem, vagy csak nagyon ritkán fordul elő, ugyanis például 3 csomópont és 1 bájt vezérlési adat esetében a keretterhelés a fejlécekkel együtt már meghaladja a 34 bájtot!

- 2) $N = k$ és a teljes vezérlési adatmennyiség elfér egyetlen egy keretben ($n_F = 1$). Ebben az esetben is figyelembe kell, vegyünk az EtherCAT telegra-

mok mellett az Ethernet fejléccet és egyéb járulékos bajtokat. A keret nagysága változó lehet, de nem haladja meg a maximális méretet. A ciklusidő a következő összefüggéssel számítható ki:

$$T_c = \frac{8N \cdot (12 + Data) + 320}{b} + N(t_D + t_M) \quad (4)$$

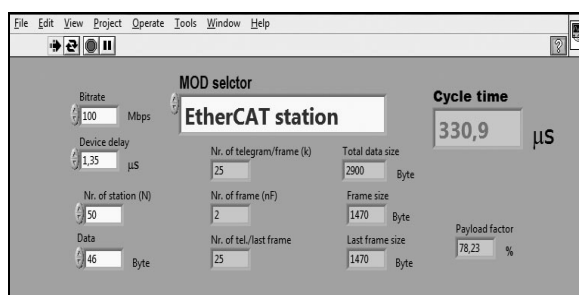
- 3) A legáltalánosabb eset az, amikor a csomópontok száma nagyobb, mint ahány telegram elhelyezhető egy keretben ($N > k$). A (2) alapján kapjuk meg, hogy hány keretre van szükség a teljes vezérlési adatmennyiség továbbítására. A ciklusidőt pedig az alábbi általánosított összefüggéssel határozhatjuk meg:

$$T_c = \frac{8}{b} [40 \cdot n_F + N \cdot (12 + Data)] + N \cdot (t_D + t_M) \quad (5)$$

Megjegyzés: A (3), (4), (5) összefüggésekben szereplő t_D és t_M , különböző értékeket kaphat. A késleltetési idő (t_D) függ a bitsebességtől, 1Gbps-es hálózatban 0,85 μ s, t_M pedig az eszközök közötti távolságtól.

5. A SZÁMÍTÁSI ALGORITMUS KIDOLGOZÁSA LABVIEW 8.2-BEN

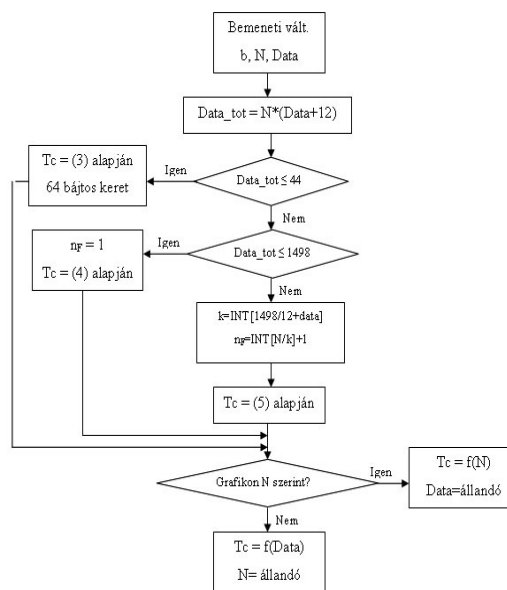
A Labviewban kidolgozott algoritmus mindhárom, az előző fejezetben bemutatott esetet kezelni tudja és minden esetben lehetőséget biztosít a ciklusidő meghatározására gyakorlatilag korlátlan számú csomóponti eszköz (N) és 1-től egészen 1486 bájtig terjedő keretterhelés függvényében. Egyaránt alkalmazható Fast Ethernet és Gigabit Ethernet hálózatra is, ennek megfelelően beállítható az eszközkésleltetési idő is. Az eszközök közötti szegmenshossz minden esetben a maximális 100 méter hosszúságú.



3. ábra. Az alkalmazás konfigurációs felülete

A ciklusidő kiszámítása mellett, ábrázolni tudja ennek változását a csomópontok, illetve a keretterhelés függvényében, a keletkező adatokat pedig Excel fájlban is tudja tárolni. Ezen kívül, mint ahogyan a 3. ábrán is láthatjuk, számos más információt is kijelvez, mint például a keretek, illetve a beágyazott telegramok számát, vagy a keretterhelési tényezőt [1].

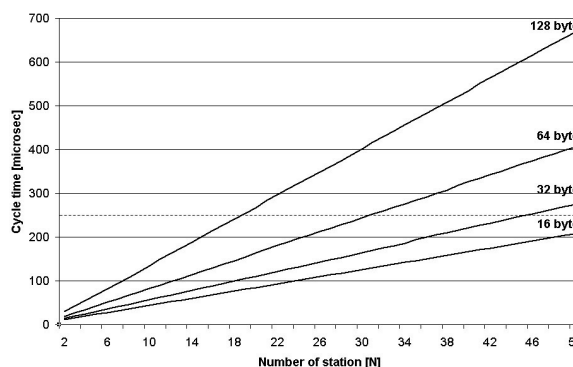
A következő ábrán a Labviewban elkészített EtherCAT minimális ciklusidő meghatározási modelljének algoritmusát láthatjuk.



4. ábra. Az EtherCAT ciklusidő számítási algoritmus

6. A CIKLUSIDŐ VÁLTOZÁSÁNAK ELEMZÉSE

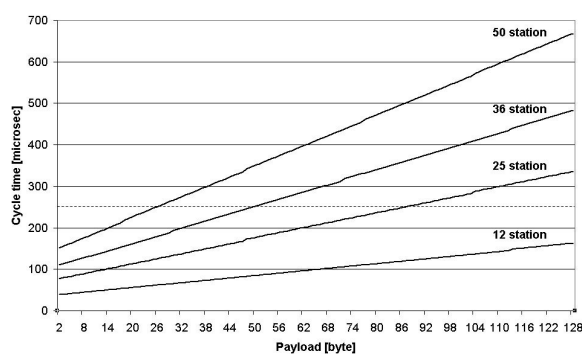
Először vizsgáljuk meg azt az esetet, amikor a ciklusidő a csomóponti eszközök száma szerint változik, maximum 50 csomópontig különböző, 16, 32, 64 illetve 128 bájtkeretterhelés mellett, Fast Ethernet (100 Mbps) hálózatban (5. ábra).



5. ábra. A ciklusidő változása az IO eszközök száma szerint

A szigorúan valós idejű rendszereknél a ciklusidő legfeljebb 1ms, vagy ennek törtrésze 1/2, 1/4, 1/8, stb. lehet. 50 csomópont esetében 16 bájtkeretterheléssel még a 0,25ms-os ciklusidő is megvalósítható. Nagyobb keretterheléseknél, például 64 bájtnál már csak 0,5ms-os ciklusidő biztosítható.

A 6. ábrán ugyancsak a ciklusidő alakulását láthatjuk, csak most a keretterhelés függvényében.



6. ábra. A ciklusidő változása a keretterhelés szerint

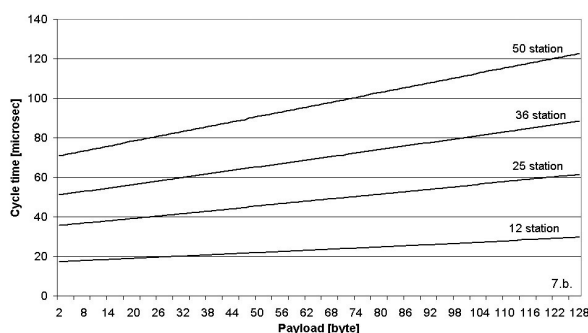
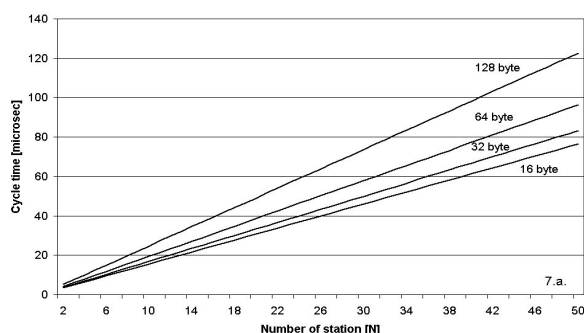
Az előző ábrákból egyértelműen látszik, hogy mindkét meghatározó tényező (N, Data) egyértelműen befolyásolja a ciklusidő növekedését. A növekedés mértéke azonban sokkal alacsonyabb, amikor az EtherCAT terhelést növeljük.

Mindkét ábrán, nyomon követhetjük, hogy azonos terhelés mellett, amikor az eszközök számát megduplázzuk, akkor a ciklusidő is kb. duplájára nő. Például 16 bájttal terhelésnél, ha az eszközök számát növeljük 24-ről 48-ra, akkor a ciklusidő 101 μ s-ról 200 μ s-ra növekszik. De ugyanez mondható el 128 bájtos terhelésnél is. Itt is kb. duplájára növekszik a ciklusidő. Viszont, ha a terhelést duplázzuk (pl. 32-ről 64-re), azonos eszközszám mellett, a ciklusidő jóval kisebb arányban növekszik, átlagban mintegy 50%-al. Ebből azt a következtetést vonhatjuk le, hogy az EtherCAT rendszer busz frissítési ciklusideje kevésbé érzékeny a vezérlési adatok nagyságának a változására. Ez abból is adódik, hogy a ciklusidő egy jelentős összetevője nagymértékben függ az eszközök által meghatározott késleltetési időtől.

Azt is láthatjuk, hogy 128 bájtos terhelésnél, már csak körülbelül 36 eszközt tudunk kiszolgálni, hogy ne haladjuk meg az 500 μ s-os korlátot, amely 50-50%-os időréssel (50% EtherCAT, 50% UDP/IP adatok) összesen, a szigorúan valós idejű rendszereknél a maximális 1ms-os buszfrissítési időt adja.

A továbbiakban azt vizsgáltam, hogy milyen hatással van a ciklusidőre a sebesség növelése. Várhatóan a keretek továbbítási ideje 10-szeressé csökken. Az eszközök késleltetési ideje viszont csak kb. a felére, így $t_D = 0,85\mu$ s értékkel számolhatunk [4].

A következő grafikonok (7. ábra) elemzéséből láthatjuk, hogy jelentősen, mintegy öt és félszeresére csökkent a ciklusidő, viszont a görbék jellege nem változott. A 7.a. ábrán láthatjuk, hogy ugyanúgy duplájára nő a ciklusidő, amikor az eszközök száma megduplázódik. Viszont a 7.b. ábrán azt is megfigyelhetjük, hogy ha az EtherCAT adatokat duplázzuk meg, például 32-ről 64-re, a ciklusidő már csak kb. 15%-al növekszik.



7. ábra. A ciklusidő alakulása Gigabit Etherneten

7. ÖSSZEFOGLALÁS

Összefoglalva az előzőeket kijelenthetjük, hogy az EtherCAT a ciklusidőt illetően 100Mbps-os hálózatban igen jól viselkedik, főleg kisebb adatterhelések (16, 32, 64 bájttal) esetében. A sebesség növelése viszont nem hozza meg a várt eredményt, mert a ciklusidő csak kb. 5,5-szörös csökkenést ér el. Azt viszont láthatjuk, hogy 128 bájtnál, nemhogy 50, de akár 100 eszközt is képes kiszolgálni a fentebb már említett időkorláton belül.

8. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

9. IRODALOMJEGYZÉK

- [1] FERENCZI I., Methods to determine a Profinet IRT bus cycle time, MicroCAD Conference 2011.
- [2] M. ROSTAN, EtherCAT Introduction, Hannover messe 2008.
- [3] EtherCAT – The Ethernet Fieldbus, 2009, www.ethercat.org
- [4] JASPERNEITE J., SCHUMACHER M., Limits of increasing the performance of Industrial Ethernet Protocols, EFTA 2007 Proceedings, 17-24 old.

Nemnewtoni folyadék kényszeráramlása síklapon porózus közegben

Forced convection flow of a non-Newtonian fluid over a flat plate in porous medium

Bognár Gabriella*, Rozgonyi Erika**

ABSTRACT

We study the steady forced convection flow of a non-Newtonian fluid over a horizontal surface embedded in a saturated Darcy-Brinkman-Forchheimer porous medium. The flow is driven by a constant pressure gradient. The mathematical model is formed by a boundary value problem which is investigated analytically. We give the solution of this problem as a power series expansion where the coefficients will be determined.

1. BEVEZETÉS

A porózus közeggel kapcsolatos kutatások a XIX. században a Darcy-féle modellhez kötődnek, melyet többféle formában továbbfejlesztettek, mint például a Darcy-Brinkman, vagy a Darcy-Brinkman-Forchheimer modell. Newtoni közegre a Darcy-féle viszkozus tagot 1947-ben Brinkman egy második taggal egészítette ki, amely modellhez 1982-ben Joseph egy újabb tagot illesztett, egy a sebesség négyzetével arányos Forchheimer-féle tagot (lásd. [8]).

1987-ben Poulikakos és Renken [11] a kényszeráramlást porózus közeggel töltött csatornában vizsgálta változó porozitás és Brinkman surlódás mellett. Hsu és Chang 1990-ben megjelent [7] cikkében a Darcy-Brinkman-Forchheimer modellt tanulmányozta newtoni közeg esetén.

Newtoni folyadékok porózus közegben való áramlásának irodalma és a nemnewtoni speciális hatványtörvény szerinti folyadékok áramlásával foglalkozó cikkek száma rendkívül nagy. A nemnewtoni folyadékokra irányuló figyelmet a mérnöki alkalmazhatóságok széles köre indokolja.

2009-ben Nield [10] átfogó ismertetést adott a kutatások fejlődéséről az utóbbi évtizedekre vonatkozóan a hatványtörvény szerinti folyadékok áramlását leíró matematikai modellre porózus közeg esetén.

Ezen dolgozatban Magyarai és Pantokratoras [9] 2010-es cikke alapján hatványtörvény szerinti folyadék

állandó nyomás gradiens melletti állandósult mozgását vizsgáljuk. Megadjuk a probléma matematikai modelljét. Magyarai és Pantokratoras a származtatott peremérték problémát véges differenciák módszerével oldotta meg. A probléma aszimptotikus megoldását a hatványkitevő két speciális értéke esetén egzakt analitikus formában írták fel és vizsgálták a Darcy és a Forchheimer számoknak az áramlásra tett hatását.

Egyes alkalmazások szempontjából fontos a megoldásfüggvények megadása végtelen sor alakjában, amely végtelen soroktól megkívánjuk, hogy bizonyos intervallumban a sorba fejtett függvényt jól közelítsék.

A peremérték problémához megadjuk, hogy a megoldás milyen hatványsor alakban létezik és módszert adunk a hatványsorban szereplő együtthatók kiszámítására.

2. A PROBLÉMA MATEMATIKAI MODELLJE

Nemnewtoni hatványtörvény szerinti folyadék áramlását vizsgáljuk Darcy-Brinkman-Forchheimer porózus közegbe ágyazott vízszintes síklapon. Jelöljük a folyadék áramlási sebességének x irányú komponensét u -val, y irányú komponensét v -vel, ahol a síklap az x tengelyen helyezkedik el, y pedig az erre merőleges koordinátát jelöli. Az állandósult kétdimenziós határréteg áramlás kontinuitási- és mozgásegyenletét összenyomhatatlan nemnewtoni folyadékokra a szokásos határréteg elmélet szerinti elhanyagolások és a Forchheimer-féle tag figyelembe vételével az alábbi egyenletekkel írhatjuk le (lásd. Nield [10]):

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (1)$$

és

$$\frac{\rho}{\varepsilon^2} \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) = -\frac{\partial p}{\partial x} + \frac{\mu}{\varepsilon^n} \frac{\partial}{\partial y} \left[\left| \frac{\partial u}{\partial y} \right|^{n-1} \frac{\partial u}{\partial y} \right] - \frac{\mu}{K^*} u^n - \frac{\rho C_F}{K^{1/2}} u^2, \quad (2)$$

* egyetemi docens, Miskolci Egyetem Analízis Tanszék

** egyetemi adjunktus, Miskolci Egyetem Analízis Tanszék

ahol p a nyomást, ε a porozitást, μ a konzisztencia együtthatót, n a hatványtörvényben lévő kitevőt, ρ

folyadék sűrűségét, K^* a módosított áteresztőképességet, K a normál áteresztőképességet, míg C_F a Forchheimer konstans jelöli. Ha $0 < n < 1$, akkor a folyadék pszeudoplasztikus, $n > 1$ esetben dilatáló, míg $n = 1$ esetben newtoni folyadékokról beszélhetünk. A peremfeltételek a síklapon ($y = 0$):

$$u = 0, v = 0, \quad (3)$$

illetve a határréteg szélén ($y \rightarrow \infty$)

$$u \rightarrow u_\infty, \quad (4)$$

ahol a folyadék áramlási sebessége a határrétegen kívül u_∞ . A konstans nyomás gradienst a határréteg szélén felírt (4) feltételből határozzuk meg:

ha $y \rightarrow \infty$, akkor

$$\frac{\partial p}{\partial x} = \frac{\mu}{K^*} u_\infty^n + \frac{\rho C_F}{K^{1/2}} u_\infty^2, \quad (5)$$

tehát a (2) egyenlet a következő alakba írható [9]:

$$\frac{\rho}{\varepsilon^2} \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) = \frac{\mu}{\varepsilon^n} \frac{\partial}{\partial y} \left[\left| \frac{\partial u}{\partial y} \right|^{n-1} \frac{\partial u}{\partial y} \right] + \frac{\mu}{K^*} (u_\infty^n - u^n) + \frac{\rho C_F}{K^{1/2}} (u_\infty^2 - u^2). \quad (6)$$

Bevezetjük az alábbi dimenziómentes mennyiségeket:

$$X = \frac{x}{L}, Y = \frac{y}{L}, U = \frac{u}{u_\infty}, V = \frac{v}{u_\infty}, \quad (7)$$

ahol

$$L = \left(\frac{\mu \varepsilon^{2-n} u_\infty^{n-2}}{\rho} \right)^{\frac{1}{n}}. \quad (8)$$

Az (1) és a (6) egyenletek dimenziómentes változókkal az alábbiak:

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0, \quad (9)$$

$$U \frac{\partial U}{\partial X} + V \frac{\partial U}{\partial Y} = \frac{\partial}{\partial Y} \left[\left| \frac{\partial U}{\partial Y} \right|^{n-1} \frac{\partial U}{\partial Y} \right] + \frac{1}{Da} (1 - U^n) + F (1 - U^2), \quad (10)$$

ahol a Da a Darcy és az F a Forchheimer számot a

$$Da = \frac{K^*}{\varepsilon^n L^{n+1}}, F = \frac{C_F \varepsilon^2}{K^{1/2}} L \quad (11)$$

kifejezésekkel értelmezzük. A (3) és (4) feltételek a dimenziómentes változókkal az

$$U(X, 0) = 0, V(X, 0) = 0, \quad (12)$$

$$U(X, \infty) = \lim_{Y \rightarrow \infty} U(X, Y) = 1 \quad (13)$$

formulákkal adhatók meg.

Feltételezzük, hogy a nemnewtoni folyadék Blasius-féle határréteg áramlásához hasonlóan a folyadék áramlás főárammal párhuzamos $U = U(X, Y)$ sebesség komponense monoton növekszik 0-tól az 1 értékig minden $X > 0$ esetén, miközben $Y \rightarrow \infty$. Tehát U deriváltja Y irányban nem negatív, azaz $\partial U / \partial Y \geq 0$. Így a (10) egyenlet az

$$U \frac{\partial U}{\partial X} + V \frac{\partial U}{\partial Y} = n \left(\frac{\partial U}{\partial Y} \right)^{n-1} \frac{\partial^2 U}{\partial Y^2} \quad (14)$$

alakba írható. A teljesen kifejlődött áramlás esetén a belépőélnél jelentkező perturbáció hatása elhanyagolható és ekkor az U sebességkomponens csak az Y változó függvénye, továbbá $V = 0$. Tehát a (9) egyenlet teljesül, a (14) parciális differenciálegyenlet pedig a következő közönséges differenciálegyenlet alakot ölti:

$$n \left(\frac{dU}{dY} \right)^{n-1} \frac{d^2 U}{dY^2} + \frac{1}{Da} (1 - U^n) + F (1 - U^2) = 0. \quad (15)$$

Ezen differenciálegyenlethez tekintsük a

$$\left. \frac{\partial U}{\partial Y} \right|_{Y=0} = S \quad (16)$$

kezdeti feltételt, amely teljesen kifejlődött áramlás esetén

$$S = \left((1/Da) + 2(n+1)F/3n \right)^{1/(n+1)} \quad (17)$$

(lásd [9]).

Ezen probléma megoldását implicit alakban Magyari és Pantokratoras [9] adta meg

$$Y = \int_0^U \left[\frac{1}{nDa} (U^{n+1} - (n+1)U + n) + \frac{(n+1)F(U^3 - 3U + 2)}{3n} \right]^{-1/(n+1)} dU \quad (18)$$

alakban.

A [9] cikkben a szerzők meghatározták az $n=1$ newtoni és az $n=2$ dilatáló folyadék esetén az egzakt megoldást. Ha $n \neq 1$ vagy $n \neq 2$, akkor a peremérték feladat megoldása zárt alakban nem ismert.

Teljesen kifejlődött áramlás esetén tetszőleges n esetén az $U(Y)$ sebességkomponenst a felírt (15) másodrendű nemlineáris differenciálegyenletből az

$$U(0) = 0 \text{ és } \left. \frac{dU}{dY} \right|_{Y=0} = S \quad (19)$$

kezdeti feltételek mellett határozzuk meg.

3. HATVÁNSOR ALAKÚ MEGOLDÁS

A megoldást $U = Y \cdot Q(Y^\alpha)$ alakban keressük, ahol az α kitevő meghatározásához az [1]-ben ismertetett módszert alkalmazzuk. A hatványsor alakú megoldás létezésére a Briot-Bouquet tételt alkalmazhatjuk. A másodrendű differenciálegyenletet speciális Briot-Bouquet egyenletekből álló egyenletrendszerrel helyettesítjük, amire érvényes a következő tétel:

3.1. Briot-Bouquet tétel: Tegyük fel, hogy a

$$\left\{ \begin{aligned} \xi (dz_1 / d\xi) &= u_1(\xi, z_1(\xi), z_2(\xi)) \\ \xi (dz_2 / d\xi) &= u_2(\xi, z_1(\xi), z_2(\xi)) \end{aligned} \right\}$$

egyenletrendszer esetén, u_1 és u_2 holomorf függvényei a ξ , $z_1(\xi)$ és $z_2(\xi)$ változóknak, továbbá $u_1(0,0,0) = u_2(0,0,0) = 0$, akkor a fenti egyenletrendszernek létezik az origó közelében holomorf megoldása, amely kielégíti a $z_1(0) = 0$, $z_2(0) = 0$ kezdeti feltételeket feltéve, ha a

$$\begin{bmatrix} \partial u_1 / \partial z_1(0,0,0) & \partial u_1 / \partial z_2(0,0,0) \\ \partial u_2 / \partial z_1(0,0,0) & \partial u_2 / \partial z_2(0,0,0) \end{bmatrix}$$

mátrix egyik sajátértéke sem pozitív egész szám.

A tétel bizonyítását lásd a [3] cikkben.

A $U = Y \cdot Q(Y^\alpha)$ alakú megoldásokat keresve meghatározzuk az

$$U'(Y) = Q(Y^\alpha) + \alpha \cdot Y^\alpha \cdot Q'(Y^\alpha) \quad (20)$$

$$U''(Y) = \alpha \cdot (\alpha + 1) \cdot Y^{\alpha-1} Q'(Y^\alpha) \quad (21)$$

deriváltakat, majd a (15) egyenletbe helyettesítve kifejezzük $Q''(Y^\alpha)$ -t.

Elvégezzük az $Y^\alpha = \xi$ helyettesítést és feltételezzük, hogy

$$Q(\xi) = b_0 + b_1 \xi + z(\xi),$$

ahol z legalább kétszer differenciálható függvény és teljesíti a $z(0) = 0, z'(0) = 0$ feltételeket. Vezessük be a következő jelöléseket:

$$\begin{cases} z_1(\xi) = z(\xi) \\ z_2(\xi) = z'(\xi), \end{cases}$$

ahol $z_1(0) = 0$ és $z_2(0) = 0$. Ezek után legyenek az u_1 és u_2 függvények:

$$\begin{cases} u_1(\xi, z_1, z_2) = \xi \cdot z_2 \\ u_2(\xi, z_1, z_2) = \xi \cdot z_2' = \xi \cdot Q''(\xi). \end{cases}$$

A (15) differenciálegyenletből az $U = Y \cdot Q(Y^\alpha)$ helyettesítéssel a

$$Q''(\xi) = -\frac{\xi^{1-2\alpha}}{n\alpha^2 [Q(\xi) + \alpha\xi Q'(\xi)]^{n-1}} \left[\frac{1}{Da} \left(1 - \xi^\alpha Q^n(\xi) \right) + F \left(1 - \xi^\alpha Q^2(\xi) \right) \right] - \frac{\alpha+1}{\alpha} \cdot \frac{1}{\xi} Q'(\xi)$$

összefüggést kapjuk.

A Briot-Bouquet tételben szereplő $u_1(0,0,0) = 0$, $u_2(0,0,0) = 0$ feltételek csak akkor teljesülhetnek, ha

$$-\frac{\xi^{\frac{1}{\alpha}-1}}{n \cdot \alpha^2 \cdot b_0^{n-1}} \left[\frac{1}{Da} + F \right] - \frac{\alpha+1}{\alpha} b_1 = 0,$$

azaz $(1/\alpha) - 1 = 0$, és innen $\alpha = 1$. Az $u_2(0,0,0) = 0$ megkötésből

$$b_1 = -\frac{\frac{1}{Da} + F}{\alpha(\alpha+1)b_0^{n-1}}$$

adódik. Tehát $\alpha = 1$ esetén a

$$b_1 = -\frac{\frac{1}{Da} + F}{2b_0^{n-1}}$$

összefüggés érvényes. Az $U'(0) = Q(0) = b_0$ kezdeti feltételt figyelembe véve (16)-tal

$$b_0 = S.$$

Mivel így a Briot-Bouquet tétel feltételei a (15) differenciálegyenletre a (19) kezdeti feltételekkel teljesülnek b_0 és b_1 megfelelő megválasztásával, ezért a Briot-Bouquet differenciálegyenlet rendszernek létezik $z_1 = \sum_{k=1}^{\infty} a_k \xi^k$ és $z_2 = \sum_{k=1}^{\infty} b_k \xi^k$ alakú megoldása.

A (15), (19) kezdetiérték probléma megoldásainak hatványsora tehát

$$U = Y \sum_{i=0}^{\infty} b_i Y^i = b_0 Y + b_1 Y^2 + b_2 Y^3 + \dots \quad (22)$$

alakban kereshető.

Ezek után meghatározzuk a még ismeretlen együtthatókat. Az

$$U' = \sum_{i=0}^{\infty} (i+1)b_i Y^i = b_0 + 2b_1 Y + 3b_2 Y^2 + \dots,$$

$$U'' = \sum_{j=1}^{\infty} j(j+1)b_j Y^{j-1} = 2b_1 + 6b_2 Y + 12b_3 Y^2 + \dots,$$

deriváltakat a (15) differenciálegyenletbe helyettesítjük.

Ahhoz, hogy megadjuk a b_i együtthatókat, ha $i \geq 2$, a

J.C.P. Miller [5] formulát alkalmazzuk, majd az egyenlet két oldalán a megfelelő kitevőjű tagok együtthatóit összehasonlítva a (22) kifejezésben lévő b_2, b_3, b_4, \dots

együtthatók a kapott rekurzív formulákból előállíthatóak.

4. ÖSSZEFOGLALÁS

A dolgozatban hatványtörvény szerinti nemnewtoni közeg vízszintes síklap feletti kényszeráramlását leíró differenciálegyenlet rendszer hatványsor alakú megoldásának létezését mutattuk meg, ha a lap Darcy-Brinkman-Forchheimer porózus közegbe van beágyazva.

Megadtuk, hogy a folytonossági- és mozgásegyenletekből származtatott az áramló közeg síklappal párhuzamos U sebesség komponensére vonatkozó közönséges differenciálegyenlet $U = Y \cdot Q(Y)$ alakú megoldásai mikor elégítik ki a kezdetiérték feltételeket. A Briot-Bouquet-tétel alapján kiszámított b_0 és b_1 együtthatókon kívül a J.C.P. Miller formulával a

$$U = Y \sum_{i=0}^{\infty} b_i Y^i$$

hatványsorban szereplő b_2, b_3, b_4, \dots együtthatók rekurzív formulákkal előállíthatóak.

KÖSZÖNETNYILVÁNÍTÁS

A publikáció a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként - az Új Magyarország Fejlesztési Terv keretében - az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

IRODALOM

- [1] Bognár, G. (2008) Local analytic solutions to some nonhomogeneous problems with p-Laplacian, *Journal of Qualitative Theory of Differential Equations*, pp.:1-8.
- [2] Bognár, G., Rozgonyi, E. (2006) The power series solutions of some nonlinear initial value problems, *WSEAS Transactions on Mathematics*, **6**, pp.: 627-635.
- [3] Briot, CH., Bouquet, J.K. (1856) Etude des fonctions d'une variable imaginaire, *Journal de l'Ecole Polytechnique, Cahier* **36**, pp.:85-131.
- [4] Hady, F.M., Ibrahim, F.S. (1997) Forced convection heat transfer on a flat plate embedded in porous media for power-law fluids. *Transport Porous Med.* **28**, pp.:125-134.
- [5] Henrici, P. (1974) Applied and computational complex analysis, Power series integration-conformal mappings-location of zeros. Wiley, New York-London-Sydney-Toronto.
- [6] Hooman, K. (2008) A perturbation solution for forced convection in a porous saturated duct. *J. Comput. Appl. Math.* **211**, pp.:57-66.
- [7] Hsu, C.T., Cheng, P. (1990) Thermal dispersion in a porous medium. *Int. J. Heat Mass Transf.* **33**, pp.:1587-1597.
- [8] Joseph, D.D., Nield, D.A., Papanicolaou, G. (1982) Nonlinear equation governing flow in a saturated porous medium *Water Resour. Res.*, **18**, pp.:1049-1052.
- [9] Magyari, E., Pantokratoras, A. (2010) Forced convection flow of power-law fluids over a flat plate embedded in a Darcy-Brinkman porous medium. *Transp. Porous Med.* **85**, pp.: 143-155.
- [10] Nield, D.A., Bejan, A. (2006) *Convection in porous media*, 3rd ed. Springer, New York.
- [11] Poulidakos, D., Renken, K. (1987) Forced convection in a channel filled with porous medium, including the effect of flow inertia, variable porosity and Brinkman friction. *ASME J. Heat Transf.* **109**, pp.: 880-888.
- [12] Shenoy, A.V. (1994) Non-newtonian fluid heat transfer in porous media. *Adv. Heat Trans.* **24**, pp.:101-190.

A YANG-BAXTER EGYENLET NÉHÁNY MEGOLDÁSÁNAK GRAFIKUS REPREZENTÁCIÓJA

Graphical representations of certain solutions of the Yang-Baxter equation

Varga Péter*

ABSTRACT

The Yang-Baxter equation is a basic equation of the theory of integrable system. It was realized later, that it can be interpreted as an invariance principle of certain structures under the third Reidemeister move of knot theory. This observation made it possible to interpret certain algebraic equations as the topological invariance of certain drawings of knots or braids. We extend that framework by adding strings that are cut at certain points. That allows us to provide a graphical interpretation for certain solutions of the Yang-Baxter equation. Our approach also motivates the introduction of a generalization of the Temperley-Lieb algebra.

1. BEVEZETÉS

A Yang-Baxter egyenlet az egzaktul megoldható kétdimenziós matematikai fizikai rendszerek elméletének az egyik sarokköve. Eredetileg a kétdimenziós Ising model (ami a mágnesességnek egy idealizált modelje) bukkant fel Onsager [1] híres munkájában. Az egyenlet további vizsgálata a kezdeti időkben leginkább Yang [2] és Baxter [3] nevéhez köthető. Ők további alkalmazásait fejlesztették ki ennek az egyenletnek az egydimenziós kvantummechanikai rendszerek, illetve a kétdimenziós, egy rácson megadott statisztikus mechanika rendszerek esetében. Eleinte az egyenletet egy nagyon komplikált harmadfokú sokismeretlenes egyenletnek tekintették, aminek a megoldásait ad hoc számításokkal származtatták. Később felfedezték, hogy az egyenlet legintuitívabb interpretációja a csomóelmélet keretében adható meg. (Itt megjegyezzük, hogy ez szigorúan véve csak az úgynevezett spektrális paraméter nélküli esetben igaz. Mi a továbbiakban ezzel az esettel foglalkozunk míg bizonyos alkalmazások az általánosabb eset vizsgálatát is megkövetelik.) Kiderült, hogy az ismeretlen, tradicionálisan R -nek nevezett mátrixra

vonatkozó tenzoriális kommutációs reláció nem más, mint csomóelméletből ismert úgynevezett harmadik Reidemeister [4] lépés algebrai interpretációja. Ebben a keretben sikerült például a csomók Jones polinomját (ami eredetileg a von Neumann algebrak elméletében merült fel [5]) egy teljesen elemi, grafikusan is ábrázolható módon interpretálni [6]. Mivel a Yang-Baxter egyenlet ezeken a területeken kívül is fontos szerepet, játszik (itt elsősorban a kvantum csoportok és a Hopf algebrak elméletét érdeme kiemelni [7,8]), meglehetősen komoly erőfeszítések irányultak az egyenlet megoldásainak a megkeresése felé. Mivel azonban az R mátrix egy n^4 ismeretlen tartalmazó objektum, a megoldások teljes klasszifikációja csak az $n=2$ esetben járt teljes sikerrel [9]. Mi több, a megoldások reprezentációja vagy interpretációja sokszor legalább olyan fontos (mint ezt a [9] dolgozat hangsúlyozza), mint azok megkeresése. Ebből a szempontból a lelegegánsabb eredmények Kaufmann nevéhez fűződnek [13], akinek sikerült igen komplikált csomóelméleti vagy algebrai számításokat grafikus manipulációkká alakítani.

Mi ezt a tradíciót kívánjuk követni a dolgozatunkban. Megvizsgáljuk, hogy mi történik, ha a csomóelméleti ábrákon megengedjük azt, hogy a csomót reprezentáló görbék elszakadjanak, ne legyenek folytonosak. Ez kibővíti a grafikusan ábrázolható megoldások körét. Mi több, ez a megközelítés motiválhatja néhány algebra általánosítását, itt leginkább a Temperley-Lieb [10], és a Birman-Murakami-Wenzl [11,12] algebrak jöhetnek számításba. A következőekben először az összehasonlítás kedvéért felidézünk a Yang-Baxter egyenletnek azt a megoldását, amelyet a Temperley-Lieb algebra szolgáltat, illetve szintén megadjuk ennek a jól ismert grafikus jelentését. Ezután megnézzük, hogyan módosul ez a séma, ha a Temperley-Lieb algebra ábrázolásához használt félköröket és szakaszokat estelegesen megszakadt vonalakkal helyettesítjük. Egy egyszerű esetben explicit módon is elvégezzük azt a kalkulációt, ami a Yang-Baxter egyenlet teljesülését biztosítja. Az általánosabb eset vizsgálatához

* egyetemi docens, Miskolci Egyetem, Analízis Tanszék

szimbólikus algebrai csomagot használunk. Végezetül megjegyezzük, hogy az így kapott grafikus kalkulus egy természetes kiterjesztését adja meg például a Temperely-Lieb algebrának.

2. GRAFIKUS KALKULUS ÉS A CSOMÓELMÉLET

Először is idézzük fel a Yang-Baxter egyenlet definícióját. Legyen R egy n^4 méretű mátrix, amihez a következő lineáris transzformáció tartozik:

$$R: V \times V \rightarrow V \times V$$

Itt V egy n dimenziós vektortér, vagyis R a V vektortér önmagával vett direkt szorzatán hat. Legyen továbbá R_{12} és R_{23} a következő:

$$R_{AB} : V \times V \times V \rightarrow V \times V \times V$$

$$R_{12} = R \times \text{id}, \quad R_{23} = \text{id} \times R.$$

Ekkor a Yang-Baxter egyenlet a következő lesz:

$$R_{12} R_{23} R_{12} = R_{23} R_{12} R_{23}.$$

Ez a reláció nem más, mint az Artin féle fonatcsoportból jól ismert nemtriviális kommutációs reláció megfelelője:

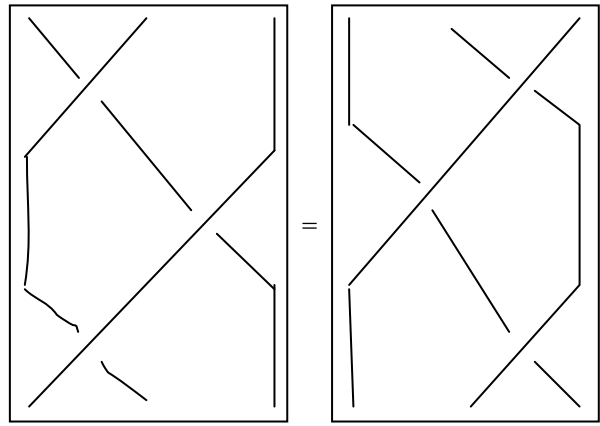
$$s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1}.$$

Itt s_i a fonat i -edik és az $(i+1)$ -edik szálainak kereszteződését jelöli. Tehát a Yang-Baxter egyenlet megoldásai reprezentálják az Artin féle kommutációs relációt, ha pedig az R mátrix invertálható, akkor a fonatcsoport reprezentációja is megkonstruálható belőlük. Ennek alapján a következő megfeleltetés állítható fel:

$$R = \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array}$$

Itt úgy gondolunk a vonalak végpontjaira, mint amelyekhez hozzá van rendelve V egy kópiája, a tenzorszorzatot pedig a vízszintesen egymás mellett elhelyezkedő végpontok reprezentálják.

Ez a megfeleltetés lehetővé teszi a Yang-Baxter egyenletnek a következő formájú grafikus interpretációját:



Ez nem más, mint a csomóelméletből ismert harmadik Reidemeister féle lépés, ha az itt látható vonalakat térben képzeljük el, akkor ez a lépés nyilvánvalóan folytonosan elvégezhető a vonalak elvágása nélkül.

A Yang-Baxter egyenletnek talán a leghíresebb megoldása (ez reprodukálja a Jones polinomot, illetve ez merül fel az $SL(2)$ csoport kvantum verziójában is) a következőképpen ábrázolható a grafikus kalkulus segítségével. Rendeljük hozzá az R mátrixhoz, vagyis egy kereszteződéshez a következő szimbólumok súlyozott összegét:

$$\begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} = q \left(\begin{array}{c} | \\ | \end{array} + \begin{array}{c} \diagdown \quad \diagup \\ \diagup \quad \diagdown \end{array} \right)$$

Ekkor az előző két ábra mindegyikén a három kereszteződés (mivel mindegyik esetében két variáció van) szétesik 8 darabra, amelyek közül nemelyik egy zárt hatszöget (hurkot) is tartalmaz is tartalmaz. Ha a hurokhoz az u skaláris szorzófaktorot rendeljük hozzá, akkor a jobb és bal oldali ábrák szorzófaktorai megegyeznek, mégpedig abban az értelemben, hogy két ábra ekvivalens, ha az alul és felül elhelyezkedő pontok ugyanazon módon vannak összekötve. Ezek az ábrák természetesen átkonvertálhatók mátrixokká is. Ekkor egy függőleges vonaldarabhoz a Kronecker delta mátrix tartozik, míg egy v alakhoz egy tetszőleges M_{ab} mátrix tartozik. Annak az érdekében, hogy egy v és egy fordított v kompozíciójaához az identikus mátrix tartozzon, a fordított v -hez M inverzét kell rendelnünk. Ekkor egy zárt hurokhoz az

$$u = \text{Tr}(M^T M^{-1})$$

faktor rendelődik hozzá.

Ezekhez a konstrukciókhoz hozzárendelhető egy érdekes algebrai struktúra, a Temperley-Lieb algebra. Ennek az elemeit a következő grafikus szimbólumok generálják:

$$U_i = \left| \begin{array}{c} \dots\dots\dots \\ \text{---} \\ 1 \end{array} \right| \quad \left| \begin{array}{c} \vee \\ \wedge \\ i \quad i+1 \end{array} \right| \quad \left| \begin{array}{c} \dots\dots\dots \end{array} \right|$$

Ez kielégíti a az alapvető

$$U_i^2 = u U$$

relációt, plusz további relációkat is, amelyeket ezenek az ábráknak a csatlakoztatása generál [10].

Mi a következőképpen próbáljuk kiterjeszteni ezt a kalkulust. Az R mátrixot megpróbáljuk felírni olyan grafikus szimbólumok formális lineáris kombinációjaként, amely ábrák elszakadt vonalakat is tartalmaznak. Vagyis

$$R = \left| \begin{array}{c} \diagup \\ \diagdown \end{array} \right| = A+B+C+D+E+F+G+H+J$$

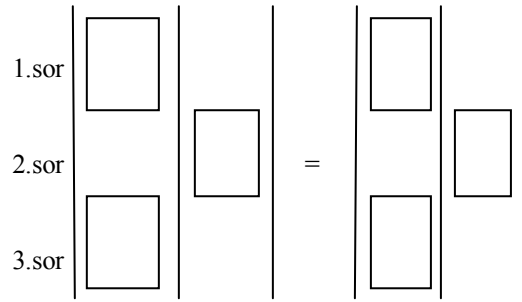
$$= \left| \begin{array}{c} | \\ | \\ | \end{array} \right| + \left| \begin{array}{c} \vee \\ \wedge \end{array} \right| + \left| \begin{array}{c} | \\ | \\ | \end{array} \right| + \left| \begin{array}{c} | \\ | \\ | \end{array} \right| + \left| \begin{array}{c} \vee \\ \wedge \end{array} \right| + \left| \begin{array}{c} | \\ | \\ | \end{array} \right|$$

$$+ \left| \begin{array}{c} | \\ / \\ | \end{array} \right| + \left| \begin{array}{c} \backslash \\ | \end{array} \right| + \left| \begin{array}{c} | \\ | \\ | \end{array} \right|$$

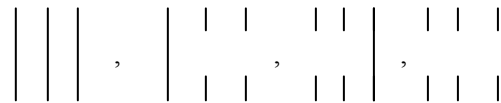
Itt az egyes szimbólumok együtthatóit nem írtuk ki. Ezek után feltehető az a kérdés, hogy milyen együtthatóválasztás esetén teljesül a Yang-Baxter egyenlet. Teljes általánosságban ez a kérdés persze csak terjedelmes, számítógépes algebrai számításokkal válaszolható meg, azonban néhány egyszerű esetben a kalkulációk kézzel is elvégezhetőek. Vegyük azt a redukált esetet, amikor csak az első és az utolsó tagoknak van nem nulla együtthatója:

$$R = A + iI = \left| \begin{array}{c} | \\ | \\ | \end{array} \right| + i \left| \begin{array}{c} | \\ | \\ | \end{array} \right|$$

Helyettesítsünk most be ezt a két szimbólumot a Yang-Baxter egyenlet grafikus reprezentációjainak a kereszteződéseket tartalmazó helyeire. Ekkor az alábbi ábrák mindegyike nyolc további ábrát generál.



Itt az ábrán látható négyzetek mindegyikébe vagy az A vagy a I szimbólumot írhatjuk. Összesen négyféle összekapcsolódási mintázat jöhet létre az alsó és afelső három pont között:



Rendeljünk továbbá minden szabad szakaszdarabhoz egy v szorzófaktorot. Ekkor a baloldali YB ábrán ennek a négy konfigurációnak a súlyfaktorai

$$1, \quad i, \quad i^2 v^2 + 2i, \quad 2i^3 v^2 + i^2$$

lesznek. Ugyanez a jobboldali ábrán

$$1, \quad i^2 v^2 + 2i, \quad i, \quad 2i^3 v^2 + i^2$$

lesz. Ennek alapján azt kapjuk, hogy

$$i = -1/v^2,$$

vagyis ezen választás mellett teljesül a Yang-Baxter egyenlet. Hogyan tudjuk ezt, vagy az ehhez hasonló, szabad végpontokat is tartalmazó megoldást numerikus R mátrixszá konvertálni? Ennek érdekében egy alappont-végpont vonalhoz rendeljük hozzá a $\delta_{0,a}$ mátrixelemet.

Bonyolultabb, több paramétert is tartalmazó R mátrix esetén ezek a számítások persze egyre hosszabbá válnak. Itt már célszerű a szimbólikus algebrai csomagok használat. Mi egy Mathematicában lekódolt program segítségével kerestük a Yang-Baxter egyenlet

$$R = A + bB + cC + cD + dE + dF + eG + eH + fI$$

alakú megoldásait. Az egyenletek megoldása nem ütközött különösebb nehézségbe, mi itt felsorolunk néhányat a kapott megoldások közül.

- a) $b=0, v=0,$
 $f=-2c^2 - 4cd - 4ce - 4de - 2e^2 - d^2u.$
- b) $u=(-1 - b^2)/b, v=0.$
- c) $b=0, v=0, d=0, f=-2(c^2 + 2ce + e^2).$
- d) $b=0, c=-(1/v), d=0.$
- e) $b=0, v=0, d=0, e=0, f=-2c^2.$
- f) $u=(-1 - b^2)/b, v=0, c=0,$
 $f=3bd^2 + 5bde + 2be^2$
 $+ 3d^2u + 5deu + 2e^2u.$
- g) $u=((-1 + b - b^2)/b), c=-(1/v), d=bc, e=0.$
- h) $u=(-1 - b^2)/b, v=0, c=0, e=-2d,$
 $f=bd^2 + (d^2).$
- i) $u=(-1 + 2b - b^2)/b, c=0, d=-((1 + I)b)/v,$
 $e=0, f=2d^2 - bd^2 - d^2u.$

Az általánosabb eset vizsgálata valószínűleg csak specializáltabb egyenletmegoldó programok segítségével lenne lehetséges. További vizsgálatokat igényelne az a kérdés is, hogy mely megoldások hasonlóak (lineáris algebrai értelemben) egymáshoz.

5. ÖSSZEFOGLALÁS

A Yang-Baxter egyenlet megoldásait vizsgáltuk egy, a csomóelmélethez kapcsolódó grafikus kalkulus segítségével. Ezt, a téma irodalmában sokat vizsgált módszert azzal az ötlettel bővítettük ki, hogy olyan ábrákat is használtunk, ahol megengedtük, hogy a Yang-Baxter egyenletet reprezentáló csomóelméleti ábrákon megengedtük a szálak elszakadását. Az így kapott ábrák vizsgálatát egy viszonylag egyszerű esetben kézzel, bonyolultabb esetekben számítógéppel végeztük el.

