F 1879

híradástechnika

VOLUME LVII.

2002/3

Március

E 870 FF
8 000001 362819



Szoftver technológia

Hálózatok kialakítása

Távközlés politika

A Hírközlési és Informatikai Tudományos Egyesület folyóirata

Tartalom

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
Műszaki Információs Központ és Könyvtár
2001

Előszó a februári számhoz 1

SZOFTVERTECHNOLÓGIA

Kovács András és Pete Bálint
IPv4-IPv6 átállási technológiák vizsgálata 3

Andriska Zoltán, Bátori Gábor, Wu-Hen-Chang Antal és dr. Csopaki Gyula
Automatikus tesztkiválasztás formális specifikáció alapján 11

Gecse Roland, Szabó J. Zoltán, Csöndes Tibor
Időzített véges automata alapú vizsgálati módszer 19

HÁLÓZATOK KIALAKÍTÁSA

Füzesi Péter, Vidács Attila
Hálózatméretezési kérdések játékelméleti megközelítése 25

E. Grampín, J. Rubio, N. Vardalachos, A. Galis, J. Serrat
Jogosultságalapú hálózatmenedzselő rendszerek megvalósítási kérdései 31

Ladányi Zsuzsanna, Szász András
Monitorozás és számlázás DiffServ hálózatokon 37

Biró Péter, Lója Krisztina, Sütő Márton
Elosztott frekvenciakiosztási algoritmusok tervezése és vizsgálata 43

Juhász Ákos, Ulrich Ferenc, Dr. Eged Bertalan, Kubinszky Ferenc
Vezeték nélküli hálózat teljesítményvizsgálata 49

TÁVKÖZLÉSPOLITIKA

Kozma Imréné
A magyar televíziózás jövője 57

Horváth Gyula
A válság túlélése 61

Sipos László
„Minőségbiztosítás a nukleáris energetikában" c. ankét –
A biztonsági kultúra fejlesztése, hatékony minőségbiztosítással 65
Roska Tamás akadémikus a 2002. évi Bolyai-díj kitüntetettje 67

Nagy Beatrix Havaska
Baross Gábor – A vasminiszter 69

Főszerkesztő
ZOMBORY LÁSZLÓ

Szerkesztőbizottság
Elnök: LAJTHA GYÖRGY

BARTOLITS ISTVÁN
BOTTKA SÁNDOR
CSAPODI CSABA
DIBUZ SAROLTA
DROZDY GYŐZŐ
GORDOS GÉZA
GÖDÖR ÉVA
HUSZTY GÁBOR
JAMBRIK MIHÁLY
KAZI KÁROLY
MARADI ISTVÁN
MEGYESI CSABA
PAP LÁSZLÓ
SALLAI GYULA
TARNAY KATALIN
TORMÁSI GYÖRGY

Előszó a márciusi számhoz

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
Műszaki Információs Központ és Könyvtár
2001

A híradástechnika fogalmába egyre több feladat sorolható. Összenőtt a távközlés, az informatika, sőt némely területen már a technológiába épült automatika, távirányítás kérdéseit is ide sorolják. Nem szinonimája, hanem bővített formája ezért a híradástechnikának az infokommunikáció. A szóösszetétel mutatja, hogy a távközlést a kapcsolatos szolgáltatásokkal és végberendezésekkel együtt tárgyalja. Bár ez a szó még idegenül hangzik, de lehet, hogy e két latin szóból kialakított kifejezés nyelv függetlenül terjed majd el a világban. Hasonlóan minden latin eredetű szóhoz a különböző országok, saját helyesírásukkal, vagy saját kiejtésükkel honosítják majd, hasonlóan a telefonhoz, vagy az automobilhoz.

A távközlési szolgáltatások kiterjesztését és rugalmasságát a különböző szoftverek segítik elő. Bár ez régebben a számítástechnikához rendelt tudomány volt, de a távközlő rendszerek működéséhez és az alkalmazásokhoz is nélkülözhetetlen. Ugyanakkor a követelmények is más jellegűek, mint a számítástechnikai szoftvereké. Míg a számítástechnikában általában feltételezik, hogy a számítógép elé megfelelően képzett szakemberek ülnek le, addig a távközlést mindenki használhatja és ennek megfelelően váratlan jelzéseket, információkat, parancsokat küldhetnek a központi berendezések felé, melyeket azoknak kezelni kell. Ha a parancs értelmezhetetlen, akkor sem szabad, hogy ugyanaz a berendezés mások számára használhatatlanná váljék. Mint minden többfelhasználós szolgáltatásnál itt a megbízhatóság egyik kritériuma, hogy egy felhasználó ne tehesse a szolgáltatást használhatatlanná, vagy hozzáférhetetlenné.

Az infokommunikációs rendszerek szoftverjének további szigorú megkötöttsége, hogy világméretű hálózatokon keresztül kell parancsokat továbbítani, folyamatos üzemet megvalósítani, de szükség van az elszámoláshoz a tarifainformációkra és a különböző gyártmányú rendszerek között az együttműködésre. A felhasználók felé nemzetközi szabványú jeleket, hangokat kell küldeni, mert ország és nyelv függetlenül bárki használhatja ezeket a hálózatokat. Ez nagymértékű robusztusságot igényel.

Indokolt ezért, hogy egy Híradástechnika címet viselő újság foglalkozzon távközlési, és a távközléssel kapcsolatos felhasználói szoftverekkel. Ezt korábban is alkalmanként megtettük, most azonban e havi számunk döntő mértékben foglalkozik szoftver problémákkal. Az első néhány cikk a szoftverek vizsgálatát tűzte ki célul. Mivel rendkívül sok eset fordulhat elő, nehéz egy szoftvert teljes biztonsággal hibátlannak ítélni. Valamennyi létező eset kipróbálása rendkívül hosszú ideig tartana, ezért ésszerű optimumot kell keresni a vizsgálati idő és a hibátlanság valószínűsége között. Több kiváló hazai kutató tanulmányozza ezt a kérdést, és a körülöttük felnövő doktorandusok is számos értékes eredményre jutottak. Ezekből mutatunk be néhányat.

Szoftverek szabják meg a felhasználók számára felkínálható szolgáltatások mennyiségi és minőségi paramétereit, ezért a hálózatok mérésére és igény szerinti kialakítására alkalmas megoldásokat is bemutatunk. Egyik cikkünk a gazdasági kérdéseket, több felhasználó igényének optimális kielégítését és játékelméleti alapon a hálózati minőségi osztályok összerendelését tárgyalja. Bár a játékelmélet már közel 100 éves múltra tekinthet vissza, gyakorlati alkalmazása csak az utóbbi 1-2 évtizedben terjedt el. A különböző matematikai programok ugyanis gyorsabban teszik lehetővé 1-1 ilyen probléma megoldását, és így nem igényel a megoldás a matematikusoktól túl sok időt. Várhatóan a játékelmélet és az ehhez kapcsolódó új szoftverek sok olyan probléma megoldását segítik elő, ahol a döntések egymást befolyásolják. Ez érvényes akár emberekre, ahol az optimális, mindenki számára legkedvezőbb megoldás csak akkor található meg, ha feltételezzük, hogy a felhasználók azonosan ésszerű logikával keresik a közös optimumot.

Bár nehéznek látszik a teljes elektronikai és infokommunikációs műszaki terület lefedése, valamint az ehhez kapcsolódó összes újdonság bemutatása, mégis szeretnénk a lehetőségekhez képest tájékoztatást adni a legérdekesebb területekről. Sőt a jövőben a mikroelektronika, a nanotechnika is szerepelni fog palettánkon.


DALLOS

IPv4-IPv6 átállási technológiák vizsgálata

KOVÁCS ANDRÁS ÉS PETE BÁLINT

A BME műszaki informatika szakos végzős hallgatói

Az internet hálózati protokollja, az IPv4 már közel két évtizede áll szolgálatban. A növekedés miatt gyorsan fogyó hálózati címtartomány bővítéséhez és a hálózattal szemben támasztott új igények kielégítéséhez szükség van az IPv6, a protokoll bevezetésére. Az internetprotokoll I ecserélését átállási technológiák fogják segíteni, melyek lehetővé teszik a két protokoll egyidejű alkalmazását az áttérés köztes időszakában. Jelen cikk keretében néhány, a következő években várhatóan fontos szerepet játszó IPv4-IPv6 átállási mechanizmus (Tunneling, NAT-PT, SOCKS64) vizsgálatára vállalkoztunk. A teljesítményteszteket két széles körben elterjedt operációs rendszeren (FreeBSD és Windows 2000 Professional) végeztük el.

1. Az internetprotokoll

Az IPv4 (Internet Protocol version 4) protokoll az 1960-as években létrehozott ARPA projekt keretében került kifejlesztésre. Három évtized elteltével az eredetileg katonai céloknak alárendelt ARPANET a hidegháború megszűnésével széles körű alkalmazást nyert. Ennek következtében az ARPANET és a vele párhuzamosan működő civil számítógép-hálózatok összekapcsolásával létrejött internet a mai napig töretlen fejlődést mutat. A fejlődés új technológiák, protokollok és módszerek alkalmazásában, új szolgáltatások bevezetésében, valamint a felhasználók számának nagymértékű növekedésében mutatkozik meg.

Az 1990-es évek legelején világossá vált, hogy a 70-es és 80-as évek igényei szerint kidolgozott IPv4-es protokoll hiányosságainak nagy része megoldható volt az IPv4 bővítésével. A megváltozott igények nyomán keletkezett „lyukak" befoltozására újabb protokollok és módszerek kerültek kidolgozásra (pl. alacsony hatékonyságú címzési architektúra lecserélése, biztonság, minőségbiztosítás stb.). Az évek során azonban a legnagyobb problémát a felhasználók számának exponenciális növekedése képezte, amely világossá tette, hogy a harmadik évezred beköszöntével az IPv4-es protokoll 32 bites címtartománya már nem lesz elegendő.

Az internet hosszú távú növekedése érdekében a szakemberek az internetprotokoll újabb verziója, az IPv6 (Internet Protocol version 6) bevezetése mellett döntöttek. Az IPv4-es protokollról az IPv6-osra való átállás napjainkban kezd kitüntetett figyelmet élvezni, ugyanis a protokollváltás megkezdésének szükségessége fenyegetően közeleg. Az átállást az egyik protokollról a másikra átállási mechanizmusok segítségével kívánják megoldani. A cél egy hosszú átállási folyamat után az IPv6-os protokoll kizárólagos alkalmazása.

2. IPv4-IPv6 átállási technológiák áttekintése

Ebben a szakaszban azokról az IPv4-IPv6 átállási technológiákról lesz szó, amelyek vizsgálataink tárgyát képezték. A cél nem a szabványok kimerítő ismertetése, hanem egy olyan áttekintés, amely lehetővé teszi néhány fontosabb átállási technológia működési elvének megértését, vizsgálataink értelmezését.

2.1. Alagutazás

Az átállási mechanizmusok közül az alagutazás (tunneling) célja, hogy a már létező IPv4 infrastruktúrát IPv6-os csomagok továbbítására is felhasználhassuk, ezáltal biztosítva a közvetlen IPv6-os kapcsolattal nem rendelkező végpontok hozzáférését az új protokollt használó hálózatokhoz [1]. Az alagutazás lényege, hogy a segítségével kommunikáló végpontok (az alagút végpontjai) között az IPv6-os csomagok továbbítása a hálózat IPv4-es csomagjainak adatmezejében, egy külön e célra kiosztott protokollazonosítót használnak. A protokollváltás későbbi szakaszaiban, amikor már széles körben kiépült az IPv6-os infrastruktúra, megfordul a helyzet, ekkor az IPv4-es csomagok IPv6 feletti továbbítására lesz szükség.

Az alagút küldő végpontja a továbbítandó IPv6-os üzenetet egy IPv4-es csomagba helyezi (encapsulation, becsomagolás), majd továbbítja azt a másik végpontnak a hagyományos IPv4-es mechanizmusok által. Az alagút másik végén a vevő, a megfelelő biztonsági ellenőrzések után kiveszi az adatmezőben lévő IPv6-os csomagot (decapsulation, kicsomagolás), amit a továbbítási szabályok alapján fel is dolgoz. Egy-egy ilyen alagutas összeköttetés az IPv6 útválasztás szempontjából megfelel egy hagyományos hálózati összeköttetésnek (link), amelynek hossza egy ugrás (hop). A tipikus megvalósítás pszeudo interfészt alkalmaz, amely

a felső (IPv6) réteg számára egy adatkapcsolati összeköttetésként jelenik meg az alagút másik végpontját alkotó berendezéshez.

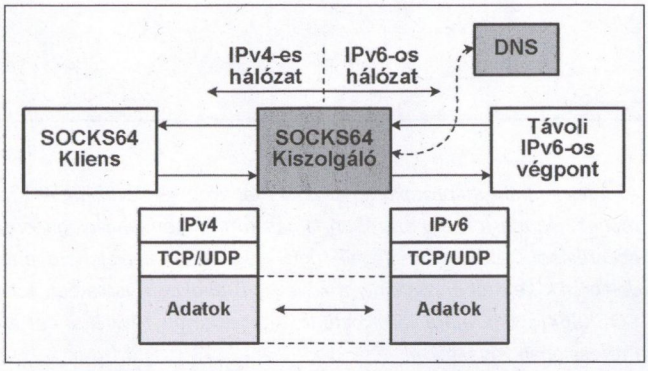
Az alagutak lehetnek automatikusak vagy statikusak (konfigurált). Az automatikus alagutak esetében az IPv4-es célcím az IPv6-os csomag úti céljából meghatározható (amikor az egy beágyazott IPv4-es címet tartalmaz), míg egyéb esetekben a célcímet előzetes konfiguráció és a csomagtovábbítási táblázatok alapján kell meghatározni. Az utóbbi esetben az alagút beágyazó végpontjának előre tudnia kell, hogy mely – beágyazott IPv6-os csomagokat elfogadó – útválasztón keresztül mely hálózatok érhetők el, valamint a kicsomagoló útválasztókat is fel kell készíteni az alagutazott datagramok fogadására. A konfigurált alagutakat nem kell mindig manuálisan beállítani, léteznek úgynevezett „alagútbrókerek" amelyektől a végpontok alagutas összeköttetést igényelhetnek egy IPv6-os útválasztóval.

2.2. SOCKS64

Napjainkban az internetfelhasználók nagy száma miatt sok szervezet üzemeltet tűzfalakat (Firewall), hogy jogosulatlan felhasználók ne tudjanak a szervezet belső hálózatára behatolni, és a belső hálózaton lévő felhasználók ne tudjanak biztonságot csökkentő forgalmat a külső hálózatra juttatni (pl. TELNET, FTP). A SOCKS5 (SOCKS Protocol Version 5) protokollt arra dolgozták ki, hogy biztosítható legyen az egyes felhasználói protokollok tűzfalon keresztüli – hitelesítéshez kötött – transzparenshasználata. Később aztán a SOCKS5 protokoll kisebb módosításával született meg a SOCKS64, amely már magában foglalja az IPv4-IPv6 kommunikációt támogató protokollelemeket is [2].

A SOCKS64 protokoll egy kliens-szerver alapú megoldás, amely tulajdonképpen tűzfalak külső oldalán – a kliens által meghatározott távoli címre – TCP/UDP socketek vezérelt nyitását teszi lehetővé. A protokoll egy rendkívül érdekes megoldással biztosítja a kliens oldali adatcserét: a felhasznált alkalmazás hálózati kommunikációjához szükséges rendszerhívásokat a SOCKS „elkapja", majd a kiszolgálóval történő egyeztetés után szabad utat biztosít a végalkalmazások közötti adatcserének. Természetesen ez a működési elv megköveteli a kliens oldali szoftverbővítést, amely hátránya a SOCKS protokolloknak. A szoftverbővítés általában a hálózati függvénykönyvtárak telepítését, rosszabb esetben a teljes alkalmazás újrafordítását jelenti, de maga az alkalmazás is támogathatja a SOCKS üzenetek kezelését.

Kövessük végig, hogy miként épül fel egy SOCKS kommunikáció! Első lépésként a SOCKS kiszolgáló „belső" oldalán lévő kliens megpróbál csatlakozni a felhasználó által megadott távoli végponthoz. Mivel ez a SOCKS szoftveren keresztül történik, az alkalmazás nem a kívánt távoli végponthoz csatlakozik, hanem a SOCKS kiszolgálóhoz. Ezután következik a felhasználó hitelesítése (pl. felhasználó/jelszó, Kerberos stb.), majd – sikeres hitelesítés esetén – a kliens oldali SOCKS szoftver elküldi a kívánt távoli végpont címét a kiszolgálónak, amely – immár a külső oldalon – csatlakozik a távoli végponthoz. Innen kezdve megindulhat a két végpont közötti adatcsere, a kiszolgáló feladata csupán az, hogy az egyik hálózat irányából érkező szállítási szintű forgalmat áthelyezze a másik hálózat felé menő csomagokba. A TCP protokoll használata esetén a kiszolgáló mindkét oldalán egy független TCP kapcsolat meglétét tételezi fel (1. ábra). A felhasznált hálózati protokoll már a SOCKS5 esetében lehetett IPv4 vagy IPv6, de kizárólagosan csak az egyik.



1. ábra SOCKS64 hálózati elrendezés

Mivel a kiszolgáló belső és külső oldalán lévő TCP kapcsolatok egymástól függetlenek, felmerült a heterogén hálózati protokollok használatának lehetősége, amelyet a SOCKS64 protokoll támogat. A kommunikációs elrendezés itt csak IPv4- vagy csak IPv6-képes klienseket feltételez a SOCKS64 kiszolgáló belső hálózatán, míg a külső részen pont a másik hálózati protokollt ismerő célszámítógépek helyezkednek el. Így a SOCKS lehetővé teszi az IPv4 és IPv6 bármely kombinációjának alkalmazását.

Említettük már a fentiekben, hogy a kliens a kívánt végpont címét elküldi a SOCKS kiszolgálónak, ez a SOCKS5 esetében lehetett IPv4/IPv6-os hálózati cím vagy domainnév. A SOCKS64 ezt úgy viszi tovább, hogy a kliens oldali szoftverekben helyet kapnak a DNS névfeloldó függvények SOCKS64-re átültetett változatai is. A működéshez a következő példát mutatjuk be.

Tekintsünk adottnak a belső hálózaton egy csak IPv4-képes végpontot, amely egy SOCKS64 kiszolgálón keresztül szeretne kapcsolatba lépni egy távoli IPv6-os számítógéppel. Első lépésként a felhasználó megadja az IPv6-os célcímet, amelyet természetesen számítógépe nem ismer, hiszen a felhasználói alkalmazások csak az IPv4-es címeket kezelik. A megadott IPv6-os hálózati címeket az IPv4-es alkalmazások domain névként fogják értelmezni, így a DNS rendszerhez fordulnak. A kliens oldali SOCKS64 szoftver természetesen a DNS névfeloldást végző rendszerhívásokat lecseréli, így becsaphatja azzal az IPv4-es számítógépet, hogy egy hamis IPv4-es címet ad vissza (pl. 0.0.0.X), miközben az IPv6-os címet elküldi a kiszolgálónak értelmezésre. Az a csomópont, amelyen a kiszolgáló fut, mind az IPv6-os, mind pedig az IPv4-es protokollt ismeri, így nem határozhatja meg a távoli végpont címét. A kommunikáció közben a kliens oldali alkalmazás a hamis IPv4-es hálózati cím segítségével

„kommunikál", miközben nem tud arról, hogy a kiszolgáló másik oldalán már IPv6 felett továbbítódnak adatai. A kommunikáció fordított esete hasonlóan zajlik, így erre nem térünk ki.

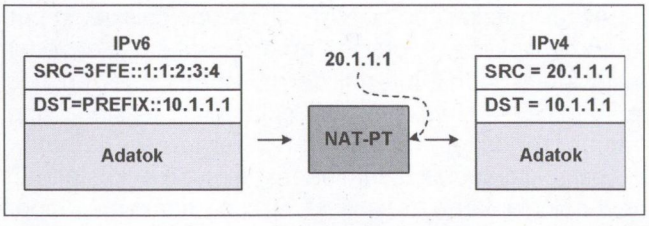
A SOCKS64 alkalmazása azért lesz nagyon fontos az IPv4-IPv6 átállás folyamán, mert a két hálózati protokoll minden kombinációját ismeri. Szintén fontos szerepet játszhat elszigetelt vállalati hálózatok IPv6-os protokollra való áttérésében. Nagy hátránya a módszernek, hogy csak a szállítási szintű forgalmat képes átvinni, ez akár okozhat kisebb-nagyobb bonyodalmakat is (pl. ICMP hibaüzenetek). További hátrányként értékelhető, hogy a SOCKS64 kliens oldali szoftveres bővítést igényel, amely valószínűleg riasztó lesz a nagy hálózatokat igazgató rendszergazdák számára.

2.3. NAT-PT

A NAT-PT (Network Address Translation – Protocol Translation) egy rugalmas módszer, amely transzparens módon teszi lehetővé az IPv4-es és IPv6-os, valamint a fordított irányú vég-végpont kommunikációt. A NAT-PT technológiát az IPv4-IPv4 címfordítást végző, a címtartomány fogyásának lassítására kidolgozott NAT (Network Address Translation) alapján hozták létre, amelyet kibővítettek IPv4-IPv6 címfeloldási mechanizmusokkal és protokollfordítást biztosító eljárásokkal [3].

A NAT-PT átjáróként (gateway) funkcionál az IPv4-es és IPv6-os hálózatok határfelületén. A hálózat IPv6-os oldalán a NAT-PT kommunikáció számára egy speciális címelőtagot (prefix) tartanak fenn, amely lehetővé teszi, hogy ezzel a címelőtaggal rendelkező csomagokat a NAT-PT átjáróhoz irányítsák. A NAT-PT számára egy 96 bit hosszúságú prefixet kell fenntartani, amely a belső hálózaton az útválasztók által kerül hirdetésre, továbbá a NAT-PT mechanizmust használó IPv6-os hálózatnak szüksége van egy IPv4-es címtartományra is.

Amennyiben egy IPv6-os végpont egy – a NAT-PT átjáró túloldalán lévő – IPv4-es számítógéppel szeretne kommunikálni, akkor speciális IPv6-os célcímmel rendelkező adatcsomagokat kell küldenie a NAT-PT átjáró felé. A célcím első 96 bitje természetesen a NAT-PT számára fenntartott előtag, az alsó 32 bitje viszont a célszámítógép IPv4-es címe. Amint a NAT-PT átjáróhoz megérkezik a csomag, a protokoll-fordítás megtörténik, az IPv4-es célcím az IPv6-os célcím alsó 32 bitje, míg az IPv4-es forráscím az IPv6-os hálózat számára fenntartott IPv4-es címtartományból fog dinamikusan kikerülni (2. ábra). Ennek legfőbb előnye, hogy több IPv6-os NAT-PT-t alkalmazó számítógép lehet a hálózatban, mint amekkora a rendelkezésre álló IPv4-es címtartomány. Természetesen a NAT-PT átjáró egy belső táblázatban a kapcsolat időtartamára feljegyzi az IPv6-IPv4-es címek összerendelését, amely meghatározza az egyes csomagok cél- és forráscímeit.



2. ábra IPv6-IPv4 irányú kommunikáció a NAT-PT segítségével

Jóval bonyolultabb a helyzet, ha a kommunikáció az IPv4-es oldalról indul, hiszen a 32 bit hosszúságú IPv4-es címek nem tartalmazhatnak járulékos információkat az IPv6-os hálózatban lévő célszámítógépek kijelölésére. Ebben az esetben a NAT-PT átjáró belső táblázatában statikus IPv4-IPv6 címösszerendelésekre van szükség, amiből pedig azonnal következik, hogy annyi IPv6-os végpont alkalmazhat NAT-PT-t a hálózatban, amekkora a rendelkezésre álló IPv4-es címtartomány. Itt röviden megjegyezzük, hogy ez a probléma feloldható, amennyiben a NAT-PT átjáró az IPv4-es végpont által az IPv6-os hálózatba küldött DNS lekérdezések tartalmának elemzésével készíti el belső táblázatában a címfordítási szabályokat és oszt ki dinamikusan IPv4-es célcímet az IPv6-os végpontnak.

Röviden szólnunk kell a protokollfordításról is, amely nem más, mint az IPv4-es és IPv6-os csomagok fejlécmezőinek kölcsönös megfeleltetése, valamint a felsőbb szintű protokollok által szállított hálózati információk módosítása (pl. FTP). A fordítás mechanizmusát az SIIT (Stateless IP/ICMP Translation Algorithm) algoritmus definiálja [4], amely – mint a neve is mutatja – állapotmentes, azaz a rajta áthaladó datagramokat egymástól függetlenül kezeli.

A NAT-PT egy nagyon fontos átállási mechanizmusnak ígérkezik, amelynek alkalmazhatósága széles körű. Konfigurációja egyszerű, kevés szakértelmet követel, továbbá nem igényel a végpontokon szoftveres bővítést. Több NAT-PT implementáció is létezik ma már, néhány ezek közül ingyenesen hozzáférhető.

3. Megvalósítások vizsgálata

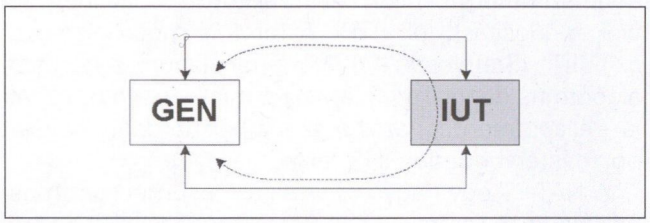
Elvégeztük a fentiekben vázlatosan ismertetett IPv4-IPv6 átállási technológiák megvalósításának vizsgálatát, azokat értékeltük, valamint meghatároztuk az egyes technológiák alkalmazhatósági körét. Az általunk elvégzett teljesítménymérések az egyes mechanizmusokat megvalósító implementációk csomagtovábbító képességeire terjedtek ki, nem foglalkoztunk tehát kliens oldali mérésekkel, csupán az útválasztói funkciókra koncentráltunk.

A mérések során vizsgáltuk az ismertetett átállási technológiák átlagos csomagkésleltetését és csomagvesztését a másodpercenkénti csomagszám függvényében, továbbá meghatároztuk az egyes technológiák kapcsolatfelépítési idejét a TCP protokoll esetében. Végül figyelemmel kísértük az útválasztóként működő számítógép-processzor kihasználtságát, hogy az átállási technológiákat a számítási kapacitásigény szerint is osztályozhassuk. Fontosnak tartjuk, hogy a rendelkezésre álló alkalmazások által nyújtott teljesítményparaméterek több – széles körben elterjedt – platformon is

rendelkezésre álljanak, így a méréseket FreeBSD, valamint Windows 2000 operációs rendszereken végeztük. Előre leszögezzük azonban, hogy vizsgálatainkat korántsem az operációs rendszerek összehasonlítása vezette, hanem – a fejlesztés szakaszában lévő – IPv4-IPv6 átállási technológiák szemléletes összemérése és két elterjedt platformon történő kipróbálása.

Teljesítménymérés

A mérés során alkalmazott hálózat logikai rajza a 3. ábrán látható. Az IPv4-IPv6 átállási technológiák implementációit az IUT-n (Implementation Under Test) futtattuk, amely egy Intel Pentium II 400 MHz-es központi egységgel rendelkező számítógép. Az IUT FreeBSD 4.3 operációs rendszerrel és a KAME csoport IPv6-os protokollvermével, valamint Windows 2000 Professional operációs rendszerrel és a Microsoft kísérleti IPv6-os protokollvermével volt felszerelve. A mérőforgalom előállítása és elemzése az ábrán GEN-el jelölt számítógép feladata volt, amely egy Intel Pentium III 733MHz-es CPU-val rendelkezett, operációs rendszere pedig egy SuSe Linux 6.2 volt.



3. ábra A mérőhálózat elrendezése

A 3. ábrán látható hálózat fizikailag egy 100Mbps sebességű Ethernet hálózat volt. Az IUT és a GEN számítógépeket többletkésleltetést okozó hálózati eszközök nélkül, csupán keresztkábellel kötöttük össze. Mind a GEN, mind az IUT két hálózati kártyával rendelkezett.

A mérések során a GEN számítógép egyik hálózati interfészén csomagokat küld az IUT felé, amely a megfelelő IPv4-IPv6 átállási technológiát alkalmazva, útválasztóként működve továbbítja a GEN másik interfésze felé az adatcsomagokat. A körülfordulási idő mérése a GEN számítógépen a csomagba helyezett időbélyeggel történik, amelyet levonunk a csomag érkezési idejéből. A mérőforgalom előállítását, fogadását, továbbá az időbélyegek kezelését egy általunk telepített szoftverrel végeztük el.

Minden esetben 200 byte méretű csomagokat használtunk, amely azért volt fontos, mert az ilyen jellegű méréseknél nem az átvitt adatok mennyisége a mérvadó, hanem a csomagok nagy száma.

Fontos feltérképezni azokat a nem kívánt hatásokat, amelyek rossz irányban befolyásolhatják a mérések eredményét. Az első és legfontosabb kérdés az lehetne, hogy vajon miként kerülhetjük el a GEN számítógép protokollvermét, hiszen ha az előállított forgalom induláskor és érkezéskor is keresztülhalad ezen, akkor a GEN is jelentősen beleszólhat a mért eredmények alakulásába. Erre megoldás a Linux operációs rendszer PACKET interfészének használata, amely lehetőséget biztosít a felhasználónak arra, hogy a teljes protokollvermet megkerülve küldje/fogadja az adatcsomagokat. A PACKET interfész alkalmazásakor még az adatkapcsolati keretet is a felhasználónak kell összeállítania, amely küldés esetén azonnal a hálózati kártyára kerül.

A körülfordulási késleltetést a csomag adatmezejében utazó időbélyeg segít meghatározni, amelyet közvetlenül a keret elküldése előtt helyezünk el benne. Kritikus lehet az érkezési időbélyeg számítása, hiszen a GEN bemenetén hosszú várakozási sor alakulhat ki, ettől jócskán torzulhatnak a késleltetésértékek. A Linux operációs rendszer azonban automatikusan rögzíti minden csomag beérkezési idejét, ezt fogjuk felhasználni érkezési időbélyegként.

A következő zavaró hatások befolyásolhatják méréseink eredményét. Egyrészt gond lehet a hálózati interfészek keretfeldolgozási késleltetése, másrészt az IUT eszközmeghajtó programjaitól és természetesen operációs rendszerének gyorsaságától is nagyban függhetnek a mért körülfordulási idők. További kérdésként merül fel, hogy a Linux operációs rendszer milyen pontos időbélyeget képes szolgáltatni, de az is, hogy mekkora az ütemezés által okozott veszteség. Ezen problémák vizsgálata kívül esik tanulmányunk keretein.

Mérési eredmények

Ebben a szakaszban bemutatjuk az IPv4-IPv6-os átállási technológiákkal kapcsolatos csomagkésleltetési időt és csomagáteresztést, valamint az IPv4-IPv6 átállási technológiákat futtató útválasztó CPU-kihasználtságát vizsgáljuk. Míg az előbbi paramétereket egy általunk implementált szoftverrel (ld. korábban), addig utóbbit az útválasztó operációs rendszerével (FreeBSD, Windows) szállított segédprogramokkal rögzítettük (Top program ill. Task Manager). Itt vetítenénk előre, hogy a FreeBSD operációs rendszeren néhány további vizsgálatot is elvégeztünk, közelebbről az IPv4-IPv6 átállási technológiákon keresztüli TCP kapcsolatfelépítés idejét, valamint a NAT-PT belső táblázatának méretétől függő késleltetés megnövekedését figyeltük meg.

A mérőforgalom minden esetben 5 másodpercig áramlott át az útválasztás feladatait ellátó számítógépen, ezalatt minden egyes csomag késleltetését rögzítettük, majd a késleltetési értékekből átlagot számítottunk. A csomagáteresztés nagyságát ezer csomag/másodpercben adtuk meg, amelyet – a mérési intervallumon belül – a forgalomgenerátor által kibocsátott, ill. fogadott csomagok száma alapján határoztunk meg.

Az alagutazás és a NAT-PT technológiákra kapott eredményeket – operációs rendszerenként – egyetlen nagy grafikonon tüntettük fel (4. ábra), amelyen felül a FreeBSD 4.3, alul a Windows 2000 Professional operációs rendszer teljesítményparaméterei láthatók. A bal oldali ábrák szemléltetik a csomagkésleltetés technológiánkénti alakulását a csomagok másodpercenkénti számának függvényében, míg a jobb oldaliak

a csomagáteresztés értékét mutatják. A két különböző átállási mechanizmus működése közben az útválasztón mért CPU-kihasználtság pedig az 5. ábrán látható, szintén a másodpercenkénti csomagsebesség függvényében. Az 6. ábrán a FreeBSD operációs rendszere felett futó Permeo Technologies INC. SOCKS64 implementációján mért paraméterek láthatók hasonló formában. A SOCKS64 technológiával kapcsolatos mérések – megfelelő implementáció hiányában – nem kerültek elvégzésre Windows 2000 Professional operációs rendszer alatt. Végül az egyes átállási technológiákon megfigyelt TCP kapcsolatfelépítés idejét a 7. ábrán, a NAT-PT belső táblázatával kapcsolatos vizsgálat eredményét pedig a 8. ábrán tüntettük fel, ezek szintén csak a FreeBSD operációs rendszeren készültek el.

IPv4 és IPv6 csomagtovábbítás

Legelső lépésként megvizsgáltuk az IUT IPv4-IPv4 és IPv6-IPv6 csomagtovábbítási képességeit, amely biztos összehasonlítási alapot nyújt a technológiák további értékeléséhez. A 4. ábrán látható, hogy kisebb csomagsebességeknél a két különböző hálózati protokoll csomagtovábbítási képességei hasonlók, kismértékben azonban az IPv6-IPv6 átvitel bizonyult gyengébbnek. Ez megfelel várakozásainknak, hiszen egy IPv4-es datagram továbbítása jóval kisebb terhet ró egy útválasztóra, mint egy IPv6-os csomagé, hiszen a 128 bit hosszúságú IPv6-os célcímek keresése a csomagtovábbítási táblázatban nagyobb számítási kapacitást igényel.

A nagyobb tartományoknál már drasztikusabb a különbség. Jól megfigyelhető az IPv6-os átvitel alakulásán, hogy másodpercenként 28 ezer csomag alatt egyik esetben sincs még csomagvesztés, azonban ennél nagyobb terhelésnél drasztikus késleltetésnövekedést és csomagvesztést tapasztalunk mindkét útválasztó esetén. A veszteség növekedése oly nagy, hogy a fordulópont után – ahol megkezdődik a csomagok eldobása – már egyre kisebb adatmennyiséget képes továbbítani az IUT. Ezzel szemben az IPv4 továbbításnál a növekedő terhelés mellett is fennmarad a fordulóponton mért csomagátvitel – a túlterhelés nem okoz csökkenést az átbocsátásban. A leírt jelenség annak tudható be, hogy még jelenleg is tart a vizsgált IPv6 (KAME és Microsoft Research) fejlesztése, ezért ezek még nem olyan kiforrottak, mint a már másfél évtizede folyamatosan használt és fejlesztett IPv4.

Az átvitt csomagok számának visszaesése a FreeBSD-nél valamivel később (30 ezer) jelentkezik, mint a Windows 2000-nél (28 ezer), azonban ez a különbség elhanyagolható. Másodpercenkénti 50 ezer csomagnál – ez már nagyon közel van a 100Mbps sebességű Ethernet határához – mindkét implementáció visszaesik 20 ezer csomagra. Egy másik érdekes megfigyelés, hogy a nagy sebességnél a Windows 2000 eléri a 100%-os processzorkihasználtságot (28 ezer), a FreeBSD ekkor még csak 90%-ot használ fel, és csak 50 ezer csomagnál éri el a teljes kihasználtságot – tehát elméletileg mindenképpen lehetséges lenne jobb teljesítmény.

Az IPv4 továbbításnál is van egy figyelemre méltó eltérés a két operációs rendszer között – a FreeBSD-nél másodpercenkénti 40 ezer csomag felett az IPv4 továbbításnál is jelentkeznek a fent leírt problémák –, az átvitt csomagmennyiség leromlik a stabilizálódás helyett, valamint a késleltetés is megnövekszik.

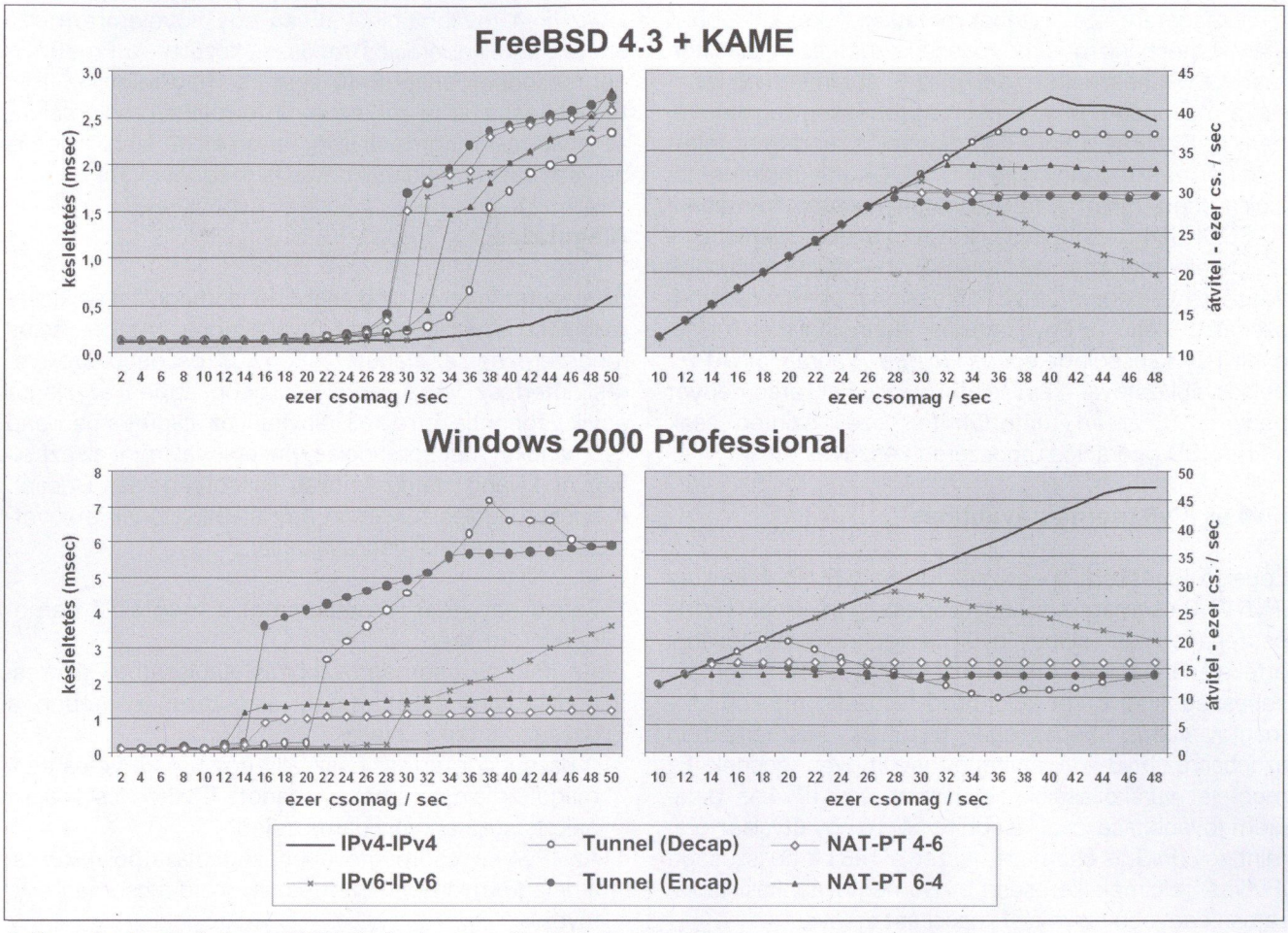
Alagutazás

Az alagutazás technológiánál a becsomagolást és kicsomagolást is vizsgáltuk az útválasztónál. Az operációs rendszerekben az alagutazást egy ún. pszeudo (szoftveres) interfész végzi, amely tulajdonképpen az alagút egyik végpontja, erre kell irányítani az alagútba bemenő csomagokat (becsomagolás, Encap), valamint az azt elhagyni kívánó datagramokat (kicsomagolás, Decap). A becsomagolás folyamán egy útválasztónak a következő lépéseket kell végrehajtania:

1. Fogad egy IPv6-os datagramot a megfelelő bemeneti interfészén.
2. Az IPv6-os csomagtovábbítási táblázatból kikeresi a datagram célját, amely alagutazás esetében a pszeudo interfész lesz.
3. A pszeudo interfész összeállít egy üres IPv4-es csomagot, majd az imént fogadott IPv6-os csomagot belemásolja annak adatmezejébe.
4. Az IPv4-es csomagtovábbítási táblázatból kikeresi a datagram kimenő interfészét, majd a csomag erre távozik.

A kicsomagolás ehhez képest annyiban változik, hogy a bejövő IPv4-es csomag adatmezejéből ki kell venni az IPv6-os datagramot, majd a csomagtovábbítási táblázat alapján a megfelelő kimeneti interfészen azt ki kell adni. Jól látható, hogy mindkét esetben egy IPv4-es, illetve egy IPv6-os útválasztási döntésre van szükség, azonban csak a becsomagolás esetében szükséges egy teljesen új csomag összeállítása.

A mért eredményekből megfigyelhető, hogy mindkét implementáció esetében a becsomagolás valamivel rosszabbul teljesített, mint a kicsomagolás. Ez pontosan megfelel várakozásainknak, ugyanis a becsomagolás esetében többletmunkát jelent az IPv4-os fejléc összeállítása. A FreeBSD esetén sem a becsomagolásnál, sem a kicsomagolásnál nem figyelhető meg a degradáció, amit az IPv6-nál tapasztaltunk. Ráadásul a kicsomagolás másodpercenkénti 37 ezer csomagos maximális átbocsátása jelentősen kedvezőbb az IPv6 esetében mért 30 ezer csomagnál. Ez arra enged következtetni, hogy az IPv6-nál tapasztalt drasztikus teljesítményromlás okát nem az IPv6 csomagok küldésében, hanem minden valószínűség szerint a vételi oldalon vagy a várakozási sorokban kell keresni. A becsomagolás (amikor egy IPv6-os csomagot veszünk, és egy IPv4-es csomagot küldünk) csúcsteljesítménye körülbelül megegyezik az IPv6 teljesítményével, bár ezen esetben nem tapasztalható az átbocsátás csökkenése növekvő terhelés mellett.



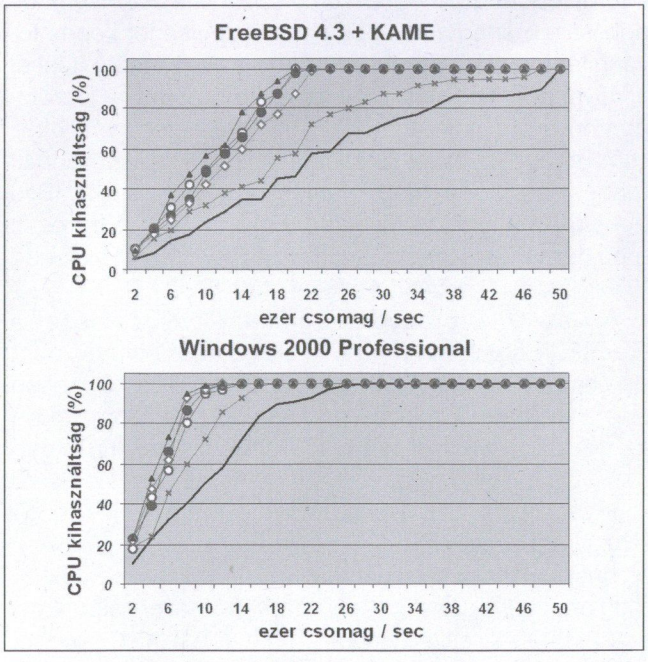
4. ábra Az alagutazás és a NAT-PT teljesítményparaméterei

A Windows 2000 esetében kicsit más a helyzet. Itt is megfigyelhető, hogy a kicsomagolás általánosságban véve jobban teljesít a becsomagolásnál, azonban ez esetben egyik átbocsátás sem hasonlítható az IPv6 képességeihez, ugyanis 15-21 ezer csomagnál nem lehet vele többet átvinni másodpercenként. A kicsomagolásnál ráadásul 20 ezer csomagos terhelés felett az IPv6-hoz hasonlóan leromlik az átbocsátás és megnövekszik a késleltetés, majd 36 ezer csomagtól lassan visszaáll a 20 ezer csomagos átbocsátásra és a becsomagolásnál mért késleltetésértékre. Ennek a szokatlan viselkedésnek az oka valószínűleg az ütemezésben keresendő, ugyanis a Windows 2000-ben a teljes csomagtovábbítási rendszer – aminek része az alagút pszeudo interfész meghajtója – egy szolgáltatásként fut, míg a FreeBSD esetében az a monolitikus kernel részét képzi, függetlenül attól, hogy közvetlenül fordítjuk a kernelbe vagy modulként, futás közben töltjük be.

NAT-PT

A NAT-PT esetében az IPv4-IPv6 és az IPv6-IPv4 irányú csomagtovábbítást vizsgáltuk. Azt várjuk, hogy az eddig tárgyalt módszerek közül a NAT-PT fogja a legrosszabb teljesítményeredményeket produkálni. Az IPv4, illetve IPv6 protokollok fejléceinek fordítása, a NAT-PT belső táblázatának kezelése mind-mind arra engednek következtetni, hogy ez a technológia követeli a legtöbb erőforrást az útválasztótól.

Az eredmények is ezt támasztják alá, hiszen a FreeBSD esetében az IPv4-IPv6 irány teljesítménye körülbelül azonos a becsomagolásnál mértekkel, bár egy hajszálnyival



5. ábra Az útválasztó CPU-kihasználtsága

talán jobb is annál, míg az IPv6-IPv4 irány végig a be- és kicsomagolás értékei között van félúton. Érthető, hogy az IPv6-IPv4 irány jobban teljesít, mint a fordított, ugyanis egy IPv6-os fejléc összerakása és továbbítása nehezebb feladat, mint egy IPv4-es csomag küldése. Érdekes azonban, hogy a NAT-PT, bár IPv6-os csomagokat küld és fogad, nem tanúsítja az IPv6 továbbításnál tapasztalt teljesítményromlást. Az IPv6-IPv4 irány csúcsteljesítménye nagyjából megfelel az IPv6 csúcsteljesítményének, a IPv4-IPv6 teljesítménye pedig meg is haladja azt – ez ismét alátámasztja, hogy nem a csomagküldésben keresendő az IPv6 továbbítás teljesítményromlásának oka.

Windows 2000 esetében azonban a NAT-PT mindkét irányának átbocsátóképessége hasonlóan alakul, mint az alagutazás két iránya, bár itt nem jelentkezik az átbocsátás ingadozása, csökkenése, ráadásul a csomagkésleltetés is elég kedvező, a nagyobb terhelés mellett sem növekszik tovább. Ennek az oka minden valószínűség szerint ismét az ütemezés, ugyanis a NAT-PT az alagutazáshoz hasonlóan szolgáltatásként fut, de nem a csomagtovábbítás részeként, hanem önállóan.