KÖSZÖNETNYÍLVÁNÍTÁS

A tanulmány a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként - az Új Magyarország Fejlesztési Terv keretében - az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg. A kutatást az Országos Tudományos Kutatási Alapprogramok támogatta az OTKA T 75678 számú projekt keretében.

IRODALOM

- [1] Onsager L (1971) The Ising model in twodimensions. In: Mills RE, Ascher E. and Jaffee RL. (eds) Critical phenomena in alloys, magnets and superconductors, pp xix–xxiv, 3–12. McGraw-Hill, New York.
- [2] Yang CN (1967) Some exact results for the many-body problem in one dimension with repulsive delta-function interaction. Phys. Rev. Lett. 19: 1312–1314.

- [3] Baxter RJ (1982) Exactly Solved Models in Statistical Mechanics. Academic Press, London.
- [4] Reidemeister K (1926) Knoten und Gruppen. Abh. Math. Sem. Hamburg. Univ. 5: 7–23. Elementare Begründung der Knotentheorie. ibid. 24–32.
- [5] Louis H. Kauffman, State models and the Jones polynomial. Topology 26 (1987), no. 3, 395–407.
- [6] Louis H. Kauffman, Temperley-Lieb Recoupling Theory and Invariants of 3-Manifolds, with Sostenes Links, Princeton University Press, 312 pp, 1994.
- [7] Kassel, Christian (1995), Quantum groups, Graduate Texts in Mathematics, 155, Berlin, New York: Springer-Verlag
- [8] Dăscălescu, Sorin; Năstăsescu, Constantin; Raianu, Şerban (2001), Hopf Algebras, Pure and Applied Mathematics, 235 (1st ed.), Marcel Dekker,
- [9] J. Hietarinta: All solutions to the constant quantum Yang-Baxter equation in two dimensions, Phys. Lett. A 165, 245-251 (1992).
- [10] N. Temperley, E. Lieb, Relations between the percolation and colouring problem and other graph-theoretical problems associated with regular planar lattices: some exact results for the percolation problem. Proceedings of the Royal Soc. Series A322(1971), 25.280.
- [11] Birman, Joan S.; Wenzl, Hans (1989), "Braids, link polynomials and a new algebra", Transactions of the American Mathematical Society (American Mathematical Society) 313 (1): 249–273,
- [12] Murakami, Jun (1987), "The Kauffman polynomial of links and representation theory", Osaka Journal of Mathematics 24 (4): 745–758

NÉHÁNY ÁLTALÁNOSÍTOTT TODA RÁCS MÓDOSÍTOTT LAX REPEZENTÁCIÓJA

MODIFIED LAX REPRESENTATION OF CERTAIN GENERALIZED TODA CHAINS

Varga Péter*

ABSTRACT

We present a modified Lax pair representation of certain generalized Toda chains. We use a scheme of Drinfeld and Sokolov that replaces the usual commutator of matrices by a more general expression. We apply a reduction of their general setting to rederive the equation of motion of certain Toda chains. Our method uses (in some sense) smaller groups for the description of these dynamical systems.

1. BEVEZETÉS

A teljesen integrálható dinamikai rendszerek elméletének egyik alapvető dolgozata [1], ahol Lax Péter egy mátrix (vagy egy differenciáloperátor) mozgásegyenletét az

$$L' = LP - PL = [L, P]$$

alakban írta fel. Ez az L operátor infinitezimális hasonlósági transzformációja, így megőrzi L spektrális invariánsait. Ennek a sémának egy általánosítását adta meg Drinfeld és Sokolov [2], akik az $L=AB$ szorzat hasonlósági transzformációjának az A és a B mátrixokra való hatását vizsgálták. Konkrét eseteit ennek a módszernek Dodd-Fordy és Hirota-Satsuma tanulmányozta [3,4]. Egy régebbi dolgozatom [5] azt a kérdést vizsgálta, hogy mi történik akkor, ha A és B egymás adjungáltjai, vagyis $A^*=B$. A fent említett munkák mindegyike parciális differenciálegyenleteket írt le, míg a jelenlegi dolgozatom a közönséges differenciálegyenletekkel foglalkozik az úgynevezett Toda [6,7,8] rácsok esetében. Miután az összehasonlítás végett felírjuk az A_3 rács Lax párját és mozgásegyenletét, megmutatjuk, hogy egy két oszloppal kibővített L mátrix segítségével megkapható például a B_3 (az egyik ortogonális csoporthoz tartozó) Toda rács. Az irodalomban általánosan használt reprezentáció megduplázná a mátrix méretét.

2. MÓDOSÍTOTT LAX PÁROK

Jól ismert, hogy az

$$L' = LP - PL = [L, P]$$

egyenlet az

$$L(0) \rightarrow \exp(-tP) L(0) \exp(tP)$$

hasonlósági transzformáció infinitezimális alakja. Ez biztosítja, hogy L és hatványainak a nyomai időben állandóak, megmaradó mennyiségek. Nézzük most az általánosított

$$L \rightarrow \exp(-tB) L \exp(tA)$$

transzformációt.

Tekintsük most az LL^T időbeli változását:

$$LL^T \rightarrow \exp(-tB) L \exp(tA) \exp(tA^T) L^T \exp(-tB^T)$$

Ez továbbra is hasonlósági transzformáció marad, ha B transzponáltja $-B$, illetve, ha $A^T=-A$, hiszen ekkor az L és az L^T közötti tag 1. Jegyezzük meg, hogy bár az A és B mátrixoknak kötelező négyzet alakúaknak lenniük, az L mátrix nyugodtan lehet téglalap alakú is. Hasonló érvelés alkalmazható az $L^T L$ szorzat esetében is, de ez a mi szempontunkból nem vezet új megmaradó mennyiségek létezéséhez, hiszen pl. $\text{Tr}(LL^T)=\text{Tr}(L^T L)$.

Mindezek alapján levonhatjuk a következtetést, hogy ha egy L mátrix kielégíti a következő egyenleteket:

$$L' = LA - BL, \quad A = -A^T, \quad B = -B^T,$$

akkor az $L^T L$ szorzat sajátértékei megmaradó mennyiségek. Természetesen a megmaradó mennyiségek száma általában csekély a mátrix elemeinek a számához képest, így csak a megfelelően választott L , A , B hármas esetén kapunk teljesen integrálható rendszert.

* egyetemi docens, Miskolci Egyetem, Analízis Tanszék

Ez az eljárás egy speciális esete Drinfeld és Sokolov általánosabb sémájának [2]. Ők az MN sorozat izospektrális deformációit vizsgálták. megfigyelték, hogy ha

$$MN \rightarrow \exp(-tB) M \exp(tA) \exp(-tA) N \exp(tB),$$

akkor M és N mozgásegyenletei

$$M' = MA - BM, N' = NB - AN.$$

Ennek a sémának a fentebb leírt variánásán kívül más lehetőségek is vannak, például vizsgálható az LL^* , az LL^{T-1} , LL^{*-1} operátorok izospektrális deformációja is [5].

3. EGY TODA RÁCS ÉS ANNAK STANDARD LAX PÁRJA

A legegyszerűbb példa a Toda rácsokra az A_N sorozat. Ennek tagjainak a mozgásegyenlete a következő:

$$q_{ij} = \exp(x_i - x_j), p_i = x'_i, \\ p'_i = x''_i = -q_{i,i+1}^2 + q_{i-1,i}^2,$$

kivéve az első és az utolsó tagokat, ahol az egyik négyzet értelemszerűen hiányzik (itt $i=1..N$).

Ennek az egyenletnek a Lax reprezentációja az $N=3$ esetben a következő:

$$L = \begin{pmatrix} p_1 & q_{12} & 0 \\ q_{12} & p_2 & q_{23} \\ 0 & q_{23} & p_3 \end{pmatrix}$$

$$A = \begin{pmatrix} 0 & q_{12} & 0 \\ -q_{12} & 0 & q_{23} \\ 0 & -q_{23} & 0 \end{pmatrix}$$

A két mátrix $L^2=LA-AL$ kommutátora:

$$L' = \begin{pmatrix} -2q_{12}^2 & p_1q_{12}-p_2q_{12} & 0 \\ p_1q_{12}-p_2q_{12} & 2q_{12}^2-2q_{23}^2 & p_2q_{23}-p_3q_{23} \\ 0 & p_2q_{23}-p_3q_{23} & 2q_{23}^2 \end{pmatrix}$$

Ez pontosan a fentebb leírt mozgásegyenleteket szolgáltatja. Ennek a konstrukciónak többféle általánosítása is létezik [6,7,8]. Ezek nagy része Lie-algebrái eszközöket használ, aminek következtében megváltozik a rács végpontjaiban lévő részecskék dinamikája. A legáltalánosabb végponti dinamika Izmontsev [9] nevéhez fűződik, azonban azt lehet

mondani, hogy a leghatékonyabb eljárásnak a végponti dinamika generálásához a Sklyanin [10] által kidolgozott tükrözési R métrix nevezhető. A következőekben leírunk egy eljárást, amivel az így kapott mozgásegyenleteknek egy részalmlaza reprodukálható.

4. EGY ÁLTALÁNOSÍTOTT TODA RÁCS ÉS ANNAK NEM STANDARD LAX PÁRJA

Bővítsük az L mátrixunkat két extra oszloppal a következő módon:

$$L = \begin{pmatrix} q_1 & p_1 & q_{12} & 0 & 0 \\ 0 & q_{12} & p_2 & q_{23} & 0 \\ 0 & 0 & q_{23} & p_3 & q_3 \end{pmatrix}$$

Válasszuk a következő két antiszimmetrikus mátrixot A és B szerepébe:

$$A = \begin{pmatrix} 0 & q_1 & 0 & 0 & 0 \\ -q_1 & 0 & q_{12} & 0 & 0 \\ 0 & -q_{12} & 0 & q_{23} & 0 \\ 0 & 0 & -q_{23} & 0 & q_3 \\ 0 & 0 & 0 & -q_3 & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & q_{12} & 0 \\ -q_{12} & 0 & q_{23} \\ 0 & -q_{23} & 0 \end{pmatrix}$$

Ekkor a módosított Lax egyenlet a következő alakú lesz:

$$L' = \begin{pmatrix} -p_1q_1 & q_1^2-2q_{12}^2 & p_1q_{12}-p_2q_{12} & 0 & 0 \\ 0 & p_1q_{12}-p_2q_{12} & 2q_{12}^2-2q_{23}^2 & p_2q_{23}-p_3q_{23} & 0 \\ 0 & 0 & p_2q_{23}-p_3q_{23} & 2q_{23}^2-q_3^2 & p_3q_3 \end{pmatrix}$$

Az extra q_1 és q_2 mennyiségeket a következő egyenlet határozza meg:

$$q_1 = \exp(-x_1), p_1 = x'_1, \\ q_3 = \exp(x_3), p_3 = x'_3.$$

Látható, hogy a végpontok, vagyis $x_{1,3}$ dinamikája megváltozott, azok alakja most

$$p'_1 = x''_1 = -q_{1,2}^2 + q_1^2, \\ p'_3 = x''_3 = -q_3^2 + q_{23}^2.$$

Ha csak az utolsó oszloppal végeztük volna el a bővítést, akkor pontosan a B_3 néven ismert, az egyik

ortogonális csoporthoz tartozó Toda rácsot kapnánk. Az itt leírt konstrukció viszont lehetőséget ad arra, hogy ugyanezt a dinamikai rendszert az A_3 Lie algebra segítségével konstruáljuk meg. Mi több, a mindkét végén kibővített rácsok csoportelméleti leírása meglehetősen komplikált, így az itt ismertetett konstrukció alkalmas arra, hogy néhány eset tárgyalását egyesítse. Természetesen hangsúlyozzuk, hogy a Skylanin féle végpontbeli reflexiók mátrixok elmélete általánosabb rendszerek megkonstruálását is lehetővé teszi. Mi a példáinkat a három dimenziós esetben írtuk le, ezek általánosítása tetszőleges dimenziókra legfeljebb jelölésbeli nehézségekkel járna.

A következőekben megpróbálkozunk az itt leírt eljárás egy további általánosításával, azonban, mint az később ki fog derülni, ilyen módon már lényegében nem kapunk új dinamikai rendszereket. Felbuzdulva az eddigi eredményeinken, próbáljuk meg az A_4 rács standard Lax mátrixa jobb szélét két extra oszloppal bővíteni, vagyis legyen L a következő alakú:

$$\begin{pmatrix} p_1 & q_{12} & 0 & 0 & 0 & 0 \\ q_{12} & p_2 & q_{23} & 0 & 0 & 0 \\ 0 & q_{23} & p_3 & q_{34} & 0 & 0 \\ 0 & 0 & q_{34} & p_4 & a & d \end{pmatrix}$$

Ezzel a formával összhangban válasszuk A -t a következő mátrixnak:

$$\begin{pmatrix} 0 & q_{12} & 0 & 0 & 0 & 0 \\ -q_{12} & 0 & q_{23} & 0 & 0 & 0 \\ 0 & -q_{23} & 0 & q_{34} & 0 & 0 \\ 0 & 0 & -q_{34} & 0 & a & d \\ 0 & 0 & 0 & -a & 0 & p_4 \\ 0 & 0 & 0 & -d & -p_4 & 0 \end{pmatrix}$$

Továbbá legyen a B mátrix formája

$$\begin{pmatrix} 0 & q_{12} & 0 & 0 \\ -q_{12} & 0 & q_{23} & 0 \\ 0 & -q_{23} & 0 & q_{34} \\ 0 & 0 & -q_{34} & 0 \end{pmatrix}$$

Ekkor az

$$L' = LA - BL,$$

kommutátorra a következő kifejezést kapjuk:

$$\begin{pmatrix} -2q_{12}^2 & p_1q_{12} - p_2q_{12} & 0 & 0 & 0 & 0 \\ p_1q_{12} - p_2q_{12} & 2q_{12}^2 - 2q_{23}^2 & p_2q_{23} - p_3q_{23} & 0 & 0 & 0 \\ 0 & p_2q_{23} - p_3q_{23} & 2q_{23}^2 - 2q_{34}^2 & p_3q_{34} - p_4q_{34} & 0 & 0 \\ 0 & 0 & p_3q_{34} - p_4q_{34} & -a^2 - d^2 + 2q_{34}^2 & ap_4 - dp_4 & ap_4 + dp_4 \end{pmatrix}$$

A legszó két nem nulla eleme ennek a mátrixnak a következő differenciálegyenleteket adja a két ismeretlen a és b függvényekre:

$$a'(x) = a(x) - d(x), d'(x) = a(x) + d(x).$$

Ennek az egyenletrendszernek a legáltalánosabb megoldása

$$a(x) = \exp(x) (u \cos(x) + v \sin(x)), \\ d(x) = \exp(x) (v \cos(x) - u \sin(x)).$$

Mivel az $x''_4 = p'_4$ gyorsulást leíró mátrixelemben a következő extra tag: $-d^2 - a^2$ jelenik meg, ami egyenlő $\exp(x)$ többszörösével, így az így kapott dinamikát lehetséges a „d” oszlop hozzáadása nélkül is generálni. Mindazonáltal az itt leírt módszer többféle irányban is általánosítható. Sikerült az $Sp(2n)$ szimplektikus csoporthoz tartozó standard Lax párt úgy módosítani, hogy a végpontokban az Izmotsev féle általánosított Toda rácsoknál megjelenő inverz koszinusz hiperbolikus potenciál hasson. Ezt egy későbbi dolgozatban tervezzük publikálni.

A Lax pár létezése természetesen csak a nagyszámú megmaradó mennyiség létezését garantálja, a Liouville féle teljes integrálhatóságot ez önmagában még nem biztosítja. A Lax párok klasszikus elméletében a teljes integrálhatóság, vagyis az eset, amikor a független megmaradó mennyiségek száma egyenlő a fázisér dimenziójának a felével és ezek Poisson zárójel mind nulla egymással, a leeffektívebben a következő tétel segítségével bizonyítható.

Tétel []: Ha létezik olyan r mátrix, hogy

$$\{L_1, L_2\} = [r_{12}, L_1] - [r_{21}, L_2],$$

akkor a $\text{Tr}(L^n)$ mennyiségek Poisson zárójel nulla.

Ezt a tételt a következőképpen módosítottuk:

Létezzenek olyan l és r mátrixok, hogy

$$\{L_1, L_2\} = (l_{12} L_1 - L_1 r_{12}) - (l_{21} L_2 - L_2 r_{21}).$$

Ekkor a

$$H_n = \text{Tr}(LL^T)$$

mennyiségek Poisson zárójelle egymással nulla.

Ennek a tételnek az alkalmazása további kutatások tárgyát képezi.

5. ÖSSZEFOGLALÁS

A Drinfeld és Sokolov által kidolgozott módosított Lax párok elméletét a közönséges differenciálegyenletek elméletében, az általánosított Toda rácsok esetében alkalmaztuk. Ezen a módon sikerült néhány esetben alternatív, egyszerűbb, kisebb méretű Lax reprezentációt találnunk. Többek között sikerült a B_n típusú rácsokat az A_n rácsokhoz hasonló mátrixokkal reprezentálni. Végezetül megadtunk egy tételt, amelynek segítségével alternatív bizonyítás adható bizonyos rácsok Liouville féle integrálhatóságára.

KÖSZÖNETNYÍLVÁNÍTÁS

A tanulmány a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként - az Új Magyarország Fejlesztési Terv keretében - az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg. A kutatást az Országos Tudományos Kutatási Alapprogramok támogatta az OTKA T 75678 számú projekt keretében.

IRODALOM

- [1] Lax, P. (1968), "Integrals of nonlinear equations of evolution and solitary waves", *Comm. Pure Applied Math.* 21 (5): 467–490
- [2] Drinfeld, V.G.; Sokolov, V.V. Lie algebras and equations of KortewegdeVries type. (Russian) *Current problems in mathematics*, Vol. 24, 81–180, Itogi Nauki i Tekhniki, Akad. Nauk SSSR, Vsesoyuz. Inst. Nauchn. i Tekhn. Inform., Moscow, 1984.
- [3] R.K. Dodd and A.P. Fordy, On the integrability of a system of coupled KdV equations. *Phys.Letts. A* , 89 168-70, 1982.
- [4] W. Oevel, On the integrability of the Hirota-Satsuma system, *Physics Letters A*, Volume 94, Issue 9, 4 April 1983, Pages 404–407.

[5] Péter Varga, Unitary deformations and complex soliton equations, *J. Math. Phys.* 40, 3404 (1999)

[6] Toda, Morikazu (1989), *Theory of Nonlinear Lattices* (2 ed.), Berlin: Springer ISBN 978-0387102245

[7] Bertram Kostant. The solution to a generalized Toda lattice and representation theory. *Adv. in Math.*, 34(3):195–338, 1979.

[8] M. Olshanetsky, A. Perelomov, Classical integrable finite-dimensional systems related to Lie algebras, *Physics Reports*, v.71 (1981), 313-400

[9] Inozemtsev V.I., The finite Toda lattices, *Comm. Math. Phys.* 121 (1989), 629-638.

[10] E.K. Sklyanin, Boundary conditions for integrable quantum systems, *J. Phys. A: Math. Gen.*, 21, 2375-2389, (1988).

RENDEZÉS-KONGRUENCIÁK ALKALMAZÁSA TÖBBGÉPES PÁRHUZAMOS ÜTEMEZÉSEK ESETÉN

AN APPLICATION OF ORDER CONGRUENCES IN PARALLEL MULTI-MACHINES SCHEDULING PROBLEMS

*Szilágyi Szilvia**

ABSTRACT

Scheduling, sorting and searching problems are among the most common instances of order. Typically an order must be transformed to another, say a partial extension or a linear extension, which itself may represent a schedule or a sort. In a scheduling problem we have to find an optimal schedule of jobs. We consider the parallel machines case, where m machines are given and we can use them to schedule the jobs. Using the greedy algorithm by Trotter (for finding linear extensions of a poset) we exhibit and implement an algorithm for determining minimal linear order congruences, solving the given parallel scheduling problem.

1. BEVEZETÉS

Ha adott elvégzendő munkák egy M halmaza, akkor ennek egy ütemezésén az M halmaz elemeinek olyan permutációját értjük, amely megadja, hogy az egyes munkákat milyen sorrendben kell végrehajtani. Általános szabály, hogy minden, a munkák elvégzéséhez felhasznált erőforrás (gép) egy időben legfeljebb egy munkán dolgozhat és minden munkát egy időben legfeljebb egy erőforráson (gépen) végezhetünk. A termelési illetve gyártási folyamatok egy ismert modelljét úgy kapjuk, hogy egy részben-rendezett halmaz segítségével ábrázoljuk a gyártási folyamat egyes fázisai közötti összefüggéseket. Az ütemezési feladat klasszikus megoldása ekkor egy olyan lineáris rendezés, amely az eredeti részben-rendezés kiterjesztése. Ha egy termék gyártása során számos technológiai fázis van, akkor a termék általában igen különböző technológiai útvonalakat járhat be. Ha az üzemben az egyes fázisokon párhuzamos berendezések is vannak, akkor a termék által bejárható útvonalak száma meghatározható. Egy lehetséges megoldás, ha a gépeket úgy csoportosítjuk (térbeli és/vagy logikai dekompozíció), hogy azok a termék előállításához szükséges műveletek egy csoportját képesek legyenek

elvégezni. Ütemezési feladatok megoldásakor tehát lineáris rendezéseket keresünk. Szpilrajn klasszikus tétele értelmében bármely részben-rendezés kiterjeszhető lineárisra [10]. Vannak azonban a gyakorlatban olyan esetek, amikor nem szükséges az elemek között teljes lineáris rendezést megadni, hanem érdemesebb nagyobb egységeket (csoportokat, blokkokat, tömböket, osztályokat) képezni. Ilyenek például bizonyos csoporttechnológiai feladatok. Nyilvánvalóan ekkor olyan osztályozás érdekel bennünket, ahol az osztályok közötti részben-rendezés lineáris. Az osztályok elkészítésekor bizonyos esetekben arra is figyelniük kell, hogy azok arányos méretűek legyenek. Adott tehát egy véges elemszámú halmaz, amelyet diszjunkt részhalmazokra (osztályokra) kell felosztani úgy, hogy az osztályok között létrejövő részben-rendezés lineáris legyen, az osztályokon belül pedig egy ekvivalenciareláció érvényesüljön. Lényegében tehát egy véges elemszámú részben-rendezett halmazt osztályozunk úgy, hogy az eredeti részben-rendezés a részhalmazok között egy újabb részben-rendezést indukáljon. Az ilyen felbontások alapján vezettük be a rendezés-kongruencia fogalmát és vizsgáltuk tulajdonságait az [7] dolgozatban.

2. EGY ÜTEMEZÉSI FELADAT

A termelési illetve gyártási folyamatok esetén az elvégzendő munkák egy véges $M = \{m_1, m_2, \dots, m_n\}$ halmazán az $m_i \leq m_j$ jelölést vezetjük be, ha az i -edik munka elvégzése időben megelőzi a j -edik munkát ($i, j \in \{1, 2, \dots, n\}$). Nyilvánvaló, hogy ekkor (M, \leq) részben-rendezett halmaz. Ha M ütemezését klasszikus értelemben keressük, akkor az nem más, mint \leq egy lineáris kiterjesztése. Több megoldást is kaphatunk, hiszen egy részben-rendezésnek több lineáris kiterjesztése is lehet. Tegyük fel, hogy egymással párhuzamosan több munka elvégzésére is lehetőségünk van. Ez azt jelenti, hogy M elemeit célszerű diszjunkt részhalmazokra felosztanunk. Alkalmazzuk a továbbiakban az $\{M_1, M_2, \dots, M_t\}, (t \leq n)$ jelölést egy ilyen osztályozásra. Ekkor $M = M_1 \cup M_2 \cup \dots \cup M_t$. Lényegében lehetséges gyártási fázisokat hozunk létre,

* egyetemi adjunktus, Miskolci Egyetem, Analízis Tanszék

amely jelenthet időbeli dekompozíciót vagy logikai dekompozíciót. A részhalmazok kialakításakor nemcsak azt tartjuk szem előtt, hogy az egyes részhalmazokba egymással szabadon felcserélhető elemek kerüljenek, hanem azt is, hogy a létrehozott tömbök egymással összehasonlíthatóak legyenek, amely lényegét tekintve egy olyan részben-rendezés, amely az eredeti \leq részben-rendezés által indukált lineáris rendezések egyike. Az eddig megtett lépésekhez könnyen találunk matematikai sémát. Mivel $\{M_1, M_2, \dots, M_t\}, (t \leq n)$ osztályozása M -nek, így tartozik hozzá egy ρ ekvivalenciareláció. Ennek megfelelően

$$M/\rho = \{M_1, M_2, \dots, M_t\}, \quad (1)$$

azaz a vizsgált partíció a ρ ekvivalenciareláció faktorhalmaza. Tekintsük továbbá azt a

$$\varphi: M \rightarrow M/\rho \quad (2)$$

függvényt, amely az m_i ($i \in \{1, 2, \dots, n\}$) munkához hozzárendeli azt az M_j ($j \in \{1, 2, \dots, t\}$) fázist, amelyben a munka elvégzésre kerül, azaz

$$\varphi(m_i) := M_j, \quad (3)$$

ha $m_i \in M_j$. A rendezés-kongruenciának a [7] dolgozatban megadott fogalmát esetünkben az alábbi megfogalmazásban célszerű használni.

2.1. Definíció: A ρ ekvivalencia relációt rendezés-kongruenciának nevezzük az M halmazon, ha az M/ρ faktorhalmazon értelmezhető olyan \leq_ρ részben-rendezés, amelyre nézve a φ függvény rendezésőrző, azaz bármely $m_i \leq m_j$ ($m_i, m_j \in M$) esetén

$$\varphi(m_i) \leq \varphi(m_j) \quad (4)$$

teljesül.

A fenti definícióban alkalmazott \leq_ρ jelölésre az indukált részben-rendezés elnevezést használjuk. A 2.1. Definícióból azonnal adódik, hogy ha ρ rendezés-kongruenciája az (M, \leq) részben-rendezett halmaznak, akkor ρ egybeesik a φ izoton függvény magjával. Nyilvánvaló az is, hogy (4) akkor és csak akkor teljesül, ha van olyan $m_i, m_j \in M$, amelyre $m_i \leq m_j$ ($i, j \in \{1, 2, \dots, n\}$).

Az (M, \leq) részben-rendezett halmaz rendezés-kongruenciáinak halmazát jelölje $\vartheta(M)$. Az $(\vartheta(M), \subseteq)$ háló relatív komplementumos, atomisztikus és duálisan atomisztikus, továbbá teljesíti a Jordan-Hölder láncfeltételt [7]. Az ütemezési feladat megoldásához olyan $\rho \in \vartheta(M)$ kongruenciára van szükségünk, amelyre a kialakított részhalmazok között létrejövő \leq_ρ indukált részben-rendezés lineáris, tehát az $(M/\rho, \leq_\rho)$ faktorhalmaz lánc.

2.2. Definíció: Egy $\rho \in \vartheta(M)$ rendezés-kongruenciát az (M, \leq) lineáris rendezés-kongruenciájának nevezzük, ha az $(M/\rho, \leq_\rho)$ faktorhalmaz lánc.

Az ütemezési feladat megoldása tehát az (M, \leq) részben-rendezett halmaz egy lineáris rendezés-kongruenciája lesz. Meg kell azonban jegyeznünk, hogy a megoldást adó lineáris rendezés-kongruenciának még egy további tulajdonsággal is rendelkeznie kell. Ugyanis az (M, \leq) részben-rendezett halmaz lineáris rendezés-kongruenciái közül csak olyanra van szükségünk, amely tovább már nem finomítható $\{M_1, M_2, \dots, M_t\}, (t \leq n)$ partíciót eredményez.

2.3. Definíció: Egy $\rho \in \vartheta(M)$ rendezés-kongruenciát az (M, \leq) minimális lineáris rendezés-kongruenciájának nevezzük, ha (M, \leq) -nek nincs olyan $\theta \neq \rho$ lineáris rendezés-kongruenciája, amelyre $\theta \subseteq \rho$.

Az ütemezési feladat megoldásához célunk az (M, \leq) részben-rendezett halmaz minimális lineáris rendezés-kongruenciáinak meghatározása. A [7] dolgozatban igazolásra került, hogy ha a ρ ekvivalencia-reláció az (M, R) részben-rendezett halmaz intervallum-kongruenciája, akkor ρ rendezés-kongruenciája is (M, R) -nek. Ha R a \leq részben-rendezés lineáris kiterjesztése, akkor könnyen igazolható, hogy a ρ reláció (M, \leq) -nek rendezés-kongruenciája, azaz (M, \leq) rendezés-kongruenciáinak keresése visszavezethető az eredeti \leq részben-rendezés valamely lineáris kiterjesztésének intervallumokra történő feldarabolására. Az ütemezési feladat klasszikus megoldásához tehát az első lépést az (M, \leq) részben-rendezett halmaz esetén a \leq reláció lineáris kiterjesztéseinek meghatározása jelentheti. Erre az irodalomban több kész megoldás is olvasható. Ezek egyike Trotter-től származik [11].

3. A TROTTER-FÉLE MOHÓ ALGORITMUS

Megoldásunk első lépésével kapcsolatos legfőbb probléma az, hogy egy részben-rendezett halmaz lineáris kiterjesztéseinek a száma az alaphalmaz elemszámának függvényében exponenciálisan nő. Ennek a problémának a leküzdésére használjuk a Trotter-féle mohó algoritmust [11].

Jelöljük n -nel az X halmaz elemszámát. A mohó algoritmus az (X, \leq) részben-rendezett halmaz esetén a \leq reláció lineáris kiterjesztését úgy állítja elő, hogy először X minimális elemei közül választ egyet. Ha a lánc első x_1, \dots, x_i tagja már megvan, akkor az $(i + 1)$ -edik elemet az

$$(X \setminus \{x_1, \dots, x_i\}, \leq) \quad (5)$$

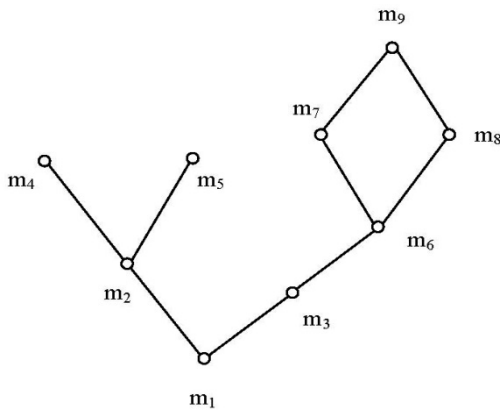
részben-rendezett halmaz minimális elemeinek halmazából választja a mohó feltétel szerint. Ez azt jelenti, hogy $i > 0$ esetén nem tetszőlegesen választ a minimális elemek közül, hanem azokra a minimális elemekre szűkít, amelyek az előző lépésben kiválasztott elemmel összehasonlíthatóak, amennyiben van ilyen elem.

3.1. Algoritmus:

1. **Proc** Trotter
2. **Input:** X halmaz
3. P halmaz (részben-rendezés)
4. **Output:** L vektor (lánc)
5. $X_0 \leftarrow X$
6. $P_0 \leftarrow P$
7. $\mathbf{P}_0 \leftarrow (X_0, P_0)$
8. $G_0 \leftarrow (S_0 \leftarrow \min(\mathbf{P}_0)$ tetszőleges eleme)
9. **for** $i \leftarrow 0$ **to** $n - 1$ **do**
10. **if** $i > 0$ **then**
 $S'_i = \{x \in S_i | x_i < x \text{ } P \text{ - ben}\}$
11. **if** $S'_i = \emptyset$ **then** $G_i = S_i$
12. **if** $S'_i \neq \emptyset$ **then** $G_i = S'_i$
13. $x_{i+1} \leftarrow G_i$ egy tetszőleges eleme
14. $L[i + 1] \leftarrow x_{i+1}$
15. $X_{i+1} \leftarrow X_i \setminus \{x_{i+1}\}$
16. $P_{i+1} \leftarrow P(X_{i+1})$
17. $\mathbf{P}_{i+1} \leftarrow (X_{i+1}, P_{i+1})$
18. $S_{i+1} \leftarrow \min(\mathbf{P}_{i+1})$
19. **RETURN**

Az algoritmus működésének szemléltetésére tekintsük az alábbi példát.

3.2. Példa: Az (M, \leq) részben-rendezett halmazt Hasse-diagramjával szemléltetjük.



1. ábra

Ekkor például $L_1: [m_1, m_3, m_6, m_7, m_2, m_4, m_5, m_8, m_9]$ és $L_2: [m_1, m_2, m_4, m_5, m_3, m_6, m_7, m_8, m_9]$ olyan lineáris kiterjesztései \leq -nek, amelyeket a 3.1. Algoritmus eredményezhet.

4. AZ ÜTEMEZÉSI FELADAT MEGOLDÁSA

Jól látható, hogy klasszikus értelemben, azaz amikor egy időben csak egy gép dolgozik (tehát nincs lehetőség párhuzamos munkák végzésére) az ütemezési feladatnak a Trotter-féle mohó algoritmus eredményeként kapott lineáris kiterjesztés minden esetben egy megoldását adja. A minimális lineáris rendezés-kongruenciák

meghatározásával azt az alternatív ütemezési feladatot fogjuk megoldani, amikor megengedjük, hogy egy időben több gép is dolgozzon. Lényegében logikai dekompozíciót végzünk, amelynek eredményeként azt várjuk, hogy a gyártási (futási) idő csökkenjen. Az ütemezési feladatot reprezentáló (M, \leq) részben-rendezett halmaz esetén \leq lineáris kiterjesztését a Trotter-féle mohó algoritmus szolgáltatja. Ha tekintjük az $L: x_1, \dots, x_n$ ($x_i = m_j$, $i, j \in \{1, 2, \dots, n\}$) láncot (ahol L a \leq lineáris kiterjesztése), akkor elő kell állítanunk az $M_1 = [x_1, x_{i^1}]$, $M_2 = [x_{i^1+1}, x_{i^2}]$, \dots , $M_t = [x_{i^{t-1}}, x_n]$ intervallumokat. Az így kialakított partíció az M halmaz rendezés-kongruenciája. Lineáris rendezés-kongruenciát a 2.2. Definíció értelmében akkor kapunk, ha az

$$\{M_1, M_2, \dots, M_t\}, \quad (t \leq n) \quad (6)$$

halmazon indukált részben-rendezés lineáris. Ez akkor következik be, ha bármely két egymást követő intervallum esetén léteznek olyan $x_k \in M_j$ és $x_l \in M_{j+1}$ ($1 \leq k < l \leq n$) elemek, amelyekre $x_k \leq x_l$ teljesül. Ahhoz, hogy minimális lineáris rendezés-kongruenciát állítsunk elő elégséges, ha a kialakított M_1, M_2, \dots, M_t intervallumok (M, \leq) -ben antiláncok és bármely két szomszédos M_j, M_{j+1} ($j \in \{1, \dots, t-1\}$) intervallum esetén legyenek olyan $x_k \in M_j$ és $x_l \in M_{j+1}$ elemek ($1 \leq k < l \leq n$), amelyekre $x_k < x_l$, azaz x_k rákövetkezője az x_l elem, tehát nincs olyan $x_s \in M$ elem, amelyre $x_k < x_s < x_l$ teljesül. Ha a Trotter-féle mohó algoritmust kiegészítjük a fenti észrevételeinkkel, akkor az ütemezési feladat megoldását minimális lineáris rendezés-kongruencia formájában kapjuk meg. Jelöljük n -nel az L -vektor hosszát.

4.1. Algoritmus:

1. **Proc** Minimális lineáris rendezés-kongruencia
2. **Input:** X halmaz
3. P halmaz (részben-rendezés)
4. L vektor (Trotter-féle algoritmusból)
5. **Output:** \mathcal{A} (minimális lineáris kongruencia)
6. $X_1 \leftarrow X$
7. $P_1 \leftarrow P$
8. $\mathbf{P}_1 \leftarrow (X_1, P_1)$
9. $A_1 \leftarrow \{L[1]\}$
10. $\mathcal{A} \leftarrow A_1$
11. **for** $i \leftarrow 1$ **to** $n - 1$ **do**
12. **if** $(L[i], L[i + 1]) \in \mathbf{P}_i$
13. **then** $A_{i+1} \leftarrow \{L[i + 1]\}$
14. **if** $(L[i], L[i + 1]) \notin \mathbf{P}_i$
15. **then** $A_{i+1} \leftarrow A_i \cup \{L[i + 1]\}$
16. $\mathcal{A} \leftarrow \mathcal{A} \cup \{A_{i+1}\}$
17. **if** $A_i \subseteq A_{i+1}$ **and** $A_i \neq A_{i+1}$
18. **then** az \mathcal{A} halmazból $\{A_i\}$ törlése
19. $X_{i+1} \leftarrow X_i \setminus \{L[i]\}$
20. $P_{i+1} \leftarrow P(X_{i+1})$
21. $\mathbf{P}_{i+1} \leftarrow (X_{i+1}, P_{i+1})$
22. **RETURN**

A 4.1. Algoritmus az (X, \leq) részben-rendezett halmaz esetén egy ekvivalencia osztályaival megadott minimális lineáris rendezés-kongruenciát úgy állít elő, hogy a bemenetként kapott L lánc első eleméből kialakít egy osztályt. Most $L[i]$ jelöli az L lánc i -edik elemét. Ha $i \geq 1$, akkor az algoritmus megvizsgálja, hogy a lánc i -edik és $i + 1$ -edik eleme között fennáll-e a \leq reláció. Ha $L[i] \leq L[i + 1]$, akkor új osztályt hoz létre, amely az $L[i + 1]$ elemet tartalmazza. Ha $L[i] \not\leq L[i + 1]$, akkor az új osztályt úgy alakítja ki, hogy az előző lépésben létrehozott osztályt bővíti az $L[i + 1]$ elemmel. Ha az előző lépésben létrehozott osztály valódi részhalmaza az i -edik lépésben megalkotott halmaznak, akkor azt töröljük, hiszen annak elemeit az új osztály tartalmazza. Könnyen látható, hogy az egy osztályba kerülő elemek antiláncot képeznek az (X, \leq) részben-rendezett halmazban, továbbá bármely két egymás után kialakított (és nem törölt) osztály (intervallum) esetén találunk olyan elemeket, amelyekre a megkövetelt rákövetkezési tulajdonság teljesül.

Sikerült tehát a kitűzött ütemezési feladatunkhoz olyan megoldást találni, amelynek segítségével az egyes elvégzendő munkák olyan logikai dekompozíciója végezhető el, amely csökkenti a teljes munka elvégzéséhez szükséges időt, azáltal, hogy lehetővé válik bizonyos munkafázisok során a párhuzamos munkavégzés.

4.2. Példa: Ha a 3.2. Példában megadott L_1 lineáris kiterjesztést tekintjük, akkor a 4.1. Algoritmus az alábbi minimális lineáris rendezés-kongruenciát eredményezi:

$$\{\{m_1\}, \{m_3\}, \{m_6\}, \{m_7, m_2\}, \{m_4, m_5, m_8\}, \{m_9\}\} \quad (7)$$

Az L_2 lánc esetén pedig a

$$\{\{m_1\}, \{m_2\}, \{m_4, m_5, m_3\}, \{m_6\}, \{m_7, m_8\}, \{m_9\}\} \quad (8)$$

megoldást kapjuk. Látható, hogy az elvégzendő munkát hat fázisra bontottuk mindkét esetben és legfeljebb három gép dolgozhat párhuzamosan. Az is jól látható, hogy ebben az esetben a minimális lineáris rendezés-kongruencia alapján elvégzett logikai dekompozíció kilenc időegységről hat időegységre csökkenti az előállítási (futási) időt, amennyiben feltételezzük, hogy minden elvégzendő munka ugyanakkora időigénnyel bír.

Az is könnyen észrevehető, hogy ha a 4.1. Algoritmus bemenetként megadott láncot nem a Trotter-féle mohó algoritmus eredményezi, akkor a minimális lineáris rendezés-kongruenciára megadott feltételek általában nem teljesülnek a kialakított ekvivalencia-osztályokra, tehát nem szolgáltatnak optimális megoldást az ütemezési feladatra.

4.3. Példa: Tekintsük a 3.2. Példában az (M, \leq) részben-rendezett halmaz esetén az

$$L_3: [m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9] \quad (9)$$

lineáris kiterjesztést. Könnyen látható, hogy ezt a láncot a 3.1. Algoritmus nem állítja elő. Ekkor ugyanis $x_1 = m_1$ és $x_2 = m_2$, így $S_2 = \{m_4, m_5, m_3\}$ és $G_2 = \{m_4, m_5\}$. Emiatt x_3 -nak a mohó algoritmus nem választhatja m_3 -at, mert $m_3 \notin G_2$. Ha az L_3 lineáris kiterjesztés a 4.1. Algoritmus bemenete, akkor az alábbi osztályozást kapjuk:

$$\{\{m_1\}, \{m_2, m_3, m_4, m_5, m_6\}, \{m_7, m_8\}, \{m_9\}\}. \quad (10)$$

Ekkor azonban az $\{m_2, m_3, m_4, m_5, m_6\}$ osztály elemei nem képeznek antiláncot, mert

$$m_2 \leq m_4, \quad m_2 \leq m_5, \quad m_3 \leq m_6. \quad (11)$$

Az ütemezési feladatnak az így elkészített osztályozás nyilvánvalóan nem megoldása.

KÖSZÖNETNYÍLVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

IRODALOM

- [1] CHAJDA I., SNÁŠEL V. (1998) *Congruences in ordered sets*, Czech Math. Journal **123**, 95-100.
- [2] DAEVY B. A., PRIESTLEY H. A. (1990) *Introduction to Lattices and Order*, Cambridge University Press, Cambridge.
- [3] FISHBURN P.C. (1985) *Interval Orders and Interval Graphs*, Willey, New York.
- [4] FOLDES S., RADELECZKI S. (2004) *On interval decomposition lattices*, *Discussiones Mathematicae, General Algebra and Applications* **24**, 95-114.
- [5] HALAŠ R. (1995) *Congruences on pos ets*, *Contributions to General Algebra* **12**, Velad Johannes Heyn, Klagenfurt, 195-210.
- [6] KOLIBIAR M. (1987) *Congruence relations and direct decompositions of ordered sets*, *Acta Sci. Math. (Szeged)* **51**, 129-135.
- [7] KÖRTESEI P., RADELECZKI S., SZILAGYI Sz. (2005) *Congruences and isotone maps on partially ordered sets*, *Mathematica Pannonica* **16/1**, 39-55.
- [8] ORE O. (1943) *Chains in partially ordered sets*, *Bull. Amer. Math. Soc.* **49**, 558-566.
- [9] PINEDO M. (2002) *Scheduling (Theory, algorithms and systems)*, Prentice Hall.
- [10] SZPILRAJN E. (1930) *Sur l'extension de l'ordre partiel*, *Fund. Math.* **16**, 386-389.
- [11] TROTTER W. T. (1992) *Combinatorics and Partially ordered sets, Dimension Theory*, The Johns Hopkins University Press, Baltimore-London.

PARANCSKINYERÉS MAGYAR NYELVŰ SZÖVEGBŐL

FUNCTION EXTRACTION FROM HUNGARIAN TEXT

Barabás Péter*

ABSTRACT

In human-computer interaction the natural language processing is an important area. Communicating with computer generally is quite difficult in natural language, thus processing sentences will be limited to a specific domain to be able to build knowledge base up and to evaluate questions and commands in tolerable time. The goal of our research is to develop a natural language interface framework with help of which applications and systems can be controlled with Hungarian language.

1. BEVEZETÉS

A számítógépek megjelenése magával hozta bizonyos mesterséges nyelvek kialakulását, amelyek segítségével kommunikálni tudunk a gépi rendszerekkel. Ezen nyelvek tipikus példái a programozási nyelvek. Az emberek a számítógépek megjelenése óta szeretnék megvalósítani, hogy a gépekkel az egymás között megszokott, természetes nyelven „beszélgessenek”. Ezen interakciónak a gyakorlati megvalósulását a természetes nyelvi felületek jelentik. A természetes nyelvi interfésszel (NLI) rendelkező információs rendszerek gyökerei 1970-re nyúlnak vissza. Az úttörő LUNAR [1] projekt a holdkőzetek adatbázisában való lekérdezésekhez dolgozott ki természetes nyelvű interfész felületet. A RENDEZVOUS (Codd, 1977) rendszer volt az első általános célú adatbázis NLI modul. Az NLI modulok egyik alapfeladata a természetes nyelven beérkező parancsok átkonvertálása a feldolgozó modul saját parancsnyelvére. Ezen konverzió megvalósítása több lépcsőben történik kezdve a természetes nyelvi mondat szintaktikai ellenőrzésével, elemzésével és folytatva a szemantikai analízissel, a tématérület ismert fogalmainak detektálásával.

Több ismert NLP keretrendszer is található a piacon, ezek közül a legismertebbek az Apache nyílt forráskódú OpenNLP [3] és UIMA [4] rendszerei, illetve a Stanford Egyetem statisztikai alapú Stanford NLP [5] rendszere. Az előzőekben felsorolt

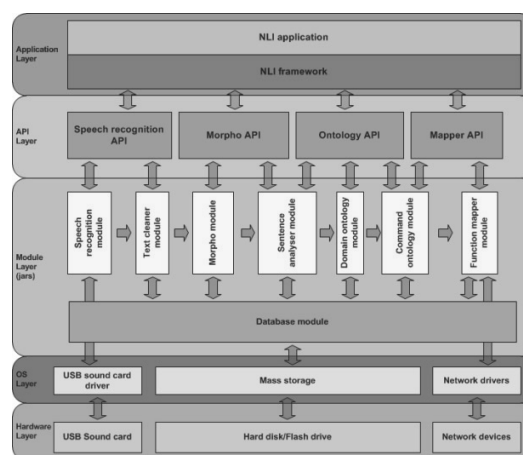
rendszerekben egyaránt megtalálhatóak a legfontosabb szöveg feldolgozási modulok, mint a mondat- és szódetektáló, tulajdonnév azonosító, szótövező, mondat osztályozó, stb. A legtöbb létező keretrendszer alapvetően az angol nyelvet, illetve még esetleg néhány „könnyebben” feldolgozható nyelvet támogat. Esetenként előfordul, mint az UIMA esetén is, hogy a keretrendszer adaptálható különböző nyelvekre.

A magyar nyelv egy nehezen feldolgozható nyelv, amely az agglutináló tulajdonsága révén szavak és toldalékok nagyszámú kombinációjával rendelkezik, melyek között sok szabályszerűség és kivételes ragozás fedezhető fel. Magyar nyelvű kutatásokban [6][7] is találkozhatunk az előzőekben említett rendszerek használatával.

A kutatásunk célja, hogy az előzőekben említett NLP keretrendszerek egyes funkcióit magyar nyelvű szöveg feldolgozására implementáljuk és kiterjesszük a parancsnyelvi funkciókra történő konvertálással és azok implementációjának dinamikus meghívásával.

2. NLP RENDSZER STRUKTÚRÁJA

A szöveges input elemzése egy meglehetősen bonyolult és összetett folyamat, mely al folyamatok, almodulok láncolataként kezelve oldható meg



1. ábra. NLP rendszer logikai struktúrája

*egyetemi tanársegéd, Miskolci Egyetem, ÁIT

hatékonyan. Az 1. ábrán látható, hogy az NLI rendszer a következő almodulokra bontható:

- Bemeneti eszközhöz kapcsolódó modulok
 - Beszédfelismerő modul
 - Szövegtisztító, átalakító modul
- Morfológiai modul
- Fogalmak feldolgozásához kapcsolódó modulok
 - Domain ontológiai modul
 - Mondatelemző modul
- Parancsfeldolgozás moduljai
 - Parancsontológiai modul
 - Funkciókinyerő modul

A modulok két részre oszthatók a nyelvfüggőség tekintetében:

- Nyelvfüggő modulok: HID modulok és a morfológiai modul
- Nyelvfüggetlen v. kvázi-nyelvfüggetlen modulok: ontológiai modulok (domain, parancs)

2.1. Szövegtisztító modul

A szövegtisztító modulnak képesnek kell lennie a bemeneti szöveg hibáinak detektálására, javítására. Mivel a bemenet származhat többféle forrásból is (beszéd, billentyűzet, lapolvasó, kézírás), az analóg-digitális konverterek a különböző források esetén más és más hibákat ejthetnek. A billentyűzetről származó szövegben az elütések okozhatnak hibákat, melyek javításánál a billentyűk elhelyezkedését tudjuk figyelembe venni, míg egy scannerből származó forrásban a hasonló formájú karakterek lehetnek hibásan felismerve. Ezen hibák javítására különböző algoritmusok használhatóak a hibaforrás függvényében.

Ahhoz azonban, hogy a további modulok számára használható bemenetet biztosítsunk, a modul funkciói közé tartoznak a következők:

- Szótárkezelés
- Mondathatárolók és szóhatárolók kezelése
- Mondatok detektálása
- Szavak detektálása
- Szavak helyesírás elemzése, javítása

2.2. Morfológiai modul

A morfológiai modul szerepe a mondat(ok) szavainak nyelvtani elemzése. A további modulok feladata a mondatok jelentésének meghatározása, melyhet elengedhetetlen a szavak morfológiai analízise, ugyanis ugyanaz a szótó más-más toldalékkal teljesen más jelentést hordozhat. A morfológiai analízisen túl

szükség lehet egy tetszőleges szó adott szabály szerinti ragozott alakjának meghatározására is. Ez utóbbi funkció a válaszgenerálásnál kaphat jelentősebb szerepet.

A morfológiai modulnak a következő funkcionalitással kell rendelkeznie:

- Szótövek tárolása, kezelése
- Nyelvtani szabályok tárolása, kezelése
- Kivételesen ragozandó szavak kezelése
- Szavak morfológiai analízise
- Szavak ragozása
- Analízis pontosságának meghatározása adott teszt szóhalmazra

2.3. Domain ontológiai modul

A domain ontológiai modul kezeli a jelentését a szövegnek felhasználva az előző modulok eredményeit. A jelentés meghatározásának alapegységei a fogalmak, melyek között igen változatos kapcsolatrendszer építhető ki. Egy adott domain fogalmait, fogalmi kapcsolódásait lehet ezen modul funkcióival felépíteni.

Az egyes fogalmak írásbeli, illetve beszédbeli megvalósulásai a szavak, melyek nyelvenként eltérőek, azonban a jelentésükről elmondható, hogy többekévesb nyelvfüggetlenek. Például amikor meghalljuk azt a szót, hogy telefon, mindenki egy készülékre gondol, amellyen keresztül távol lévő emberek tudnak egymással beszélgetni, függetlenül attól, hogy ezt egy magyar ember teszi, aki "telefon"-nak nevezi a készüléket vagy egy angol ember teszi, aki "phone"-nak vagy "telephone"-nak.

A domain ontológiai modul feladatai közé tartoznak a következők:

- Fogalmak kezelése
- Fogalmak detektálása a mondatból
- Mondat fogalmi kapcsolatainak meghatározása

2.4. Mondatelemző modul

A mondatelemző modul a morfológiai modul kimenetét használja ahhoz, hogy felépítse a bemenet mondatnyi elemzését. Ahhoz, hogy a mondat jelentését megértsük, szükség van arra, hogy az egyes szavakról, szókapcsolatokról meg tudjuk mondani, hogy milyen mondatrészt töltenek be a szövegben. Itt kerülnek meghatározásra az egyes mondatrészek: alany, állítmány, tárgy, stb., és a mondatok közötti viszonyok: alá-, mellérendeltség. A mondatelemzés nem végezhető el tisztán szintaktikai úton ismerve az egyes szavak morfológiai analízisét, ugyanis nagyon sok esetben a

predikátumból, illetve annak jelentéséből tudjuk, hogy egy adott szó milyen mondatrészi szereppel bír a mondatban.

A mondatelemző modulnak a következő funkciókat kell megvalósítania:

- Mondatelemző tanítása
- Mellékmondatok meghatározása, detektálása
- Mondatrészek meghatározása

2.5. Parancsontológiai modul

A parancsontológiai modul az NLI keretrendszer utolsó modulja, melynek feladata a kiadott parancs, utasítás végrehajtása. Egy domain-orientált alkalmazásban véges számú utasítást tudunk kezelni. A parancsok leírásának általános, absztrakt módon kell történnie, hogy tetszőleges alkalmazást, eszközt ki lehessen terjeszteni a természetes nyelvi irányítással.

A végrehajtható funkciók közös tulajdonsága, hogy van neve, amely azonosítja őket, illetve nulla vagy több paraméterrel rendelkeznek. A feldolgozás során a természetes nyelvi mondatokban detektált fogalmakat kell tudni megfeleltetni a parancsleírásban szereplő elemeknek, majd a kiválasztott („nyertes”) funkciót a detektált paraméterekkel futtatni és az eredményt megjeleníteni.

A parancsontológiai modul fő feladatait a következőképpen foglalhatjuk össze:

- Funkciók kezelése
- Funkcióosztályozó kezelése, tanítása
- Funkciók, paraméterek fogalmakhoz társítása
- Kérdésgenerálás hiányzó paraméterekhez
- Parancs végrehajtás
- Eredménygenerálás

A keretrendszer legfontosabb feladata a mondatokban lévő fogalmak detektálása és azok társítása a funkciókhoz. Erről a következő fejezetben lesz szó részletesen.

3. FOGALOMKINYERÉS TERMÉSZETES SZÖVEGBŐL

A szövegtisztító és morfológiai modulok pontos működése nagymértékben hozzájárul a szövegfeldolgozás sikerességéhez. Ezen modulok a szintaktikai feldolgozást végzik, éppen ezért elkülönülnek a további moduloktól, amelyek feladata a szöveg „jelentéstartalmának” elemzése, kezelése.

Ahhoz, hogy a mondatok fogalmait meghatározhassuk, szükségünk van a domaint alkotó fogalomháló előzetes, tudásbázisként történő

definiálására. A fogalmak leírására használhatunk létező nyelveket, mint az OWL [8] vagy RDF [9], de megtehetjük ezt saját leíró struktúrák definiálásával egyaránt. A fogalmak között a következő típusú kapcsolatok definiálhatók:

- Leszármazás kapcsolat: a gyerek elem a szülőnek egy leszármazottja, egy specifikus példánya.
- Predikátum kapcsolat: a szülő elem egy predikátum, a mondatban az állítmány szerepét tölti be, a gyerek elem pedig egy másik, nem állítmány mondatrészhez tartozó fogalom
- Tulajdonság kapcsolat: a szülő egy tetszőleges fogalom, a gyerek elem pedig az ahhoz kapcsolódó tulajdonságot, többnyire jelzőt meghatározó fogalom

Egy fogalomnak több, különböző írásbeli reprezentációja is lehetséges. Ezeket nevezhetjük szinonimáknak is, azonban ezek hatásköre a domainre korlátozódik. Más szövegekörnyezetben már teljesen más jelentést hordozhat ugyanaz a szó. Ezért is fontos, hogy pontosan meghatározzuk, hogy egy adott szó melyik domain fogalmához is tartozik.

A fogalom kinyerés során egy a bejövő mondatból rendelt fogalomhalmazt kapunk, amelyben alapvetően nincs sorrendiség. Ezzel a magyar nyelv szabad szórendje is egyszerűen kezelhetővé válik. Természetesen a kapcsolatoknál, tipikusan a tulajdonság kapcsolatoknál fontos lehet a megelőzési sorrend, amely a szavak mondatbeli indexéből levezethető.

A funkciótársításhoz a detektált fogalmak mondatbeli szerepét is ismernünk kell, ugyanis egy szó különböző szövegekörnyezetben különböző jelentéssel, illetve mondatbeli szereppel bírhat. A mondatelemzést a következő lépésekben végezhetjük:

1. A predikátum meghatározása. Amennyiben nincs, a „van” létige használata.
2. A predikátum fogalom predikátum kapcsolatainak lekérdezése.
3. A predikátumhoz kapcsolható fogalmak illesztése a mondatban detektált fogalmakhoz.
 - a. A fogalom illesztése.
 - b. A toldalékok illesztése.
4. A detektált mondatrészek visszaadása a további modulok által feldolgozható formában (*XML*).

A mondatelemzés eredményként tartalmazza a bejövő mondat fogalmait és azok mondatbeli szerepét. Ezeket az információkat felhasználva már lehetőségessé válik a mondat a megfelelően definiált funkciókhoz való társítása.

4. FUNKCIÓTÁRSÍTÁS FOGALMAKHOZ

Az előzőekben kinyert információt felhasználva utolsó lépésben a megfelelő funkciót kell kiválasztanunk. A funkciók leírására a következő jelölésrendszert vezethetjük be:

$$F = (Pred, Par, Im),$$

ahol

$Pred = \{pred_i\}, i = 1, \dots, N$: a predikátumok,

$Par = \{par_j\}, j = 0, \dots, M$: a paraméterek,

$Im = \{Class, Method\}$: az implementáció

Egy funkcióhoz több predikátum fogalom is társítható, legalább egy azonban kötelező. A mondatban detektált állítmány fogalmát kell a funkció kiválasztásánál az egyes funkciók predikátum fogalmaihoz társítani.

A predikátum alapján illeszkedő funkciók implementációjának meghívásához a paraméterek megfelelő társítása elengedhetetlen. A paramétereket a következőképpen írhatjuk le:

$$par_i = (n, r, t, C, PS),$$

ahol

n : a paraméter neve,

$r = \{true, false\}$: a paraméter kötelezősége,

$t = \{string, int, float, bool\}$: a paraméter típusa,

$C = \{c_i\}, i = 1, \dots, N$: a kapcsolódó fogalmak,

$PS = \{ps_j\}, j = 1, \dots, M$: a mondatbeli szerepek

A mondat még fel nem dolgozott fogalmait illeszteniük kell az egyes paraméterek fogalomlistájára. Egy funkció esetén több paraméterhez is köthetjük ugyanazt a fogalmat, így a pontosítás végett az illesztésnél figyelembe kell vennünk a mondatbeli szerepet (PS) is.

Előfordulhat olyan eset is a természetes mondatban, hogy több ugyanazon fogalommal és mondatbeli szereppel bíró mondatrészünk van, pl. több jelző a mondatban különböző szavakhoz. Ezen fogalmak paraméterekhez társítása egy megelőzési, vagy függőségi leíró bevezetését kívánja meg a következő formában:

$$ps_i = \{psn, D\},$$

ahol

psn : a mondatbeli szerep neve,

$D = \{d_k\}, k = 0, \dots, N$: a rákövetkező mondatrészek

Így egyező mondatrészek esetén a kapcsolódó, azaz rákövetkező fogalom mondarésze dönti el, hogy melyik paraméterhez kell a fogalmat társítani.

A paraméterek társítása után azon funkció kerül kiválasztásra, amelynél egyrészt predikátumilleszkedés

van, másrészt a legtöbb „kötelező” paraméter lefedhető a mondatban kinyert fogalmakkal. Amennyiben valamely „kötelező” paraméterhez nem társítható fogalom a mondatból, a rendszer visszakérdezéssel próbálja kikényszeríteni a hiányzó paraméter megadását. Amikor a funkció kiértékelhetővé válik, a hozzá tartozó implementáció (Im) kerül meghívásra elérve ez által a kitűzött célt.

5. EREDMÉNYEK

A kutatásaink során sikerült három mintaalkalmazást készíteni eltérő domainekre (navigációs alkalmazás, árfolyam-lekérdező alkalmazás, robotvezérlés), igazolva ezzel a módszer működőképességét. A megvalósított rendszer még összetett mondatokat nem tud kezelni, ez a következő kutatásokban kerül kidolgozásra és megvalósításra.

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-0001-2010 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

6. IRODALOM

- [1] WOODS, W., KAPLAN, R.: Lunar rocks in natural English: Explorations in natural language question answering, Linguistic Structures Processing. In Fundamental Studies in Computer Science, 5, pp. 521-569, 1977.
- [2] E.F. CODD.: Seven steps to rendezvous with the casual user. In IFIP Working Conference Data Base Management, 179–200, 1974.
- [3] <http://opennlp.apache.org>
- [4] <http://uima.apache.org>
- [5] <http://nlp.stanford.edu/software/index.shtml>
- [6] Zsibrita, János; Nagy, István; Farkas, Richárd 2009: Magyar nyelvi elemző modulok az UIMA keretrendszerhez. In: Tanács Attila, Szauter Dóra, Vincze Veronika (eds.): VI. Magyar Számítógépes Nyelvészeti Konferencia. Szeged, Szegedi Tudományegyetem, pp. 394-395.
- [7] Zsibrita, János; Vincze, Veronika; Farkas, Richárd 2010: Ismeretlen kifejezések és a szófaji egyértelműsítés. In: Tanács Attila, Vincze Veronika (eds.): VII. Magyar Számítógépes Nyelvészeti Konferencia. Szeged, Szegedi Tudományegyetem, pp. 275-283.
- [8] <http://www.w3.org/TR/owl-features/>
- [9] <http://www.w3.org/RDF/>

KÖZÖS FEJLESZTŐI KERETRENDSZER FUZZY SZABÁLY INTERPOLÁCIÓS MÓDSZEREKHEZ

COMMON FUZZY RULE INTERPOLATION FRAMEWORK LIBRARY FOR DEVELOPERS

*Krizsán Zoltán**

ABSTRACT

In case of applications running in real environments one often applies a sparse rule base. although there is not any developers' library supporting such applications. Behind the scenes, the models of such real applications use Fuzzy Rule Interpolation methods to determine the consequents taking into consideration some known rules. Johanyák implemented a new common framework, which includes several FRI methods. However, the disadvantage of his solution is the cumbersome usage owing its demand on the availability of a Matlab system. This article introduces an easy-to-extend and quick framework that support the development of fuzzy systems and which can be used from any of the popular programming languages (e.g. C, C++, Java, C#).

1. BEVEZETÉS

A fuzzy rendszerek a következtetéseket a szabály bázis alapján állítják elő. A klasszikus fuzzy rendszerek megkövetelik a bemenetek teljes mértékű lefedését szabályokkal. Ez azt jelenti, hogy minden megfigyelésre biztosan létezik legalább egy szabály antecedens, így egy érvényes következtetés számolható. A teljes lefedettség nagy számú szabályt igényel. A gyakorlatban ezért gyakori a ritka szabálybázisú rendszer, mert vagy nem áll még rendelkezésre elegendő információ, vagy nem lenne hatékony a teljesen fedő rendszer.

Jelen pillanatban még nincs olyan programozói keretrendszer, melynek segítségével a ritka szabálybázisú rendszerekben a fuzzy interpolációs módszerek használhatóak. Ennek a hiánynak köszönhető, hogy a ritka szabálybázisú interpolációs módszereket nem használják a gyakorlatban.

Ha a fejlesztendő rendszer ritka szabálybázist igényel, akkor a programozónak implementálnia kell a kiválasztott módszert a használni kívánt programozói nyelven. Ha egy másik alkalommal ismét ugyanazt a módszert szeretné használni, de egy másik programozói nyelven, akkor újfent implementálnia kell azt azon a

másik nyelven is. Minden egyes módszert minden kívánt nyelven implementálni és tesztelni kell. Ezek ráfordítást, erőforrásokat igényelnek.

Ha a fejlesztő nem rendelkezik a fuzzy rendszerek területén ismerettel, tapasztalattal, akkor egy MATLAB szoftver és az FRI MATLAB Toolbox szükséges, hogy kipróbálja és összevesse a rendelkezésre álló módszereket. Ha a projekt nem követeli meg a MATLAB rendszer meglétét, akkor ez felesleges költséget, és a MATLAB szoftver rendszer felhasználói szintű ismeretét igényli.

Azon rendszerek esetén, amelyek gyors beavatkozást, reakciót igényelnek alacsony szintű implementációk szükségesek. Ez azt jelenti, hogy az algoritmusokat abban a rendszerben kell implementálni (lehetőleg C++, vagy C nyelven), ahol használják. Minden közbenső réteg, beágyazott környezet drasztikus teljesítmény veszteséggel járhat. A már létező FRI MATLAB Toolbox a MATLAB speciális környezetére írt, annak speciális program nyelvéen készült. Használata vagy MATLAB alatt, vagy beágyazva egy C++ alkalmazásba lehetséges. A beágyazás esetén az alkalmazásunkhoz kell csatolni a MATLAB natív könyvtárait, és az FRI MATLAB Toolbox instrukciói egy interpretált környezetben futnak. Az eredmények kiértékelése nem elég gyors a beágyazott futtatás esetén.

Elsődleges célunk egy olyan modul vagy komponens kifejlesztése, amely ritka szabálybázisú alkalmazások esetén is ad eredményt, könnyen használható, több nyelvet is támogat (lehetőleg azonos felülettel), MATLAB és ingyenes matematikai szoftverekben is használható, elegendően gyors a valós idejű szabályozásokhoz, vezérlésekhez.

Első lépésként Windows és Linux operációs rendszereket támogatjuk, törekedünk a manapság népszerű programozó nyelvek egységes támogatására (C, C++, Java, C#, Python).

A cikk első részében megvizsgáljuk milyen módszerek állnak rendelkezésre, melyeket kellene támogatnia a rendszerünknek, majd megvizsgáljuk, hogy milyen feltételeket kell figyelembe vennie. Ezután a rendszerünk felépítését, kiterjesztésének lehetőségét ismertetjük, végül a keretrendszer használatának módját.

* egyetemi tanársegéd, Miskolci Egyetem, ALT

2. FRI MÓDSZEREK, KÖZÖS JELLEMZŐI

A ritka szabálybázisú fuzzy rendszerekben történő következtetésre kifejlesztett módszereket két csoportba sorolhatjuk.

Az egy lépéses vagy direkt módszerek a következményt egy lépésben állítják elő. Ezen csoport meghatározó képviselői: a KH [1] melyet Kóczy és Hirota vezetett be, a MACI [2] (Tikk and Baranyi), a FIVE [3] (Kovács és Kóczy), az IMUL [4] (Wong, Gedeon, and Tikk), és VKK [5] (Vass, Kalmár and Kóczy) módszerek.

Találhatóak olyan módszerek is melyek nem egy, hanem két lépésben határozzák meg a következményt. Első lépésben egy ideiglenes szabályt hoz létre a megfigyelés helyén, majd második lépésben ezen ideiglenes szabály alapján meghatározza a következményt a megfelelő helyen megfelelő alakokkal. Ezen eljárások lépéseit Baranyi és társai definiálták [6]. Ennek a családnak a módszerei az ST [7] (Yan, Mizumoto, and Qiao), az IGRV [8] Huang and Shen, , és a Jenei által javasolt [9] módszerek.

Ezen módszereket egyesítette Johanyák kifejlesztve egy MATLAB Fuzzy Toolbox-ot (FRI TB) [10]. Ez a MATLAB keretrendszer Matlab metódusok gyűjteménye. A keretrendszer letölthető a [11] helyről, és használható a „GNU General Public License” licenz szerint. MATLAB R14 alatt készült, így használható Windows és Linux rendszerek alatt is. Az FRI TB fájlt használ a fuzzy rendszer definiálására ugyanúgy, mint a Fuzzy Toolbox is, azonban a magában foglalt módszereknek köszönhetően tudja kezelni a hiányos szabálybázisú rendszereket is. Ezen keretrendszer segítségével olyan felhasználók is tudják használni az FRI módszereket, akik nem jártasak ezen a területen. A keretrendszer grafikus felületén megadhatóak a paraméterek, majd az eredményeket a MATLAB képes megjeleníteni. Ha a felhasználói rendszer futtat a MATLAB alól, akkor ez a keretrendszer ideális az eredmények előállítására.

Saját keretrendszerük kifejlesztésének alapjait ezen rendszer adja, azonban a mi implementációnkat mások kódjából fogják használni (nem grafikuson), így meg kell terveznünk egy egységes szerkezetet. A módszerek egységes kezeléséhez, helyes működése érdekében meg kell vizsgálnunk azokat a közös vonásokat, feltételeket, melyeket minden FRI módszernek ki kell elégítenie.

Tikk és szerzőtársai [12] általános feltételeket definiáltak a fuzzy interpolációs módszerekkel szemben. Ezek azonban csak kívánalmak az egységesítés érdekében, nem kötelező jellegűek.

3. FRI PROGRAMOZÓI KERETRENDSZER

A keretrendszerrel szemben támasztott főbb követelmények a következők:

- *gyorsaság*, azaz a valós időben működő szabályzó, vezérlő alkalmazások válasziideje kellően kicsi lehet csak,
- *széleskörű támogatottság*: minél több projektben lehessen alkalmazni. Ez azt követeli, hogy több programozói nyelvből is lehessen használni. Célul tűztük ki a matematikai szoftverek támogatását is: MATLAB, Freemat.
- *egységes megközelítés*: A rendszerünk több módszert is támogat. Cél olyan felépítés kidolgozása, ami a módszerek között nem tesz különbséget, azonos felületen keresztül lehet azokat használni.

Az előbb említett követelmények miatt több formában is szállítjuk a keretrendszert: egy programozó könyvtár (C++ library), dinamikusan betölthető könyvtár (*.dll Windows esetén és *.lo Linux esetén) és beágyazó osztályokat a népszerűbb programozó nyelvekhez (C#, Java, Python), amelyek a dll-t használják.

Az egységes használat érdekében úgy terveztük meg az osztály hierarchiát, hogy bármely azonos típusú FRI módszert azonos felületen keresztül lehessen használni.

Két fajta használatot is támogatunk: *egyszerű* és *asszisztált*.

Az egyszerű használati mód azt jelenti, hogy az osztályokat közvetlenül lehet használni, azaz példányosítani, majd ezután metódusait hívni. Ebben az esetben a fejlesztő feladata és felelőssége létrehozni, tárolni és kellő időben törölni a fuzzy rendszert és az azokhoz kapcsolódó segéd objektumokat. A rendszerek tárolására a dinamikus vector struktúra javasolt, mert ekkor az új elemek hozzáadása, létezők törlése, megfelelő elemek kiolvasása megoldott a már létező operátorok segítségével.

Az asszisztált használati mód pedig azt jelenti, hogy a fejlesztő csak kéréseket küld a rendszerünknek egy új rendszer létrehozását, vagy létező rendszer törlését kérve.

Tikk és szerzőtársai által meghatározott tulajdonságokból csak az első támogatható, hiszen a megvizsgált módszerek több tagja sem elégíti ki a többi feltételt. Az FRI módszerek implementációi kivételeket idéznek elő közvetlen használat esetén. Asszisztált használat esetén a metódusok visszatérési értéke képviseli a műveletek sikeres voltát. A sikeresen futott függvények nulla értékkel térnek vissza, a sikertelenek pedig egy hibakóddal, ami egy negatív érték. Minden hibának saját hibakódja van az egyértelmű hibakezelés érdekében.

3.1. A keretrendszer szerkezete

A keretrendszer magja C++ nyelvben íródott annak gyorsasága, és az objektum orientált volta miatt. Az objektum orientáltság lehetőséget nyújt az újrahaználható és az átdefiniálható program elemek

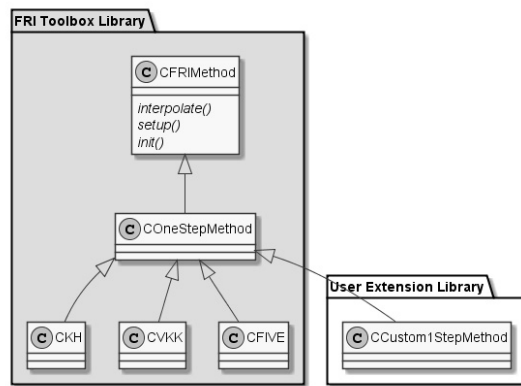
használatára, így az előre megírt kódot a végfelhasználó tetszőleg módosíthatja, újra értelmezheti bizonyos elemeit.

Mivel a rendszer osztályok halmaza, így egyes részeit is használhatja saját logikájának megvalósítására. A mi könyvtárunk támogatja mind az egy lépéses, mind a két lépéses módszereket. A rendszer definiálására ugyanúgy a FIS file-t használja, mint az FRI Matlab Toolbox, a kompatibilitás érdekében. Ha a végfelhasználó már használta a Matlab fuzzy toolbox-át, vagy a Johanyák által kifejlesztett FRI Matlab Toolbox-ot, akkor ugyanazokat a rendszert leíró fájlokat itt is használhatja. Miután a rendszerünk létrehozta megfelelő fuzzy rendszert, eltárolja azt. A fejlesztő nem férhet hozzá közvetlenül a belső objektumokhoz, csak hivatkozhat azokra minden használat alkalmával. A fuzzy rendszer létrehozása és törlése egyszerű metódus hívás, és nem memóriát kezelő operátor használat. Keretrendszerünk használatára ez utóbbi lehetőség javasolt.

Ha a keretrendszerünket egy C++ alkalmazásba kívánják használni mindkét módot választhatják, mert a könyvtár tartalmazza mind az osztályok hierarchiáját, mind a menedzselő kód is. A dll használata esetén azonban csak az asszisztált mód lehetséges, kompatibilitási és biztonsági okokból. Ha a burkoló osztályokat (wrapper) használják (C#, Java, vagy Python) akkor is csak asszisztált használat lehetséges, mert a különböző programozói nyelvek csak a C-ben megírt dll-eket támogatják.

Az egységes használat érdekében minden módszer osztálya közvetve vagy közvetlenül a CFRIMethod osztályból származik. Ez a közös ősosztály biztosítja az a különböző módszerek egységes interfészét. A felhasználó négy lépésben tudja használni az adott FRI módszert:

1. Rendszer létrehozás: Első lépésként a Fuzzy rendszert kell létrehozni (Ennek szabálybázisa lehet ritka is.)
2. FRI módszer kiválasztása: Itt adja meg a felhasználó, hogy most éppen melyik módszert fogja a későbbiekben használni. A módszer neve alapján létrejön a megfelelő objektum a háttérben. Ez a lépés később többször is elvégezhető a setup metódus hívásával. Így lehetőség nyílik a program futása során módszert váltani.
3. FRI módszer inicializálása: A módszerek különbözősége miatt lehetőséget biztosítunk egyéb módszer függő paraméterek használatára, melyet az init metódussal lehet megadni.
4. Következmény lekérdezése: A használat fő lépése, mely során az adott megfigyeléshez tartozó konzekvens lekérdezhető. Az interpolate metódus hívásán keresztül az éppen kiválasztott FRI módszer meghatározza a következményt.



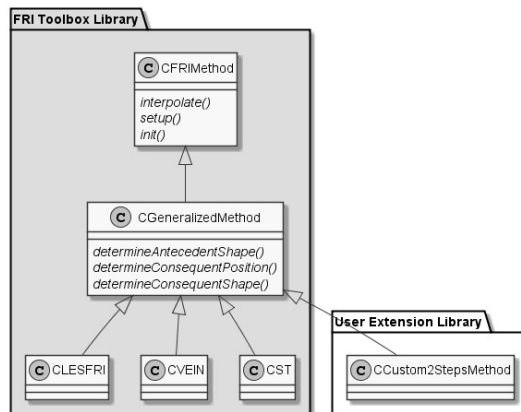
1. ábra. Egy lépéses FRI módszerek struktúrája

Az egy lépéses FRI módszerek mindegyike a COneStepMethod osztályból származik és felül kell definiálni az interpolate metódust. Ha új egy lépéses módszert szeretnénk létrehozni, akkor ezen szabályok szerint kell létrehozni, mint CCustom1StepMethod osztály az 1. ábrán.

A keretrendszer már tartalmaz néhány ilyen módszert, melyek KH (CKK osztály), VKK (CVKK osztály) és FIVE (CFIVE osztály) módszerek.

A kétlépéses FRI módszerek pedig CGeneralizedMethod osztályból származnak és felül kell definiálnia a determineAntecedentShapes, determineConsequentPositions és determineConsequentShapes metódusokat. Ha egy két lépéses módszert szeretnénk bevezetni, akkor ezen szabályok szerint kell létrehozni, mint CCustom2StepMethod osztály a [2] ábrán.

A keretrendszerben már található néhány ilyen módszer, melyek LESFRI (CLESFRI osztály), VEIN (CVEIN osztály) és ST (CST osztály) módszerek.



2. ábra. Két lépéses FRI módszerek struktúrája

1.1. A keretrendszer használata

Az új keretrendszert mind programkódból, mind szoftverek (Matlab, FreeMat) alól lehet használni. C++ projektek esetén a meglévő osztályokat közvetlenül használhatjuk, valamint a legjobb teljesítmény is így érhető el, azonban ilyenkor több a fejlesztő feladata és felelőssége is. Minden más esetben a dll-en keresztül használhatjuk a keretrendszert.

A dll használata szinte minden mai programozói kódból megoldott csupán az adott környezet leírását kell tanulmányoznunk. Az egyszerű használat érdekében a legnépszerűbb nyelvekhez készítettünk beágyazó osztályokat (C#, Java, Python, VB.NET), amelyeket csak példányosítani kell, majd hívni a megfelelő metódusokat.

Elsődleges célunk között szerepelt a MATLAB támogatása, annak népszerűsége, és gazdag matematikai metódusgyűjteménye miatt. A MATLAB lehetőséget biztosít dll betöltésére, és a benne levő globális függvények használatára. Ehhez a header és a dll fájl, valamint a loadlibrary és calllib függvények használata szükséges.

A MATLAB mellett az ingyenes matematikai rendszerek támogatása is fontos számunkra, ezért a FreeMat ingyenesen használható, kis méretű szoftvert támogatjuk elsőként, amely letölthető a [13] helyről. A MATLAB-hoz hasonló módon itt is lehetőség van a dll-ben levő függvények használatára. Itt az import függvényt kell használnunk, hogy a külső függvényeket belsőként tudjuk elérni.

2. KÖVETKEZTETÉS

A ritka szabálybázisú rendszerek esetében is használható, módszereket implementálta Johanyák MATLAB rendszerben, azonban a gyakorlati alkalmazása valós környezetben nehézkes.

Ezen cikk által bevezetett Fuzzy szabály interpolációs programozói keretrendszert (FRI Toolbox library), amely szabadon elérhető programozói könyvtár és dll. Ezen új megoldás használatával valós szabályzó és vezérlő alkalmazások fejleszthetők, köszönhetően a gyors válaszidőnek és a könnyű használatnak. Ezt a több módszert is implementáló keretrendszert könnyen használhatjuk bármely manapság népszerű nyelv esetén, és nincs szükség a MATLAB komplex matematikai rendszerre.

Azoknak a fejlesztőknek javasoljuk, akik valós környezetben futó rendszereket fejlesztenek.

A rendszerünk továbbfejlesztéseként több FRI metódust szeretnénk integrálni valamint egy benchmark rendszert kifejleszteni, amelyben az implementált módszereket össze lehet vetni minta adatok alapján.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

6. IRODALOM

- [1.] Kóczy L. T. and Hirota K., Rule interpolation by α -level sets in fuzzy approximate reasoning, BUSEFAL, vol. 46, no. Automne, pp. 115–123, 1991.
- [2.] D. Tikk and P. Baranyi, "Comprehensive analysis of a new fuzzy rule interpolation method," IEEE Trans. On Fuzzy Systems, vol. 8, no. 3, pp. 281–296, 2000.
- [3.] S. Kovács and L. T. Kóczy, "Application of an approximate fuzzy logic controller in an agv steering system, path tracking and collision avoidance strategy," Tatra Mountains Math. Publ., vol. 16, pp. 456–467, 1999.
- [4.] K. W. Wong, T. D. Gedeon, and D. Tikk, "An improved multidimensional α -cut based fuzzy interpolation technique," in Proc. of the Int. Conf. on Artificial Intelligence in Science and Technology (AISAT'00), V. Karri and M. Negnevitsky, Eds., Hobart, Tasmania, Australia, December, 2000, pp. 33–38.
- [5.] G. Vass, L. Kalmár, and L. T. Kóczy, "Extension of the fuzzy rule interpolation method," in Proc. of the Int. Conf. on Fuzzy Sets Theory and its Applications (FSTA'92), Liptovsk' y Jan, Slovakia, 1992, pp. 1–6.
- [6.] P. Baranyi, L. T. Kóczy, and T. D. Gedeon, "A generalized concept for fuzzy rule interpolation," IEEE Trans. on Fuzzy Systems, vol. 12, no. 6, pp. 820–837, December 2004.
- [7.] S. Yan, M. Mizumoto, and W. Z. Qiao, "An improvement to Kóczy and Hirota's interpolative reasoning in sparse fuzzy rule bases," Int. J. of Approximate Reasoning, vol. 15, pp. 185–201, 1996.
- [8.] Z. H. Huang and Q. Shen, "Fuzzy interpolation with generalized representative values," in Proc. of the UK Workshop on Computational Intelligence, Loughborough, UK, September, 2004, pp. 161–171.
- [9.] S. Jenei, "Interpolation and extrapolation of fuzzy quantities revisited - (I) An axiomatic approach," Soft Computing, vol. 5, pp. 179–193, 2001.
- [10.] Johanyák Zs. Cs., Tikk D., Kovács Sz., és Wong K. W., Fuzzy rule interpolation Matlab toolbox – FRI toolbox, in Proc. of the IEEE World Congress on Computational Intelligence (WCCI'06), 15th Int. Conf. on Fuzzy Systems (FUZZ-IEEE'06).
- [11.] Z. Johanyák. Fuzzy rule interpolation matlab toolbox website. [Online]. Available: <http://fri.gamf.hu>
- [12.] D. Tikk, Z. C. Johanyák, S. Kovács, and K. W. Wong, "Fuzzy rule interpolation and extrapolation techniques: Criteria and evaluation guidelines," Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 15, pp. 254–263, 2011.
- [13.] Freemat website. [Online]. Available: <http://freemat.sourceforge.net>

FOGALOMHÁLÓ ALAPÚ OSZTÁLYOZÁSI MÓDSZEREK

CLASSIFICATION METHOD BASED ON CONCEPT LATTICE ARCHITECTURE

*Kovács László**

ABSTRACT

The classification is a central task in data mining and knowledge engineering. There is a huge literature on the classic methods of classification. The paper presents an approach on development of a novel algorithm using concept lattice structure. In the classification based on Formal Concept Analysis (FCA), the class label is treated as special category attribute. The FCA provides an efficient mechanism to perform generalization of the attribute structure. The decision tree generated from the lattice provides better results than the traditional methods for a test grammar induction problem.

1. OSZTÁLYOZÁS FELADATKÖRE

Az számítógépi intelligencia problémaköreinek egyik fő eleme az osztályozás feladata, melynek során az objektumokat előre definiált osztálykódokhoz rendeljük. Az osztályozás feladatánál feltesszük, hogy az objektumok egy magadott tulajdonsághalmazzal jellemezhetők és az objektum osztálykódja az objektum tulajdonságaitól függ. Az osztályozás célja az ismeretlen kapcsolat leíró f függvény mellett egy olyan

$$g : O \rightarrow 2^C$$

osztályozó függvény meghatározása, melyre

$$E(f, g, S) \rightarrow \min$$

teljesül, ahol az $E(\cdot, \cdot)$ függvény a hibafüggvényt, S a tanítóhalmazt és C az osztálykódok halmazát jelöli. Az így kapott g függvény felhasználható az f függvény közelítésére is a teljes O halmazon. A hibát egy adott objektumnál valós értékű osztályozó függvények mellett rendszerint az eltérésnégyzettel mérjük:

$$E(f, g) = \sum_{o \in S} (f(o) - g(o))^2 .$$

Az osztályozó függvény statisztikai alapú meghatározása egy felügyelt tanítással történik. A felügyelt tanítás során a tanítóhalmaz

$$T = \{(o, f(o)) \mid o \in S\}$$

alakú, tehát a mintában minden objektumhoz ismertek az odatarozó kategória értékek. Az ismeretlen g osztályozó függvény meghatározására alapvetően kétféle módszer terjedt el:

- particionáló módszer, melyben az objektumok vektorterét különböző kategóriákhoz rendeljük;
- osztályvalószínűségi függvény alapú, amikor egy $P(c|o)$ valószínűségi értéket határozzunk meg minden érintett kategóriára.

Az irodalomban az osztályozás területe egy aktívan vizsgált terület, igen sok osztályozási algoritmus változatot dolgoztak ki. A teljesség igénye nélkül megemlíthetjük az elterjedtebb algoritmusokat. Az egyik csoportba tartoznak a döntési fa alapú algoritmusok, mint az ID3 [1] és a C4.5. A szabályalapú működést megvalósító módszerek közül kiemelhetők a CN2, CL2 [2] módszerek. A tiszta statisztikai elveken működő eljárásokhoz tartozik a CART, a Bayes osztályozók [3] és a genetikus algoritmus alapú változatok. A nemlineáris modellek közül kiemelhető a neurális háló alapú modell [4] és a nemlineáris regressziós eljárás. A kvadratus programozás alapú osztályozók közé tartozik az SVM [5] módszer. Az induktív logika eszközrendszerével dolgozó osztályozók közé tartozik a [6]-ben javasolt rendszer.

2. FOGALOMHÁLÓ ARCHITEKTÚRÁJA

A fogalomhálók módszere a formális fogalom analízis (FCA: formal concept analysis)[7] elméletéhez kapcsolódik. A rendszer kiindulása egy objektumhalmaz, ahol mindegyik objektum egy megadott tulajdonsághalmazból vehet fel jellemzőket. Ezen halmazra vonatkozólag fogalom alatt objektumok és tulajdonságok halmazainak olyan párosát értjük, melyre teljesül, hogy az objektumok mindegyike rendelkezik a megadott tulajdonsággal,

**egyetemi docens, Miskolci Egyetem Általános Informatikai Tanszék*

és a halmazon kívüli objektumoknál a megadott tulajdonsághalmaz együtt már nem fordul elő; a tulajdonsághalmaz tartalmazza az objektumok minden közös tulajdonságát. Ha a tulajdonsághalmaz szerinti jelölését vesszük, akkor az egyes fogalmak között a leíró tulajdonsághalmaz szerinti tartalmazási relációt kivetíthetjük a kapcsolódó fogalmak közötti tartalmazási relációra. Ezáltal egy részben rendezést kapunk a fogalmakra, s az is belátható, hogy a kapott struktúra egy háló. A kapott hálót nevezik fogalomhálónak.

Kontextus alatt egy $K(G,M,I)$ hármast értünk, ahol G az objektumok halmaza, M a tulajdonságok halmaza és I egy reláció G és M vonatkozásában. Az $A \subseteq G$ halmazokra egy deriválási operátor értelmezett, melynek jelentése:

$$A' = \{a \in M \mid \forall g \in A: gIa\}$$

és ennek analógiájára a tulajdonság halmazokra is felvesszünk egy hasonló operátort:

$$B' = \{g \in G \mid \forall a \in B: gIa\}.$$

A deriválási operátorra igaz, hogy minden A_i -re

$$(\cup_i A_i)' = \cap_i A_i'$$

illetve minden B_i -re

$$(\cup_i B_i)' = \cap_i B_i'$$

teljesül. A $C(A,B)$ párost fogalomnak nevezik a K kontextusra vonatkozólag, ha

$$A \subseteq G; B \subseteq M; A' = B; B' = A$$

teljesül. A fogalomnak az objektum részét nevezik extent-nek, a tulajdonság részét pedig intent-nek. A $K(G,M,I)$ -re vonatkozó fogalmak halmazát jelöljük Φ -vel. Ekkor a Φ elemei között egy rendezési reláció értelmezhető a következő módon:

$$C_1 \leq C_2 \Leftrightarrow A_1 \subseteq A_2,$$

ahol C_1 és C_2 tetszőleges fogalmak. Belátható, hogy minden (C_1, C_2) fogalompárosra teljesül, hogy:

$$C_1 \wedge C_2 \in \Phi, C_1 \vee C_2 \in \Phi.$$

Ez alapján a $\Lambda(\Phi, \leq)$ páros hálónak (lattice) tekinthető, melyet fogalomhálónak neveznek. A háló felépítésének

algoritmusát elemzi többek között a [10],[12] publikációk.

3. FCA ALAPÚ OSZTÁLYOZÁS

Az osztályozó algoritmusok célja osztálykategóriák objektumokhoz való rendelése az objektumok tulajdonságai alapján. A fogalomháló alkalmas eszköz arra, hogy az összes lényeges, zárt attribútum csoportra megvizsgáljuk az osztályhovatartozás jellegét [11]. A fogalomhálók osztályozási feladatokra történő alkalmazására tett első javaslatok között kiemelhető Zhao [8] dolgozata. A felvázolt modellben az objektum attribútumok közül ez egyiket osztálykategóriaként kezeljük. A klaszifikációs szabály leírja a kategória címkének a logikai formuláktól való függőségének mikéntjét.

A Zhao által kidolgozott modellben egy adott $K(G,M, I)$ kontextus esetén, a klaszifikációs szabály $f \Rightarrow c$ alakú. A szabályban az f jelöli a logikai formulákat, míg a c szimbólum az a_c kategória attribútum egy lehetséges értékét, tehát egy osztálykategóriát jelöl. A megadott szabály jósága a konfidencia és az általánosíthatóság mérőszámaival határozható meg, ahol

$$\text{conf}(f \Rightarrow c) = \frac{|m(f \cap (a_c = c))|}{|m(f)|};$$

$$\text{generality}(f \Rightarrow c) = \frac{|m(f)|}{|O_K|}.$$

Minél nagyobb a konfidencia érték annál pontosabb az osztályozás eredménye. A konzisztens osztályozási szabály olyan osztályozási szabály, ahol a konfidencia értéke 1, azaz

$$|m(f \cap c)| = |m(f)|$$

teljesül. Az $m(f)$ szimbólum azon objektumok halmazát jelöli, melyek kielégítik az f predikátumot. A (X, f) konjunktív fogalmat konzisztens fogalomnak nevezik, ha az egy egyedi kategória címkét indukál és konfidencia értéke 1. A legáltalánosabb konzisztens fogalom olyan konzisztens fogalom, ahol a szülő fogalmak egyike sem konzisztens fogalom. Be lehet bizonyítani, hogy a legáltalánosabb konzisztens fogalmak az univerzumok lefedését alkotják. Emiatt a legáltalánosabb konzisztens fogalmak elegendőek az osztályozási feladatok elvégzésére.

A halmaz felépítésének brute-force megközelítésében, az összes fogalom hálóját építik el először. Ezt követően a konzisztens illetve a legáltalánosabb konzisztens fogalmak válogatjuk ki.. Az algoritmus az alábbi lépéseket tartalmazza:

1. A konzisztens klasszifikációs szabályok generálása az alábbi algoritmus alapján:
 - (a) ciklus az összes kategória címkére (c).
 - (b) a $c|\varphi$ valószínűség értékének meghatározása.
 - (c) a legvalószínűbb kategória kiválasztása.
2. Konzisztens definíciós pár generálása minden szabályhoz.
3. A konjunktív konzisztens fogalmak halmazának előállítás.
4. Azon konjunktív konzisztens fogalmak eliminálása, melyek nem a legáltalánosabb fogalmak közé tartoznak.

A tapasztalatok [9] azt mutatják, hogy a fogalmi hálón alapuló osztályozási módszerek jobb osztályozási pontosságot tudnak elérni, mint az ID3 alapú döntési fák. A háló alapú megközelítés előnye, hogy a formulák minden lehetséges zárt csoportja feldolgozásra kerül a legjobb kategória korrelációs értékkel bíró formulák meghatározásakor. Hátránya, hogy a háló felépítése nagyon költséges folyamat.

A fenti elképzelésből kiindulva egy olyan hibrid modell került kidolgozásra hoztam létre, amelyben a fogalomhálóból egy döntési fa generálódik, s az így kapott fa közvetlenül felhasználható az objektumoknak az attribútumaik szerinti osztályozására.

Jelölje Λ a megadott $K(G,M,I)$ kontextushoz tartozó fogalomhálót. Bővítsük az attribútumok körét a kategória attribútummal. Az így kapott fogalomhálót jelöljük Λ^* -vel. Adott $\Lambda = (\{A_i, B_i\}, \leq)$ háléhoz a kibővített $\Lambda^* = (\{A_i, B_i\}, \leq, c, d, s)$ fogalomháló tartalmaz egy

- s: $T \rightarrow N^+$ támogatottsági értéket;
- c: $T \rightarrow C$ kategória címkét;
- d: $T \rightarrow C$ default kategória címkét.

A T szimbólum az Λ háló fogalmainak halmazát jelöli, a támogatottság a fogalomtól kisebb (\leq reláció) dominált fogalmak száma. A nem atomi fogalmaknál a c értéke a dominált fogalmak kategóriáinak metszete. A default kategória értéke egyenlő azon nem üres kategóriával, amely a dominált fogalmak között a legnagyobb támogatottsággal rendelkezik. Az osztályozásnál a maximálisan konzisztens fogalmak attribútum elemei alapján határozhatjuk meg a bejövő attribútum halmazhoz rendelhető osztálykategóriát.

Az Λ^* hálóban a következő szabályok érvényesülnek:

- Ha egy adott c kategóriához csak egyetlen χ maximálisan konzisztens fogalom létezik, melynek intenzió része χ_a , akkor

$$\chi_a \Leftrightarrow c$$

teljesül, azaz c akkor és csak akkor érvényes, ha χ_a érvényes. Így a kapcsolódó döntési fában egyetlen csomópont elegendő ezen kategóriához.

- Ha egy adott c kategóriához a χ egy konzisztens fogalom, melynek intenzió része χ_a , akkor

$$\chi_a \Rightarrow c$$

teljesül. A kapcsolódó döntési fában a χ_a -hez egy olyan csomópont fog tartozni, melynek egyetlen homogén c -hez kapcsolódó leszármazottja van.