SOCKS64

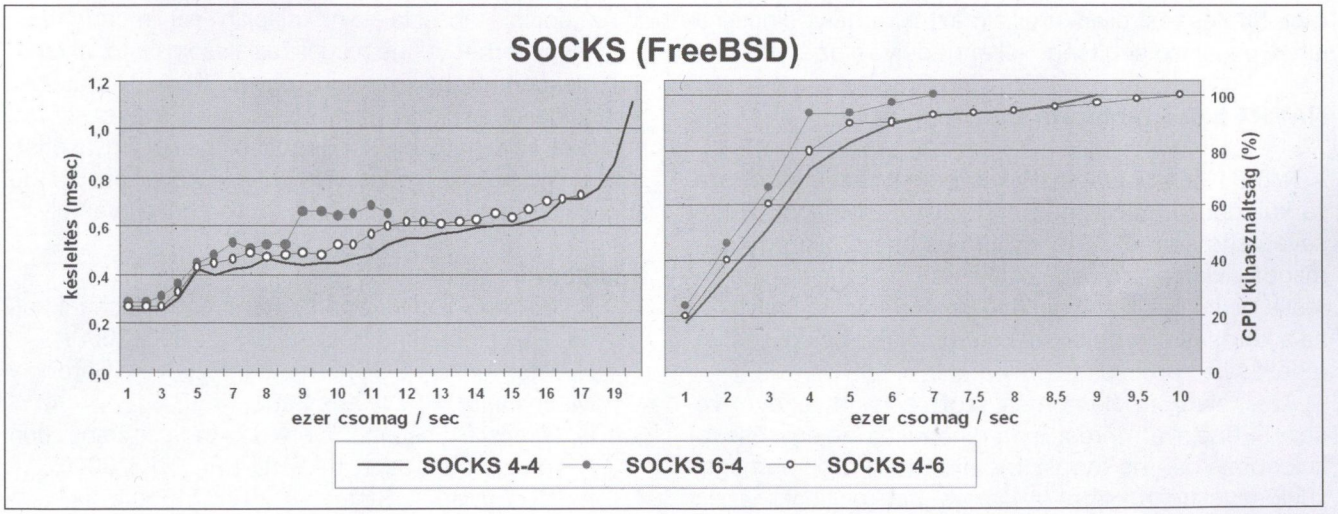
Ha a SOCKS esetében is a Linux operációs rendszer PACKET interfész használata nehéz, hiszen implementálnunk kellene a TCP protokollt, annak minden bonyolultságával együtt. Ezt elkerülendő, a szoftverünket úgy módosítottuk, hogy egy szimpla TCP socketen keresztül állítsa elő a mérőforgalmat. Óriási hátrány, hogy ebben az esetben kétszer haladnak keresztül csomagjaink a helyi protokollvermen, egyszer a küldési, egyszer a fogadási oldalon. Ehhez a méréshez már a mérést végző Linux operációs rendszert is bővíteni kellett az IPv6-os protokollal, itt az USAGI csoport IPv6-os Linux magját (kernel) használtuk.

A mérőforgalom jellege az előzőekhez hasonló, a TCP adatfolyam 200 byte nagyságú csomagokban visszük át. Szólni kell azonban arról is, hogy a TCP folyamprotokoll jellege miatt nem engedi beleszólni a felhasználókat az átvitt TCP csomagok méretébe, így ezt a TCP szegmens méretének (egy csomagban maximálisan átvihető TCP adat mennyisége) megváltoztatásával oldottuk meg.

Az általunk készített forgalomgenerátort a Permeo által szállított SOCKS64 programozási könyvtárakkal (library) kellett összefűzni, amelyet gond nélkül sikerült végrehajtanunk. Mielőtt rátérnénk az eredmények ismertetésére, meg kell jegyeznünk, hogy az IPv6-IPv6 irányú forgalom vizsgálata kimaradt, ugyanis a SOCKS64 könyvtárakon keresztül képtelenek voltunk ilyen jellegű átvitelt létrehozni.

A SOCKS protokollok közül az IPv4-IPv4 átvitel okozta a legkisebb késleltetést (6. ábra). A 4. ábrával összehasonlítva az eredményeket, azonnal kitűnik, hogy a SOCKS – alacsony csomagsebességek esetén is – kb. háromszoros késleltetésnövekedést okoz, mint a többi átállási technológia. Nagyobb csomagsebességek felé haladva a három különböző SOCKS alapú átviteli módszer viszonylag egyenletesen veszi fel a terhelést, azonban a tízezres tartományt elérve a technológia kifullad. Ez főleg a nagy számítási kapacitást igénylő dupla TCP kapcsolatra érvényes, amely annyira terheli a FreeBSD operációs rendszer magját, hogy a SOCKS kiszolgáló nem jut elég processzoridőhöz, így feladatát nem képes ellátni. Jól megfigyelhető ez a 6. ábra jobb oldali grafikonján, amely az IUT CPU-kihasználtságát mutatja: 5-6 ezer átvitt csomagnál már minden esetben 90% körüli a processzor terheltsége.

Az átvitel közben történő csomagvesztés vizsgálatakor arra jutottunk, hogy amint az IUT központi egysége elér egy terhelési szintet (kb. 90%), az átvitt csomagok száma gyakorlatilag néhány száz csomag/másodpercre esik vissza. Ez pedig már nem tekinthető megbízható átvitelnek. A megfigyelt három átviteli irány közül a második helyen az IPv4-IPv6 irányú SOCKS átvitel futott be, majd ezt követte az IPv6-IPv4 irány. Ennek megfelelő a CPU-kihasználtság is, hiszen az utóbbi mérés során kaptuk a legnagyobb számítási kapacitásigényt. Jól látható, hogy már ötezer csomag/másodperces sebességnél kritikus szintet üt meg a központi egység kihasználtsága, amely aztán 11-12 ezer cso-



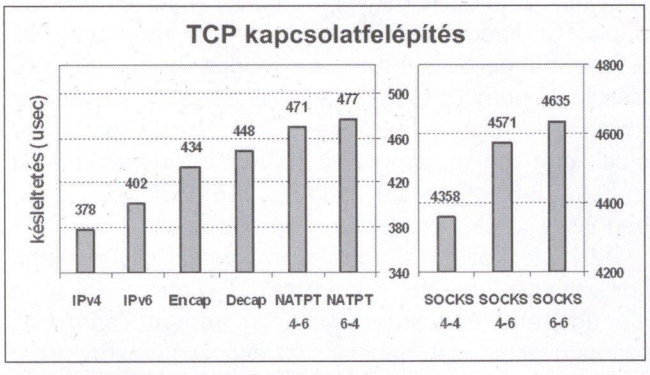
6. ábra A SOCKS protokollok teljesítményparaméterei

mag másodpercenkénti átvitelét teszi csak lehetővé az IPv6-os és IPv4-es hálózatok között.

TCP kapcsolatfelépítés ideje

A TCP kapcsolatfelépítés idejét a GEN számítógépen kiadott blokkolódó connect() függvényhívás visszatérési idejének mérésével határoztuk meg. Technológiánként a kapcsolatfelépítés idejét ötvenszer figyeltük meg, majd az így kapott értékek átlagát ábrázoltuk a 7. ábrán. Megjegyezzük, hogy ez a mérés csak FreeBSD 4.3 operációs rendszeren készült el.

A TCP kapcsolat leggyorsabban – összhangban a 4. ábra eredményeivel – az IPv4-IPv4, majd az IPv6-IPv6 esetén épült fel. Ezt követte az alagutazás, amely érdekes módon fordított eredményt adott: a becsomagolás valamivel jobb eredményt produkált, mint a kicsomagolás. Az alagutazást a NAT-PT követi a sorban, ahol mind az IPv4-IPv6, mind az IPv6-IPv4 irány hasonló eredményt szolgáltatott.



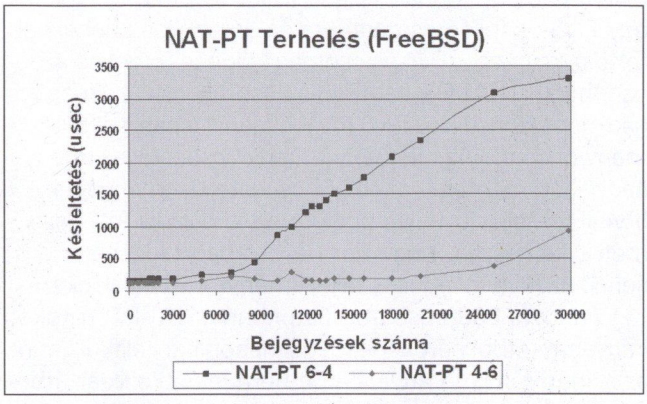
7. ábra A TCP kapcsolatfelépítés ideje

A 7. ábrán a SOCKS64 technológia is szerepel, a várakozásokkal egybevágóan ennél kapjuk a leghosszabb kapcsolatfelépítési időt. Az ábrából kitűnik, hogy a SOCKS alapú kommunikáció megkezdéséhez kb. tízszer annyi idő kell, mint az alagutazás és a NAT-PT esetében. Ez érthető, mert a SOCKS két független TCP kapcsolatot épít fel a kiszolgáló külső és belső oldalán, továbbá hitelesíti is a felhasználót. Pedig a legegyszerűbb hitelesítési eljárást alkalmaztuk, amely felhasználónév és jelszó segítségével engedélyezi az átvitelt.

NAT-PT belső táblázata

A NAT-PT átjáróban dinamikus bejegyzések létrehozása különböző célcímmel rendelkező csomagok küldésével lehetséges. Amennyiben az átjáró kap egy csomagot, akkor a csomag küldője és címzettje alapján keresi a belső táblázatában lévő címfordítási szabályokat, ha a keresés sikertelen, akkor egy újabb bejegyzés létrehozására kerül sor.

Az általunk implementált szoftver segítségével végeztük el ezt a mérést is, a dinamikus bejegyzéseket különböző cél- és forráscímmel rendelkező csomagok küldésével hoztuk létre. A 8. ábra grafikonjának vízszintes tengelyén látható a bejegyzések száma, a függőleges tengelyén pedig az ehhez tartozó késleltetést tüntettük fel mikroszekundumban.



8. ábra A NAT-PT belső táblázatának növekedésével járó veszteségek

A 4. ábrán látható eredményekhez képest a 8. ábrán pontosan fordított eredményt kaptunk, ugyanis a bejegyzések számának növekedésével az IPv6-IPv4 irány hozta a rosszabb eredményeket.

4. Értékelés

Az előzőekben három IPv4-IPv6 átállási mechanizmus néhány teljesítményparaméterét vizsgáltuk. A mérési eredmények mutatják, hogy az általunk vizsgált implementációk összességében megállják helyüket.

Az alagutazás egy olyan fontos technika, amelyet már ma is széles körben használnak. Alkalmazása lehetővé teszi IPv6-os végpontok kommunikációját a már meglévő IPv4-es infrastruktúrán. A NAT-PT-ről annyi mondható el, hogy meglepő hatékonysággal működik, lehetővé teszi IPv4-es és IPv6-os számítógépek transzparenskapcsolatát. Egyszerűségéből és hatékonyságából következtethetünk arra, hogy fontos szereplője lesz a protokollváltásnak. E két technológiával szemben a SOCKS64 tudja csak biztosítani az IPv4-es és IPv6-os protokollok bármely kombinációjának alkalmazását, ennek azonban ára van: a kliens oldali szoftvercsere, jóval alacsonyabb hatékonyság, nagyobb hardverigény.

Az eddigi tapasztalataink alapján azt mondhatjuk, hogy nagy terhet venne magára az a szolgáltató, aki már ma élesben próbálkozna az egyes IPv4-IPv6 átállási technológiák alkalmazásával. Talán még éveket is várnunk kell arra, hogy ezeket a technológiákat a gyakorlatban – minden kétkedés nélkül – kipróbálhassuk, de aggodalomra nincs ok, hiszen a munka gőzerővel folyik.

Irodalom

1. R. Gilligan, E. Nordmark: Transition Mechanisms for IPv6 Hosts and Routers, RFC 2893, 2000.
2. H. Kitamura: A SOCKS-based IPv6/IPv4 Gateway Mechanism, RFC 3089, 2001.
3. G. Tsirtsis, P. Srisuresh: Network Address Translation – Protocol Translation (NAT-PT), RFC 2766, 2000.
4. E. Nordmark: Stateless IP/ICMP Translation Algorithm (SIIT), RFC 2765, 2000.

Automatikus tesztkiválasztás formális specifikáció alapján

ANDRISKA ZOLTÁN, BÁTORI GÁBOR, WU-HEN-CHANG ANTAL ÉS DR. CSOPAKI GYULA

Budapesti Műszaki és Gazdaságtudományi Egyetem
Távközlési és Telematikai Tanszék
Nagysebességű Hálózatok Laboratórium

Cikkünkben bemutatunk egy eljárást és egy ehhez kapcsolódó – általunk írt – szoftvert, mely segítségével egy formális protokollspecifikációból (pl. SDL – Specification and Description Language) mint kiindulási alapból, lehetséges tesztesetek automatikus válogatása. A tesztesetek kiválogatásához a mutációelemzés módszerét használjuk fel.

1. Bevezetés

A különböző gyártóktól származó távközlési szoftverekkel szemben támasztott egyik legfontosabb követelmény az együttműködés más gyártóktól származó rendszerekkel. Ezt általában szabványosítási szervezetek segítségével érik el, amelyek rendszerspecifikációkat hoznak létre. A gyártók ezek felhasználásával biztosítják a kompatibilitást. A távközlési termék (hardver/szoftver) fejlesztési folyamatának végén a végső és egyben legfontosabb lépés annak a vizsgálata, hogy a termék megfelel-e a specifikációban leírtaknak.

A távközlőrendszerekben problémát jelent gyors és megbízhatóan működő szoftverek létrehozása. A tesztelési folyamat igen időigényes és sok jól képzett szakembert igényel, ezért az eljárások egyszerűsítése a folyamat automatizálásával egy aktuális probléma. Ennek egy lehetséges megoldását mutatjuk be cikkünkben.

2. Protokollok formális leírása

A kommunikációs protokollok szabályok, melyek irányítják az elosztott rendszerek különböző komponensei közötti kommunikációt.

A protokollok fejlesztésének ciklusa több részből áll össze, a legfontosabbak a következők:

- a protokoll tervezése, specifikációja,
- a specifikáció logikai helyességének ellenőrzése (validálás),
- a megvalósítás (implementálás) – szoftver/hardver termék készítése,
- egy adott megvalósítás hibamentességének vizsgálata (konformancia, együttműködési stb. vizsgálatok).

Ezekhez a lépésekhez mindenképpen szükség van a protokollokat specifikáló dokumentumokra: a vázlat jellegű specifikációk a protokollrendszer különböző részeit tervező mérnökök könnyebb együttműködését szolgálják. Később, miután a teljes, végleges specifikáció elkészült, ellenőrizni kell annak logikai helyességét. E vizsgálat alapját is a rendszert leíró specifikáció képezi. A protokollokat megvalósító implementációkat a specifikációkból kiindulva hozzák létre, azok megfelelőségét is a specifikáció alapján vizsgálják.

A specifikációkat természetes nyelven fogalmazzák meg. Az elbeszélő jellegű informális leírás a könnyű érthetőség illúzióját nyújtja. Mindazonáltal a hosszú leírások tartalmazhatnak többértelmű részeket. További hátránya az informális specifikációnak, hogy a leírt rendszer logikai teljességének és helyességének ellenőrzése is nehéz.

Az informális leírások hiányosságainak pótlására sok különböző formális leíró technikát javasoltak a protokollok fejlesztéséhez: véges állapotgépeket, Petri-hálókat, magas szintű programozási nyelveket. A formális leíró technika eleget tesz az egyértelműség és az ellentmondás-mentesség követelményének, ezen túlmenően lehetővé teszi a gépi elemezhetőséget is.

A 80-as évek elején, az OSI (Open System Interconnection) szabványosításának kezdetén a CCITT (Comité Consultatif International Téléphonique et Télégraphique) és az ISO (International Organization for Standardization) speciális munkacsoportokat hozott létre, hogy tanulmányozzák a formális specifikációk felhasználását az OSI protokollok és szolgálatok definiálására. Ennek a munkának az eredményei az Estelle, a LOTOS (Language Of Temporal Ordering Specification) és az SDL formális leírónyelvek.

2.1. Az SDL (Specification and Description Language)

Az SDL [7] alapvetően leírónyelv, alkalmazások készítésére nem alkalmas. Lehetőséget biztosít azonban arra,

hogy a kiválasztott rendszert, vagy annak bármely funkcionális részét tetszőleges absztrakciós szinten prezentáljuk, mind felépítését, mind viselkedését illetően. Leírható vele tetszőleges valós idejű, interaktív, elosztott, rétegezett rendszer.

Matematikai alapja a kommunikáló kiterjesztett véges automata (CEFSM) modell, amely a rendszer működését gerjesztés-válasz módon határozza meg. A modellezett rendszer több egymással és a rendszerkörnyezettel kommunikáló állapotgépből áll, ahol a kommunikáció FIFO (First-In First-Out) elven működő csatornákon és jelutakon szállított jelekkel történik.

A teljes rendszer viselkedését az egyes állapotgépek viselkedésének összessége határozza meg. Formálisan az automaták viselkedését a viselkedésgráffal adhatjuk meg. Funkció szerint hierarchiába szervezve az egymással kommunikáló automatákat, a leírást a rendszer, blokk és processz hierarchiaszintek jellemzik. Egy folyamat maga egy véges állapotú automata.

2.2 Az MSC (Message Sequence Chart)

Az MSC-t, az ITU-T Z.120 ajánlását [6], elsősorban abból a célból fejlesztették, hogy megfelelő szabványos felületet nyújtson a rendszerek kommunikációs viselkedésének leírásához. Ezt a rendszerkomponensek közötti jeltovábbítás-sorozat formális leírásával valósítja meg. Az MSC használható valós idejű rendszerek, követelmények, interfészek, szimulációk és tesztek specifikálására. A nyelv fontos érdeme, hogy szabványosan kidolgozott szintaktikája és szemantikája egyértelmű rendszerleírást tesz lehetővé.

3. A formális leíráson alapuló tesztkiválasztó eljárás bemutatása

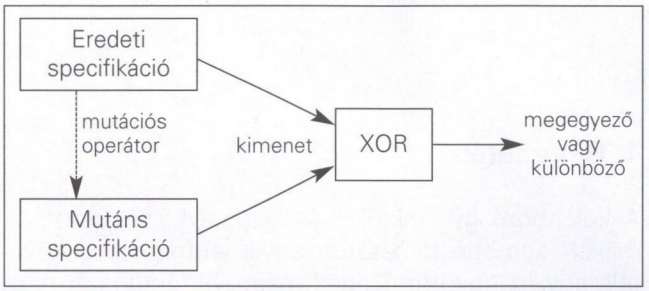
Az eljárás [1] egy rendszer (jelen esetben távközlési protokoll) egy véges méretű, de strukturálatlan és erősen redundáns tesztesetekből álló készlet felhasználásával, mutációelemzés segítségével hoz létre valamilyen szempontból optimális tesztkészleteket. Ezzel ellenőrizhető a formális specifikáció. Az egyes tesztesetek osztályozhatók, kiválasztható belőlük egy kisebb méretű, kevés tesztből álló készlet, amely azonban nagy valószínűséggel képes felderíteni a rendszer hibáit.

A tesztkiválasztó eljárás a mutációelemzésen alapul. A mutációelemzés egy fehér dobozos tesztmódszer, azaz szükséges a rendszer belső felépítésének és viselkedésének ismerete. A hagyományos mutációelemzés programkódban előforduló hibák felderítésére alkalmazható, most azonban a programok helyett specifikációkra fogjuk alkalmazni, az optimális tesztesetek kiválasztása céljából. Ezeket tesztek végrehajtásakor már fekete dobozos rendszereken – amelyeknek nem ismerjük a belső működését – is alkalmazhatjuk.

Egy mutációelemző rendszer mutációs operátorok egy halmazát definiálja, ahol minden operátor egy adott típusú atomi szintaktikai változtatásnak felel meg. Ezen operátorok használata két okból praktikus. Egyrészt lehetővé teszik a hibatípusok formális leírását, másrészt a mutációs operátorokkal az automatikus mutánsgenerálás is lehetségessé válik.

Egy mutációelemző rendszer három komponensből áll:

- eredeti rendszer,
- mutáns rendszer, ez szintaktikus változata az eredetinek, ami azt jelenti, hogy egy kis változtatásban különbözik attól. A mutánsok létrehozhatók úgy, hogy a mutációs operátorokat az eredeti rendszeren alkalmazzuk.
- Orákulum – egy személy, vagy legtöbb esetben egy program, ami az eredeti rendszert megkülönbözteti a mutánstól a rendszereknek a környezettel való interakciójuk alapján.



1. ábra Mutációelemző rendszer komponensei

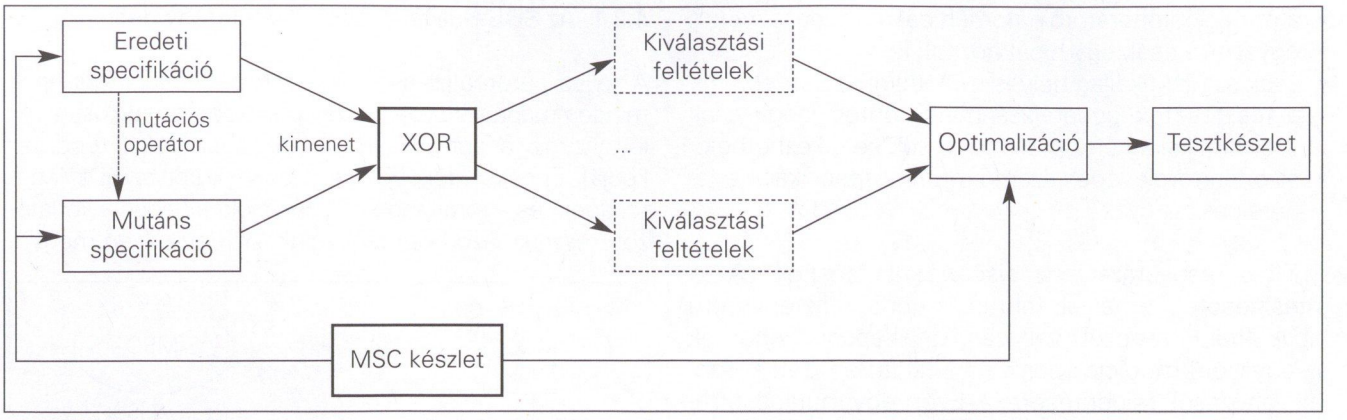
A tesztkiválasztó algoritmus feltételezi, hogy kiindulásképpen létezik egy nagyméretű, strukturálatlan tesztkészletünk és a vizsgált rendszer formális specifikációja. Az eljárás lépéseinek ismertetéséhez felhasználunk egy mátrixot, ami a tesztesetek mutáns felderítő képességéről tárol adatokat.

A kiválasztás során a rendszert a környezetből érkező bemenetekkel gerjesztjük és a környezetbe távozó kimeneteknél keresünk ellentmondásokat (2. ábra).

Az eljárás algoritmusa:

1. Az eredeti specifikációból létrehozunk egy mutáns leírást.
2. A tesztkészlet minden elemét végrehajtjuk az adott mutánson, és hibákat keresünk.
3. Ezt a folyamatot (1. és 2. lépés) minden mutánsra megismételjük.
4. Az eredményeket egy logikai elemekből álló mátrixban tároljuk, ahol egy sor egy mutánsnak, egy oszlop pedig egy kiinduló készletbeli tesztesetnek felel meg.
5. Végül kiválasztjuk az eredeti készletből az optimális tesztkészletet az eredményeket tároló mátrix alapján.

A mátrix elemei logikai típusúak, minden sora egy mutánsnak, minden oszlopa egy tesztesetnek felel meg. Ez a mátrix minden tesztesetre leírja, hogy az melyik mutánst volt képes megkülönböztetni az eredeti specifikációtól. Amennyiben egy teszteset képes felderíteni egy mutációt, akkor a mátrix megfelelő mező-



2. ábra Teszteseteket automatikusan kiválasztó rendszer felépítése

jét igazra állítjuk. Ha egy oszlopban csak hamis elemek vannak, akkor az azt jelenti, hogy a megfelelő MSC teszteset egyik mutánst sem volt képes detektálni, így a mátrixot egyszerűsíthetjük az adott oszlop törlésével. Ha egy sorban minden elem hamis értékű, ez vagy azt jelenti, hogy az adott mutáns ugyanúgy viselkedik, mint az eredeti rendszer, vagy pedig azt, hogy egyetlen olyan teszteset sem volt a kiinduló készletben, ami meg tudja különböztetni az eredetitől.

4. A szoftverrendszer általános felépítése

Az előző fejezetben bemutatott eljáráshoz egy Java nyelvű szoftvert [2, 3] hoztunk létre. Formális leírónyelvként az SDL-t, míg a tesztesetek reprezentálására az MSC-t használtuk fel.

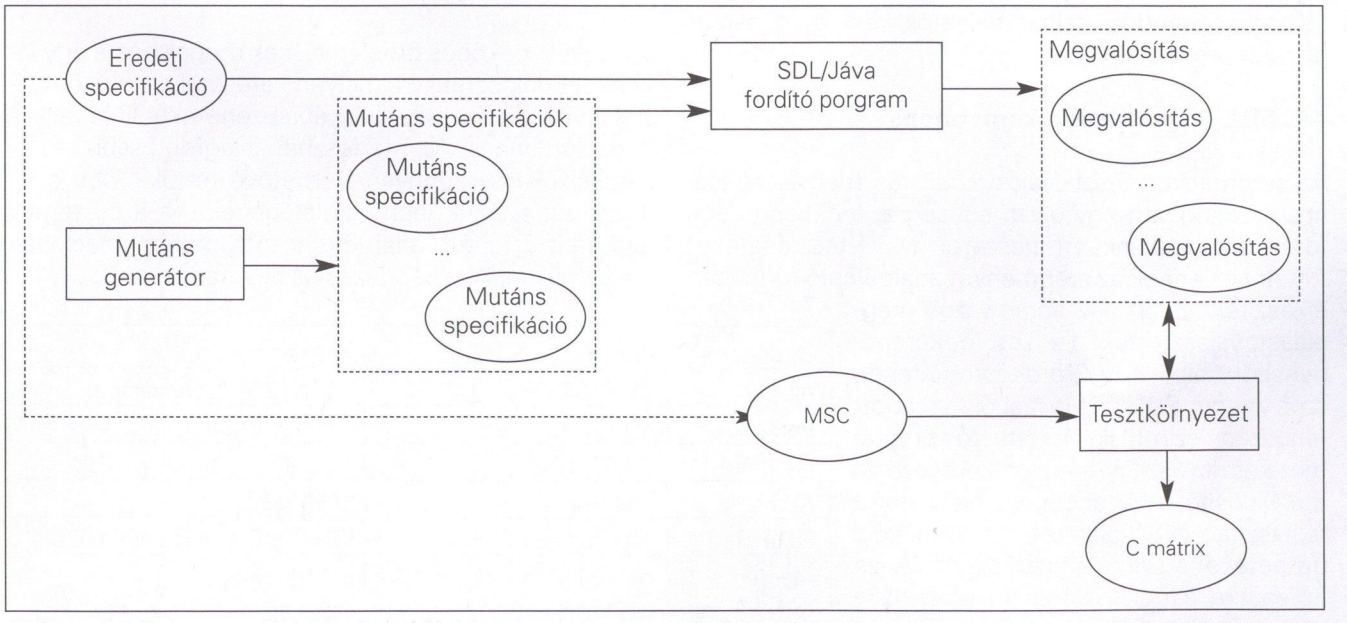
A bemutatott tesztkiválasztó eljárás első lépése egy mutáns specifikáció létrehozása az eredeti specifikáció alapján. Megvalósításunkban ezt a feladatot a 3. ábrán látható mutánsgenerátor végzi el, amely a mutációs operátorok alkalmazásával elkészíti a bemenetként adott specifikáció mutánsait.

Az eljárás második lépéséhez, nevezetesen az MSC tesztkészlet összes elemének lefuttatásához a mutáns rendszeren, mindenképpen szükséges egy eszköz, ami a specifikációból futtatható kódot állít elő. Jelen esetben ezt a feladatot az SDLÆJava fordítóprogram végzi. Ez a részegység a megkapott sok mutáns és az egyetlen eredeti specifikációból elkészíti a leírásoknak megfelelően viselkedő Java nyelvű forráskódot.

A forráskódokból ezek után már csak a végrehajtható tárgykódot kell létrehozni. Így a tesztkörnyezetnek rendelkezésére állnak a megvalósítások, amelyeken lefuttathatja a teszteseteket. A tesztkörnyezet hozza létre az eredményeket tároló mátrixot, amelyet felhasználva egy optimalizáló komponens kiválasztja a legfontosabb teszteseteket.

4.1. A mutánsgenerátor komponens

A mutánsgenerátor komponens az eljárás első lépését valósítja meg, azaz az SDL-leírásból elkészíti annak mutánsait. A mutációelemzés felhasználásához definiálni kell a mutációs operátorokat. Ezek meghatározásakor a következő szempontokat tartottuk szem előtt:



3. ábra Az eljárást megvalósító rendszer felépítése

- a mutációs operátorok atomi hibákat modelleznek,
- egyszerre csak egy hibát hoznak létre,
- csak szintaktikailag helyes mutánsok születnek,
- a tesztesetek generálása miatt fontos, hogy csak szemantikailag értelmes mutánsok keletkezhetnek,
- az operátorok véges számú mutáns specifikációt generálnak.

Mi a magyarázata az első két kikötésnek? Azok a tesztesetek, amelyek fel tudják deríteni az egyszerű hibák által létrehozott változásokat, képesek lehetnek az egyszerű mutációk sorozata által létrehozott komplex változások felderítésére is. Ezért egy mutáns létrehozásakor egyszerre csak egy operátort használunk fel, ami kizárólag egyszerű hibát hoz létre. Fontos hangsúlyoznunk, hogy a mutációs operátorok használata után is értelmes kódot kell kapnunk, hiszen a mutáns rendszeren később teszteseteket szeretnénk majd futtatni, ezért szükséges, hogy az operátor által megváltoztatott SDL-leírás szintaktikailag és szemantikailag is helyes legyen.

A mutációs operátorokat – a fenti alapelveket figyelembe véve – úgy határoztuk meg, hogy lehetőleg mindenféle specifikációs hiba előforduljon. Összesen 5 különböző típusú operátorosztályt kaptunk, amelyeket aszerint különböztettünk meg, hogy alkalmazásukkal a specifikáció melyik részét változtatják meg.

Az 1. táblázatban látható az operátorok működésére egy-egy példa.

Az operátorok között vannak olyanok, amik egyszerűen kitörölnek egy-egy részt a specifikációból (pl. az akciómódosító operátor a 4. sorban), és vannak olyanok is, amelyek megváltoztatják a kódot úgy, hogy kicserélnek egy elemet egy másikra.

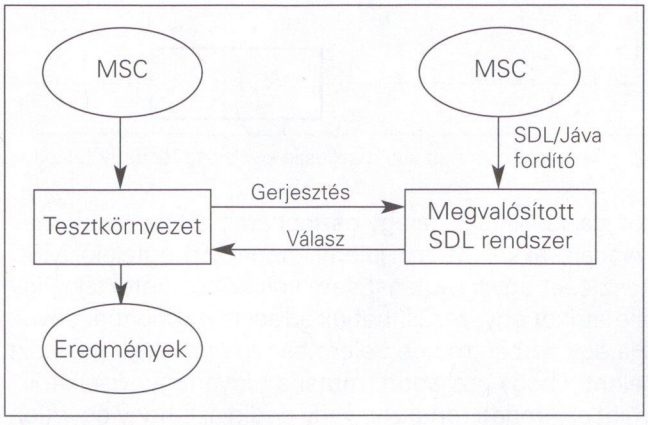
Mivel az elkészítendő mutáns generátor egy az SDL nyelvtani szabályainak szigorúan megfelelő leírásból néhány változtatás alkalmazásával ugyanolyan nyelvtanilag szabályos leírást készít, ezért ez az eszköz fordítóprogramnak is felfogható. Így – a generátor implementálását segítendő – felhasználhatók különböző parser generátor eszközök.

4.2. SDL→Java fordító komponens

Az automatikus tesztkiválasztó eljárás megvalósításához szükség van egy olyan egységre, ami képes egy formálisan specifikált rendszeren MSC teszteseteket futtatni és ennek az eredményét analizálható formában elkészíteni. Ezért a 4. ábrán vázolt megoldást választottuk. Ez két fő komponensből áll: egy fordítóprogramból, amely egy SDL/PR leírásból az adott rendszert szimuláló futtatható programot készít, és egy – a tesztkörnyezetet megvalósító – vezérlőegységből, ami a felhasznált MSC-nek megfelelően küld üzeneteket a szimulált rendszer felé, és az ezekre a gerjesztésekre adott válaszokat értékeli.

4.2.1. Az SDL→Java fordító követelmény definíciója

A teljes automatizálás érdekében a fordítóprogramnak minden manuális beavatkozás nélkül, automatikusan kell létrehoznia a specifikáció alapján a szimulált rendszer kódját. Ennek érdekében korlátoznunk kell az SDL lehetőségeit, és a fordítandó SDL leírásoknak ebbe a korlátozott részhalmazba kell tartozniuk. Az egyik ilyen megkötés, hogy elegendő állapotgép (pontosabban kommunikáló, kiterjesztett, véges állapotú, automata) specifikációkat fordítani, és az SDL által tartalmazott egyéb – ritkán használt – kiegészítésekkel nem kell foglalkozni.



4. ábra SDL→Java fordító szerepe

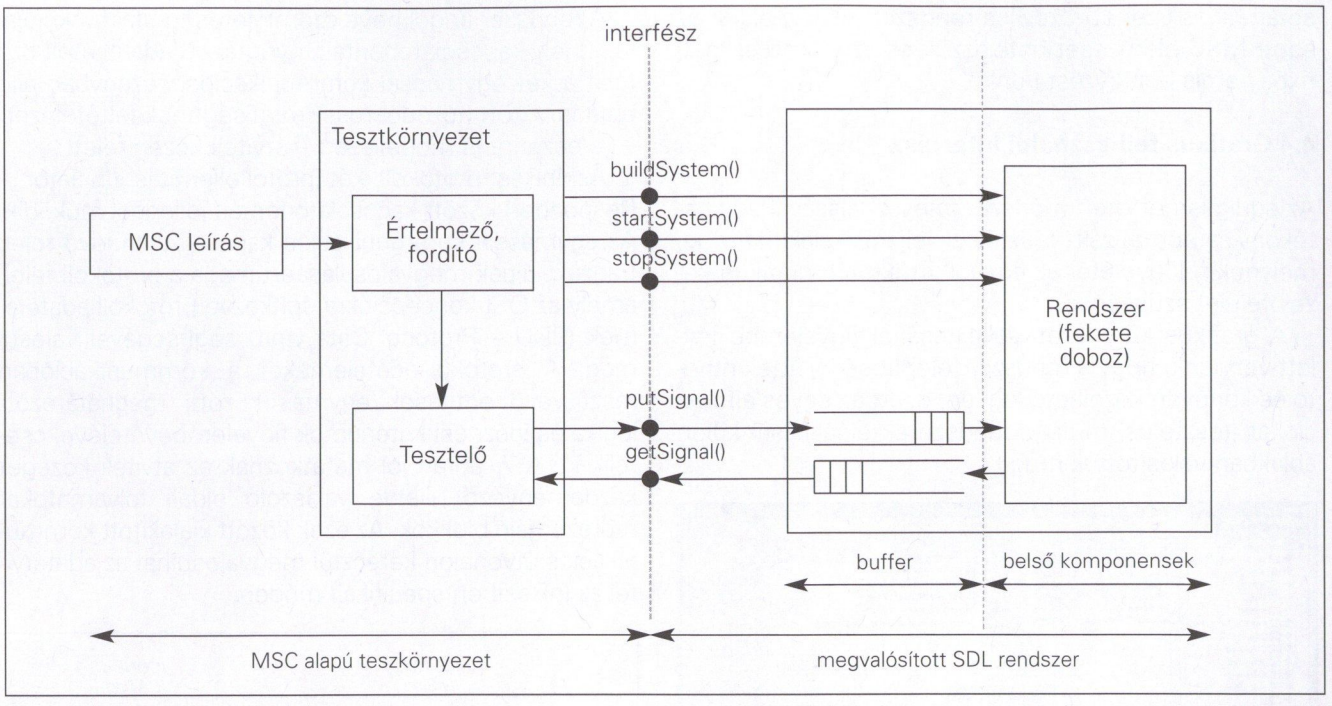
A fordítóprogram által generált rendszernek képesnek kell lennie az együttműködésre az MSC-leírásokat feldolgozó komponenssel. A két részegység között szükség van egy kétirányú kommunikációs csatornára az üzenetváltásokhoz, és egy egyirányú vezérlőcsatornára, amelyen a tesztkörnyezetet megvalósító MSC feldolgozóegység a vizsgált rendszer indítását, leállítását stb. is kontrollálhatja.

4.2.2. A tesztkörnyezet és a mutáns megvalósítás közötti kapcsolat

Az együttműködés problémájának megoldására egy interfészt dolgoztunk ki, melyen keresztül a tesztkörnyezet a vizsgált rendszer irányába üzeneteket küldhet, és fogadhat onnan. Az interfészben a legfontosabb vezérlőfunkciókat is könnyen elérhetővé tettük. Több ötlet közül azt találtuk legmegfelelőbbnek, ha a csatornák buffereit a fordító által létrehozott implementációban helyezzük el. Az elrendezés az 5. ábrán látható.

Operátorok	*Eredeti*	*Mutáns*
Állapot	NEXTSTATE wait;	NEXTSTATE connected;
Bemenet	INPUT ICONreq;	INPUT IDISreq;
Kimenet	OUTPUT CC;	OUTPUT DT (number, d);
Akció	Counter:=1;	/* Missing */
Feltétel	DECISION sdu!id = CC;	DECISION NOT(sdu!id=CC);
Mentés	SAVE IDATreq(d);	/* Missing */

1. táblázat Az SDL mutációs operátorai



5. ábra A tesztkörnyezet és az implementáció közötti együttműködés

A két oldal közti kapcsolat másik sarkalatos pontja, hogy kommunikációs csatornán egymásnak elküldött üzeneteket a két entitás fel tudja dolgozni. Az SDL-ben az üzeneteket nevükkel és az esetleges paramétereik típusával kell definiálni. Ennek alapján egy általános Java osztályt dolgoztunk ki, mellyel bármilyen paraméterezésű üzenet megvalósítható.

4.2.3 Az SDL→Java fordító megvalósítása

A létrehozás előtt célszerű minden SDL/PR szintaktikai elemhez egy annak funkcionalitását megvalósító Java kódmintát rendelni, hiszen miután a megfelelő leképezések megvannak, a fordítóprogram kódgeneráló részét ezekre a mintákra tudjuk építeni.

A leképezések elkészítésekor a [4, 5] forrásokra támaszkodtunk. Néhány helyen azonban az általuk javasolt kódmintákon módosítanunk kellett a meghatározott követelmények miatt. Például az általunk generált rendszernek az időzítőket nem kell megvalósítania, mivel az azok által generált üzeneteket a rendszer nem önmaga állítja elő, hanem kívülről, az őt vezérlő tesztkörnyezettől kapja meg. Ezzel a módszerrel több MSC teszteset lefuttatása esetén sok időt lehet megtakarítani.

Az SDL komponensei (jelek, processzek, csatornák, blokkok stb.) objektumalapúak, ezért érdemes megfeleltetni őket egy-egy Java osztállyal, melyek tartalmazzák a komponensre jellemző tulajdonságokat, eljárásokat. Ezeket az osztályokat aztán célszerű könyvtárba szervezni, hogy a forráskód bonyolultságát csökkentsük és az implementációk működésének megbízhatóságát növeljük a kód újrafelhasználása révén.

4.3 MSC elemző, tesztkörnyezet-komponens

A rendszer középpontjában a tesztvégrehajtó komponens áll. Ez az egység az előállított mutánsok megvalósításait és az MSC teszteseteket felhasználva állítja elő az eredménymátrixot. Ebből a mátrixból az optimálizáló egység létrehozza az optimális tesztkészletet.

Mivel az MSC szekvenciális felépítésű feltételes elágazások nélkül, így az elemzés alatti dinamikus végrehajtás is megfelelő megoldást ad. Dinamikus végrehajtás során az adott MSC-leírás egy sorának elemzése, és az annak megfelelő utasítás végrehajtása után döntünk az elemzés folytatásáról. Így ugyanaz az elemző használható különböző MSC fájlok esetén is. Ez lehetőséget ad arra, hogy az adott rendszert egy időben több egymástól különböző MSC-leírással teszteljük.

Egy sor elemzése után az elemzési információk alapján elküldjük a kívánt üzenetet, vagy várunk egy üzenet érkezésére, amellyel a feldolgozás során létrejött üzenetet összehasonlíthatjuk.

A szokásos üzeneteken kívül egy speciális üzenet is elküldésre kerülhet. Ez a jel az időzítők lejártakor keletkezik (timeout kulcsszó az MSC-leírásban). Az időzítőjel-paramétert nem hordozó névvel ellátott üzenet pontos kezelése a tesztelt rendszer feladata. Azzal, hogy az időzítők kezelését a tesztelő rutin végzi, jelentős idő nyerhető a tesztelési folyamatból, mivel ebben az esetben nem kell a rendszernek tétlenül várnia, hogy az időzítő lejárjon.

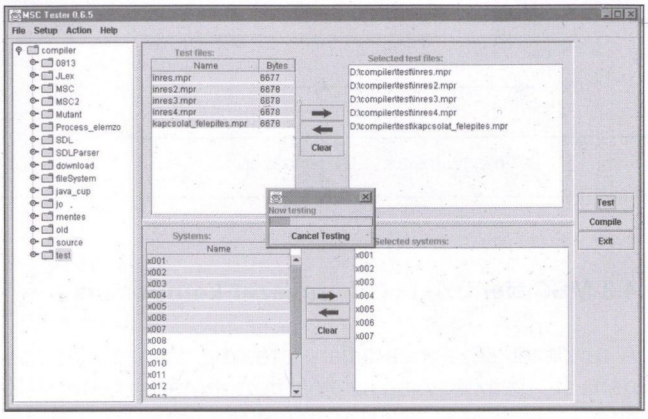
Két üzenetet akkor tekintünk azonosnak, ha a nevük és a paraméterlistájuk megegyezik. A fogadott üzeneteket az MSC-leírás alapján előállított tesztjellel hasonlítjuk össze. A tesztelést akkor tekintjük sikeres kimenetelűnek, ha az MSC-ben leírt minden üzenet helyes formában érkezik meg a tesztelőhöz. Ha a tesztelés

során hibás üzenet érkezik a rendszerünkhöz, akkor az adott MSC elemzését befejezzük és az elemzési mátrixba hamis bejegyzést írunk.

4.4 Grafikus felhasználói interfész

Az eddig ismertetett módszer felhasználásával egy hatékony automatizált tesztelési eljárás alakítható ki, melynek felügyeletéhez egy jól strukturált grafikus kezelőfelület szükséges.

A grafikus környezet kialakításánál figyelembe kellett vennünk, hogy a rendszer felépítése jól áttekinthető és könnyen kezelhető legyen, ezért az egyes alfunkciókat (tesztelés, mutáns rendszerek előállítása) külön ablakban valósítottuk meg.



6. ábra A grafikus felhasználói felület

A 6. ábrán a tesztkörnyezet főablaka látható. Ebben az ablakban lehet elvégezni a tesztesetek és a mutáns rendszerek kiválasztását. Miután a tesztek és a mutánsok kiválasztása megtörtént, csak a Test gombra kell kattintani, és a rendszer automatikusan elvégzi a tesztelési eljárást. A tesztelés eredményéről táblázatos formában kapunk tájékoztatást, ami nem más, mint az eredménymátrix grafikus reprezentációja. A Compile gombra kattintva egy új ablak jelenik meg, melyben a mutánsok generálása történik az SDL specifikáció kiválasztása után.

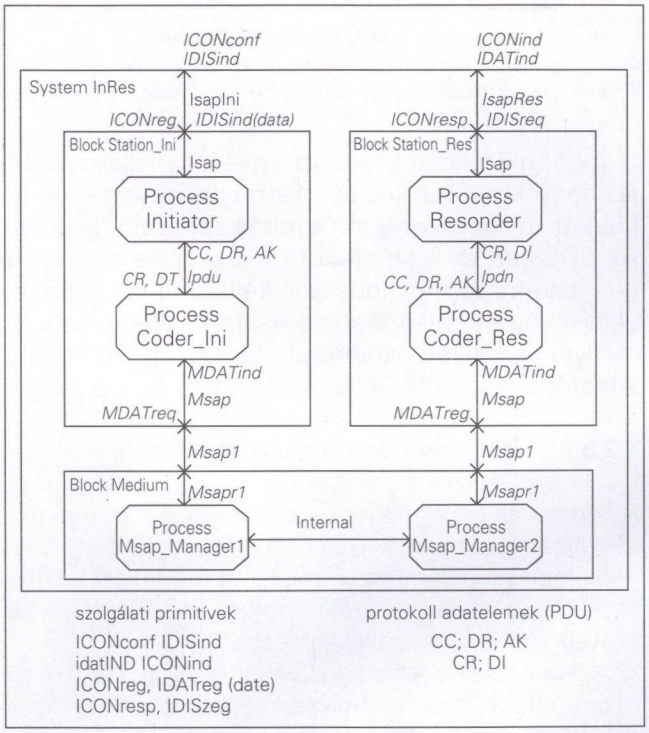
5. Próba az InRes protokollon, eredmények értékelése

5.1. Az InRes rendszer bemutatása

Az InRes (Initiator – kezdeményezŒ; Responder – válaszoló) rendszer bár nem valós protokoll, mégis előszeretettel veszik elő rétegezett kommunikációs rendszerek bemutatásához, hiszen tömör, kis méretének és világos struktúrájának köszönhetően az OSI (Open System Interconnection) modell koncepciója hatásosan szemléltethető. Tulajdonságai hasonlítanak a valós rendszerekéhez, az SDL specifikációjában az SDL szintaktikus elemek nagy része megtalálható.

A rendszer angol neve szemléletesen utal funkciójára, amely kapcsolatorientált, nyugtázott adatátvitelt biztosít a két egymással kommunikációs viszonyban álló entitás között húzódó, rossz minőségűnek feltételezett – és eszerint is modellezett – átviteli közeg felett.

Az InRes protokoll két protokollentitás (Initiator – Responder) között kapcsolatorientált jelleggel működik. Az egymással kommunikációs kapcsolatban lévő felek tranzakcióinak megvalósulásáért maga a protokoll felel, amely az OSI koncepciókra építkezve protokoll-adatelemek (PDU – Protocol Data Unit) segítségével valósul meg. A protokoll-adatelemeket a kommunikációban részt vevő entitások egymás között, meghatározott logika és időzítési kritériumok figyelembevételével cserélik ki. A 7. ábrán jól mutatkoznak az átviteli közeget kezdeményező, illetve válaszoló oldali folyamatokat reprezentáló blokkok. Az ezek között kialakított kommunikációs útvonalon keresztül megvalósulhat az adatátvitel az InRes-ben specifikált módon.



7. ábra InRes protokoll felépítése

5.2. A kísérlet eredményei

Összesen 87 MSC tesztesetet generáltunk a Telelogic Tau segítségével az InRes protokollhoz. A tesztkészletben szándékosan fordultak elő használhatatlan és jó felderítőképességgel rendelkező tesztesetek is, amelyeknek nagy részét véletlenszerűen állítottuk elő. Az operátorok felhasználásával 118 mutánst hoztunk létre automatikusan. Az eredeti tesztesetek közül 9 teszteset került az optimális készletbe a tesztkiválasztó eljárás eredményeként. Bekerültek olyan tesztesetek is, amelyekről már a rendszer ismerete alapján sejthető volt, hogy megfelelők lesznek.