A generált döntési fa előnye, hogy könnyen érthető a külső szakemberek által is és jól követi az emberi gondolkodásmódot.

A döntési fa generáló algoritmus további sajátossága, hogy azon csomópontok, melyek nem konzisztensek, a dominált csomópontok alapján legvalószínűbb kategória kódot kapják meg jellemzőként. A generált fa hatékonyabb kezelésére az elkészült fában egyes elemek eliminálásra kerülnek. A redukció azon gyerekeket eliminálja, melyek osztálykategória jelzője megegyezik a szülő kategóriájával.

4. KISÉRLETI EREDMÉNYEK

A kidolgozott módszer tesztelésére egy, a szavak ragozását végző feladat került kiválasztásra. A szavak ragozása egy osztályozási feladatként is értelmezhető, ahol a rag megfelel az osztálykódnak. Az induló kísérlet csak minimális, pár száz méretű tanító halmazzal dolgozott. A kísérletben a FCA alapú módszer mellett FSA és HMM alapú módszerek kerültek implementálásra. A mintarendszer fejlesztése Java nyelven az Oracle JDeveloper eszköz segítségével történt. A betanítás sikerességét az U és T kontroll halmazokkal ellenőrizzük. A kis minta esetén a kontroll halmazok is kisebb méretűek. A T halmazban a már betanított szavak szerepelnek, míg az U halmaz a még ismeretlen szavakat tartalmazza. A minta U halmaz a következő szavakat fogta össze:

$U = \{\text{gatyá, labda, tanár, krumpli, fej, korong, csapat, kód, ló, korom}\}$

A ragozott alakok ekkor a következők:

$R = \{\text{gatyát, labdát, tanárt, krumplit, fejet, korongot, csapatot, kódot, lovat, kormot}\}$

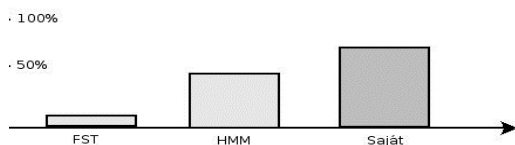
Az FST módszernél igen rossz eredmény született az U halmaz vizsgálatánál, hiszen ezekre még nem kapott példát a modell:

$R' = \{\text{g, l, tanítót, k, fecskét, k, csattot, kö, lócát, k}\}$

A HMM esetében is hasonló rossz eredmények születtek a kis tanító halmaz esetében. A saját háló alapú módszernél viszont már előfordultak jó becslések is:

$$R' = \{\text{gatyát, labdát, tanárt, krumplit, fejt, korongot, csapatot, ködöt, lőt, koromet}\}$$

Látható, hogy itt 60%-os pontosságot generált a modell. Az is észrevehető, hogy a tévesen ragozott alakok halmaza is sokkal közelebb áll a humán gondolkodáshoz, mint a másik két módszernél kapott eredménylista.



1. ábra. Találati pontosságok összehasonlítása

A pontosság mellett a háló alapú módszer másik előnye, hogy viszonylag gyorsan tud döntést hozni. A transzformált alak előállítását itt lényegesen hatékonyabb, mint a HMM esetében, ahol dinamikus algoritmussal kell meghatározni a legkisebb összköltségű utat.

5. ÖSSZEFOGLALÁS

A tanulmány a fogalomháló alapú osztályozási rendszer működési elvét mutatja be kiegészítve annak döntési fával kombinált változatával. A kidolgozott módszer előnye az általánosítási operátor erőteljessége. A mintarendszerben a módszer hatékony osztályozási eszköznek bizonyult.

6. KÖSZÖNETNYÍLVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

7. IRODALOM

- [1] QUINLAN J. : Induction of decision trees. Machine Learning, 1:85–106, 1986
- [2] CLARK P, NIBLETT T.: The CN2 induction algorithm. Machine Learning;3(4): 261–83., 1989
- [3] FRIEDMAN N, GEIGER D, GOLDSMIDT M.: Bayesian network classifiers. Machine Learning; 29 (2):131–63, 1997

- [4] RUMELHART DE, HINTON GE, WILLIAMS RJ.: Learning internal representations by error propagation. Parallel distributed processing: explorations in the micro-structure of cognition. MIT Press; pp. 318–63., 1986
- [5] VAPNIK V.N.: The nature of statistical learning theory. New York: Springer; 2000
- [6] DZEROSKI S.: Inductive logic programming and knowledge discovery in databases. Advances in knowledge discovery and data mining. p. 117–52., 1996
- [7] GANTER B. AND WILLE R.: Formal Concept Analysis, Mathematical Foundations, Springer Verlag, 1999.
- [8] ZHAO Y. –YAO Y.: Classification based on logical concept analysis, Proceedings of 19th Conference on the Canadian Society for Computational Studies of Intelligence (Canadian AI 2006), Québec City, Québec, Canada, June 7-9, pp. 419-430, 2006
- [9] FERRE, S., RIDOUX, O.: A logical generalization of formal concept analysis, 2001
- [10] KOVACS L.: Efficiency Analysis of Building Concept Lattice, Proceedings of 2nd ISHR on Computational Intelligence, Budapest, 2001
- [11] KOVACS L. , BARANYI P.: Document Clustering based on Concept Lattice, Proc. of IEEE SMC, Tunisia, pp 67-75, 2002
- [12] VALTCHEV. P., MISSAOUI, R. : Building concept (Galois) lattices from parts: generalizing the incremental methods., Proc. of ICCS01, pp 290-303. 2001

OSZTÁLYOZÁSI FELADATOK A KÉRDÉSGENERÁLÁSI MINTARENDSZERBEN

CLASSIFICATION TASKS IN THE SYSTEM OF QUESTION GENERATION MODEL

Bednarik László, dr. Kovács László***

ABSTRACT

Automatic question generation on text documents has become a more and more important field of the recent researches of artificial intelligence. Therefore analysing text documents and gaining useful information from them requires more and more effective methods. One stage in our results gained in this field is to classify elements of documents. In the concept described below we realised this task by building on the neural net based learning theory. The new idea in this method is to fit the general theory to the specific features of the task to be carried out. The correctness of the model was proven by an application implemented in Java language that supports to demonstrate teaching with involving specialist and results of classification in an interactive graphical interface.

1. BEVEZETÉS

A dokumentumok szavainak klaszterezése révén előálltak mindazok az objektív koordináták, melyek az emberi tényezőktől függetlenül leírják minden szó elhelyezkedését a definiált hét dimenziós objektív térben. A cél egy olyan szabályrendszer feltárása, mely képes az objektív térben adott szavakat (mint pontokat) automatikusan elhelyezni az igényeink szerint definiált szubjektív térben. Az osztályozási feladat magját egy előrecsatolt háromrétegű neurális-hálóval valósítottuk meg.

2. AZ OSZTÁLYOZÁS IRODALMI ÁTTEKINTÉSE

A szöveges adatok egyik jellemzője, hogy az információt strukturálatlan vagy gyengén strukturált formában tartalmazzák [8]. A strukturálatlan adatok kezelésére részben megoldást jelent a strukturált tárolásuk, azaz a dokumentumok hierarchikus rendszerbe (taxonómiába) való rendszerezése [1].

Az osztályozás célja, hogy az objektumok halmazán egy tulajdonsághoz osztálykódot rendelő függvényt alkossunk meg, amely a tanuló mintára adja vissza az ismert osztálykódot [4].

Dokumentumok osztályozására a legelterjedtebben alkalmazott módszereket a döntési fa, a legközelebbi szomszéd (k -NN), a Bayes háló, a szupportvektor gép, a neurális háló, valamint a CPN háló algoritmusainak feladatspecifikus továbbfejlesztései alkotják. Az említett módszerek közül kettő kerül bemutatásra.

2.1. Neurális hálózat alapú osztályozó

Az osztályozást végző modellt úgy terveztük meg, hogy képes legyen bármilyen tématerülethez tartozó szöveg kezelésére. Az egyik legelterjedtebben alkalmazott módszer a neurális hálóval végzett tanulás, melynek elvét McCulloch és Pitts publikálta 1943-ban. A neurális háló irányított kapcsolatokkal összekötött egységekből (neuron) állnak. A j jelű neuronból az i jelű felé vezető kapcsolat hivatott a j jelű neuron aktivációs állapotát (jele: a_j) az i jelű neuronhoz továbbítani. Minden egyes kapcsolat rendelkezik egy hozzá társított $W_{j,i}$ numerikus súllyal, ami meghatározza a kapcsolat erősségét és előjelét. Minden egyes neuron először a bemeneteinek egy súlyozott összegét számítja ki az 2.1. összefüggés alapján.

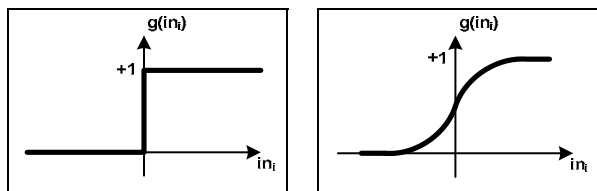
$$in_i = \sum_{j=0}^n W_{j,i} * a_j \quad (2.1)$$

Az i -edik neuron aktivációs állapota a bemeneteinek súlyozott összegére alkalmazott g jelű aktivációs függvényrel kerül meghatározásra az 2.2. összefüggés szerint [5].

$$a_i = g(in_i) = g\left(\sum_{j=0}^n W_{j,i} * a_j\right) \quad (2.2)$$

A 2.1. ábra két lehetséges aktivációs függvényt mutat be, a küszöbérték függvényt, és a szigmoid függvényt [3].

Mindkét függvény küszöbpontja az $x = 0$ értéknél van. A megoldott feladat során ezt az értéket minden neuron esetén $x = 0$ -ra állítottuk be ám lehetőséget biztosítottunk a későbbi módosíthatóságára.



2.1. ábra. Két lehetséges aktivációs függvény a neuron kimenetének meghatározásához

Az információk áramlási iránya alapján a neurális hálózatok két fő csoportba sorolhatóak. Ezek az előrecsatolt hálók (*feed-forward network*) [7] illetve a visszacsatolt hálók (*recurrent network*) [6].

Az előrecsatolt neurális hálózatok megkülönböztethetők egymástól a rétegek száma, az egyes rétegekben lévő neuronok száma, valamint a neuronok közötti kapcsolatok alapján is. Általánosságban egy neurális hálózat három különböző feladatot ellátó rétegből épül fel. Ezek a bemeneti réteg, rejtett réteg, kimeneti réteg [9].

A CPN (*Counter-Propagation Network*) hálózat alapú osztályozók lényegében a neurális hálózatok egy feladatspecifikus alkalmazásai. A CPN hálózat bemeneti rétege itt is az osztályozandó objektumok leírását tartalmazza. A rejtett rétegben (Kohonen réteg) lévő neuronok ebben az esetben a klasztereket reprezentálják. A kimeneti réteg neuronjai pedig azokat az osztályokat írják le, amelyekbe a bemeneten lévő objektumok besorolásra kerülnek. A neuronok bemeneti függvénye a 2.3 összefüggés alapján kerül meghatározásra.

$$s = \sum w_i * s_i \quad (2.3)$$

Az s_i jelöli a bejövő jelet, w_i pedig az él súlyát. Az s érték megfelel a w és az s vektorok skaláris szorzatának, ami a közbezárt szög koszinuszával arányos. A tanulási folyamat során a győztes neuron mindig közelebb kerül a bemenetre kapcsolt objektumhoz.

3. AZ OSZTÁLYOZÁSI FELADAT MEGVALÓSÍTÁSA

3.1. Az algoritmus bemeneteinek és kimeneteinek megtervezése

A megvalósításra került osztályozó algoritmus a dokumentum szavainak objektív koordinátái illetve a tanító minta alapján állítja elő szavak szubjektív koordinátáit. Az objektív koordináták megválasztásánál a szavak olyan tulajdonságainak a megtalálása volt a célunk, melyekkel jól jellemezhetőek a vizsgált szavak, ugyanakkor meghatározásuk nem ró lényeges számítási többletet az osztályozást végző programmodulra. Ezek alapján minden szó objektív vetülete hét koordinátával került definiálásra. Az alkalmazott objektív koordináták [2] és lehetséges értékei: szófaj [-1, 0, 1, 2, 3,4, 5, 6, 7], ragozottság [-1, 0, 1], szó száma a dokumentumban [1 ... előford. szám], szó száma a mondatban [1 ... előford. szám], az adott szó helye a mondatban [1 ... mondat szavai], klaszter sorszáma [-1, +1] egész szám, szó távolságátlaga [0.0 ... 1.0] valós szám.

Az osztályozást végző algoritmus kimenetét a szavaknak az emberi tényezőktől függő (szubjektív) koordinátái alkotják. Az osztályozás eredményeként szolgáló szubjektív koordináták: relevancia, specialitás, értelmesség, nehézség és kérdésként kiemelésre kerüljön-e a szó.

3.2. Az alkalmazott neurális hálózat struktúrája

Az osztályozás egy előrecsatolt háromrétegű (egy belső réteg) neurális háló alkalmazásával került megvalósításra. A neuronok közötti kapcsolatok súlyértékei [-1.0, ..., 1.0] tartományba tartozó valós számok. A háló neuronjainak aktiváltsági állapota eltolás nélküli küszöbérték függvényvel lett reprezentálva, mely a bemeneti jeleket súlyozottan összegző bemeneti függvény értékét a [0, 1] egész intervallumra képezi le az 3.1. összefüggés alapján.

$$a_i = 1, ha \sum_{j=0}^n W_{i,j} * a_j > 0.0$$

$$a_i = 0, egyébként \quad (3.1)$$

A háromrétegű neurális hálózat első rétegét a bemeneti neuronok alkotják, melyeknek feladata, hogy az objektív koordináták értékeit a hálózat belső (rejtett) rétegében lévő neuronok felé továbbítsák. A kimeneti neuronok az osztályozandó szavak szubjektív koordinátáira vannak leképezve. A neurális hálózatok építésének egyik kulcskérdését jelenti a belső rétegben lévő neuronok számának meghatározása. Az optimális értéktől alacsonyabb számú neuron alkalmazása esetén a

hálózat nem lesz képes a feladat megtanulásához szükséges mennyiségű információ tárolására.

Jelen feladat esetében a belső rétegben helyezendő neuronok számának pontos meghatározása továbbfejlesztési lehetőségként jelentkezik. Első közelítésként számukat azonosnak vettük a kimeneti rétegben lévő neuronok számával. Eredményként a vizsgált dokumentumok szinte mindegyikénél sikerült 0.01 pontossággal megközelíteni a tökéletes tudásszintnek számító 1.0 értéket. Ez tehát azt támasztja alá, hogy a közbenső rétegben lévő neuronok száma nem kevesebb az optimális értéktől.

3.3. Az alkalmazott neurális hálózat tanulási folyamata

A neurális hálózat tanítása a felügyelt tanítás szabályai szerint lett megvalósítva. A tanulás első lépéseként az előállított hálózat neuronjai közötti kapcsolatok súlyértékei véletlenszerűen beállításra kerülnek a $[-1.0, \dots, 1.0]$ intervallumból. A véletlen számokat generáló algoritmus az adott intervallumon egyenletes eloszlás szerint állítja elő a véletlen számokat.

A súlyok kezdeti értékének beállítását követően az algoritmus sorra veszi a tanítóminta szavait és egymás után a hálózat bemenetére kapcsolja azokat. Ezt egy speciális illesztőmodulon keresztül végzi el, mely a szó minden objektív koordinátájának értékéhez meghatározza a neurális hálózatnak azt a neuronját a dinamikusan felépített bemeneti rétegben, amely az adott koordináta értékét hivatott reprezentálni. Az illesztés eredményeként az adott szónak az objektív térben való elhelyezkedését megadó koordinátákat reprezentáló bemeneti neuronok aktív állapotba kerülnek. A rejtett rétegben lévő neuronok aktiváltsági állapotának beállítását követően kerül sor a hálózat kimeneti rétegében lévő neuronok aktiváltsági állapotának meghatározására. Ehhez a 2.3. összefüggés kerül alkalmazásra, de ebben az esetben a_i az i jelű külső rétegbeli neuron aktiváltsági állapotát, a_j pedig a j jelű rejtett rétegbeli neuronét jelöli. Összefoglalóan az i jelű – kimeneti rétegben lévő – neuron aktiváltsági állapota a bemeneti rétegben lévő neuronok aktiváltsági állapotának függvényében a 3.2. összefüggés alapján határozható meg.

$$a_i = g \left(\sum_{j=0}^n W_{j,i} * g \left(\sum_{k=0}^m W_{k,j} * a_k \right) \right) \quad (3.2)$$

ahol

a_k : a k jelű bemeneti rétegbeli neuron aktiváltsági állapota,

a_i : az i jelű kimeneti rétegbeli neuron aktiváltsági állapota,

m : a bemeneti rétegben lévő neuronok száma,

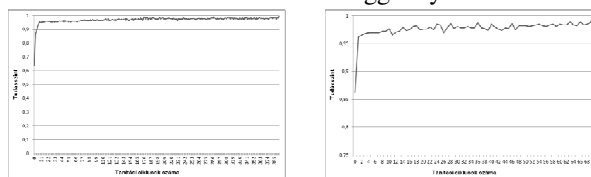
n : a rejtett rétegben lévő neuronok száma,

$W_{k,j}$: a k jelű bemeneti rétegbeli neuron és a j jelű rejtett rétegbeli neuron közötti kapcsolat erőssége,

$W_{j,i}$: a j jelű rejtett rétegbeli neuron és az i jelű kimeneti rétegbeli neuron közötti kapcsolat erőssége,

g : aktivációs függvény, mely 1-es értéket reprezentál, ha a paramétereként kapott összeg pozitív, 0-at egyébként.

A kimeneti rétegben lévő neuronok aktiváltsági állapotai írják le a neurális hálónak a bemenetként kapott szó objektív koordinátáira adott válaszát a szubjektív térben. A tanulás akkor fejeződik be, ha a tanítómintában szereplő szavak osztályozásának átlagos hibája már nem haladja meg a megengedett mértéket, vagy a tanítóminta szavainak ismételt tanulása már eléri az engedélyezett ismétlési számot. A 3.1. ábra a tanulás *súlyegység* paraméterének különböző értékei esetén a háló tudásszintjében tapasztalt változást mutatja a tanítóminta ismétlési számának függvényében.



súlyegység: 0.01

súlyegység: 0.1

3.1. ábra: A tanulás súlyegység paraméterének különböző értékei

A 3.1. ábrán szemléltetett tesztek alapján megfigyelhető, hogy kisebb súlyegység esetén a tanulás sokkal pontosabban tart a célállapot felé. A neurális háló a betanulását követően a tanítómintában szereplő szavak esetén is megőrizte a paraméterként meghatározott elvárt pontosságot.

3.4. A tanulás eredményeinek újrahasznosítása

Az általunk alkalmazott neurális háló struktúrája mindig dinamikusan alkalmazkodik az éppen beolvasott dokumentum szerkezetéhez.

Az ismertetett módszer kiegészítéseként implementálásra került egy másik tudás-újrahasznosítási stratégia is. Ebben nem a betanult neurális hálózat adatait, hanem az osztályozás eredményeül szolgáló mondatokat menti el a rendszer. A megvalósított módszer lényege, hogy az osztályozott mondatok az

osztályozás eredményeül szolgáló szubjektív koordinátáikkal együtt egy adatbázisban kerülnek eltárolásra.

4. AZ ELKÉSZÍTETT ALKALMAZÁS BEMUTATÁSA

Az 3.1. pontokban ismertetett osztályozási feladatot megvalósító modellt alapján elkészítettünk egy Java nyelven implementált alkalmazást, mely grafikus interaktív felhasználói felülete révén támogatást nyújt mind a szakértő bevonásával végzett tanítás, mind pedig az osztályozás eredményeinek demonstrálása területén. A 4.1. ábrán látható képernyőn a szakértő kiválaszthatja azokat a mondatokat, amelyek a neurális háló tanítására szolgáló tanítómintát fogják képezni.

4.1. ábra. A tanítóminta mondatainak kiválasztására szolgáló dialógusablak

A tanítóminta mondatainak beállításai kívánt szavain jobb egérgombbal kattintva megjelenik egy dialógusablak, melyben a szakértő megadhatja a kiválasztott szónak a szubjektív térben való elhelyezkedését leíró koordinátákat. A tanítóminta szavainak a szubjektív térben való elhelyezését követően a neurális háló tanítása a **Tanítás** menüpont kiválasztásával indítható el.

A tanulás befejezésével a neurális háló a megtanult hipotézisfüggvény alkalmazásával automatikusan kitölti a tanítómintába nem tartozó mondatok szavainak szubjektív koordinátáit.

5. AZ OSZTÁLYOZÁS EREDMÉNYEINEK ÖSSZEFOGLALÁSA

A dokumentum szavainak osztályozását végző algoritmus eredményeként megvalósításra került a szavak szubjektív vetületét reprezentáló koordináták automatizált feltárása, valamint megalapozott iránymutatást adtunk a szöveg mondataiból kérdésként kiemelendő szó meghatározására. Ezen lépések képezik alapját a kérdésként kiemelt szó lehetséges

helyettesítésére felkínálható szavak automatikus generálásának, valamint a kérdésre adott válasz értékelésének.

6. KÖSZÖNETNYÍLVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

7. IRODALOM

[1] BLUMBERG R., SHAKU A.: The problem with unstructured data, DM Review, 1 February. 2003b

[2] CSENDES D., CSIRIK J., GYIMÓTHY T.: The Szeged Corpus: A POS Tagged and Syntactically Annotated Hungarian Natural Language Corpus. In: Proceedings of the 5th International Workshop on Linguistically Interpreted Corpora (LINC 2004) Geneva, Switzerland, 23-29 August, pp. 19-23. 2004

[3] DAGAN I., KAROV Y., ROTH D.: Mistake-driven learning in text categorization, In proc. of EMNLP-97, 2nd Conf. on Empirical Methods in Natural Language Processing, pp. 55-63, Providence, USA. 1997

[4] FABRIZIO S.: Machine Learning in Automated Text Categorization, ACM Computing Surveys, 34(1): 1-47. 2002

[5] MC.CULLOCH W. S., PITTS W.: A logical calculus of the ideas immanent in nervous activity, Bulletin of Mathematical Biophysics, 5:115-137. 1943

[6] RAMASUNDARAM R., VICTOR S. P.: Text Categorization by Backpropagation Network, International Journal of Computer Applications (0975 – 8887), Volume 8– No.6, 2010

[7] ROBERT H. N.: Theory of the Backpropagation Neural Network, Washington, DC, USA, 1989

[8] RUDIN, K., CRESSY D.: Will the Real Analytic Application Please Stand Up? DM Review, (13)3, pp. 30-34. 2003

[9] STUART R., PETER N.: Mesterséges intelligencia modern megközelítésben, ISBN 963 545 411 2, Panem könyvkiadó, Budapest, 2005

[10] TIKK D.: Szövegbányászat, Typotex, Budapest, 2007

A "DOUBLE-LINEAR" FUZZY INTERPOLÁCIÓS MÓDSZER KITERJESZTÉSE A FUZZISÁG INTERPOLÁCIÓJÁVAL

EXTENDING THE "DOUBLE-LINEAR" FUZZY INTERPOLATION METHOD WITH THE INTERPOLATION OF FUZZINESS

*Kovács Szilveszter**

ABSTRACT

Most Fuzzy Rule Interpolation (FRI) methods have difficulties in describing the changes of the conclusion's fuzziness. For example, it is difficult to describe cases in which the conclusion for a crisp observation must be fuzzy, or in which an increase in the fuzziness of an observation yields less fuzziness in the conclusion. This problem is mainly inherited from a lack of information in the model, and originates from the deficiency of the fuzzy rule representation. In this paper, a novel rule representation concept called "double fuzzy dot" is suggested, which helps the elaboration of FRI methods in order to be able to better handle the fuzziness of the conclusion. A simple example of such FRI methods is also briefly introduced in this paper.

1. BEVEZETÉS

A fuzzy interpolációs (Fuzzy Rule Interpolation - FRI) következtetési módszerek alapvető problémája, hogy nehézkesen kezelik a következmény fuzzy halmaz fuzziságának változását. Ilyen eset például amikor crisp megfigyeléshez fuzzy halmaz tartozik, vagy az, amikor a megfigyelés halmaz fuzziságának növekedése a következmény halmaz fuzziságának csökkenését kell, hogy eredményezze. A probléma gyökere az információ hiányban, a fuzzy szabályok „fuzzy pont” tudásábrázolási formájában rejlik. Megoldásként jelen cikkben egy új szabályábrázolási formát, a „dupla fuzzy pont” formát vezetjük be. A „dupla fuzzy pont” szabályforma használatával lehetőség nyílik olyan fuzzy következtetési módszerek kidolgozására, melyek a következmény fuzziságának változását is képesek leírni. Az így kialakítható módszerek egy egyszerű példája jelen cikkben is bemutatásra kerül.

2. FUZZY FÜGGVÉNYEK ÉS INTERPOLÁCIÓ

A fuzzy függvények megadására két módszer kínálkozik. Az első a klasszikus leírás alapú, ahol a

fuzzy függvényt fuzzy relációként értelmezzük. A második a deklaratív forma, ahol a fuzzy függvényt két fuzzy univerzum egymáshoz rendeléseként értelmezzük.

A leírás alapú nézet szerint a fuzzy szabályok az antecedens és konzekvens fuzzy halmazok relációjaként értelmezhetők [8]. Ebben az esetben a teljes fuzzy szabálybázis is reláció, az antecedens és konzekvens univerzumok relációja, mely a fuzzy szabály relációk diszjunkciójaként adódik. A klasszikus kompozíciós fuzzy következtetés [3] szabálybázisa is leírás alapú, a következmény fuzzy halmaz a fuzzy megfigyelés és a fuzzy szabálybázis reláció kompozíciójaként számítható.

A cikk további részében a fuzzy függvény deklaratív leírását követjük, azaz két fuzzy univerzum egymáshoz rendeléseként értelmezzük. A Perfilieva *et al.* [1], [2] által bevezetett jelölést használva a fuzzy függvény, mint hozzárendelés az alábbi módon definiálható:

Legyen L^X és L^Y fuzzy univerzum az X és Y alaphalmazon.

Az $f: L^X \rightarrow L^Y$ leképezés fuzzy függvény, ha valamennyi $A, B \in L^X$, $A = B$ esetén $f(A) = f(B)$. $A = B$ iff $\forall x \in X, A(x) = B(x)$.

Egy fuzzy függvény $f: L^X \rightarrow L^Y$ kiterjesztett, ha valamennyi $A, B \in L^X$, $(A \equiv B) \leq (f(A) \equiv f(B))$, ahol \equiv egy L^X, L^Y -on definiált hasonlóság reláció.

Ebben az esetben a fuzzy adatok fuzzy interpolációja a fuzzy függvény kiterjesztéseként értelmezhető:

Legyen $\{(A_i, B_i) \mid i = 1, \dots, n\}$ párokkal megadott fuzzy adat, ahol $A_i \in L^X, B_i \in L^Y, i = 1, \dots, n$, és A_1, \dots, A_n az $=$ szerint eltérő. Mindemellett legyen $f: A_i \rightarrow B_i$ egy fuzzy függvény a $A = \{A_1, \dots, A_n\}$ alaphalmazon, és $P \subseteq \{\emptyset \neq \emptyset : D \rightarrow L^Y\}$, ahol $A \subset D \subseteq L^X$, a lehetséges fuzzy függvények egy kiválasztott részhalmaza. A feladat egy olyan $\emptyset \in P$ (P az interpolációs függvények halmaza) keresése, ahol az interpolációs feltétel $\emptyset(A_i) = B_i, i = 1, \dots, n$ teljesül. Ebben az esetben \emptyset egy az $\{(A_i, B_i) \mid i = 1, \dots, n\}$ fuzzy adatot leíró interpolációs fuzzy függvény. Azaz a \emptyset interpolációs fuzzy függvény az f fuzzy függvény kiterjesztése a D alaphalmazon.

* egyetemi docens, Miskolci Egyetem, ALT

3. FUZZY SZABÁLY INTERPOLÁCIÓS MÓDSZEREK

A fuzzy függvény és a fuzzy interpoláció terminológiáját alkalmazva a fuzzy szabályokra és a fuzzy szabály interpolációra a fuzzy függvény deklaratív leírását követve, a fuzzy szabálybázis n fuzzy szabálya a „fuzzy adat” $\{(A_i, B_i) \mid i = 1, \dots, n\}$, más néven „fuzzy pont”. Ebben az esetben a fuzzy szabály interpoláció alapú fuzzy következtetés nem más, mint egy alkalmas $\varphi(A_i) = B_i, i = 1, \dots, n$ interpolációs fuzzy függvény keresése. Az interpolációs fuzzy függvénynek megfelelően az $x \in L^X$ megfigyelésre kapott $y \in L^Y$ következtetés egyszerűen $y = \varphi(x)$.

Számos fuzzy szabály interpolációs (FRI) módszer született az elmúlt 20 évben. Az egyik első módszer Kóczy és Hirota [5] nevéhez (KH módszer) kötődik. Kizárólag konvex és normális fuzzy halmazokon értelmezték. A következtetés fuzzy halmazt a megfigyelést közrefogó két szabály alapján határozza meg α -vágatonként úgy, hogy a megfigyelés szabály antecedens távolság arányok megegyezzenek a szabály konzekvens, keresett következtetés távolság arányokkal. A KH módszernek számos variánsa született.

Az FRI lényegesen más megközelítést javasolja Baranyi *et al.* [4] tetszőleges formájú halmazokra kiterjesztve az alaplódszert. A javasolt kétlépéses módszer „Általános Módszertan” („General Methodology”, „GM”) először egy interpolált szabályt hoz létre, mely antecedens referencia pontja egybeesik a megfigyelés referencia pontjával. Majd második lépésként egy „egyszabályos következtetéssel” („single rule reasoning”) az interpolált szabály antecedense és a megfigyelés hasonlósága alapján az interpolált szabály konzekvenséből kiindulva meghatározza a következmény fuzzy halmaz végső formáját. A GM módszer gyakorlati alkalmazására ad példát Johanyák *et al.* [9]. A javasolt „Polar Cut” interpoláció polár metszetek segítségével végzi az egyszabályos következtetést.

Számos FRI módszer implementációja a MATLAB FRI Toolbox [6] részeként egy ritka fuzzy szabálybázis modell identifikációs toolbox-al [7] egyetlen fellelhető [11] alatt.

4. A FUZZISÁG INTERPOLÁCIÓJA

Annak érdekében, hogy a fuzzióság változását is kifejezhessük, a fuzzy pont szabály reprezentációt ki kell bővítenünk a fuzzy pont helyén további fuzzióság ábrázolási képességekkel.

Erre a célra jelen cikkben egy egyszerű módszert javasolunk, a fuzzy pont szabály reprezentációnak két egymásra illeszkedő, de eltérő fuzzióságú fuzzy pontra („dupla fuzzy pont”) való cseréjét. A javasolt szabályreprezentációban egy fuzzy szabály valójában

két olyan fuzzy szabály egyesítéseként adódik, ahol az illeszkedő szabályok antecedens és konzekvens referencia pontjai egybeesnek, de fuzzióságuk eltérő.

Az egyszerűbb magyarázat és vizualizáció miatt a továbbiakban a fuzzy halmazok formáját egyenlőszárú háromszögekre korlátozzuk. Ezen megkötés csak a magyarázatot egyszerűsíti, maga az alapkoncepció tetszőleges formájú halmazokra is kiterjeszthető.

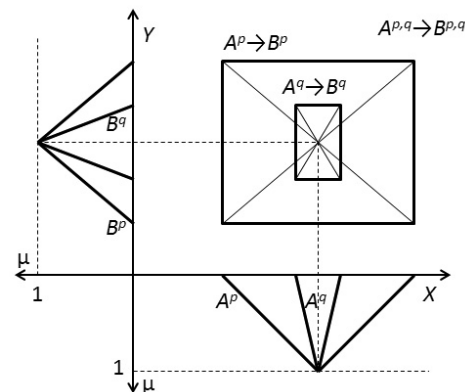
Egyenlőszárú háromszög alakú fuzzy halmazok esetén a referencia pont egyetlen pont, a halmaz magja:

$$\mu_{core(A)} = 1$$

Ez esetben az i^{th} szabály dupla fuzzy pont („double fuzzy dot”) alakja az alábbi formában adható meg:

$$A_i^{p,q} \rightarrow B_i^{p,q},$$

ami két egymásra illeszkedő fuzzy szabály egyesítéséből adódik: $A_i^p \rightarrow B_i^p$ és $A_i^q \rightarrow B_i^q$, ahol $\varphi(A_i^p) = B_i^p$ és $\varphi(A_i^q) = B_i^q$, valamint mindkét szabály antecedensének és konzekvensének megegyezik a referencia pontja: $core(A_i^p) = core(A_i^q)$, $core(B_i^p) = core(B_i^q)$, $i = 1, \dots, n$, de a fuzzióságuk mértéke eltérhet (lásd pl. 1. ábra).



1. ábra „Double fuzzy dot” szabályreprezentáció, abban az esetben ha a fuzzióság változásának iránya megegyezik az antecedens és a konzekvens oldalon.

5. A "DOUBLE-LINEAR" FUZZY INTERPOLÁCIÓ

A „double fuzzy dot” szabályreprezentáció alkalmazásának bemutatása érdekében a következőkben az egyszerűsített egydimenziós „double-linear” fuzzy szabály interpolációs módszer fuzzióság interpolációjának képességével való kiterjesztését vizsgáljuk.

A példa egyszerűsítése az alap „double-linear” fuzzy szabály interpolációs módszer háromszög és trapéz formájú fuzzy halmazokra korlátozásában áll. A vizsgált FRI módszer így elég egyszerűvé és jól vizualizálhatóvá válik annak érdekében, hogy a „double fuzzy dot” szabályreprezentáció lényegét bemutathassuk.

Az eredeti "double-linear" fuzzy szabály interpolációs módszert egydimenziós antecedens univerzumra és a valós számok alaphalmazán értelmezett konvex és normális fuzzy halmazokra vezette be Detyniecki *et al.* [10]. A szabályrendszer mindössze két szabályból áll: $A_1 \rightarrow B_1$ és $A_2 \rightarrow B_2$, ahol $\{A_1, A_2\} \in L^X, X \subseteq \mathfrak{R}$ és $\{B_1, B_2\} \in L^Y, Y \subseteq \mathfrak{R}$ fuzzy halmazok. A szabálybázis ritka, A_1 és A_2 diszjunktak, valamint $A_1 \leq A_2$, ahol \leq a precedencia reláció. A \leq precedencia reláció B_1, B_2 között is fennáll.

A háromszög alakú A_i, B_i fuzzy halmazoknak megfelelően módosítva [10] egyenleteit, az ugyancsak háromszög alakú A_* megfigyelésre a B_* következtetésre az alábbi lineáris összefüggések adódnak (lásd a 2. ábrát a jelölésekért):

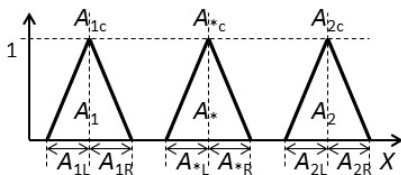
$$A_{*c} = \alpha \cdot A_{1c} + (1 - \alpha) \cdot A_{2c} \quad (1)$$

$$B_{*c} = \alpha \cdot B_{1c} + (1 - \alpha) \cdot B_{2c} \quad (2)$$

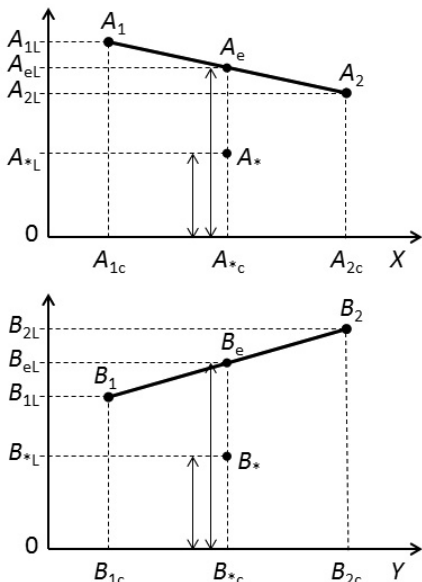
$$\alpha = \frac{A_{2c} - A_{*c}}{A_{2c} - A_{1c}} \quad (3)$$

$$B_{*c} = \frac{A_{2c} - A_{*c}}{A_{2c} - A_{1c}} \cdot B_{1c} + \frac{A_{*c} - A_{1c}}{A_{2c} - A_{1c}} \cdot B_{2c} \quad (4)$$

ahol B_{*c} a következtetés magja (referencia pontja).



2. ábra Az A_1, A_2 és A_* háromszög alakú fuzzy szabály antecedenseknél alkalmazott jelölésrendszer (a jelölés megegyezik a B_1, B_2 és B_* konzekvensenek esetében is).



3. ábra Az interpolált szabály $A_e \rightarrow B_e$ bal A_{eL}, B_{eL} bizonytalanságainak és a következménynek $A_* \rightarrow B_*$ a számítása.

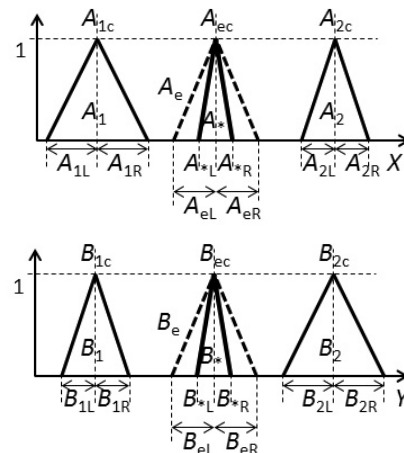
Az interpolált szabály $A_e \rightarrow B_e$ bal A_{eL}, B_{eL} (és jobb A_{eR}, B_{eR}) bizonytalanságaira az előző egyenletekhez hasonló lineáris összefüggések adódnak (lásd a 3. ábrát):

$$A_{eL} = \alpha \cdot A_{1L} + (1 - \alpha) \cdot A_{2L} \quad (5)$$

$$B_{eL} = \alpha \cdot B_{1L} + (1 - \alpha) \cdot B_{2L} \quad (6)$$

A következmény bal B_{*L} (és jobb B_{*R}) bizonytalansága a megfigyelés és az interpolált szabály relatív bizonytalansága alapján számítható, azaz a megfigyelés A_{*L} és az interpolált szabály antecedens A_{eL} relatív bizonytalansága egyezzen meg az interpolált szabály konzekvens B_{eL} és a konklúzió B_{*L} relatív bizonytalanságával (példát lásd a 3. ábrán):

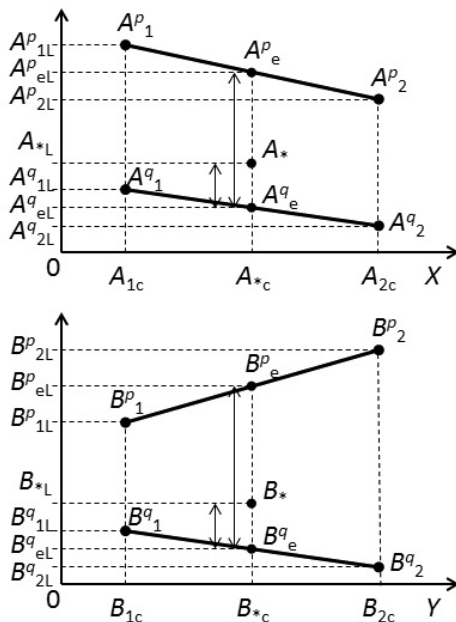
$$B_{*L} = \frac{A_{*L}}{A_{eL}} \cdot B_{eL} \quad (7)$$



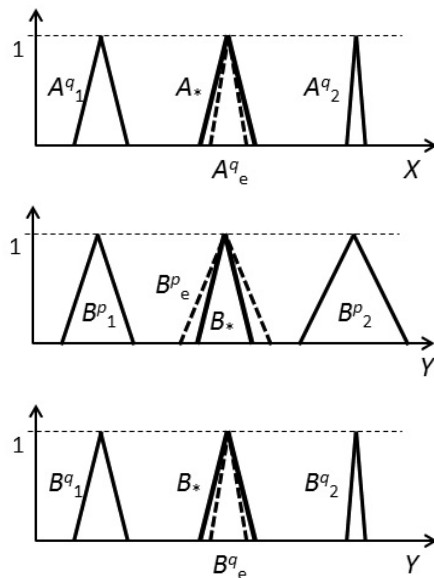
4. ábra Két fuzzy szabály $A_1 \rightarrow B_1, A_2 \rightarrow B_2$ „double-linear” interpolációja, az interpolált fuzzy szabály $A_e \rightarrow B_e$ és a konklúzió $A_* \rightarrow B_*$.

A "double-linear" fuzzy interpolációs módszer nagyon egyszerűen kiterjeszhető a "double fuzzy dot" szabályreprezentációra. Az alapkonceptió változatlan marad. Alapjaiban nem változik az (1) – (6) egyenlet, csak az interpolált fuzzy szabályt külön ki kell számolni a "double fuzzy dot" szabály $A^p_i \rightarrow B^p_i$ és $A^q_i \rightarrow B^q_i$ komponensére. Értelem szerűen így az interpolált szabályt is "double fuzzy dot" formában kapjuk $A^{pq}_e \rightarrow B^{pq}_e$ (példát lásd a 6. ábrán). A konklúzió bal B_{*L} (és jobb B_{*R}) bizonytalanságának számításához újból az eredeti módszert követjük (7), attól eltekintve, hogy itt most a relatív bizonytalanságot a megfigyelés és a két interpolált szabály között kell vizsgálnunk. Azaz a megfigyelés A_{*L} és a két interpolált szabály antecedens A^p_{eL} és A^q_{eL} relatív bizonytalansága egyezzen meg a két interpolált szabály konzekvens B^p_{eL}, B^q_{eL} és a konklúzió B_{*L} relatív bizonytalanságával (pl. lásd az 5-7. ábrát):

$$B_{*L} = \frac{A^p_{eL} - A_{*L}}{A^p_{eL} - A^q_{eL}} \cdot B^q_{eL} + \frac{A_{*L} - A^q_{eL}}{A^p_{eL} - A^q_{eL}} \cdot B^p_{eL} \quad (8)$$



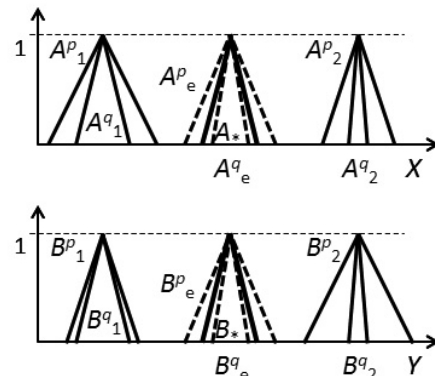
5. ábra Az interpolált szabály $A^{p,q}_e \rightarrow B^{p,q}_e$ bal $A^{p,q}_{eL}$, $B^{p,q}_{eL}$ bizonytalanságainak és a következménynek $A_* \rightarrow B_*$ a számítása "double fuzzy dot" szabályreprezentáció esetén.



6. ábra Két fuzzy szabály $A^{p,q}_1 \rightarrow B^{p,q}_1$, $A^{p,q}_2 \rightarrow B^{p,q}_2$ „double-linear” interpolációja és az interpolált fuzzy szabály $A^{p,q}_e \rightarrow B^{p,q}_e$ „double fuzzy dot” szabályreprezentáció esetén.

4. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutatómunka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.



7. ábra Az interpolált fuzzy szabály $A^{p,q}_e \rightarrow B^{p,q}_e$ és a konklúzió $A_* \rightarrow B_*$ „double fuzzy dot” szabályreprezentáció esetén.

6. IRODALOM

- [1] I. Perfilieva, “Fuzzy function as an approximate solution to a system of fuzzy relation equations”, Fuzzy Sets and Systems 147, 2004, pp. 363–383.
- [2] I. Perfilieva, et al., “Interpolation of fuzzy data: Analytical approach and overview”, Fuzzy Sets and Systems, 2010, doi:10.1016/j.fss.2010.08.005
- [3] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller”, Int. J. of Man Machine Studies, (7), 1975, pp.1-13.
- [4] P. Baranyi, L. T. Kóczy, and T. D. Gedeon, “A Generalized Concept for Fuzzy Rule Interpolation”, IEEE Trans. on Fuzzy Systems, vol. 12, No. 6, 2004, pp. 820-837.
- [5] L. T. Kóczy and K. Hirota, “Rule interpolation by α -level sets in fuzzy approximate reasoning”, In J. BUSEFAL, Automne, URA-CNRS. Vol. 46. Toulouse, France, 1991, pp. 115-123.
- [6] Zs. Cs. Johanyák, D. Tikk, Sz. Kovács, K. W. Wong “Fuzzy Rule Interpolation Matlab Toolbox – FRI Toolbox”, Proc. of the IEEE World Congress on Computational Intelligence (WCCI'06), 15th Int. Conf. on Fuzzy Systems (FUZZ-IEEE'06), July 16-21, Vancouver, BC, Canada, 2006, pp. 1427-1433.
- [7] Zs. Cs. Johanyák, “Sparse fuzzy model identification Matlab toolbox – RuleMaker toolbox”, Proceedings of IEEE 6th International Conference on Computational Cybernetics ICC3, Stara Lesná, Slovakia, 2008, pp. 69-74.
- [8] Zs. Cs. Johanyák: Fuzzy Rule Interpolation based on Subsethood Values, Proc. of 2010 IEEE International Conference on Systems Man, and Cybernetics (SMC 2010), 10-13 October 2010, pp. 2387-2393.
- [9] Zs. Cs. Johanyák, Sz. Kovács, “Fuzzy rule interpolation based on polar cuts”, Computational Intelligence, Theory and Applications, B. Reusch (Ed.), Springer, 2006 pp. 499-511.
- [10] M. Detyniecki, C. Marsala, and M. Rifqi, “Double-Linear Fuzzy Interpolation Method”, Proceedings of the 2011 IEEE International Conference On Fuzzy Systems, (FUZZ-IEEE 2011), Taipei, Taiwan June 27-30, 2011, pp. 455-462.
- [11] FRI Toolbox: <http://fri.gamf.hu/>

FUZZY SZABÁLY-INTERPOLÁCIÓS MÓDSZEREK BEÁGYAZOTT IRÁNYÍTÁSI ALKALMAZÁSOKBAN

FUZZY RULE INTERPOLATION IN EMBEDDED CONTROL APPLICATIONS

Kovács Szilveszter^{*}

ABSTRACT

The main goal of this paper is giving a short introduction to Fuzzy Rule Interpolation (FRI) methods together with a simple demonstration of their application benefits. There are relatively few FRI techniques used in practical fuzzy rule based applications. On one hand the FRI methods are not widely known, and many of them have limitations from practical application point of view, e.g. can be applied only in one dimensional case, or defined based on the two closest surrounding rules of the actual observation. On the other hand enabling the application of sparse rule bases the FRI methods can dramatically simplify methods of fuzzy rule base creation. These methods can save the expert from dealing with derivable rules and help to concentrate on cardinal actions only and hence simplify the rule base creation itself. In this paper, among the brief structure of several FRI methods, for demonstrating the benefits of the interpolation-based fuzzy reasoning as systematic approach, the construction of a fuzzy rule base through a simple example will be also discussed.