Ahogy a 2. táblázatban látható, 20 mutánst nem sikerült megkülönböztetni az eredetitől az MSC teszte-

Mutáció operátorok	Előállított mutánsok száma	Felderített	Felderítési arány [%]
Állapot	24	14	58
Bemenet	28	28	100
Kimenet	36	33	92
Akció	21	15	71
Feltétel	8	7	87
Mentés	1	1	100

2. táblázat Mutációelemzés alkalmazása az InRes SDL leírásán

seteknek. A legrosszabb felderítési arányt az állapotot, míg a legjobbat a bemenetet módosító operátoroknál tapasztaltunk. A mutánsok közül – melyeket nem sikerült megkülönböztetni az eredeti specifikációtól – a legtöbb (12) az InRes legnagyobb processzében, az Initiatorben tér el az eredetitől. 6 mutánst a Responderben, továbbá 1-1-et a CoderResben és CoderIniben nem sikerült felismerni.

Voltak olyan mutánsok, melyeket csak egyetlen tesztteset derített fel, ezeket kritikus mutánsoknak nevezzük. Azokat a teszteseteket, amelyek ezeket felismerik, mindenképpen be kell tenni az eredményként kapott tesztkészletbe. Érdekes, hogy a kapott kilencből nyolc teszteset ilyen volt. A 8 kritikus mutáns közül kettőt feltétel-, kettőt kimenet-, hármat állapot- és egyet akciómódosító operátor alkalmazásával hoztuk létre. Bemenetmutáció nem produkált kritikus mutánst.

Ezek a tapasztalatok és a mutációfelderítési arány is arra mutat, hogy a bemenetmutáció nagyon durván módosított rendszert eredményez. Azaz olyan radikális változásokat hoz létre a rendszerben, hogy szinte bármelyik teszteset képes felismerni a különbséget. Ellenben az állapotmutáció olyan hibákat képez, amelyeket csak a tesztesetek kis része tud felderíteni, de 3 kritikus mutánst állított elő, többet, mint más típusú operátorok.

6. Összefoglalás

A hibaalapú tesztelés a specifikáció és egy hibamodell segítségével megpróbálja felfedni az implementáció hibáit. A jelen cikkben bemutatott eljárást és a definiált mutációs operátorokat kifejezetten az automatikus tesztkiválasztás megteremtésének érdekében dolgoztuk ki. Felhasználásukkal nincs szükség egy specifikáció vagy megvalósítás tipikus hibáinak ismeretére. Ehelyett az eljárás hoz létre hibás specifikációkat, amelyek megfelelő alapot szolgáltatnak a tesztesetek válogatásához.

A módszerre épülő eszközök nemcsak egyszerű protokollok, hanem igazi, komplex távközlési rendszerek SDL specifikációján is sikerrel használhatók. Az automatikusan létrehozott hatékony tesztkészletekkel a különböző implementációk gyorsan és könnyen megvizsgálhatók.

Irodalomjegyzék

1. Kovács Gábor, Pap Zoltán, dr. Csopaki Gyula: „Automatic test selection based on CEFSM specifications", 7th Symposium on Programming Languages and Software Tools, pages 200–212, Szeged, Hungary, 2001.
2. Le Viet Dung, Wu-Hen-Chang Antal: „Automatikus tesztkiválasztás SDL specifikáció alapján mutációs elemzéssel", TDK dolgozat, Budapesti Műszaki és Gazdaságtudományi Egyetem, 2001.
3. Andriska Zoltán, Bátori Gábor: „Tesztkörnyezet kialakítása MSC alapú mutációelemzőhöz", TDK dolgozat, Budapesti Műszaki és Gazdaságtudományi Egyetem, 2001.
4. Horváth G.: „Formális nyelven leírt protokollok Java nyelven történő megvalósítása", Diplomaterv, Budapesti Műszaki és Gazdaságtudományi Egyetem, 2000.
5. Kovács G.: „Az SS7-MTP2 protokoll megvalósítása Java nyelvi környezetben", Diplomaterv, Budapesti Műszaki és Gazdaságtudományi Egyetem, 2000.
6. ITU-T recommendation Z.120 Message Sequence Chart, 1996.
7. ITU-T recommendation Z.100 Specification and Description Language, 1992.

Hírek

Az Egyesült Királyság lakásainak 37%-ában van digitális televízió (jórészt műholdról), az Európai Unió tagállamaiban csak a lakások 16%-ában (Strategy Analysis).

Megnövekedett a használt telefonközpontok kereskedelme az Egyesült Királyságban. Hagyományos a használt alközpontok kereskedelme, de a gazdasági válság miatt közforgalmú központok is piacra kerültek. BT nemrég 10 db AXE 10 típusú központot adott el. Az eladók csődbe ment vagy súlyos gazdálkodási gondokkal küszködő vállalatok, köztük olyanok is, amelyek raktárkészletüket akarják csökkenteni, vagy gyorsan készpénzhez jutni. Kevés viszont a vevő. Előfordul, hogy használhatatlan berendezéseket az eredeti vételár 2%-áért lehet megvenni. Tőkeerős vállalatok külön csoportot állítottak fel az eladóközpontok felkutatására, mások büszkeségből nem vesznek másodkézből központot.

Forrás: Communication International (H. Gy.)

❖ ❖ ❖

Vezető távközlési cégek együttműködése az MMS-kompatibilitás érdekében

A CMG Wireless Data Solutions, a Comverse, az Ericsson, a Logica, a Motorola, a Nokia, a Siemens és a Sony Ericsson bejelentette a Harmadik Generációs Együttműködési Projekt (3GPP) és a Wap Szolgáltatás (MMS) Együttműködési Csoport megalapítását. A mobil adatátviteli szolgáltatás problémamentes végpont-végpont együttműködést tesz majd lehetővé a különböző gyártók MMS készülékei és szerverei között.

A csoport tevékenységi körébe tartozik az MMS-kompatibilitás tesztelése és koordinálása, és az ilyen tesztekkel kapcsolatos műszaki információk közzététele. Az eredményeket a 3GPP ismerteti. Ez segíti a kialakítást, valamint a termékek gyorsabb piaci bevezetését eredményezi.

A roved szöveges üzenet (SMS) továbbfejlesztése, multimédiás üzenetkezelési szolgáltatás jóval többet nyújt elődjénél. Segítségével szöveg, hang, grafika, kép, fotó, animáció és videofájl továbbítható mobilterminálok, illetve terminálok és tartalomszerverek között. Jelenleg az SMS a legfontosabb profitképező mobil adatátviteli szolgáltatás. Üzleti modelljének sikerére építve a multimédiás üzenetkezelésnek minden esélye megvan arra, hogy tömegpiaci mobil adatátviteli szolgáltatássá váljon, és kiterjessze a szolgáltatás határait.

❖ ❖ ❖

Az Ericsson jelentős előrelépést tett a „Wurekess Vullage" kezdeményezés terén azzal, hogy az UNPS képes együttműködni a SonyEricsson, a Motorola és a Nokia készülékeivel. A bemutató betekintést nyújtott a közvetlen mobil üzenetkezelési megoldások jövőjébe, és egyben megoldja a különböző üzenetkezelő és a jelenlétet figyelő szolgáltatások általános problémáját.

A Wireles sVillage kezdeményezés fontos, mivel segítségével újabb szolgáltatások kínálhatók. Integrálja mind az Ericsson, mind a Sony Ericsson készülékeibe.

❖ ❖ ❖

A NIIFI-ben működő Sun Enterprise 10 000 teljesítménybővítéssel csaknem kétszeresére nőtt, mellyel a világ legerősebb számítógépeinek jelenlegi listáján a 361. helyen szerepel.

Az oktatási, kutatási, tudományos és közgyűjteményi szféra számítási igényeit kiszolgáló berendezés, mely átadásakor 96 UltraSPARC processzort és 32 gigabyte memóriát tartalmazott, a felhasználói igények rohamos növekedésével immáron hosszabb ideje 100%-os kihasználtsággal működött. A színvonalas kiszolgálás érdekben szükségessé vált a StartFire bővítése, melynek eredményeképpen a magyar tudomány szolgálatába állított berendezés 128 UltraSPARC processzorral, 64 gigabyte memóriával, 1 Terabyte-ot meghaladó háttértár- és 3 Terabyte-nyi adatmentési kapacitással 100,5 Gflops maximális teljesítmény leadására képes, mellyel újra a legerősebb számítógép-konfiguráció, mely Magyarországon működik.

Időzített véges automata alapú vizsgálati módszer

GECSE ROLAND, SZABÓ J. ZOLTÁN, CSÖNDES TIBOR

Cikkünkben egy újfajta teljesítményvizsgálati módszert mutatunk be, melynek lényege, hogy a teszteléshez szükséges terhelést a felhasználók sokaságának emulációjával állítjuk elő. A felhasználók viselkedésének leírására egy új, időzített véges automata modellt dolgoztunk ki. Ismertetjük a módszer gyakorlati alkalmazásához kifejlesztett elosztott, konkurens teszt végrehajtó rendszert továbbá az eljárás kipróbálása során szerzett tapasztalatainkat is.

1. Bevezető

Mekkora forgalom kiszolgálására képes egy távközlési rendszer? Hogyan tűri a túlterhelést? Várhatóan mennyi ideig fog meghibásodás nélkül üzemelni? A teljesítménytesztelés ilyen és ezekhez hasonló kérdésekre keresi a választ. A tervezési fázisban végzett szimulációkkal ellentétben a teljesítménytesztelés a már elkészült rendszer működését ellenőrzi adott terhelés mellett. Legfőbb célja annak megállapítása, hogy a kész rendszer teljesíti-e a tervezés során kitűzött követelményeket. Például képes-e a központ ezer felhasználó párhuzamos kiszolgálására, és valóban alkalmas-e száz Erlang forgalom lebonyolítására.

Cikkünkben egy újfajta, időzített véges automata alapú teljesítmény teszt módszert mutatunk be. A módszer lényege, hogy a vizsgálatokhoz szükséges reális terhelést felhasználók sokaságának emulációjával állítjuk elő. A felhasználók viselkedését időzítővel kiterjesztett véges automatával modellezzük. Az új modell segítségével valószínűségi változóval időzített állapotátmenet is megvalósítható, így például az A felhasználót modellező automatát beállíthatjuk, hogy T idő elteltével hívja B-t. A T valószínűségi változót egy eloszlásfüggvénnyel specifikáljuk.

A modell szempontjából egyformának tekintett felhasználókat csoportokba rendezzük; az egy csoporton belüli felhasználók időzítőit azonos eloszlásokkal jellemezzük. A vizsgáló berendezés forgalom generátorai egy-egy ilyen csoport automatáit valósítják meg. A terhelés a csoportok egymás közti kommunikációjának eredményeként áll elő.

Az elméleti modellen túl bemutatjuk a módszer kipróbálásához kidolgozott elosztott, konkurens teljesítmény tesztelési architektúrát és a rendszer elemei közti kommunikációs protokollt.

Ismertetjük a forgalomgenerálási módszereket, majd részletesen bemutatjuk a kidolgozott automata modellt. Ezután a teljesítményteszt-architektúrát tekintjük át, végül röviden összefoglaljuk a módszer alkalmazása során tanultakat.

2. Forgalomgenerálás

A tesztelt rendszer (System Under Test, SUT) teljesítményének pontos meghatározásához azt környezetétől elkülönítve, fekete dobozként vizsgáljuk. A rendszert mesterséges terhelés alá helyezzük, minőségi paramétereit a kimenetein mért értékekből számítjuk. A megfelelő terhelés előállítása és mérése egyaránt a rendszer feladata.

A terhelés laboratóriumi körülmények közötti előállítására két alapvető módszer létezik. Az első a rendszer aggregált terhelés modellje szerint dolgozik, amelyet matematikailag, szimulációval vagy empirikus úton határoznak meg. A forgalmat a modell paraméterek (felhasználók száma, hívásintenzitás, hívásblokkolás stb.) behelyettesítése után kapott formulák alapján generáljuk. A második módszer ezzel szemben az egyes felhasználókat modellezi, és a kívánt forgalmat e modellek sokaságának „párhuzamos" működtetésével állítja elő. A két módszer közti alapvető különbség, hogy az első hálózati, a második felhasználói szinten működik. Vizsgálataink során mi az módszert alkalmazzuk.

E megközelítés előnye, hogy a felhasználók modellezése jóval egyszerűbb az aggregált forgalmat leíró pontos modell megalkotásánál. Sok esetben – például az internetnél – nem is létezik olyan modell, amely hálózati szinten megfelelő pontossággal írná le az összesített forgalmat. A módszer további előnye, hogy elegendően nagy számú felhasználó esetén a mérés kevésbé érzékeny a felhasználói modell pontosságára. Ezzel szemben aggregált esetben a pontatlanság a generált forgalommal arányosan nő. A felhasználónkénti

modellezésnél a jelzés- és adatforgalom felhasználónként is nyomon követhető. Ebből adódóan akár csatornánkénti beszédminőség vizsgálatot is végezhetünk.

Az alkalmazott modell hátránya, hogy megvalósítása az aggregált modell szerinti forgalomgenerálásnál jóval nagyobb számítási teljesítményt igényel, valamint működtetéséhez hatékony párhuzamos végrehajtást támogató futtató környezet szükséges.

3. Időzített véges automata modell

A felhasználók matematikai modelljéhez sztochasztikus, időzített véges automatát készítettünk.

1. Definíció:
Az időzített véges automata egy rendezett 7-es $\langle S, s_0, I, O, C, dens, T\rangle$, amelyben

- S az automata állapotainak véges halmaza;
- $s_0 \in S$ a kiindulási állapot;
- I a bemeneti szimbólumok véges halmaza;
- O a kimeneti szimbólumok véges halmaza;
- C az órák véges halmaza;
- $dens{:}C \rightarrow PB$ egy függvény, amely minden órához hozzárendel egy eloszlásfüggvényt; PB ezen eloszlásfüggvények halmaza; valamint
- $T \subseteq S \times (I \cup C) \times S \times (O \cup \{\varepsilon\}) \times 2^C$ az állapotátmenetek véges halmaza; ε az üres kimeneti szimbólum, amely azt jelzi, hogy az állapotátmenet során az automata kimenetén nem jelenik meg kimeneti szimbólum.

Az automata működése lényegében megegyezik a véges automatáéval (Finite State Machine, FSM [3]). A különbségek az órák megjelenéséből adódnak:

1. Az automata $s' \in S$ következő állapotát az $s \in S$ aktuális állapot és az $i \in I$ input szimbólum vagy egy $c \in C$ időzítő határozza meg: $S \times (I \cup C) \rightarrow S$;
2. az állapotátmenet eredménye az $o \in O \cup \{\varepsilon\}$ kimeneti szimbólum mellett az időmérők egy részének $X \in 2^C$ újraindítása is lehet: $S \times (I \cup C) \rightarrow (O \cup \{\varepsilon\}) \times 2^C$

A modell időzítői vagy inaktív állapotban várnak, vagy aktív állapotban számlálnak visszafelé. Kezdetben minden óra inaktív, lekérdezéskor τ (lejárt) értéket ad vissza. Indításkor az órák automatikusan kezdőértéket kapnak a hozzájuk rendelt eloszlásfüggvény szerint, és aktív állapotba kerülnek. A visszaszámlálás az időzítők lejártával megszakad, és az órák újra inaktív állapotba jutnak. A modell érdekessége, hogy az aktív órák is újraindíthatók. Az órák elindítása kizárólag állapotátmenet következményeként lehetséges. Az időzítők állapota, τ vagy $-\tau$, a bemeneti szimbólumokhoz hasonlóan használható állapotátmenetek triggerelésére. Az órák konkrét értékének lekérdezésére valamint az órák leállítására modellünk nem kínál lehetőséget.

Állapotátmenet tehát két alapvető esetben mehet végbe. Az $\langle s, i, s', o, \{c_1,c_2\}\rangle$ állapotátmenet például az i bemeneti szimbólum érkezésére aktiválódik, s állapotból s' állapotba visz át. A kimeneten az o szimbólum jelenik meg és a c_1,c_2 órák is újraindulnak a hozzájuk rendelt $dens(c_1)$, $dens(c_2)$ eloszlásfüggvényekből adódó kezdőértékekkel. Az $\langle s, c_1, s', \varepsilon, \{c_1\}\rangle$ állapotátmenet a c_1 óra lejártakor lép s-ből s'-be, a kimeneten semmi sem jelenik meg, mindössze az épp lejárt c_1 óra indul újra.

Az időt, mint bemeneti paramétert is számba vevő automaták irodalmát tekintve modellünk a Timed Automata [1] módosított változataként is felfogható, a következő eltéréssekkel:

1. az órák pillanatnyi állása számunkra nem hordoz információt;
2. az órák nem előre, hanem visszafelé haladnak, lepergésükkor megállnak;
3. az időzítők állása az állapotátmeneteknél sem játszik szerepet, így az óra korlátok bevezetése is szükségtelen; végül
4. az órákat tetszőleges eloszlás szerinti valószínűségi változóval lehet inicializálni.

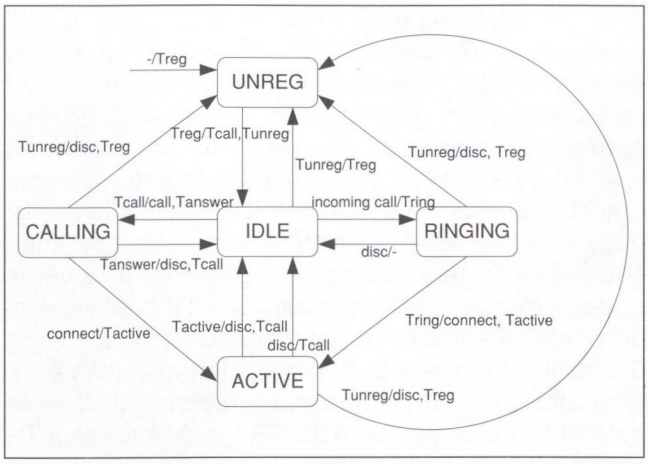
A bemutatott modell a Probabilistic Timed Automata [2] szűkítésének is tekinthető az 1, 2, 3 megszorításokkal, valamint azzal a korlátozással, hogy a nem-determinizmus kizárólag idő jellegű lehet. Azaz, automatánk bármely állapotában minden input szimbólum vagy timeout esemény legfeljebb egyszer szerepelhet.

4. Felhasználói modell

Az ismertetett modellt felhasználók magas szintű viselkedésének leírásához terveztük. Az 1. ábrán felvázolt példa automatája egy tipikus H.323 kliens [5] időzített véges automatáját tartalmazza. A H.323 felhasználó annyiban különbözik egy hagyományos telefonos felhasználótól, hogy egy Gatekeeper-nek nevezett központi szerverhez kell regisztrálnia, hogy hívást kezdeményezhessen vagy fogadhasson[1]. A modell működésének bemutatásához tekintsük az IDLE állapotot. Az állapot elhagyása háromféleképpen lehetséges:

1. A T_{unreg} óra lejártakor a felhasználó regisztrációja lejár, azaz nem kezdeményezhet vagy fogadhat újabb hívást. Az átmenet hatására inicializálódik és elindul a T_{reg} óra, amely a következő regisztrációig szükséges időt méri.
2. Az IDLE állapotba érkezéskor inicializált T_{call} óra lejártakor az automata új hívást kezdeményez (call), átmegy a CALLING állapotba és a T_{answer} időzítő elindításával meghatározza azt a maximális időtartamot, ameddig a hívott fél válaszára vár.

[1] Az automata feltételezi, hogy UNREG állapotban a T_{reg} óra lejártával a regisztráció, ill. egyéb állapotokban a T_{unreg} időzítő lejártával a kiregisztrálás implicite történik! E két esemény feltüntetése számos, a bemutatott módszer megértésének szempontjából jelentéktelen átvezető állapot felvételét tenné szükségessé.



1. ábra H.323 kliens egyszerűsített időzített véges automatája

3. Végül ha a T_{call} időzítő lejárta előtt beérkezik egy hívás (incomingCall), az automata a RINGING állapotba kerül és kisorsolja a T_{ring} időtartamot, amely azt modellezi, hogy mennyi idő elteltével fogja megválaszolni a hívást, feltéve, hogy addig a hívó fél nem bont.

A teljesítménytesztelésheż szükséges forgalmat ilyen felhasználói modellek sokaságának párhuzamos futtatásával állítjuk elő. Az automatákat csoportokba rendezzük, egy csoport azonos típusú időzítőihez azonos eloszlásfüggvényt rendelünk. A forgalmi paraméterek kalibrálása a csoportokhoz rendelt eloszlásfüggvények megfelelő megválasztásával történik. Tesztvégrehajtás során mindössze megfelelő számú automatát kell elindítanunk és a felhasználók sztochasztikus viselkedése automatikusan generálja a kívánt forgalmat.

5. Eloszlástípusok

Bár elméleti modellünk tetszőleges *PB*-beli eloszlásfüggvényre értelmezhető, gyakorlati alkalmazásoknál célszerű a használható eloszlástípusok korlátozása. Tesztrendszerünk a konstans, egyenletes, exponenciális és normális eloszlástípusokat támogatja. Az eloszlástípusok szintakszisát ABNF [6] formátumban ismertetjük.

```
distribution = constant / uniform / exponential
/ normal
```

Konstans eloszlásfüggvénnyel történő inicializálás esetén időzítőnk mindig a megadott értékkel indul. A konstans eloszlásfüggvény formátuma:

```
constant = „Constant" value
```

Egyenletes eloszlás esetén a valószínűségi változó értéke a két kötelező paraméter közé esik és az intervallumhoz tartozó összes érték előfordulási esélye azonos. Az egyenletes eloszlásfüggvény szintaktikája:

```
uniform = „Uniform {„minValue", „maxValue"}"
```

Exponenciális eloszlású valószínűségi változó választásakor a generált értékek a megadott várható érték (mean) körül szóródnak. A valószínűségi változó értékkészlete opcionálisan a \left[minValue,maxValue\right] intervallumra szűkíthető:

```
exponential = "Exponential {"
mean "," [minValue] "," [maxValue] "}"
```

Normális eloszlásfüggvénynél a várható érték mellett a szórást (stdDev) is kötelező megadni. Az exponenciális eloszláshoz hasonlóan itt is lehetőség az értékkészlet korlátozására:

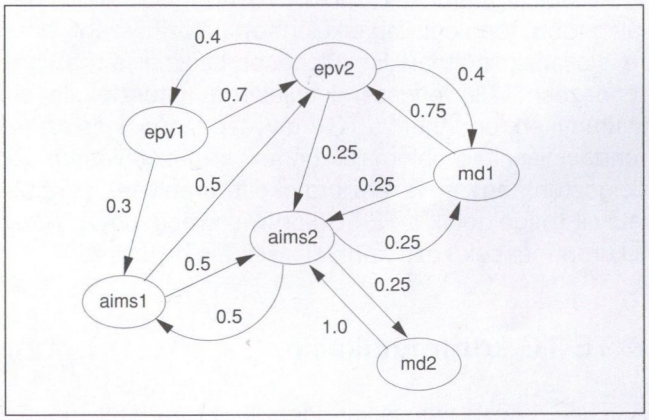
```
normal = "Normal {"
mean "," stdDev "," [minValue] "," [maxValue]
"}"
```

6. A hívott fél kiválasztása

A híváskezdeményezéskor a hívott fél kiválasztása párbeszéd-relációs gráf (Conversation Relation Graph, CRG) alapján történik. A CRG olyan irányított gráf, melynek pontjait a mérésben részt vevő csoportok alkotják. A gráf egy éle azt fejezi ki, hogy a kezdőpont csoportja választhat hívott felet a végpont csoportjából. Az élre írt szám a célcsoportba irányuló híváskezdeményezések valószínűségét mutatja. A gráf nem szükségszerűen összefüggő, de minden pont kimenő fokszáma legalább 1. Praktikus okokból a csoporton belüli hívások nem megengedettek, azaz a CRG nem tartalmazhat hurokéleket.

A fenti példa (2. ábra) CRG-ben az aims2 csoport 1/2 valószínűséggel az aims1, 1/4-1/4-del az md1, md2 csoport egy tagját hívja. Adott csoporton belül a hívott fél kiválasztása véletlenszerűen, egyenletes eloszlás szerint történik.

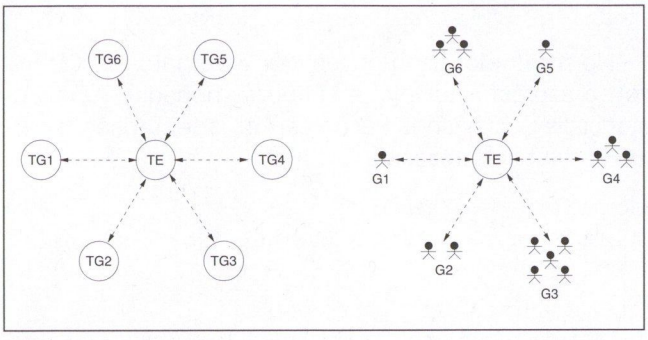
A hívott fél ilyetén kiválasztási módszerének előnye, hogy tetszőleges összeállítású forgalomelegy hozható létre, amelynek főként különböző típusú terminálokat modellező csoportok esetén van jelentősége. Továbbá, forgalmi szempontból jelentősen eltérő felhasználói csoportokból álló tesztelrendezés is könnyen kialakítható (pl. nagy forgalmat bonyolító kis csoport, call center jellegű felhasználó stb.).



2. ábra CRG példa

7. Teljesítményteszt-architektúra

A teljesítménytesztelésre kialakított elosztott architektúra egy központi teszt végrehajtóból (Test Executor, TE) és forgalom generátorokból (Traffic Generator, TG) épül fel. A 3. ábra szerinti elrendezés 6 TG-ből áll.



3. ábra TE-TG kapcsolat

A TE nem áll közvetlen kapcsolatban a tesztelt rendszerrel (SUT), feladata a tesztesetek végrehajtása és a TG-k vezérlése. Teszt végrehajtó prototípusunk TTCN-3 (Testing and Test Control Notation [4]) teszt leíró nyelven megírt tesztesetek futtatását támogatja. A TG-k vezérlésével kapcsolatos információkat is a tesztesetek tartalmazzák. A TE ezek alapján programozza fel, indítja el, állítja le a TG-ket, ill. értékeli ki a mért eredményeket. A TG-k feladata a kívánt forgalom előállítása a korábban bemutatott automatákkal jellemzett felhasználók csoportjainak modellezésével. A párhuzamosan működtetett automaták száma és viselkedési paraméterei a teszt elindítása előtt dinamikusan kerülnek beállításra. A [TE-TG] ábra TG-i rendre 1, 2, 4, 3, 1, 3 felhasználóból álló csoportot valósítanak meg.

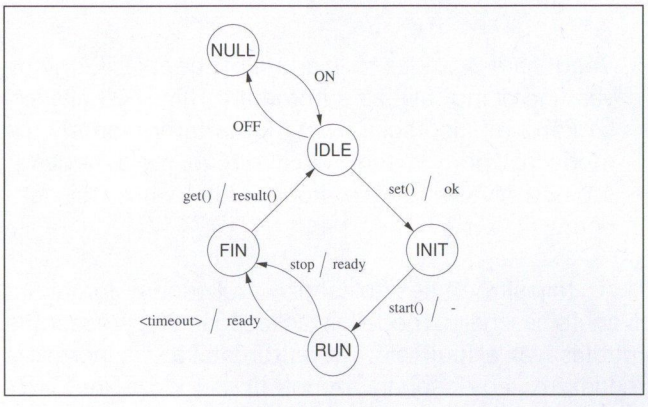
A felhasználók „megvalósítását" tekintve a TG-k két osztályba sorolhatók: emulált vagy vezérelt felhasználókkal működő TG. Míg az előbbi típus önmaga valósítja meg az emulált terminálok jelzés- és adatforgalmát, addig az utóbbi egy csoport valódi végberendezés vezérlését látja el (pl. egy mobil telefonkészülék billentyűit nyomogatja). Ebből kifolyólag egy csoporton belül csak azonos típusú (vagy emulált, vagy vezérelt) felhasználók lehetnek. Egy teszt elrendezésben természetesen mindkét típusú TG tetszőleges számban szerepelhet.

Az elosztott architektúra előnye, hogy a TG-k egyidejűleg több, topológiailag elkülönített hálózati szegmensre is csatlakozhatnak. Ez különösen hasznos a több berendezésből álló rendszerek teljesítmény tesztelésénél. Hátrányként említhető a TG-k távvezérlésének és szinkronizációjának problémája. Ennek kiküszöbölésére kidolgoztunk egy egyszerű protokollt, amelynek használatával megoldódik a TE-TG kommunikáció ugyanakkor a komponensek közti szinkronizációt is biztosítja.

8. TE-TG kommunikáció

A TE-TG közti kapcsolat-orientált, pont-pont típusú kommunikáció (3. ábra szaggatott vonalai) a kliens-szerver modell szerint működik. Maga a protokoll TCP-re (Transmission Control Protocol) épül; igény szerint dedikált hálózaton is haladhat – kiküszöbölendő a mért rendszer forgalmi viszonyainak befolyásolását. A kliens szerepét a TE tölti be, a TG-k szerverekként szolgálnak.

A TG-k működése az 4. ábrán követhető nyomon. Egy TG a bekapcsolásakor (ON) IDLE állapotban várja, hogy a TE kapcsolatot létesítsen vele. A kapcsolat létrejöttével a TG vezérlését teljes egészében a TE végzi. Egy-egy teszt eset végrehajtásakor a TE beállítja, elindítja, leállítja a tesztben résztvevő TG-ket, majd végül összesíti a mért adatokat. A teszt lefutása után a TE lekapcsolódik a TG-kről, melyek újrainicializálódnak és további feladatra várnak. A TE-TG kapcsolat csak a TG kikapcsolásakor szakad meg.



4. ábra TG állapotgép

Első lépésként a TE átküldi a generálandó forgalom paramétereit tartalmazó set üzenetet. Ennek hatására a TG felkészül a forgalomgenerálásra – háttérben létrehozza az automaták feladatát megvalósító folyamatokat vagy inicializálja a vezérelt terminálokat –, majd az ok primitívvel jelezi a TE-nek, hogy felkészült a forgalom generálásra (INIT állapotba kerül). Az inicializálás „handshake" funkciója szinkronizációs lehetőséget biztosít a különböző TG-k eltérő felállási idejének kompenzációjára. A forgalom generálás elindítása előtt a TE minden TG visszajelzését megvárja.

A megfelelő időpontban a TE a start üzenettel utasítja a TG-ket a már inicializált állapotgépek elindítására. A TG RUN állapotba kerül és mindaddig futtatja állapotgépeit, míg a TE a stop üzenettel a teszt végét nem jelzi, vagy, míg a start üzenet opcionális paraméterében kapott idő le nem telik. Ekkor a TG az összes állapotgépét leállítja (szétkapcsolja az aktív hívásokat, stb.), majd a forgalomgenerálás végeztét a ready üzenettel jelezve FIN állapotba megy át. A ready primitív, az ok-hoz hasonlóan szinkronizációs lehetőséget biztosít a TE-nek a lassabb TG-k bevárására. Következő lépésben a TE a get üzenet elküldésével kérdezi le a TG által mért értékeket, amelyeket a TG egy result üzenet formájában továbbít. A mérési eredmények alapján a TE kiszámítja a ténylegesen generált forgalmat. Ellenőrzi, hogy az valóban eleget tett a teszt céljában megfogalmazott feltételeknek, hogy a TG megfelelt-e az elvárásoknak. Eközben, a result üzenet elküldését követően, a TG fel-

szabadítja az összes lefoglalt erőforrását, visszamegy IDLE állapotba és újabb feladatra vár.

Típus	Funkció
string	csoport név
distribution	regisztráció intenzitás
distribution	regisztráció időtartama
distribution	hívásintenzitás
float	CRG él súly*
string	CRG él végpont (csoport) név*
distribution	hívástartási idő
distribution	csengetési idő
distribution	válaszidő
string	kiindulási hely azonosító#
distribution	mozgási intenzitás#

1. táblázat A set üzenet mezői<set>
* a CRG adott csoportból induló összes élére
# csak mobil terminálokra vonatkozik)

A teszt eredményét befolyásolandó, lehetőség van adott háttérforgalom felett történő hívás időzítési vagy minőségi paraméterek mérésére is. Ez esetben a TG-ktől kapott aggregált statisztikák jellemzően nem befolyásolják a teszt ítéletet.

Fontosabb üzenetek

A set kommunikációs primitív segítségével történik a TG-k távoli programozása. A TE ebben az üzenetben adja meg a TG által emulálandó csoport azonosítóját, a CRG TG-re vonatkozó részfáját és a felhasználói modell eloszlás paramétereit. Ez utóbbi határozza meg a generált forgalmat. Az üzenetet úgy dolgoztuk ki, hogy az előző fejezetben ismertetett felhasználói modell minden paraméterét tartalmazza, ezáltal bármilyen típusú terminállal rendelkező felhasználót le tudjon írni. Formátuma az 1. táblázaton található.

Az alábbi példában szereplő set primitív a következő információkat tartalmazza. A címzett TG a myGroup csoport felhasználóit fogja emulálni/vezérelni. A csoport tagjai 5 s várható értékű, 3 s szórású normális eloszlás szerint fognak regisztrálni. Egy regisztráció időtartama pontosan 300 másodperc. A terminálok regisztrálásukat követő 3-8 másodperc múlva kezdeményeznek hívást, amennyiben időközben nem érkezik be hívás. A hívott fél 1 valószínűséggel a calledGroup csoport tagjai közül kerül ki. A hívástartási idő az emulált felhasználók részéről exponenciális eloszlást követ 30 másodperc várható értékkel, de mindenen esetben legalább 10 ill. legfeljebb 50 másodperc múltán hívásbontást kezdeményez. Eközben természetesen a túloldal is bonthatja a kapcsolatot, amely a vártnál rövidebb tartási időt eredményez. Kimenő híváskor legfeljebb 10 másodpercig vár a hívott fél válaszára, míg bejövő hívásokat pontosan 4 másodperc elteltével fogadja. A myGroup felhasználói helyhez kötött terminálok, így a mozgással kapcsolatos opcionális paramétercsoport el van hagyva.

```
set(myGroup {Normal {5 3 - -} Constant 300} {Uni-
form {3 8} {{1 calledGroup}}
Exponential {30 10 50} Constant 10 Constant 4} -)
```

A TE a start üzenettel jelzi a TG-nek, hogy indítsa el az automatákat, és ezáltal kezdje meg a forgalom generálást. A start üzenet egyetlen opcionális paraméterrel rendelkezik, amely a végrehajtás időtartamát határozza meg. Ez idő elteltével a TG automatikusan leállítja az automatákat. Amennyiben a paraméter hiányzik, úgy a forgalomgenerálás határozatlan ideig tart. A leállást ekkor a TE explicit stop üzenete jelzi.

```
start ( [ float -- teszt időtartama ] )
```

Minden TG rendelkezik néhány számlálóval és órával. A számlálók bizonyos események előfordulásának számát mérik az adott csoportra vonatkozóan. A teszt eredményeként csoportonkénti összesített értékeket kapunk, melyek alapján számíthatók a generált forgalom paraméterei. A mért eredmények a result üzenetben jutnak el a TE-hez. Összesen 17 mennyiséget mérünk, melyek rendre: sikeres/sikertelen regisztrációk száma, sikeres/sikertelen regisztrációk száma, sikeres/foglalt/megválaszolatlan/egyéb sikertelen kimenő híváskezdeményezések száma, sikeres/foglalt/megválaszolatlan/egyéb sikertelen beérkező hívások száma, saját hívásbontások száma, túloldali hívásbontások száma, helyváltoztatások száma, aktív időtartam, erőforrásfoglalási időtartam.

Példa:

```
result(- - - - 171 7 10 - 185 - 11 0 180
176 0 32187 33868)
```

ahol a számok jelentése: TG felhasználói a teszt során 171 sikeres (megválaszolt) hívást bonyolítottak. A hívott fél 7 esetben jelzett foglaltat, 10-szer nem válaszolt, egyéb hibás híváskezdeményezés nem történt. Eközben a csoport felhasználói 185 bejövő hívásra válaszoltak. Az esetlegesen aktív állapotban beérkező hívások számáról nincs információnk. Ugyanakkor 11 esetben a hívó fél nem várta meg a hívott fél válaszát. 180 aktív hívást a TG felhasználói, 176-ot pedig más csoportok automatái kapcsoltak szét. Helyváltoztatás nem történt. A TG felhasználói 32187 másodpercet töltöttek beszélgetés állapotban. Ez a csengetési és csengési időkkel kiegészítve 33868 s.

A result üzenetben kapott statisztikák alapján lehetséges a generált forgalom kiértékelése is.

9. Eloszlás paraméterek kiszámítása

Modellünk legérzékenyebb pontja az automata időzítőihez rendelt eloszlásfüggvények meghatározása: minden TG-hez eloszlásfüggvényeket kell rendelni. Ezeket úgy kell kiszámítani, hogy a végleges modell várhatóan az előírt terhelést valósítsa meg. Az eloszlásfüggvények

rossz megválasztása olyan szituációt is eredményezhet, hogy egyetlen hívás felépítése sem lehetséges.

Az eloszlásfüggvények meghatározását a felhasználók csoportosításával és a CRG megtervezésével kezdjük. Tekintsük a [fsm] ábra szerinti felhasználó modellt, valamint a [CRG] ábra CRG-jét és számítsuk ki az aims2 csoport időzítőinek eloszlásfüggvényeit! A generálandó forgalom paraméterei: átlagos (sikeres és sikertelen hívásokra egyaránt) hívástartási idő = 100 s, forgalom intenzitás = 150 mErlang/felhasználó, GoS = 100\%. A számítások egyszerűsítése érdekében feltételezzük, hogy a csoport bemenő és kimenő hívásainak intenzitása azonos!

Tegyük fel, hogy a T_{ring} csengetési idő konstans 9 másodperc, a T_{answer} válaszidő pedig egyenletes eloszlást követ a 0-10 másodperc intervallumon. Ekkor a hívás felépülésének valószínűsége $P_{(answer)}$ =0.9. Az átlagos $E(T_{active})$ hívástartási idő pedig az alábbi egyenletből számítható:

$$P(answer)\,E(T_{answer} + T_{active}|answer) + (1-P(answer))\,T_{ring} = 100s$$

Ebből

$$E(T_{answer} + T_{active}|answer = E(T_{answer}|answer) + E(T_{active}|answer) = 110.11s$$

Az egyenletes eloszlás miatt

$$E(T_{answer}|answer) = \frac{9}{2}\,s = 4.5s$$

Innen

$$E(T_{active}|answer) = 105.61s$$

Feltéve, hogy a túloldal hasonló eloszlást követ, a beállítandó átlagos tartási idő a számított érték kétszerese, azaz 211.22 s. A hívások kezdete közt eltelt idő átlagosan:

$$E(T_{ci}) = \frac{100\,s}{0.15Erlang} = 666.67s$$

Mivel a T_{call} időzítő a hívás befejezésekor indul újra (lásd 1. ábra), a számított értéket egy hívás várható idejével csökkenteni kell. Továbbá, a beérkező hívások intenzitása a kimenő hívásokéval megegyezik, tehát az időzítőt a számított érték kétszeresére kell beállítani:

$$E(T_{call}) = 2\cdot(E(T_{ci}) - 100s) = 1133.33s$$

Az eredmény eloszlásfüggvények tehát: csengetési idő: Constant 9, válaszidő: `Uniform{0, 10}`, hívástartási idő: `Exponential(211.22, -, -)`, hívásgyakoriság: `Exponential{1133.33, -, -}`.

10. A mért mennyiségek kiértékelése

Mivel egy-egy mérés időtartama mindig korlátos, a mérés végén minden esetben ellenőrizni kell, sikerült-e elérni a kívánt rendszerterhelést. Ez a kérdés a mért paraméterek alapján válaszolható meg. Az időzítőkhöz képest túl rövid futási idő nagymértékű eltérést okozhat a számított értékektől.

Vegyük például a 8. fejezet minta result üzenetében kapott adatokat. Az emulált csoport a [CRG] ábra aims2 csoportja, amely a előző fejezet paramétereivel rendelkezik. A csoport 64 felhasználóból áll, a teszt futási ideje egy óra, azaz 3600 s volt. A forgalom intenzitása:

$$A = \frac{100\,s}{0.15Erlang} \approx 0.147Erlang$$

Tehát a célként kitűzött felhasználónkénti 150 mErlang hívásintenzitást jó közelítéssel teljesítettük.

Bár a TG-k csak aggregált statisztikát szolgáltatnak, felhasználónkénti adatok kinyerésére is kínálkozik lehetőség. Létrehozhatunk egyetlen terminálból álló csoportokat vagy utólagosan feldolgozhatjuk a futás során készült naplót. Az előbbi módszer hátránya, hogy lelassítja a teszt rendszert, az utóbbi csak többletmunkát jelent.

11. Összefoglalás

Bemutattunk egy új modellre, az időzített véges automatára épülő magas szintű teljesítménytesztelési módszert. Az elméleti modell ismertetésén túl, betekintést nyújtottunk az általunk megvalósított elosztott, konkurens teszt végrehajtó rendszer prototípusának működésébe is. A teljesség igénye nélkül, egyszerű példákkal illusztráltuk a módszer felhasználásának részleteit; a csoportokhoz rendelt eloszlások kiválasztását és a teszt eredményeinek feldolgozását.

Az ismertetett teljesítménytesztelési módszert sikerrel alkalmaztuk egy prototípus H.323 alapú internet telefonrendszer vizsgálata során. Tesztjeink során emulált (PC-s) és távvezérelt (ISDN, GSM) végberendezéseket megvalósító TG-ket is felhasználtunk.

References

1. R. Alur, D. L. Dill: "A Theory of Timed Automata", Theoretical Computer Science, 126:183–235, 1994.
2. M. Kwiatkowska, G. Norman, R. Segala, J. Sproston: "Verifying Quantitative Properties of Continuous Probabilistic Timed Automata", CONCUR'2000, 2000.
3. G. V. Bochmann: "Finite State Description of Communication Protocols", Computer Networks, Vol.2, No.4/5, pp.361-372, 1978.
4. Methods for Testing and Specification (MTS); Testing and Test Control Notation; Part 1: TTCN-3 Core Language, ETSI ES 201 873-1.
5. Packet-based multimedia communications systems, ITU-T Recommendation H.323 (11/00).

6D. Crocker (Ed.), P. Overell: "Augmented BNF for syntax specifications: A BNF", RFC-2234, November 1997.

Hálózatméretezési kérdések játékelméleti megközelítése

FÜZESI PÉTER, VIDÁCS ATTILA

Budapesti Műszaki és Gazdaságtudományi Egyetem
Távközlési és Telematikai Tanszék
Nagysebességű Hálózatok Laboratórium

Ahogy a garantált minőséget nyújtó szolgáltatások fontosabbá válnak, a hagyományos hálózatméretezési módszerek egyre kevésbé használhatók olyan környezetben, ahol a szolgáltatások ára is szempont. Jól definiált szolgáltatási osztályok esetén a felhasználókat egymással versengő játékosokként leírva a játékelmélet eszköz lehet a hálózat viselkedésének leírására. Alkalmazva a nem kooperatív játékok szabályait, ahol a játékosok csak saját céljaik elérésében érdekeltek, a Nash-egyensúly alkalmas lehet a rendszer „nyugvópontjainak" leírására. A cikkben szimulációs eredmények segítségével mutatjuk be a hagyományos és a játékelméleti hálózatméretezési megközelítés közötti különbségeket. Emellett kritériumokat adunk olyan kapacitásméretezési esetekhez, ahol a felhasználók minőség- és költségérzékenyek.

1. Bevezető

A garantált minőségű szolgáltatások (Quality of Service – QoS) egyre fontosabbá válnak az interneten. A valós idejű (pl. multimédia-) szolgáltatások új kívánalmakat támasztottak a hálózattal szemben [9]. A késleltetés, a csomagvesztési arány fontos paraméterekké váltak. Korlátokat adva ezekre a paraméterekre, különböző szolgáltatási osztályokat definiálhatunk.

Az internet általában csupán kapcsolatot biztosít, nincs garancia a szolgáltatás minőségére, függetlenül attól, hogy a felhasználó milyen szolgáltatást kezdeményez. Ez egyenes következménye az egyes csomópontokon működő „legjobb szándék" elvének, amely minden egyes adatcsomagot azonos módon szolgál ki.

A mai számlázási módszerek az interneten legnagyobbrészt átalánydíj-alapúak. A minőségi garanciák megjelenése várhatóan változtat ezen a helyzeten, hiszen a szolgáltatások területén megjelenik a „garancia" mint új érték.

Érdemes megvizsgálni, hogy felhasználói szemszögből milyen változásokat jelent a minőségi garanciák megjelenése. Nyilvánvaló, hogy a jobb szolgáltatásminőség javítja a felhasználók által érzékelt hasznosságot, míg egy adott szint alatt a minőség elfogadhatatlan. Emellett feltételezhető, hogy a jobb minőség egyben drágább is. Ez szintén befolyásolja a felhasználókat, aminek következménye, hogy igyekeznek a legolcsóbb, még kielégítő minőséget nyújtó szolgáltatási osztályt választani. Általánosságban a hálózatméretezés során figyelembe vett teljesítményparaméterek (mint a csomagvesztés, késleltetés) csupán részhalmazát jelentik azoknak a paramétereknek, amelyeket valójában vizsgálni kell. Emellett az összesített hasznosság, amelyet a felhasználók tapasztalnak a hálózat működése során, lényeges mérték a hálózat viselkedésének szempontjából.

A szolgáltatási osztályok megkülönböztetésére a szabványok két megközelítést ajánlanak. Az „integrált szolgáltatások" (Integrated Services – IntServ) [10] architektúra az egyes adatfolyamokat külön kezelve biztosít szolgáltatásminőséget. Az IntServ skálázhatósági problémái miatt azonban a „megkülönböztetett szolgáltatások" (Differentiated Services – DiffServ) [2] architektúra a közeljövőben ígéretesebb megoldásnak tűnik. Ez az architektúra csupán a hálózati tartományt határoló útvonalválasztókon (edge router) különbözteti meg az egyes adatfolyamokat, ahol az egyes csomagokat megjelölve különböző szolgáltatási osztályokba osztja azokat. A hálózaton belüli útvonalválasztók (core router) ezek után az egyes szolgáltatási osztályokba eső aggregált folyamokat különböztetik csak meg, így javítva az architektúra skálázhatóságát.

A játékelmélet és hálózatméretezés szempontjából a felhasználók játékosoknak is tekinthetők, akik a hálózati erőforrásokért versengenek. A játékosok nem kooperatív játékot játszanak, a résztvevők céljaik elérése érdekében egyedi döntéseket hoznak [5]. Azokat az állapotokat, ahol valamennyi játékos olyan helyzetbe jutott, ahol az érzékelt hasznosságon egyedi döntések révén nem javíthat, Nash egyensúlyi helyzeteknek nevezzük [3, 1]. Így a Nash-egyensúly alkalmas különböző helyzetek dinamikus viselkedésének jellemzésére.

Hagyományosan a hálózati erőforrások méretezése során statisztikai és teljesítményt leíró paramétereket szokás figyelembe venni: például egy szolgáltatási osztályhoz rendelt kapacitás méretezése a várható igények és az előírt blokkolási valószínűség függvénye lehet. A játékelmélet segítségével a hálózatméretezés kérdése tovább finomítható, hiszen így egyúttal a felhasználók által érzékelt összes hasznosság is felhasználható a szolgáltatási osztályok kapacitásainak méretezése során.

A cikk két kérdést elemez. A 3. fejezetben szimuláció segítségével vizsgáljuk a kialakult Nash-egyensúly tulajdonságait, ahogy a szolgáltatási osztályokhoz rendelt kapacitások változnak. Ebben az esetben a hagyományos hálózatméretezés és a Nash-egyensúly vizsgálatán alapuló módszerek közötti különbséget szeretnénk hangsúlyozni.

A 4. fejezetben módszereket adunk a szolgáltatási osztályok kapacitásainak méretezésére abban az esetben, amikor a felhasználók által érzékelt hasznosság csupán a tapasztalt QoS paraméterektől függ, és abban az esetben is, mikor azt a költségek is befolyásolják. Megmutatjuk, hogy a két módszer nem vezet feltétlenül ugyanarra az eredményre.