1. BEVEZETÉS

A napjainkban oly gyakran alkalmazott „számítási intelligencia” módszerek (Computational Intelligence methods) népszerűségének egyik lehetséges oka az, hogy olyan problémakörökre nyújtanak megoldást, melyek a klasszikus módszerekkel, a szükséges matematikai probléma modellek hiánya, vagy túlzott bonyolultsága miatt nehezen kezelhetők. Ugyanakkor, míg a számítási intelligencia módszerek általában nem igénylik a feladat matematikai modelljének ismeretét, magának az alkalmazott módszer struktúra bonyolultságának illeszkednie kell a megoldani szándékozott feladat bonyolultságához. Neurális hálózatok alkalmazása esetén ez például a szükséges neuronok számát és elrendezését, Fuzzy Logikai rendszerek esetén a minimálisan szükséges fuzzy halmazok és szabályok számát érinti. Mindemellett a klasszikus kompozíciós fuzzy következtetésen Zadeh

[1], Mamdani [2] alapuló fuzzy logikai algoritmusokban, a teljesen fedő szabálybázis kialakításának szükségessége miatt a megoldandó probléma bonyolultságából következő minimális szabályszámnál esetenként lényegesen több szabályt is meg kell határozni. Általánosságban ez m elemű antecedens fuzzy partícók és i számú antecedens (bemeneti) dimenzió esetén m^i darab szabály megadását jelenti. Azaz klasszikus fuzzy következtetési módszerek alkalmazása esetén a szabálybázis mérete a bemeneti dimenziók számának növekedésével exponenciálisan nő. Erre a praktikus okra vezethető vissza az, hogy a gyakorlati fuzzy logikai rendszer alkalmazásban (pl. irányítástechnika) a bemeneti változók száma a legtöbb esetben nem haladja meg a 4-5 változót. Ennél magasabb bemeneti dimenziószám esetén a szabálybázis mérete általában már túlságosan nehezen kezelhető.

2. FUZZY SZABÁLY-INTERPOLÁCIÓ

Koncepcionálisan más megoldást jelenthet olyan fuzzy logikai következtetési módszerek alkalmazása (fuzzy szabály-interpolációs módszerek, Fuzzy Rule Interpolation (FRI) methods), melyek a klasszikus fuzzy következtetési módszerekkel ellentétben, abban az esetben is képesek használható (interpolált) fuzzy következményt generálni, ha a megfigyelés az illető szabályrendszer egyik szabály antecedensére sem illeszkedik. Az alkalmazott fuzzy szabálybázis „fedő” volta, illetve „teljessége” így már nem követelmény, a fuzzy szabály-interpolációs következtetés esetén a kiadódó (nem lényegi) szabályok szándékosan elhagyhatóak a szabálybázis építéskor (szándékosan „ritka” fuzzy szabálybázis építése). A lényegtelen szabályok elhagyásával jelentős mértékben csökkenthető a fuzzy modellt leíró szabálysám. Így a fuzzy szabály-interpolációs módszerek alkalmazásával egyrészt egyszerűsíthető a szabályrendszer megalkotása (elég csak a modell szempontjából lényeges szabályokat megadni), másrészt csökken a következtetés számítási igényessége. Ugyanakkor a legtöbb ismert fuzzy szabály-interpolációs módszer a gyakorlati

^{*} egyetemi docens, Miskolci Egyetem, ALT

esetekben csak nehezen alkalmazható. Gyakori a módszerek magas számításigényessége, pl. többdimenziós bemeneti univerzum esetén meg kell keresni a megfigyeléshez legközelebbi két közrefogó szabályt, illetve több jellemző α -vágatra számolt eredményből tevődik össze a fuzzy következmény.

Több fuzzy szabály-interpolációs módszer esetén a kapott fuzzy következmény nem minden esetben értelmezhető fuzzy halmazként (lásd pl. [6]). Számos tanulmány foglalkozik az interpolált fuzzy következmény halmaz értelmezhetőségének problémájával pl. [7]. A gyakorlati alkalmazások szempontjából lényeges probléma lehet még az is, hogy számos fuzzy szabály-interpolációs módszert eredetileg csak egydimenziós bemeneti univerzumra definiáltak. Ezen módszerek közül néhánynak a későbbiekben született több bemeneti dimenziót is kezelni képes változata pl. [9]. K. W. Wong *et al.* [5] összevet néhány ismertebb több bemeneti dimenziót is kezelni képes fuzzy szabály-interpolációs módszert. Számos újabb keletű szabály-interpolációs módszer esetében a gyakorlati alkalmazások szempontjából leglényegesebb probléma továbbra is a módszer magas számításigényessége. Sok módszer esetében a következmény fuzzy halmaz számításához meg kell keresni a fuzzy megfigyeléshez legközelebbi két közrefogó antecedensű szabályt, mely többdimenziós bemeneti univerzum esetén önmagában is magas számításigényű probléma. Néhány szabály-interpolációs módszer pl. [9], [10] úgy oldja meg ezt a problémát, hogy a következmény halmaz számításakor, a megfigyelés és a szabály antecedensek távolságával súlyozottan a szabálybázis valamennyi szabályát figyelembe veszi.

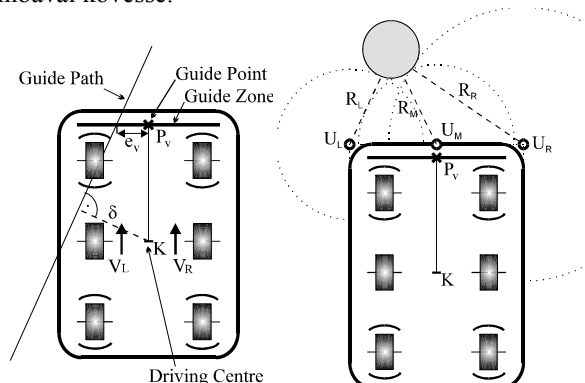
A számításigény szempontjából további problémát jelenthet a következmény fuzzy halmaz volta. Azon szabály-interpolációs módszerek esetében, ahol a következmény fuzzy halmaz formájában adódik, ez vagy számos ismételt interpolációs lépést (pl. a releváns α -vágatonként [3], vagy egy másodlagos, a következmény fuzzy halmaz alakját meghatározó lépést jelent pl. [8]. A közvetlen gyakorlati irányítástechnikai alkalmazhatóság szempontjait szem előtt tartva került kialakításra a „FIVE” fuzzy szabály-interpolációs módszer [4]. A módszer többdimenziós megfigyelés térben, a szabályszámmal arányos idő alatt egyetlen interpolációs lépéssel hoz nem fuzzy következtetést.

2. FUZZY INTERPOLÁCIÓS SZABÁLYRENDSZER TERVEZÉS

A bemutatni kívánt fuzzy interpolációs (FRI) szabálybázis tervezés példa egy vezetőlíni targonca (automated guided vehicle - AGV) nyomkövetési és ütközés elkerülési stratégiájának szabály alapú leírása [13]. Az FRI módszereket a mérnöki modellezés számos

más területén is sikerrel alkalmazták pl. szennyvíztisztító reaktor [11], vagy szerszám élettartam modellezés [12].

A mintának szánt AGV kormányzás beágyazott FRI példában a szabályokkal leírni kívánt stratégiának két célja van. Ezek a nyomkövetés (egy előre kijelölt vezetőlíni követése) és az ütközés elkerülés. A vezetőlíni követésénél a nyomkövető rendszer valamely vezetőlíni hangolt rendszerrel (pl. érzékelő sáv) érzékeli a vezetőlíni pozícióját. A nyomkövető kormányzási stratégia célja, hogy a jármű útja során a vezetőlíniot minél kisebb nyomkövetési hibával kövesse.



1. ábra Differenciális kormányzású AGV, nyomvonal érzékelő sávval. δ a nyomkövetési hiba, e_v a nyomvonal és a nyomkövetési referenciapont (P_v) távolsága, K a kormányzási középpont. R_L , R_R , R_M az U_L , U_R , U_M akadályérzékelők által mért távolságok.

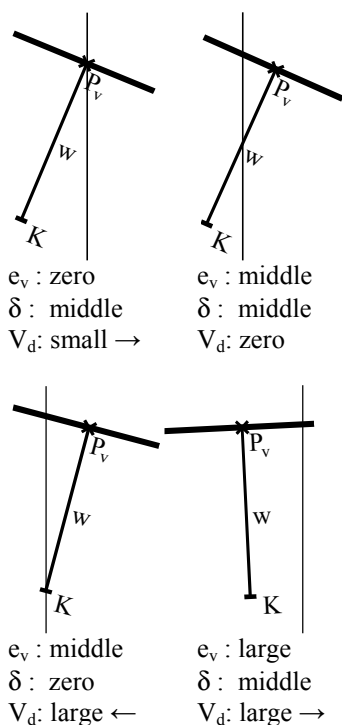
3. A NYOMKÖVETÉS SZABÁLYBÁZIS

A kormányzási stratégia leírása két részre bontható. Az első a nyomkövetési szabálybázis, a második az akadályelkerülés. A szabálybázis megalkotásának egyik legegyszerűbb módja a szakértői szabályok rögzítése. Össze kell gyűjteni azokat a szabályokat, melyek a legmeghatározóbbak a nyomkövetési kormányzás során. Ezek a szabályok alkotják majd a nyomkövetési szabálybázist. FRI alkalmazáspéldáról lévén szó, elég összegyűjteni a legjellemzőbb szabályokat.

A szabályok nagyon egyszerűek. A járműnek úgy kell haladnia, hogy a K a kormányzási középpontja minél közelebb kerüljön a nyomvonalhoz, majd ha ez megtörténik, akkor a helyes irányba kell fordulnia.

Ezen stratégia leírásához két megfigyelésre van szükség, a (δ) a nyomkövetési hibára és a nyomvonal és a nyomkövetési referenciapont távolságára (e_v) (a jelölést lásd az 1. ábrán). A megfigyelések alapján hozza meg a kormányzási stratégia a kormányzás mértékét illető döntését (V_d).

A szakértői tudást összegyűjtve a kormányzási stratégiát mindössze négy lényeges helyzet és kormányzási döntés határozza meg (lásd a 2. ábrán).



2. ábra A kormányzási stratégiát meghatározó lényeges helyzetek és kormányzási döntések

R_{V_d} :	e_v	δ	V_d
Rule 1:	NL		PL
Rule 2:	PL		NL
Rule 3:	NM	Z	PL
Rule 4:	PM	Z	NL
Rule 5:	NM	PM	Z
Rule 6:	PM	NM	Z
Rule 7:	Z	PM	NS
Rule 8:	Z	NM	PS

1. táblázat Nyomkövetési szabályrendszer (1), ritka (FRI) szabálybázis.

Két megfigyelés alapján kell következtetést hozni, így az i . fuzzy szabály alakja a következő:

$$R_{V_d(i)} : \text{If } e_v=A_{1,i} \text{ And } \delta=A_{2,i} \text{ Then } V_d=B_i. \quad (1)$$

Amennyiben elkészítjük az antecedens és konzekvens univerzumok fuzzy partícióit, és meghatározzuk a nyelvi elemeket mindkét antecedens univerzumra (negatív nagy (NL), negatív közepes (NM), pozitív közepes (PM), pozitív nagy (PL)), valamint a konzekvens univerzumra (negatív, pozitív nagy (L), kicsi (S) és zérus (Z)), úgy alkalmas FRI következtetést választva rögtön egy működő rendszert kapunk. Megfelelő teljesítménymérték és paraméterbecslő eljárással kiegészítve a modellt, a szabálybázis paraméterek a rendszer futása közben tovább hangolhatók. A 2. ábrán bemutatott szakértői tudás alapján a kormányzási stratégiára az 1. táblázatban

összefoglalt szabálybázis adódik.

3. A NYOMKÖVETÉS ÉS ÜTKÖZÉS ELKERÜLÉS

A nyomkövetési stratégia ütközéssel való kiegészítésének érdekében első lépésként meg kell vizsgálni, hogy milyen új megfigyelésekre van még szükség. Az ütközésselkerülési stratégia szempontjából két ütközési helyzet különböztethető meg. A frontális ütközés és az oldalsó ütközés. Az 1. ábrának megfelelően a jármű három távolságérzékelő szenzorral van felszerelve, egy a jármű elején középen (U_M), egy elöl a bal- (U_L) és egy a jobboldalon (U_R). A mért akadály távolság értékek rendre R_M , R_L , R_R elégségesek a frontális ütközési helyzet felismerésére. Az oldal ütközés helyzetek felismerése közvetlenül nem lehetséges. A megelőző akadály távolságok és a mérési pozíciók rögzítése esetén azonban álló akadályok esetén az oldalsó akadályok helyzete is becsülhető [13], [14].

Az ütközésselkerülés érdekében ezért öt új megfigyelésünk lesz, a jármű elejére szerelt távolságérzékelők mérési eredményei (R_L , R_M , R_R) és a számított becsült maximális bal és jobb biztonságos fordulási szög (α_{ML} , α_{MR}). Mindösszesen így a nyomkövetési és akadályelkerülési stratégiának 7 megfigyelés alapján kell meghoznia a kormányzást illető döntését.

A nyomkövetési és akadályelkerülési stratégia szabálybázisának szabályai így a következő formában írhatók fel $R_{V_d(i)}$:

$$\text{If } e_v=A_{1,i} \text{ And } \delta=A_{2,i} \text{ And } R_L=A_{3,i} \text{ And } R_R=A_{4,i} \text{ And } R_M=A_{5,i} \text{ And } \alpha_{ML}=A_{6,i} \text{ And } \alpha_{MR}=A_{7,i} \text{ Then } V_d=B_i. \quad (2)$$

R_{V_d} :	e_v	δ	R_L	R_R	R_M	α_{ML}	α_{MR}	V_d
Rule 1:	NL							PL
Rule 2:	PL							NL
Rule 3:	NM	Z					L	PL
Rule 4:	PM	Z				L		NL
Rule 5:	NM	PM	L		L	L		Z
Rule 6:	PM	NM		L	L		L	Z
Rule 7:	Z	PM	L		L	L		NS
Rule 8:	Z	NM		L	L		L	PS
Rule 9:	Z	PM	S		S			PL
Rule 10:	Z	NM		S	S			NL
Rule 11:	Z	Z	L	S	S			NL
Rule 12:	Z	Z	S	L	S			PL

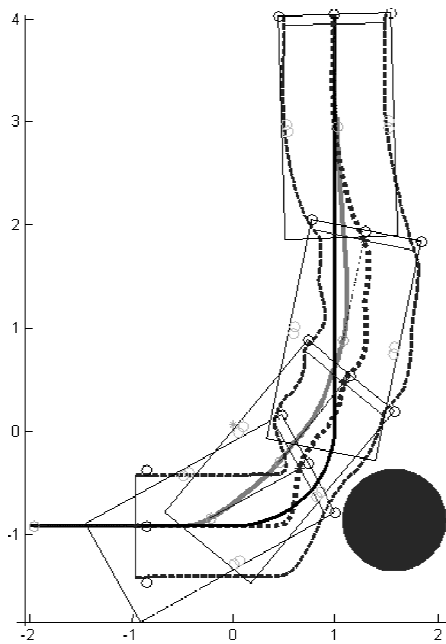
2. táblázat Teljes nyomkövetési és akadályelkerülési szabályrendszer (2).

A következő lépés a nyomkövetési szabályrendszer (1. táblázat) ütközésselkerüléssel való kiegészítése. A kiegészítés két lépésben történhet. Az első a

nyomkövetési szabályok módosítása úgy, hogy a nyomkövetési szabályok csak akkor jussanak érvényre, ha a végrehajtásuk nem okoz ütközési helyzetet. Ez a szabályok kiegészítését jelenti úgy, hogy az antecedensek között megjelenik feltételként az is, hogy a kívánt elmozdulás irányában nincs akadály (távolsága: L). Az így kiegészített szabályokat a 2. táblázat Rule 1-8 szabályai foglalják össze.

A következő lépés a szabálybázis bővítése olyan szabályokkal, amik az ütközési helyzeteket közvetlenül kezelik. A példa esetében ez újabb négy szabály felvételét jelent (Rule 9-12). A teljes nyomkövetési és akadályelkerülési szabályrendszert a 2. táblázat foglalja össze.

A fuzzy partíciók hangolását követő futási eredményekre mutat be példát a 3. ábra.



3. ábra Szimulált nyomvonal a 3. táblázat szerinti nyomkövetési és akadályelkerülési stratégia esetén.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutatómunka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

6. IRODALOM

- [1] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes", IEEE Trans. on SMC, (3), 1973, pp.28-44.
- [2] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller", Int. J. of Man Machine Studies, (7), 1975, pp.1-13.

- [3] L. T. Kóczy and K. Hirota, "Rule interpolation by α -level sets in fuzzy approximate reasoning", In J. BUSEFAL, Automne, URA-CNRS. Vol. 46. Toulouse, France, 1991, pp. 115-123.
- [4] Sz. Kovács, "Extending the Fuzzy Rule Interpolation "FIVE" by Fuzzy Observation", Advances in Soft Computing, Computational Intelligence, Theory and Applications, Bernd Reusch (Ed.), Springer Germany, 2006, pp. 485-497.
- [5] K. W. Wong, D. Tikk, T. D. Gedeon, and L. T. Kóczy, "Fuzzy Rule Interpolation for Multidimensional Input Spaces With Applications", IEEE Transactions on Fuzzy Systems, Vol. 13, No. 6, December, 2005, pp. 809-819.
- [6] L. T. Kóczy and Sz. Kovács, "On the preservation of the convexity and piecewise linearity in linear fuzzy rule interpolation", Tokyo Inst. Technol., Yokohama, Japan, Tech. Rep. TR 93-94/402, LIFE Chair Fuzzy Theory, 1993.
- [7] D. Tikk and P. Baranyi, "Comprehensive analysis of a new fuzzy rule interpolation method", In IEEE Trans. Fuzzy Syst., vol. 8, No. 3, June, 2000, pp. 281-296.
- [8] P. Baranyi, L. T. Kóczy, and Gedeon, T. D., "A Generalized Concept for Fuzzy Rule Interpolation", IEEE Trans. on Fuzzy Systems, vol. 12, No. 6, 2004, pp. 820-837.
- [9] S. Jenei, E. P. Klement and R. Konzel, "Interpolation and extrapolation of fuzzy quantities – The multiple-dimensional case", Soft Comput., vol. 6., 2002, 258-270.
- [10] Zs. Cs. Johanyák, Sz. Kovács, "Fuzzy Rule Interpolation Based on Polar Cuts", Computational Intelligence, Theory and Applications, Springer Berlin Heidelberg, 2006, pp. 499-511.
- [11] Zs. Cs. Johanyák, R. Parthiban, and G. Sekaran, "Fuzzy Modeling for an Anaerobic Tapered Fluidized Bed Reactor", SCIENTIFIC BULLETIN of "Politehnica" University of Timisoara, ROMANIA, Transactions on AUTOMATIC CONTROL and COMPUTER SCIENCE, ISSN 1224-600X, Vol:52(66) No:2, 2007, pp. 67-72.
- [12] Zs. Cs. Johanyák and A. Szabó, "Tool life modelling using RBE-DSS method and LESFRI inference mechanism", A GAMF Közleményei, Kecskemét, XXII. 2008, pp. 5-16.
- [13] Sz. Kovács and L. T. Kóczy, "Application of an approximate fuzzy logic controller in an AGV steering system, path tracking and collision avoidance strategy", Fuzzy Set Theory and Applications, In Tatra Mountains Mathematical Publications, Mathematical Institute Slovak Academy of Sciences, vol.16, Bratislava, Slovakia, 1999, pp. 456-467.
- [14] Sz. Kovács: Fuzzy Rule Interpolation from a Practical Point of View, Acta Universitas Jaurinensis, Series Intelligentia Computatorica, Vol. 1, No. 3, ISSN 1789-6932, 2008, pp. 595-611.

HATÉKONY TECHNIKÁK ÉS MEGOLDÁSOK A SZOFTVERES RASZTERIZÁCIÓBAN

EFFICIENT TECHNIQUES AND METHODS FOR SOFTWARE RENDERING

Mileff Péter, Dudra Judit***

ABSTRACT

Computer visualization, under continuous changes has arrived again to a major milestone. Necessary modifications are required at the currently applied physical devices and technologies. This publication presents an overview about new performance improvement methods, what today CPUs can provide utilizing their modern instruction sets and how these methods can be applied in the specific field of the computer graphics, called software rendering. Furthermore the paper focuses on the GPGPU based parallel computing as a new technology for computer graphics where a TBR based software rasterization method is investigated in terms of performance and efficiency.

1. BEVEZETÉS

A számítógépes grafika területe sok éves fejlődésen ment keresztül az utóbbi néhány évtizedben. Legfontosabb állomása a grafikus célprocesszorok megjelenése volt, amely számtalan új lehetőséget nyitott meg a fejlesztők előtt.

Bár kezdetben a kártyák grafikus csővezetéke és programozhatósága nagyon egyszerű modellt nyújtott, az addig érvényben lévő szoftveres raszterizáció gyorsan elvesztette jelentőségét, mert az akkori CPU-k nem voltak képesek versenyezni a célhardverek teljesítményével. A gyártók és az ipar szemszögéből elsősorban a gyorsaság került előtérbe a robusztuság és programozás rugalmassággal szemben.

Az utóbbi években a videokártyák technológia fejlődése ezért főként a csővezeték programozhatóságának növelésére irányult. Ma a fejlődés pedig már egészen kiforrot irányba halad tovább, bevezetve a grafikus processzorok egy teljesen új generációját, az általános célú grafikus processzorokat, melyek már nem csupán a megjelenítés gyorsítására alkalmasak, hanem a CPU-hoz hasonlóan általános számítások végzésére is.

Mindezek ellenére ki kell emelni a GPU alapú számítógépes raszterizáció problémáit. A programozási

logikája és modellje lassan eléri határait, a fejlődés intenzitása láthatóan csökken. Hiába áll rendelkezésre gyors hardveres támogatású csővezeték, az évek során a hardver korlátaihoz igazított csővezeték nem ad kellő rugalmasságot a programozónak. A GPU egység általános célú programozása pedig a memória modell és a párhuzamos szálak futtatását végző fix funkciók blokkok miatt limitált [1]. Jól mutatja ez a mai számítógépes játékok homogén és kiszámítható „kinézete” és viselkedése. A mai GPU architektúra tehát megkérdőjelezhető és átalakításra szorul, melyet a vezető ipari felhasználók is erősen szorgalmaznak [2,15].

Mindezek mellett az elmúlt években a számítógépek központi egysége hatalmas fejlődésen ment keresztül. A processzor gyártók kibővített utasításkészlettel reagáltak a piaci igényekre, lehetővé téve a CPU-kon is a gyorsabb és főként vektorizált (SIMD) feldolgozást.

Az új technológiáknak köszönhetően az utasításkészletet megfelelően kihasználó szoftver akár 2-10x sebességnövekedést is elérhet. Mindehhez társítva a GPGPU technológiát felmerül a kérdés, hogy miért ne lehetne készíteni egy teljesen szoftveres megvalósítású grafikus csővezetékét. Jelen cikk központi témája tehát annak a vizsgálata, hogy miként lehetséges egy a mai központi egységekben rejlő technológiai előnyöket kihasználó, előre mutató szoftveres megjelenítőt készíteni.

2. IRODALMI ÁTTEKINTÉS

A szoftveres képszintézis az első számítógépek óta jelen van domináns szereppel bírva egészen 2003-ig, a GPU megjelenéséig. A korai évek során született raszterizálók közül legkimagaslóbb eredmény az ID software által készített Quake I, II szoftveres renderer (1996), amely az első MMX utasításkészletre optimalizált valós háromdimenziós motor volt. A később született eredmények közül főleg az Unreal motort (1998) lehet kiemelni, amely szintén nagyon gazdag funkcionalitással rendelkezett.

* adjunktus, Miskolci Egyetem, Ált. Inf. T.

**tudományos munkatárs, Bay Zoltán Alkalmazott Kutatási Közhasznú Nonprofit Kft., Miskolc

A GPU renderelés folyamatos térnyerése után a szoftveres megjelenítés egyre inkább háttérbe szorult. Ennek ellenére született néhány nagyszerű eredmény, mint a Rad Game Tools által fejlesztett Pixomatic 1, 2, 3 [3] és a TrasGaming által készített Swiftshader [2]. Mindkét termék nagyon komplex, magas szinten optimalizált raszterizációt tesz lehetővé.

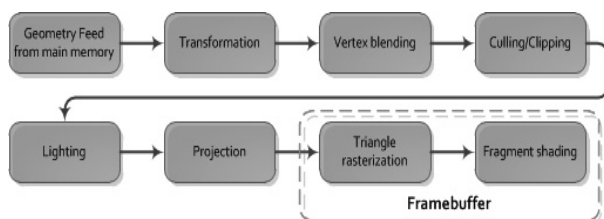
Bár a Microsoft a DirectX fejlesztésével nyújtott alapot a GPU technológiák terjedésének, mindezek mellett kifejlesztette saját szoftveres megjelenítőjét a WARP-ot [1]. A renderer jól skálázható több szála egyaránt és teljesítményben felveszi a versenyt az alacsony kategóriás integrált grafikus kártyákkal.

A problémákat és az igényeket jól felmérve az Intel 2008-ban egy hibrid, x86 alapú videokártya (Larrabee) fejlesztését tűzte ki célul [9], melynek célja egy teljesen programozható csővezeték kidolgozása volt [4].

Loop and Eisenacher [2009] parametrikus felületek számára készített GPU alapú szoftveres megjelenítőt. Az NVidia pedig saját GPGPU CUDA platformjának segítségével készített tanulmányt [6]. Napjaink vezető grafikai színvonalú számítógépes játéka [5] új technológiájaként egy részben SPU alapú raszterizációt valósított meg a fényforrás hatékonyan kezelésére. Az [1] publikációban a szerző egy több szálat használó modern „Tile” alapú megjelenítő technikát vázol.

3. SZOFTVERES RASZTERIZÁCIÓ

Szoftveres raszterizációnak nevezzük azt a képalkotási folyamatot, amikor a képszintézis teljes folyamatát a számítógép központi egysége (CPU) végzi. Az alakzatokat felépítő geometriai primitívek a központi memóriában helyezkednek el. A képalkotás logikája egyszerű: a központi egység elvégzi a szükséges műveleteket a tárolt adatokon, ennek eredményét eltárolja egy *framebuffer*-ben, majd az elkészített képet kiküldi a videó vezérlőnek. A következő ábra ennek általános csővezetékét mutatja be.



1. ábra. Általános grafikus csővezeték

Az 1. ábrán bemutatott csővezetékben a számítások két domináns csoportja alakul ki egy képszintézis során. Az első csoportba főként *vertex* transzformációs műveletek tartoznak, melyek egészen a *Framebuffer* műveletekig tartanak. Ezekben a fázisokban számos mátrix és vektortranszformációval előkészítjük a pixel szintű raszterizációt. A második csoportba olyan

pixelenkénti operációk tartoznak, mint a háromszögek diszkrétizálása és a pixelek árnyalása. A grafikus motorok számára a megjelenítés szempontjából e két csoport, de főként a második a leginkább számításigényes feladat.

4. ÁLTALÁNOS GYORSÍTÁSI LEHETŐSÉGEK

A mai modern processzor architektúrájának köszönhetően számos lehetőség áll rendelkezésre a grafikus csővezeték számításigényes részeinek gyorsítására. Az igazán jó eredmény elérése érdekében természetesen ezek együttes alkalmazása szükséges, azonban jelen cikkben terjedelmi okok miatt csak néhány megközelítést emelünk ki.

4.1 SSE alapú csővezeték optimalizálás

Az utóbbi évek során született legfontosabb újítás a bevezetett SIMD processzor utasításkészletek köré csoportosul (Intel – SSE család, AMD - 3DNow, Apple - Altivec, ARM - NEON). Segítségükkel lehetővé válik a számítások vektorizált gyorsítása, melyekkel akár többszörös sebességnövekedés is elérhető. Fontos szempont azonban a programozhatóság és a hordozhatóság kérdése is. Az SSE család mára már széles körben elfogadott, az utasításkészlet alapú programozást pedig a mai fordítók (GCC, Intel) már jól támogatják. A modern fordítók több lehetőséget is biztosítanak erre. Egyrészt Assembly kódként implementálni, amely mély programozói ismereteket igényel és a programkód nem lesz mindig hordozható, vagy használni a fordítók magasabb szintű *Intrinsics* függvénykönyvtárát. A második lehetőség (pl. GCC: `__m128 z = _mm_setzero_ps();`) már elég magas szintű, kényelmes programozást tesz lehetővé és nem korlátozza a hordozhatóságot sem.

4.1.1 SSE alapú vektor optimalizálás

Az SSE (Streaming SIMD Extensions) az Intel által kidolgozott SIMD instrukciókészlet család (jelenleg SSE4.2) az x86 architektúrák számára. Legfőbb újítása a 8 darab, külön-külön 128 bit hosszúságú SSE vektor (`xmm0 ~ xmm7`). A kibővített utasításkészlet pedig lehetőséget nyújt arra, hogy két vektoron elvégzett műveletet a processzor a vektor adatain párhuzamosan hajtsa végre.

A grafikus csővezeték első műveleti csoportja jellemzően valamilyen nagyobb adathalmazon elvégzendő vektortranszformáció. Háromdimenziós megjelenítés esetén tehát az SSE instrukciókészlet alkalmazása lehetővé teszi, hogy a 128 bites SSE vektorban 4 darab különböző 32 bites floating point számot tároljunk, mégpedig *x, y, z,* és *w*-t. Ez azt jelenti,

hogy ezen a 4 számon egyszerre tudunk számításokat végezni, amely jelentős sebességnövekedést eredményez a csővezeték transzformációs műveleteinek végrehajtásában.

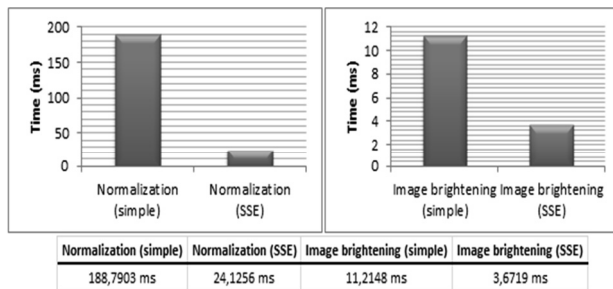
4.1.2 SSE alapú képfeldolgozás

Az SSE család szintén sikeresen alkalmazható a csővezeték raszterizációs fázisában, mert a pixeleken végzett műveletek nagy része független egymástól, így párhuzamosan egyszerre végrehajtható. A mai grafikus motorok tipikusan 32 bites (RGBA) színkomponensű megjelenítést használnak, ahol minden komponens 4 byte hosszúságú. Ez a fajta leképzés jól illeszkedik az SSE alapú megközelítéshez, mert egy 128 bites regiszterben 4 darab különböző pixel színét tudjuk tárolni és műveletet végezni rajta párhuzamosan.

4.1.3 SSE teszteredmények

A továbbiakban két tesztfeladaton keresztül bemutatjuk az SSE megoldás hatékonyságát. Az első feladatban egy minden grafikai motorban megtalálható számítást, a vektor normalizációt vizsgáltuk. A formula számításgényes, mert gyökvonást és osztást is tartalmaz. A feladatban 50.000 vektor normalizációját ismételjük 200.000-szer.

A második teszt számára egy mindennapi pixelszintű képfeldolgozási feladatot választottunk. Példánkban egy 32 bites 1024x768 felbontású képen egy világosítási transzformációt végzünk el 1000-szer. A programok elkészítéséhez az SSE2 utasításkészletet, a C++ nyelvet és a GCC 4.4.1 fordítót használtuk, a méréseket pedig egy Core i7 870-es 2.93GHz CPU-val végeztük. Az alábbi táblázat tartalmazza mindkét feladat eredményeit:



2. ábra. Számítási eredmények összehasonlítása

Az eredmények kiértékelése jól mutatja az SSE alapú programozás erősségét. Míg az első esetben ez 7.8 szoros, addig a második teszt 3.05 szörös sebességnövekedést jelent a hagyományos kódhoz képest.

4.2 Adatok igazítása

A grafikus csővezeték, így a raszterizáció sebessége tovább gyorsítható, ha az adatokat a memóriában megfelelően igazítva tároljuk. Az igazítás azt jelenti, hogy az adat címe milyen értékkel (1,2,4,8) osztható, reprezentálva ezzel az igazítás byte hossz értékét. A CPU nem byte-onként olvassa vagy írja a memóriát, hanem 2, 4, 8, 16, vagy 32 byte-os darabokban. Ezért ha a grafikus csővezeték összefüggő adathalmazait nem igazítjuk megfelelően 4, 8, vagy 16 byte-os határra, akkor ez súlyos sebességcsökkentést idézhet elő, mert a CPU-nak extra műveletekre van szüksége az adatok betöltésekor [6].

Az igazítási probléma a csővezeték struktúrái esetén viszonylag egyszerűen orvosolható az alacsonyabb szintű nyelvekben (pl. C, C++, D). Az igazítást mindig a leginkább korlátozó tagra kell alapozni, amely általában a legnagyobb valós típus. Ezért egy struktúra adatait tárolási méretük alapján csökkenő sorrendbe kell rendezni, hogy igazított memóriaterületű struktúrához jussunk. Nagyobb struktúratömbök esetén így nemcsak memóriát takarítunk meg, hanem jelentősen növelhetjük a raszterizáció hatékonyságát.

5. GPGPU ALAPÚ SZOFTVERES CSŐVEZETÉK

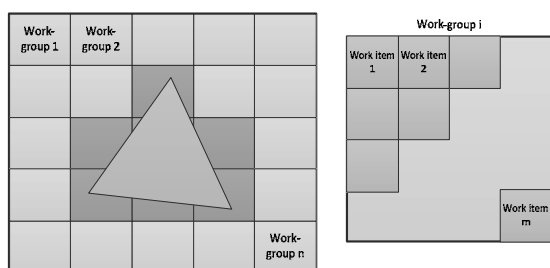
A GPU alapú megjelenítési technológia napjainkban az általános célú feldolgozás irányába halad tovább. A legújabb GPGPU kártyák hatalmas számítási kapacitást rejtenek (~1 Tflops/s) magukban köszönhetően a bennük rejlő egyre növekvő processzorok számának (pl. NVidia GTX 480, 480 mag), a gyors elérésű memóriának (GDDR5), és a fejlett technológiának.

Logikailag a GPU általános célú lehetőségei a grafikus csővezeték bármely szakaszában használhatók bizonyos korlátozásokkal. Mégis mivel a raszterizáció lényegesen számításgényesebb feladat, így célszerű, hogy a projekciós fázistól kezdve a GPU végezze el a feladatokat kihasználva a GPU párhuzamosítási erősségeit.

5.1 GPGPU alapú képszintézis

A raszterizáció fázisának feladata a megjelenítendő alakzatok háromszög halmazainak pixelekre való leképzése. A folyamat GPGPU-val való ideális megoldáshoz figyelembe kell venni az eszköz jellegzetességeit és programozás nyelvi (OpenCL, CUDA) lehetőségeit. A programozási modell legkisebb egysége a végrehajtandó kódot (kernel) futtató work-item, melyek work-group-okba csoportosulnak. Minden work-group külön processzoron fut szeparáltan, az egy csoportba tartozó work-item-ek pedig azonos számítási egységen. Ezen jellegzetességekhez logikailag legközelebbi a „Tile” alapú megjelenítés áll

legközelebb. A következő ábra a TBR renderelési megoldást szemlélteti a GPGPU szemszögéből.



3. ábra. GPGPU alapú ideális képszintézis

A megoldás ötlete, hogy a *framebuffert* felosztjuk azonos méretű területekre. A párhuzamos végrehajtás kihasználása végett minden területet egy külön work-group-hoz célszerű rendelni. A rasterizáció során a csővezeték minden háromszöget egy-egy területhez rendeljük az alapján, hogy átfedi-e vagy sem. Minden terület egymástól függetlenül, párhuzamosan kerül feldolgozásra külön processzoron, ahol a pixelenkénti rasterizációt pedig a work-item-ek végzik el. A framebuffer felosztásánál célszerű tipikusan 16x16 vagy 32x32 méretű részeket választani. A felbontás ekkor ideális a mai videokártyák hardveres jellemzőik alapján.

A csoporton belül egy „Tile” képének előállításáért a work-item-ek felelősek. Sorra veszik a csoporthoz tartozó háromszögeket, kiszámítják azok határait, és elvégzik a rasterizációt. Egy csoporton belül a work-item-ek szintén párhuzamosan futnak és azonos lokális memóriaterület használnak. Minden item egy külön háromszöget dolgoz fel. A háromszögek takarási problémáinak megoldására az itemek közötti szinkronizáció nyújt lehetőséget. A teljes kép akkor készül el, amikor mindegyik csoport elvégezte a saját feladatát.

5.2 Megszorítások

A GPU és a központi memória távol van egymástól és így az adatmozgatás a GPU és központi memória között a PCIe bus transzfer sebessége által erősen limitált (~2.4 GB/s). Célszerű tehát a csővezeték háromszöghalmazának minden adatát a kártyamemóriában tárolni és minimalizálni az adatok mozgatását. Rövid tesztként egy központi memóriában tárolt 1024x768 méretű 32 bites üres *framebuffert* osztottunk meg a GPU számára. Egyéb számítás nem végzünk, csak az adatok megosztását és a *framebuffer* megjelenítését. A tesztben felhasznált hardver: ATI Radeon HD 5670 1 GB RAM. A csővezeték GPU megosztás nélküli változatában az üres *framebuffer* rajzolási sebessége ~1100 FPS, míg a GPU megosztás következményeképpen ez rögtön ~620 FPS-re redukálódott.

További probléma, hogy egy kernel futtatása egy meghatározott előkészítési időt igényel. Második tesztként megvizsgáltuk, hogy hogyan befolyásolja a kernel inicializálása a futási sebességet. A tesztben egy üres kernelt hoztunk létre, amelynek 4 darab float típusú változót osztottunk meg. A sebesség ezúttal ~580 FPS-re redukálódott. Természetesen vannak lehetőségek a kommunikációból fakadó sebességvesztés javítására. Például ha az egész framebuffer-t a kártya memóriájában tároljuk OpenGL textúráként. Ezzel azonban elveszítjük a csővezeték szoftveres jellegének egy részét.

4. ÖSSZEFOGLALÁS

A cikkben bemutatott technikák rámutatnak arra, hogy egy megfelelően gyors szoftveres rasterizáló készítése sok erőfeszítést igényel. Nélkülözhetetlen a több technológia együttes alkalmazása valamint alacsonyabb szintű nyelvek használata. A központi egység sokat fejlődött az elmúlt években, lehetőségeinek megfelelő kihasználásával a szoftveres csővezeték sebessége sokszorosára gyorsítható. Bemutattuk, hogy miként alkalmazható a GPGPU technológia új megközelítésként a rasterizációs fázisban. A párhuzamosítási lehetőségei bizonyos korlátok mellett jól adaptálhatók a TBR renderelési eljáráshoz.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

6. IRODALOM

- [1.] ZACH B.: A Modern Approach to Software Rasterization, University Workshop, Taylor University 2011.
- [2.] SWENNY T.: The End of the GPU Roadmap. Proceedings of the Conference on High Performance Graphics, pp. 45-52, 2009.
- [3.] RAD GAME TOOLS: Pixomatic Advanced Software rasterizer, 2012.
- [4.] TRANSGAMING Inc: Swiftshader Software GPU Toolkit, 2012.
- [5.] COFFIN C.: SPU-based Deferred Shading for Battlefield 3 on Playstation 3. Game Developer Conference Presentation, March 8, 2011.
- [6.] LAINE S., KARRAS T.: High-Performance Software Rasterization on GPUs. High Performance Graphics, Vancouver, Canada, aug 5. 2011.

FORMÁLIS NYELVTANI MODUL A META KERET- RENDSZER SZÁMÁRA

FORMAL GRAMMAR MODUL IN GI META FRAMEWORK

Tóth Zsolt*

ABSTRACT

There are huge amount of knowledge encoded in written format in a sort of natural language. Hence text processing and information extraction are becoming more and more important fields of computer science and knowledge engineering these days. These processes can be improved by knowledge about the structure of grammar of the language. Theory of formal languages and grammars give the mathematical background to model natural and artificial languages. We are interested in the automatic induction of formal grammars, especially the induction of context free grammars. In this paper the formal grammar module of META¹ framework, which is a grammar induction (GI) and text mining framework, is presented. The presented framework and its module will be used in our further research on context free grammar induction methods.

1. BEVEZETÉS

Napjainkban a tárolt információ jelentős részét természetes nyelven, írott szöveges dokumentumok például levelek, jelentések, elemzések, könyvek formájában tárolják. A korszerű információkereső rendszerek célja a dokumentumok feldolgozása, kivonatolása, tárolása, indexelése a gyors és hatékony keresés céljából. A dokumentum feldolgozás hatékonyságának javítására egy elterjedt módszer, hogy a feldolgozó motort információval látják el az adott hordozó nyelv struktúrájáról. A nyelvtani szabályok ismeretében a feldolgozó motor hatékonyabban képes kinyerni a hordozott információt, így javítva a rendszer pontosságát, működési paramétereit.

Az egyes nyelvtanok matematikai leírására a formális nyelvek szolgálnak, ahol a nyelvet az azt leíró nyelvtan segítségével lehet megadni. A nyelvtan manuális előállítás szakértelmet igénylő és időigényes feladat. Amíg a

nyelvtanok automatikus generálása NP nehéz probléma. Ennek az NP nehéz problémának a megoldására különböző megközelítésű, hatékonyságú módszerek léteznek. Ezért az automatikus nyelvtan generálás napjainkban is a számítástudomány aktív kutatási területének számít.

A nyelvtanok automatikus generálása nem csak a természetes nyelvű szövegek feldolgozása során alkalmazható, számos olyan tudományterület létezik, ahol az egyes tevékenységek, elemek leírására új mesterséges nyelveket lehet bevezetni. Ilyen területek például a biológia, a kémia és a biokémia, ahol az egyes kölcsönhatásokat bonyolult, összetett mondatok formájában adják meg [1]. A nyelvtan ezen felül egy tömör leírását adja a nyelvnek, így akár tömörítési feladatokra is alkalmazható [2].

A szöveges adatok és dokumentumok feldolgozására, elemzésére számos szabvány és megközelítés létezik, de a formális nyelvek kezelésére ezek viszonylag kevés lehetőséget biztosítanak. A formális nyelvek azon belül a környezet független nyelvek, automatikus generálásával kapcsolatos kutatásaink során felmerült az igény a különböző módszerek közös környezetben történő implementálására, így lehetővé téve tesztelésüket, összehasonlításukat, elemzésüket. Jelen munkában a META keretrendszert [3] ismertetem, aminek a fő célja, hogy a különböző formális nyelvek feldolgozására használt módszereket keretbe foglalja.

A továbbiakban röviden ismertetem a formális nyelvek leírását, kiemelve a számunkra fontos környezet független nyelvek osztályát. Majd áttekintem a legismertebb jelenleg elérhető adatbányászati, szövegbányászati rendszerek fő funkcióit. A jelenlegi rendszerek vizsgálata után ismertetem a META rendszerrel szembeni elvárásokat és részletezem a fő tervezési lépéseket. Ezután a formális nyelvekkel kapcsolatos modult részletesen ismertetem, majd az irodalomból jól ismert algoritmusokat mutatok be. Végül összegezem a kitűzött és az elért eredményeinket, illetve ismertetem a rendszer továbbfejlesztési lehetőségeit.

*Phd hallgató - Miskolci Egyetem Általános Informatikai Tanszék

¹ Miskolcian Environment for Text Analysis

2. FORMÁLIS NYELVEK

Az egyes természetes és mesterséges nyelveket matematikai szempontból a formális nyelvek segítségével lehet modellezni, ahol egy adott véges ábécé (Σ) felett értelmezett nyelv ($L = \{\omega\}$) mint mondatainak ($\omega \in \Sigma^*$) halmazaként van megadva. Az egyes nyelveket a nyelvtanukkal (G) lehet megadni, és azt mondjuk, hogy az L_G nyelvet a G nyelvtan állítja elő. A nyelvtant a $\langle T, N, P, S \rangle$ négyes határozza meg, ahol:

- T a terminális szimbólumok véges halmaza
- N a nem terminális szimbólumok véges halmaza
- P a képzési szabályok halmaza ($P = \{\alpha \rightarrow \beta \mid \alpha, \beta \in (T \cup N)^*\}$)
- S a mondat kezdő szimbólumok halmaza ($S \subseteq N$)

A formális nyelvek egyik jól ismert osztályozása a Chomsky hierarchia, amely az egyes nyelveket a képzési szabályaik alapján egyre szigorúbb csoportokba sorolja (lásd 1. táblázat). A Chomsky hierarchia osztályai közül a természetes nyelvek leírására a CSG és a CFG nyelvek alkalmasak, ezek közül most csak a CFG nyelvekkel foglalkozok, mert ezekkel hatékonyan lehet műveleteket végrehajtani.

Nyelv	Megvalósítás	Képzési szabály
Rekurzívan felsorolható nyelvek	Turing gép	$\alpha \rightarrow \beta$
Környezet függő nyelvek (CSG)	Linear-bounded automata	$\alpha A \beta \rightarrow \alpha \gamma \beta$
Környezet független nyelvek (CFG)	Push-down automata	$A \rightarrow \alpha$
Reguláris nyelvek	Véges állapotú automata	$A \rightarrow aB$ vagy $A \rightarrow Ba$ és $A \rightarrow a$

1. táblázat Chomsky hierarchia

2.1. Sztochasztikus környezet független nyelvek

A környezet független nyelvek egy speciális csoportja a sztochasztikus környezet független nyelvek (Probabilistic Context Free Grammar) [4], ahol az egyes szabályoknál egy bal oldalhoz több különböző valószínűségű jobb oldal is tartozhat. A PCFG képzési szabályai a CFG nyelvekhez hasonlóan: $A \rightarrow \alpha$ alakúak, de minden szabály minden jobb oldalához meg lehet határozni a valószínűségét $Pr(A \rightarrow \alpha_i)$ úgy, hogy $\sum_{A \rightarrow \alpha_i \in P} Pr(A \rightarrow \alpha_i) = 1$. A PCFG előnye a CFG-vel

szemben, hogy méri a bizonytalanságot is, illetve a PCFG nyelvtanok automatikus előállításához pozitív minták is elegendőek. Ezzel szemben a CFG nyelvtanok automatikus előállítása során negatív tanítómintákat is meg kell adni, aminek az előállítása időigényes feladat lehet.