2. Játékelméleti modell

Több publikáció foglalkozott már a hálózat viselkedésének játékelméleti megközelítésével nem kooperatív környezetben. Az útvonalválasztó algoritmusokat [6] és a különböző erőforrás-foglalási módszereket szintén megvizsgálták ebből a szempontból is [5].

A felhasználói viselkedés is vizsgálható játékelméleti eszközökkel, amikor különböző szolgáltatási osztályok állnak rendelkezésre [7, 8]. Ebben a modellben minden szolgáltatási osztályhoz tartozik egy QoS függvény, ami meghatározza az osztályban kialakuló QoS paramétereket az átmenő forgalom függvényében. Emellett minden felhasználónak van egy sávszélességigénye és egy adott küszöbértéke, amely a még kielégítő QoS paramétereket jelenti. Alapesetben a felhasználó stratégiáját a rendelkezésre álló szolgáltatási osztályok (felhasználói szempontból) minél hatékonyabb kihasználása jelenti, vagyis hogy a sávszélességigényt mely szolgáltatási osztályok és milyen arányban szolgálják ki. A modell egyszerűsítéseként az igények oszthatatlannak is tekinthetők. Ebben az esetben a felhasználó csupán egy szolgáltatási osztályt választhat, vagyis a stratégia a megfelelő osztály kiválasztását jelenti. A megfelelő, más szóval kielégítő szolgáltatási osztály ebben a modellben az az osztály, ahol a felhasználó igényével együtt kialakult forgalommennyiséggel számolva a tapasztalt szolgáltatásminőség még elfogadható (pl. a késleltetés kisebb mint 20 ms). Általánosan igaz, hogy az egyszerűsített modell megfelel a tipikus erőforrás-foglalási eseményeknek. A továbbiakban feltételezzük az igények oszthatatlanságát.

A felhasználók aktuális foglalásait a „ erőforrás-foglalási mátrix a következőképpen definiálja:

$$\Lambda = [\lambda_{ij}], \quad i \in [1,m], \quad j \in [1,n], \tag{1}$$

Vagyis m felhasználó és n szolgáltatási osztály mellett λ_{ij} jelenti az i. felhasználó által a j. szolgáltatási osztályba küldött forgalom mennyiségét, míg λ_i az i. felhasználó összesített sávszélességigényét adja meg:

$$\lambda_i = \sum_{j=1}^{n} \lambda_{ij}. \tag{2}$$

Az igények oszthatatlansága esetén egy felhasználó csak egy szolgáltatási osztályt választhat, vagyis igaz, hogy $\lambda_i=\lambda_{ij}$, ahol j jelöli a választott szolgáltatási osztályt.

Az összes, j. osztályba eső forgalom mennyiségét a következő összefüggés adja:

$$q_j = \sum_{i=1}^{m} \lambda_{ij}. \tag{3}$$

A $c_j(q_j)$ függvény határozza meg a j. szolgáltatásban kialakult minőségi paramétereket q_j terhelési szint mellett. Ebben a modellben skalár értéket használunk a QoS jellemzésére. Legyen ez a skalár érték a csomagvesztési valószínűség (emellett természetesen választhatnánk akár késleltetést vagy késleltetésingadozást is). Feltételezhetjük, hogy $c_j(q_j)$ a terhelés függvényében monoton növő és a nagyobb csomagvesztési valószínűség rosszabb szolgáltatásminőséget jelent.

Minden felhasználó rendelkezik egy Θ_i küszöbértékkel, amely a felhasználó által még tolerálható maximális csomagvesztési valószínűséget jelöli. Az alkalmazott modell szerint U_{ij} hasznossági függvény rendelhető a j. szolgáltatási osztályt választó i. felhasználóhoz úgy, hogy annak értéke nulla legyen, ha a tapasztalt minőség nem kielégítő, és λ_{ij}, ha igen:

$$U_{ij} = \begin{cases} 0, & if \theta_i \langle c_j(q_j) \\ \lambda_{ij} & if \theta_i \langle c_j(q_j) \end{cases}, \tag{4}$$

így az érzékelt hasznosság a felhasználók számára elfogadható minőségben kiszolgált sávszélesség mennyisége. Mivel a felhasználók célja minél nagyobb hasznosság elérése, megpróbálnak olyan szolgáltatási osztályt találni, amely kielégíti minőségi követelményeiket, így maximalizálva a hasznossági függvény értékét:

$$U_i^{\max} = \max_j (U_{ij}). \tag{5}$$

A Nash-egyensúly elérésére létezik egy egyszerű algoritmus abban az esetben, amikor a $c_j(q_j)$ függvény skalár értéket határoz meg [7]. Miután minden játékost rendezünk a QoS küszöbértékük alapján (6) szerint, a szabály az, hogy mindig a „legérzéketlenebb" játékos lép először, vagyis az, aki a leginkább tolerálja a minőségromlást.

$$\forall i,j \in [1,m], \quad i \langle j: \quad \Theta_i \geq \Theta_j. \tag{6}$$

Figyelembe véve a második szabályt, amely szerint egy felhasználó nem lép akkor, ha nem tudja azzal javítani az általa érzékelt hasznosságot, az algoritmus biztosítja, hogy az érzékenyebb felhasználók semmilyen lépéssel sem tudják „elrontani" a náluk érzéketlenebb felhasználók hasznosságérzetét. Bizonyított, hogy ez az algoritmus azonnal Nash-egyensúlyhoz vezet.

A Nash egyensúlyi helyzetek jól jellemezhetők egy olyan értékkel, amely az összes, a felhasználó által érzékelt hasznosság alapján számolható ki. Ez az érték

a „Nash-egyensúly szintjét" írja le, jelölése LNE (Level of Nash Equilibria):

$$\mathrm{LNE} = \sum_{i=1}^{m} U_i^{\max} \qquad \text{Nash egyensúly esetén.} \tag{7}$$

Feltételezzük a szolgáltatási osztályok egy sorrendjét, ahol az első osztály a legrosszabb, az utolsó a legjobb szolgáltatásminőséget nyújtja. Ha c_j^{min} jelöli a j. osztályban elérhető QoS értéket terhelés nélküli esetben, vagyis

$$c_j^{\min} = c_j(0) \tag{8}$$

n szolgáltatási osztály esetén, akkor

$$\forall i,j \in [1,n], \quad i\langle j: \; c_i^{\min} \rangle c_j^{\min}. \tag{9}$$

Feltéve, hogy a jobb minőséget nyújtó szolgáltatási osztály drágább, mindegyik felhasználó célja, hogy a legolcsóbb, még kielégítő szolgáltatási osztályt válassza, vagyis azt, ahol a nyújtott szolgáltatásminőségi paraméter, pl. csomagvesztési valószínűség a felhasználói küszöbérték alatt van. Ha Θ_i jelöli az i. felhasználó QoS igényét, jelölje S_i a felhasználóhoz rendelhető kielégítő szolgáltatások halmazát:

$$S_i = \left\{ j: \quad j \in [1,n], \quad c_i^{\min} \le \Theta_i \right\}. \tag{10}$$

Ezek után jelentse P_i az i. felhasználó preferált szolgáltatási osztályának indexét, amelyik a legolcsóbb, még kielégítő szolgáltatási osztály:

$$P_i = \min_{j \in S_i} j. \tag{11}$$

Vezessük be a j. osztályhoz rendelhető F_j elsődleges felhasználói halmazt, amelybe azon felhasználók sorszámai tartoznak, akik a j. osztályt választják preferáltként. Így m felhasználót feltételezve

$$F_j = \left\{ i : P_i = j, \quad i \in [1,m] \right\}. \tag{12}$$

Definiáljuk a D_j j. osztályba eső elsődleges sávszélességigényt, mint azt az összesített igényt, amely a j. osztályba esik azáltal, hogy az F_j halmazba eső felhasználók ezt választják preferáltként. Ha C_j jelöli a j. osztály kapacitását, és λ_i jelenti az i. felhasználó sávszélességigényét, a definíció:

$$D_j = \sum_{i \in F_j} \lambda_i. \tag{13}$$

Az LNE fogalmát mint a hálózatméretezés fontos paraméterét használva, és a felhasználók költségérzékenységét figyelembe véve fontos megjegyzéseket tehetünk.

3. Szimulációk

A szolgáltatási osztályokhoz rendelt kapacitásváltozás függvényében a felhasználók által érzékelt hasznosság szimulációk segítségével vizsgálható. A szimulációk esetén is egyetlen QoS paramétert veszünk figyelembe. Így minden felhasználó jellemezhető a hozzá rendelhető λ_i QoS küszöbértékkel és Θ_i sávszélességigénnyel. A felhasználók által érzékelt hasznosságot a szimulációk során is (4) határozza meg.

A korábban bevezetett $c_j(q_j)$ függvényt a szimulációk során a következőképpen definiáltuk. Ha a j. szolgáltatási osztályba eső forgalom kisebb, mint az allokált C_j kapacitás, a nyújtott minőségi paraméter a c_j^{min} érték. Ha az osztály túl van terhelve ($D_j > C_j$), a paraméter „elég nagy", vagyis minden felhasználó által elfogadhatatlan (pl. csomagvesztési valószínűséget feltételezve közel van egyhez).

A szimulációk során három algoritmust vizsgáltunk.

A1 – Egyensúly túlterhelés nélkül. Ebben az esetben az első (minőségre leginkább érzéketlen) felhasználó kiválasztja a legolcsóbb, megfelelő (vagyis preferált) osztályt a c_j^{min} értékek figyelembevételével. A többi felhasználó a sorrendet követve úgy keres megfelelő szolgáltatási osztályt, hogy a saját sávszélességigényével együtt is megfelelő legyen még az osztályban keletkező szolgáltatásminőség, vagyis

$$i = 2, 3, ..., n \qquad \Theta_i \ge c_i \left(\sum_{k=1}^{i} \lambda_{kj} \right). \tag{14}$$

Ha egy felhasználó nem talál megfelelő szolgáltatási osztályt, nem küld forgalmat a hálózatba, elkerülve így a hálózat felesleges túlterhelését.

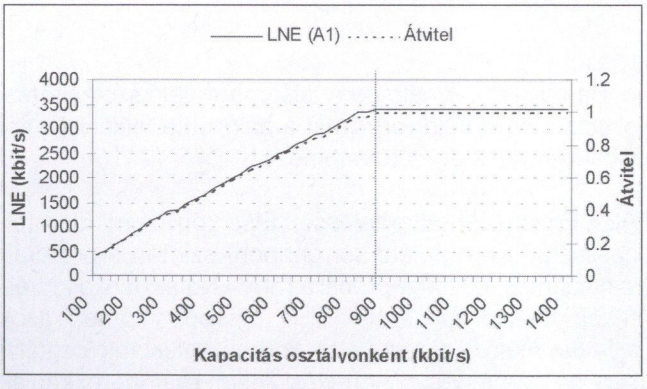
A2 – Egyensúly túlterheléssel. Első körben minden felhasználó (a megfelelő sorrendben) szintén a preferált szolgáltatási osztályt próbálja meg kiválasztani a c_j^{min} értékek figyelembevételével. A második körben azok a felhasználók, akik az első körben választott osztályban az esetleges túlterhelés miatt már nem kapják meg a kívánt szolgáltatásminőséget, megpróbálnak javítani a helyzetükön, ha lehetséges, azzal, hogy másik szolgáltatási osztályt keresnek a sávszélességigényükhöz. Ezt attól függetlenül megteszik, hogy a jobb minőséget biztosító osztály egyben drágább is, vagyis ebben az esetben a felhasználók költségérzékenysége kevésbé játszik szerepet. Ha ez nem sikerül, a hálózatba küldött forgalom marad az eredetileg választott szolgáltatási osztályban.

A3 – Egyensúly túlterheléssel és visszalépésekkel. Ez az algoritmus megegyezik A2-vel azzal a különbséggel, hogy ebben az esetben a felhasználók költségérzékenysége is számít, vagyis feltételezzük, hogy a jobb minőséget nyújtó szolgáltatási osztály drágább. Ha lehetséges, minden felhasználó igyekszik a saját prefe-

rált szolgáltatási osztályát választani, még akkor is, ha ez egy kevésbé jó, de még elfogadható minőséget biztosító osztályba történő visszalépést is jelent. Ez az algoritmus nem köti meg a lehetséges lépések számát, vagyis valójában egy eljárást azonosít, ami nem feltétlenül éri el, csupán megközelíti az egyensúlyi állapotot.

A szimulációs modellben öt szolgáltatási osztályt definiáltunk, különböző QoS paraméterekkel (csomagvesztési arányt meghatározva, 0,1%, 0,05%, 0,01%, 0,001% és 0,0001% értékekkel). Az erőforrásokért versengő felhasználók három csoportba oszthatók érzékenységük szerint, a csoportokban szereplő felhasználókhoz rendelt küszöbértékek egyenletes eloszlású változók a következő halmazokból véve: [0, 0,01%], [0, 0,1%], [0, 0,5%]. Minden felhasználóhoz tartozik egy λ_i sávszélességigény, ami egyenletes eloszlású valószínűségi változó a [10 kbit/s, 100 kbit/s] halmazból. A szimuláció során 60 felhasználó játszott, összesen körülbelül 3400 kbit/s sávszélességigényt generálva. A szolgáltatási osztályokhoz rendelt kapacitás minden szimulációs lépésben változott, és az így kialakult konfigurációkban határoztuk meg az LNE értékét külön-külön. A szimulációk során a szolgáltatási osztályokhoz azonos kapacitásértéket rendeltünk, amelyet 25 kbit/s lépésekkel növeltünk 100 kbit/s és 1400 kbit/s között. Egy szimulációs lépés tehát egy konkrét helyzetet írt le, amelyben az elérhető összes hasznosság a három különböző algoritmus segítségével volt vizsgálható.

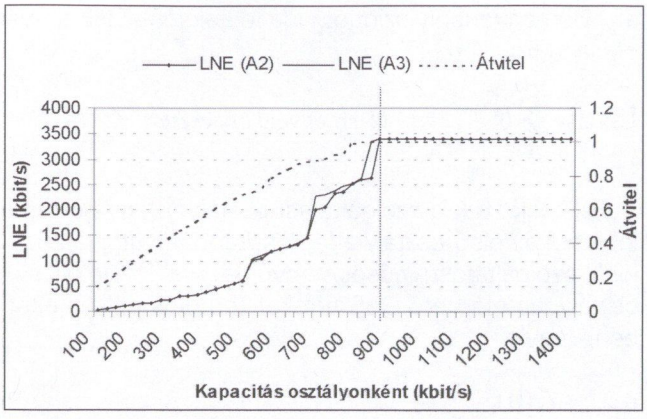
3.1. Eredmények



1. ábra LNE értékei A1 esetben

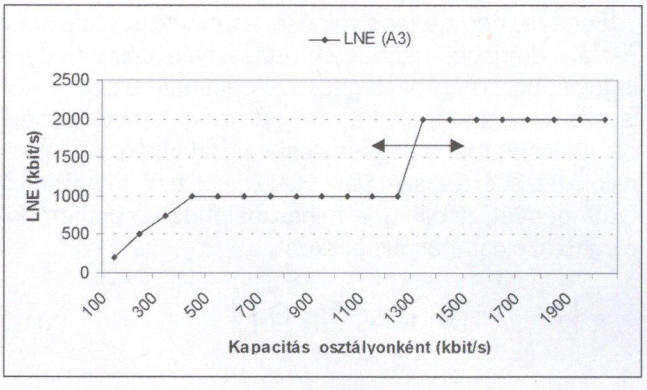
Az 1. ábrán LNE értéke látszik növekvő kapacitások mellett. Mivel A1 esetben a túlterhelés nincs megengedve, az LNE közel lineárisan nő, ahogy egyre több felhasználói igény kerül kiszolgálásra. Az ábra az összes, az igényelt minőségi szinten kiszolgált sávszélesség arányát is mutatja az összes sávszélességigényhez képest (átvitel), a hálózati osztályokban keletkező csomagvesztési arányok szerint. A teljes átvitel és a felhasználók által érzékelt hasznosság azonos helyen érik el maximális értéküket.

A 2. ábra szintén LNE értékét mutatja, de az A2 és A3 algoritmusok eredményeként. A két eredmény között nincs jelentős különbség, ám mindkét görbe az A1 esetben tapasztalt lineáris görbe alatt fut. A jelenség oka,



2. ábra LNE értékei A2 és A3 esetben

hogy ebben az esetben a hálózat túlterhelését megengedjük, a görbe kis kapacitások esetén lassan nő.



3. ábra Egy szélsőséges eset

A 3. ábra egy szélsőséges esetet mutat az A3 algoritmus használata mellett. Ebben az esetben két felhasználói csoport vett részt a „versenyben", az előbbiekben bemutatott szolgáltatási osztályok mellett. Az első csoportba érzéketlen, a másodikba érzékeny felhasználók tartoztak. Az a pont, ahol az LNE eléri maximumát, közvetlenül állítható annak az osztálynak a kapacitásával, amelyik a legjobb szolgáltatási osztályt nyújtja.

4. A hálózatméretezési probléma

Különböző hálózatméretezési stratégiákat célszerű követni abban az esetben, ha csupán a QoS igények kielégítése a cél, vagy a felhasználók költségérzékenysége is számít. Nyilvánvaló kritérium mindkét esetben, hogy az összes kapacitás ne legyen kisebb, mint az összes igény:

$$\sum_{j=1}^{n} C_j \geq \sum_{i=1}^{m} \lambda_i = \sum_{j=1}^{n} D_j \,. \tag{15}$$

4.1. QoS igények kielégítése

Ahhoz, hogy minden felhasználó QoS igénye ki legyen elégítve, elég egy egyszerű, rekurzív módszert követni

a kapacitások méretezésekor. Első lépésben ki kell számolni a legjobb minőséget nyújtó szolgáltatási osztály kapacitását. Ez egyszerű, hiszen azon felhasználók igényei, akiknek ez, vagyis az n. osztály a preferált osztály, csak itt szolgálhatók ki. Így

$$C_n \geq D_n. \tag{16}$$

Azon felhasználók esetében, akiknek a preferált osztálya az $(n-1)$-ik osztály, elég, ha csak annyit kötünk ki, hogy az n. vagy az $(n-1)$-ik szolgáltatási osztályba essenek az igényeik. Így, általános formulát adva C_j méretezéséhez, kapjuk:

$$\forall k \in [1,n]: \quad \sum_{j=k}^{n} C_j \geq \sum_{j=k} D_j. \tag{17}$$

4.2. QoS igények és költségérzékenység

Szigorúbb, de egyszerűbb szabályokat alkalmazva elérhető, hogy minden felhasználó a preferált szolgáltatási osztályba küldhesse forgalmát:

$$\forall j \in [1,n]: \quad C_j \geq D_j. \tag{18}$$

Definiálva a C_j kapacitásokat úgy, hogy (18) igaz legyen, minden egyes felhasználó külön-külön a legolcsóbb, még megfelelő szolgáltatási osztályban kap helyet. Így biztosítható, hogy a felhasználók minőségi követelményei és költségbeli elvárásai egyszerre legyenek kielégítve.

Természetesen mindkét módszer kielégíti az eredeti (15) feltételt. A leglényegesebb különbség a két módszer között az, hogy az első, csupán QoS igényeket figyelembe vevő módszer megengedi, hogy az i. osztályhoz rendelt kapacitást növeljük a j. szolgáltatási osztály kapacitásának rovására, ha i nagyobb, mint j.

5. Összegzés

Nem kooperatív felhasználók esetén a Nash-egyensúly hasznos eszköz a felhasználók által érzékelhető összes hasznosság leírására minőségi garanciákat nyújtó hálózatokban. A Nash-egyensúly szintje (LNE) adott konfigurációkban határozza meg az összesített, felhasználók által érzékelt hasznosságot Nash-egyensúly esetén, amely mint új hálózatméretezési paraméter használható.

Szimulációs eredmények bizonyítják, hogy hagyományos méretezési paraméterek (mint például a csomagvesztési valószínűség) és a felhasználók által érzékelt hasznosság karakterisztikája különböző. Az érzékelt hasznosság érzékenyebb a hálózat túlterhelésére, különösen az elérhető maximum környékén.

Annak ellenére, hogy adott esetben minden felhasználó QoS igénye ki van elégítve, lehetséges, hogy a felhasználók költségérzékenységét is figyelembe véve nincsenek a legjobb helyzetben, mert a kívánt minőséget nem a legolcsóbb módon kapják meg. Megmutattuk, hogy a hálózatméretezési módszereknek milyen kritériumoknak kell eleget tenniük ahhoz, hogy azok a QoS igények mellett a felhasználók költségbeli preferenciáit is kielégítsék.

További kutatási terület lehet a felhasználók költségérzékenysége és az általuk érzékelt hasznosság kapcsolatának pontosabb definiálása. Ehhez kapcsolódóan a későbbiekben célszerű megvizsgálni az összefüggést a szolgáltatás ára és a felhasználók által igényelt sávszélesség mennyisége között is - azonos hasznosságérzet mellett a felhasználók várhatóan jobb minőséget nyújtó és drágább szolgáltatási osztályból kevesebbet igényelnek.

Mivel LNE kiszámítása időigényes, célszerű közelítő függvényt meghatározni a Nash-egyensúly szintjének meghatározására. Ebben az esetben lehetővé válna a szimulációs vizsgálatok gyorsabbá tétele és összetettebb konfigurációk vizsgálata.

Az LNE gyors kiszámítása lehetővé tenné, hogy az érzékelt hasznosság szerepet játsszék aukcióalapú számlázási módszerek esetén is [4]. Ebben az esetben a hálózat képes lenne arra, hogy úgy ossza szét a felhasználók között a szabad erőforrásokat, hogy az azonnal Nash-egyensúlyhoz vezessen.

Hivatkozások

1. T. Basar and G. J. Olsder, „Dynamic Noncooprative Game Theory", Academic Press, második kiadás, 1995.
2. S. Blake et al., „An Architecture for Differentiated Services", IETF RFC, rfc2475.txt, 1998.
3. J.W. Friedman, „Game Theory with Applications to Economics", Oxford University Press, 1986.
4. R. J. Gibbens, „Control and Pricing for Communication Networks", Kutatási beszámoló, Statistical Laboratory, University of Cambridge, 1999.
5. Y. Korilis, A. Lazar and A. Orda, „Architecting Noncooperative Netowrks", IEEE J. Select. Areas Commun, 13(7):1241-1251, 1995.
6. Y. Korilis, A. Lazar and A. Orda, „Achieving Network Optima Using Stackelberg Routing Strategies", IEEE/ACM Trans. on Networking}, 5(1):161-173, 1997.
7. K. Park, M. Sitharam and S. Chen, „Quality of Service Provision in Noncooperative Networks: Heterogenous Preferences, Multi-Dimensional QoS Vectors, and Burstiness", Műszaki beszámoló CSD-TR-97-039A, Dept. of Computer Sciences, Purdue University, 1998.
8. H. Ren, K. Park, „Toward a Theory of Differentiated Services", In Proc. IEEE/IFIP International Workshop on Quality of Service}, pp. 211-220, 2000.
9. J.W. Roberts, „Engineering for Quality of Service", fejezet a „Self-similar Network Traffic and Performance Evaluation" c. munkához, Szerkesztő: K. Park and W. Willinger, 1998.
10. J. Wroclawski, „The Use of RSVP with IETF Integrated Services", IETF RFC, rfc2210.txt, 1997.



Jogosultságalapú hálózatmenedzselő rendszerek megvalósítási kérdései

E. Grampín[1], J. Rubio[1], N. Vardalachos[2], A. Galis[2], J. Serrat[1]

[1]*Universitat Politècnica Catalunya, Dept Teoria Senyal I Communicacions, Jordi Girona 1-3, 08034 Barcelona, Spain, tel: +34934016786, e-mail: {grampin, jrubio, serrat}@tsc.upc.es*

[2]*University College of London, Dept. of Electrical Engineering, Torrington Place, London, WC1E7JE, UK, tel: +442076793954, e-mail: {nvardala, a.galis}@ee.ucl.ac.uk*

A jogosultságalapú hálózatmenedzsment (Policy Based Network Management, PBNM) a hatékony és egyéni igényekre szabható menedzselő rendszerek paradigmájaként vált ismertté. Bár az IETF magát az elgondolást közzétette, de az elképzelés alkalmazásmódját az ajánlás nem tartalmazza. Ez a cikk az implementálható architektúrát mutatja be és ismerteti azokat a már meglevő eszközöket, amelyek segítségével ez a gyakorlatba átültethető.

Kulcsszavak: hálózatmenedzsment, jogosultságalapú menedzsment

1. Bevezetés

A menedzsmentrendszerek szabályok összességén alapulnak. Néha ezek a szabályok meglehetősen egyszerűek, a konfigurációk menedzselése esetén, ugyanakkor lehetnek felettébb bonyolultak is, mint a vészjelző algoritmusokat irányító szabályok, amelyekkel hibakezelő rendszerekben találkozhatunk [1].

Bonyolultságuktól függetlenül a „hagyományos rendszereket" jellemzi, hogy ezeket a szabályokat rögzítik. Másképpen fogalmazva, valószínűtlen, hogy ezek a szabályok a rendszer élete során megváltozzanak. Ezeket a tervezés fázisában definiálják és kódjuk az operációs rendszerbe kerül. Annak ellenére, hogy az ezen stratégia felhasználásával tervezett menedzsmentrendszerek hasznosnak bizonyultak [2], napjainkban a szolgáltatók, a hálózatok üzemeltetői, sőt még a végfelhasználók is azt igénylik, hogy az új szolgáltatások telepítése könnyű és rugalmas legyen, ami általában magas színvonalú testreszabhatóságot von maga után.

A hagyományos operációs rendszereket nem lehet könnyen az egyedi igényeknek megfelelően átalakítani. Bizonyos esetekben a programmodulok teljes cseréje szükséges, rosszabb esetben a teljes alkalmazást ki kell cserélni. Ennek következtében új programozási paradigmákat kellett keresni, hogy le lehessen győzni a hagyományos rendszerek rugalmatlanságát. Ezen új paradigmáknak a menedzsmentrendszert irányító szabályok on-line módon történő programozásán kell alapulniuk. A szabályokat egy arra jogosult szereplőnek bármikor tudnia kell módosítani vagy cserélni. Ezzel egyidejűleg a szabályoknak el kell rejteniük a hálózatot alkotó fizikai eszközök jellegzetességeit, ezért magasabb, felhasználói szinten kell azokat létrehozni. Ezen jellegzetességek határozzák meg a jogosultságalapú menedzsment paradigmájának tulajdonságait. Ezért a jogosultság alapon menedzselt rendszerek lehetővé teszik, hogy az irányító szabályokat bármikor meg lehessen határozni, így ezek a rendszerek sokkal rugalmasabbak, könnyebben skálázhatók és az egyéni igényeknek megfelelően alakíthatók [3].

Számottevő figyelmet fordítottak arra, hogy különböző szempontok figyelembevételével fejlesszék ki az ilyen rendszereket [4]. Ezzel egyidejűleg megjelentek a piacon az ezen elgondoláson alapuló sajátságos megoldások [5] [10]. Mindemellett még hosszú utat kell megtenni annak a rendszernek a megszületéséig, amely a jogosultságalapú menedzsmenthez kapcsolódó összes előnyt képes kihasználni. Ezt a célt szem előtt tartva ez a cikk néhány, a PBNM keretrendszerének implementációjához kapcsolódó eredményt mutat be. A következő fejezet a funkcionális felépítést ismerteti, míg az utolsó fejezetben áttekintjük azokat a piacon megtalálható eszközöket, amelyek a keretrendszer bizonyos részeinek támogatására használhatók.

2. A jogosultságalapú hálózatmenedzsment funkcionális architektúrája

A jogosultságkezelés alapelve egy vagy több szabállyal van összefüggésben, amelyek feltételekre és műveletekre bonthatók tovább. Mindannyiszor, amikor a feltétel(ek) teljesül(nek), művelet(ek)et hajtanak végre. Lássunk az elmondottakra egy példát!

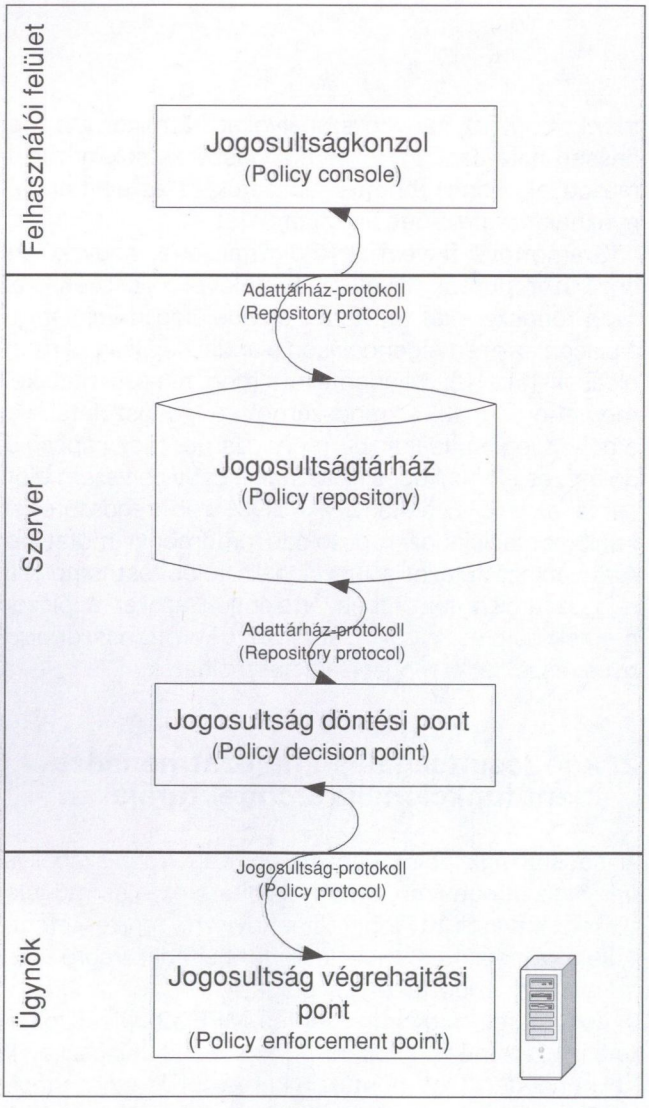
Tegyük fel, hogy jittermentes MPEG2 videoszolgáltatást szeretnénk nyújtani erre jogosult felhasználóknak, engedélyezett pontokon keresztül és előre megadott időpontokban. A következő feltételek alapján definiálunk egy szabályt:

„HA (a felhasználó a jogosult felhasználók között van)
ÉS (a szolgáltatás video)
ÉS (a forrás a videoforrások között található)
ÉS (a célállomás a video-célállomások között található)

ÉS (az időpont az előre megadott időpontok között van) a művelet a következő:
AKKOR (biztosítsunk jittermentes MPEG2-t).

A szabálycsoport irányítja a rendszert bármely tartományban és bármely működési szinten. Következésképpen a következő jogosultságokról beszélhetünk: biztonsági (security policy), hozzáférési (access policy), szolgáltatásminőségi (QoS policy), hálózati elem (network element level policy) stb. Ezzel szemben a jogosultságoknak félreérthetetleneknek és ellenőrizhetőknek kell lenniük, valamint létezniük kell olyan módszereknek, amelyek alkalmasak a jogosultságok definiálására, összeállítására és hozzáférésre is.

A PNMB rendszer működését az IETF által indítványozott architektúra révén lehet leírni, amelyet az 1. ábra szemléltet, és amelynek funkcionális alkotóelemei az alábbiak.



1. ábra A PBNM architektúrája

- Jogosultságkonzol (policy console)
  A jogosultságokat egy adminisztrátor, vagy bármely más szereplő kezelheti. A jogosultság a jogosultságkonzolon keresztül kezelhető, amely felhasználói felületként funkcionál, továbbá lehetővé teszi jogosultságok létrehozását, kialakítását és a jogosultságok által irányított környezet állapotának vizsgálatát. Egy PNBM rendszernek eszközökre van szüksége ahhoz, hogy jogosultságokat tudjon meghatározni. A jogosultságokkal kapcsolatban különféle specifikációs nyelvek használhatók, ilyen módon a jogosultságoknak számtalan formátuma létezik. Ezek közül a következőket használják a leggyakrabban:
  1. Természetes nyelvek formátuma: egy felhasználóbarát grafikus felhasználói felület támogatja az ügyfelet abban, hogy könnyedén vehessen fel és felügyelhessen különböző jogosultságokat. Ez a felhasználói felület tervezésétől függ. Ezt a formátumot a markup language formátumnak kell megfeleltetni.
  2. Markup language formátumot egy számítógép le tudja fordítani, és fel tudja dolgozni. De ezt nem kell megfeleltetni egy olyan programkódnak, amelyet közvetlenül le lehet futtatni. Ennek a nyelvnek a legismertebb példája az XML.
  3. Szabályalapú formátum a jogosultságot szabályok sorozatává fordítja le, amelyben minden szabály egy egyszerű feltétel-műveletpárral rendelkezik. Az ilyen módon megadott szabályokat egyszerűbb elemezni, mint a markup language vagy a Java osztályok által leírt szabályokat. Az ITEF specifikációjában a szabályalapú jogosultságleírás található.

- Jogosultságdöntési pont (policy decision point, PDP)
  Ez dönti el, hogy a jogosultságra vonatkozó feltételek teljesültek-e, és teljesülésük esetén a jogosultsághoz tartozó műveletek végrehajtását kezdeményezi. Ezen, legfontosabbnak tartott funkció mellett ennek az alkotóelemnek tulajdoníthatjuk a következő feladatokat is: a jogosultságütközések felderítése, külső események által igényelt fontos jogosultságok visszakeresése, valamint kapcsolattartás az alább következő PEP alkotóelemmel.

- Jogosultság végrehajtásának helye (policy enforcement point, PEP)
  Ez biztosítja, hogy a PDP által kívánt művelet végrehajtódjék. Feladata, hogy a jogosultság teljesülésének ellenőrzéséhez méréseket és vizsgálatokat hajtson végre.

- Jogosultságtárház (policy repository)
  Könyvtár és/vagy más tárolóeszköz (például relációs adatbázis), ahol a jogosultságok és a hozzájuk kapcsolódó információk tárolhatók.

- Jogosultságkommunikációs protokollok (policy communication protocol)
  Megfelelő kommunikációs protokoll is szükséges az adatcserék végrehajtásához. A PDP és a PEP közti kommunikáció támogatásával kapcsolatban gyakran említik a Common Object Policy Service (COPS)

protokollt. Mindazonáltal más módszereket, mint például a SOAP-ot (Simple Object Access Protocol) és az XML-RPC-t (XML Remote Procedure Call) is javasolják. Az adattárházhoz történő hozzáféréshez az LDAP-ot és az SQL-t használják gyakran, attól függően, hogy az adattárházat könyvtárként vagy adatbázisként implementálták.

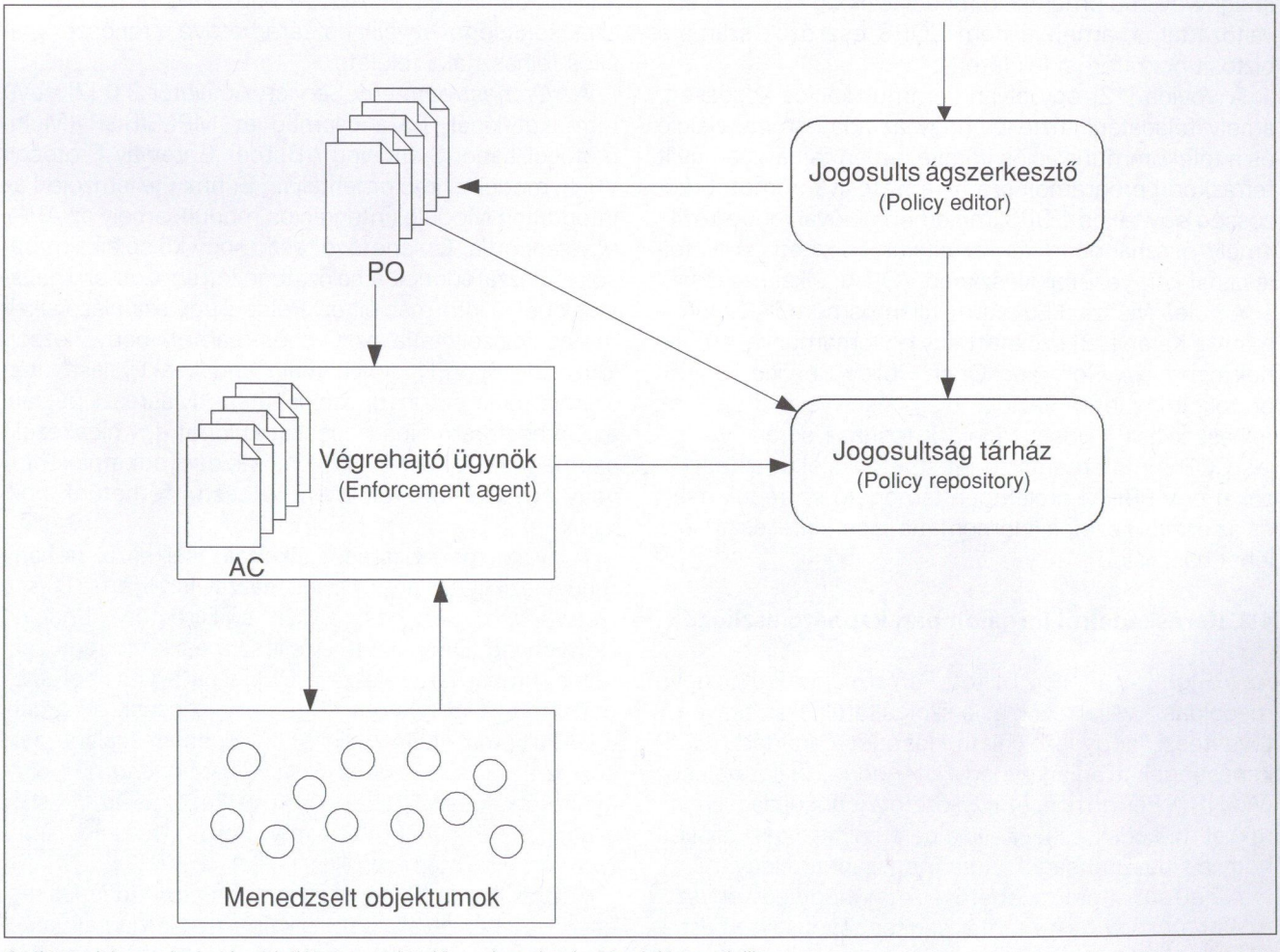
3. Implementációs modell

A 2. ábra a fent ismertetett modell szerkezetét mutatja, amelynek alkotóelemeit alább ismertetjük.

- Jogosultságszerkesztő (Policy Editor)
  A jogosultságszerkesztő biztosítja a jogosultságkezelő rendszer és a rendszeradminisztrátor közti kapcsolódási felületet. Ez fordítja a jogosultságokat jogosultságosztályokká, amelyek futási időben a jogosultságobjektumok (Policy Object) lesznek. Másrészről, ez a modul továbbítja a rendszeradminisztrátor által megszerkesztett jogosultságokat a jogosultságtárház felé.
- Jogosultságtárház (Policy Repository)
  Ez felügyeli a tartományokba csoportosított jogosultságobjektumok elosztott hierarchiáját. A tárházat LDAP szerver segítségével lehet létrehozni, amely a könyvtár tagjainak megváltozása esetén eseményeket generál, valamint lehetővé teszi, hogy az objektumok egynél több tartomány elemei legyenek.

- Jogosultságobjektumok (Policy Object, PO)
  A jogosultságobjektumok a rendszeren belül definiált jogosultságokat modellezi. A jogosultságok terjesztésének kulcsa az, hogy meghatározzuk, melyik végrehajtó ügynök felelős az adott jogosultság feloldásáért. A jogosultságobjektum értékeli ki azon felügyelt objektumok célcsoportját, amelyekhez társították, és ez határozza meg a célcsoport minden célobjektumának hozzáférés-vezérlőjét. Ezt a 2. ábrán a PO és az AC blokkok, valamint a jogosultságtárház közti nyilak jelölik.

- Végrehajtó ügynökök és hozzáférés-vezérlők (Enforcement Agent, Access Controller, AC)
  A végrehajtó ügynök az az alrendszer, amely a PDP-nek és a PEP-nek az 2. ábrán látható modellben szereplő feladatait végrehajtja. A hozzáférés-vezérlő a végrehajtó ügynök alkotórésze, és feladata, hogy megkülönböztesse a kezelt objektumok csoportjait. Ebben a modellben minden felügyelt objektumnak egyetlen hozzáférés-vezérlője (Access Controller,



2. ábra A jogosultság alapú hálózatmenedzselő rendszer implementációs modellje

AC) van, amely minden, a célobjektumra vonatkozó jogosultságot érvénybe léptethet. Az AC műveleteket küld a felügyelt objektumnak és fogadja annak jelentéseit.

4. A jogosultságalapú hálózatmenedzsment eszközei

Számtalan PBNM eszköz fejődik a tudományos és a kereskedelmi világban. A következők mutatják meg, hogy az eszközök miképpen illeszkednek az architektúrába. A továbbiakban ezeket három kategóriába soroljuk, nevezetesen a nyílt forráskódú, a kereskedelmi forgalomban kapható és a jogosultságdefiníciós nyelvek csoportjába.

4.1. Nyílt forráskódú eszközök

Az Intel [10] és az Iphighway [11] nyilvánossá tette COPS kliensük implementációját, hogy a routerek gyártói alkalmazhassák az általuk definiált protokollt. A fejlesztők ezeket az SDK-kat (Software Development Kit, programfejlesztő készlet) olyan, PEP oldali COPS kliensek kifejlesztésére használhatják, amelyeket a nem e cégek által készített jogosultságszerverekkel való együttműködésnek megfelelően alakítottak ki. Az Iphighway implementáció rendelkezik egy proxy kliens változattal is, amely a nem COPS eszközök számára biztosít hozzáférési felületet.

A Vovida [12] egy olyan kommunikációs közösség, amely feladatául tűzte ki, hogy az adatkommunikáció és a telekommunikációs környezetben felhasznált nyílt forráskódú programok számára biztosít fórumot. E közösség egy teljes COPS implementációval rendelkezik, amely összhangban van az általuk készített, VoIP fejlesztési környezetbe illeszkedő VOCAL alkalmazással.

A Luleå Műszaki Egyetem Villamosmérnöki és Informatikai Karán [13] született egy diplomamunka, amelynek címe: „A Common Open Policy Service (COPS) protokoll implementációja, elemzése, valamint ennek felhasználása jogosultságok biztosítása során." A teljes COPS implementáció letölthető. Ezeket az eszközöket egy PBNM projektben támogató szerepük miatt és az eszközkezelők (element manager) kifejlesztésére lehet használni.

4.2. Kereskedelmi forgalomban kapható eszközök

Az *IPHighway* által biztosított *PerformancePro* [5] nevű megoldás a vállalkozók és a szolgáltatók számára lehetővé teszi, hogy szolgálatminőséget garantáló (QoS) képességeket adjanak nagy kiterjedésű hálózataikhoz (WAN). A PerformancePro elosztott jogosultságszerverekkel működik, és úgy állítja be a hálózati eszközöket, hogy a kihasználtságuk előre megszabható legyen.

A PerformancePro lehetővé teszi, hogy egy egyszerű és hatékony üzleti szótár segítségével sávszélességet lehessen biztosítani, valamint egy jogosultságokat meghatározó grafikus felhasználói felület segítségével prioritás szerint lehessen sorrendezni az alkalmazások által generált forgalmat. Például, az üzleti szempontból fontos alkalmazásokat előbbre lehet sorolni, mint egy alkalmi webböngészést, vagy egy továbbítási időre érzékeny e-business tranzakció számára kisebb várakozást lehet garantálni, mint egy, a késleltetésre nem érzékeny alkalmazás, például az állományátvitel számára. Ezen eszköz rendelkezik egy publikus API-val (Application Programming Interface, alkalmazásprogramozói felület), amely a jogosultságadminisztrátor (Policy Administrator) számára biztosít hozzáférést az XML-hez.

A *Nortel Policy Services* [6] egy integrált IP szolgáltatásmenedzsmentet biztosít, amely könyvtáralapú funkcionalitás segítségével kezeli az IP címzéseket és a szolgáltatásminőséget. Kezeli a RADIUS felhasználói profilokat, valamint egy központi adattárház segítségével az IP VPN (Virtual Private Network, virtuális magánhálózat) szolgáltatás biztosítását.

A Nortel Policy Services moduláris felépítésű, amelynek alkotóelemei a következők: Dynamic Host Configuration Protocol (dinamikus gazdagép-beállító protokoll, DHCP), Domain Name Service (doménnévszolgáltatás, DNS), Quality of Service, RADIUS és IP VPN szolgáltatás. Az eszköz bizonyos mértékben kiterjeszthető az LDAP API segítségével, amely biztosítja a Directoryhoz és a JAVA API-hoz való hozzáférést – ezek segítségével válik hozzáférhetővé a rendszer grafikus felhasználói felülete.

Az *Orchestream* egy Service Activator 3.0 [7] nevű terméket kínál. Ez a csomag az MPLS/BGP (Multiprotocol Label Switching / Border Gateway Protocol) VPN modellt implementálja. Ennek jellemzője az Integration Module (integrációs modul), amely az API-k egy csoportja. Ez lehetővé teszi, hogy külső alkalmazások is hozzáférjenek a hálózatban létrehozott szolgáltatások belső információihoz. Például egy számlázó alkalmazás felszólíthatja az Orchestreamet, hogy hozzon létre egy új VPN oldalt, majd annak aktiválása után számlázhatja ezt az új szolgáltatást. Másrészt például az Orchestream hibacsapdákat küldhet egy hibakezelő csomagnak. Ezekről SNMP csapdák adhatnak hírt, vagy egy CORBA felületen keresztül férhetünk hozzájuk.

A *Cisco* menedzsment alkalmazásai közül néhány rendelkezik OSS integrációs vázzal is. Ilyen a *Cisco Networking Services* (CNS) a Directory Enabled Networking (DEN) szoftverfejlesztő eszközök egy csoportja, amely tartalmazza a DEN-t és a DEN-ből származtatott Lightweight Directory Access Protocol (LDAP) sémát, LDAP klienst. A jogosultságalapú eszközöket a *Cisco Assure Policy Networking* név alatt gyűjtötték össze. Ezek a következők: QoS Policy Manager (QPM) [8], Security Policy Manager és a Network Registar.

A QoS Policy Managere központosított jogosultságellenőrzést és a vállalkozói alkalmazások számára automatikus jogosultságkiosztást biztosít. Mindemellett

ezek az eszközök kevéssé támogatják a külső alkalmazásokkal való együttműködést.

A HP is rendelkezik menedzsmentalkalmazásokkal [9]. A *PolicyXpert* [14] a kihasználtság és a szolgáltatások kezelésének (Performance, Service Management) területét fedi le. Alapvető feladata, hogy a sávszélesség-foglalásokat és a szolgáltatásminőséget (QoS) intuitív módon kezelje. Képes IP QoS felügyeletére, biztosítja a megkülönböztetett szolgáltatásokat (diff serv).

A PolicyXpert háromrétegű elosztott architektúrával rendelkezik: a konzol, a szerver és az ügynökök. Ezek működhetnek egyetlen, vagy elosztott rendszeren belül is, ahol a konzolok és az ügynökök a jogosultságszerverhez csatlakozhatnak. Az elemek közti kommunikációt a COPS biztosítja. Az eszközök gyártói által kifejlesztett proxy ügynökök a jogosultságszerver és az egyes hálózati eszközök közti fordítóként működnek. Egy backup szerver is beállítható az automatikus átkapcsolás biztosítására.

4.3. Jogosultságdefiníciós nyelvek / eszközök

A *Routing Policy Specification Language* (RPSL) [15] lehetővé teszi a hálózat működtetője számára, hogy az internet hierarchiáján belüli különböző szinteknek (például az Autonomous System, autonóm rendszerek szintjének) megfelelően adjon meg útvonal-választási jogosultságokat. A jogosultságokat az RPSL segítségével olyan mélységben lehet megadni, hogy ezekből az útvonalválasztók alacsony szintű konfigurációja elkészíthető.

Az *RAToolSet* [16] a Routing Arbiter olyan eszközkészlete, amely ténylegesen implementálja az RPSL-t annak érdekében, hogy útvonal-választási jogosultságokat lehessen definiálni.

Az *Ilog Rules* [17] egy általános célú, magas szintű rendszergenerátor. Ezen eszköz segítségével szabályalapú alkalmazások készíthetők, a szabályalapú technikák és az objektumorientált programozás kombinálásával. Eme eszköz az alkalmazáshoz döntéstámogatást és adatfolyam-felügyeletet biztosít. Intelligens ügynökök és üzleti feladatú processzorok kifejlesztésére alkalmas. Ezen eszköz a megadott alkalmazást a felhasználó által definiált objektumokat felhasználva integrálja a szabályalapú döntéseket. Ennek az eszköznek saját jogosultságnyelve (Policy Language) van. A jogosultságok definiálását a jogosultságkonzolon (Policy Console) keresztül lehet elvégezni, az alkalmazások felgyorsítása érdekében a jogosultságokat le lehet fordítani. A szabályokat XML állományból is lehet importálni.

A *Ponder* [18] egy deklaratív, objektumorientált nyelv, amely elosztott objektumos rendszerek biztonsági és kezelési jogainak meghatározására alkalmas. A nyelv a következő alapvető jogosultságtípusok létrehozását teszi lehetővé: hozzáférési jogosultságok (Authorization Policy), amelyek megadják az elvégezhető műveleteket; eseményvezérelt kötelező jogosultságok (Event-triggered Obligation Policy), amelyek a menedzserügynökök által végrehajtandó műveleteket tartalmazzák; tiltó jogosultságok (Refrain Policy), amelyek olyan műveleteket határoznak meg, amelyek végrehajtásától az eszközöknek tartózkodniuk kell; és átruházási jogok (Delegation Policy), amelyek megadják, hogy mely hozzáférések ruházhatók át és kinek. A végrehajtó ügynököket, amelyek mind a PDP, mind a PEP műveleteket végrehajtják, a felhasználónak kell kifejlesztenie annak érdekében, hogy megvalósítsa a jogosultságszervert (Policy Server).

5. Összefoglalás

A jogosultságalapú menedzsmentrendszer fizikai felépítését azzal a céllal javasoltuk, hogy a jogosultságalapú menedzsment koncepciójának minden előnye kihasználható legyen. Ez az architektúra négy fő alkotóelemből áll, amelyek funkcionalitása illeszkedik az általános IETF keretrendszer entitásaihoz.

Számos jogosultságkezelő eszközt áttekintettünk, hogy azokból az ismertetett architektúrához néhányat kiválasszunk. A kereskedelmi forgalomban kapható eszközök a gyártóra jellemző tulajdonságokkal rendelkeznek, önállóan működő rendszernek készültek, ezért ezeket nehéz egy PBNM fejlesztésbe integrálni. A jogosultságleíró nyelveket legalább jogosultságszerkesztőként (Policy Editor) a jogosultságok definiálására, valamint jogosultságfordítóként (Policy Compiler) jogosultság-objektumok (Policy Object) készítésére lehet használni. Néhányuk, mint például az IlogRules, a végrehajtó ügynökök (Enforcement Agent) funkcionalitásának egy részét is biztosítják. Ezeket egyszerű a felhasználók által kifejlesztett programokkal integrálni. Az eszközkezelő (Element Manager) szintjén a megvizsgált nyílt forráskódú eszközök közül a COPS könyvtárak tudnak a hálózati eszközökkel kommunikálni.

Köszönetnyilvánítás

E cikk egy 2000 és 2002 között, két és fél évig zajló, IST projekt, a WINMAN keretein belül végzett és jelenleg is folyó munkát foglalja össze. Az IST programot részben az Európai Unió finanszírozza.

Irodalom

1. M. Sloman, „Network and Distributed Systems Management", Addison-Wesley, 1994.
2. A. Galis, „Multidomain Communicaitons Management Systems", CRC Press, 2000.
3. HP White Paper, „A Primer on Policy-based Network Management", 1999. szeptember.
4. M. Sloman, J. Lobo, „Policies for Distributed Systems and Networks", Int. Workshop Poilicies, Springer, 2001.
5. http://www.iphighway.com/performancepro.htm
6. http://www.nortelnetworks.com/products/04/ps/index.html
7. http://www.orchestream.com/
8. http://www.cisco.dom/warp/public/cc/pd/wr2k/qoppmn/

9. http://www.openview.hp.com/prodcategories/Documents/General_HTML-246.asp
10. http://www.developer.intel.com/ial/cops/index.html
11. http://www.iphighway.com/opensource1.html
12. http://www.vovida.org/protocols/downloads/cops/
13. http://www.sm.luth.se/csee/
14. http://www.openview.hp.com/products/policyexpert/
15. http://www.irr.net/docs/rpsl.html
16. ftp://ftp.isi.edu/ra/RAToolSet/
17. http://www..ilog.com/products/rules/
18. http://www-dse.doc.ic.ac.uk/research/policies/ponder.html

Fordította:
Dreilinger Tímea PhD hallgató BME Távközlési és Telematikai Tanszék
„Tímea Dreilinger" <s8624dre@ural2.hszk.bme.hu>

Hírek

Az AT&T Wireess és az Ericsson együttműködésének eredménye az első élő 3G UMTS hanghívás az Egyesült Államokban. Az EDGE-technológiára alkalmas GSM/GPRS rendszerek telepítése az a haladási irány, amely biztosítja a zökkenőmentes áttérést az EDGE-re, majd az UMTS-re. Az UMTS megoldás lehetővé teszi áramkör- és csomagkapcsolt szolgáltatások vegyes választékának nyújtását, egyszerre több, változó sávszélesség felhasználásával.

❖ ❖ ❖

Az Oracle Corporation új internetalapú oktatásszervező rendszert mutatott be iLearning néven, amellyel új funkcióinak köszönhetően a vállalatok saját maguk alakíthatják ki belső oktatási rendszerük felépítését, tartalmát, formáját és számonkérési módszereit.

A szoftver arra épül, hogy az Oracle E-Business Suite integrált vállalati rendszerének moduljai (pénzügyi, humán erőforrás, beszerzés) kapcsolatban állnak egymással, így az összes szükséges információ birtokában jelentősen egyszerűsödik az oktatás folyamata.

Az egyetlen vállalati vagy akár számos, földrajzilag elkülönülő telephellyel rendelkező szervezet önálló oktatási rendszere is kialakítható. Az iLearning ipari szabványoknak megfelelő alkalmazása képes több tartalomszolgáltató anyagaiból automatikusan összeválogatni, és a megadott szempontok alapján összeállítani a kívánt anyagokat.

❖ ❖ ❖

A 3G bevezetésével alapvetően változtatja a hozzáférés-átviteli rendszerek forgalomkövetelményeit. A forgalmat a bázisállomásoktól és azokig szállító szolgáltatások egyre növekvő számának következménye.

A felhasználók számának és a mobil adatforgalomnak a növekedésével a szolgáltatás minősége szempontjából létfontosságúvá válik, hogy a hozzáférés-átvitel rugalmasan ki tudja elégíteni ezeket az igényeket.

A mobilhálózat-üzemeltetők számára kifejlesztett mikrohullámú termékválaszték segítségével rugalmasan reagálhatnak a növekvő forgalomra.

Az Ericsson elkötelezettsége a nyílt szabványok iránt az új termékválaszték kifejlesztésénél is megmutatkozik, hiszen az teljes mértékben kompatíbilis a többi szállító hálózatával. Az adatok szerint az Ericsson által leszállított mikrohullámú rendszerek 25%-a olyan mobilhálózat-üzemeltetőkhöz kerül, akik más szállítóktól szerzik be hozzáférési rádióhálózatukat.

❖ ❖ ❖

A Siemens és a Deutsche Telekomhoz tartozó T-Mobile International (UMTS mobiltechnológiában) keretszerződést kötött hosszú távú együttműködésre. A DT Berlinben megtartott nemzetközi sajtótájékoztatóján írták alá a tíz évre szóló megállapodást, mely szerint a Siemens mobilkommunikációs ágazata (IC Mobile) UMTS mobilhálózatokat és kapcsolási rendszereket szállít és telepít.

Monitorozás és számlázás DiffServ hálózatokon

LADÁNYI ZSUZSANNA, SZÁSZ ANDRÁS

Az internet alkalmazásakor fontos feladat, hogy a hálózati erőforrásokat hatékonyan és igazságosan lehessen megosztani a növekvő számú felhasználók között. Ehhez elengedhetetlen, hogy pontos információk álljanak rendelkezésre a hálózat kihasználtságáról. Monitorozás alatt a hálózati információk gyűjtését és feldolgozását értjük. A cikk célja a monitorozás feladatainak összegyűjtése, valamint az ezekre az információkra épülő számlázás szükségességének hangsúlyozása. Bemutatunk több számlázási módszert, amelyek a felhasználókat hatékonyabb hálózatkihasználtságra ösztönzik. Végül ismertetünk egy általunk megvalósított rendszert, és az ott alkalmazott számlázási módszert.

1. Bevezetés

A gyors technológiai fejlődés és az újabb alkalmazások terjedése széles körben használt médiummá tette az internetet. Az interneten továbbított forgalom mennyisége nő, ami maga után vonja az igazságos és hatékony erőforrás-megosztás problémáját. Már internet-telefónia szolgáltatások is igénybe vehetők, és videó is nézhető csomagkapcsolt hálózaton keresztül. A szolgáltatási típusok differenciálódnak, és ezek különböző továbbítási módokat kívánnak. Jelenleg a legjobb szándék (best-effort) elvén alapuló algoritmus használható TCP/IP esetén, ami nem elégséges jó minőségű hang-, videoforgalom és egyéb valós idejű alkalmazás kiszolgálásához.

Az előzőkből következik, hogy különböző szolgáltatási minőségeket kell biztosítanunk, más garanciákra van szükség a különböző forgalomtípusok esetén. Az egyik irányzat az Integrated Services (IntServ) [3] architektúra, ahol az útvonalválasztók külön kezelik a folyamatokat. Ez a megoldás nehézkes nagy kiterjedésű hálózatok esetén, mert a hálózati eszközöknek túl sok folyamot kell kezelniük, ami skálázhatósági problémákkal jár. A második megközelítés a Differentiated Services (DiffServ) [2] architektúra, ahol a minőségi paramétereknek (például: maximális csomagvesztés, késleltetés, dzsitter) megfelelően a folyamokat szolgáltatási osztályokba sorolják, és az egy osztályba tartozó folyamokat együttesen továbbítja. Ez skálázhatóbb módja a forgalom továbbításának, ennélfogva nagy lehetőségeket tartogat a közeljövőben, így a cikk is a DiffServ hálózatokra helyezi a hangsúlyt.

Általános módja az internet elérésének a hagyományos telefonhálózat. A számla általában tartalmazza a havi előfizetői díjat, ami az internetszolgáltatónak fizetendő, és ehhez társul a telefonhasználati díj, amit általában helyi hívásnak megfelelően felszámolnak a kapcsolási díjjal együtt. Az Egyesült Államokban a helyi hívások átalánydíjasok (flat rate), így a felhasználók hosszú időre kapcsolódnak az internethez. Továbbá jelenleg, ha a felhasználó hozzáférhetett az internethez, akkor annyi forgalmat generálhat, amennyit csak akar, nincs korlátozás. A forgalom a legjobb szándék elve szerint továbbítódik, így nincs lehetőség garantált szolgáltatási minőség biztosítására. Ezek a problémák motiválják a szolgáltatókat, hogy nagyobb hangsúlyt fektessenek a megfelelő számlázási módszerek alkalmazására.

Az internet elérését biztosító hálózati szolgáltatók egymásnak is adnak el sávszélességet, és ehhez ők is szerződéseket kötnek. Az „ügyfél" tartományszerződést köt a „szolgáltató" tartománnyal. Ebben a szerződésben az ügyfél meghatározza, hogy mikor, melyik IP címről, melyik tartományba szeretne forgalmat generálni, milyen sávszélességre és milyen minőségi garanciákra lenne szüksége (kiválasztja a neki megfelelő szolgáltatási osztályt). Ezek a szerződések lehetnek alapjai a várható forgalom előrejelzésének, melynek előnye, hogy a hálózat előre felkészíthető a várható forgalomra, előre konfigurálható lesz. Az ügyfél generálhat nagyobb forgalmat, mint amennyit előzetesen meghatározott, de ebben az esetben büntetést kell fizetnie, ami ösztönözheti az ügyfelet arra, hogy betartsa a szerződésben leírtakat. A pontos forgalom-előrejelzés nagyon fontos, ennek következtében megfelelő számlázási módszer kiválasztása szükséges, amivel hatni lehet az előfizetőkre.

Számlázás nem létezhet monitorozás nélkül. A monitorozás fő feladata az információgyűjtés a hálózat használatáról. Ez képezi az alapját a minőségi (Quality of Service – QoS) paraméterek meghatározásának, és ellátja a számlázást részletes információkkal a generált forgalomról.

Az általunk megvalósított rendszer tartalmaz monitorozó és számlázó modult is. A monitorozó modul

adatokat gyűjt az útválasztó eszközöktől, és azokat adatbázisban tárolja. A számlázó modul összehasonlítja ezeket az adatokat a forgalmi előrejelzésekkel, és meghatározza a felszámolandó díjat. A rendszerhez tartozik egy webalapú felület, amin keresztül az ügyfelek nyomon követhetik számlájuk alakulását.

Monitorozásmenedzsment

Az internet szabványosításával foglalkozó szervezet, az IETF (Internet Engineering Task Force) „Bevezetés a monitorozás menedzsmentbe" címmel 2000 októberében megjelentetett egy ajánlást [1] a kapacitáskihasználás és az adatforgalomban jelentkező tendenciák analízisére. Vagyis a mért adatok alapján megállapítható a sávszélesség iránt felmerülő igény mértéke, és ezek alapján következtetések vonhatók le jövőbeni változásáról. Továbbá a célok közé tartozik, hogy lehetőség nyíljon ellenőrzésre és számlázásra.

Egy monitorozó rendszer működtetése megkívánja a hálózati eszközök, monitorozó szerverek és számlázó szerverek együttműködését. A hálózati eszközök gyűjtik az erőforrások használatára vonatkozó információkat, és létrehozzák a szakaszrekordot. Ez a rekord összesíti egy felhasználó erőforrás használatát az egész folyamat során, és ezt továbbítja a monitorozó szerver felé szabványos monitorozó protokollon keresztül, vagy megegyezés alapján. Ez a szerver feldolgozza a beérkezett adatokat, tárolja a szükségeseket, vagy továbbküldi azokat. Megkülönböztethetők a tartományok közötti és a tartományon belüli monitorozás eseményei, és azokat megfelelő módon kell irányítani. Tartományok közötti monitorozás esetén (a szakaszrekord tartalmazza a hálózat-hozzáférési leírót – Network Access Identifier, NAI –, ami meghatározza a továbbításhoz szükséges irányt) a csomag egy másik adminisztratív tartományba kerül átküldésre. Tartományon belüli monitorozás esetén az adatok egyenesen mehetnek a számlázó szerverhez.

A számlázáshoz nagyon fontos, hogy monitorozási információk nem veszhetnek el, mert felhasználásalapú számlázás esetén ezek az információk nélkülözhetetlenek. Ennek elkerülése érdekében ajánlatos pillanatnyi monitorozást alkalmazni, ami ellenőrzőpontokat hoz létre, és bármilyen probléma esetén (például: eszköz újraindulása) minden monitorozási adat-, szakaszrekord helyreállítható.

Garantált minőségű szolgáltatás

A különböző forgalomtípusok különböző szolgáltatási osztályokba sorolhatók. A szolgáltatási minőségnek három fontos jellemzője van [7].

Az áttetszőség vonatkozik az átvitt adat idő szerinti és szemantikai sértetlenségére. Valós idejű forgalom esetén a késleltetés nem megengedett, de bizonyos mértékű csomagvesztés tolerálható. Ellenben adatátvitel esetén a késleltetés nem kritikus tényező, de a szemantikai sértetlenség nagyon fontos.

A hozzáférhetőség a hálózathoz való hozzáférés megtagadásának valószínűségére vonatkozik, a kérés blokkolása esetén pedig a várható késleltetést határozza meg. Az interneten jelenleg az azonosítás megtörténte után nincs lehetőség a forgalom további korlátozására.

A megvalósított áteresztőképesség a legfontosabb mérhető jellemzője a szolgáltatás minőségének, például adatátvitel esetén megmutatja az adat letöltésének a sebességét. Mértékegysége például a kbit/s.

Differentiated Services

A Differentiated Services (DiffServ) [2] architektúra végponttól végpontig terjedő minőséget határoz meg. A DiffServben különböző szolgáltatási osztályokat definiálnak, amelyeknek különböző minőségi paraméterei vannak. A DiffServ előnye, hogy csak a határoló csomópontok végeznek formálást (shaping), eldobást (dropping), osztályozást (classifying) és forgalommérési feladatokat, míg a belső csomópontok (core router) csak továbbítják a csomagokat, ezzel csökkentve az adminisztrációt, skálázható megoldást nyújtva az adatátvitelhez.

Számlázási módszerek

Az internethasználat számlázási módszerei leggyakrabban az átalánydíjas elszámoláson alapulnak. A minőségi garanciák megváltoztatják ezt a helyzetet, mivel a garanciák új értéket képviselnek, és ez az, amiért az ügyfelek fizetnek a hálózati szolgáltatóknak. Így egy megfelelő számlázási módszer választása nagyon fontos feladat. Ebben a részben röviden összefoglalunk pár fontos számlázási módszert [5].

Az átalánydíjas számlázás fix havidíj felszámolását jelenti, ami független a felhasználó által generált forgalom mennyiségétől. Ennek a módszernek előnye az egyszerűsége, de igazságtalan abban a tekintetben, hogy ugyanannyit fizet az a felhasználó, aki keveset, és aki sokat használja az internetet.

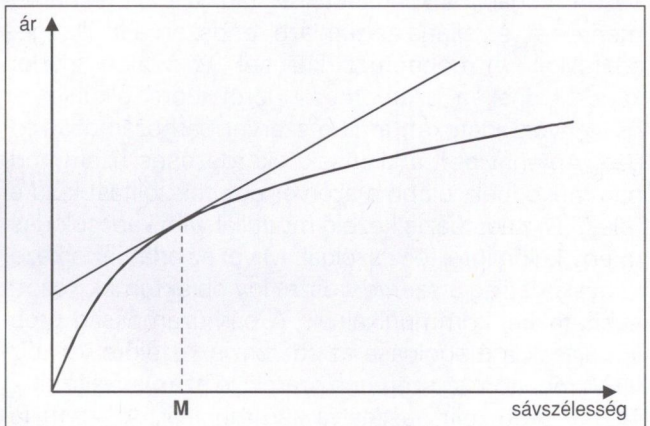
A következő módszer szerint a felhasználónak a hálózat használatának mértéke szerint kell fizetnie. Ennek két fajtája létezik, az idő és az átvitt adatmennyiség szerinti számlázás. Az idő szerinti számlázás azt veszi figyelembe, hogy a felhasználó mennyi időt töltött az internetszolgáltatóhoz kapcsolódva, és az eltöltött idő növekedésével nő a számla értéke is. Az átvitt adatmennyiség szerinti számlázás valamilyen egységenként határozza meg a fizetendő összeget, és regisztrálja, hogy a felhasználó mennyi adatot vitt át a hálózaton, és/vagy mennyi adatot fogadott. Ez a számlázási módszer továbbfejleszthető azzal, hogy különböző tarifák alkalmazhatók a nap különböző szakaszaiban, miszerint például a csúcsidőszak lenne a legdrágább.

A Smart market számlázásnak a felhasználó értékeket csatol az átvinni kívánt csomagokhoz, és ez azt jelenti, hogy a felhasználó mennyit lenne hajlandó kifizet-

ni, ha csomagjait átvitték. Vagyis a csomagok „versenyeznek". Azok a csomagok továbbítódnak legelőször, amelyek még az adott kapacitáson belül maradnak, és a legmagasabb érték tartozik hozzájuk, a többi eldobásra kerül. Minden felhasználónak annyit számolnak az átvitel után, amennyi a legnagyobb, már eldobott csomag értéke volt. Így a felhasználók visszatarthatók attól, hogy egyre nagyobb és nagyobb értékeket csatoljanak felelőtlenül a csomagjaikhoz. A gazdasági szaknyelv szerint ez „aukció".

A párizsi metró modelljén alapuló számlázás elméleti jelentőségű. A hálózatot két egyenlő kapacitású alhálózatra kell felbontani, ahol a két részhez különböző tarifa tartozik (első és másodosztály). Valószínűleg kevesebb felhasználó fizetné meg az első osztály magasabb tarifáját, így a hálózatnak azt a részét kevesebben használnák, itt nagyobb sávszélesség maradna, ezáltal a jobb minőség biztosítható.

A legvégül bemutatásra kerülő számlázási elv az effektív sávszélességen [4], [6] alapszik, amit hatásos módszernek tartanak, így most ez a legfontosabb modell számunkra. Az 1. ábra mutatja, hogy hogyan változnak a szolgáltató költségei, ahogy a felhasznált sávszélesség nő.



1. ábra Effektív sávszélességen alapuló számlázási módszer

Az effektív sávszélesség egy skalár, ami összegzi egy kapcsolat erőforrásigényét, figyelembe véve saját és a multiplexelt más kapcsolatok minőségi követelményeit. Az ábrán látható, hogy az effektív sávszélesség függvénye nemlineáris, míg a felszámolásra kerülő érték függvénye lineáris, ami egy egyszerű módszert eredményez. Az ügyfél szerződést köt a szolgáltatóval, amely előrejelzésben meghatározza az M paramétert, ami azt jelenti, hogy mekkora sávszélességre van szükség egy adott periódusban. A szolgáltató az M pontban meghúzza az effektív sávszélességhez az érintőt, és ha az ügyfél valamilyen irányban eltér a jósolt értéktől, akkor büntetést kell fizetnie, aminek az értéke az érintő függvénye szerint változik. Amint az ábra mutatja, az effektív sávszélesség és az érintő közötti különbség egyre nagyobb, melynek hatása az ügyfélre, hogy ne térjen el a szerződésben megadott értéktől. Büntetést kell fizetni abban az esetben is, ha az ügyfél negatív irányban tér el a megadottól, mert a szolgáltatónak kompenzálnia kell azt a kiesést, ami a nem értékesített sávszélességet kompenzálja.

3. A megvalósított rendszer

A megvalósított monitorozó és számlázó rendszert a svéd Telia Research AB és a Budapesti Műszaki és Gazdaságtudományi Egyetemen található Nagysebességű Hálózatok Laboratórium (HSNLAB) együttműködésével, a TraFIPAX projektben valósítottunk meg. Ennek célja végponttól végpontig terjedő minőségi szolgáltatás (end-to-end QoS) biztosítása.

A gazdasági szempontokat tekintve a szolgáltatóknak érdekük, hogy a hálózatuk a lehető legnagyobb mértékig ki legyen használva mindaddig, amíg ez még nem okoz torlódást a hálózatban. Ehhez (és a minőségi garanciák biztosításához) szükségük van információkra a várható sávszélességigényekről. Ez az előzetes információigény kielégíthető az előzőekben bevezetett forgalmi előrejelzések használatával.

Ezek a forgalmi előrejelzések egy adatbázisban tárolódnak. Ennek az adatbázisnak minden egyes rekordja egy útvonalat azonosít. Pontosabban egy határoló útválasztó eszköz valamelyik interfésze (ez az adatcsomagoknak a hálózatba való belépési pontja) és egy céltartomány (ez pedig meghatározza a hálózatból való kilépési pontot) közötti összefüggést ír le. Minden rekordon belül az alábbi mezőket találjuk meg:

- a kívánt minőségű szolgáltatás azonosítója (a DiffServ CodePoint – DSCP),
- az érvényesség ideje (kezdő- és végpont),
- a kívánt sávszélesség,
- a határoló útválasztó interfészének azonosítója (IP cím és alhálózati maszk), amivel így a küldő fél azonosítható,
- a céltartomány azonosítója (IP cím és alhálózati maszk), amivel így a kilépési tartomány azonosítható.

Ezeket az adatokat felhasználva a szolgáltató előre számolhat a várható forgalommal. Így az egyes ügyfeleknek a megadott irányokba a megfelelő minőségű átvitelhez szükséges erőforrásokat biztosítani tudja. Ehhez rendelkeznie kell olyan eszközökkel, amelyekkel ellenőrizni tudja az ügyfelek forgalmát, és rá tudja őket venni az előrejelzéseik betartására.

Erre szolgál ez a megvalósított rendszer is. Képes a hálózati forgalom mérésére, az előrejelzésnek nem megfelelő forgalom észlelésére, ennek szankcionálására és az erőforrások használatának kiszámlázására a felhasználók felé. A rendszer két fő részből áll: egy monitorozó és egy számlázó részből, amelyek szoros kapcsolatban állnak egymással. A monitorozó rész gyűjti és tárolja a hálózathasználati adatokat, a számlázó rész kiszámolja és megjeleníti a számlát.

A monitorozó rendszer

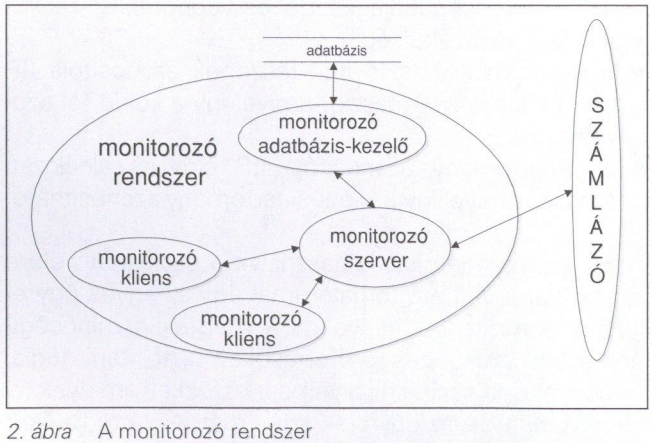
Ennek az a feladata, hogy ügyfelekre és szolgáltatási osztályokra bontva mérje a hálózati forgalmat, valamint a számlázó modult ellássa megfelelő adatokkal. Itt a kapcsolódó hálózatokat (alhálózatokat) ügyfeleknek tekintjük, akik a megfigyelt DiffServ tartományon keresztül adatokat juttatnak el, vagy adatokat fogadnak egy harmadik (al)hálózatból. Így csak a hálózaton keresztül menő forgalom mérése a feladat.

A hálózaton átmenő forgalom pontos méréséhez elegendő a hálózatba belépő forgalom nagyságát megfigyelni. Ezt azért tehetjük meg, mert feltételezhetjük, hogy a saját tartományunk csak átmenő forgalmat bonyolít le – legalábbis a fellépő forgalom túlnyomó többsége áthalad rajta. Ez a megközelítés tipikusan igaz a több hálózat összekötését megvalósító gerinchálózatokra. Így tehát a forgalommérés csak a hálózatot határoló útválasztók belépő oldalán történik.

A megvalósított rendszerben minden határoló útválasztó méri a belépő forgalmat, és átmenetileg tárolja az eredményeket. Egy bizonyos mennyiségű adat összegyűjtése után elküldi azt egy központi szervernek, mely képes ezeket a forgalmi adatokat hosszú időre tárolni. Továbbá ennek a központi szervernek a feladata, hogy a számlázó rendszert ellássa a megfelelő adatokkal.

A monitorozó alrendszer három fő modulból tevődik össze (2. ábra):

- monitorozó kliens (a határoló útválasztókon),
- monitorozó szerver (a központi szerveren),
- monitorozó adatbázis-kezelő (a szerverrel azonos gépen).



2. ábra A monitorozó rendszer

A monitorozó kliens

A kliens feladata a hálózatba érkező forgalom mérése, és a mérési adatok továbbítása a központi gépen futó monitorozó szerverhez. Ez a modul megkülönböztet azoktól a hálózaton áthaladó csomagoktól, amelynek a címzettje a hálózat belsejében található. Ez azért fontos, mivel minket csak az átmenő forgalom érdekel.

A kliensnek képesnek kell lennie konfiguráló parancsok fogadására. Ezáltal a rendszer rugalmasnak tekinthető, mivel menet közben, akár a szerverről állíthatunk be fontos működési paramétereket. Másik hasznos képessége, hogy egy megjelenítő programot el tud látni valós idejű adatokkal, így kis hálózat esetében valós időben követhető a forgalom nagysága. Szintén fontos szempont, hogy a kliens a lehető legkisebbre csökkentse a monitorozással keletkező extra hálózati terhelést pufferelléssel és egy jól megválasztott protokoll használatával.

A monitorozó kliens az adatok elküldése szerint a szerverhez való kétféle üzemmódban működhet [1]. Eseményvezérelt üzemmódban egy esemény bekövetkeztekor küldi el az adatokat (pl. egy megadott idő leteltekor). Lekérdezéses üzemmódban a mérési eredményeket csak a szerver felszólítására küldi el.

Az adattovábbítás módján kívül van még két paraméter, amelyek alapvetően befolyásolják a kliens működését. Az összegzési idő változtatásával a pontosabb számlázás és a kisebb hálózati terhelés közti optimum állítható be. Az összegzések száma pedig megadja, hogy maximálisan mennyi adatot lehet ideiglenesen tárolni a szerverhez való elküldés előtt.

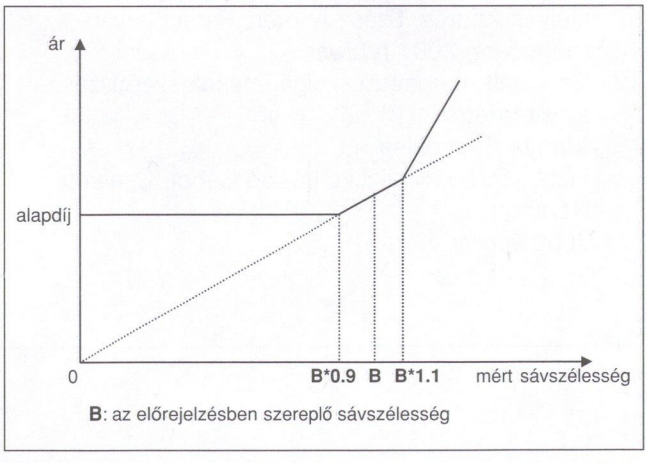
A monitorozó szerver legfontosabb feladata, hogy összekötőkapocsként szolgáljon a különböző modulok között. Fogadja a kliensektől és tárolja a mérési eredményeket, és ellátja a számlázó rendszert a szükséges adatokkal. A monitorozó kliensek közvetlen módon kapcsolódnak a szerverhez. Időről időre elküldik az összegyűlt adatokat, amit a szerver párhuzamosan fogad. Amennyiben a kliensek lekérdezéses üzemmódban működnek, előbb a szerver egy felszólítást küld el feléjük. Az adatbázis kezelő modullal való kapcsolódásra egy külön interfész szolgál. Mivel az adatbázis kezelő gyakorlatilag a szerver része, így objektumok közötti üzenetekkel kommunikálnak. A párhuzamosság problémájának a megoldása az adatbázis kezelőre van bízva. A monitorozó szerver a számlázó szerverrel szintén TCP/IP protokollt használva kommunikál. A kapott lekérdezést továbbítja az adatbázis-illesztőnek, majd a választ (hálózathasználati adatokat) visszaküldi a számlázónak. Ezek a szolgáltatások mellett közvetlen kapcsolatot tart a felhasználóval egy parancssoros kezelőfelületen keresztül. Ezt használva befolyásolhatjuk a rendszer működését (pl. bekapcsolhatjuk a naplózást, működési paramétereket küldhetünk a klienseknek, leállíthatjuk a szervert).

A monitorozó adatbázis-kezelő modul feladata, hogy a szerver által küldött forgalmi adatokat tárolja, később pedig kérésre azokat előhívja az adatbázisból. Az adatok tárolására miniSQL adatbázis-kezelő rendszert [8] használ. Mivel a modul szabványos SQL utasításokat használ, az alatta futó adatbázis-kezelő rendszer egyszerűen lecserélhető. Mivel az adatbázis-kezelő a szerver modul része, így a többi modul, amelyek adatot akarnak küldeni vagy adatot akarnak lekérdezni, csak a szerveren keresztül tudnak vele kapcsolatot teremteni. A szerver és az adatbázis modul közti kommunikáció puffereken keresztül zajlik. A szerver minden szálához tartozik az adatbázis modulból egy példány, így az adat-

bázishoz való párhuzamos hozzáférés zárakkal van megvalósítva. Az adatbázisban tárolt adatok megfelelnek a fejezet bevezető részében leírtaknak. Mindössze két eltérés található: az igényelt sávszélesség helyett a forgalmazott adatmennyiség, valamint a mérést végző útválasztó azonosítója kerül eltárolásra.

A számlázó rendszer

A számlázó rendszer feladata, hogy a monitorozó rendszer által összegyűjtött forgalmi adatok alapján előállítsa az ügyfelek számláit. Ez történhet az üzemeltető kérésére, vagy pedig előre is ütemezhető. Ekkor le kell kérdeznie a monitorozó rendszertől a számlázáshoz szükséges adatokat, majd végül az üzemeltető számára meg kell jelenítenie az eredményeket. A korrekt számla kiállításához szükséges a forgalmi előrejelzésekben található információ, amit a szolgáltató felhasználhat a számla létrehozásakor. Amikor a felhasználó kevesebb adattal terheli a hálózatot, mint ami az előrejelzésben található, a fel nem használt sávszélesség a hálózati szolgáltató vesztesége lenne. A jelzett mennyiségnél több adat forgalmazása esetén olyan problémák léphetnek fel a hálózatban, mint torlódás, magasabb csomagvesztési ráta, és a vállalt minőségi szolgáltatások követelményei más folyamok esetén sem teljesülnének. Ezekre a problémákra kínál megoldást a megvalósított számlázási rendszer. Ennek a használatával az ügyfél igyekszik az előrejelzéseinek a betartására és arra, hogy minél pontosabb előrejelzéseket adjon. Ehhez egy olyan számlázó módszert használ, ami hasonló a már előzőleg bemutatott effektív sávszélességen alapuló megoldásokhoz. Amennyiben az ügyfél eltér az előrejelzésében szereplő adatmennyiségtől, akkor büntetőtarifát számol fel neki. Így, mivel többet kell fizetnie, a saját érdekében igyekszik pontos adatokat adni előrejelzésként.

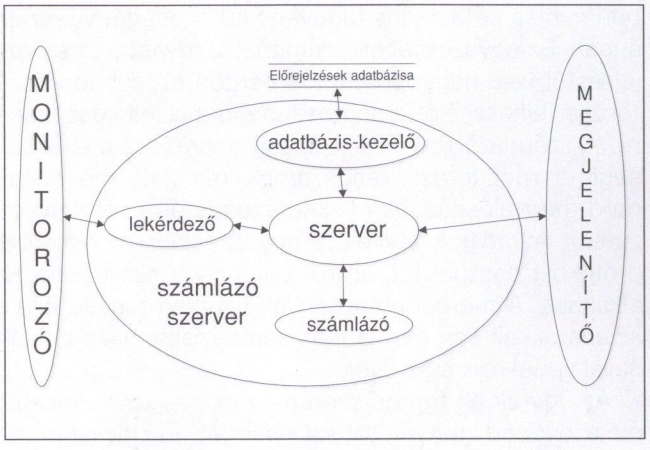


3. ábra A számlázó függvény

A számlázáshoz használt függvényt a 3. ábra mutatja. Az aktuálisan felhasznált sávszélességet összehasonlítja az előrejelzett értékkel, és ha az egy megadott környezetébe esik (esetünkben ez ± 10%), akkor a normál használat szerinti díjat számlázza. Amennyiben az alsó határnál lejjebb esik, akkor egy rögzített alapárral számol. Végül, ha a felső korlátnál is nagyobb a használt sávszélesség, akkor büntetőtarifát használ a számlázáshoz. Ezt a módszert alkalmazva kellően igazságos számlázást lehet megvalósítani.

A számlázó rendszer két részből áll: a számlázó szerverből és a megjelenítő felületből.

A számlázó szerver feladata, hogy számlalekérdezéseket várjon az operátortól, kiszámolja a számlát, majd ezt visszaküldje a lekérdezőnek. A szerver a forgalmi előrejelzéseket egy adatbázisból kapja, majd ezek segítségével állítja elő a végleges számlát. A hálózathasználati adatok a monitorozó rendszerben tárolódnak, így a számlázó szervernek le kell ezeket kérdeznie a monitorozó szervertől. Az előző fejezetben ismertetett számlázó függvény lecserélhető, így egy új számlázási módszer egyszerűen megvalósítható. A szerver többszálú, több kliens tud egy időben csatlakozni hozzá, így egyszerre több lekérdezés is futhat. A rendszer többi elemével TCP/IP protokollt használva kommunikál.



4. ábra A számlázó szerver

A számlázó szerver négy modulból áll (4. ábra). A szerver modul felelős a modulok működésének összehangolásáért, szinkronizálásáért, biztosítja az interfészt a megjelenítő felület felé. A monitorozó rendszerhez illeszkedő lekérdező modul feladata, hogy megvalósítsa a kommunikációt a számlázó és a monitorozó rendszer között. Ez alatt azt értjük, hogy a szerver modultól megkapott lekérdezést el kell juttatnia a monitorozó rendszerhez, majd az onnan válaszul visszaérkező adatokat továbbítania kell a szerver modul felé. Az adatbázis-kezelő modul felelős azért, hogy a forgalmi előrejelzés létezését ellenőrizze, és visszaadja a szervernek az ott tárolt várható sávszélesség értéket. Végül pedig a számlázó modul feladata, hogy a várható és a ténylegesen mért adatok alapján a megfelelő számlázási módszert alkalmazva kiszámolja a fizetendő összeget.

A megjelenítő felület szolgál felhasználói interfészként az ismertetett monitorozó és számlázó rendszerhez. Ez CGI [9] modulként került megvalósításra, így a lekérdezés paraméterei egy web lapon található űrlapon adhatók meg, és az eredmények egy új oldalon

jelennek meg. Az űrlapon megadott paramétereket ez a modul még a továbbküldés előtt alapos vizsgálatoknak veti alá. Ezáltal elkerülhető az esetleges értelmetlen kérések lefuttatása. Az itt, valamint a rendszer más moduljaiban történt hibákról a kezelőfelületen keresztül tájékoztatást kapunk.

4. Összefoglalás

Az ismertetett számlázó rendszer érdekeltté teszi az ügyfeleket a pontos forgalmi előrejelzésében. Ezeket az előrejelzéseket használva a hálózati operátor számolhat a várható forgalommal, előre elvégezheti a hálózat beállítását annak érdekében, hogy a kihasználtság közel optimális szinten legyen.

A megvalósított rendszerben használt számlázási módszer hasonló alapokon nyugszik, mint az effektív sávszélességre épülő megoldás. Azonban míg ez utóbbi egy logaritmikus és egy lineáris függvény közti eltérést tekinti többletszámlázásnak, addig a leírt módszerben mindig két lineáris függvény különbségét vesszük annak. Ez egyszerűbben számítható, továbbá az előrejelzéstől való nagyobb eltérés esetén erősebben ösztönzi a felhasználót a pontosabb előrejelzés készítésére. A számlázó rendszer ellátása naprakész adatokkal segít a monitorozó rendszernek mindkét működési mód megvalósításában (lekérdezéses és eseményvezérelt). Az adatok átvitelére egy speciális monitorozó protokollt használ [1], amely parancsok szállítására is alkalmas. A mérési eredmények váratlan törlése ellen az adatoknak egy permanens adatbázisban való tárolásával védekezik a szerver.

Az ügyfélnek fontos szerepe van: meg kell becsülnie a jövőben igényelt sávszélességét. Ez úgy lehetséges, hogy közelebb hozzuk hozzá a számlázás folyamatát, vagyis számára is érthető algoritmust fogalmazunk meg, és követhetővé tesszük számára számlája egyenlegének követését. Ehhez használható a rendszerben található WWW alapú lekérdező felület.

Az internet további terjedése, a hálózati forgalom összetételének változása a számlázási módszerek fejlődését vonhatja maga után. A forgalmi előrejelzésen alapuló számlázási módszerek segíthetnek megoldani a rendelkezésre álló sávszélesség optimális kihasználásának problémáját. Ezek a módszerek kiegészítői lehetnek a DiffServ hálózatokon való garantált minőséget nyújtó szolgáltatásoknak.

Irodalom

1. B. Aboba, J. Arkko, D. Harrington, Introduction to Accounting Management, IETF RFC 2975, 2000. október
2. S. Blake et al., An Architecture for Differentiated Services, IETF RFC 2475, 1998. december
3. R. Braden, D. Clark, S. Shenker, Integrated Services in the Internet Architecture: an Overview, IETF RFC 1633, 1994. június
4. C. Courcoubetis, Vasilios A. Siris, An Evaluation of Pricing Schemes that are based on Effective Usage, IEEE International Conference on Communications (ICC '98), Atlanta, Georgia, USA, 1998. június
5. R. Gibbens, Control and pricing for communication networks, Kutatási beszámoló, Statistical Laboratory, University of Cambridge, 1999.
6. F. P. Kelly, Notes on effective bandwidths, fejezet a Stochastic Networks című könyv Theory and Applications fejezetéhez (141-168. oldal), szerkesztők: F. P. Kelly, S. Zachary, and I. Zeidins, Oxford University Press, 1996.
7. J. W. Roberts, Engineering for Quality of Service, fejezet a Self-similar Network Traffic and Performance Evaluation c. munkához, szerkesztők: K.Park and W. Willinger, 1998.
8. MiniSQL: www.hughes.com.au/library/msql/
9. CGI: Common Gateway Interface: http://www.w3.org/CGI/
10. Z. Heszberger: Pricing ATM services, Magyar Távközlés – Hungarian Telecommunications Periodical, volume 8, pages 44–47, July, 1997
11. számlázható és forgalomméró rendszerek a BME TTT hálózat menedzsment laboratóriumában
    Magyar Távközlés
    Gulyás András, Szász András, Pataki István
    megjelent, 2001 február
12. Garantált minőségű szolgáltatások számlázásas az Interneten
    Magyar Távközlés
    Füzesi Péter, Malicskó László Gábor, Cselényi István
    2000 április

Hír

A Sun a Linux egyik legnagyobb intellektuális támogatója marad a linuxos és a nyílt forráskódú közösségnek, hozzájárulva a Linux kernel fejlesztéséhez is. A Sun szándékában áll nyílt forráskódként megjelentetni a közelgő Liberty technológiáit.

Elosztott frekvenciakiosztási algoritmusok tervezése és vizsgálata

BIRÓ PÉTER–LÓJA KRISZTINA–SÜTŐ MÁRTON

Budapesti Műszaki és Gazdaságtudományi Egyetem (BME) matematikushallgatók

Bevezetés

Egy általános tendencia a mobil, celluláris hálózatok fejlődésében a mind kisebb és kisebb méretű cellák alkalmazása. Minderre egyrészt az egyre nagyobb frekvenciák használata, másrészt az egyre nagyobb kapacitás iránti igények miatt van szükség. A GHz tartományban a rádióhullámok terjedése a fény terjedéséhez hasonlítható, ahol a rádióhullámok könnyen elnyelődnek a különféle tereptárgyakon és akadályokon. Ebben a sorban rendszerint közvetlen rálátás szükséges az adó és a vevő között a kapcsolat érdekében. Mindez a terepviszonyoktól függően kisebb és kisebb cellák alkalmazását teszi szükségessé, ugyanakkor a kapacitás növelésének is az egyik módja a kisebb cellák használata. Ilyenkor ugyanazon bázisállomás kapacitása egy kisebb területen belül áll rendelkezésre, amely lehetővé teszi nagyobb felhasználósűrűségű területek kiszolgálását.

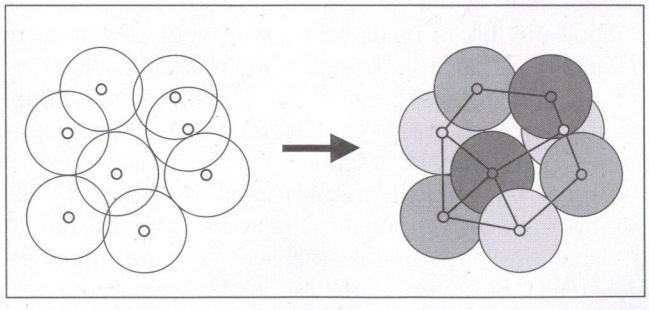
Az ilyen mikro- vagy pikócellás környezetek egy nagyváros sűrűn lakott vagy üzleti negyedeiben, illetve egy épületen belüli mobil rendszerként képzelhetők el. A konkrét környezettől és rendszertől függetlenül a cellák nagy száma miatt egy ilyen hálózat konfigurációja, cellatervezése, valamint a frekvenciaspektrum használatának koordinálása egyre nehézkesebbé válik. Továbbá ezen rendszerek egy része, tipikusan az épületen belüli rendszerek a felhasználó által is telepíthetők kell hogy legyenek (pl. mai Wireless LAN rendszerek), ahol nem várható el a felhasználótól, hogy részletes cella- és spektrumtervezést végezzen, vagy egyéb bonyolult konfigurációs műveleteket hajtson végre. Ehelyett megköveteljük a rendszertől, hogy autokonfiguráció útján a szükséges működési paramétereket saját maga automatikusan állítsa be.

Az egyik legfontosabb konfigurációs lépés a frekvenciák kiosztása úgy, hogy a szomszédos bázisállomások különböző frekvenciát használjanak és így a cellák közötti interferencia elkerülhető legyen. Ezt a jelenlegi cellás hálózatokban általában a hálózat telepítését megelőző, „off-line" módon végrehajtott frekvenciatervezéssel oldják meg.

Ezen cikk keretében az automatikus frekvenciaválasztás problémájával foglalkozunk, ahol a feladat a fenti, frekvenciakiosztási probléma megoldása elosztott, „on-line" módon. Olyan algoritmusok tervezésével és vizsgálatával foglalkozunk, amelyek az egyes bázisállomásokban működve képesek a frekvenciájukat úgy meghatározni, hogy a szomszédos bázisállomások által használt frekvenciák különbözők legyenek. Fontos különbség a hagyományos cellatervezéshez képest, hogy ezek az algoritmusok on-line módon és tipikusan lokális információt felhasználva működnek, míg egy off-line cellatervezést az egész hálózat figyelembevételével és esetleg időigényes, bonyolult algoritmusok használatával végzik.

1. Matematikai modell

A probléma matematikai modelljét az alább definiált, úgynevezett interferencia-gráf adja. A gráf pontjai felelnek meg az állomásoknak, két pont között pontosan akkor megy él a gráfban, ha a megfelelő adók egy adott kritikus távolságnál (d) közelebb vannak egymáshoz; $d \geq 2R$, ahol R a bázisállomások hatósugara.



1. ábra Az interferencia-gráf származtatása

Feladatunk a gráf pontjainak kiszínezése oly módon, hogy a szomszédos pontok mind különböző színűek legyenek. Továbbá a színezést végrehajtó algoritmus elosztott módon megvalósítható kell legyen úgy, hogy az állomások vagy kizárólag lokális információ felhasználásával, vagy a szomszédokkal folytatott minimális információcsere alapján választanak frekvenciát. Feltételezzük, hogy a bázisállomások képesek mérni az interferencia szintjét az összes létező frekvencián és ez alapján érzékelni, hogy a környezetükben mely frekvenciák szabadok és melyek nem. Ez adja az alap, loká-

lis információt, amely minden esetben rendelkezésünkre áll. Ezek után egy érdekes kérdés annak vizsgálata, hogy a bázisállomások közötti további információcsere bevezetésével mennyiben javítható az egyes algoritmusok teljesítménye.