3. ADATBÁNYÁSZATI ÉS NLP RENDSZEREK

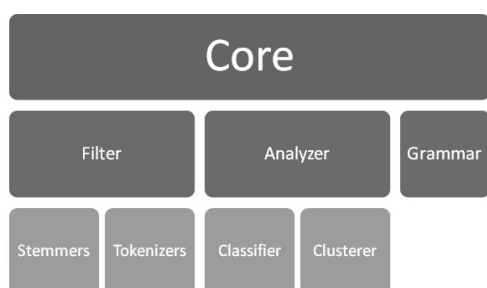
Az irodalomban számos módszer található a CFG és PCFG nyelvek generálására [5] és feldolgozására, illetve számos adatbányászati keretrendszer érhető el, melyek különböző mértékben szöveg feldolgozási funkciókat is megvalósítanak. Az egyik legismertebb ilyen keretrendszer a Weka ami egy adatbányászati és gépi tanulási keretrendszer. A Weka mellett számos adatbányászati alkalmazás és rendszer érhető el a piacon, de ezeket a terjedelemre való tekintettel nem ismertetem. A formális nyelvek feldolgozására egy minta alkalmazás a JavaCFG, amit oktatási céllal fejlesztettek a Columbia Egyetemen. Bár a megvizsgált keretrendszerek és alkalmazások számos szolgáltatást nyújtanak, amik segítenek a CFG generálás területén végzett munkát, de egyik rendszer sem elégíti ki teljes mértékben az igényeket.

A Weka az egyik legismertebb adatbányászati keretrendszer, ami számos osztályozó, klaszterező, előfeldolgozó algoritmust tartalmaz adatbányászati feladatok ellátására, viszont nem támogatja kellően a formális nyelvek modellezését. A rendszer a nyíltsága és jó dokumentációja miatt népszerű és elterjedt, ezért vettem alapul a META keretrendszer fejlesztése során. A Weka rendszerhez a könnyű használat érdekében egy egyszerű grafikus felhasználói felület is tartozik, ami segítségével könnyen lehet grafikus szemléltetni az eredményeket.

Az irodalomban található számos mintaillesztő [6] algoritmus és nyelvtan generáló módszer [7,8] környezet független nyelvek számára. Az környezet független nyelvek generálásával kapcsolatban végzett legfrissebb eredményekről, tendenciákról jó áttekintés található D'Ulizia munkájában [5]. Ezeket a módszereket különböző programozási nyelveken implementálták és tesztelték. Az egyes módszerek jelentősen eltérhetnek egymástól mind alapötletükben, mind alkalmazott technikájukban. A cél egy közös környezetbe implementálni a legismertebb módszereket, hogy megfelelően tesztelni, összehasonlítani lehessen azokat.

4. META KERETRENDSZER

A META [3] egy formális nyelvi és szöveg feldolgozási keretrendszer, amely fő célja, hogy közös keretbe foglalja az egyes formális nyelveken végezhető módszereket. A rendszerrel szembeni alapvető elvárás, hogy lehetőséget nyújtson a formális nyelvek modellezésére, kezelésére és könnyen bővíthető legyen új módszerekkel. A rendszer ezen felül egyéb funkciókat is nyújt a felhasználók számára, mint például a naplózás. Mivel a nyelvtan generálás szorosan kapcsolódik a természetes nyelvek feldolgozásához és a szövegbányászathoz, ezért a keretrendszer lehetővé teszi, hogy később további szövegbányászati módszerekkel való kiegészítésre. De a rendszer jelenlegi verziója csak az egyes interfészeket definiálja, konkrét algoritmust nem valósít meg.



1. ábra A META keretrendszer struktúrája

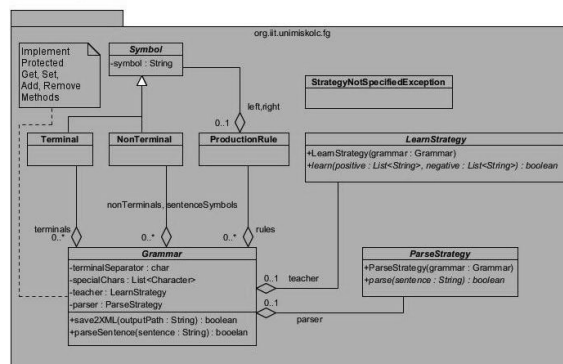
Az 1. ábra mutatja a META keretrendszer struktúráját, amin látható, hogy a keretrendszer a Core modul köré épül. A Core modul nyújtja a többi modul számára az alapvető szolgáltatásokat és adatstruktúrákat. Az egyes algoritmusokat a megfelelő modulok valósítják meg, például az előfeldolgozást végző különböző módszereket a Filter modul foglalja össze, amíg az elemző funkciókat az Analyzer valósítja meg. A nyelvtani módszereket a Grammar modul valósítja meg. Így a rendszerstruktúra lehetővé teszi számos módszer átlátható implementálását a rendszerbe.

4.1. A környezet független nyelvi modul

A Grammar modul a formális nyelvek leírását valósítja meg, valamint lehetőséget ad saját mintaillesztő és nyelvtan tanuló módszerek implementálására. A formális nyelvi csomag UML osztálydiagramja a 2. ábra látható. A modul a különböző típusú nyelvtan elemeket külön osztályok segítségével valósítja meg. A Grammar osztály ezen osztályok kollekciónak tartalmazva valósítja meg a nyelvtan leírását. A bővíthetőség érdekében az egyes nyelvtanok különböző mintaillesztő és tanuló stratégiákat alkalmazhatnak, így az új módszer hozzá-

adása a megfelelő ősztyály leszármaztatásával érhető el. Az használni kívánt tanító illetve mintaillesztő módszer a Grammar objektumnak be kell állítani, a learn illetve a parse metódus első meghívása előtt. Ha a felhasználó elmulasztotta a stratégia meghatározását, akkor StrategyNotSpecified-Exception kivétel keletkezik a learn vagy parse metódusának meghívásakor.

A nyelvtanok ilyen fajta megvalósítása nagy szabadságot enged a saját módszerek implementációja során. A nyelvtanok modellezését segíti, hogy a képzési szabályok a rekurzívan felsorolható nyelvek szabályait valószínűsítik meg, azaz a szabályok $\alpha \rightarrow \beta$ alakúak. A programozónak kell felügyelnie, hogy milyen típusú nyelvtant állít elő. A szabályok között a rendszer megkülönbözteti az egyszerű és a sztochasztikus képzési szabályokat, valamint megkülönbözteti az egyszerű és a sztochasztikus nyelvtanokat, mivel a szabályaik kezelése jelentősen eltér egymástól.



2. ábra A formális nyelvi modul struktúrája

A Grammar csomag egy részcsomagja a CFG csomag, ami a környezet független nyelvek feldolgozásához kapcsolódó módszereket foglalja össze. A csomag számos az irodalomban megtalálható módszer megvalósít, többek között mintaillesztő algoritmusok közül a CYK algoritmust és különböző tanító algoritmusokat, mint például a TBL [8], az ITBL [7], és az InduktívCYK [9] algoritmust. Az egyes módszerek implementálásával láthatóvá vált, hogy a META keretrendszer képes összefogni a különböző módszereket, így teljesíti a legfőbb elvárásokat.

5. TOVÁBBFELJELSZTÉSI LEHETŐSÉGEK

A META keretrendszer jelenlegi verziója az egyes módszereknek API-n keresztül történő elérését támogatja. Bár a jelenlegi rendszer megfelelő szolgáltatások biztosít különböző módszerek implementálására és a

rendszer kiterjesztésére, az egyes módszerek tesztelése, a futási információk megjelenítése, elemzése kényelmetlen és költséges feladat. A könnyebb használat érdekében egy korszerű, áttekinthető GUI-s alkalmazás készítését tervezem.

A grafikus alkalmazás fejlesztése mellett további cél, az irodalomban fellelhető [5] algoritmusok implementálása a META keretrendszerbe. A későbbiekben szeretném részletesen elemezni a különböző módszerek hatékonyságát, időkölttségét.

A különböző módszerek elemzésén túlmutatóan saját CFG, PCFG generáló módszer kidolgozását tűztem ki célul, melyet a META keretrendszerbe fogok implementálni. A tervezett módszert így számos az irodalomban korábban megjelent algoritmussal össze tudom majd hasonlítani.

6. ÖSSZEFOGLALÁS

A tanulmányban egy új, korszerű formális nyelvi keretrendszert és annak környezetet független nyelvi modulját mutattam be. A formális nyelvek matematikai formalizmusának rövid áttekintését követően ismertettem néhány elérhető, hasonló rendszert. Az egyes rendszerek előnyeit, lehetőségeit tömören ismertettem, kitértem az egyes rendszerek korlátaira. Megállapítottam, hogy egyik ismertett rendszer sem nyújt teljes körű támogatást a formális nyelvek kezelésére. Az ismertett rendszerek áttekintése után megfogalmaztam a META rendszerrel szemben támasztott követelményeket, igényeket.

Az igények összegzése után bemutattam a tervezett rendszer architektúráját, ismertettem a felépítés előnyeit, az egyes fő modulokat. Külön kiemeltem a rendszer lehetőségeit és röviden kitértem a szövegbányászati modulra. A rendszer áttekintése után a formális nyelvi Grammar modult részleteztem. A Grammar modul ismertetése után a környezetet független nyelvtanok kezelésére szolgáló algoritmusok implementációját ismertettem. A CFG modul számos az irodalomban fellelhető, jól ismert módszert tartalmaz. A rendszer bemutatását követően a továbbfejlesztési lehetőségeket, valamint a további felhasználási lehetőségeket foglaltam össze.

7. KÖSZÖNETNYÍLVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

8. IRODALOMJEGYZÉK

- [1] **Gavrilis, D. and Tsoulos, I. and Dermatas, E.** Evolutionary Grammar Induction for Protein Relation Extraction.
- [2] *Compression by induction of hierarchical grammars.* **Nevill-Manning, C.G. and Witten, I.H. and Mulsby, D.L.** Data Compression Conference : IEEE, 1994.
- [3] *META: A novel grammar induction and text mining framework.* **Tóth Zs., Kovács L.** XXVI. MicroCAD konferencia - Miskolc : Miskolci Egyetem, 2012.
- [4] *PCFG models of linguistic tree representations.* **Johnson, M.** 4, Computational Linguistics : MIT Press, 1998., 24. kötet.
- [5] *A survey of grammatical inference methods for natural language learning.* **Arianna D'Ulizia, Fernando Ferri, Patrizia Grifoni.** 1-27, Artificial Intelligence Review : Springer, 2011., 36. kötet.
- [6] *A generalized CYK algorithm for parsing stochastic CFG.* **Chappelier, J.C. and Rajman, M.** First Workshop on Tabulation in Parsing and Deduction : Citeseer, 1998.
- [7] *Improved TBL algorithm for learning context-free grammar.* **Jaworski, M. and Unold, O.** Proceedings of the International Multiconference on ISSN : ismeretlen szerző, 2007.
- [8] *Ga-based learning of context-free grammars using tabular representations.* **Sakakibara, Y. Kondo, M.** Machine Learning - International Workshop then Conference : Morgan Kaufmann Publishers Inc., 1999.
- [9] *Synthesizing context free grammars from sample strings based on inductive CYK algorithm.* **Nakamura, K. Ishiwata, T.** Grammatical Inference: Algorithms and Applications : Springer, 2000.

EGY ÚJSZERŰ HOZZÁFÉRÉS SZABÁLYOZÁSI MÓDSZER A LINUX OPERÁCIÓS RENDSZER KERNELÉBEN

A NOVEL ACCESS CONTROL METHOD IN THE KERNEL OF THE LINUX OPERATING SYSTEM

*Vincze Dávid**

ABSTRACT

This paper presents a novel access control method, which can help in situations where long-running processes take care of many independent and untrusted data or program code in a sequential execution order. After a short overview of the basics of access control methods, the concepts of the new method are introduced. This new method tends to be a both fast and secure solution. Also the implementation of this method in the Linux kernel is briefly explained. Finally an application example making use of this new method is presented, which can be useful in real world scenarios. In this example the Apache web server was modified to make use of the introduced new method. Opposed to other solutions which either have security or performance weaknesses, this new method preserves the required level of security while causing only a very low performance overhead.

1. BEVEZETÉS

A több felhasználós számítógép rendszerekben elsődleges szempont az adatok, illetve az alaprendszer védelme, mind a felhasználók, mind pedig a külső hatások ellen. A legtöbb biztonságot fokozó megoldás, a biztonsági szintek növekedésével jelentősen csökkentheti a rendszer használhatóságát, illetve teljesítményét. Az üzemeltetők számára hamar világossá válik, hogy a teljesítmény és a biztonság együttes növelése egymást kizáró tényezők. A teljesítmény általában fontosabb szerepet játszik a döntésekben, mivel gazdasági szempontból egyértelműen ez az előnyösebb: gyorsabb végrehajtás, több előfizető egy kiszolgálón, kevesebb erőforrás elegendő ugyanannyi ügyfélhez, stb. Ilyen módon a biztonság sokszor háttérbe szorul, holott néha a biztonsági hiányosságok okozta anyagi kár nagyobb, mint amennyit a megelőzésre lehetett volna fordítani, nem beszélve az elszenvedett presztízs, reputáció veszteségről.

Jelen cikk egy olyan újszerű megközelítést mutat be a hozzáférés szabályozás olyan lehetőségére, ahol a biztonság fokozása mellett a teljesítmény veszteség minimális, illetve a használhatóság sem csökken, sőt bizonyos esetekben bővül a használhatóság. Az első fejezetben egy rövid áttekintés olvasható a hozzáférés szabályozás lehetőségeiről egy modern, Linux kernel alapú rendszeren, majd bemutatásra kerül az ajánlott új módszer és végül az új eljárás egy lehetséges gyakorlati alkalmazási példája.

2. HOZZÁFÉRÉS SZABÁLYOZÁS

Jogkör ellenőrzés szempontjából a hagyományos UNIX-ok – a Linux-ot is ide sorolva – felhasználói módban két jogosultsági szintet különböztetnek meg: egyik a rendszergazdai szint (0-s azonosító), a másik a normál felhasználói szint (nem 0-s azonosító). Az alapvető hozzáférés ellenőrzése egyszerű: a rendszergazdai szinten futó folyamatoknál (processzeknél) az ellenőrzés elmarad, viszont a normál felhasználói szinten a felhasználói és csoportazonosítók kerülnek összevetésre az elérni kívánt erőforrás által megkövetelttel. Az összevetés három különböző módon ellenőrzi a hozzáférést: először az erőforrás tulajdonosának azonosítójával veti össze a kezdeményező felhasználó azonosítóját, ugyanez a lépés megtörténik a csoport azonosítók összevetésével, majd utolsó lépésben pedig, ha egyiknél sem volt egyezés, akkor a csoport és tulajdonoson kívül az egyéb („mások”) felhasználók számára is definiálhatóak külön hozzáférési jogosultságok. Utóbbinál fontos megjegyezni, hogy ebben a csoportban nem tesz különbséget a felhasználók között, mindenki egyenrangúan tartozik a „mások”-ba.

Ezen felül egyes erőforrás típusokhoz (pl. fileok, IPC erőforrások) úgynevezett ACL-ek (Access Control List) [6], azaz hozzáférés szabályozási listák rendelhetők. Ezek a listák a hagyományos tulajdonos/csoport/mások hármason felül további jogosultságokat határozhatnak

**egyetemi tanársegéd, Miskolci Egyetem, Ált. Inf. Tsz.*

meg konkrét felhasználók vagy csoportok számára, ezzel kiküszöbölve az említett tradicionális modell hiányosságait.

A POSIX (Portable Operating System Interface) kompatibilis több felhasználós operációs rendszereken a felhasználók elkülönítése érdekében minden felhasználóhoz rendelve van legalább egy felhasználói azonosító (uid – userid), valamint egy alapértelmezett csoport (gid – groupid), amibe beletartozik. Ezek alapján történik az alapvető azonosítás és jogkör ellenőrzés. A POSIX.1-ben [5] szabványosított *setuid()* / *seteuid()* / *setreuid()* / *setresuid()* rendszerhívások lehetővé teszik az aktuális processz felhasználói azonosítójának megváltoztatását. Ugyanígy léteznek rendszerhívások a processz csoport azonosítójának megváltoztatására is: *setgid()* / *setegid()* / *setregid()* / *setresgid()*.

Direkt módon abban az esetben szokás használni, amikor egy rendszergazda által indított és így rendszergazdai jogosultságokkal futó programnak csak néhány dolog elvégzéséhez van szüksége a privilegizált jogokra. Miután ezeket a feladatokat elvégezte a *setuid()* rendszerhívás meghívásával átvált egy normál felhasználói azonosítóra, így megszabadul a rendszergazdai jogosultságoktól és folytatja tovább a futását. Fontos, hogy miután megszabadult a processz a rendszergazdai jogosultságoktól, már nem lesz joga újból sikeresen végrehajtani a *setuid()* rendszerhívást, így már nem lesz képes újból átállni a processz más felhasználói azonosítóra.

A POSIX.1e [6] szabvány definiálja a capabilities rendszert, miszerint a fenténél (rendszergazda vs. nem rendszergazda) kifinomultabb jogkör kiosztás és ellenőrzés is használható. A POSIX.1e szabvány nem készült el teljes egészében, csak egy tervezet készült róla, és később azt is visszavonták, de ettől függetlenül a capabilities rendszer több operációs rendszeren is részben megvalósított. A capabilities rendszer több privilegiumot, képességet határoz meg, amelyek bármely processzre ráruházhatóak, bármely processztől elvehetőek, felhasználótól függetlenül. A Linux alatti megvalósításban [7] használható képességek közül néhány:

- CAP_SETPCAP: Egy másik processz képességeinek beállításához való képesség. Természetesen csak az aktuális processz képességeinek részhalmazából lehet továbbadni, illetve elvenni. Felhasználói azonosító ellenőrzés a két processz között nem történik, így a használata kockázatos lehet. Éppen ezért újabb kernelekben már nincs használatban.
- CAP_SETUID: Tetszés szerinti változtatását enged meg az aktuális processz csoport azonosítóján (*setuid()* hívások engedélyezése).
- CAP_SETGID: Tetszés szerinti változtatását engedi meg az aktuális processz csoport

azonosítóján és mellékcsoport azonosító listáján. A *setgid()* rendszerhívások megengedése.

Alaphelyzetben a rendszergazda az összes lehetséges képességgel rendelkezik, a normál felhasználók pedig nem rendelkeznek egyetlen képességgel sem. Abban az esetben, amikor rendszergazdai jogosultsággal indul egy program, akkor menet közben átállhat egy másik normál felhasználóra (lásd korábban *setuid()*), viszont a képességek közül megtarthat kijelölt capability-eket normál felhasználóként is. Előfordulhat, hogy az adott képességre egész élete során szüksége van a folyamatnak, viszont amennyiben csak pl. inicializációhoz (pl. naplófile megnyitása vagy 1024 alatti TCP/UDP porthoz kötés), akkor a feladat elvégzése után a képesség eldobható.

A bemutatott megoldások különböző megközelítéseket, módszereket alkalmaznak, de mindegyik megoldásnál az a közös pont, hogy az alapján dönti el a hozzáférés jogosultságát, hogy a kérvényező folyamatnak éppen aktuálisan mi a felhasználói, illetve csoport azonosítója.

A hozzáférés szabályozás lehetőségeinek rövid áttekintése után bemutatásra kerül a következő fejezetben egy újszerű módszer, ami egy lehetséges megoldást ad a futásközbeni felhasználói azonosító változtatás biztonságossá tételére.

3. A JAVASOLT ÚJ MÓDSZER

A módszer lényege a futásidőben való felhasználói azonosító változtatás ugyanazon a processzen. Az előző fejezetben megismert módon rendszergazdai felhasználói azonosítóról bármikor át lehet váltani bármelyik normál felhasználói azonosítóra. Viszont innentől kezdve már nincs lehetősége a processznek újból változtatni a felhasználói azonosítóján. Ellenben ez jól kihasználható olyan alkalmazásoknál, amelyek nem megbízható adatokkal dolgoznak, netán nem megbízható külső programkódrészeket futtatnak. Megoldás lehet a CAP_SETUID beállítása az adott processzre, hogy továbbra is legyen jogosultsága a *setuid()* rendszerhívás családot sikeresen végrehajtani. Ez biztonsági szempontból hátrányos lehet, hiszen így maga a nem megbízható kódrészlet is kérvényezheti az felhasználói azonosító átállítást, holott csak adott felhasználói azonosítóval volna szabad futnia. Tehát ez a megoldás nem kielégítő. Célszerű volna egy olyan új operációs rendszer (kernel) funkciót implementálni, ami adott feltétel teljesülése esetén visszaadhat meghatározott jogosultságokat a futó processz számára (pl. felruhazza CAP_SETUID képességgel).

Az alapvető problémát igazából annak az egyértelmű eldöntése jelenti, hogy éppen hol jár a végrehajtás egy adott processzen belül. Nem lehet tudni, hogy épp

megbízható, vagy épp egy kívülről (hálózatról, más felhasználótól) származó, nem megbízható kódrész kérvényezi-e a jogosultságok visszakérését.

Ezért keresni kell egy olyan módszert, ami különbséget tud tenni a két állapot között, hogy megbízható kód fut-e éppen vagy sem. Megfelelő megoldásnak lehet az alapján különbséget tenni, hogy éppen milyen memóriacímen jár a végrehajtás a kérvényező processzben. Tehát meg kell vizsgálni minden egyes jogosultság visszakövetelésekor, hogy honnan lett kérvényezve az adott processzen belül.

A vázolt problémákat nem lehet kizárólag felhasználói térben futó programokkal megoldani. Egy kernel térben futó eljárásra is szükség van a megvalósításhoz, mivel a kernel fogja meghozni a döntést, hogy az adott processz megkaphatja-e a többlét jogosultságot, illetve csak a kernel képes feljogosítani egy processzt új képességekkel. A továbbiakban egy olyan új rendszerhívás kerül bemutatásra, ami a felhasználói térbe való visszatérési cím alapján dönt és adott jogosultsági szintre emeli a hívó processzt.

A gyakorlatban ez úgy valósult meg, hogy az új rendszerhívás (*getmyretaddr()*) meghívásakor – mivel a kernelbe lépés előtt eltárolódik a veremben (stack) az a cím, ahová majd a hívás végeztekor vissza kell térni a kernelből – a veremben visszafelé haladva megkeresi a visszatérési címet. Ha ez a cím egy korábban megbízhatónak minősített címről jön, akkor többlét jogokhoz juthat a processz, ha ugyanez a hívás máshonnan hívódik meg, nem lesznek kiváltságai a processznek.

Egy rendszerhívás meghívásakor a processz kontextus egy része is lementődik a verembe. Miután a rendszerhívás véget ért, a vezérlés egy *ret* instrukcióval tér vissza a rendszerhívás utáni instrukcióra a hívó processzen belülre. Azt pedig, hogy hová térjen vissza, a veremből olvassa ki. Ez utóbbi címet kell megkeresni, ezt használhatjuk hívó címként, (valójában ez az éppen utána következő instrukció címe). A hívó cím birtokában úgy fog megtörténni a jogosultság ellenőrzés, hogy a *getmyretaddr()* rendszerhívás első meghívásakor inicializálódik a processz kontextushoz újonnan hozzáadott *retaddr* nevű változó, aminek az alaphelyzetbeli értéke 0. Az inicializálás csak akkor lesz sikeres, ha a processz rendelkezik a *CAP_SETUID* és a *CAP_SETGID* képességekkel. A későbbiekben a *getmyretaddr()* hívásakor ha a *retaddr* értéke nem 0, és az eltárolt *retaddr* és az aktuális hívócím megegyeznek, akkor a processz felruházódik a *CAP_SETUID* és *CAP_SETGID* képességekkel. Ez után már a felhasználói térben futó program dolga, hogy átállítsa a felhasználói és csoport azonosítókat és megszabaduljon a két képességétől, hogy a nem megbízható kód résznek ne legyen csak minimális jogköre.

Mindenképpen említést kell tenni a módszer okozta, újonnan nyílt biztonsági kockázatokról és a lehetséges védekezési módokról.

Az egyik ilyen a futó kód felülírásának a lehetősége. A processz kontextusban tárolt *retaddr* értékét közvetlenül nem lehet lekérdezni a kerneltől, ebből adódóan azt egy felhasználói térben futó program nem tudhatja meg ilyen módon. Azonban a saját programkódját tudja olvasni a processz, így a *getmyretaddr()* hívás címe visszafejthető. Így ha a megfelelő memóriacímre saját programkódot illeszt be egy lehetséges támadó, akkor a *CAP_SETUID* és *CAP_SETGID* képességek birtokába juthat, mivel ki tudja kerülni a képességek eldobását. Alaphelyzetben az a memória rész, ahol a kód fut, nem írható, de ezt át tudja állítani az *mprotect()* rendszerhívással a támadó, hogy írhatóvá váljon. Ennek kiküszöbölésére beillesztettem egy ellenőrzést az *mprotect()* rendszerhívás kódjába: ha olyan memórialapra kér a processz írási jogot, amibe beleesik a *retaddr* értéke, akkor a rendszerhívás nem fog végrehajthatni, és hibával tér vissza.

Egy másik kijátszási módszer lehet, ha már egy processz kontextusában inicializálva van a *retaddr* változó és ez a processz végrehajt egy *exec()* rendszerhívást, ami új kóddal írja felül a processz futtatott kódját. Ilyenkor a *retaddr* inicializálva marad, így ha a megfelelő címről hívódik meg a *getmyretaddr()* rendszerhívás, akkor joggalansul hozzájuthat a két képességhez. A védekezési módszer egyszerű: az *exec()* rendszerhívásnak ki kell nulláznia a *retaddr* értéket.

A harmadik eljárás a veremben lévő adatok meghamisítása lehet. Erre azért van lehetősége a feltételezett támadónak, mert a rendszerhívás a *glibc syscall()* csomagoló függvényével lett meghívva. Ha közvetlenül hívja meg a támadó a rendszerhívást, akkor a kernel által *retaddr*-nak vett címet meg tudja hamisítani. Megoldás, hogy az eljárást használó processzből is közvetlenül kell meghívni a *getmyretaddr()* hívást.

4. ALKALMAZÁSI PÉLDA

Napjainkban az Internet második legfontosabb szolgáltatása (az e-mail után) a web. A különböző webalkalmazások felépítése változatos lehet, különböző programnyelveken, nem ritkán egyedi keretrendszereket felhasználva íródnak, számos különböző technológia alkalmazásával. Külső közös kapcsolódási pontjuk a web szerver, amely az egész webalkalmazást egy szabványosított interfészen keresztül elérhetővé teszi a hálózaton. Alapvetően két fajta kiszolgáló (itt erőforrás értelemben) figyelhető meg. Egyik ahol egy dedikált alkalmazás fut, ami egy megbízható felhasználóhoz, ügyfélhez tartozik. A másik típus, pedig, ahol sok egymástól különböző, egymástól idegen, nem megbízható felhasználó számos, egymáshoz semmilyen

módon nem kapcsolódó webalkalmazása fut. Ez utóbbival foglalkozunk a továbbiakban.

Több felhasználós rendszereken a legtöbb esetben ezeket az alkalmazásokat kiszolgáló web szerverek a gyors feldolgozás érdekében minden felhasználó programjait, szkriptjeit ugyanazzal a közös felhasználói azonosítóval és jogkörrel futtatják. A futtatott szkriptek így egyenrangúak, függetlenül attól, hogy melyik felhasználóhoz tartoznak, mivel mind a web szerver közös jogosultságával fut. Ez sok esetben kényelmetlen lehet és rosszindulatú felhasználók ezt könnyedén kihasználhatják [2].

Az egyik legelterjedtebb web szerver az Apache HTTPD [1], így kézenfekvő választás ezen tesztelni a bemutatott új eljárást. Tradicionálisan az Apache webkiszolgálók a *prefork* modell (MPM) szerint működnek. Ebben a modellben egyszerre több, meghatározott számú gyermek processzt indít el és felügyel a web szerver fő processze. Egy-egy gyermek processz élete során sok kérést ki tud szolgálni, tipikusan több százat, de akár ezeket is, így nem szükséges minden kéréshez új processzt létrehozni, inicializálni, ami jelentős erőforrás megtakarítást jelent.

A fentebb leírt problémák kiküszöbölésére számos megoldás látott napvilágot az Apache web szerverhez [3], [4], azonban jelentős teljesítmény csökkenéssel vagy más jellegű biztonsági kockázattal jár az alkalmazásuk. Az előbbieken bemutatott új módszert alkalmazva az alap probléma olyan módon hidalható át, hogy egyszerre gyors és biztonságos környezetben futhatnak a webalkalmazások.

A *prefork* MPM-be került be az előbbieken bemutatott új módszer implementációja. Az alapötlet röviden a következő: az adott gyermek processz alaphelyzetben rendelkezik a *CAP_SETUID* és *CAP_SETGID* képességekkel, a kérés feldolgozásának kezdetén meghívja a web szerver a *getmyretaddr()* rendszerhívást, így az első hívásnál az inicializálás és a felhasználó váltás sikeres lesz. A váltás után eldobja a két képességet és kiszolgálja a kérést immáron a megfelelően beállított azonosítókkal. Egy következő kérés feldolgozásakor már nem rendelkezik a két képességgel a *getmyretaddr()* meghívásakor, de ha azt ugyanarról a helyről hívták meg, mint a legelső esetben, akkor újra birtokolni fogja őket, de az azonosító váltások (*setuid()* és *setgid()* rendszerhívások) után ismét megszabadul tőlük és elkezd feldolgozni a következő kérést.

Ezekkel a módosításokkal az Apache képes lett egy gyors és az eredetinel biztonságosabb módszert alkalmazni, annak érdekében, hogy a hozzá intézett kéréseket szükség esetén más és más felhasználói, illetve csoport azonosítóval szolgálja ki, megkönnyítve a webalkalmazás fejlesztők és üzemeltetők munkáját.

5. ÖSSZEFOGLALÁS

A cikkben bemutatásra került egy újszerű hozzáférés szabályozási módszer. A módszer lényege, hogy a jogosultság eldöntéséhez azt vizsgálja, hogy honnan lett kérelmezve a jogosultság emelés a kérelmező processzen belülről. Az új eljárás a Linux operációs rendszer kernelében lett implementálva. Konkrét alkalmazás példaként pedig a jelenleg legelterjedtebb web szerver (Apache HTTPD) lett felkészítve a módszer használatára. Így az Apache HTTPD alkalmassá válik arra, hogy a különböző felhasználók programjait, szkriptjeit a saját hozzájuk tartozó felhasználói azonosítójával. Így az adott felhasználó jogosultságaival tudja futtatni ezeket a web szerver, más hasonló módszereknél jelentősen alacsonyabb erőforrás igényvel.

6. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg

7. IRODALOM

- [1.] NetCraft Ltd. Web Server Survey, April2012: <http://news.netcraft.com/archives//2012/04/04/april-2012-web-server-survey.html>
- [2.] S. GARFINKEL: Web Security, Privacy & Commerce, Second Edition, O'Reilly Media Inc., Jan. 2002, ISBN: 978-0596000455
- [3.] S. H. GUNDERSON: The Apache 2 ITK MPM <http://mpm-itk.sesse.net> (Hozzáférve: 2012. Apr.)
- [4.] D. HARA, Y. NAKAYAMA: Secure and High-performance Web Server System for Shared Hosting Service, Proceedings of the 12th International Conf. on Parallel and Distributed Systems (ICPADS'06), Minneapolis, USA, 2006
- [5.] IEEE Standard for Information Technology 1003.1-2004: Portable Operating System Interface (POSIX)
- [6.] IEEE Standard for Information Technology P1003.1E Draft: Portable Operating System Interface (POSIX)
- [7.] B. TOBOTRAS: Linux Capabilities FAQ, 1999 <http://ftp.kernel.org/pub/linux/libs/security/linux-privs/kernel-2.4/capfaq-0.2.txt> (Hozzáférve: 2012. Apr.)

(HELYI) HÁLÓZATOK AKTUÁLIS BIZTONSÁGI PROBLÉMÁI

SECURITY PROBLEMS OF (LOCAL) NETWORKS

Wagner György*, Tóth Tibor**

ABSTRACT

The use of cloud computing in Hungary is becoming more prevalent. Companies may face several security problems when partially or fully introducing this technology. Their data, their systems and also their internal policies should therefore be reviewed and modified if necessary. The paper discusses the concept of clouds, their variants and security issues concerning their use.

1. BEVEZETÉS

A számítási felhők használata Magyarországon is egyre jobban terjed. A vállalatok ezek részleges vagy teljes bevezetésekor többféle biztonsági problémával kerülhetnek szembe. Adataikat, rendszereiket, belső szabályzatukat ennek megfelelően érdemes felülvizsgálni, szükség esetén módosítani. A cikk kitér a felhők fogalmára, változataira, biztonsági kérdésekre.

2. ELŐZMÉNYEK

Az informatikai eszközök kialakulásuk óta folyamatosan jelentős változáson estek át. Ez a Moore-nak tulajdonított (1965-ös tapasztalati) törvénynek köszönhetően (egyszerűsítve: az integrált áramkörök összetettsége 18 havonta megduplázódik) jól ismert, és elfogadott [1]. A számítógépeket kezdetben szigorúan zárt, klimatizált termekbe telepítették, sokszor csak pormentes ruhákban és csak kiképzett személyzet részére voltak közvetlenül elérhetők. Elfogadott volt a kötegelte (batch) program futtatás, amelyek a kor fejlettségének megfelelően valamilyen lyukkártyás – lyukszalagos adatbeviteli- és kiviteli eszközöket használtak. A batch alapú rendszerek a hatékony programfejlesztést, a gyors hibakeresést erősen korlátozták. Az igényeknek megfelelően először az interaktív operációs rendszerek jelentek meg, majd a személyi számítógépek kifejlesztésének, és elterjedésének köszönhetően bekövetkezett a számítóközpontok előnyökkel és hátrányokkal

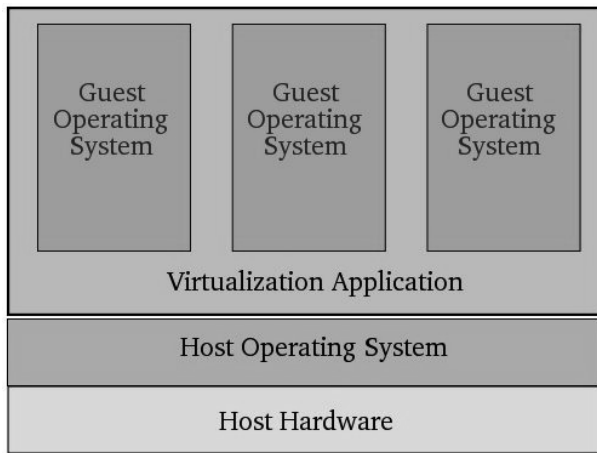
együttjáró decentralizációja. Felismerték, hogy a személyi számítógépek számítási kapacitása (többek között az alkalmazott architektúrának is köszönhetően) jóval kisebb, mint egy specializált rendszeré. Emiatt kifejlesztésre kerültek, és elterjedtek a fejlettebb, költségesebb architektúrájú szerver feladatokat ellátó számítógépek. A felhasználók számára kifejlesztett szoftverek ehhez illeszkedve szintén két részből tevődtek össze (két rétegűek voltak). Az egyik komponens a szerveren futott, a másik a felhasználó (kliens) számítógépén. A szoftverek idővel tovább rétegződtek annak érdekében, hogy az egyes feladatokat megvalósító részmegoldások könnyebben cserélhetők legyenek.

Az idő múlásával az asztalai számítógépek számítási teljesítménye jelentős mértékben megnőtt. Ennek kihasználására egy számítógépre több szerver szoftver is telepítésre került. Az egyes szoftvereket kiadásuk után a gyártók folyamatosan tesztelték, javították, majd a javítást (patch) valamilyen formában hozzáférhetővé tették. A patch telepítése után a számítógépet sok esetben újra kellett indítani. Az újraindítás azonban azokra a szerver funkciókat biztosító rendszereket is leállította, majd elindította, amelyek nem voltak érintettek a frissítésben. (Előfordulhatott, hogy ugyanazon a számítógépen futott egy SQL szerver és egy Web szerver. Ha az SQL szerverre megjelent egy patch, aminek telepítése után a számítógép újraindításra került, akkor ez idő alatt a Web szerver sem volt elérhető).

A virtualizációs technika kifejlesztésével lehetőség nyílt arra, hogy egy fizikai számítógépen több virtuális számítógépet lehessen emulálni. A virtuális gépekre egymástól független operációs rendszereket lehetett telepíteni, majd arra a szerver feladatot ellátó szoftvert. Így minden egyes virtuális gépen csak az adott szerver funkciót biztosító szoftver futott. A virtualizációt biztosító szoftver beállításától függően képes biztosítani, hogy a fizikai gépen futó egyes virtuális gépek elérjék egymást akár valós, akár belső, virtuális hálózaton keresztül. A virtualizációs szoftver lehet host alapú, amikor a virtualizációt megvalósító program csak egy az operációs rendszer által futtatott programok között (1. ábra), illetve lehet hypervisor alapú, amikor a

* egyetemi tanársegéd, Miskolci Egyetem, IIT

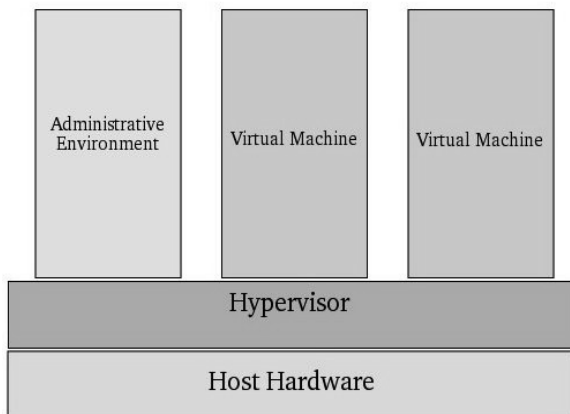
** egyetemi tanár, Miskolci Egyetem, AIT



1. ábra Hypervisor alapú virtualizáció

virtualizációt biztosító program közvetlenül a hardveren fut, operációs rendszertől függetlenül (2. ábra).

Host alapú például a VMWare-nek a Workstation elnevezésű terméke, vagy a Microsoft-nak a Virtual PC-je. Hypervisor alapú a VMWare-nek az ESX terméke, illetve a Microsoft-nak a Hyper-V szervere.



2. ábra Hypervisor alapú virtualizáció

A virtualizációt biztosító szoftver képes olyan virtuális gép emulálására, amely a valós géptől független: több-kevesebb memória, más grafikus kártya, más hangkártya, eltérő processzor.

Egy virtuális gépre telepített szerver funkciót biztosító szoftver lehet például egy file server (adattárolási szolgáltatással). Az a cég, amelyik ezt az adattárolást a saját gépparkján esetleg virtualizációval oldja meg, ilyen esetekben is tudja, hogy fizikailag hol tárolódnak az adatai. Az alkalmazott mentési stratégiának megfelelően például rendszeres időközönként archiválást végez a fizikai diszkről.

3. FELHŐK (CLOUD COMPUTING)

Létezhet azonban olyan eset, hogy egy felhasználó ezt a tárolási kapacitást bérlí egy olyan személytől (vagy cégtől), aki a tárolást úgy oldja meg, hogy megadja ugyan a tárolási szolgáltatás elérhetőségét, de az adatok tárolását több tárolóhely között saját maga végzi. Ezek a tároló helyek között lehetnek valós és virtuális gépek is vegyesen. A biztonság érdekében a tárolás akár redundáns is lehet. Ebben az esetben az, aki a tárolási szolgáltatást igénybe veszi, nem tudja, hogy adatai a világon hol tárolódnak. Valójában (ha csak valami ok miatt nem szükséges ezt tudnia), akkor ez számára nem is fontos. Ilyen jellegű szolgáltatások esetében felhőkről (felhő alapú szolgáltatásokról) beszélünk. Bár a későbbiekben a felhő alapú szolgáltatások részletezésre kerülnek, érdemes elgondolkodni egyrészt azon, ez mit is jelent, másrészt pedig azon, miért lehet erre bárkinek is szüksége.

Az a felhasználó (fogyasztó) aki a szolgáltatást igénybe veszi tárolási szolgáltatás esetén saját személyes (céges) adatainak tárolását bízta rá egy idegen (akár konkurens) cégre. Ez a cég a nála tárolt adatokat képes lehet olvasni, akár módosítani is. Ugyanakkor működési problémák is előfordulhatnak, és a tárolt adatok megsérülhetnek. Ki, milyen mértékben felelős? Kinek kell adatmentésről gondoskodni?

A másik kérdés, hogy mi indokolhatja az ilyen jellegű szolgáltatások használatát, már könnyebb a válasz. Több olyan cég is létezik, ahol bizonyos időszak(ok)ban megnövelt hardver teljesítményre van szükség. Ilyen lehet például egy adóbevallás, amikor rövid idő alatt igen nagy sávsebességre, sűrű mentésre lehet igénye a megfelelő hatóságoknak. Ennek érdekében saját informatikai eszközparkjukat úgy kell méretezniük, hogy azok képesek legyenek ennek az időszaknak a megnövekedett igényét kiszolgálni. Az időszak leteltével azonban erre a teljesítményre már nincs szükségük egy évig. Az informatikai eszközöket beszerzés után rövid idővel később eladni általában nem kifizetődő. Az eszközök tehát kihasználatlanok lesznek. Ha azonban ezt határozott időszakra, szolgáltatásként igénylik, akkor a bevallási időszak elteltével egyszerűen csökkent mértékű szolgáltatást kell csak igényelniük.

Látható, hogy valóban van létjogosultságuk az ilyen számítási felhő alapú szolgáltatásoknak, de a biztonság speciális kérdéseket vet fel.

3. FELHŐ ALAPÚ SZOLGÁLTATÁSOK

Összefoglalva tehát: vannak szolgáltatók, akik az általuk biztosított szolgáltatás kivitelezését több számítógép segítségével úgy oldják meg, hogy az azt igénybe vevő felhasználó elérí ugyan a szolgáltatást, de nem tudja, hogy a szolgáltatást biztosító számítógép(ek)

a világban hol is van(nak). A felhő alapú szolgáltatások rövidítve xaaS-ként (x as a Service) adhatók meg.

3.1 SaaS (Software as a Service)

Az egyik legelterjedtebb szolgáltatás a szoftverek felhő alapú szolgáltatása. Ha nem is ilyen néven, de maga a fogalom korábban is létezett. A szoftverek license szerződése a legtöbb esetben külön kitér arra, hogy a szoftver vásárlója a vásárlással nem tulajdonjogot szerez a szoftver felett, hanem csak annak használhatóságáért fizet a vevő. Létezik több olyan konstrukció is, ahol rendszeres időközönként (például évente) újra és újra fizetni kell a szoftverért, a szoftver használatáért (tipikusan ilyenek a vírusölő, vírus kereső szoftverek). Sokan nem rendelkeznek az általában a scanner-ekhez ingyenesen adott szövegfelismerő szoftverrel. Erre alapozva több olyan (ingyenes illetve fizetős) szolgáltatás érhető el az Interneten, amelyet úgy lehet igénybe venni, hogy fel kell tölteni azt a kép file-t, vagy pdf állományt, amelyre szöveg file-ként van szükség, majd a feldolgozott eredmény file letölthető szintén az Interneten keresztül. Vagy a jobb minőségű szolgáltatásért, vagy ha nagyobb méretű a file, általában fizetni kell.

Hasonlóan szolgáltatásként érhető el a Microsoft Office 365 nevű terméke: (<http://www.microsoft.com/hu-hu/office365/online-software.aspx>), illetve a Google cég Google Docs szoftvere (<https://docs.google.com>). Jellemző, hogy mivel a szoftver a szolgáltatónál fut, csak annyit tud, amire megírták. Amíg a szolgáltató nem bővíti a szoftver lehetőségeit, addig ez nem is változik. Fizetni a szoftverért a felhasználás mértékének megfelelően kell. Ugyancsak jellemző, hogy az ilyen szoftverek nem veszik figyelembe a felhasználó egyedi igényeit. Nem egyedi megrendelésre készülnek, cél minél több példányban eladni. [2]

3.2 PaaS (Platform as a Service)

Sok esetben a felhasználó rendelkezik valamilyen szoftverrel, de nem rendelkezik megfelelő informatikai eszközzel, és azt működtető operációs rendszerrel. Szolgáltatásként azonban igénybe vehet ilyet. A legegyszerűbb ezt úgy elképzelni, hogy a felhasználó megkapja egy megfelelően specifikált virtuális gép elérhetőségét, amire az igényének megfelelő operációs rendszer már telepítve van. Erre felinstallálja saját szoftverét, és használja azt, majd fizet érte. Előnye a szolgáltatásnak, hogy nincs valódi beruházás, az operációs rendszert nem kell megvásárolnia, gyakorlatilag bérlő az egész rendszert. Nem kell szerverszobát kiépíteni, klimatizálást megvalósítani. Nincsenek fenntartási költségek [2].

3.3 IaaS (Infrastructure as a Service)

Ez annyiban különbözik a PaaS-tól, hogy nincs operációs rendszer, csak magát az infrastruktúrát veszi igénybe felhasználó, és azért fizet [2]. Fontos, hogy ebben az esetben (is) pontosan meg kell adnia, mit kell tudnia a hardvernek (memória méret, hálózati sebesség, tárolási és számítási kapacitás, stb.). A felhasználó valójában nem fogja tudni azt, hogy a szolgáltató ezt milyen módon biztosítja neki. Elképzelhető hogy több virtuális gép segítségével áll össze az igényelt teljesítmény. A biztosított gép rendszerfelügyelete, üzemeltetése már a felhasználó feladata.

4. BIZTONSÁGI KÉRDÉSEK

Felhő alapú szolgáltatások esetében fontos, hogy az igényelt szolgáltatás használatakor keletkező eredményhez ki férhet hozzá. Az eredmény ebben az esetben nem feltétlenül számítási eredmény, file tárolás esetében „eredmény” a feltöltött file. Három jellegzetes esetet szokás elkülöníteni:

- magán felhő (private cloud)
- közösségi felhő (Community cloud)
- nyilvános felhő (public cloud)

Magán felhő esetében szolgáltatást csak egy felhasználó érheti el. Erről a szolgáltatónak kell gondoskodnia. A biztosított szolgáltatás mások számára nem szabad, hogy hozzáférhető legyen. Sok esetben ezt a cég tűzfalán belül oldják meg a biztonság érdekében. Itt lehet a legkevésbé a költségeket megosztani, ezért a felhasználó számára általában ez a megoldás a legköltségesebb. A tűzfal azonban többnyire védelmet nyújt a kívülről érkező támadások ellen.