Az algoritmusok teljesítményének egyik legfontosabb mérőszáma a felhasznált színek száma, illetve a meghiúsult, úgynevezett „blokkolt" esetek aránya, amikor az algoritmus egy adott színszámkorlát mellett nem tudta a gráfot kiszínezni. Később az algoritmusok kiértékelésekor ezen mérőszámokat használjuk majd.

Az algoritmusok teljesítményét kétfajta gráf esetén vizsgáltuk. Egyrészt olyan gráfokon végeztük a számítást, melyeknek ismerjük a kromatikus számát, ekkor ugyanis pontosan tudjuk, hogy az algoritmus mennyit hibázott, azaz mennyivel több színt használt a minimálisan szükségesnél. Másrészt olyan gráfokat is vizsgáltunk, amelyek egy tipikus bázisállomás-elhelyezkedés esetén állhatnak elő. Ez utóbbi esetben két jellemző szempontot vettünk figyelembe. Az egyik az adott terület közel egyenletes lefedése, míg a másik az ad hoc jellegű telepítésből eredő véletlenszerűség.

2. A színezés elméleti korlátai

Egy gráf kiszínezhetőségének eldöntése ($k \geq 3$ konstans esetén) egyike az ismert NP-teljes problémáknak. De ugyanígy NP-nehéz probléma egy gráf kiszínezése akkor is, ha ismerjük a gráf kromatikus számát ($\chi(G)$-t), ahol χ és G, tehát $\chi(G)$ azt a legkisebb számot, ahány színnel létezik jó színezése a gráfnak. Mindez azt jelenti, hogy – bár a számítástudomány talán legnagyobb sejtése: $P \neq NP$ még nem nyert bizonyítást – nem tudunk adni olyan polinom idejű algoritmust, ami általánosságban mindig optimálisan oldja meg a feladatot. Sőt Johnson tétele [5] szerint általános esetben még multiplikatív hibán belüli színezhetőséget sem tudunk garantálni, ha ugyanis létezne olyan polinom idejű algoritmus, ami legfeljebb $c\chi(G)$ ($c>0$ konstans) színt használ minden G gráfra, akkor létezne polinom idejű algoritmus $\chi(G)$ meghatározására is.

Vannak ugyan olyan gráfosztályok, melyekre sikerült pozitív tételeket kimondani. A legismertebb ezek közül a síkba rajzolható gráfok osztálya, ahol az Appel és Haken által bizonyított híres 4-szín tétel szerint minden gráf kiszínezhető legfeljebb négy színnel. Érdekes számunkra Borodin tétele [5] is, mely szerint minden olyan gráf, amelyet le lehet úgy rajzolni a síkon, hogy minden élét legfeljebb csak egy másik metszhet, mindig 6-színezhető.

Sajnos az általunk vizsgált eset az állomások rendezetlen elhelyezkedése miatt ennél általánosabb, és így, korlátozó feltételek híján nem remélhető olyan polinom idejű algoritmus, amely mindig optimálisan oldaná meg a feladatot. Általános tapasztalat, hogy az exponenciális algoritmusok körülbelül 20 pontig még képesek kezelni az ilyen típusú feladatokat, de ezenfelül már rendkívül gyorsan növekszik a futásigényük. Mindezek okán célunk arra korlátozódott, hogy olyan polinomrendű algoritmusokat keressünk, melyek hipotetikusan jól közelítik az optimális megoldást, ennek megfelelően algoritmusaink futási ideje csupán másodpercekben mérhető.

3. Az algoritmusok bemutatása

Az egyes színező algoritmusokat három fő szempont szerint csoportosítottuk. Az első a gráf pontjainak bejárása, a második szempont a színválasztási stratégia, míg a harmadik az úgynevezett javító eljárás lehetősége.

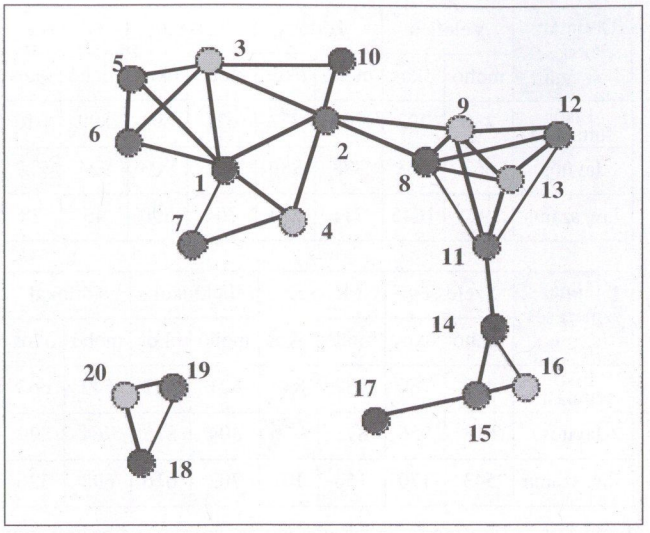
A szimulációk során megfigyeltük, hogy egy színező algoritmus hatékonysága azon múlik, hogy milyen sorrendben járja be a gráf pontjait. A színválasztás módja, illetve egy esetleges javító eljárás használata csak kisebb mértékben befolyásolják a hatékonyságot, de ennek ellenére szerepük nem elhanyagolható.

Bejárás: A véletlen bejárás mellett három alapbejárást vizsgáltunk, melyekhez szükséges a teljes gráf ismerete, majd ezt követően ezekből fejlesztettünk ki négy olyan bejárási módot, melyek a lokalitás feltételének is megfelelnek és így valóban megoldásai a kitűzött feladatnak.

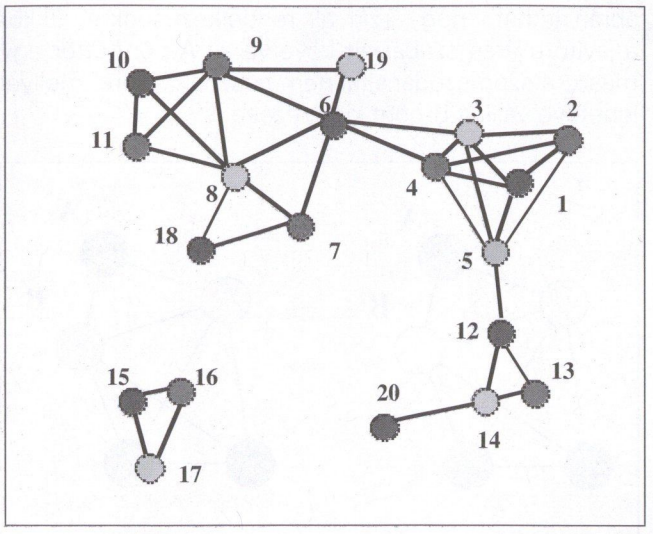
- *Véletlen bejárás:* A pontokat véletlen sorrendben járjuk be. Ez gyakorlatilag annak a megfelelője, hogy az állomásokat előzetes tervezés nélkül, véletlenszerűen kapcsoljuk be. Jelen cikk egyik témája pontosan annak megmutatása, hogy ez a gyakorta használt eljárás mekkora pazarlással jár a felhasznált színek számát tekintve.
- *Törlős bejárás:* A gráf pontjai közül sorban kitöröljük a legkisebb fokszámút, és közben visszafelé sorszámozzuk őket. Amelyik pont utoljára marad, azt fogja kiszínezni először az algoritmus. Az interferenciagráfok esetén látni fogjuk, hogy ez az eljárás különösen hatékonynak bizonyul.
- *Fokszám szerinti bejárás:* Az algoritmus a fokszámok szerint csökkenő sorrendben számozza meg a pontokat. Először tehát a legnagyobb fokút, és utoljára a legkisebb fokút fogja kiszínezni.
- *Globális bejárás:* A pontok bejárásának sorrendjét az eddigiektől eltérően dinamikusan határozza meg, vagyis figyelembe veszi, hogy addig mely pontok és miként lettek kiszínezve. Mindig azt a pontot fogjuk kiszínezni előbb, melynek a szomszédjai már a legtöbb színt elhasználták, vagyis amelyiknek a legkevesebb a választási lehetősége. A pontok fokszáma csak másodsorban dönt.

Saját fejlesztésű bejárások

- *Szélességi bejárás:* Az előző eljárás szerint, de csak lokális környezetből (a szomszédok közül) választ új pontot, irányított szélességi bejárás során.
- *Mélységi bejárás:* Hasonlóképpen az előzőhöz, csak irányított mélységi bejárást használ.
- *Lokális fokszámos bejárás:* Minden olyan pontot, melyek fokszáma nagyobb a szomszédainál, sor-



2. ábra Szélességi bejárás



3. ábra Törlős bejárás

rendben megszámozzuk, majd a maradék pontokon ismételjük meg az előző eljárást, így kapjuk meg a pontok bejárásának sorrendjét.

- *Lokális törlős bejárás:* Minden olyan pontot, melynek fokszáma kisebb a szomszédainál, visszafelé megszámozunk, majd töröljük a gráfból. A maradék gráfon folytatjuk az előző eljárást, amíg a pontok el nem fogynak.

A következő ábrákon a törlős és a szélességi bejárás szerinti színezésekre láthatunk egy példát. A pontokat jelző számok a bejárás sorrendjét, míg a színek a pont választott színét jelzik (2., 3. ábra).

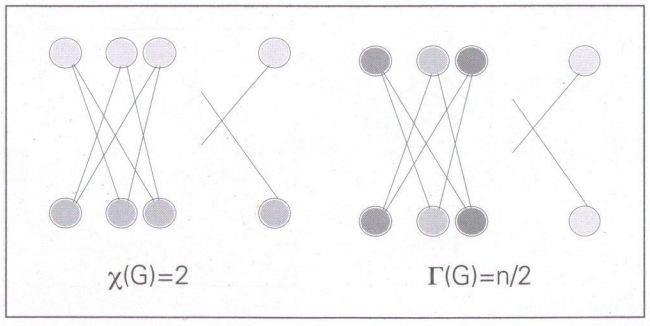
Színválasztás: Tesztjeink során több színválasztást is kipróbáltunk, és végül két olyat tartottunk meg, melyek a leghatékonyabb bejárások szerint a legjobb eredményre vezettek.

- *Mohó színezés:* Egy pont színének (egy állomás frekvenciájának) meghatározására a legegyszerűbb, de mégis viszonylag hatékony, és ezért gyakran használt eljárás; melynek során a megsorszámozott színek közül az új pont mindig a legkisebb lehetséges, – a szomszédok által nem használt – színt választja.

Hátránya, hogy a bejárás módja nagymértékben befolyásolja az algoritmus működését. Legjobb esetben, ha például egy, a kromatikus számmal történő színezés színosztályai szerint megy végig a pontokon, a mohó színezés optimális lesz. A legrosszabb esetek vizsgálatának széles körű irodalma van. Azt a legnagyobb k számot, amelyre létezik a G gráfnak olyan mohó színezése, amely pontosan k színt használ, a G gráf Grundy-számának (G(G)) nevezzük Elméleti kitekintés helyett lássunk példaként egy speciális páros gráfot, ahol a kromatikus szám n/4-szeresét is elérheti a felhasznált színek száma (n a gráf pontjainak száma), abban az esetben ha a legszerencsétlenebb sorrendben járjuk be a gráf pontjait.

Ez az arány Lovász Saks, Trotter [6] tétele alapján maximális, vagyis ennél szélsőségesebb példa már nem képzelhető el. Christen és Selkow [3] több további példát is mutattak a G(G)/c(G) arányának maximumára és más gráfszínezési paraméterek szélső értékeire.

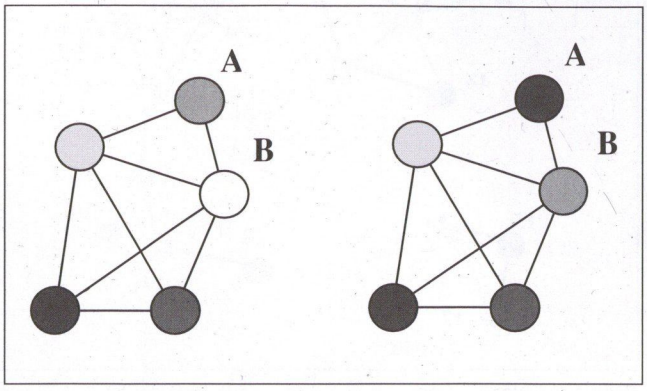
- Okos színezés: A lehetséges színek közül azt választja, amelyet a pont szomszédainak a szomszédai a legtöbben használnak, másodsorban pedig a legkisebb sorszámút. Az elgondolás lényege, hogy megpróbáljuk minél többször újrahasználni az addig felhasznált színeket. Az okos színezés technikai feltétele, hogy az állomásoknak tudniuk kell kommunikálni egymással.



4. ábra Az optimális és legrosszabb színezés esete páros gráf esetén

Javítás: Abban az esetben, amikor az aktuális pontnak új színt kellene választania, tehát az algoritmus által addig használt színek mindegyike megtalálható a pont szomszédai közt, akkor egy visszalépéses eljárás megvizsgálja, hogy lehet-e javítani. Javítani akkor tudunk, ha a szomszédok között van egy olyan színosztály, amelyben minden pont át tud térni egy lehetséges másik színre. Ha ez fennáll, akkor a pontunk megkaphatja a színváltó színosztály régi színét, annak pontjai pedig mindannyian egy másik, már felhasznált színt választanak; így nem kell új színt használnunk. Az 5. ábrán látható példában a B pontnak nem választható szín úgy, hogy az ne ütközzön már korábban beszínezett szomszédos pontok színeivel. (Feltételezzük, hogy csak az

ábrán látható négy szín áll rendelkezésünkre) Ekkor a javító eljárás szabályait követve az A pont áttér egy másik, a szomszédai által nem használt színre, mellyel lehetővé válik a B pont kiszínezése is.



5. ábra Példa a javító eljárás működésére

4. Eredmények a vizsgált gráfosztályok szerint

A fenti három szempont szerint az algoritmusok összes kombinációját elkészítettük (2×8×2=32 db algoritmus), majd különböző gráfosztályokból vett nagyszámú mintán futtattuk, és az átlagos színhasználatuk alapján rangsoroltuk őket, majd ezek alapján vontunk le következtetéseket a hatékonyságukra nézve. Először egyszerű véletlen gráfokat, majd ismert kromatikus számú véletlen gráfokat használtunk. Ezután a valóságot jobban modellező interferencia-gráfokat generáltunk, és ezeken vizsgáltuk az algoritmusok működését.

Véletlen gráfok:

Véletlen gráf: Adott pontszám és élbehúzási valószínűség megadásával generálódik.

Ismert kromatikus számú véletlen gráf: A gráf kromatikus számát előre beállíthatjuk.

Ezen gráfok generálása úgy történik, hogy a véletlen gráf pontjait adott számú (k) színosztályra bontjuk, a színosztályok közti éleket kitöröljük, majd hozzáadunk egy teljes k-s részgráfot, amellyel biztosítjuk, hogy a kromatikus szám pontosan k legyen.

Az eredmények mindkét gráfosztály esetén hasonlók, de az algoritmusok közti különbségek ismert kromatikus gráfok esetén számottevőbbek. A tesztek közül lássunk egy kiragadott és jellemző példát: A 32 algoritmus 1000 db ismert (5) kromatikus számú gráfon lefuttatva, amint a pontok száma 20 és 29 között egyenletesen, 100-as mintánként növekedett. A táblázatban szereplő értékek megadják, hogy hány esetben sikerült az adott algoritmusnak optimálisan kiszíneznie a gráfot, továbbá szerepel az is, hogy hány esetben használták a futás során az algoritmusok a javító eljárást:

Optimális színezések száma	Véletlen		Törlős		Fokszám		Globális	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	230	292	709	773	621	691	899	910
Javítós	663	663	878	889	827	903	924	927
Jav. száma	1921	1645	311	234	704	590	46	38

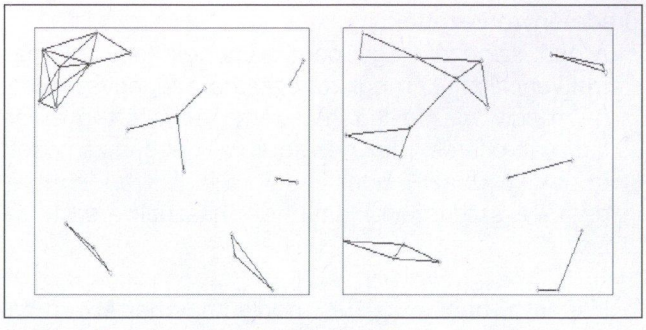
Optimális színezések száma	Szélességi		Mélységi		Foklokál		Törlokál	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	497	780	787	849	621	667	542	663
Javítós	818	856	873	910	804	815	764	829
Jav. száma	543	170	156	107	702	616	698	526

1. táblázat Teszteredmények egy ismert kromatikus számú gráfosztályon

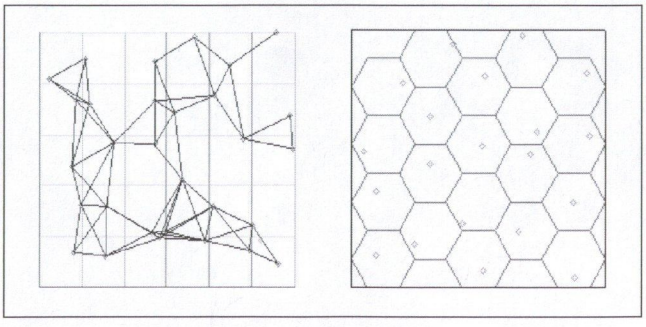
A legjobb algoritmusnak a teljes gráf ismeretében a „Globális okos javítós" változat bizonyult, míg lokális esetben a „Mélységi okos javítós" algoritmus. Az okos színezés minden esetben jobb, mint a mohó; a javító eljárás szerepe főként a kevésbé hatékony bejárások esetén nagy.

Interferencia-gráfok

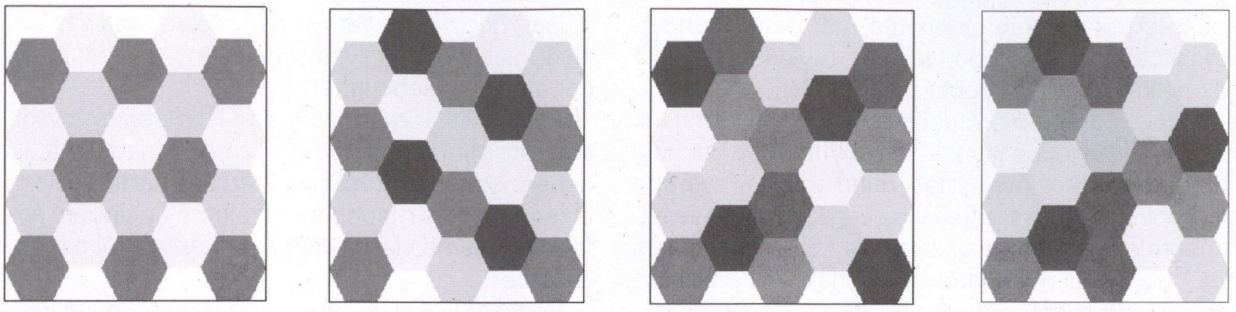
A valóságot jól modellező interferencia-gráfok korábban említett két legfontosabb tulajdonsága a viszonylag jó síklefedés és a véletlenszerűség. Az általunk generált öt osztály is e szempontok szerint halad a teljesen rendezetlen állapottól egészen az ideális síklefedést biztosító háromszögrács esetéig.



6. ábra Egyenletes és pontelhagyással módosított eloszlás



7. ábra Négyzetrács- és hatszögcellagráf

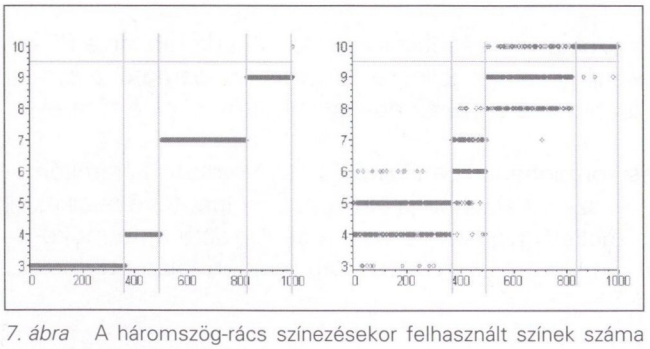


(c= a hatszög magasságának fele, C= a hatszög oldalának hossza, $K = \sqrt{(4.5C)^2 + c^2}$)

8. ábra A hatszögrács optimális színezése a kritikus távolság (d) változása mellett

a) *Egyenletes eloszlás* esetén: A pontokat egyenletes eloszlás szerint helyezzük el a síkon.
b) *Pontelhagyással módosított* eloszlás: Az egyenletesebb síklefedés érdekében, ha egy pont túl közel kerülne egy másikhoz, csak egy bizonyos valószínűséggel helyezzük el ott, amely valószínűség a már lerakott pontoktól távolodva egyenletesen nő.
c) *Négyzetrácsgráf:* Ebben a modellben egy négyzetrács celláiba teszünk egy-egy pontot egyenletes valószínűséggel. A síklefedés egyenletesebb, de még jelen van a rendezetlenség.
d) *Hatszögcellagráf:* Az előzőhöz hasonlóan, most hatszögcellákba helyezzük el véletlenül, egyenletes eloszlás szerint a pontokat (6., 7. ábra).
e) *Háromszögrács:* A bázisállomások a hatszögcellák közepébe kerülnek, vagyis egy szabályos háromszögrácsot alkotnak. Ez az utolsó eset abból a szempontból különösen érdekes számunkra, hogy ezen gráfosztályon a szükséges színek száma csak a hatszögek oldalhosszának és a kritikus távolságnak az arányától függ.

A törlős bejárást használó algoritmus legfeljebb 9 színt használt a színezéshez mind az 1000 gráf esetén (az ábrán vékony folytonos vonal a 10-es érték alatt), a véletlen bejárást használó algoritmus azonban sok esetben túllépte ezt a korlátot. A törlős bejárás esetén minden gráf az adott intervallumnak megfelelő optimális színszámmal színezhető, véletlen bejárás esetén a felhasznált színek számában jelentős szórás figyelhető meg, ezt mutatják az egymás felett elhelyezkedő ponthalmazok, melyek némely esetben párhuzamos vonalként jelentkeznek.



7. ábra A háromszög-rács színezésekor felhasznált színek száma Törlős és Véletlen bejárás esetén

A többi vizsgált interferencia-gráf esetén az eredmények az osztályoktól függetlenül mind hasonlóak lettek. Lássuk példaként egy négyzetrács gráfosztályon elvégzett tesztet, amelyben szintén 1000 gráfon futtattuk le mind a 32 algoritmust. A négyzetrácsgráf 6×5-ös, tehát minden esetben 30 pontból áll, a kritikus távolság pedig fut 1-től 2-ig, ahol 1 a négyzetcella oldalának hossza. A táblázatban a felhasznált színek átlagain túl szerepel az is, hogy az egyes algoritmusok összesen hányszor használták a javító eljárást.

Felhasznált színek átlaga	Véletlen		Törlős		Fokszám		Globális	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	5.368	5.252	4.680	4.677	4.867	4.789	4.712	4.709
Javítós	4.907	4.866	4.674	4.673	4.723	4.714	4.692	4.691
Jav. száma	1282	1007	12	9	385	241	32	28

Felhasznált színek átlaga	Szélességi		Mélységi		Foklokál		Törlokál	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	4.739	4.719	4.776	4.750	4.867	4.819	4.879	4.828
Javítós	4.700	4.696	4.716	4.707	4.744	4.734	4.770	4.752
Jav. száma	57	35	100	66	310	226	230	166

2. táblázat A háromszögrács színezésekor felhasznált színek száma Törlős (a) és Véletlen (b) bejárás esetén, ahol a vízszintes tengely pontjainak egy-egy lehetséges gráf felel meg (összesen 1000 db), amelyek mindegyikéhez hozzárendeljük a megfelelő (véletlen [b] vagy törlős [a]) bejárású algoritmus által a kiszínezésükhöz szükséges színek számát, miközben a kritikus távolság egyenletesen nő. Itt a függőleges tengelyre kellene írni, hogy a szükséges színek száma, a vízszintes tengely alá pedig: a gráfok száma

Bár a bonyolultabb algoritmusok közti különbségek a fenti interferencia-gráfok esetén már nem olyan látványosak, a legegyszerűbb, véletlen bejárást használó javítás nélküli esethez képest továbbra is javulás érhető el. A teljes gráf ismeretében működő algoritmusok között a „Törlős okos javítós" bizonyult a legjobbnak, de itt már mind a színválasztás, mind a javítás szerepe elhanyagolható. A lokális, tehát az eredeti feltevéseinknek megfelelő algoritmusok között a „Szélességi okos javítós" bizonyult a leghatékonyabbnak. A színválasztás és a javítás szerepe érzékelhető.

Felvetődik a kérdés, hogy mi lehet az oka annak, hogy a Törlős algoritmus interferencia-gráfok esetén a legjobban közelíti meg az optimumot. A hipotetikus magyarázat erre a kérdésre, hogy a Törlős algoritmus során először az interferencia-gráf szélén lévő kis fokszámú pontokat hagyjuk el, és fokozatosan megyünk a síkrész belseje felé, míg a legvégén a színezés szempontjából kritikus csomók, klikkek maradnak. A színezés során tehát először beszínezzük a kritikus pontokat, a klikkeket és a nagyfokú pontokat, majd a végén a szélső pontokat már egyszerűen, a javító eljárás használata nélkül tudjuk kiszínezni. Sajnos ennek lokalizált változata már kevésbé hatékony, és ebben az esetben a bonyolultabb és az állomások között több kommunikációt igénylő irányított szélességi bejárás a jobb.

Összefoglalás

Munkánk során tehát sikerült olyan algoritmusokat alkotnunk, melyek az adott feltételek mellett a valóságot jól közelítő modelleken egyszerűen, gyorsan és az optimálist vélhetően jól közelítő eredményt adnak. Az algoritmus kiválasztásakor ugyanakkor nem szabad megfeledkeznünk arról sem, hogy mind az állomásokba beépített technológiai megoldásoknak, mind a köztük lévő kommunikációnak ára van. Épp ezért előfordulhat, hogy a tényleges megvalósításkor egy egyszerűbb, olcsóbb, de kevésbé hatékony változat összességében előnyösebb lehet.

Megemlítjük a témakör egy lehetséges továbbgondolását. Minthogy a rádióhullámok terjedése a háromdimenziós térben történik, ezért az interferencia-gráfok is természetes módon elképzelhetők a térben, amely a fenti probléma vizsgálatának szó szerint új dimenzióját jelentheti.

Végezetül szeretnénk köszönetet mondani Rácz Andrásnak (Traffic Lab. Ericsson), aki konzulensként két éven keresztül nagy segítségünkre volt munkánkban, melyet az ún. témalabor tárgy keretében végeztünk el.

Irodalomjegyzék

1. Biró Péter, Lója Krisztina, Sütő Márton. Elosztott frekvenciakiosztási algoritmusok tervezése és vizsgálata. (TDK-dolgozat) http//:www.math.bme.hu/~biro/tdk
2. Bollobás B. The chromatic number of random graphs Combinatorica, vol. 8, 49–55, (1988).
3. C. A. Christen, S. M. Selkow. Some perfect colouring properties of graphs. J. Combin. Theory Ser. B 27, 49–59 (1979).
4. R. Graham, M. Grötschel, L. Lovász. Handbook of Combinatorics 4.fejezet (1995).
5. T. R. Jensen, B. Toft. Graph Coloring Problems John Wiley and Sons, New York, (1995).
6. L. Lovász, M. Saks, W. T. Trotter. An on-line graph coloring algorithm with sublinear performance ratio. Discrete Math. 75, 319–325, (1989).
7. R. A. Murphey, P. .M. Pardalos, M. G. C. Resende. Frequency assignment problems. Handbook of Combinatorial Optimalization, Kluwer Academic Publishers, (1999).

Hír

Az Ericsson 3G készülékplatformokra vonatkozó szerződést kötött az LG Electronicsszal. Az Ericsson technológiáján alapuló saját, értéknövelt GPRS/UMTS kettős üzemmódú 3G készülékeket fejleszt, és a technológiáért licencdíjat fizet.

A mobilkészülék-ipar jelenleg átalakulás alatt áll. Ugyanazok a változások zajlanak le, mint korábban a PC-iparban: elmozdulás a teljes termékválasztékot szállító néhány vállalat jellemezte vertikális iparágtól a számos specializált vállalatot magában foglaló horizontális iparág felé, ahol az egyes vállalatok csak a termék bizonyos alkotórészeit állítják elő.

A technológiaplatformok magukban foglalják az összes komponens specifikációját, a nyomtatott áramköri kártyák vázlatát és a szoftvert, a gyártó pedig saját igényei szerint alakíthatja át azokat. Ez lehetővé teszi az LG Electronics számára, hogy minimális K+F beruházás mellett gyorsan vezessen be a piacra új termékeket, ugyanakkor jelentős készülékszállítóként lépjen fel, amikor az ügyfelek a világ minden táján áttérnek a 3G-re.

Vezeték nélküli hálózat teljesítményvizsgálata

JUHÁSZ ÁKOS–ULRICH FERENC–DR. EGED BERTALAN–KUBINSZKY FERENC

Budapesti Műszaki és Közgazdaság-tudományi Egyetem
Széles sávú Hírközlő Rendszerek Tanszék
Vezeték nélküli Információtechnológia Laboratórium

Az internetes technológiák fejlődésének köszönhetően egyre több felhasználó kapcsolódik lokális vagy globális hálózatokhoz. A vezetékes hálózatok megbízhatóságuk és a nagy sávszélesség miatt mára széleskörűen elterjedtek. Azonban a technikai fejlődés az elmúlt években más irányt vett. A felhasználók új igénye a hálózatokkal szemben: a mobilitás. A vezetékes hálózatok a fix kábelezés miatt ezt már nem tudják biztosítani. Az új igényeknek csak mobil technológiákkal lehet megfelelni. Azonban az alkalmazott vezeték nélküli eszközök drágábbak, ezért a telepítési költségek magasabbak, mint a vezetékes hálózatoknál.

Bevezetés

Egy stabil és megbízható WLAN (vezeték nélküli lokális hálózat) kiépítéséhez gondos tervezés szükséges. Az átvitel során számos probléma léphet fel az alkalmazott nagy frekvenciás rádiócsatorna miatt. A zajos csatorna vagy a zsúfolt beltéri környezet jelentősen növelheti a hibavalószínűséget. A tervezőnek ismernie kell a WLAN rendszerek lehetőségeit és korlátait. Munkánk során egy WLAN rendszer működését vizsgáltuk különböző körülmények között.

IEEE 802.11 szabvány áttekintése

Az IEEE 802.11 szabvány az ISM (2.4 – 2.483GHz) sávot jelöli ki távközlési csatornaként. Az maximális adóteljesítmény országonként változó, így pl. Európában 1mW, USA-ban 1W. A keskeny sávú interferenciák kiszűrésére a szabvány szórt spektrumú modulációt ír elő. Ez lehet frekvenciaugratásos (FHSS – Frequency-Hopping Spread Spectrum) és direkt szekvenciális (DSSS – Direct Sequence Spread Spectrum) is. A leggyakrabban használt eszközök direkt szekvenciális szórt spektrumú modulációt alkalmaznak. A szabvány 11 chipes Barker kódot definiál a spektrumszóráshoz. Különböző modulációk alkalmazásával eltérő adatsebességek érhetők el. A szabvány DBPSK, DQPSK, illetve CCK modulációt definiál. Ezekkel 1, 2, 5, 11 Mbs-os adatsebesség érhető el.

A 83 MHz széles ISM sávban a szabvány 13 csatornát definiál, 5 MHz csatornatávolsággal. Mivel a jel sávszélessége a csatornában 22 MHz, így a szomszédos csatornák között jelentős átfedés van spektrumban. A zavarmentes kapcsolathoz a szabvány minimum 3 csatorna távolságot ír elő azonos területen lévő párhuzamos kommunikációkhoz.

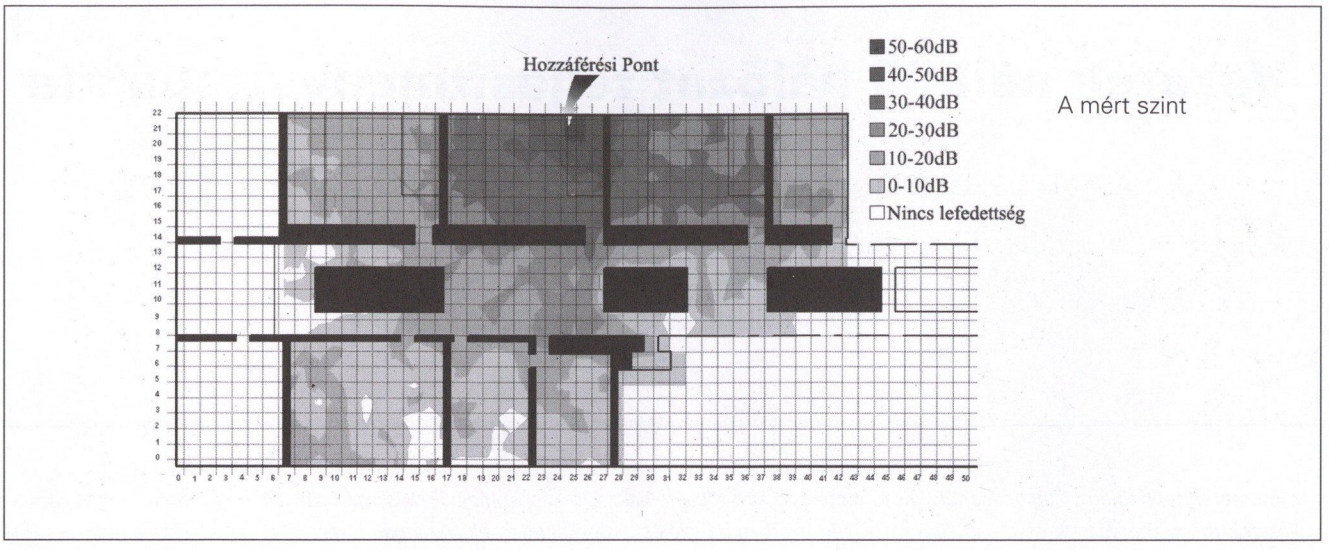
A vezetékes hálózatokban a csatorna-hozzáférést CSMA/CD (Carrier Sense, Multiple Access with Collision Detection) ütközéselkerülő algoritmussal szabályozzák. Ennek a protokollnak az alkalmazása vezeték nélküli hálózatokon nehézkes, mert:

1. Az algoritmushoz full duplex rádió szükséges.
2. Az algoritmus működésének elengedhetetlen feltétele hogy minden állomás képes legyen venni a többi jeleit.

Az említett problémák kiküszöbölésére a 802.11 szabványütközés-elkerüléses (CSMA/CA; Carrier Sense, Multiple Access with Collision Avoidance) algoritmust definiál pozitív nyugtával kiegészítve.

1. Az adóállomás az adás megkezdése előtt egy előre definiált ideig (DIFS, Distributed Inter Frame Space) figyeli a csatornát. Amennyiben a csatornát szabadnak érzékelte, megkezdi adását, különben késlelteti a csomag küldését.
2. A vevőállomás ellenőrzi a csomag CRC kódját, majd nyugtát küld, ha helyesnek találta. Amennyiben nem érkezik nyugta az adóállomáshoz, a csomagot újraadják.

A mérési eredmények egy WaveLANTM vezeték nélküli hálózati kártyákból felépülő hálózatra vonatkoznak. Ezen kártyák támogatják a 802.11 szabvány által definiált ad-hoc és infrastruktúra-üzemmódot is. Egy ad-hoc hálózatban az állomások közvetlenül egymással lépnek kapcsolatba a már említett csatorna-hozzáférési eljárásokat alkalmazva. Amennyiben a kommunikálni kívánó állomások egymás hatósugarán kívül kerülnek, lehetőség van további állomások bevonására a kommunikációba, melyeknek csomagtovábbító és útvonalválasztó szerepük lehet. Ez az üzemmód nagyfokú szabadságot biztosít a felhasználónak, mivel a hálózat



1. ábra Lefedettségi térkép

minden helyzetben dinamikusan követi a topológia változását.

A felépülő hálózat infrastruktúrája cellás rendszerű. Minden cella felügyeletét és vezérlését egy hozzáférési pont (Access Point) végzi. A cellában tartózkodó állomások csak a hozzáférési ponttal kommunikálnak. A hozzáférési pontok összeköthetők egy elosztóhálózattal, így bővíthető a lefedett terület mérete. A szabvány definiálja a vándorlás fogalmát is, azonban a folyamatot nem rögzíti pontosan, csak eszközöket ad a könnyebb megvalósításhoz. Egy vándorló állomás a cellahatárhoz érve csatlakozhat a következő cellát vezérlő hozzáférési ponthoz, így a kommunikáció nagyobb lefedettségi területen biztosítható.

Mérési körülmények

Méréseink során a vezeték nélküli hálózatok teljesítményét vizsgáltuk mind zajos, mind ideális csatornán. Mivel csak a rádiós hálózat okozta hatások iránt érdeklődtünk, ezért minden esetben csupán két állomásból álló hálózat viselkedését vizsgáltuk. A fellépő hibákat eltérő zajosságú környezeteken vizsgáltuk különböző topológiájú hálózatokban. A méréseket elvégeztük mind ad-hoc, mind infrastruktúra-alapú hálózatban. A maximális értékek méréséhez ideális csatornát alakítottunk ki, minimális interferenciára törekedve.

A teszteket Linux operációs rendszer alatt végeztük. Az eldobott csomagok számát rendszerfájlokból olvastuk ki. A jel/zaj viszony értékeket az alkalmazott eszközök meghajtóprogramja biztosította. Azonban a mérési eredmények értékelésekor figyelembe kell venni, hogy az eszközök ezt az értéket nem mérik, hanem számolják. A szükséges adatokat tartalmazó rendszerfájlokat csak csomagok küldésekor vagy érkezésekor frissítik, ezért minden méréshez forgalmat kellett generálni a csatornában. A mérési eredmények rögzítéséhez C nyelven írt programokat alkalmaztunk, a mentett információt pedig Excel táblázatkezelővel dolgoztuk fel. Netperf 2.2 alkalmazás segítségével generáltunk TCP ill. UDP forgalmat a mérések elvégzéséhez.

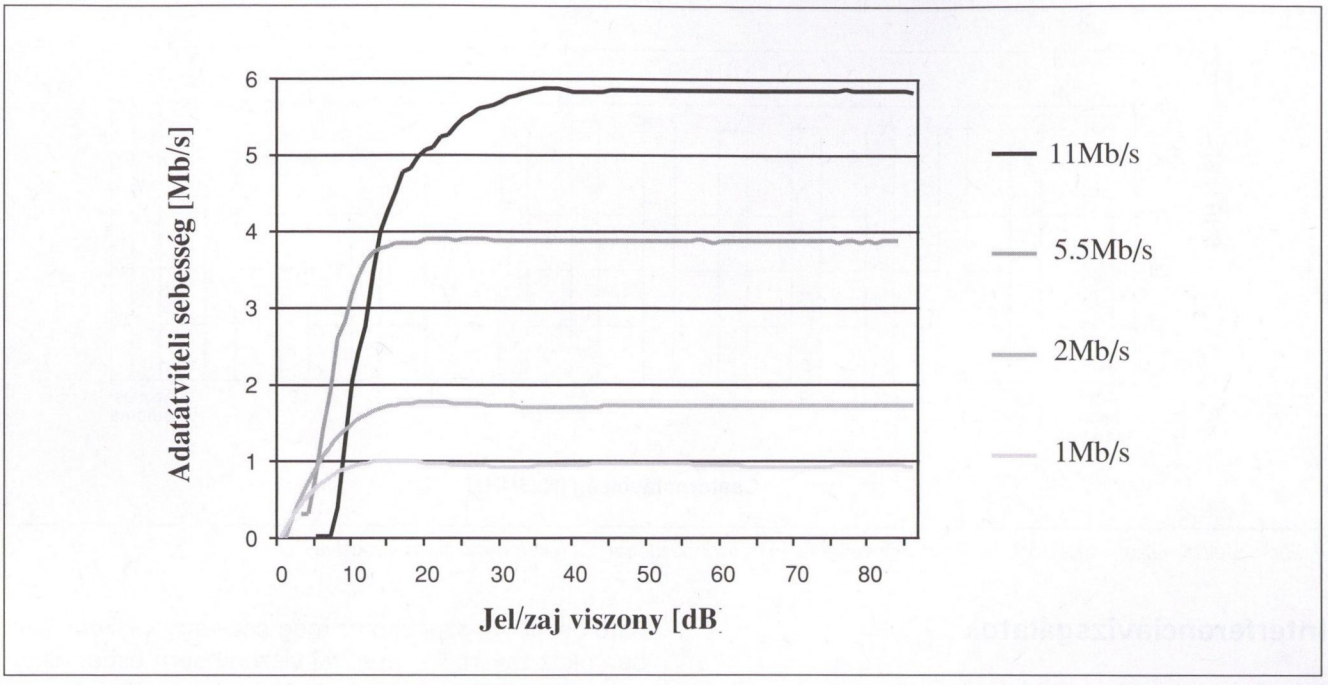
Lefedettségi mérések

A maximális beltéri hatótávolság vizsgálatához egy infrastruktúra-alapú hálózatot használtunk. Az elrendezés két állomásból épült fel. Az egyik állomás Ethernet hálózaton csatlakozott a hozzáférési ponthoz, míg a másik a vezeték nélküli hálózaton keresztül. A mérési eredményekből következtetéseket vonhatunk le a beltéri kialakítástól függő terjedési viszonyokra is.

A jel/zaj viszony mérését 1x1 méteres felbontásban végeztük el a tanszék területén, majd elkészítettük a teljes lefedettségi térképet. A hozzáférési pont az ábrán jelölt helyen a laboratóriumunk területén került elhelyezésre.

A lefedettségi térképet vizsgálva megállapíthatjuk, hogy egy átlagos beépítettségű beltéri területen az átlagos csillapítás 4.3dB méterenként, a maximális beltéri hatótávolság pedig 25 m. Természetesen ezen értékek nagymértékben változhatnak más környezetben a beépítettségtől és az épületben használt anyagoktól függően. Mindettől függetlenül az ábra látványosan szemlélteti a különböző anyagok okozta csillapításokat. Például a laboratóriumunkban lévő üvegfal-csillapítás még részletesebb ábrán sem vehető észre, míg a térelválasztó falak hatása már ezen az ábrán is szembeötlő. Az ajtóknál is megfigyelhető ugyanez a hatás. A gyengébb lefedettségi területeken pedig jól megfigyelhető a hirtelen emelkedő térerő, mely a reflexiók hatása. Az ISM sávban az egyik legnagyobb csillapítása a víznek van. Mivel az emberi test igen nagy része víz, így az emberek is jelentős csillapítást okozhatnak. A maximális értékek méréséhez a méréseink időpontját úgy választottuk meg, hogy a vizsgált területen kevés személy tartózkodjon.

A lefedettségi térkép hasznos információkat szolgáltat, melyek egy cellás rendszer kialakításakor nagy



2. ábra Adatátviteli sebesség a jel/zaj viszony függvényében ad-hoc üzemmódban

segítséget nyújthatnak a hozzáférési pontok ideális pozíciójának meghatározásakor.

Átviteli mérések

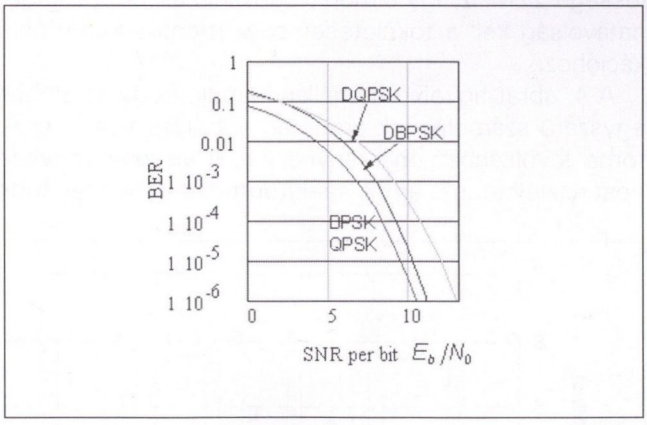
A kommunikáló állomások közötti távolságot vagy a csatorna zajosságát változtatva a jel/zaj viszony is változik. Meghatározott jel/zaj viszony alatt a hibavalószínűség is emelkedni kezd, ezzel növelve a csomagvesztés valószínűségét. Nagyobb adatátviteli sebességet biztosító üzemmódok esetén a fejlécet lassabban továbbítja, mivel ezek kevésbé érzékenyek a zaj hatásaira. Így biztosítható, hogy a csomag átviteléhez szükséges információk hibavalószínűsége kisebb legyen.

A jel/zaj viszony változásának az adatátviteli sebességre gyakorolt hatását vizsgáltuk ezen mérések elvégzésével. A méréseket elvégeztük minden elérhető adatátviteli sebességen mind ad-hoc, mind infrastruktúra üzemmódban. A jel/zaj viszony csökkentéséhez a kommunikáló állomások között növeltük a távolságot.

A 2. ábráról leolvasható a különböző adatátviteli sebességekkel elérhető maximális adatátviteli sebesség. Nagyobb adatátviteli sebességeken nagyobb jel/zaj viszony szükséges a konstans adatátviteli sebesség eléréséhez.

Megfigyelhető, hogy a valóságban mért maximális adatátviteli sebességek jóval kisebbek a névleges adatátviteli sebességeknél. A csomagfejléc és a szabványban definiált várakozási idők a csomagok között mind negatív hatásként jelentkeznek. Nagyobb adatátviteli sebességen a nagyobb veszteség a már említett fejlécvédelmi eljárások hatása. Például 11 Mbs-on a maximális adatátviteli sebesség mindösszesen ~5.9 Mbs. Ennek oka az, hogy a csomagközi várakozások és a fejlécek küldési ideje közel megegyezik egy 1500 bájtos csomag küldési idejével.

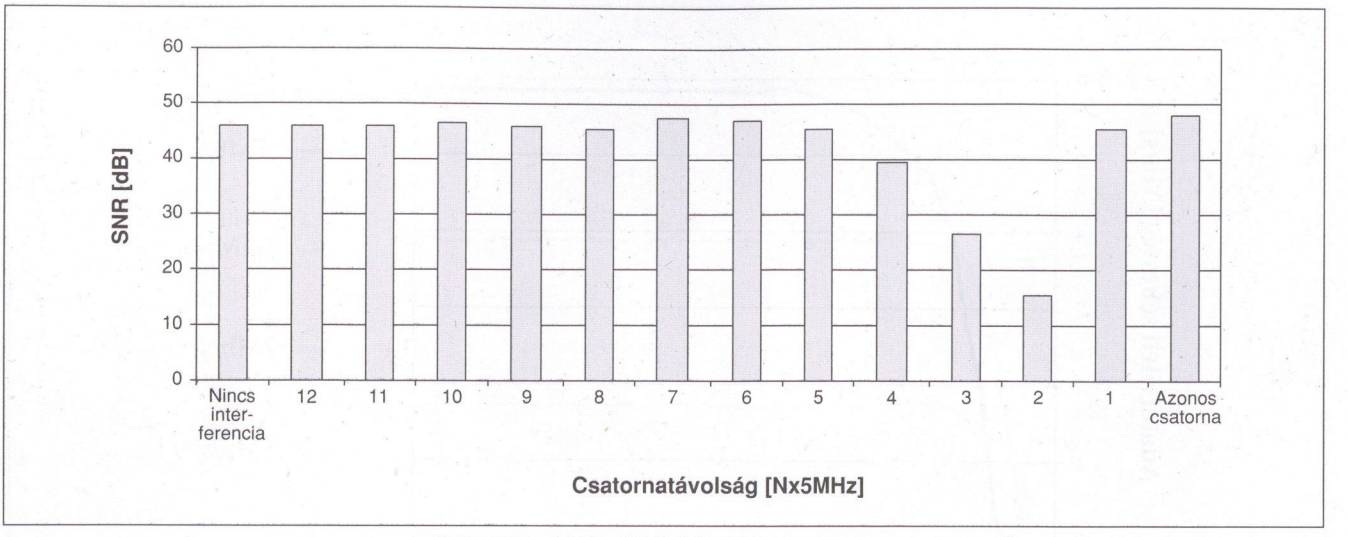
Megfigyelhető, hogy a görbék bizonyos pontokon metszik egymást. A BPSK és QPSK modulációhoz tartozó elméleti BER (Bit Error Rate) görbék ugyanezt a tulajdonságot mutatják. Ennek oka az, hogy a jel/zaj viszony csökkenés következtében csökkenő adatátviteli sebesség a csomagvesztések eredménye. A csomagvesztést viszony a bithibák gyakorisága befolyásolja (3. ábra).