Közösségi felhő esetében azonos feladatot ellátó felhasználók esetében szokás beszélni. Például egyetemi felvételi esetében a felvételizőkről információkat itt helyezhet el a központi szervezet. A felhasználóknál levő alkalmazásoknak illeszkedniük kell ehhez a megoldáshoz, és komoly problémát okozhat, ha az egyes felhasználók mások adataihoz (is) hozzáférnek (kiderül, ki hova felvételizik még).

Nyilvános felhő esetében többen is hozzáférhetnek a szolgáltatáshoz. A szolgáltató feladata annak biztosítása, hogy az egyes felhasználók adatai ne keveredjenek össze, illetve mindenki csak a saját adataihoz férhessen hozzá. Komoly problémát okozhatna, ha például egy szövegfelismerésre feltöltött képfőle esetében nem a saját felismert szöveg érkezne vissza, hanem egy másik cég szerződés tervezete, annak minden bizalmas adatával.

Ilyen nyilvános felhőre épül több elterjedt, részben ingyenes tárolási szolgáltatás: a Microsoft Skydrive-ja,

Google Drive-ja, vagy a szintén sokak által használt Dropbox, illetve SugarSync. Mindegyik szolgáltató esetében kritikus, hogy mennyire megbízható az adatok bizalmas kezelése.

Általában leszögezhető, hogy valódi biztonságot nem az egyedi megoldások jelentenek, hanem a szabványos biztonsági megoldások helyes alkalmazása. Olyan szolgáltatót érdemes választani, amelyik garantálja az adatok bizalmas kezelését, és azt, hogy az adatok nem lesznek jogosulatlanul hozzáférhetőek.

Javasolt a felhő alapú szolgáltatások lépésenkénti bevezetése. Először olyan felhő alapú szolgáltatásokra érdemes áttérni, ahol a legkevesebb biztonsági probléma merülhet fel (például nincsnek szigorú törvényi kötelezettségek, mint a személyes adatok kezelése esetében).

A szolgáltatások jellemzően egy előzetes szerződés kötés után lesznek elérhetőek, A szerződésnek célszerű arra kitérnie, hogy egyes problémás bekövetkezése esetén (például adatvesztés) ki milyen mértékben felelős, valamint arra, hogy a szolgáltatás használatához kinek milyen módon kell hozzájárulnia (kinek a feladata például egy PaaS esetén a rendszeres archiválás).

A szolgáltatást biztosító számítógépeket természetesen érhetik a szokásos, elterjedt támadások:

- a DoS támadás (a szolgáltatás lassítása, leállítása),
- a spoofing (lehallgatással történő információ szerzés),
- a hálózati forgalom meghamisítása,
- a megszemélyesítés, stb.

Ezek ellen a szolgáltatónak megfelelő módon védekeznie kell, illetve a szerződésnek megfelelően segítséget nyújtania a felhasználó számára, hogy abból az irányból a szolgáltatás lehetőleg ne legyen támadható.

5. ÖSSZEGRZÉS

A felhő alapú szolgáltatások előnyök mind a szolgáltatást biztosítók, mind az azt igénybevevők számára. A számítási felhők elterjedése egyre gyorsabb. Ennek köszönhetően várhatóan egyre szélesebb körben fogják a szolgáltatást biztosító eszközparkot illetve a felhasználók eszközeit támadni. A támadás célja vagy információ szerzés, vagy az információk megváltoztatása, törlése. A felhasználók egyik legfontosabb feladata a szolgáltatást biztosító cégek megfelelő kiválasztása, helyes szolgáltató szerződés kötése, megfelelő biztonsági szabályzat kialakítása, betartása. [3]

6. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

7. IRODALOM

- [1.] <http://hu.wikipedia.org/wiki/Moore-törvény>
- [2.] <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
(A NIST Definition of Cloud Computing)
- [3.] <http://csrc.nist.gov/publications/nistpubs/800-144/SP800-144.pdf>
(Guidelines on Security and Privacy in Public Cloud Computing)

TERMELÉSTERVEZÉSI ÉS -IRÁNYÍTÁSI FELADATOK MEGOLDÁSA TÖBBCÉLŰ KERESÉSI MÓDSZER ALKALMAZÁSÁVAL

SOLVING PRODUCTION PLANNING AND CONTROL TASKS USING MULTI-OBJECTIVE SEARCH METHOD

Dr. Kulcsár Gyula^{}, Dr. Bikfalvi Péter^{**}*

ABSTRACT

The paper outlines a new approach to solve some tasks of production planning and control in discrete production processes. A mathematical model based on relative comparison is developed to support qualifying solutions according to multiple objectives. The paper presents the details of the mathematical model and outlines the basic methods of the proposed approach for solving production fine scheduling problems.

1. BEVEZETÉS

Napjainkban a korszerű vállalatirányítási tevékenység kiterjed a vállalat működésével összefüggő összes területre. Egy vállalat egészének működését és annak folyamatait célszerű a tervezés, az előkészítés, a végrehajtás és az ellenőrzés szakaszaira bontani.

Az ipari gyártási technológiák egy részében a termékegyedek és a folyamatlemek térben és időben egymástól jól elkülöníthetők. Az ilyen jellegű technológiai folyamatokra alapozott termelést szokás diszkrét termelési folyamatnak nevezni (ilyen például az alkatrészgyártás és szerelés).

Általában a diszkrét termelési folyamatok tervezése és irányítása több, hierarchikusan egymásra épülő szinten megy végbe. A magasabb szint hosszabb időszakot fog át és csak a fontosabb szempontokkal foglalkozik. Az alacsonyabb szinten a felsőbb szintről kapott irányítási döntések alapján hozzák meg az egyre részletesebb, de egyre rövidebb időperiódusra vonatkozó irányítási döntéseket. Ha az alacsonyabb szint megoldhatatlan feladatot kap (például azért, mert a körülmények közben megváltoztak), akkor visszajelez a közvetlenül felette lévő szintre, ahol módosítást kell végrehajtani.

A gyakorlatban általában jól elkülönül a hosszú távú stratégiai tervezés, a közép távú aggregált tervezés valamint a rövid távú termelésütemezés, -programozás és végrehajtás feladatköre.

2. DISZKRÉT TERMELÉSI FOLYAMATOK TERVEZÉSE

Vállalati szinten, időben, az éves gazdálkodási és termelési terv elkészítése az egyik legmeghatározóbb tervezési szakasz. Ez, a vállalati stratégiának alávetve, a vállalati profitorientált gazdasági célok megvalósítását tűzi ki. Az éves tervezés jellemzően a konkrét vevői rendelések és a várható értékesítések prognózisadatainak alapján készített különböző tervvariációk integrálását jelenti. A termelés szempontjából a vállalati vezetés ebben a szakaszban egy olyan tételes végtermék-kiszállítási (aggregált) tervet készít, amelynek az erőforrások oldaláról reális fedezete van.

Az így kialakított termelési főterv (Master Plan) jellemzően nem csak a végtermékekre, hanem a tartalék szerelvényekre és szerviz alkatrészekre is vonatkozik.

Az aggregált tervezés második fázisa a termelési főterv primér szükségleteiből részben a konkrét piaci igényeknek (határidőknek) megfelelően, részben a különböző termékek gyártmányszerkezetének lebontása után a készletek alakulását figyelembe véve sorozatos „nettósítást” végez, megadva minden egyes tétel ún. nettó anyag- és erőforrás szükségletét.

Az aggregált termelési tervezési és -irányítási feladatok megoldása néhány kulcsfontosságú teljesítménymutató mentén szokott megvalósulni. Ilyen mutatók utalhatnak például a (vevők által elfogadott) szállítókészségre, a (minél alacsonyabb) készletszintre és a (vállalati vezetés által elfogadható) kapacitások kihasználtságára [1].

A vállalatirányításban használt modern ERP (Enterprise Resource Planning) rendszerek ezt a feladatot jellemzően elfogadható módon el tudják végezni. Napjainkban az anyagszükséglet tervezés által meghatározott rendelési mennyiségek a vásárolt tételeket tekintve általában elfogadhatók és indíthatók, azonban a saját gyártású és/vagy szerelésű tételeknél szükség van a további finomításokra, részletesebb kivitelezési programokra, amelyek figyelembe veszik a rendelkezésre álló berendezések kapacitáskorlátait.

^{*} egyetemi docens, Miskolci Egyetem Alkalmazott Informatikai Tanszék

^{**} egyetemi adjunktus, Miskolci Egyetem Alkalmazott Informatikai Tanszék

3. DISZKRÉT TERMELÉSI FOLYAMATOK IRÁNYÍTÁSA

A diszkrét gyártási folyamatok alapeleme a művelet. A műveletek meghatározott gépeken, berendezéseken, munkahelyeken meghatározott sorrendben, különböző szerszámok, készülékek használatával, meghatározott technológiai adatokkal végezhető el. A műveletek között rendszerint bonyolult kapcsolatrendszer áll fenn (pl. megelőzési relációk, stb.). A műveletek végrehajtását a munkahelyek (pl. kézi munkahelyek, automatizált technológiai berendezések, stb.) valósítják meg. A munkahelyek rendszerint nagyobb irányítási egységekbe (gyártócsoportokba, gyártócellákba, gyártórendszerekbe, gyártóműhelyekbe) szervezve funkcionálnak. Az ilyen összetett, sok erőforrásból felépülő gyártórendszerek irányítása során fellépő részfeladatokat szoros integrációban célszerű megoldani. Ennek megfelelően, a korszerű gyártásirányítás legfontosabb feladataihoz tartozik: (1) a középtávú termelési tervek rövidtávú feladatokra bontása, a feladatok ütemezése, finomprogramozása; (2) a feladatok végrehajtásához szükséges anyagi, személyi és információs feltételek biztosítása; (3) a feladatok eszközökhöz rendelése, kiosztása és elindítása; (4) a folyamatok valós idejű felügyelete, megfigyelése és irányítása; (5) a végrehajtás minőségének biztosítása, (6) teljesítménymutatók számítása és az eredmények értékelése; (7) a bizonytalanságok és a váratlan események kezelése.

A gyártásirányítási feladatok támogatására több funkcionális komponensből álló MES (Manufacturing Execution System) rendszerek alakultak ki. A MES célja a gyártási folyamatok végrehajtásának optimális irányítása, az aktuális üzleti és műszaki célok lehető legjobb megvalósítása. Ennek megfelelően a MES hatóköre alapvetően a belső rendelések kibocsátásától a késztermék előállításáig terjed és kettős szerepet játszik. Egyrészt aktuális adatok gyűjtésével, ellenőrzésével, elemzésével és felhasználásával gondoskodik a tervezett tevékenységek időbeli ütemezéséről, elindításáról, irányításáról, a végrehajtás felügyeletéről, másrészt a termelési aktivitásokról feladat-specifikus információkat szolgáltat a vállalat döntési folyamatainak támogatásához.

A MES-funkciók egy többszörösen visszacsatolt, bonyolult szabályozási rendszert alkotnak. A különböző modulok illesztése, konfigurálása és finomhangolása a kereskedelmi MES rendszerek alkalmazásának egyik elengedhetetlen feltétele. Mivel a technológiai folyamatok nagyon különbözőek lehetnek, ezért hatékony irányításuk is különböző modelleket igényel. Az ilyen rendszerek funkcióinak és szolgáltatásainak fejlesztése széleskörű elmélet- és gyakorlatorientált kutatás tárgyát képezi napjainkban is.

4. MEGOLDÁS-VÁLTOZATOK ÉRTÉKELÉSÉNEK MATEMATIKAI MODELLJE

A termelésstervezés és –irányítás különböző szintjein jelentkező optimalizálási feladatokban fontos szerepet játszik a különböző megengedett megoldás-változatok összehasonlítását és értékelését támogató modell. Az olyan optimalizálási feladatokban, amelyekben az optimalizálás célja egyetlen célfüggvénnyel megadható, két különböző megoldás minőségének összehasonlítása egyszerű, mert elegendő a különböző megoldások célfüggvény-értékeit kiszámítani és azok alapján könnyen számszerűsíthető a minőségbeli különbség.

Sokkal nehezebb a különböző megoldások minőségének összehasonlítása a többcélú optimalizálási feladatokban. Több optimalizálási cél esetén ugyanis különböző dimenziójú és értékészletű célfüggvények szerepelhetnek, amelyek gyakran egymással valamilyen (legtöbbször pontosan nem is ismert vagy nehezen értelmezhető) kölcsönkapcsolatokban állhatnak. Az ilyen feladatoknak csak kivételes esetekben van olyan megoldása, amely az összes célfüggvény szempontjából egyszerre tekinthető optimálisnak. Kompromisszumos megoldás értelmezéséhez és megtalálásához szükséges annak definiálása, hogy milyen szempontok szerint és hogyan hasonlíthatók össze a lehetséges megoldások.

Egy új szemléletű, a különböző szoftverekben rugalmasan és könnyen adaptálható matematikai modellt fejlesztettünk ki a különböző típusú, értékészletű és fontosságú összetevőkből felépülő változatos célfüggvény-rendszereket kezelésére. A módszer alapelve a következő: Két megoldás jóságának az összehasonlítása nem a megoldások külön-külön vett abszolút jóságának valamilyen módszerrel végrehajtott számszerűsítésén, majd ezek összehasonlításának eredményén alapul, hanem az egyik megoldásnak a másikkal viszonyított (relatív) jósága kerül számszerűsítésre, és ennek alapján dönthető el az, hogy melyik tekinthető jobb megoldásnak.

Jelölje S a megengedett megoldások halmazát, és fogalmazzuk meg a K darab különböző célfüggvényt minimalizálandó alakban a következőképpen:

$$f_k : S \rightarrow \mathbb{R}^+ \cup \{0\}, \forall k \in \{1, 2, \dots, K\}. \quad (1)$$

Jelentsen s_x és s_y két különböző lehetséges megoldást, melyek a teljes S megoldási térben értelmezettek, továbbá legyen w_k az f_k célfüggvény fontosságát kifejező nem negatív egész érték (prioritásérték). Vezessük be a következő kétváltozós függvényeket:

$$F : S^2 \rightarrow \mathbb{R}, F(s_x, s_y) = \sum_{k=1}^K (w_k \cdot D(f_k(s_x), f_k(s_y))), \quad (2)$$

$$D : \mathbb{R}^2 \rightarrow \mathbb{R}, D(a, b) = \begin{cases} 0, & \text{ha } \max(a, b) = 0 \\ \frac{b-a}{\max(a, b)}, & \text{egyébként} \end{cases} \cdot (3)$$

Ezek alapján az $F(s_x, s_y)$ előjeles függvényérték jelenti az s_y megoldásnak az s_x megoldáshoz viszonyított relatív jóságát, melynek felhasználásával az alapvető relációs operátorok ($<$, \leq , $>$, \geq , $=$, \neq) értelmezése kiterjeszthető az S -beli s_x és s_y megoldásokra a következőképpen:

$$(s_x \ ? \ s_y) := (F(s_x, s_y) \ ? \ 0). \quad (4)$$

Ha az $F(s_x, s_y)$ kétváltozós függvény értéke negatív, akkor az s_x jobb megoldás mint az s_y , ha pozitív, akkor rosszabb. Nulla érték esetén egyformán jó a két megoldás.

Az ilyen módon – a célfüggvények értékeire alapozva közvetlenül a lehetséges feladatmegoldások halmazán – értelmezett relációs operátorok felhasználásával lehetővé válik a különböző megoldások páronkénti összehasonlítása és értékelése. Ezáltal a valós számok körében alkalmazott relációs operátorokkal teljes mértékben megegyező módon, különböző stratégiák szerint kiválasztható a legjobb megoldás (ha csak egyetlen optimum létezik) vagy egy megoldás a legjobb megoldások halmazából (ha több optimum létezik).

A D függvényben a max helyett más, alkalmasan megválasztott függvény is használható (pl. $(a+b)/2$, stb.). A módszer lényege – a páronkénti relatív minősítés elve – megmarad. A módszer még tovább általánosítható, ha a mindenkor optimalizálási feladatban szereplő célfüggvények jellegzetességeit figyelembe véve, komponensenként (vagy komponenscsoportonként) eltérő D függvényvariánsok szerepelnek az F függvényben.

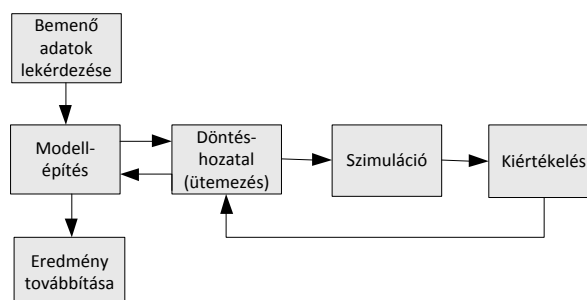
A módszer többféle feladat-megoldási stratégiában is jól felhasználható az aktuális céloknek legjobban megfelelő megoldás megtalálására. Ez azt jelenti, hogy a módszer széles problémakörben felhasználható ismert keresési algoritmusok és meta-heurisztikák (pl. tabukeresés, szimulált hűtés, genetikus és evolúciós algoritmusok, stb.) részeként, kombinatorikus többcélú optimalizálási feladatok megoldására.

5. TÖBBCÉLÚ KERESÉSI MÓDSZERRE ALAPOZOTT TERMELÉSPROGRAMOZÁS

A diszkrét termelési folyamatok műhelyszintű előidejű (prediktív) és valós idejű (reaktív) ütemezése mind elméleti, mind gyakorlati szempontból az erősen modellfüggő és komplex kombinatorikus optimalizálási feladatok közé tartozik. Ennek következtében nagyon sokféle ütemezési feladattípus ismert [6]. A napjaink ipari gyakorlatában egyre fontosabbá váló, rugalmas és igény szerinti gyártási folyamatok irányítása szükségessé teszi az ismert ütemezési modellek további jelentős kiterjesztését annak érdekében, hogy olyan szoftvermodulokat tudjunk implementálni, amelyek a különböző modellváltozatokat is képesek hatékonyan támogatni.

A termelési folyamatok finomprogramozásakor a termelő rendszer lehetőségeit, képességeit és korlátait figyelembe véve, a belső rendelések teljesítéséhez szükséges munkák időbeli végrehajtását kell megtervezni. Ennek során a szükséges erőforrások allokálását és a feladatok végrehajtásának indítási időpontját kell megtervezni úgy, hogy a vállalat magasabb szintjén megfogalmazott célok és a gyártásirányítás járulékos saját belső céljai egyaránt megvalósuljanak.

A különböző ütemezési feladat-osztályokba besorolható feladatok megoldására egy többcélú, integrált, heurisztikus keresési technikára alapozott megoldási módszert fejlesztettünk ki. Ennek részletei megtalálhatók pl. a [3], [4] és [5] cikkekben. A megoldási koncepció elvi vázolata az 1. ábrán látható.



1. ábra. Szimulációra alapozott termelésprogramozás

A termelésinformatikai rendszer adatbázisában tárolt adatok és az interaktív felhasználói felületen keresztül beállított adatok felhasználásával egy modellépítő komponens definiálja a rendszerben lévő objektumokat. Kiemelt fontosságú feladata a belső rendelésekhez kapcsolódó munkák és az érvényben lévő korlátozások definiálása. A korlátozások közül az előírt vagy számítható belső határidőket „puha” (adott esetben megsérthető) korlátozásoknak tekintve alkalmas célfüggvények megválasztásával a csúszások és késések minimalizálása ütemezési kritériumként jelenik meg. A további (pl. technológiára, anyagmozgatásra, erőforrásra, végrehajtásra vonatkozó stb.) korlátozások „kemény” (meg nem sérthető) előírásokká válnak. A szükséges rendelkezésre állási, alkalmazhatósági és megvalósíthatósági vizsgálatok elvégzését követően a modellépítő komponens felépíti a modellobjektumok között fennálló teljes kapcsolatrendszerét. Ennek elsődleges célja az, hogy a feladatmegoldás során minden egyes döntési helyzetben a választható alternatív lehetőségek (a döntési változók megengedett értékei) világosan és könnyen felismerhetők legyenek.

A belső rendelések ütemezési alapegységekre bontásával önálló munkák jönnek létre. Ezek ütemezése önálló független döntési változók értékének megválasztásával valósul meg. Az ütemező modul minden egyes munkához hozzárendel egy megfelelő végrehajtási útvonalat, továbbá hozzárendel egy

megfelelő gépet a kiválasztott útvonal minden egyes végrehajtási lépésének megfelelő gépcsoportból, és meghatározza minden érintett gépen a munkák végrehajtási sorban elfoglalt pozícióját. Ezáltal az ütemező algoritmus a munkákhoz hozzárendeli a konkrét feladatlistát, és így a gépeken a gyártási sorozatnagyságok és az azokat elválasztó átállítási műveletek a döntési változók függvényében ütemezés közben alakulnak ki. Nincs szükség előzetes sorozatnagyság tervezésre, ez a funkció az ütemező hatáskörébe tartozik. A gyártási modelltípustól függően bizonyos esetekben az ütemező az erőforrások rendelkezésre állási időintervallumait is döntési változóként kezeli (integrált műszakbeosztási funkció), míg más esetekben ezek egy bemenő adathalmazból származtatott korlátozó feltételek formájában jelennek meg. Az ütemezési (döntéshozatali) folyamat eredményeként elkészül egy termelési ütemterv.

Az ütemtervben szereplő feladatok végrehajtásához kapcsolódó időadatok számítását egy megfelelően gyors végrehajtás-vezérelt szimulációs eljárás végzi el. A szimuláció figyelembe veszi az egyes gépek rendelkezésre állási időintervallumait, az egyes gépeken az adott munkák sorrendje által meghatározott átállítási időket, a munka-gép összerendelések alapján számítható megmunkálási időket és az egyéb kapcsolódó járulékos időket. A szimuláció közben ismertté válik az egyes feladatok gépenkénti tervezett – és következményként a munkák, valamint a megrendelések származtatott – indítási és befejezési időpontja.

A szimuláció által számított időadatok és egyéb értékek felhasználásával egy értékelő komponens kiszámítja a megoldásra vonatkozó célfüggvény értékeket és teljesítménymutatókat. Ezek figyelembevételével az aktuálisan vizsgált megoldás és az addig legjobbnak ítélt megoldás egymáshoz viszonyított (relatív) minőségének számszerűsítésével kiértékelésre kerül a megoldás.

Az ütemező modul iteratívan módosítja az aktuális ütemtervet, konzisztens változtatásokkal új megoldás-változatokat készít, majd szimulációt és kiértékelést követően a célfüggvény-értékektől függően tovább folytatódhat a legjobb megoldás keresése.

6. ÖSSZEFOGLALÁS

A cikkben röviden vázoltuk a diszkrét termelési folyamatok számítógépes tervezésével és –irányításával kapcsolatban felmerülő legfontosabb feladatokat. Bemutattunk egy új szemléletű, többcélú megoldás-értékelő modellt, valamint annak keresési algoritmussal és szimulációval kombinált hatékony alkalmazását termelésprogramozási feladatok megoldására.

Munkánk során újabb és újabb gyártási jellemzőket, alternatívákat és korlátozásokat magukba foglaló termelésprogramozási modelleket dolgoztunk ki.

Modelljeink egyre általánosabb érvényességgel definiálják és kezelik a gyártási műveletek jellemzőit, a gyártóeszközök képességeit, a technológiai útvonalak alternatíváit, a szerszámok alkalmazhatóságát, a műveletközi tárolók kapacitását, a műveletvégzési és anyagmozgatási időket.

A kidolgozott modellek és módszerek alapján implementált termelésprogramozó szoftvereink a bemenő adatok formájában megadott aktuális gyártási erőforrás-környezet, belső rendelkezések, korlátozások és célfüggvények alapján képesek automatikusan elkészíteni a kijelölt gépek részletes termelési finomprogramját. A szoftverek automatikus, kézi és kombinált üzemmódokat egyaránt támogatnak.

KÖSZÖNETNYÍLVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

IRODALOM

- [1] BIKFALVI, P., ERDÉLYI, F., TÓTH, T. (2012) Multi-objective Approach for Performance Management of Manufacturing Systems. XXVI. microCAD International Scientific Conference, Miskolc, Hungary, (CD).
- [2] ERDÉLYI F., TÓTH T., KULCSÁR GY., MILEFF P., HORNYÁK O., NEHÉZ K., KÖREI A. (2009) New Models and Methods for Increasing the Efficiency of Customized Mass Production. Journal of Machine Manufacturing, XLIX (E2):11-17.
- [3] KULCSÁR GY. (2011) A Practice-Oriented Approach for Solving Production Scheduling Problems. XXV microCAD International Scientific Conference, Miskolc, Hungary, 61-66.
- [4] KULCSÁR GY., ERDÉLYI F. (2007) A New Approach to Solve Multi-Objective Scheduling and Rescheduling Tasks. International Journal of Computational Intelligence Research, 3 (4):343-351.
- [5] KULCSÁR GY., KULCSÁRNÉ F. M. (2009) Solving Multi-Objective Production Scheduling Problems Using a New Approach. Production Systems and Information Engineering, A Publication of the University of Miskolc, 5:81-94.
- [6] LEI D. (2009) Multi-objective production scheduling: a survey. The International Journal of Advanced Manufacturing Technology, 43 (9-10):926-938.
- [7] VENKATESWARAN J., SON Y.J. (2005) Hybrid system dynamic—discrete event simulation-based architecture for hierarchical production planning. International Journal of Production Research, 43 (20):4397-4429.

GYÁRTÓCELLÁK KIALAKÍTÁSA META- HEURISZTIKUS MÓDSZEREK ALKALMAZÁSÁVAL

METAHEURISTICS FOR SOLVING THE CELL FORMATION PROBLEM

Körei Attila⁽¹⁾, Tóth Tibor⁽²⁾

ABSTRACT

One of the most important tasks in Cellular Manufacturing is determination of the part families and the corresponding machine cells. The methods for solving the problem usually start from the machine-part incidence matrix. As the machine-part grouping problem is a NP complete problem, recently the research focuses on heuristic methods. This paper introduces some of these new techniques.

1. BEVEZETÉS

A csoporttechnológia (Group Technology, GT) az alkatrészgyártás átfogó szervezési elve, melynek lényege a különböző alkatrészek, gépek, folyamatok közötti hasonlóságok felismerése és e tudás hasznosítása a gyártási folyamatban. A csoporttechnológia egyik alapfeladata a csoportos rendszerű termelés megvalósítása, ezen belül a gyártócellák és alkatrészcsaládok egymáshoz rendelésének számítógépes támogatása. Ebben a folyamatban a csoporttechnológia filozófiája a következő módon érvényesül: azok az alkatrészek kerülnek egy családba, melyek megmunkálási igénye hasonló, az azonos, vagy hasonló technológiával készülő alkatrészcsoportok előállítására alkalmas gépeket pedig összevonjuk egy gyártócellába. A csoporttechnológia elveinek bevezetésével a termelés irányítása egyszerűbbé válik, a specializálódott munkahelyek révén magasabb termelési színvonal érhető el, a szerszámozás és készülékezés költségei csökkenthetőek, az anyagmozgatási távolságok rövidülnek, a termékáramlás egyenletesebbé válik [12].

(1) egyetemi docens, Miskolci Egyetem, Alkalmazott Matematikai Tanszék

(2) egyetemi tanár, Miskolci Egyetem, Alkalmazott Informatikai Tanszék

2. A FELADAT MODELLEZÉSE GÉP- ALKATRÉSZ INCIDENCIA-MÁTRIX SEGÍTSÉGÉVEL

Tekintsünk egy gyártórendszert, melyben n darab különböző gép k számú, különböző alkatrész előállítását végzi. A megmunkálási viszonyokat az $M=[m_{ij}]$, $n \times k$ típusú, 0 és 1 elemeket tartalmazó gép-alkatrész mátrix segítségével írjuk le, ahol $m_{ij}=1$, ha az i -edik gép részt vesz a j -edik alkatrész megmunkálásában, egyébként pedig $m_{ij}=0$. Például, az 1. táblázatban szereplő mátrix 7 gép és 8 alkatrész viszonyát szemlélteti:

M		Alkatrészek							
		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
Gépek	g_1	0	1	1	0	1	0	0	0
	g_2	1	0	0	0	0	1	0	0
	g_3	0	0	0	1	0	0	1	0
	g_4	1	0	0	0	0	1	0	0
	g_5	0	0	1	1	1	0	0	1
	g_6	0	0	0	1	0	0	0	0
	g_7	0	1	1	0	1	0	0	1

1. táblázat. Példa gép-alkatrész mátrixra

Célunk a hasonló megmunkálási igényű alkatrészek csoportjainak (az alkatrészcsaládoknak) és az azokat megmunkáló gépek együtteseinek (a gyártócelláknak) a meghatározása oly módon, hogy az egyes cellákon belül a gépek kihasználtsága a lehető legnagyobb legyen, a cellák közötti alkatrészmozgás pedig a lehető legkevesebb. A megoldást a gép-alkatrész mátrix sorainak és oszlopainak alkalmas átrendezése után kaphatjuk meg, az átrendezett mátrix főátlója mentén kialakult blokkok elemeinek leolvasásával. A gépek

kihasználtsága akkor a legnagyobb, ha a lehető legkevesebb 0 szerepel blokkokon belül, a cellaközi mozgás pedig akkor minimális, ha minél kevesebb 1-es áll a blokkokon kívül. A sorok és az oszlopok alkalmas átrendezésével a 2. táblázatból kiolvasható mátrixot nyerjük. A táblázatbeli mátrix „főátlója” mentén kialakult három részmatrix kijelöli a keresett csoportokat, hiszen jól elkülöníthetőek a $\{g_1, g_5, g_7\}$, $\{g_2, g_4\}$ és $\{g_3, g_6\}$ gyártócellák, valamint a hozzájuk tartozó $\{a_2, a_3, a_5, a_8\}$, $\{a_1, a_6\}$ és $\{a_4, a_7\}$ alkatrészcsaládok.

M		Alkatrészek							
		a_2	a_3	a_5	a_8	a_1	a_6	a_4	a_7
Gépek	g_1	1	1	1	0	0	0	0	0
	g_5	0	1	1	1	0	0	1	0
	g_7	1	1	1	1	0	0	0	0
	g_2	0	0	0	0	1	1	0	0
	g_4	0	0	0	0	1	1	0	0
	g_3	0	0	0	0	0	0	1	1
	g_6	0	0	0	0	0	0	1	0

2. táblázat. Az átrendezett gép-alkatrész mátrix

Az egyes megoldások minőségének meghatározására számos mérőszám ismeretes az irodalomban, az egyik legáltalánosabban használt mutató a *Kumar* és *Chandrasekharan* által bevezetett csoportképzési hatásfok (*grouping efficacy*), amely következő képlettel számítható:

$$GE = 1 - \frac{n_0 + n_k}{n_0 + n_1},$$

ahol n_1 az 1-esek száma az eredeti gép-alkatrész mátrixban, n_0 a 0-k száma a kialakított csoportokon belül, n_k pedig összegzi azon eseteket, amikor egy alkatrészt saját csoportján kívüli gépnek is meg kell munkálnia [8]. A 2. táblázatban szereplő megoldás csoportképzési hatásfoka:

$$GE = 1 - \frac{3 + 1}{3 + 17} = 0,8.$$

Ideális esetben az egyes cellákba tartozó gépek a cellához rendelt alkatrészcsalád összes elemét megmunkálják, és nincs egyetlen kivételes alkatrész sem (azaz olyan alkatrész, amit egy másik gyártócellába is át kell vinni megmunkálásra), tehát $n_0=0$ és $n_k=0$, így a maximális csoportképzési hatásfok értéke 1.

A gép-alkatrész mátrix a valós gyártókörnyezetek erősen egyszerűsített modelljének tekinthető, melyben számos fontos paraméter figyelembevételére nincs lehetőség (pl. várakozási idők, átállítási idők, gépek általános kihasználtsága); abban azonban a kutatók nagy része egyetért, hogy a mátrix blokk-diagonális alakra hozása úgy, hogy az n_0 és n_k paraméterek értékét minimális szinten tartsuk, kulcsfontosságú mozzanat a hatékony cellakialakítás folyamatában.

3. METAHEURISZTIKUS MÓDSZEREK

Ballakur és *Steudel* megállapították, hogy a gyártócellák és alkatrészcsaládok meghatározásának feladata az NP - teljes problémák családjába tartozik, azaz az optimális megoldás kiszámításához szükséges idő mennyisége a feladat méretének exponenciális függvénye [1]. Az ilyen jellegű feladatok megoldására az egzakt módszerek általában nem hatékonyak, ezért helyettük olyan közelítő algoritmusokat használunk, amelyek ha nem is optimális, de kielégítő megoldást szolgáltatnak elfogadható időn belül. Az ilyen, kvázioptimális megoldásokat előállító metaheurisztikák olyan kereső eljárások, amelyek a megengedett megoldások halmazán (a keresési térben) iteratív módon haladva egyre jobb megoldásokat állítanak elő, mindaddig, míg valamilyen stacionárius állapotot el nem érnek, vagy a keresés meg nem halad egy adott időkorlátot. Nevüket onnan kapták, hogy a keresési tér következő lépésben kiértékelendő csomópontjait valamilyen heurisztika segítségével választják ki, ami az optimális megoldás elhelyezkedésére ad becslést. A metaheurisztikák alapalgorithmusai probléma-függetlenek, ennek köszönhető, hogy rugalmas, széles körben alkalmazható optimalizáló módszerek tartoznak közéjük [4]. Az egyes konkrét feladatokra vonatkozó, probléma-specifikus tudást elegendő a keresési térben való lépéseket kijelölő, heurisztikus kiértékelő függvény definíciójában megadni. A gyártócella-kialakítás legújabb megoldási módszereit átfogó módon bemutató [6] publikáció tanúsága szerint a metaheurisztikák közül a tabu-keresés, a szimulált hűtés algoritmus és a populáció-alapú algoritmusok különböző változatai a feladat megoldására leggyakrabban alkalmazott módszerek. Ugyanerre a következtetésre jutottak a [10] és [11] áttekintő cikkek szerzői is. A továbbiakban röviden bemutatunk néhány populáció-alapú algoritmust, különös tekintettel a csoportképző genetikus algoritmusra, mellyel az áttekintő cikkek sem foglalkoznak jelentőségének megfelelő mélységben.

3.1. Evolúciós algoritmusok

Az evolúciós algoritmusok egy adott rendszer fokozatos átalakulásának, fejlődésének a folyamatát

modellezik. Ennek a folyamatnak minden állapotához hozzátartozik a megoldandó feladat lehetséges megoldásainak egy halmaza (egy populáció), melynek elemei az egyedek. A módszer a természetes kiválasztódás mintájára a legrátermettebb egyedek túlélésének biztosításával igyekszik az adott feladat optimálshoz közeli megoldását előállítani. Az egyes populációk manipulálására (az újabb generációk előállítására) három alapvető operátort használnak: a szelekciót, a keresztezést és a mutációt. A szelekció során valamilyen előre meghatározott stratégia alapján megtörténik a legrátermettebb, a szaporodásra érdemesnek tartott egyedek kiválogatása. A keresztezés két egyed véletlenszerű kombinálása egy vagy több új egyed létrehozása érdekében. A mutáció az egyedek kismértékű, véletlenszerű megváltoztatását jelenti, azzal a céllal, hogy megakadályozza a populáció belterjességét és lehetőség nyíljon a keresési tér addig feltáratlan részeire való eljutásra. Egy adott egyed rátermettségének mérésére a fitness függvényt használjuk, ez az a heurisztika, mely a megoldások minőségét megadja: minél nagyobb az egyed fitness értéke, annál jobb megoldást reprezentál. Az evolúciós algoritmus első lépéseként általában véletlen generálással létrehozuk a kezdő populáció egyedeit, és kiszámítjuk minden egyed fitness értékét. Ezután kiválasztunk adott számú egyedet, melyeken végrehajtjuk a keresztezést. A szülők és utódok halmazából pedig kiválogatjuk az új generáció egyedeit. Ezt a folyamatot addig ismételjük, amíg az előre beállított megállási feltétel nem teljesül, ami lehet például egy meghatározott generációs szám, fitness érték vagy maximális futási idő elérése, vagy az, hogy az egymást követő generációk fitness értékei között már minimális szintű az eltérés.

Az evolúciós algoritmusoknak több alaptípusa is ismert, melyek az egyedek reprezentálására szolgáló módszerben vagy az alkalmazott operátorokban térnek el egymástól. A gyártócella-kialakítás feladatában leggyakrabban a genetikai algoritmus valamelyik változatát alkalmazzák, ahol az egyedek, vagyis a feladat lehetséges megoldásai bitsorozatként, vagy egy véges abc feletti szavak formájában kódoltak. Például a korábbi példa hét gépet tartalmazó rendszerében a

g_1	g_2	g_3	g_4	g_5	g_6	g_7
1	2	3	2	1	3	1

táblázat alsó sorában szereplő sorozat reprezentálja a 2. táblázatban feltüntetett megoldást: három gyártócellát hozunk létre, az elsőben a g_1, g_5, g_7 jelű gépekkel, a másodikba g_2 és g_4 tartozik, a harmadik gyártócellába pedig g_3 és g_6 kerül.

A klasszikus genetikai algoritmus bevezetése *John Holland* nevéhez fűződik [7]. *Falkenauer* [5]-ben

azonban rámutatott, hogy a genetikai algoritmus klasszikus változata kevésbé hatékony az olyan feladatokban, ahol adott objektumok közül az egymáshoz leginkább hasonló csoportjait kell kialakítanunk. Problémát okoz például a kódolásból eredő redundancia, hiszen ha tekintjük az előző táblázat kódsorában szereplő elemek permutálásával nyert

g_1	g_2	g_3	g_4	g_5	g_6	g_7
2	1	3	1	2	3	2

táblázatot, láthatjuk, hogy ugyanúgy a $\{g_1, g_5, g_7\}$, $\{g_2, g_4\}$, $\{g_3, g_6\}$ megoldást reprezentálja. A gépek számának növekedésével ez a hatás még erőteljesebben jelentkezik, jelentősen megnehezítve ezzel a keresési tér bejárását.

Falkenauer a problémák kiküszöbölésére a genetikai algoritmus egy speciális változatát, a csoportképző genetikai algoritmus alkalmazását javasolta (*Grouping Genetic Algorithm*), néhány jellegzetes feladatban megmutatva, hogy az új változat hatékonyabb a standard genetikai algoritmusnál. A gyártócella-kialakítás feladatára a csoportképző genetikai algoritmust először *De Lit*, *Falkenauer* és *Delchambre* alkalmazta [3]. A klasszikus genetikai algoritmus és csoportképző változata között két alapvető eltérés van: egyrészt a kódolásban, másrészt az alkalmazott genetikai operátorokban. A kódolást tekintve a csoportképző genetikai algoritmus

$$a_1 a_2 a_3 \dots a_k | g_1 g_2 g_3 \dots g_n | c_1 c_2 c_3 \dots c_S$$

alakú kromoszómákat használ a megoldások reprezentálására, ahol S a megoldásban szereplő gyártócellák és egyúttal alkatrészcsaládok száma, c_i a gyártócella és a hozzátartozó alkatrészcsalád közös sorszáma. Az a_i érték megmutatja, hogy az i -edik alkatrész melyik sorszámú csoportba került ($i=1,2,\dots,k$), hasonlóképpen g_j ($j=1,2,\dots,n$) jelenti a j -edik gép cellájának a sorszámát. Például a 2. táblázatban szereplő megoldást a

$$21131231 | 1232131 | 123$$

sorozat kódolja. Vegyük észre, hogy míg a kód első két része rögzített hosszúságú, és értékeit a $\{c_1, c_2, c_3, \dots, c_S\}$ halmazból veszi, addig a kód harmadik részének hosszúsága változó lehet. Mivel a genetikai operátorok működése leginkább erre a harmadik részre koncentrálódik, az operátorok definiálásánál speciális szabályokra volt szükség, de éppen ezek a definíciók teszik a klasszikus változathoz hatékonyabbá a csoportképző genetikai algoritmust. A csoportképző genetikai algoritmus operátorainak pontos leírásával szolgál és a gyártócella-kialakítás feladatára való

alkalmazásának tapasztalatairól számol be *Brown* és *Sumichrast* [2] publikációja.

3.2. További metaheurisztikák

A gyártócella-kialakítás feladatának megoldásában használt populációalapú algoritmusok közül három, az utóbbi években kifejlesztett, biológiai ihletésű eljárás alapgondolatát ismertetjük. Az algoritmusok pszeudokódjai és az alkalmazásukat bemutató publikációk felsorolása megtalálható a már említett áttekintő műben [6].

3.2.1. Hangyakolónia alapú optimalizáció (*Ant-Colony Optimization*)

A módszer alapgondolata a hangyák viselkedésének megfigyeléséből adódik: a táplálékszerzés folyamán a hangyák nem véletlenszerűen mozognak, hanem egy idő után megtalálják az élelemforrás és a boly közötti legrövidebb útvonalat. A jelenség magyarázatát a hangyák sajátos kommunikációja adja: a táplálékforrásokhoz vezető útvonalakat a bolyhoz visszavezető úton a hangyák egy speciális kémiai anyag, feromon kibocsátásával jelölik meg. A többi hangya, felismerve egy ilyen megjelölt ösvényt, nagyobb valószínűséggel indul el azon, saját feromonkibocsátásával is növelve annak vonzerejét. Ha egy élelemforráshoz két út is vezet, a rövidebb útvonalon gyakrabban fordulnak a hangyák, fenntartva annak feromon-szintjét. (Ugyanis a feromon egy idő után elpárolog, így a ritkábban használt, hosszabb utat egyre kevesebb hangya választja).

3.2.2. Méh algoritmus (*Bees Algorithm*)

A méhek esetében a táplálékforrások felkutatása a felderítő méhek feladata. A kaptárhoz visszatérve a felderítők gondosan koreografált táncot járnak, ezzel adva a többiek tudtára a nektár lelőhelyét. A méh potrohának különböző ütemű riszálása, valamint az előadást kísérő sípoló hangjelzések a táplálékforrás távolságáról és a lelőhely gazdagságáról is tartalmaznak információkat. Ezt a tudást kihasználva a méhcsalád nagy távolságra levő, gazdag lelőhelyeken is hatékonyan gyűjthet élelmet, a megfelelő létszámú csoport kiküldésével.

3.2.3. A részecske-rajzás algoritmus (*Particle Swarm Algorithm*)

Az algoritmus az élelmet kereső madár- illetve halrajok viselkedésének szimulációja alapján alakult ki. Ezekben a csoportokban mindig létezik egy vezér-egyed, akinek a mozgását követik a többiek. Az algoritmus implementálásakor az egyedeket kis részecskék szimbolizálják a paraméterértékek terében, melyek mindegyike irányvektorral és sebességgel rendelkezik. A heurisztikus célfüggvény kiértékelése után az összes részecske az éppen legjobb megoldást adó részecske irányába mozdul el, így kerül a teljes csoport egyre kedvezőbb helyzetbe.

KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1. B-10/2/KONV-2010-0001 jelű projekt részeként, az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

IRODALOM

- [1] BALLAKUR A., STEUDEL H. J. (1987) A within cell utilization based heuristic for designing cellular manufacturing systems. *International Journal of Production Research*, 25, pp.639-665.
- [2] BROWN E. C., SUMICHRASST R. T. (2001) CF-GGA: a grouping genetic algorithm for the cell formation problem. *International Journal of Production Research*, 39, pp.3651-3669.
- [3] DE LIT P., FALKENAUER E., DELCHAMBRE A. (2000) Grouping genetic algorithms: an efficient method to solve the cell formation problem. *Mathematics and Computers in Simulation*, 51, pp.257-271.
- [4] DOMBI J. (2009) *Intelligens rendszerek és alkalmazásaik*. Elektronikus jegyzet, Szegedi Tudományegyetem.
- [5] FALKENAUER E. (1998) *Genetic Algorithms for Grouping Problems*. New York, Wiley.
- [6] GHOSH T., SENGUPTA S., CHATTOPADHYAY M., DAN P. K. (2011) Metaheuristics in cellular manufacturing: A state-of-the-art review. *International Journal of Industrial Engineering Computations*, 2, pp.87-122.
- [7] HOLLAND J. H. (1975) *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Application to Biology, Control and Artificial Intelligence*. Ann Arbor, MI: University of Michigan Press.
- [8] KUMAR C., CHANDRASEKHARAN M. (1990) Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *International Journal of Production Research*, 28, pp.233-243.
- [10] PAPAIOANNOU G., WILSON J. M. (2010) The evolution of cell formation problem methodologies based on recent studies (1997–2008): Review and directions for future research. *European Journal of Operational Research*, 206 (3), pp.509-521.
- [11] SHAHRAM S., SOLIMANPUR M., MAHDAVI I., NIKBAKHSH J. (2010) Heuristic Approaches for Cell Formation in Cellular Manufacturing. *J. Software Engineering & Applications*, 3, pp.674-682.
- [12] TÓTH T. (2004) *Termelési rendszerek és folyamatok*. Miskolci Egyetemi Kiadó.

Felhő technológia a kis és közepes vállalatoknál

CLOUD TECHNOLOGY FOR SMALL AND MEDIUM SIZED ENTERPRISES

Hornyák Olivér, Nehéz Károly**

ABSTRACT

Nowadays cloud computing is a concept, where remote servers and resources are utilized by users to access data or applications. Let us consider for example the popular Gmail services so you can understand the basics of cloud computing. Service providers need to use high capacity servers and enormous hardware capacity to establish quick and efficient client services. They store data for their clients without sharing the complexity of their internal servers with the users. Cloud computing provides load balancing of resources and can quickly react to the fast changes of the dynamic usage. One of its major advantages is, that you can access services, applications, data what you need. Its disadvantage is that we don't know where our data resides physically: in India or in the USA, and we may have security concerns, who can hijack our data. In this paper an overview is given of the resources of Department of Information Engineering at University of Miskolc.

1. BEVEZETÉS

A számítási felhő fogalmát úgy érthetjük meg, ha olyan alkalmazásokra gondolunk, ahol a felhasználó egy fizikailag távoli szerveren elhelyezett alkalmazásokat és erőforrásokat használ [1]. A 60-as években már ismert fogalom volt, ekkor vetették fel azt az elképzelését, hogy a számítástechnikának közüzemi szinten kellene működni. Manapság a Gmail népszerű levelező működési elvére gondolva - felhasználói szemszögből - lényege még könnyebben megérthető [2]. A felhasználó egy kis teljesítményű és tárhelyű eszközön egy web böngészővel felcsatlakozik a szolgáltató szervereire, ahol igénybe veszi annak erőforrásait. A szolgáltatóknál hatalmas kapacitású és mennyiségű hardver eszközre van szükség ahhoz, hogy gyorsan és hatékonyan kiszolgálhassák az ügyfeleiket, valamint azok adatait is hatékonyan tudják tárolni [3]. A felhő egyfajta erőforrás-megosztást is megvalósít és gyorsan alkalmazkodik a változó igénybevételhez. Legfontosabb előnye, hogy bárholnan, bármikor elérjük azokat a szolgáltatásokat, alkalmazásokat, adatokat, melyekre az éppen szükség van [2].

* PhD, egyetemi docens, Miskolci Egyetem, Alkalmazott Informatikai

2. ADATBIZTONSÁG

A felhő alapú szolgáltatások esetén, a szigorú adatbiztonság és titkosítás megléte létfontosságú. Ha egy vállalat egy felhő szolgáltatót kívánja működtetni informatikai rendszerének, alkalmazásainak egy részét, akkor tulajdonképpen az adatokkal nem ő rendelkezik, hanem csak feltölti, szerkeszti és használja azokat. Emiatt bizalmatlanság figyelhető meg a kis és közepes vállalatok részéről. A leggyakoribb kérdések lehetnek: kinek a tulajdonában vannak az adatok? Hogyan szeparálják különböző vállalatok/felhasználók adatait egymástól? Hogyan titkosítja és védi a felhasználók adatait a felhő? Ilyen kérdésekre a válasz sokszor egyértelmű, hiszen már eddig is voltak tárhely szolgáltatók, ahol a titkosításra és az adatok elkülönítésére korábban is szükség volt. Azaz a védelmi technológia adott, mivel ugyanazon az elveken és szabványokon alapszik, mégis sokakban nagyon komoly kételyeket ébreszt a felhő alkalmazásának gondolata. A korábbi évtizedekben az volt a jellemző, hogy a vállalati operációs rendszereket támadták meg hackerek, amelyek védelmére nagyon jó tűzfalakat, vírusirtókat, alkalmaztak. De egy ilyen interneten elérhető felhőszolgáltatás esetében teljesen más jellegű védelemre van szükség. Talán a legnagyobb problémát nem is a védelmi módszerek hatékonysága jelenti, hanem sokkal inkább az, hogy az adott szolgáltató valóban bizalmasan kezeli-e az adatainkat. A legtöbb ilyen szolgáltatást nyújtó vállalat amerikai vagy ottani szerverparkokat alkalmaznak és ezáltal egy olyan csapdába eshetünk, hogy az adatainkat egy külföldi szabályzás miatt ezek a vállalatok adatokat szolgáltathatnak a kormány vagy esetleg a titkosszolgálatok felé. Ezt a problémát és annak jelentőségét felismerve hozott a norvég kormány olyan döntést néhány hónappal ezelőtt, hogy az országban letiltja a Google Apps nevű alkalmazást az adatbiztonsági problémákra hivatkozva. [5]

„A cloud computing egy speciális adatfeldolgozásnak tekinthető, ahol az adatfeldolgozó cég és szolgáltatás számos paramétere ismeretlen. Az adatvédelmi törvény adatfeldolgozásra vonatkozó szabályai azonban szinte alkalmazhatatlanok. Eleve sok esetben nincs írásos megbízás az adatkezelő és az adatfeldolgozó között. Sokszor nem hogy a felhasználókat nem lehet

tájékoztatni arról, hogy mely szolgáltatónál található meg a szerverek, és milyen adatbiztonsági intézkedésekkel védettek az adataink, de még maga az adatkezelő sem mindig ismeri ezeket az információkat. A magyar adatvédelmi törvény alapján külföldre csak akkor lehet adatokat átadni, ha ahhoz az érintett kifejezetten hozzájárul, vagy törvény adja a felhatalmazást. Egyszerűen lehetetlen egy ilyen szolgáltatás esetében a kifejezett hozzájárulást, pl. egy check-box segítségével megkérni.” [6]

3. PRIVÁT FELHŐK ALKALMAZÁSA

A privát felhők számos olyan előnyt kínálnak a vállalkozások számára, amelyekre a hagyományos környezetek nem képesek. Azáltal, hogy az alkalmazások és az üzleti egységek között megosztják az infrastruktúra erőforrásait, megnövelhetik a szerverek kihasználtságát. A későbbi beruházási költségek csökkentése mellett, a folyamatos működtetésre fordított összegeket is csökkenti, hiszen kevesebb energia, hűtés valamint kisebb adatközpontra van szükség. A privát felhők rugalmassága kétségtelen. Amikor az kiszolgálási igény hirtelen megnő, könnyebb újabb számítási vagy tárolási erőforrásokat indítani, majd a szokásos üzletmenetre visszatérve visszatérhetünk az eredeti kiszolgálási szintekhez. Szakemberek szerint a legnagyobb előny mégis az, hogy a privát felhők segítségével az informatikai részleg legfontosabb feladata már nem az infrastruktúra felügyelete és működtetése lesz; végre hasznosabb feladatokra koncentrálhatnak: hogyan tudnak olyan üzleti szolgáltatásokat kínálni, amelyekkel a vállalat még versenyképesebb lehet.

A privát felhő működésének alapja a virtualizáció és a dinamikus felügyeleti eszközök. Előbbi lehetővé teszi, hogy a vállalat maximálisan kihasználhassa a rendelkezésére álló hardveres erőforrásokat, szükség esetén percek alatt beüzemeljen egy új szerveret, és azáltal biztosítsa a folyamatos működést, hogy karbantartás idején egyik fizikai kiszolgálóról a másikra vigye át a virtuális gépeket. A dinamikus felügyeleti eszközök a fenti feladatok elvégzését egyszerűsítik, és lehetőség szerint automatizálják, ezzel minimalizálva az infrastruktúra napi működésének nehézségeit és költségeit. A privát felhő előnyeit igazából akkor tudja kihasználni a vállalat, ha az alkalmazott technológia módot ad a következőkre: [7]

- **Az erőforrások automatikus optimalizálása.** A szolgáltatási szinteket és a hatékonyságot akkor lehet magasán tartani, ha a felhőt felügyelő megoldás folyamatosan monitorozza a rendszer teljesítményét, és szükség esetén újraellokalja az infrastruktúrát alkotó erőforrásokat, illetve automatikusan átmozgatja

a virtuális gépeket – rendszergazdai beavatkozás nélkül.

- **Önkiszolgálás.** A felhők egyik legnagyobb erénye, hogy az üzleti egységek kezébe adja, mennyi erőforrással gazdálkodjon. Egy jól kialakított felügyeleti rendszerben vannak olyan webes eszközök, amelyekkel a belső informatikai szolgáltatások felhasználói maguk állíthatnak üzembe és deaktiválhatnak szervereket.
- **Felhasználás alapú mérés.** A felhőmodell lényege végső soron az, hogy mindig csak annyi erőforrásért „fizetünk”, amennyit éppen használunk. A rendszernek folyamatosan mérnie kell egy adott szervezeti egység adott időszakban mennyi virtuális szervert, tárolási és egyéb erőforrást használt fel, és annak megfelelően kiszámolni és „kiszámlázni” a költségeket.

Ha egy vállalat már alkalmaz virtualizációs technikákat, akkor viszonylag egyszerűen tovább tud haladni a privát felhők irányába. Először meg kell fogalmaznia, hogy pontosan milyen képességeket vár el az adatközponttól. Második lépésként érdemes kialakítani egy kisméretű pilot-felhőt (mintafelhőt), amire egy meglévő alkalmazásokat telepítjük. A próbaalkalmazások kiválasztásánál több szempontot is figyelembe kell venni: 1.) a szoftver legyen elég nagy és összetett ahhoz, hogy a rendszergazdák valós képet kaphassanak azokról a problémákról, amelyek a vállalati szintű bevezetéskor adódhatnak. 2.) olyan alkalmazással érdemes először próbálkozni, amely jól kihasználhatja a felhő méretezhetőségét (skalázhatóságát), például nagyon gyorsan változik a kiszolgálási igény, vagy periodikus a használata. Amennyiben a kezdeti próbák sikeresnek bizonyulnak, hosszabb távú terveket is érdemes figyelembe venni. Egyes szakértők óvnak a túl gyors bevezetéstől és inkább azt tanácsolják, hogy fokozatosan migráljuk az erőforrásokat és a kapacitást az új infrastruktúrához. Ha pedig „belül” már minden kiválóan működik, elgondolkodhatunk azon, hogyan lehet a privát felhő egyes elemeit a vállalaton „kívülre” helyezni, azaz valamilyen nyilvános felhőre (pl. Microsoft Azure), ezzel valamilyen vegyes felhőkörnyezetet kialakítani. [7] Az ilyen vegyes megoldások kialakítása a korábban bemutatott adatkezelési aggályokra is megoldást nyújthat.

3. SZOLGÁLTATÁSI MODELLEK

Alapvetően három szolgáltatási modellt különböztetünk meg [3]:

- Szolgáltatásként kínált szoftver (Software-as-a-Service - SaaS): Eltérve a hagyományos megközelítéstől, a megrendelő nem egy telepíthető, birtokolható szoftvert kap, hanem az alkalmazásokat interneten keresztül használja. Ez által a cégek

rugalmasan skálázhatják az alkalmazásaikat, nem kell nagy hardver-beruházásokat finanszírozniuk, üzemeltetésben, biztonságban jártas szakembereket alkalmazniuk.

- Szolgáltatásként kínált platform (Platform-as-a-Service - PaaS): Olyan platformot biztosít a fejlesztők számára, ahol gyorsan és rugalmasan együtt tudnak fejleszteni, valamint tárolási és számítási kapacitást biztosít a web-alkalmazásokhoz. Ez a modell az előző egyfajta kiegészítéseként is tekinthető. Sikeres SaaS szolgáltatásra példaként a salesforce.com-ot említhetjük.

- Szolgáltatásként kínált infrastruktúra (Infrastructure-as-a-Service - IaaS): Nemcsak alkalmazásokat és platformokat vehetünk igénybe szolgáltatásként, hanem erőforrásokat is. Például egy szolgáltatás operációs rendszert, tárterületet és távoli hozzáférést biztosíthat.

A nyilvános felhők a hatékonyság érdekében általában csak nagyon korlátozott szolgáltatást (SaaS, PaaS), míg a privát felhők akár nagyszámú alkalmazást is és teljes platformokat is kínálnak (IaaS). A felhő technológia megjelenésével új eszközök és módszerek jelentek meg az informatikában, bizonyos szakterületeken. A [4] alapján áttekintjük, milyen új szerepkörök jelennek meg a felhő technika alkalmazásakor:

- Felhő architekt;
- Felhő adminisztrátor;
- Felhő szolgáltatás menedzser;
- Felhő adat architekt;
- Felhő tárhely adminisztrátor;
- Felhő alkalmazás architekt;
- Felhő operátor;
- Felhő felhasználó;
- Felhő fejlesztő.

Az ilyen tevékenységek végrehajtásánál a következő feladatok jelentkeznek:

- egy-egy személynek kell évente egy vagy több alkalommal a helyszínre kiszállni és a tevékenységet ellátni,
- a tevékenység ellátáshoz alkatrészek, szerszámok, gépek egyéb eszközök helyszínre való ki- és visszaszállítására is szükséges lehet,
- az is belátható, hogy a tevékenységet ellátó személyek vagy szakértőknek a kiszolgált terület objektumaihoz közel lakjanak, mert így képesek kis időráfordítással, költséghatékonyan tevékenykedni,
- a szükséges anyagok, alkatrészek, eszközök, gépek a rendszer különböző pontjaiban telepített raktárakban találhatóak, ezekből történik ki- ill. visszaszállításuk,
- illetve előfordulhatnak a helyszínen nem javítható szerkezeti elemek, amelyeket kiszállítás után a karbantartó üzemekben újítják fel.

Mivel az ilyen rendszerek kiterjedhetnek akár egy városra, országra vagy akár földrészekre, az optimális működtetésük jelentős logisztikai probléma, amelyet hálózatszerűen, virtuális logisztikai központ irányításával működő rendszerek látnak el.

Mivel napjainkban igen sok új optimalizációs módszert dolgoznak ki, főleg biológiai elveken alapulva, ilyenek például a részecskeraj algoritmus [1], vagy a nemrégiben kidolgozott harmónia kereső algoritmus [2], vagy a tabu keresés, amelynek általános algoritmusai alkalmasak a szomszédsági információkkal nem rendelkező rendszerek optimalizálására is [3].

Jelen cikkben nem foglalkozunk a rendszer matematikai leírásával, hiszen az már publikálásra került [4][5].

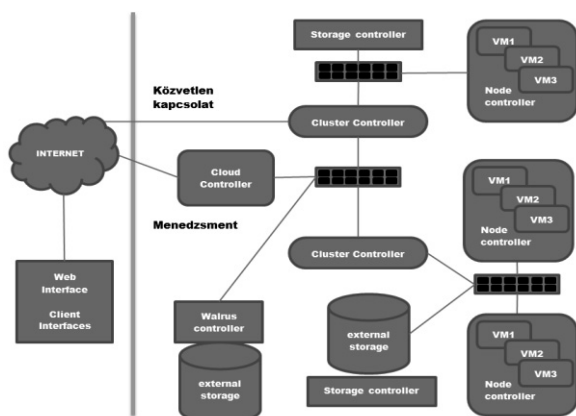
2. UBUNTU ENTERPRISE CLOUD SERVER

Az Ubuntu Linux célul tűzte ki, hogy kiválasztja és folyamatosan karbantartja azokat a nyílt forráskódú eszközöket, amelyek segítségével privát felhők alakíthatók ki. Az Amazon EC2/3 alkalmazási interfészt megvalósító Eucalyptus (<http://open.eucalyptus.com/>) projektre támaszkodnak. Az Ubuntu 9.04 szerverbe építették bele az Eucalyptus egy továbbfejlesztett változatát a kernel alapú virtuális gépi támogatással kiegészítve, és ezt a változatot UEC-nek (Ubuntu Enterprise Cloud) nevezték. Amikor a 10.10-es változat került a piacra, az UEC már a nyílt forráskódú privát felhők legsikeresebb változatává vált, amely már a hibrid felhők irányába is fontos lépéseket tett.

3.1 Az UEC komponenseinek bemutatása

Node Controller (NC): Vezérli a virtuális gépek életciklusát. Ez a szoftverkomponens alap esetben, a KVM hypervisort, mint virtuális gépet alkalmazza. Az NC egyik oldalról a futtató operációs rendszerrel és másról pedig a Cluster Controllerrel (CC) van kapcsolatban, továbbá a rugalmas bővíthetőség elősegítése miatt lekérdezi az CPU-k és a magok számát, a fizikai memóriát, a rendelkezésre álló lemezterületet, az aktuálisan futó virtuális géppéldányok számát és továbbítja a CC számára. Tehát összefoglalva, fő funkciói: az erőforrások karbantartása és a virtuális gép (VM) példányok életciklusának kezelése. Cluster Controller (CC): A CC menedzseli a Node Controllereket, telepíti a VM példányokat, és a példányok hálózatkezeléséért is felelős. Az Eucalyptus bizonyos típusú hálózati módokat alkalmaz például: IP kontroll, biztonsági csoportok, metadata szolgáltatás, VM izoláció, amelyeket a privát felhő topológiájának tervezésénél figyelembe kell venni. Összefoglalva a CC felelős azért, hogy egy új VM példány melyik NC segítségével induljon el, vezérli a hozzá tartozó virtuális

hálózatot, továbbá összegyűjti az NC-k futási adatait és továbbítja a Cloud Controller (CLC) felé. Walrus (W3): W3 perzisztens egyszerű tároló szolgáltatást kínál REST és SOAP API-n keresztül. Tárolja a VM példányok fizikai állományait, ezek mentéseit (snap-shot), valamint egyszerű fájlkiszolgálóként használható a felhőben. Storage Controller (SC): Az SC támogatja a perzisztens blokk szintű tárolók alkalmazását. Az Amazon Elastic Block Store (EBS) megvalósításának tekinthető. Ez a tároló független az őt használó VM példányoktól, azok használják, amelyeknek szüksége van adatbázisra, fájlrendszerre vagy blokk szintű tárolóra. A tárolók mérete dinamikusan növelhető, ami kellő rugalmasságot ad a használata során. Cloud Controller: A privát felhő infrastruktúra front-end komponense a Cloud Controller (CLC), amely egy webszolgáltatás interfész. Egyik oldalról a kliensek az API-n keresztül elérhetik szolgáltatásait, másik oldalról a fenti komponensekkel van kapcsolatban. Web-es interfészt is tartalmaz, és monitorozza a felhő erőforrásait VM példány szinten, továbbá eldönti, hogy egy új példány melyik CC segítségével induljon el



1. ábra Egyszerű privát felhő topológia

3. MINTA FELHŐ KIALAKÍTÁSA

A rendelkezésre álló Dell 4 magos szervereken kialakítottunk egy mintafelhőt, az Ubuntu Enterprise Cloud Server komponenseivel. A 4 kiszolgáló elkülönített alhálózatban működik. A megvalósított konfiguráció és az egyes szervereken alkalmazott szoftver komponensek az 1. ábrán láthatóak. Az első szerver, két hálózati adapterrel is rendelkezik, a felhővezérlő (CC), a W3 és CC-n kívül az SC is ezen helyezkedik el. A cél az, hogy 1 kiszolgálón helyezzék el a kiegészítő komponenseket, a többin kizárólag az NC-k legyenek. A CC segítségével a megjelenő virtuális gépek egy előre definiált tartományból kapják IP címeiket. Az ElasticFox Firefox kiegészítő segítségével a felhő adminisztrátor a virtuális gépeket tarthatja karban: vezérelheti darabszámukat, indíthat-leállíthat példányokat, biztonsági csoportokat hozhat létre

(security groups) és lemezterületeket (volumes) dinamikusan képes módosítani és pillanatfelvételeket készíthet (snap-shot).

4. ÖSSZEFOGLALÁS

A projekt keretében a privát felhők létrehozásának és menedzselésének egyes feladatait ismertettük. Bemutatásra kerültek azok a kérdések, amelyek a kis és közepes vállalkozások esetében legtöbbször felmerülnek. A felhő technika alkalmazása önmagában is olyan paradigma, amely érdekes kutatási feladatokat generálhat. Széleskörű elterjedését jelenleg a nem tisztázott adatvédelmi, adatbiztonsági problémák gátolják, viszont a privát felhők alkalmazásánál ezek a problémák nem jelentkeznek.

5. KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

6. IRODALOM

- [1.] GAUTAM SHROFF: Enterprise Cloud Computing Technology, Architecture, Applications, Cambridge University Press, ISBN 978-0-521-76095-9, 2010
- [2.] JOHN W. RITTINGHOUSE, JAMES F. RANSOME: Cloud Computing Implementation, Management, and Security, ISBN: 978-1-4398-0680-7, 2010
- [3.] MICHAEL HUGOS, DEREK HULITZKY: Business in the Cloud, ISBN 978-0-4706-1623-9, 2011
- [4.] Learn about cloud computing. <http://open.eucalyptus.com/learn/cloud-it-roles>, 2012
- [5.] Info-Security, <http://www.infosecurity-magazine.com/view/23463/use-of-google-docs-is-illegal-in-norway/>, 2012
- [6.] Adatvédelmi szakértő weboldal <http://www.adatvedelmiszakerto.hu/2011/03/cloud-computing-es-adatvedelem-vagyis-annak-hiany/>, 2012
- [7.] Felhő saját használatra, http://www.itbusiness.hu/Fooldal/hetilap/tech/Felhő_saját_használatra.html, 2012
- [8.] 13 tévhit a felhőről, <https://technetklub.hu/content/cloudmyths.aspx>, 2012

A SZÁMÍTÓGÉPES FOLYAMATTERVEZÉS ÉS A MŰHELYSZINTŰ GYÁRTÁSIRÁNYÍTÁS EGY INTEGRÁCIÓS PROBLÉMÁJA

AN ISSUE OF INTEGRATING COMPUTER AIDED PROCESS PLANNING AND MANUFACTURING EXECUTION SYSTEMS

Erdélyi Ferenc* Tóth Tibor**

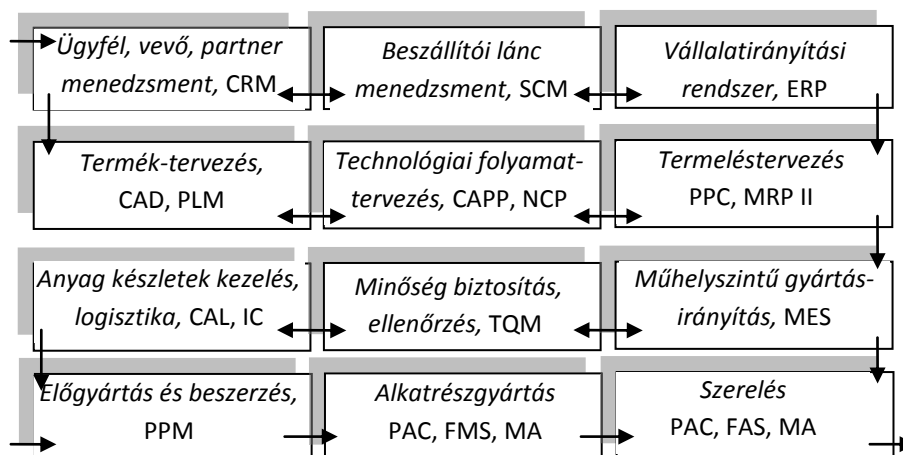
ABSTRACT

Computer integrated support of manufacturing control and of technology process planning for the work pieces to be manufactured has indicated challenge for Production Information Engineering even now. The two applications have been localized on two hierarchy levels, which require „vertical integration”. In the field of discrete manufacturing systems, hierarchical organizing the engineering and management functions is the only one „framework” utilizable in wide range at present. For integration not only solving the syntactical interface and semantic coherence problems has to be required but it is also needed to solve the problems

originating from the difference of the models used at the two hierarchy levels. The paper shows the role of the production rate concept in the IT-based integration and in realizing the flexible and robust manufacturing control.

1. GYÁRTÁSI RENDSZEREK TERVEZÉSÉNEK ÉS IRÁNYÍTÁSÁNAK FUNKCIÓI

Napjainkban a diszkrét rendszerek területén nagy vonásokban ugyanazt a jellegzetes mátrix alakú hierarchikus funkcionális struktúrát alkalmazzák [1] [2], (1. ábra).



1. ábra: A műszaki és a menedzsment funkciók hierarchikus struktúrája

Az egy időben divatos „heterarchikus”, és „holonikus” struktúrák [3] egyelőre nem léptek ki a teoretikus modellek köréből. A hierarchikus struktúrát az jellemezi, hogy: (1) a vállalatirányítási üzleti folyamatok, (2) a műszaki tervezési folyamatok, (3) az operatív termelésirányítási folyamatok, (4) a technológiai (fizikai) végrehajtási folyamatok több szintű vertikális struktúrát alkotnak. A hangsúly a

hierarchikus jelzőn van, ami azt jelenti, hogy a funkcionális komponensek nem „egyenrangúak”. A magasabb hierarchiai szintek információi az alacsonyabb szintek felé „kötelező” tervek, célok, „kemény” követelmények és korlátozások alakjában jelennek meg. Fordított irányban az információ „csak” státusz és teljesítmény-mutatók (KPI, Key Performance Indices) és jelentések, riportok formáját ölti. A struktúra

*CSc. c. egyetemi tanár, Miskolci Egyetem Alkalmazott Informatikai Tanszék

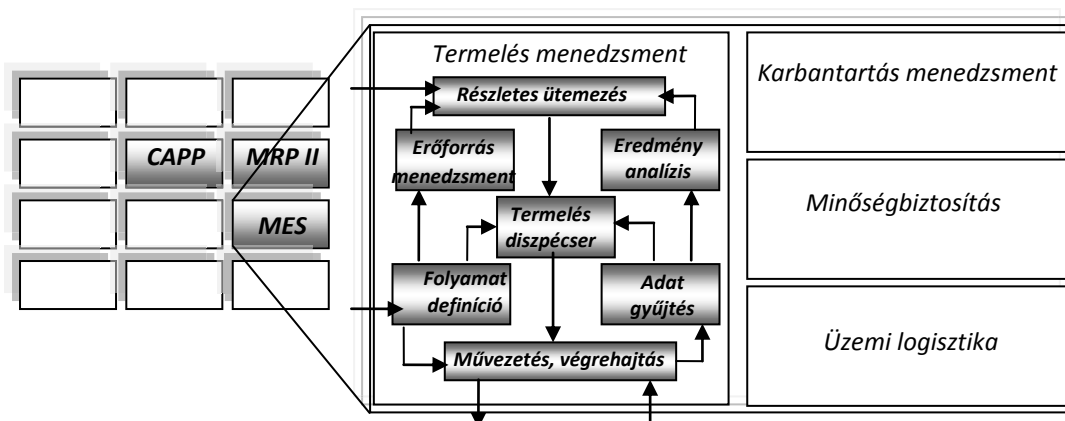
**DSc. egyetemi tanár, Miskolci Egyetem Alkalmazott Informatikai Tanszék

szintjein természetesen a funkcionális munkamegosztás érdekében specifikus funkcionális komponensek különülnek el. Ezek „horizontális” információcseréje egyenrangú [4].

2. A MODERN GYÁRTÁSIRÁNYÍTÁS (MES) FUNKCIÓI, AZ ISA-95 MODELL

A műhelyszintű gyártásirányítást napjainkban a MES (*Manufacturing Execution System*) alkalmazások támogatják. A MES-nek - a termelésinformatika alapvető komponensének - az MRP-hez hasonlóan, a mai napig nincs mindenki által elfogadott referencia modellje. Az MRP tekintetében a piac-vezető ERP gyártó - az SAP - modellje tekinthető referenciának. Ami a MES-t illeti, létezik két sikeres kísérlet a MES

funkcionális referencia modelljének definiálására. Az egyik az Egyesült Államok egyik szoftvergyártó vállalatcsoportja, a MESA modellje, a másik javaslat az USA egyik szabványosítási intézetének modellje. Az utóbbi az ISA-95 szabvány keretében egy 2 szintű hierarchikus gyártásirányítási modellt definiál (2. ábra) [6]. A „gyártás” komponens támogat minden aktivitást, amely közvetlenül kapcsolódik a műhely szintű technológiai folyamatokhoz beleértve az üzemi logisztikát, a gyártás-előkészítést az előgyártást, az alkatrészgyártást és a szerelést. Az ISA-95 gyártási komponens 7 gyártásirányítási alrendszere: (1) Részletes, finom ütemező; (2) Erőforrás menedzsment; (3) Folyamat definíció; (4) Termelés diszpécser; (5) Művezetés, végrehajtás; (6) Termelési adatgyűjtés; (7) Eredmény-követés és analízis.



2.ábra. A gyártásirányítás funkcionális modellje (ISA-95)

A „Karbantartás” modul a gépek és más erőforrások karbantartásának tervezése, és a működési állapot felügyelete funkciókat támogatja. A „Minőség” modul feladata a minőségbiztosítás szabályainak kezelése és a statisztikus minőségellenőrzés (SPC) műveleteinek végrehajtása, dokumentálása. Az „Üzemi logisztika” modul az anyagmozgatási és raktározási tevékenységek tervezését és végrehajtását támogatja.

A MES rendszereket a CAD/CAM, az ERP, valamint az MRP rendszerekkel szerzett tapasztalatok birtokában hozták létre. Ennek megfelelően a komponensek úgy vannak tervezve, hogy készen álljanak az integrációs problémák nemcsak kommunikációs (adatcsere szintű), hanem a szemantikus (ontológiai) és pragmatikus (modell) szemléletű megoldására. Ez csökkenti a szoftverek implementációjánál szükséges „testre-szabási” problémák megoldásához szükséges munkát. A funkciók erős modellfüggőségét azonban ez sem képes kiküszöbölni. Európában jelenleg is folyik olyan konzorcium-bázisú kutatás - több ország kutatóinak részvételével -, amely a műhelyszintű gyártásirányítás és logisztika ontológiai alapú modellezését és a MES szoftverek fejlesztésének támogatását tűzte ki céljaul [7].

3. A CAPP ÉS A MES FUNKCIÓK INTEGRÁCIÓJA

Az integrációs technika fejlődése első jelentős sikereit a CAD-CAPP tervezési funkciók integrálásával aratta (IGES, DXF, STEP). A következő fontos feladat az ERP-MES integráció lett, amely csak az ISA-95 szabvány elterjedése után ért el jelentősebb sikereket (B2MML). A CAPP-MES integráció ma még többnyire komoly kihívást jelent a MES rendszerek implementációjánál. A műhelyszintű gyártásirányítás és a technológiai tervezés integrációja azonban nagyon hatékonyan bizonyulhat, mert lehetővé teszi a technológiai tervezés rugalmasságának, valamint robusztusságának növelését, és a műveletek optimalizálásának új szemléletét. Ennek a illusztrálására egy CAPP-MES integrációs probléma új szemléletű megoldását vázoljuk.

Az integráció megvalósítása nemcsak adat-típustól függő (XML sémák) és szemantikai koherenciát megkövetelő probléma. Megvalósítása visszahat a funkcionális komponensek algoritmusaira, szolgáltatásaira és új követelményeket támaszt például a műveletek technológiai adatainak optimalizálása iránt. A technológiai műveletek paramétereinek

meghatározása többszintű és többcélú optimalizálási feladattá válik, amelyet csak „robosztus” szemlélettel lehet kielégítően megoldani. Korábban a „műveleti költség” fogalom szemléletével a technológiai tervezés egyetlen művelet tervezésére, az elméleti megközelítést tekintve pedig annak modelljére fókuszált. A gyártásirányítás (MES) szintjén azonban a technológiai adatok egy újabb modell (a gyártórendszer finom ütemezését tervező modell) paramétereivé válnak. A modell esemény-orientált termelési egyenleteket és aggregált teljesítmény-mutatókat használ. A termelés stabilitása szempontjából fontos szerepet kapnak az „idő- és sokaság-alapú” átlagos mutatók. Ilyenek például a gyártásban lekötött átlagos készlet szint (\bar{N}), ami fontos üzemgazdasági és logisztikai állapotjelző; a műveletek termelési intenzitásai ($q_{i,j}$); a gépek átlagos kihasználtsága (\bar{u}_j); és a gyártási sorozatok, („munkák”, job-ok) átlagos átfutási ideje (\bar{T}_i). Ezek az állapotjelzők ráadásul nem függetlenek egymástól. Kapcsolatuk fontos összefüggés, amely „termelési háromszög” egyenletként ismert [8]. A részletes indoklást ehelyütt mellőzve írhatjuk:

$$\bar{N} = \bar{q}_{i,j} \cdot \bar{u}_j \cdot \bar{T}_i, \quad \text{ahol } \bar{q}_{i,j} = \frac{n \cdot m}{\sum_{i=1}^n \sum_{j=1}^m \tau_{i,j}}$$

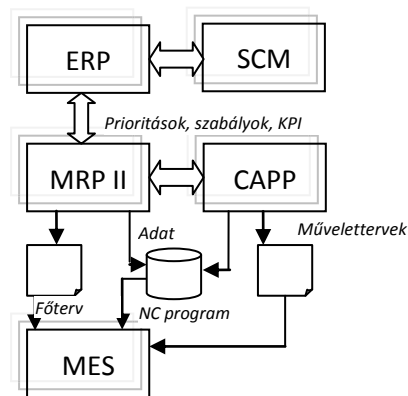
a gyártórendszer karakterisztikus intenzitása, és $\tau_{i,j}$: az i -edik job j -edik gépen végzett műveletének műveleti ideje, n az elindított munkák (job-ok) és m a gépek (munkahelyek) száma.

A műveletek intenzitása a határidők betarthatóságát (szállítókészség!), a minőségbiztosítás eredményességét, a selejtarányt és a szerszámfelhasználást is erősen befolyásolja. Ez szükségessé teszi a megengedett és több célú értelemben hatékony (*Pareto*-optimális) technológiai adatok alternatív készletének ismeretét [9]. A technológiai adatok ilyen halmazát joggal nevezhetjük robusztusnak, mert felhasználásuk a magasabb szintű főtervek (*Master Plan*) készítésénél és alacsonyabb szinten a MES részletes ütemtervének készítésénél is egyaránt hatékony. Ennek a szemléletnek egy lehetséges megvalósítása ERP→CAPP, illetve CAPP→MES integrációval a következő koncepció szerint történhet:

Az ERP-MRP termelés-tervezési szint részben a Master Plan, részben az Adat Centrum segítségével a rendelésekre jellemző szabályokat és adatokat küld a technológiai tervezésnek. A szabályok a művelettervezési célokra vonatkoznak. Az adatok a korlátokat befolyásolják. A technológiai tervezés meghatározza azokat a technológiai adattartományokat, amelyek a szabályokból származó gyártási stratégiákhoz tartoznak. Ezek segítségével *alternatív és paraméteres művelettervek*, illetve NC programok készülnek.

A technológiai tervezés *alternatív műveletterveit* a MES részletes ütemezés funkciója aktualizálja a termelési főterv alapján készült job-okra. Ez a koncepció feltételezi, hogy az ütemező rendszer képes AI módszerekkel kezelni a megengedett job-indítási, művelet-sorrendi, gép és „útvonal” alternatívák mellett a műveleti technológiai adat alternatívák változatait is. Egy ilyen ütemező rendszert a közelmúltban fejlesztett ki és publikált az ME Alkalmazott Informatikai Tanszékének kutatócsoportja [10].

A MES diszpécser funkciója a gyártási folyamatok valós idejű felügyelete segítségével folyamatosan megfigyeli a job-ok és az erőforrások státusát (állapotát), illetve meghatározza a termelési háromszög mutatók (KPI) aktuális értékeit. Szükség esetén valós idejű döntéseket hoz, a job-ok és erőforrások státusának megváltoztatására, a folyamat újra ütemezésére. Az újra ütemezés figyelembe veszi a műveletek megengedett (robusztus) alternatíváit (3. ábra).



3. ábra. Integrált CAPP-MRP-MES rendszer

A koncepció az alkalmazások koherens illesztésével valósítható meg. A számítógépes termelés-tervezési és technológiai tervezési alkalmazások *csatlakozási(interface)* felülete teljes értékű támogatást tud nyújtani a fent vázolt funkcionális tevékenységek megvalósításához. Az alkalmazások szolgáltatás-orientált együttműködését a hálózati alkalmazási rendszerek, a biztonságos Adat Centrum és a szerverek és alkalmazások virtualizációs technikával támogatott integrációja biztosítja [11].

A diszkrét alkatrészgyártás területén az ilyen robusztus technológiai adat alternatívák a következő célokhoz, illetve ezek többcélú (*multi-objective*) kombinációjához tartozhatnak:

- műveleti költséget minimalizáló technológiai adatok,
- termelési intenzitást maximalizáló technológiai adatok,
- szerszám- és segédanyag felhasználást korlátozó technológiai adatok,
- energia-felhasználást korlátozó technológiai adatok,
- minőséget garantáló, selejtet minimalizáló technológiai adatok, stb.

A koncepció fontos eleme, hogy a műveletek (pl. a forgácsolási műveletek) technológiai adatainál előnyben részesítjük az anyagleválasztás intenzitását (material removal rate, MRR) adatot ($Q, \text{cm}^3/\text{min}$), ami áttekinthetőbbé és megbízhatóbbá teszi a *Master Plan*, a job-tervező, a technológiatervező valamint az ütemező (és gyártásirányító diszpécser) munkáját.

Az ERP rendszerek a termelési költségeket nem a műveleti költségek összegzésével, hanem az erőforrások felhasználásának költségei alapján határozzák meg. A technológiai tervezés, a termelési főterv készítés és a gyártási ütemterv készítés modelljei tehát más célokra fókuszálnak. Ennek is a következménye, hogy a termelési ütemtervek elkészítésénél az ütemezőt elsősorban a lehető legnagyobb megengedett termelési intenzitás ($q_{i,j}^* = q_{i,j} \rightarrow \max$) érdekli. Ezt az intenzitást kell alkalmazni a szűk keresztmetszetet jelentő munkahelyeken (természetesen a „kemény” korlátok figyelembevételével). Minden olyan $O_{i,j}$ művelet esetében azonban, amelynek $\tau_{i,j}$ műveleti ideje a részletes ütemterv alapján növelhető anélkül, hogy a T_i job átfutási idők változnának vagy a „service level” változna, a művelet Q anyagleválasztási intenzitását csökkenteni lehet. Ez a technológiai adat korrekció a szerszám-fogyást (szerszám költségeket) és a selejt kockázatot jelentősen csökkentheti.

A technológiai tervezés, a termelésstervezés, és a gyártásirányítás a modern gyártásban ezer szállal kötődik egymáshoz. A viszonylag önálló funkcionális tervező szoftverek azonban ezt csak integrált rendszerek estén tudják hatékonyan kihasználni.

4. KÖVETKEZTETÉSEK

Az elmúlt 20 évben a termelési rendszerek és folyamatok tervezése és irányítása területén - az IT hatására - jelentős paradigma-váltás és mélyreható változások következtek be. Az IT lehetővé teszi, hogy a tervezésben és az irányításban az emberi elhatározás és invenció támogatására funkcionális teremtés-informatikai alkalmazásokat használjanak tömegesen már a kis és a közepes vállalatok is. Az alkalmazások differenciált, elkülönült modelleket használnak, amelyek termék és technológia specifikusak. Az alkalmazások számának növekedésével egyre nagyobb az igény azok „interoperabilis” együttműködésére, integrációjára. Az interoperabilitás legelvontabb szintjén a pragmatikus szempontokat a modellek koherenciája biztosíthatja. A modellekben használt célokat és korlátozásokat is koordinálni kell. Itt érvényesülnie kell a hierarchiából következő aszimmetria elvnek, a kompatibilitásnak és a koherenciának egyaránt. Ezeket az elveket az alternatív terv-változatok, a robusztus tervezés, a paraméterek esetfüggő és korlátokhoz kötött tervezése biztosíthatja. A forgácsolási technológiai tervezés klasszikus

optimális technológiai adat meghatározási feladatának fejlődése jól példázza ezt a folyamatot.

KÖSZÖNETNYILVÁNÍTÁS

A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.

IRODALOMJEGYZÉK

- [1] VERNADAT, B. F., 1996. *Enterprise Modelling and Integration*. Chapman & Hall, London. ISBN: 0 412 60550 3.
- [2] TÓTH, T., ERDÉLYI, F., 2006. *Integrated Application Systems for Manufacturing Industry*, Proceedings of the 12th International Conference on Machine Design and Production (UMTIK 2006), 5-8 September 2006, Kusadasi, Turkey, pp. 105-122.
- [3] MONOSTORI, L.; KÁDÁR, B., 1999. *Holonic control of manufacturing systems*. 1st IFAC Workshop on Multi-Agent-Systems in Production, December 2-4, 1999, Vienna, Austria, pp. 109-114.
- [4] ASKIN R. G., STANDRIDGE, C. R., 1993. *Modelling and Analysis of Manufacturing Systems*. J. Wiley Inc. New York. 1993.
- [5] TÓTH TIBOR, 1998. *Tervezési elvek, modellek és módszerek a számítógéppel integrált gyártásban*. Miskolci Egyetemi Kiadó, 1998. ISBN 963 661 339 7.
- [6] SCHOLTEN, B., 2007. *The Road to Integration. A guide to Applying the ISA-95 Standard in Manufacturing*. ISA Edition. ISBN 0 9792343 8-7.
- [7] Virtual Factory Framework. EU Seventh framework programme. <http://www.ims.org/2011/11/vff-virtual-factory-framework/>
- [8] ERDÉLYI FERENC, 2011. *Modeling the Performance of Shop Floor Logistics by Means of Key Performance Indices*. 60 éves a Közlekedésmérnöki és Járműmérnöki kar, Budapesti Műszaki Egyetem, 2011 Konferencia CD kiadvány. ISBN 978 963 313 040-7.
- [9] HOPP, W. J., SPEARMAN, M. I., 2003. *Factory Physics*. Second edition, Singapore. Mc Graw-Hill.
- [10] KULCSÁR, Gy., ERDÉLYI, F., 2007. *New Approach to Solve Multi-Objective Scheduling and Rescheduling Tasks*. International Journal of Computational Intelligence Research IJCIR, Vol. 3, No. 4, pp. 343-351
- [11] PFADENHAUER, K.; KITTL, B.; DUSTDAR, S., and LEVY, D., 2006. *Shop Floor Information Management and SOA*. BPM Workshops, Vol. 41.03 Springer, 2006, pp. 237-248.

CONTENTS

1. TÓTH T.: New methods for computer aided planning of discrete technology processes	3	16. SZILÁGYI SZ.: An application of order congruences in parallel multi-machines scheduling problems	67
2. TURÁN, L. OVSENIK, M. TURÁN, VÁSÁRHELYI J.: Atmosphere quality monitoring system design and measurement	11	17. BARABÁS P.: Function extraction from hungarian text.....	71
3. TROHÁK A.: Design of a test system for load distribution strategies.....	15	18. KRIZSÁN Z.: Common Fuzzy Rule Interpolation Framework library for developers.....	75
4. PINTÉR J., TROHÁK A.: Development of a voice-commanded interface for a monorail	19	19. KOVÁCS L.: Classification method based on concept lattice architecture.....	79
5. RÁDI P., TROHÁK A.: Design, simulation and examination of IWLAN industrial communication systems.....	23	20. BEDNARIK L., DR. KOVÁCS L.: Classification tasks in the system of question generation model	83
6. MÉHES L., TROHÁK A.: Uniquely designed telemetry system development to a Bosch elektromobil vehicle on ZigBee wireless network	27	21. KOVÁCS SZ.: Extending the „Double-Linear” Fuzzy Interpolation Method with the interpolation of fuzziness.....	87
7. BIRÓ Z., KOLOZSI-TÓTH M., TROHÁK A.: Remote diagnostics of vehicles via gsm network.....	31	22. KOVÁCS SZ.: Fuzzy Rule Interpolation in embedded control applications.....	91
8. VARGA A. K.: Location determination algorithms in self-organizing wireless sensor networks	35	23. MILEFF P., DUDRA J.: Efficient techniques and methods for software rendering	95
9. VARGA A. K.: Development of the Program network interface at the University of Miskolc	39	24. TÓTH ZS.: Formal grammar modul in gi meta framework.....	99
10. TROHÁK A., HALUSKA ZS.: The measurement of RF spectrums of wireless industrial communication systems.....	43	25. VINCZE D.: A novel access control method in the kernel of the linux operating system	103
11. DR. CZAP L., PINTÉR J.: Voice-controlled logitics systems	47	26. WAGNER GY., TÓTH T.: Security problems of (local) networks.....	107
12. FERENCZI I.: Cycle time of embedded datagram used industrial ethernet frames	51	27. DR. KULCSÁR GY., DR. BIKFALVI P.: Solving production planning and control tasks using multi-objective search method	111
13. BOGNÁR G., ROZGONYI E.: Forced convection flow of a non-Newtonian fluid over a flat plate in porous medium.....	55	28. KÖREI A., TÓTH T.: Metaheuristics for solving the cell formation problem	115
14. VARGA P.: Graphical representations of certain solutions of the Yang-Baxter equation	59	29. HORNYÁK O., NEHÉZ K.: Cloud technology for small and medium sized enterprises	119
15. VARGA P.: Modified Lax representation of certain generalized Toda chains.....	63	30. ERDÉLYI F., TÓTH T.: An issue of integrating computer aided process planning and manufacturing execution systems	123

GÉP

INFORMATIVE JOURNAL

for Technics, Enterprises, Investments, Sales, Research-Development, Market of the Scientific Society of
Mechanical Engineering

Dr. Döbröczöni Ádám

President of Editorial Board

Vesza József

General Editor

Dr. Jáрмаi Károly

Dr. Péter József

Dr. Szabó Szilárd

Deputy

Dr. Barkóczy István

Bányai Zoltán

Dr. Beke János

Dr. Bercsey Tibor

Dr. Bukoveczky György

Dr. Czitán Gábor

Dr. Danyi József

Dr. Dudás Illés

Dr. Gáti József

Dr. Horváth Sándor

Dr. Illés Béla

Kármán Antal

Dr. Kulcsár Béla

Dr. Kalmár Ferenc

Dr. Orbán Ferenc

Dr. Pálincás István

Dr. Patkó Gyula

Dr. Péter László

Dr. Penninger Antal

Dr. Rittinger János

Dr. Szabó István

Dr. Szántó Jenő

Dr. Tímár Imre

Dr. Tóth László

Dr. Varga Emilné Dr. Szűcs Edit

Dear Reader,

In the present, 10th issue of the review „Gép”, research results of the Centre of Excellence of Mechatronics and Logistics are presented in the form of publications. The project has been supported by the European Union, co-financed by the European Social Fund, as well as the contribution of the units of the University of Miskolc participating in the research. The excellence centre is made up of four scientific workshops, all of them being related to mechatronics and logistics. Complex research on the fields of mechatronics and logistics and the implementation of the research results seem inevitable, since both are consequences of the trends of Hungarian economic development and the policy of the Hungarian government (dynamic expansion of the automobile industry, the creation of regional logistical centres, etc.). Intelligent systems are only marketable with adequate costs, communication and legal environment, therefore the research from the above aspects concerning mechatronics and logistics are also beneficial.

The Centre of Excellence of Mechatronics and Logistics are made up of the scientific workshops Research and Development of Elements of Mechatronical Systems, Research Of Performance Enhancement Processes And of Logistic Systems, Developing Reliability of Wired and Wireless Communication Systems and Innovative Solutions For Enhancement of Competitiveness of Organizations. During the last year’s research period numerous lecturers, researchers and students have been given the opportunity to present their research results at acknowledged Hungarian and international conferences. Among its main objectives, the centre intends to keep young lecturers, researchers in the region, to build networks with industrial companies and implement joint research with them.

The inspection and modeling of the communication systems of wired and self-organizing wireless sensor networks especially regarding to the energy optimization, error recognition and time critical operation. Theory of controlling and the automated devices. Investigation of efficient concept and grammar representation models based on ontology. Development of command interfaces using stochastic and soft computing methods. Improvement of production engineering and mathematical models and algorithms. New production scheduling algorithms implemented on supercomputers. Industrial applications of manufacturing- and logistics oriented simulations.

The present collection of articles represents an important part of the recent year’s work of the excellence centre.

Prof. Dr. Béla Illés
university professor,
leader of the Center of Excellence

Dr. Czap László
associate professor,
leader of the Research Group

Managing Editor: Vesza József. Editor’s address: 3534 Miskolc, Szervezet utca 67.
Postage-address: 3501. Pf. 55. Phone/fax: (+36-46) 379-530, (+36-30) 9-450-270 • e-mail: mail@gepujsag.hu

Published by the Scientific Society of Mechanical Engineering, 1027 Budapest, Fő u. 68.
Postage-address: 1371, Bp, Pf. 433
Phone: 202-0656, Fax: 202-0252, E-mail: a.gaby@gteportal.eu, Internet: www.gte.mtesz.hu
Responsible Publisher: Dr. Igaz Jenő Managing Director

<http://www.gepujsag.hu>
Printed by Gazdász Nyomda Kft. 3534 Miskolc, Szervezet u. 67.
Price per month: 1260 Ft.
Distribution in foreign countries by Kultúra Könyv és Hírlap Külkereskedelmi Vállalat H-1389
Budapest, Pf. 149. and Magyar Média H-1392 Budapest, Pf. 272.

INDEX: 25 343 ISSN 0016-8572

All articles are peer reviewed.