3. ábra

Definiálva a minimálisan szükséges adatátviteli sebességet, meghatározható a minimális jel/zaj viszony. Ezt összevetve a lefedettségi térképpel kaphatjuk meg a hozzáférési pontok optimális helyét egy cellás rendszerű vezeték nélküli hálózatban.

Az adatátviteli méréseket elvégeztük infrastruktúra-alapú hálózatban is. A mérési eredmények ebben az esetben megközelítőleg megegyeznek az ad-hoc mérési eredményekkel, azonban a maximális adatátviteli sebességek minden üzemmódban néhány százalékkal kisebbek, mivel a hozzáférési pontban a csomag késleltetést szenved.



4. ábra Jel/zaj viszony a csatornatávolság függvényében két azonos időben zajló kommunikációt vizsgálva

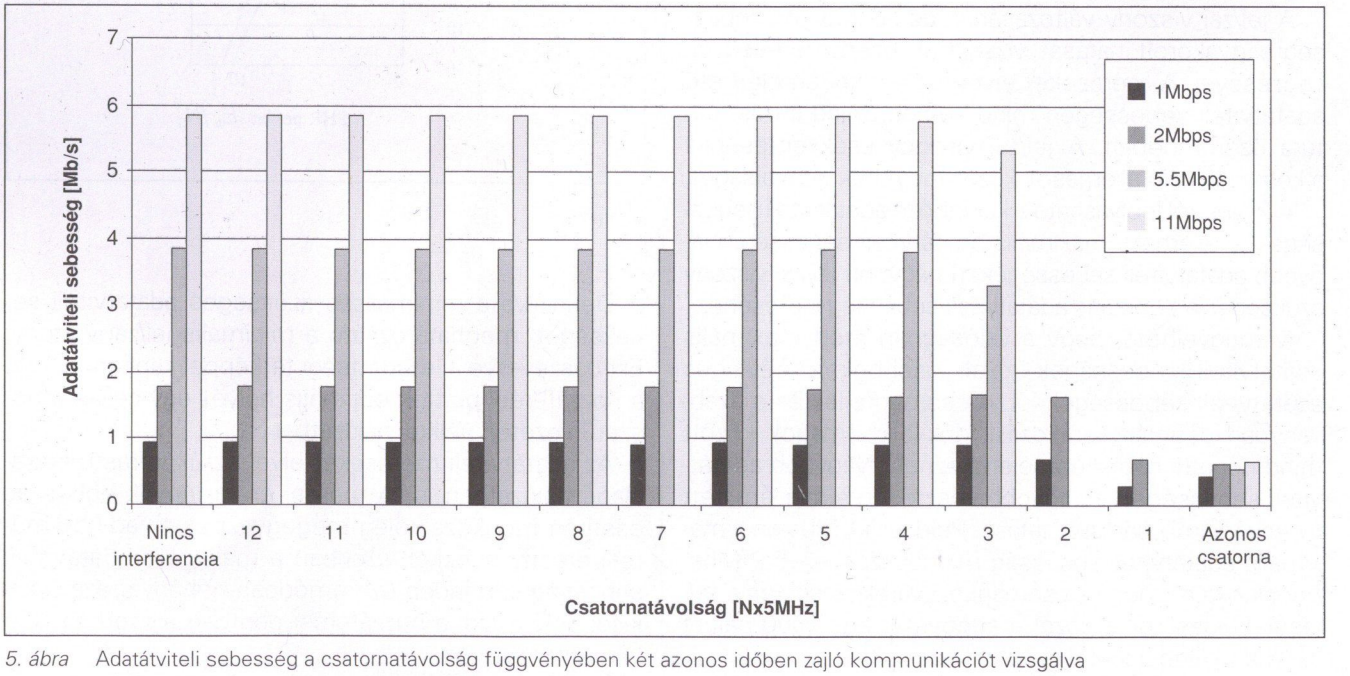
Interferenciavizsgálatok

A szabvány ismertetésekor megállapítottuk, hogy a szomszédos csatornák közti spektrumbeli átfedés következtében a definiált 13 csatorna nem független. Amennyiben ezt nem vesszük figyelembe, és két összeköttetést túl közeli csatornákra helyezünk, interferencia keletkezik. A mérések során ezen interferenciának a hatását vizsgáltuk.

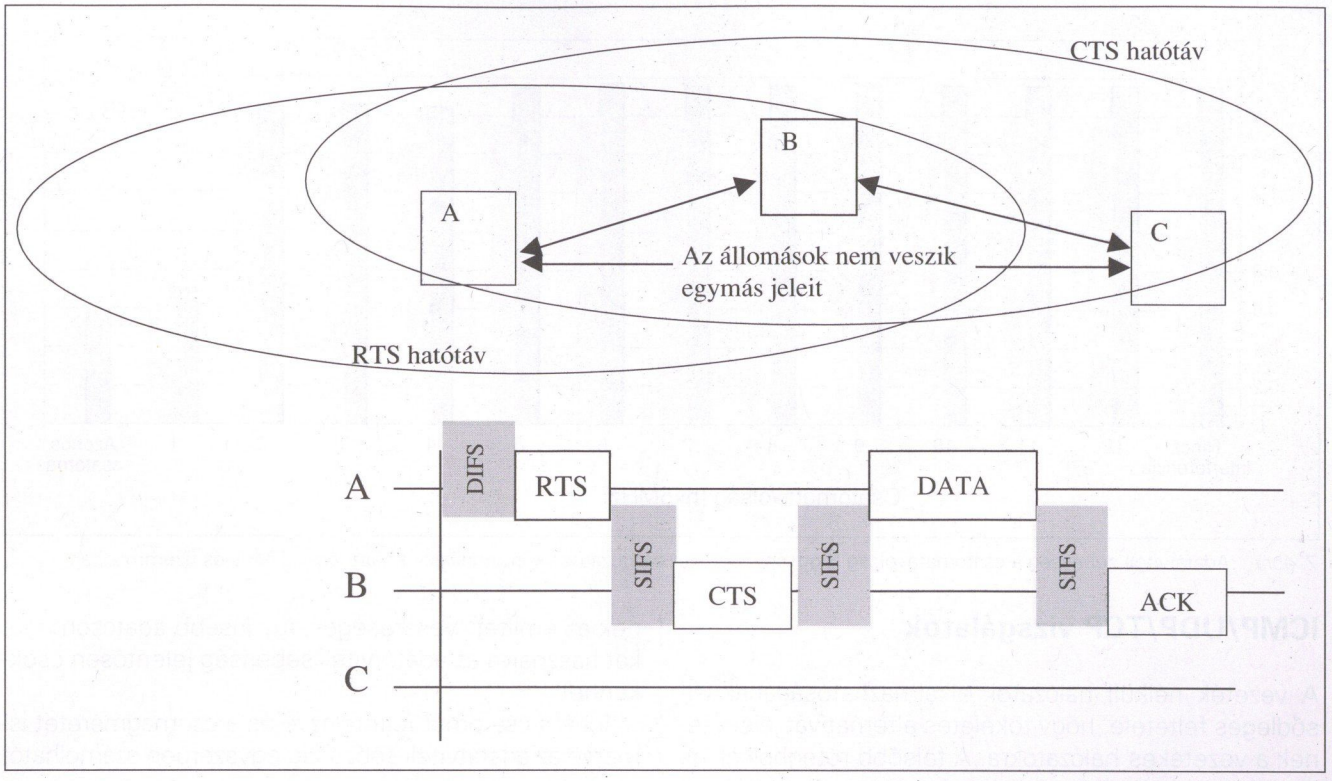
Az adatokat ismerve gyors számolást végezhetünk. Mivel a csatornatávolság 5MHz és a csatornák sávszélessége 22MHz, így elméletileg minimálisan 5 csatornatávolság kell a tökéletesen zavarmentes kommunikációhoz.

A 4. ábrát figyelve megállapíthatjuk, hogy az előbbi egyszerű számolásunk eredménye helyes volt. 4 csatorna távolságban már enyhe jel/zaj viszony csökkenést észlelhetünk, ami a spektrumbeli átfedésnek tudható be. Mivel azonban ez még csak egy keskeny sávban okoz zavart, így a jel/zaj viszony sem eshet nagymértékben. Közelebbi csatornára lépve a jel/zaj viszony tovább esik. Érdekes pontja az eredménynek, hogy azonos és szomszédos csatornán kommunikálva a jel/zaj viszony újra zavartalan. Ezt az eredményt akkor érthetjük meg, ha az adatátviteli sebességet is megvizsgáljuk a csatornatávolság függvényében.

Az 5. ábrát figyelve megállapíthatjuk, hogy a 4 csatorna távolságban jelentkező enyhe jel/zaj viszony-csökkenés egyik üzemmódban sem okoz számottevő adatátvitelisebesség-csökkenést. Ezt a spektrumszórásnak köszönhetjük, mely hatékony védelmet nyújt a keskeny sávú zavarok ellen. 3 csatorna távolságban a zavarás mértéke már olyan jelentős, hogy az adatátviteli sebesség is jelentősen csökken. Láthatjuk, hogy a kisebb adatátviteli sebességek zavarérzékenysége interferencia mellett is stabilabb és megbízhatóbb kommunikációt ígér.



5. ábra Adatátviteli sebesség a csatornatávolság függvényében két azonos időben zajló kommunikációt vizsgálva



6. ábra A rejtett terminál probléma és az RTS/CTS üzemmód

Azonos vagy szomszédos csatornára helyezve a kommunikációt, az adatátviteli sebesség újra emelkedik. Sajnos a mérések során 5.5 és 11 Mbs-os sebességen nem tudunk eredményekkel szolgálni, mivel csak két megfelelő eszközünk volt ilyen sebességekre. Azonos csatornán kommunikálva életbe lép a CSMA/CA ütközéselkerülő algoritmus, minek következtében valós interferencia nem jelentkezik a csatornában. A csomagok nem ütköznek, ám a csatornát két kapcsolat használja, így az adatátviteli sebesség csökken (azonos üzemmódot használva mindkét csatorna hozzávetőlegesen feleződik), ám ez jel/zaj viszonyban nem jelentkezik. Szomszédos csatornán kisebb adatátviteli sebességeken a kis csatornatávolság miatt ugyanez a jelenség tapasztalható.

RTS/CTS mechanizmus

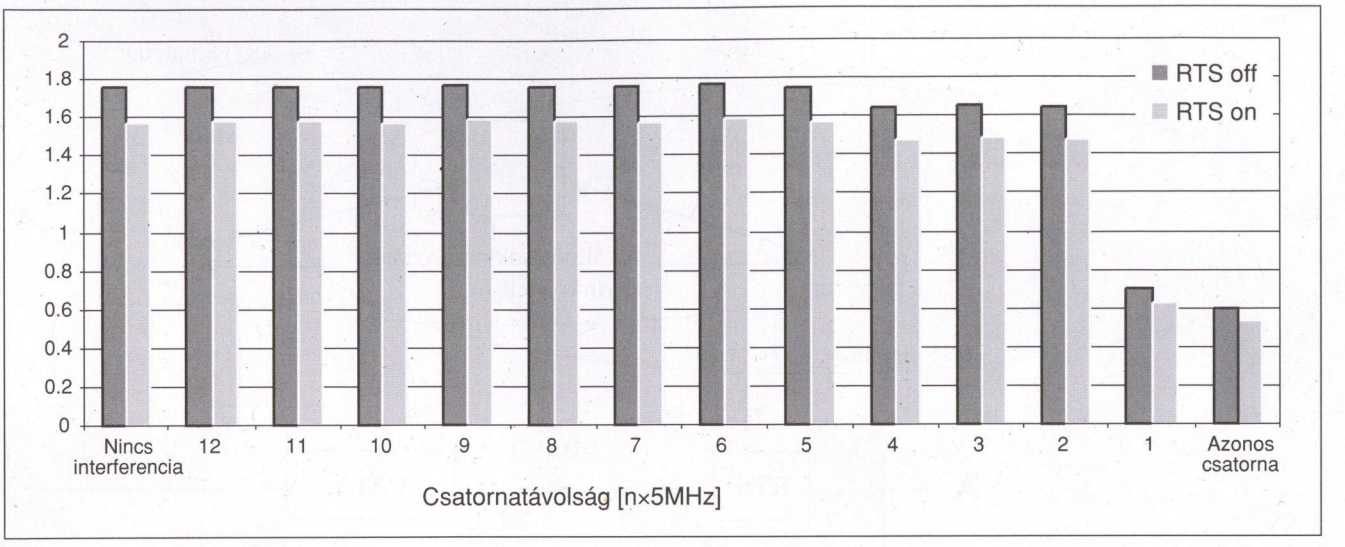
A CSMA/CA protokoll segítségével megoldható volt, hogy half duplex adókat alkalmazva is biztosítsuk az ütközésmentes átvitelt, ám ez a protokoll is igényli, hogy minden állomás érzékelje a másik adását. Vannak olyan topológiák, ahol azonban ez nem biztosított. Az egyik ilyen elrendezés a rejtett terminál. Ebben az elrendezésben a két adni kívánó állomás nem képes venni egymás jeleit, így mindketten szabadnak érzékelik a csatornát és adni kezdenek, ám ez a vevő állomásnál csomagütközéshez vezet. A rejtett terminál probléma kiküszöböléséhez a szabvány definiálja a RTS/CTS mechanizmust.

A már említett rejtett terminál elrendezésben nevezzük a két adni kívánó állomást A-nak és B-nek. Legyen C a vevőállomás, mely mindkét adó jelét képes érzékelni. Tételezzük fel, hogy A állomás adást kezdeményez. Mivel szabadnak érzékeli a csatornát, küld egy RTS (Request To Send) csomagot a vevőállomásnak. Ebben a csomagban jelzi hogy mennyi időre lenne szüksége a csomagátvitelre. A vevőállomás az RTS csomagra válaszul egy CTS (Clear To Send) csomaggal válaszol, melyben szintén közli a csatornafoglalás idejét. Mivel az RTS és CTS csomag egyikét biztosan veszi mindegyik állomás, melyeknek adása csomagütközést eredményez, így mindegyikhez eljut az információ, hogy adott ideig foglalt a csatorna. Ezen állomások akkor sem kezdeményeznek adást, ha ezek után szabadnak érzékelik a csatornát. A csomag vételét a vevőállomás egy ACK (Acknowledge) csomaggal nyugtázza.

DIFS (Distributed Inter-Frame Space) az az idő, amennyi ideig szabadnak kell érzékelnie egy állomásnak a csatornát, mielőtt kommunikációt kezdeményezne. SIFS (Short Inter-Frame Space) idő rövidebb, mint a DIFS és a vevőállomás használja a csomag vétele és a nyugta küldése között. Azért van szükség két eltérő hosszúságú idő használatára, hogy biztosítva legyen, hogy más állomás nem kezdhet adást a nyugta küldése előtt.

Amennyiben mindhárom állomás képes venni a másik kettő jeleit, akkor az RTS/CTS mechanizmus használata szükségtelen többletinformációval jár, mely tovább csökkenti az adatátviteli sebességet.

A 7. ábra az említett adatátvitelisebesség-csökkenést mutatja. Mivel az RTS és CTS csomagok küldési sebessége csak 1, ill. 2 Mbs lehet. Az ábráról leolvasható hogy az adatátvitelisebesség-csökkenés, amit az RTS, CTS és ACK csomagok okoznak, 2 Mbs-t használva hozzávetőlegesen 10%.



7. ábra Adatátviteli sebesség a csatornatávolság függvényében két párhuzamos kommunikációt használva, 2 Mbs-os üzemmódban

ICMP/UDP/TCP vizsgálatok

A vezeték nélküli hálózatok alkalmazhatóságának elsődleges feltétele, hogy tökéletes alternatívát jelentsenek a vezetékes hálózatokra. A felsőbb rétegbeli alkalmazások számára nem tűnhet fel, hogy a kapcsolatot vezeték nélküli hálózat biztosítja. Éppen ezért a következőkben a fix hálózatokra kidolgozott adatátviteli protokollok működését elemezzük mobil linken.

Ahogy már korábban is említettük, a csomag vezérlőinformációit minden esetben alacsonyabb adatátviteli sebességen kell küldeni. Az 1. táblázat mutatja a szükséges csomagkomponenseket, és a hozzájuk tartozó küldési időket különböző adatátviteli sebességeken.

Az üzemmódok nevében lévő adatátviteli értékek a fizikai rétegbeli adatátvitelt jelentik. Mint láthattuk, a harmadik rétegben az adatátviteli sebesség jóval kisebb is lehet. Azonban a már említett fejlécek és várakozási idők okozta veszteségen kívül más is befolyásolhatja az adatátviteli sebességet. Például a csomagméret. Mivel csomagmérettől függetlenül jelentkeznek az említett veszteségek, így kisebb adatcsomagokat használva az adatátviteli sebesség jelentősen csökkenhet.

Ideális csatornát feltételezve és a csomagméretet ismerve az adatátviteli sebesség egyszerűen számolható:

$$T_{put} = \frac{1}{t_{data} + t_{ovehead} + t_{backoff}}$$

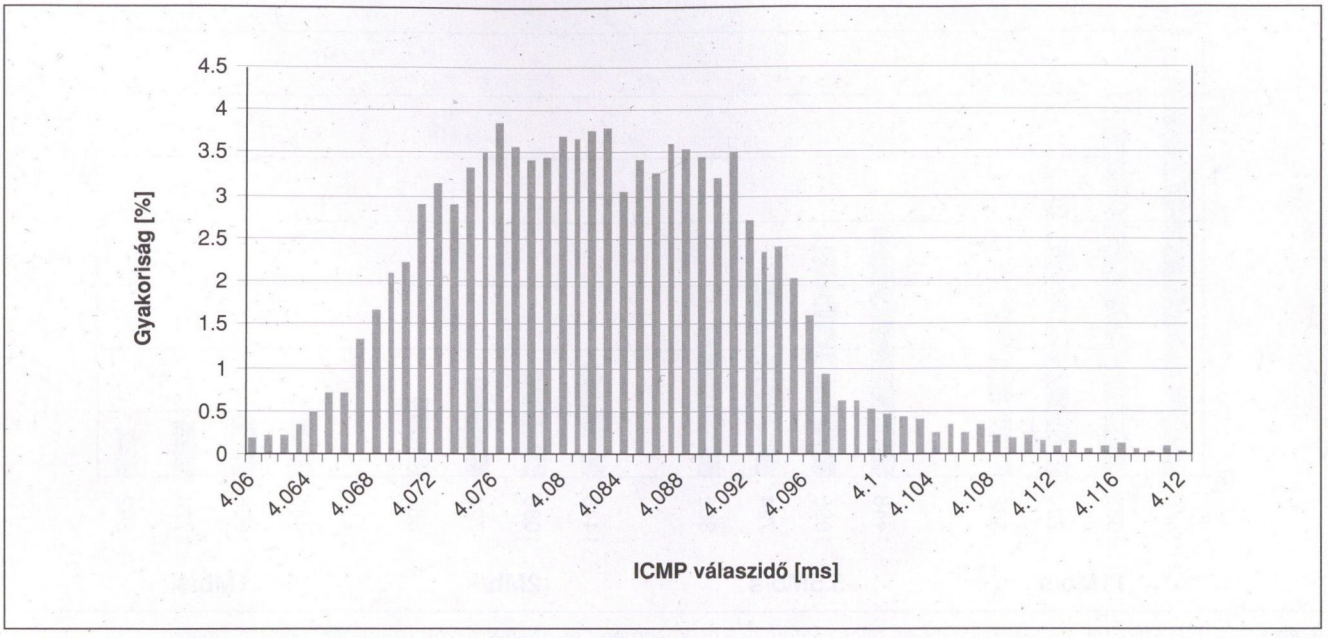
ahol:

t_{data} az adat továbbításához szükséges idő
$t_{overhead}$ a fejléc továbbításához szükséges idő
$t_{backoff}$ a CSMA/CA protokoll által használt véletlen „backoff" idő

Az alkalmazott adatátviteli üzemmód és a csomagméret meghatározza a t_{data} időt. A fent adott táblázatból kiolvasható a fejlécek továbbításához szükséges idő. Az egyetlen probléma a képlettel a backoff idő. A backoff idő mérésére a mi eszközeinkkel nem volt lehetőség, azonban ICMP (Internet Control Messagge

Paraméter	Byte	11 Mb/s	5.5 Mb/s	2 Mb/s	1 Mb/s
UDP/IP fejléc	28	20,4μs	40,7μs	112μs	224μs
Mac fejléc					
SNAP	8	5,8μs	11,6μs	32μs	64μs
MPDU/FCS	34	24,7μs	49,5μs	136μs	272μs
ACK	14	56μs	56μs	56μs	112μs
RTS	20	80μs	80μs	80μs	160μs
CTS	14	56μs	56μs	56μs	112μs
Fizikai réteg fejléc					
PLCP	24	192μs	192μs	192μs	192μs
IFS fejléc					
DIFS		50μs	50μs	50μs	50μs
SIFS		10μs	10μs	10μs	10μs

1. táblázat Csomagelemek és küldési idők az adatátviteli sebesség függvényében



8. ábra Leolvashatjuk, hogy az átlagos válaszidő 4080 us. Ebbe természetesen beletartozik az ISMP echo-request és az ICMP echo-reply csomag is

Protocol) üzenetek segítségével a várható értéke és a szórása becsülhető. A 8. ábrán az ICMP csomagok válaszidejének eloszlását ábrázoltuk.

Az UDP protokollt speciális valós idejű alkalmazásokhoz fejlesztették. UDP protokollt olyan alkalmazások használnak csak, melyeknél nem okoz problémát néhány csomag elvesztése.

	csomagméret	számolt sebesség	mért sebesség	különbség
	1448 byte	6,73 Mbps	6,13 Mbps	8,92%
	1086 byte	5,96 Mbps	5,34 Mbps	10,4%
	724 byte	4,85 Mbps	4,23 Mbps	12,78%
11Mbps	362 byte	3,11 Mbps	2,62 Mbps	15,76%
	1448 byte	4,1 Mbps	3,9 Mbps	4,88%
	1086 byte	3,78 Mbps	3,58 Mbps	5,29%
	724 byte	3,27 Mbps	3,01 Mbps	7,95%
5.5Mbps	362 byte	2,32 Mbps	2,08 Mbps	10,34%
	1448 byte	1,73 Mbps	1,77 Mbps	2,31%
	1086 byte	1,66 Mbps	1,68 Mbps	1,2%
	724 byte	1,53 Mbps	1,51 Mbps	1,31%
2Mbps	362 byte	1,23 Mbps	1,21 Mbps	1,63%
	1448 byte	0,9 Mbps	0,98 Mbps	8,89%
	1086 byte	0,88 Mbps	0,96 Mbps	9,09%
	724 byte	0,82 Mbps	0,86 Mbps	4,88%
1Mbps	362 byte	0,7 Mbps	0,74 Mbps	5,71%

A táblázat a csomagméret és az adatátviteli üzemmód függvényében adja meg a mért és számított adatátviteli sebességeket, és ezek különbségét.

15dB jel /zaj viszonyú csatornát biztosítva végeztünk zajos esetre méréseket. Az alábbi mérési eredmények 15dB jel/zaj viszony esetén helyesek, de a protokoll működését zajos csatornán jól illusztrálják (9. ábra).

Az ábráról jól látszik, hogy a nagyobb átviteli sebesség esetén alkalmazott modulációk zavarérzékenysége mennyivel nagyobb azonos zavartatás esetén. Jól látható, hogy ez a zavartatás 1 és 2 Mbs-os adatátviteli üzemmódban nem okoz észrevehető sebességcsökkenést, míg 11 Mbs-os esetben közel 10% sebességcsökkenés tapasztalható.

A TCP protokoll megbízható vég-vég kapcsolatot biztosít csomagújraadásokkal. Minden sikeres csomagátvitelt egy nyugtázó csomag követ. TCP protokoll esetén az RTT (Round Trip Time, körbefutási idő) idő vizsgálata adhat értékes eredményeket. Az RTT idő a csomag küldésétől a nyugta megérkezéséig eltelt idő. Zajos csatornán a bithibák miatt jelentkező csomagújraadások növelik ezt az időt, így ezen keresztül alkothatunk képet a protokoll működéséről zajos csatornán.

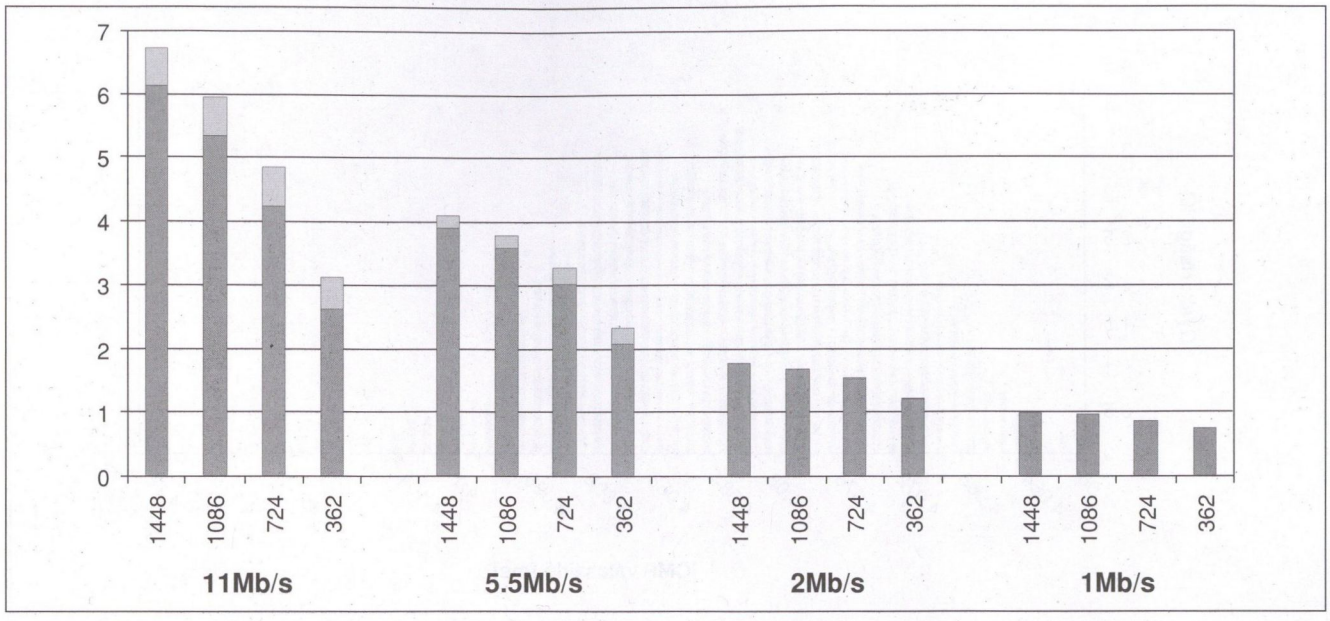
Az előzőekben ismertetett zajos környezetben a következő eredmények tapasztalhatók (10. ábra).

Megfigyelhető, hogy az RTT érték drasztikusan növekszik magasabb adatátviteli sebességnél a gyakori újraadások miatt. Mivel a bithibák véletlenszerűen tűnnek fel, így az RTT minimális értéke konstans, hiszen vannak tökéletesen átfutó csomagok. Ezzel szemben a maximális érték széles hatások között ingadozik.

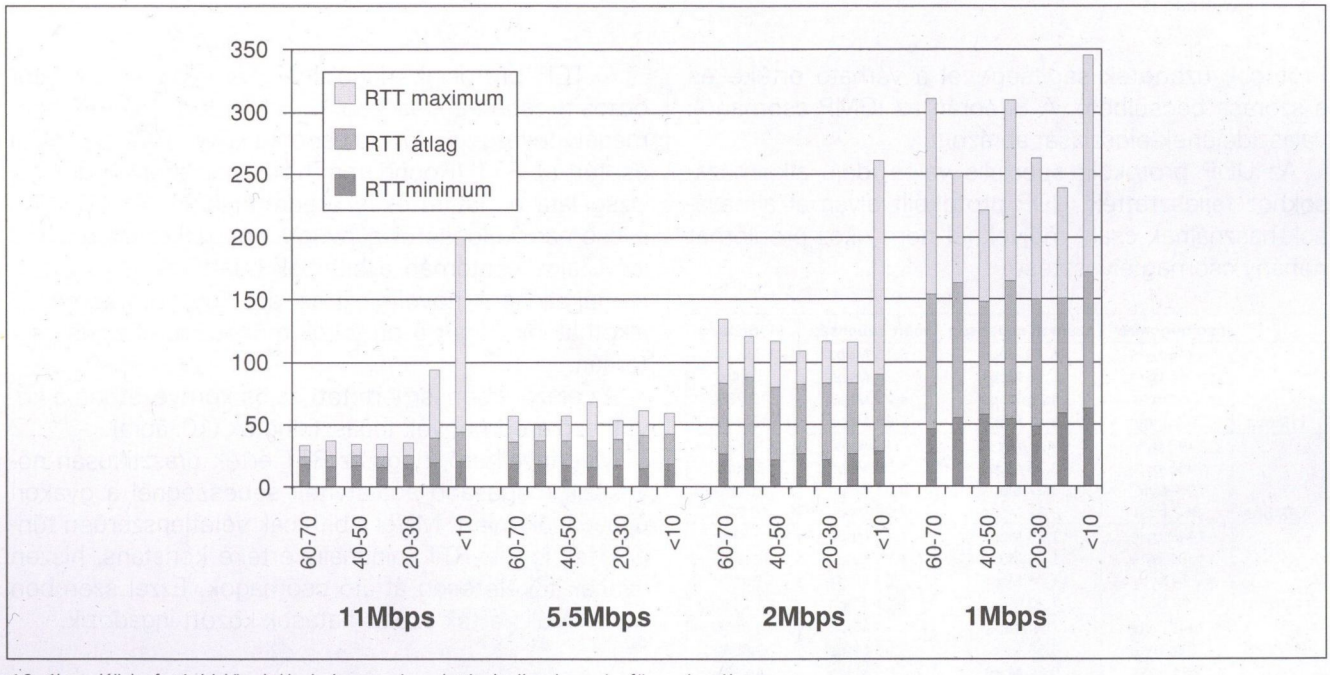
Következtetések

Méréseink során kiderült, hogy a vezeték nélküli mobilhálózatok teszteléséhez/fejlesztéséhez egy testbed szükséges. Ennek segítségével könnyen előállíthatunk különleges topológiákat a mérések igényei szerint. Mivel a külső zavarok kiszűrése igen nehéz, így ideális csatornát sem lehet biztosítani. Eddigi mérési eredményeink elemzésével egyszerűen tervezhetünk cellás vezeték nélküli hálózatot, és átfogó képet kaptunk a 2.4GHz-es ISM sávban működő mobil ad-hoc hálózatok előnyeiről, hátrányairól és különböző tulajdonságairól.

Megállapíthatjuk, hogy a mobilhálózatok jó alternatívát biztosítanak a fix hálózatoknak. A jelenleg elterjedt adatátviteli protokollok a vezeték nélküli hálózatokon is



9. ábra UDP adatátviteli sebesség ideális és zajos csatornán a csomagméret és az alkalmazott üzemmód függvényében



10. ábra Körbefutási idő a jel/zaj viszony és adatátviteli sebesség függvényében

biztonságosan alkalmazhatók, de az alkalmazásoknak fel kell készülni a nagyobb hibavalószínűségre és késleltetésekre. Egy gondosan tervezett cellás rendszer azonban biztos és megbízható kommunikációt biztosít az egész lefedettségi területen.

Referenciák:

1. Robert C. Dixon: Spread Spectrum Systems with Commercial Applications
2. Spread Spectrum Scene at www.sss-mag.com.
3. Lucent Technologies WaveLAN IEEE 802.11 Pc Card User's Guide
4. D. Duchamp and N. F. Reynolds. Measured performance of a wireless LAN. In Proceedings of the 17th Conference on Local Computer Networks, pages 494–499. IEEE, Sep 1992.
5. B. Tuch. Development of WaveLAN, an ISM band wireless LAN. AT&T Technical Journal, pages 27–37, July/August 1993.

A magyar televíziózás jövője

KOZMA IMRÉNÉ

történész, társadalomkutató főmunkatárs
Magyar Informatikai Szövetség

A Híradástechnika ez évi első számában (január, 55–58. old.) megkezdtük a beszámolást a fenti címmel megrendezett konferenciáról. A jelentős tanácskozás további két érdekes előadásának néhány gondolatát az alábbiakban tesszük közzé.

A második nap délutánján műszaki témájú szakmai előadások hangzottak el. Az elnöki tisztet betöltő Garádi János (a MIKE elnöke) kiemelte, hogy azért választották a „jövő technikai lehetőségei" címet ennek a szekciónak, mert a jövő alapvetően meghatározó Magyarország számára.

Az IBM Magyarország Kft. képviseletében Ákos György kereskedelmi igazgató „E-business megoldások a média iparág részére az IBM-től" címmel tartotta meg előadását.

Az elmúlt években forradalmi változás kezdődött a tartalom előállítását, kezelését és elosztását biztosító eszközök területén. A tradicionális megoldások mellett egyre több helyen számos feladatot már IT eszközök látnak el. Ez és a különböző technológiák gyors fejlődése jól azonosítható trendeket mutat, melyek a következők:

– Digitális eszközök és technológiák megjelenése:

A változásokat e folyamat teszi lehetővé, ugyanis a tartalmat digitális módon állítják elő, menedzselik és továbbítják. Ez számos nem csak technikai jellegű változást von maga után. Az alkalmazott megoldások függvényében a legnagyobb mértékben például a munkafolyamatok és az üzleti modellek módosultak. Például a nagy sebességű hálózatok alkalmazásával lehetőségünk van az interaktív, testreszabott tartalom on-line elérésére.

– Digitális termékek megjelenése:

A kézzelfogható termékek mellett ma már gyakran találkozhatunk a digitális termékekkel is, melyekkel a kereskedés a „hagyományos" kereskedelmi modelleket nagymértékben módosítja és a tartalom biztonságos elosztását és szerzői jogainak védelmét teszi szükségessé. Mivel nagyon sok szervezet a bevételeit a tulajdonában lévő tartalom felhasználásával vagy értékesítésével szerzi, ezért a tartalom nagyon fontos értékké vált. Kulcsfontosságú lett a tartalom biztonságos tárolása, továbbítása és kereskedelme is.

– Tartalom elérési módjának megváltozása:

Az új médiumok megjelenése (például az internet) lehetővé teszi, hogy a befogadó az általa kiválasztott digitális tartalmat akkor hasznosítsa (hallgassa, olvassa, nézze stb.), amikor szüksége van rá. A tartalom „on-demand" elérése jobban igazodik a befogadó életéhez, ennek következtében a befogadó döntheti el, hogy mikor, mit és milyen feltételek mellett használ.

– Az értékesítési modellek megváltozása:

A termékek és szolgáltatások hatékony értékesítéséhez ma már nem minden esetben elegendők az átlagos vásárló általános ismeretei. A vállalkozásoknak arra kell törekedniük, hogy az egyes ügyfeleket és azok egy bizonyos termékkel vagy termékcsoporttal kapcsolatos viszonyát és viselkedését ismerjék. Talán a leghétköznapibb példa az internetezők és egy adott portál viszonya lehet. Nem minden esetben elegendő azt tudni, hogy az egyes oldalaknak hány látogatója volt, viszont az adott portál fejlesztői számára értékes információkat hordozhatnak az olyan mutatók és összefüggések, mint például hogy az egyes korosztályba eső házas vagy egyedül álló férfiak és nők mely témák iránt érdeklődtek a legnagyobb arányban.

Az IBM is kidolgozta stratégiáját, amely a következő három nagyobb egységből áll:

1. Az üzletvitel átalakítása az az út, amely a jelenlegi helyzetből az elképzelt állapot eléréséhez vezet. A változások többek között a munkafolyamatokban, a szervezeti felépítésben, az alkalmazottak tudásában is szükségesek. Ez pontos tervezést és az átalakítási folyamat céltudatos irányítását kívánja.
2. Az üzletvitel meghatározása vagy átalakítása után a funkcionális rétegben a legnagyobb szerepet a tartalom létrehozása és elosztásának menedzselése kapja. Ehhez a tartalomhoz a vele kapcsolatos metaadatokat rendelünk, és tároljuk. Később ezen informá-

ciók alapján keressük vissza azokat. Természetesen minél részletesebb és pontosabb a tartalomhoz rendelt metaadathalmaz, annál kevesebb időt igényel az archivált tartalom visszakeresése.

3. A piramis alján helyezkedik el a megbízható üzemelést biztosító infrastruktúra-réteg, amely többek között tranzakciófeldolgozó és adattároló komponensekből áll.

Bálint Zoltán ügyvezető igazgató „Új lehetőségek az interaktív televíziózásban és a nézettségmérésben" címmel tartotta meg előadását.

A televíziózás történetében mindig igény volt arra, hogy a műsorok készítői, a nézők népes táborával valamilyen (direkt, vagy indirekt) kapcsolatba kerüljenek. A televíziós műsorok nézőiben is kezdettől létezik a természetes igény a műsorokhoz fűződő véleményük kinyilvánítására, melyet jellemzően levél vagy telefon valósít meg. Az utóbbi időben terjedő SMS, vagy webkapcsolat sem változtat gyökeresen a véleménynyilvánítás hatékonyságán. Az interaktív televíziózás új alapokra helyezi ezt a problémát. Ennek jelentősége, hogy a műsorok alkotói, készítői és a nézők a direkt véleménykérő és véleménynyilvánító kapcsolatuk útján alakíthatják ki az optimális műsorszerkezetet.

Új típusú műsorok jelenhetnek meg a tv-ben, amelyek sok néző közvetlen, interaktív bevonásával bonyolíthatók le. Az alábbiakban ismertetett rendszer lehetővé teszi azonos időben, több tv-csatornán, a nézők különböző csoportjaival az interaktív kapcsolatot. Előnye az egyéni azonosításnak, hogy az interaktív televíziózást hatékonyan lehet használni oktatási célokra, és ez jól alkalmazható a közölt ismeretanyag megértésének vizsgálatára. Az oktatási műsor közben időnként 5-7 tesztkérdést tesznek fel az elhangzottakról és a nézők által adott válaszok alapján vagy továbblépnek, vagy sokhibás válasz esetén újból elmagyarázzák az adott témakört. Hasonló elven teszttípusú távvizsgáztatás is lehetséges az egyéni azonosítás lehetőségének felhasználásával. A rendszer tehát terjesztheti a televízión keresztüli távoktatást, biztosítva az adott terület legkiválóbb szakembereinek oktató munkáját.

A rendszer három fő részre bontható. A tv-nézőknél található végberendezésekre (kommunikátor, személyi készülék), lekérdező adatgyűjtő központra, és a tv-társaságok végberendezéseire, amelyek az adatgyűjtő központ feldolgozott adatait töltik le.

A tévénézők végberendezései

A kommunikátor

Kommunikátor teremt kapcsolatot a tv-t néző családtagok és az adatgyűjtő központ között. A kommunikátor a következő interfészekkel rendelkezik:

ISDN 2 adapter, amely lehetővé teszi a kommunikátor gyors lekérdezését az adatgyűjtő központ oldaláról. Bizonyos helyzetekben a kommunikátor is kezdeményezhet hívást az adatgyűjtő központ felé.

Videobemenet, amellyel a család tv-készülékének videojel szintű kimenetéhez csatlakozik (SCART, vagy RCA csatlakozó). Ma már minden készüléken található alapsávi videojel-kimenet. A videojelet figyelve a kommunikátor automatikusan megállapítja, hogy a tv be van-e kapcsolva és éppen melyik tv-társaság műsorára van hangolva. Ebből kétféle további információ nyerhető, melyek lényege, hogy a beállított (bármikor átállítható) csatornaszámtól, a különböző kábeltévé szolgáltatók által más-más vivőfrekvencián sugárzott tv-állomások azonosíthatók.

Rádiófrekvenciás interfész, amely vezeték nélküli kapcsolatot tart fenn a családtagok személyi készülékeivel. A kapcsolattartásnak kétféle szintje létezik. Az automatikus jelenlét érzékelés, amikor a kommunikátor a személyi készülékkel folytatott kapcsolatból megállapítja, hogy kik tartózkodnak a bekapcsolt tv készülék előtt. Lekérdezés, melynél a személyi készülék, a tv műsorral kapcsolatos személyes információt közöl a kommunikátorral.

A személyi készülék

A személyi készülék 5-10 nyomógombot és egy LCD kijelzőt tartalmaz. Az LCD kijelzőre írt információ és a gombok segítségével valósítható meg a kapcsolattartás a kommunikátoron keresztül az adatgyűjtő központtal. Minden családtagnak saját személyi készüléke van, amely hordozza a rá jellemző adatokat (pl. életkor, iskolai végzettség stb.), azokat, amelyek különböző nézettségi, és egyéb statisztikai értékeléskor használhatók. A név statisztikai elemzések számára közömbös, de interaktív vetélkedők és interaktív oktatás kapcsán már előtérbe kerülhet. Az adatgyűjtő központ feladata ennek a kettősségnek a megoldása, a néző személyiségének védelme, az adott információszolgáltatáskor anonimitásának megőrzése. Ha a rendszer megvalósul, akkor a személyi készülék más rendszerekben alkalmazható személyazonosítóként, személyes adatok tárolójaként. Átvehet rendszerekben csipkártya funkciókat, mert biztonsági rendszere a csipkártyáéval azonossá tehető. Szándékunk szerint a személyi készüléket felkészítjük ilyen irányú felhasználásra is. Processzorteljesítményét, memóriaméretét úgy választjuk meg, hogy azon RSA vagy más kriptográfiai algoritmus futtatható legyen. A személyi készülék programkódja kívülről cserélhető (letölthető) lesz, annak érdekében, hogy az új szolgáltatások bevezetésekor ne kelljen készüléket cserélni. A rádiófrekvenciás kapcsolatot a „blue tooth" technológiára alapozzuk. A kommunikátor és a személyi készülék ergonómiai és formatervi kivitelének is komoly figyelmet kell szentelni.

Központi készülékek

A közforgalmú távközlőhálózat és az IP-alapú kapcsolatok egyaránt használhatók információgyűjtés céljára. A lekérdezés ideje rövidebb, mint a kapcsolatfelépítés.

Ennek gazdaságos megoldása vagy az ISDN D csatornáján, vagy „allways on" módon lehetséges. Ezek elterjedtsége azonban még nem elegendő reprezentatív felmérésre.

Felhasználás a nézettségmérés és a médiakutatás területén

A rendszer jelentős változást hozhat a magyarországi nézettségmérésben, a műsorok vagy televíziós reklámkampányok értékelésében. A jelenleg Magyarországon üzemelő rendszer (ennek megrendelője az összes fontosabb televíziótársaság és reklámügynökség) pusztán a nézők számának mérésére alkalmas. A magyarországihoz hasonló elveken működő műszereket alkalmaznak műszeres nézettségmérésre a többi európai országban is.

A tervezett eszköz legfontosabb előnye, hogy nem csupán a nézés tényét képes érzékelni, hanem össze tudja gyűjteni a műsorral, látottakkal kapcsolatos nézői reakciókat, véleményeket is. A nézők véleményének ismerete igen fontos médiakutatási szempontból, hiszen támpontokat adhat valamely műsor jövőbeli nézettségének előrebecsléséhez. Lehetővé válna a műsor vagy reklámkampány értékelése, ami elősegítené, hogy a műsor olyan módon alakuljon adásróladásra, ahogyan az a nézők igényeit, tetszését kielégíti.

A televíziók, hirdetők és reklámügynökségek keresik a választ arra, hogy a nézők hogyan fogadnak adott műsort vagy reklámot. Elképzeléseik szerint ezen vizsgálatok jórészt megválaszolhatók a tervezett rendszerben. Az üzemeltetés során szerzett tapasztalatok alapján alakul ki, hogy a műszerekkel felszerelt családokat mire kérjük, azokat milyen intenzitású együttműködéssel terheljük. Valószínűleg statisztikai szempontok miatt tanácsos fenntartani egy mintát, melyben a résztvevőktől csak minimális együttműködést igényelünk. Ezektől az emberektől nem kérünk mást, mint hogy tévézés közben hordják személyi készüléküket. A résztvevők túlzott terhelése ugyanis azt eredményezi, hogy nő a válaszmegtagadók (a részvételt visszautasítók) aránya és ezáltal hibásak lesznek a becsült eredmények. (A statisztikailag nagyon fontos véletlen kiválasztást akadályozza, ha nagy a visszautasítási arány, és csak bizonyos „típusú" emberek kerülnek a mintába.)

Ugyanakkor, egy másik mintán lehetne nem feltétlenül kvantitatív jellegű kutatásokat folytatni. A nézők válaszolnának a nekik a személyi készülékükön keresztül feltett kérdésekre (pl. mi a véleményük a műsorvezetőről, a díszletről, elég informatívan, érdekesen tárgyalnak-e valamely témát stb.), a válaszok értékelése nem csupán a nagyobb nézőszám elérését segítheti, hanem valamely rétegműsor esetén felmérhető lenne, hogy a megcélzott csoport hogyan vélekedik egy műsorról. (Pl. a testi fogyatékosoknak szánt műsort mindig kevesen nézik, ezért nem tudhatjuk, hogy maguk az érintettek mit gondolnak a műsorról, számukra érdekes vagy hasznos-e, megfelelő időben kerül-e adásba. Míg a nézettségbecslésre használt „alapmintában" módszertani okokból csak kevesebb személycserét lehet végrehajtani, ez a másik minta tetszőlegesen „átszerelhető", esetleg éppen annak a társadalmi csoportnak a tagjaihoz, akiket valamely vizsgálatba be akarunk vonni.

Felhasználás a közvélemény- és piackutatásban

A közvélemény- és piackutatásban kvalitatív és kvantitatív módszerek használatosak. A kvalitatív módszerek azt célozzák, hogy szociodemográfiailag nem reprezentatív mintán egyes személyek, típusok véleményét, szokásait mélyebben felmérjék, mint ahogyan az a nagyobb, reprezentatív mintán lehetséges. Ezeknél a kutatásoknál is alkalmazható a műszer, vagyis egy kisebb mintán több kérdést tennének fel a nézőnek, kinek a tévéképernyőjén látható lenne valamilyen, a kutatandó témával kapcsolatos összeállítás. Körülbelül száz-kétszáz műszer kihelyezésével és a részt vevő személyek időnkénti cseréjével mindig biztosítható, hogy ki tudjunk választani, 10-20 embert olyan módon, ahogyan azt a kutatás éppen megköveteli.

Ennél is alkalmasabb azonban a műszer kvantitatív kutatások végzésére, hiszen itt általában csak néhány kérdésre kell válaszolni. Az ilyen kutatások jelenleg hosszú tervezési és terepmunkát igényelnek, amelyek jelentősen lerövidíthetők, ha mindig „tartunk készenlétben" olyan, Magyarország lakosságát megfelelően reprezentáló mintát, ahol bizonyos kérdésekre gyors választ lehet kapni a résztvevőktől. A társadalmat érdeklő és foglalkoztató kérdésekről tehát gyors kutatásokat lehetne végezni, aminek nagy szerepe lehet a szociológiai kutatásokban.

A felhasználás további technikai és jövőbeni lehetőségei

A családoknál elhelyezett kommunikátor teletextdekódert is tartalmaz. Alkalmas előre meghatározott teletextoldalakon közölt információ továbbítására a tévénézők felé. Ez a kommunikáció egyirányú. A tv-társaságoktól az adatgyűjtő központ kihagyásával a nézők felé irányul. Természetesen csak akkor működik, ha a tv-készülék az adott tv-társaság csatornájára van hangolva. Az információ kódolt, vizuálisan nem értelmezhető. Ez az egyirányú kommunikáció számtalan lehetőséget rejt magában a tv-nézőkkel történő kapcsolattartásban. A személyi készülék LCD kijelzőjére kiírt – a kijelölt teletextoldalakról dekódolt – információ tájékoztat az esetleges műsorváltozásokról, műsorcsúszásokról. Előfizetéses csatornák esetén a díjfizetés esedékességéről, vagy adott előfizető hátralékáról.

A személyi készülékek nem és életkor szerint is kategorizálva vannak. Egy adott kategóriát érintő műsor kezdetére a tv-társaság külön figyelmeztetést küldhet a teletexten keresztül. A rendszer érzékeli, hogy ki van

a készülék előtt, és ki tartózkodik elérhető (rádiófrekvenciásan) távolságban – például a lakás egy másik helyiségében –, és csak annak küld jelzést és üzenetet egy műsor kezdetéről, aki nem a tv előtt tartózkodik. A tv-műsorok kezdetére akár előre programozottan is figyelmeztet a rendszer. Ennek egy lehetséges megoldása, hogy egy kijelölt teletextoldalon a tv-társaság műsora sorszámozva (1-től n-ig) található. A kijelölendő műsorok sorszámát a tv-néző beírja a személyi készülékébe. A műsor kezdetekor a tv-társaság jelzést ad. Egy diszkrét rövid hangjelzés (mint például a digitális karórák egészóra-jelzése) és az LCD kijelzőre írt szöveges üzenet figyelmeztet.

A teletext lehetőséget ad a műsorok osztályozására is. A teletexten keresztül érkező szöveges üzenet a műsorra vonatkozó kérdéseket is tartalmazhat, amelyre a személyi készülék nyomógombjai segítségével válaszok adhatók. A válaszokat a kommunikátor rögzíti, majd a következő nézettséglekérdezéssel együtt továbbítja az adatgyűjtő központnak.

Elnézést!

2002 évi 1. számában az 56. oldalon Michelberger Pál akadémikust a MTESZ elnökét alelnöknek neveztük, viszont az 55. oldalon Vizi E. Szilvesztert az MTA alelnökét elnöknek neveztük.

2002 évi 1. számban Kozma Imréné nevét helytelenül írtuk, elnézést kérünk a szerzőtől.

Hírek

Világviszonylatban a Fülöp-szigetek mobil-előfizetői járnak az élen az SMS-ek használatában: SMS-forgalmuk a világ SMS-forgalmának 10%-át teszi ki. A Fülöp-szigeteken 2000 decemberében előfizetőnként napi 9 SMS-t küldtek átlagosan. A mobilpiac fellendülését Délkelet-Ázsiában más adatok is jelzik: 2001 végén hat országban (Szingapúr, Malajzia, Brunei, Kambodzsa, Thaiföld, Fülöp-szigetek) több volt a mobiltelefon, mint a vezetékes. A mobiltelefont gazdagabb országokban másodtelefonként, szegényebbekben a vezetékes telefon pótlékaként használják. A mobil terjedését elősegíti, hogy a térségben egyöntetűen a digitális GMS-t fogadták el, bevezették a WAP-ot és több országban elindultak a GPRS felé. Nehézségek mutatkoznak viszont a mobilkészülékek adaptálásában a nemzeti nyelvekre. (H. Gy.)

❖ ❖ ❖

A Revolution Software szerint egyre több hazai kis- és középvállalat számára válik elérhetővé, hogy racionalizálja ügyviteli feladatait. A külföldi rendszereknél alacsonyabb áron bevezethető, hazai fejlesztésű integrált alkalmazások már olyan megtérülő beruházást jelenthetnek, amelyek hozzájárulnak életképességük megtartásához; a Pioneer Productions egyéves tapasztalatra alapozva jó döntésként értékelte a rEVOLUTION Iroda++ integrált ügyviteli alkalmazás bevezetését.

Az Iroda++ az integrált ügyviteli rendszerrel a számlázási, a pénztár- és tárgyieszköz-nyilvántartási, valamint a főkönyvi könyvelési feladatokat helyezte egységes rendszerbe. A választott rendszer megfelel néhány nem hagyományos lekérdezési feltételnek, és lehetőség egyedi fejlesztések kivitelezésére is.

❖ ❖ ❖

Az elmúlt év legjobb német technológiai innovációját elismerő díjat a német gazdaság, tudomány és technika 32 prominens képviselőjéből álló kuratórium a müncheni félvezetőgyártónak ítélte oda. A díjjal két teljesítmény-félvezető termékcsaládot, a CoolMOS-t és az IGBT-t tüntették ki. Előbbi PC-k, laptopok tápegységében és mobil akkutöltő készülékekben alkalmazható. Az IGBT félvezetők háztartási gépek, klímaberendezések és ipari berendezések motorvezérlő elektronikájába beépítve akár 30%-kal is csökkenthetik a fogyasztók energiafelvételét. A frankfurti Régi Operaházban lezajlott díjátadási gálán Ulrich Schumacher elnök mellett dr. Tihanyi Jenő, az integrált áramkörök fejlesztésének vezetője képviselte az Infineont, akit néhány éve az év Siemens-feltalálója címmel tüntettek ki. Az Infineon AG-ben a Siemens AG-nek a közelmúltbeli részvényeladás ellenére is jelentős tulajdonrésze van.

A válság túlélése

HORVÁTH GYULA

Tanácsadó mérnök

A 2000/2001. évek fordulóján a dot-com cégek tömeges tönkremenése már kétségtelenül jelezte, hogy a gazdasági válság legérzékenyebben a távközlési vállalatokat fogja érinteni. Bár a magyarországi helyzetjelentések még nem szólnak súlyos tünetekről, mégis célszerű áttekinteni azokat a tanácsokat, melyeket a szakirodalom a válsággal küszködő külföldi távközlési és informatikai vállalkozásoknak megfontolásra javasol. A válságot túlélők nem követhetik a korábban jól kitaposott utakat. Nem elég kimászni az átmeneti bajból, a túléléshez új stratégiára, új üzleti modellre és új műszaki megoldásokra van szükség. Csak kevés általános érvényű megoldást lehet találni, mert a döntéseket befolyásoló tényezők vállalatonként lényegesen eltérők. Ezért a továbbiakban csak példaképpen válogattunk össze a szakirodalomból kiemelt néhány hasznosítható ötletet.

Stratégia

Megfigyelhető a kockázattól való idegenkedés, aminek egyik módja az adósságok keletkezésének lehető elkerülése. (Intő példa a vállalat gazdasági erejéhez képest túl nagy kölcsönök felvétele az Egyesült Királyságban és Németországban a 3G licencek finanszírozására.) Más lehetőség a kockázat szétosztása sok ügylet között. Azok az eladók, akik részt vesznek vevőjük beruházásának finanszírozásában, vevőjükkel együtt gondosan térképezzék föl a tervezett beruházással járó kockázatokat.

A konjunktúra idején létrejött igen jelentős, ma kihasználatlan kapacitások (pl. túl sok használatlan szál az óceánok alatti fényvezető kábelekben, betelepülést szolgáltató vállalatoknál félig üres termek stb.) várnak a még habozó ügyfelekre. Következésképpen az új termékek megválasztásakor, a bevezetésükhöz szükséges ráfordítások mértékének meghatározásakor igen fontossá vált az értékesítők piaci tapasztalatain alapuló véleménye a várható értékesítési lehetőségekről.

A beruházások kedvezőnek tartott megtérülési ideje tovább csökken, főleg a bizonytalanság miatt. A finanszírozók a nemrég még elfogadhatónak tekintett kétéves megtérülés helyett negyed- és fél évet emlegetnek.

Finanszírozás nélkül is megvalósítható kétféle stratégia: a méretgazdaságosság növelése más vállalatok fölvásárlásával vagy velük egyesüléssel, vagy az ügyfelek szűkebb csoportja számára több szolgáltatást felölelő csomagok kínálása. Könnyebb finanszírozni azokat az ötleteket, amelyek megvalósításához jelenleg kihasználatlan kapacitást lehet igénybe venni.

Kimerülőben van a pánikhangulatban gyorsan bevetett költségcsökkentő módszer: az alkalmazottak tömeges elbocsátása. Közismertek az igazán nagy cégeknél a személyzet tízezres nagyságrendű, szerencsére nem egyszerre végrehajtott leépítése, amelyből hazánk sem maradt ki. Nemzetgazdasági szinten az ilyen intézkedés, amikor már túlmegy a munka és a munkaerő néha célszerű átcsoportosításán, jelentősen megnövelheti a munkanélküli segélyre fordítandó összeget. Ezenkívül a munkavállalók jövedelemadójának és társadalombiztosítási befizetéseinek elmaradásán keresztül a szükséges közkiadások elhalasztására, a költségvetési kötelezettségek elmaradására vezet, vagy végső soron a munkában maradottakat terheli.

Üzleti modell, költségcsökkentés

Az új jelszó: „a szolgáltatás a legfőbb úr", aminek jegyében a távközlési vállalat legértékesebb vagyontárgynak már nem a műszaki berendezéseket, hanem a rendelkezésre álló szolgáltatásokat tekinti. A hagyományos üzleti gondolkodás nem volt eléggé felhasználóközpontú. A távközlési vállalatoknak nemcsak hatékonyabban kell dolgozniuk, hanem fontos állandó kiadásaik csökkentése és karcsúsítása is. Továbbá az ügyfelek igényeire is koncentrálniuk kell. Utóbbi azt is jelenti, hogy pl. a széles sávú szolgáltatások iránti igények felbecsülésekor nem a legjobb kiindulópont az olyan magasabb szintű statisztikai adatok elemzése, mint a szóba jövő háztartásokban fejenként távközlésre fordított pénzösszeg megbecsülése. Ehelyett az adott régió, település stb. valóságos infrastruktúraszükségletéből célszerűbb kiindulni. Erre a célra széles sávú szolgálatok számára a hagyományos felülről lefelé modell helyett alulról fölfelé modellben a lappangó, az igényeket föltáró modellt javasolnak.

Üzleti terv készítésekor fejtörést okoz, hogy az új generációs széles sávú hálózatok létrehozásának költségei gyorsabban fognak növekedni, mint a velük elérhető bevételek. Ennek egyik megoldása a tisztán optikai hálózat építése, mert abban elektronikus-optikai és az optikai-elektronikus átalakítók nem szükségesek, sőt a hálózat is hamarabb épül meg.

A költségcsökkentés három fő útja – egy brüsszeli szolgáltató cég főnöke szerint – az egyszerűség, a termelékenység és az integrálás. Ami az egyszerűséget illeti, sokan vannak amellett, hogy a fix hálózatot valamelyik IP szolgálati réteget támogató kapcsolt hullámhosszok rétegéig kell leegyszerűsíteni, de ez kezdetben komplikáltabb megoldásokra vezet. Viszont az IP szolgálatok egyre inkább mind a hét réteget igénybe veszik. Mégis, a hagyományos távközlési vállalatok nem jutnak el az alkalmazási rétegig, és tipikusan nem mennek az IP szállítási rétegbe vagy a fölé. Ugyanakkor újabb rétegek felhasználása a bonyolultságot fokozza. Ez napjaink egyik dilemmája.

A másik dilemma az elsőbbség eldöntése az üzemeltetési és a befektetési költségek között. A befektetések gyors megtérülésére törekvés a befektetési költségek csökkentésére koncentrál, de újabb kezdeményezések az üzemeltetési költségek csökkentését helyezik előtérbe. Ezeket éppen napjainkban növelik a mellőzhetetlen ügyfélkapcsolatnak, a számlázásnak, az üzemeltetés támogatásának és a vállalati erőforrások tervezésének költségei.

Nehézséget az jelent, hogy könnyebb befolyásolni a kiadásokat, mint a bevételeket, pedig a mai nehéz körülmények között mindent kézben kell tartani, amit csak lehetséges. A két költségtényező egymásra hatását is mérlegelni kell: az üzemeltetési költségek némi megnövekedésének árán is előnyösebbnek bizonyulhat egy nagyobb költséggel járó befektetés. Semmi esetre sem szabad azonban megfeledkezni a vállalat alapfeladatáról, a szolgáltatásról.

Egyes befészkelt szolgáltatók a kiszervezés folytán fölöslegessé vált ingatlanaik egy részét is eladják. Az ily módon szerzett pénzt például a France Telecom adósságainak csökkentésére használta fel. Más vállalatok is éves jelentéseikben büszkélkednek adósságaik csökkenésével.

További, szöges ellentét megoldását követelő dilemma a hálózatok és a szolgálatok között mutatkozik, mert ezek egymással vetélkednek az erőforrások és a tőke megszerzésében . A már megtelepedettek inkább a hálózat, az újonnan jöttek inkább a szolgálatok oldalán állnak. Márpedig az üzletet végső soron a használóval kötik, aki a távközlőhálózattól szolgálatokat vár, és nem túlságosan érdekli, hogy azokat milyen műszaki eszközökkel valósítják meg. A fontos műszaki paraméterek, mint a megbízhatóság, az átvitel jósága és mások mellett a humán vonatkozások kerülnek előtérbe. Ilyenek a végberendezések egyszerű, világos, áttekinthető kezelése egyértelmű, jól érthető, minden fontos részletre kiterjedő kezelési utasítások segítségével. Legfontosabb azonban az ügyfélkapcsolatok kezelési módja a szolgáltató részéről (CRM). Nem csupán az ezt végző munkatársak modora, stílusa nyom sokat a latban, hanem még inkább az a képességük, hogy az előfizető által felvetett ügyeket milyen jól, gyorsan, az esetleges nehézségeket ügyesen leküzdve intézik el. Ez személyiségükön, felkészültségükön és a rendelkezésre álló információtechnikán múlik. Ha a szolgáltató a technikát és a munkatársakat a használók igényeivel jól összehangolja, akkor a bekezdés elején említett dilemma feloldódik, versenyelőnnyé változik.

Műszaki megoldások

A vita további tárgya a teljes szolgálatú hálózati modell túlélése, amire a legutóbbi időkig világszerte sok üzemeltető vállalat lankadatlanul számított. Az egyik oldalon amellett érvelnek, hogy a teljes szolgálatú hálózati modell hosszú távon nem életképes, mert nem lehet minden szolgálatot integrálni, többféle igény válik közel egyformán fontossá.

A másik oldalon viszont arra hivatkoznak, hogy egy növekedni kívánó új vállalkozásnak fölfelé, nem pedig vízszintesen kell terjeszkednie. Utóbbi egy befészkelt szolgáltató számára nem az ügyfelek számának mindenáron való növelését, hanem új, műszakilag már kiforrott területek meghódítását, mint pl. tevékenységük kiterjesztését a mobil hírközlésre, a széles sávú, esetleg más szolgáltatásokra jelenti.

Ezektől függetlenül a nagyobb szolgáltatók körében az az irányzat alakul ki, hogy szervezzék ki a szolgáltatásuk lényegéhez nem tartozó tevékenységeiket. Meggondolásaik kiindulópontja annak felismerése, hogy a költséges befektetések elkerülése végett tulajdonképpen „kiszipolyozzák" meglévő vagyontárgyaikat, mert a szoftverkészítőknek megvannak az eszközeik ahhoz, hogy csekély díjazás ellenében szoftvereikkel új bevételi forrásokat nyissanak meg. Ezen keresztül megbízóik menedzselt szolgáltatókká válhatnak. Ennek haszna az, hogy erejüket ügyfeleik segítésére koncentrálhatják a válság előttinél kedvezőbb helyzetbe kerüléséhez. Ezáltal likviditásuk is javul, legalábbis a gondokkal teli 2002. évben.

Az Ovum tanácsadó cég szerint elengedhetetlen, hogy a befészkelt cégeknek világos és határozott stratégiájuk legyen vezetékes hálózatukban a széles sávú szolgálatok bevezetésére. Előrejelzésük szerint ugyanis a teljes vezetékes piac csak 11%-kal fog növekedni, csak ennyin osztozkodhatnak az alternatív szolgáltatókkal. Arra is figyelmeztetnek, hogy az elemzők hajlamosak a befészkeltek értékét 30-50%-kal alá-, az alternatívok értékét nagyon, néha 100%-kal is túlbecsülni.

A befészkelteknek a BT mutatott példát, amikor külön vállalatokba szervezte a mobil, az előfizetői és a kiemelt előfizetői tevékenységét (szétszervezés); ezen túlmenően következetesen megszabadult kisebb, mellékes szerepet játszó részlegeitől. A példamutatás ellenére a befektetők nem biztosak abban, hogy az átszer-

vezések növekedésre vezetnek. A szétszervezéshez fizetőképes kereslet is szükséges.

Az egyik felvetett költségcsökkentő módszer a rendelések és felügyeletük interaktív intézése (ISS, interactive selling solutions) . Ez elkészíti az ajánlatot, biztosítja a tapasztalat szerint az esetek 20-60%-ában előforduló tervezési hibák elkerülését, esetleg alternatív javaslatot is készít. Szükség esetén a programmal készített ajánlatot még egy eladóval meg lehet beszélni. Előnye idő és pénz megtakarításában mutatkozik, mert az ajánlat gyorsan elkészíthető, letárgyalható és a rendelés kiadható, a hibák kijavításával nem kell bajlódni, és időt tölteni.

A módszer jól használható kisebb rendszerek, például alközpontok és más, konfigurálható rendszerek értékesítésekor.

A sajtó a válságról

Feltűnő, hogy mind a napi, mind a szaksajtó óvatosan értékeli a válságjelenségeket. Hacsak nem elfogult, a válságra utaló tényeket (elbocsátások, részvényárfolyamok zuhanása, csődök stb.) szárazon közli. Szóhasználata is visszafogott, a válság kialakulásának fokozatait lassulásnak, recessziónak, depressziónak, krízisnek nevezi. Szerintem az ok pszichológiai meggondolásból fakad: ha pontosan jellemzik a gazdaság helyzetét, akkor az önbeteljesítő jóslattá válik, ilyet viszont nem akarnak terjeszteni. A fellendülés kezdetét a valóságosnál korábbra teszik, még ha negyed-, fél évvel, vagy még többel később jelentkezik. Ugyanakkor a fellendülés kezdeti jeleiről – dicséretes módon – gyorsan hírt adnak.

Kitekintés

Sokan a 2002. évet nehezebbnek vélik, mint amilyen a 2001. volt. Szerintük sok cégnek kötéltáncot kell járnia.

Mint a korábbiakból is kitűnik, a szolgálatok iránti igények felmérésében nagy a bizonytalanság. Egy dolog viszont majdnem biztos: válság idején sem szabad a K+F tevékenységet csökkenteni, hogy a fellendülés kezdetén már készen álljanak azok az innovációs eredmények, amelyek a keletkező fizetőképes kereslet kielégítésére, ezzel a piaci részesedés növelésére alkalmasak. Mégis, egyetlen szolgáltató sincs felmentve az igények minél pontosabb felmérése alól mind mennyiségi, mind minőségi tekintetben, már a K+F témák kiválasztása előtt sem.

A válságok idején gyakran előforduló jelentős árcsökkenések egyik oka az lehet, hogy csődbe ment szolgáltatók berendezéseik egy részét elárverezik, amit más, ugyancsak rossz helyzetű szolgáltatók megvesznek. Ezzel az új berendezések árát is lenyomják, melyek gyártóinak helyzete is romlik.

Végül nem szabad elfeledkezni a régi tapasztalatról: minden konkrét ötletről, javaslatról csak gondosan elvégzett, ellenőrzött adatokból kiinduló gazdaságossági számítások alapján szabad dönteni. A bevezetőben utaltam arra, hogy csak kevés általánosan, vagy sok esetben alkalmazható megoldás mutatkozik, ezért kell a nem ritkán divatos szakmai jelszavak hatására érdekesnek látszó megoldásokat alapos gazdaságossági számításokkal ellenőrizni.

Hír

Rádiótelefonnal szívek találnak egymásra

A Siemens mobilkommunikációs ágazatának megkérdezéses vizsgálata az európai mobilosok meglepő szokásaira és kívánalmaira mutatott rá; az európaiak SMS-sel flörtölnek. A felhasználók a szolgáltatást a rövid, romantikus üzenetek hírvivőjeként fedezték fel. A vizsgálat az elküldött üzenetek tartalmát és a hozzájuk kapcsolt képes üzeneteket vette górcső alá. Kiderült, hogy a szerelem és a romantika egyaránt kifejezésre jut a kis kijelzőkön, mely üzenetek helyi jellegzetességei külön érdekesek. Különösen „tüzeseknek" az angol üzenetek bizonyultak. A leggyakrabban az olaszok tévednek „mellékutakra" a handy segítségével. A férfiak inkább általános dolgokról írnak, míg a hölgyek szeretnek kifinomult burkolt célozgatásokat tenni.

Hírek

A GSM Szövetség, a GSM díjazási program keretén belül Ericsson Segítségnyújtási Programot, a katasztrófák okozta emberi szenvedés gyors és hatékony enyhítését célzó globális kezdeményezést a „vezeték nélküli technológia leghatékonyabb vészhelyzeti alkalmazásaként" ismerte el.

Ezzel a díjjal olyan programokat honorálnak, amelyek természeti események, humanitárius katasztrófák és egyéb vészhelyzetek folytán krízishelyzetbe került területeken nyújtanak különböző szolgáltatásokat.

Az Ericsson Segítségnyújtási Program az egyes katasztrófahelyzetek egyedi kommunikációs igényeit kielégítő egység, amely távközlési berendezésekből és az azokat telepítő és működtető személyekből áll.

A Kutatási és Műszaki Referenciacsoport felméri a katasztrófa következményeinek elhárítása során kialakított, kommunikációval kapcsolatos tapasztalatokat, és hatékonyabb katasztrófaelhárítást célzó műszaki javaslatokat dolgoz ki, és technikai segítséget nyújt a nemzetközi humanitárius szervezetek számára.

Mindezt széles körű együttműködésben végzi.

❖ ❖ ❖

Az Oracle elindította a Windows Fejlesztői Központ (Windows Developer Center) szolgáltatását a vállalat online fejlesztői hálózatán, amellyel azoknak a fejlesztőknek és partnereknek a munkáját szeretné segíteni, akik jelenleg Microsoft Windows alapú technológiákat használnak az Oracle szoftvereivel együtt. A Windows Fejlesztői Központ ingyenesen elérhető fejlesztőeszközöket és szolgáltatásokat nyújt, amelyek az Oracle Technology Network (OTN) hálózatán érhetők el.

A Windows Fejlesztői Központban megtalálják mindazokat az erőforrásokat, amelyekkel nagyobb teljesítményt biztosíthatnak alkalmazásaiknak, és segítséget kaphatnak fejlesztéseikhez.

❖ ❖ ❖

Az idén Cannes-ban megrendezett 3GSM világkongresszuson a Siemens IC Mobile ágazata számos új mobilkommunikációs alkalmazást és eszközt mutatott be, és a GPRS és UMTS élő bemutatójára is sor került. Az előbbiek között új hálózati technológiák és információs szolgáltatások, valamint a nagyfelhasználóknak szánt GPRS és UMTS alapú hálózati megoldásportfólió is szerepelt.

A középpontban az m.traction nevű alkalmazáscsalád állt, amely multimédia üzenetküldésre, játékra, videojel-átvitelre és vásárlásra is képes, akár a rendszer új, vezeték nélküli modulkiegészítői. A cannes-i bemutatóútján az EDGE és UMTS technológiák innovatív megoldásait lehetett megtekinteni.

❖ ❖ ❖

Siemens Mobile összefog a Nokiával és a Sony Ericssonnal a mobiltelefonos multimédiás üzenetszolgáltatások (Multimedia Messaging Service, azaz MMS) minél gyorsabb fejlődése és globális elterjedése érdekében. A gyártók az AU-System internetes konzultációs céggel közösen olyan kompatibilis eszközök és szolgáltatások kifejlesztését tervezik, melyek segítségével az MMS alkalmazások vizsgálata – majd ezt követően használata – különféle márkájú mobiltelefonok és szerverek esetében egyaránt lehetővé válik.

Az MMS – mely közvetlenül az SMS-ből, azaz a rövid, 160 karakternyi szöveges információk megjelenítésére alkalmas üzenetszolgáltatásból fejlődött ki – olyan globális szabvány, amely alapján a mobiltelefonok szövegek, hangok, grafikák, fényképek, sőt akár videofilmek továbbítására is használhatók. Az első MMS funkciós készülékek előreláthatóan már a 2002-es évben piacra kerülhetnek.

„Minőségbiztosítás a nukleáris energetikában" c. ankét

A biztonsági kultúra fejlesztése, hatékony minőségbiztosítással

SIPOS LÁSZLÓ

vezető mérnök

A Műszaki és Természettudományi Egyesületek Szövetsége (MTESZ) és a Paksi Atomerőmű Rt. (PA Rt.) 2002. március 5-én Budapesten, a MTESZ Kossuth Lajos téri székházában „A biztonsági kultúra fejlesztése, hatékony minőségbiztosítással" kihangsúlyozott mottóval, és a „Minőségbiztosítás a nukleáris energetikában" címmel, egy igazán élénk érdeklődésre számot tartó és nagy sikerű egész napos minőségügyi ankétot rendezett.

A különböző érdeklődési körű, de a nukleáris energetika jövőjéért tenni akaró műszaki és természettudományi egyesületek minőségügyi szakemberei, dr. Zettner Tamás, a MTESZ Központi Minőségügyi Bizottság elnökének vezetésével és a Paksi Atomerőmű Rt. felső vezetése, dr. Mészáros György, a PA Rt. Igazgatóság elnöke és Baji Csaba vezérigazgató még az elmúlt év végén megállapodtak egy közös szervezésű minőségügyi ankétban. Az egyeztetések során körvonalazódtak a márciusi rendezvény legfontosabb stratégiai céljai, melyeket a két fél szakembereiből álló hétfős szervezőbizottság már a kiküldött meghívókon is feltüntethetett. A magyar minőségügyi szakemberek népes táborának érdeklődését kielégítve, a paksi kollégák végre bemutathatták a saját minőségbiztosítási rendszerüket és ismertették a minőségfejlesztés egyik legjobban alkalmazható eszközét, valamint a minőségügyi felülvizsgálatok teljes atomerőművi vertikumát. A Paksi Atomerőmű minőségügyi rendszerének építése és fejlesztése során szerzett gazdag tapasztalatok bemutatása, közkinccsé tétele az előzetes felmérések szerint nagyon sok beszállítót érdekel. A piac által szabályozott területen működő cégek, intézmények szakembereinek figyelmét egy, a jogszabályok által speciálisan szabályozott terület bemutatásával is szerették volna bővíteni.

A szervezők a célok rögzítése mellett részletesen kidolgozott programot ajánlottak az ízlésesen elkészített meghívójukban, így a több mint kétszázötven jelentkezési lapot visszaküldő, és regisztrált szakemberből, cégvezetőből éppen kétszáz résztvevő élénk érdeklődése mellett végigkísértük az egész napos ankétot. Dr. Zettner Tamás MTESZ-alelnök köszöntője és bevezető gondolatai után dr. Mészáros György, a PA Rt. Igazgatóság elnökének prezentációjából megismerhette a hallgatóság a Paksi Atomerőmű Rt. stratégiai céljait, a tervezett élettartam-gazdálkodás és teljesítménynövelés várható kihatásait. Második előadóként Szerdahelyi György, a Gazdasági Minisztérium főosztályvezetője felvázolta a Paksi Atomerőmű szerepét, nemzetgazdasági jelentőségét a magyar villamosenergia-rendszerben. Őt követte a Paksi Atomerőmű tulajdonosa képviseletében Katona Kálmán, az MVM Rt. elnök-vezérigazgatója, aki előadásában ismertette az Magyar Villamos Művek Rt. minőségi követelményeit és szigorú elvárásait. Az első blokk záró részében dr. Rónaky József, az OAH főigazgatója „A nukleáris biztonság szavatolása minőségügyi eszközökkel" című előadásában jelezte a hatósági követelményeket, melyek természetesen minden beszállító cégre vonatkoznak.

Szünet után Baji Csaba, a PA Rt. vezérigazgatója vállalkozott a szinte lehetetlenre, húsz percben sűrítve igyekezett beszámolni a biztonságközpontú minőségirányítás atomerőműves tapasztalatairól. A PA Rt. Karbantartási Igazgatóság MSZ EN ISO 9001: 1996 alapú minőségbiztosítási rendszer kiépítés és tanúsítás eredményeiről Németh Gábor karbantartási igazgató beszélt, majd a közelmúltban Pakson lezajlott OSART-felülvizsgálat tapasztalatait Szucsán Sándor biztonsági igazgató továbbította a hallgatóságnak. Radnóti István minőségfelügyeleti főosztályvezető, mint a második blokk gazdája lehetőséget biztosított a megjelentek számára kérdések és hozzászólások megtételére. Az élénk érdeklődésből kitűnt a szakmai tájékozottság és a felelősségtudat, mely a Paksi Atomerőmű kezdeti építésétől a mai napig, és reméljük, a jövőben is megmutatkozik.

Ebéd után dr. Katona Tamás, a PA Rt. projektvezetője az élettartam-gazdálkodási, és a teljesítménynövelési kiemelt vállalati projekt minőségü-

gyi céljait ismertette. „A személyzetminősítés, az oktatás és továbbképzés paksi tapasztalatairól" Simon Péter főosztályvezető tartott bőséges előadást. Volent Gábor főosztályvezető, a PA Rt. Környezetközpontú Irányítási Rendszerét mutatta be, majd legvégül Bujtás Tibor, a PA Rt. sugárvédelmi osztály vezetője a minőségbiztosítás megvalósítása a személyi dozimetriában, a kibocsátás- és környezet-ellenőrzésben témakört járta körül. A harmadik blokk gazdája, Szucsán Sándor biztonsági igazgató ismét biztosított lehetőséget a hozzászólásokra, majd röviden értékelte és összegezte a végzett munkát. Mivel az előadók időben leadták előadásukat, így a résztvevők olyan CD-ROM-ot kaptak ajándékba, melyen odahaza ismét megtekinthetik az ankéton kivetített fóliákat.

Hírek

Az Oracle Corporation bemutatta az E-Business Suite vállalatirányítási alkalmazásegyüttes legújabb szolgáltatását, amely a vállalatok számára lehetővé teszi a „napi zárást". Az alkalmazás a vállalat egészét tekintve napi szinten követi az üzleti folyamatok állását, így a stratégiai és az operatív feladatokhoz is a pillanatnyi helyzetet tükröző információk azonnal rendelkezésre állnak.

A rendszer lelke egységes adatmodell, ami lehetővé teszi, hogy a vállalati adatok egyetlen példányban, közös adatbázisban kerüljenek rögzítésre. Ez egy vállalati portált is tartalmaz, amellyel a „napi zárás" adatait (pénzügyi mutatókat, eredménykimutatás-összefoglalókat és egyéb adatokat) több mint húsz különböző szempont szerint lehet webes felületen lekérdezni.

Az E-Business Suite alkalmazásai az egységes adatmodell révén akár egyszerre 29 nyelven futhatnak párhuzamosan.

❖ ❖ ❖

A Cisco és az IBM útjára bocsátja mobil irodai programját

Az IDC előrejelzése szerint a mobil munkaerő és a távmunkát végzők száma Nyugat-Európában a 2000. évi 10 millióról 2005-ig 28,8 millióra növekszik. A Cisci és az IBM évek óta biztosítja munkatársainak, hogy az irodán kívülről is kapcsolódhassanak a vállalati hálózathoz. A Cisco több mint nyolc éve indította el belső távmunkaprogramját, és az első három éves periódusban 300 százalékos megtérülést könyvelhetett el a befektetésen.

Az új program a Cisco számos vezetékes és vezeték nélküli, így többek között DSL-, kábeles, ISDN- és ethernet-útválasztókat; virtuális magánhálózatokat (VPN-eket); a PIX Firewall biztonsági hardvert és szoftvert; vezeték nélküli helyi hálózati (LAN-) termékeket, többek között nagy sebességű hozzáférési pontokat, hidakat és kliensadaptereket kínál. Az IBM átfogó mobilkapcsolati és hálózati szolgáltatásokat, valamint nyilvános, vezeték nélküli LAN-okat kínál. Az IBM Global Services az ügyfelek által választott hardvereszközöket kínálja egy csomagban az ügyfeleknek.

Roska Tamás akadémikus a 2002. évi Bolyai-díj kitüntetettje

Mádl Ferenc köztársasági elnök adta át az idén a Bolyai-díjat, melyet első alkalommal 2000-ben kapott meg egy agybiológus, akkor Göncz Árpád köztársasági elnök volt a díjbizottság elnöke. Néhány magánszemély (vállalkozó, író, bankár) alapította a díjat, ezzel fejezve ki elismerésüket a tudomány iránt. Céljuk, hogy felhívják a figyelmet a tudás, a fejlesztés fontosságára. Most, második alkalommal Roska Tamást tartotta leginkább érdemesnek a díjra.

Roska akadémikus, legjelentősebb eredménye az első programozható analogikai szuperszámítógép-elv (CNN univerzális számítógép) és csip társfeltalálója Leon. O. Chua professzorral, valamint a „CNN bionikus szem"-nek F. S. Werblin és L. O. Chua professzorokkal.

1990 óta intenzív együttműködésben dolgozik Hámori József akadémikus neurobiológus kutatócsoportjával, 1993 elején létrehozták a Neuromorf Információs Technológia Posztgraduális Központot, amely egyben egy interdiszciplináris doktori iskola is, négy egyetem közreműködésében.

Budapesten született 1940-ben. A BME Villamosmérnöki Karán szerzett kitüntetéses diplomát, 1967-ben egyetemi doktori címet, 1973-ban a műszaki tudomány kandidátusa, majd 1982-ben a műszaki tudomány doktora fokozatot szerzett.

1964-től 1970-ig a Műszeripari Kutató Intézetben dolgozott, részt vett elektronikus műszerek fejlesztésében, majd aktív hálózatok szintézisével foglalkozott, elektronikus áramköröket analizáló algoritmusokat és számítógépprogramokat dolgozott ki. Majd a Távközlési Kutató Intézetben tagja a Csurgay Arpád akadémikus alapította tudományos iskolának.

1982 óta az MTA SZTAKI-ban dolgozik. 1985-től vezetője az Analogikai és Neurális Számítógépek Kutatólaboratóriumnak. A mesterséges „neurális" áramkörök és az analogikai számítás problémáival foglalkozik, különös tekintettel a celluláris neurális áramkörökre (CNN) és az analogikai celluláris számítógépekre.

Tudományos eredményeit nemzetközi folyóiratokban, konferenciákon publikálta, és meghívott vendégelőadóként ismertette azokat külföldi egyetemeken és kutatóközpontokban (USA, Európa, Japán). Öt tankönyvet írt (hármat társszerzővel), legutóbb a Cambridge University Pressnél. A J. Wiley kiadónál szerkesztett és részben írt monográfiát. Nemzetközi konferencia sorozatok tudományos szervezőbizottságának tagja, folyóiratok főszerkesztője. Több nemzetközi tudományos közleményére sok száz hivatkozás történt. Az IEEE Transactions on Circuits and Systems – Fundamentals c. folyóirat főszerkesztője.

Felkérésre 1965-tól tanított a BME Villamosmérnöki Karán, a Veszprémi Egyetemen megszervezte a műszaki informatika szakot, majd az Információs Technológiai Kart a Pázmány Péter Katolikus Egyetemen, melynek dékánja.

A New York-i székhelyű nemzetközi műszaki-tudományos (IEEE) szervezet alelnöke, alapító elnöke a „Cellular Neural Networks and Array Computing" Bizottságnak.

1992-ben az IEEE „fellow"-vá, 1993-ban az MTA levelező, majd 1998-ban rendes, 1993-ban az Academia Europaea (London), 1994-ben az Európai Tudományos és Művészeti Akadémia (Salzburg) tagjává választotta. Megkapta az IEEE Millennium Medal és a Golden Jubilee Award kitüntetéseket.

Műszaki innovációs munkájáért Gábor Dénes-díjat (1993), az analogikai számítógép elvének kidolgozásáért Kalmár László-díjat (1993), egyetemi fakultásszervezői és tudományos iskolateremtő munkájáért Szent-Györgyi Albert-díjat (1994), tudományos eredményeiért Széchenyi-díjat kapott (1994). 1999-ben a Pro Renovanda Cultura Hungariae Nagydíját kapta.

Hírek

A Sun Microsystems a 3GSM világkongresszuson bemutatta a mobil szolgáltatási megoldásait. Az átfogó vezeték nélkül architektúrát, a szolgáltatói szintű infrastruktúra-platformot és az iForce partnereit egyesítő Sun mobilrendszerek már megjelentek és gazdaságos megoldásokat biztosítanak.

A hét rétegből felépülő mobilszolgáltatási architektúra az adatbázisokat futtató rendszerektől és alkalmazásoktól a vezeték nélküli, vezetékes és széles sávú hálózatokon át egészen a végfelhasználói eszközökig terjed. A 3GSM világkongresszuson a Sun több bemutatót is tart partnereivel – például a Convergys Systemssel, a Nauance-szel és az Ulticommal – együtt, melyek címei Portál- és tartalomszolgáltatások, Hálózatok, Mobileszközök, Fejlesztőeszközök, Integrációs szolgáltatások, A környezet változásaira érzékeny szolgáltatások.

❖ ❖ ❖

A Siemens építi Európa első üzemanyagcellás erőművét a hannoveri közművek részére. A szabványos szilárdoxid-energiacellás erőmű maximális kapacitása 250 kW elektromos energia, de normális üzemmódban 225 kW energiát állít elő. Ezzel egy időben hozzávetőleg 160 kW hőenergiát is ad a kerületi távfűtőrendszer számára. Az üzemanyagcellás erőművi blokkokat a Siemens Westinghouse fejleszti és gyártja. A 250 kW-os kogenerációs (kombinált áram- és hőtermelő) egység az első szabványos termékük.

❖ ❖ ❖

A Siemens Mobile összefog a Nokiával és a Sony Ericssonnal a mobiltelefonos multimédiás üzenetszolgáltatások (Multimedia Messaging Service, azaz MMS) minél gyorsabb fejlődése és globális elterjedése érdekében. A gyártók az AU-System internetes konzultációs céggel közösen olyan kompatibilis eszközök és szolgáltatások kifejlesztését tervezik, melyek segítségével az MMS alkalmazások tesztelése – majd ezt követően használata – különböző márkájú mobiltelefonok és szerverek esetében egyaránt lehetővé válik.

Az MMS – mely közvetlenül az SMS-ből (Short Messages Service), azaz a rövid, 160 karakternyi szöveges információk megjelenítésére alkalmas üzenetszolgáltatásból fejlődött ki – olyan globális szabvány, amely alapján a mobiltelefonok szövegek, hangok, grafikák, fényképek, sőt, akár videofilmek továbbítására is használhatók.

❖ ❖ ❖

Az EDS, a PwC Consulting, a Sun Microsystems és az Oracle összeállította az utasszállítás biztonsági kihívásaira választ adó, integrált IT megoldásainak csomagját. A csomag külön-külön vagy együttesen is alkalmazható biometrikai és kockázatelemzési technológiákat tartalmaz.

A Known Traveler (ismert utas) egy önkéntes utasregisztrációs program. Az utasok a weben vagy egy reptéri kioszkban kitöltenek egy rövid kérdőívet és hozzájárulnak adataik ellenőrzéséhez, majd egy „intelligens" azonosító kártyát kapnak. A személyazonosságuk igazolása után biometrikai adataik – egyelőre ujj- és retinalenyomatuk – elektronikus formában rákerül a kártyára. Ettől kezdve az intelligens kártya használható az utas azonosítására (és a vele kapcsolatos kockázatok felmérésére) a biztonságos Known Traveler ellenőrzőpontokon.

A Secure Employee (biztonságos alkalmazott) egy alkalmazottregisztrációs és azonosítási program, amelynek célja az előző rendszerhez hasonló, de az adott közlekedési vállalat alkalmazottaival kapcsolatban. A Secure Employee egy „tízujjas" olvasóval gyűjti be felvétel előtt a leendő alkalmazottak és reptéri beszállítók ujjlenyomatait, állapítja meg a személyazonosságot és szolgál a kockázatelemzés alapjául. Az alkalmazottak hozzáférése a bizalmas területekhez és számítógéprendszerekhez e biometrikai adatok alapján történik.

Baross Gábor – A vasminiszter

NAGY BEATRIX HAVASKA

Budapesti Közgazdaságtudományi és Államigazgatási Egyetem (BKÁE)
III. éves hallgató

„Nem volt az 1867-iki kiegyezés óta a magyar nemzetnek olyan politikusa, aki a kiegyezés pontozatait nemzete javára annyira kiaknázni tudta volna, mint Baross Gábor, kereskedelmi miniszter. A kiegyezés által adott jogok felhasználásának és a törvényesség talaján állva, miniszterségének nem egészen hét éve alatt, szünetlen és fáradhatatlan működésével hosszú évtizedek munkáját végezé, és pótólá két évtized mulasztásait a kereskedelem, az ipar és legkivált a közlekedésügy terén. A tevékenysége minden pillanatát felhasználta, hogy a magyar haza anyagi jólétét megalapítsa, és erre szolgáló reformterveit megvalósítsa" – írta róla Petrovics a közvetlenül halála után készült kiadványban (1892).

Bellusi Baross Gábor 1848. július 6-án született Pruzsinán, Trencsén vármegyében (Bellus: kisközség Trencsén vármegyében). Atyja Baross Antal királyi járásbíró. Tanulmányait Léván, Győrött és Esztergomban végezte. Huszonhárom éves korában megszerezte ügyvédi oklevelét. Ezután vármegyéjében viselt hivatalt, 1874-ben főjegyző, majd árvaszéki elnök lett.

Mint a magyarosítási törekvések lelkes támogatója, megalapította és szervezte a trencsényi magyar dalárdát, melynek sokáig volt tagja, majd a trencsényi nőegylet alapításában szerzett magának érdemeket, mint az egylet titkára. 1874–1875 között a háromnyelvű „Vágvölgyi Lapok" alapítója és szerkesztője. Az újságban megjelent cikkek felölelték az aktuális társadalmi kérdéseket. Vázolt eszméiről így nyilatkozott: „Az eszméknek éppúgy megvan az ifjúságuk, mint az emberi életnek. Az én eszméimnek múltja van, és sajnálkozva tapasztalom, hogy azok szebbek voltak akkor, mint most. Ámde most hasznosabbak és ez a fő dolog. Bizony nagy kár, hogy nem lehet mindig egyesíteni az ábrándot a valóval, vagyis: a szépet a hasznossal."

1875-ben mint az újonnan alakult szabadelvű párt híve kapott mandátumot. Tisza Kálmán környezetéhez csatlakozott, kihez mindvégig – a boszniai kérdés válságaiban, majd az ellenzéki támadások közepette is – hű maradt, sőt utóbb népszerűségével is fedezte a tekintélyében kikezdett pártvezért. Parlamenti működésének kezdetétől fogva vonzották a gazdasági és közlekedéspolitikai ügyek, melyeknek csakhamar egyik szakértője és országgyűlési előadója lett. 1883-ban a Közlekedésügyi Minisztérium államtitkárává nevezték ki. Kinevezése előtt nagyobb nyugat-európai utat tett, hogy a fejlett országok közlekedési rendszerét tanulmányozza. Vasszorgalom, kitartás, pontosság és semmitől vissza nem riadó erélyesség jellemezte hivatali munkáját. 1886–1889 között közmunka- és közlekedésügyi miniszter, 1888. augusztus–szeptember között vallás- és közoktatásügyi, majd 1889. március–június között belügyminiszter, végül 1889–1892 között kereskedelemügyi miniszter. Funkciójából fakadóan a vám- és szabadalmi ügyek és az Országos Magyar Királyi Statisztikai Hivatal felügyelője volt.

Első nagyszabású munkája a tisztviselők fegyelmi eljárására vonatkozó törvényjavaslat módosítása volt 1870-ben. Az 1875–1878. országgyűlés két fontos törvényjavaslatának elkészítése fűződik nevéhez: az egyik a Magyarország és Ausztria között kötött vám- és kereskedelmi szövetségről, a másik az osztrák–magyar Lloyd gőzhajózási és postai szerződésekről szólt.

Midőn Kemény Gábor 1886-os visszalépésével a minisztérium hivatalos vezetése is a kezébe került, valóban új korszak vette kezdetét Magyarország vasúti, közlekedési és postapolitikájában. 1886. december 29-én I. Ferenc József király kinevezi közmunka- és közlekedési miniszterré. Mindenki nagy dolgokat várt az új minisztertől, de azokról az eredményekről még csak álmodni sem mert volna senki, amiket alig 5 év alatt elért. Először a hivatalnokait szoktatta még nagyobb pontossághoz. Az új miniszter szigoráról és rendkívüli munkásságáról kiszivárgott hírek csak növelték a közönségben a Baross iránti tiszteletet.

Eleinte főleg a vasutak érdekelték, mégpedig a vasutak államosítása. A gazdasági viszonyok elmaradottsága és a különleges közjogi helyzet egyaránt arra utaltak, hogy az állam feltétlenül rendelkezzék vasútjai fölött, mert az a gazdaság fejlesztésének legfontosabb eszköze.

1891-ben díszpolgárává avatta: Budapest, Győr, Fiume, Marosvásárhely, Nagy-Károly, Deés, Szaklocza. Érdemeit őfelsége a magyar király is méltányolta, előbb első osztályú vaskoronarenddel, majd a valóságos belső titkos tanácsosi méltósággal, és végre 1892-ben a Lipót-rend nagykeresztjével tüntette ki az ország legnépszerűbb miniszterét.

1887-es intézkedésével az állami kedvezményekről és a szállítási adóról szóló törvényével lehetővé tette a pangó gyárak fellendülését, és új gyárak alapítását. Miniszterssége alatt kb. 150 gyár részesült kedvezményben, melyek jelentékeny része újonnan alapított volt. Ilyen volt például a Villamos Izzólámpagyár Rt., az Egyesült Magyarhoni Üveggyárak Rt. és az Első Magyar Csavargyár Rt.

A magyar ipar fellendülését szolgáló gondoskodását kiterjesztette a szabadalmi ügyekre is. Az Ausztriával kötött szerződésen alapuló közös szabadalmazás hatására a feltalálók csak az osztrák kormánynál nyerhettek szabadalmat. Ausztriával egy fokra emelte a szabadalmi ügyet és orgánumot alapított a feltalálók tájékoztatására. Így mód nyílt arra, hogy itthon is szabadalmaztatni lehetett a találmányokat.

Keleti politikájának tudott a legnehezebben barátokat szerezni, ebben a törekvésben talált a legtöbb ellenzőre. Politikáját a következőképpen fogalmazta meg: „Magyarország, mint közvetlen szomszédja a nyugati műveltséget felvenni kezdő Balkán-félszigeti államoknak, első sorban hivatva van, hogy ezeknek vezérévé váljék, s közvetítő legyen köztük és nyugat Európa között, s hogy míg elősegíti a Balkán-félszigeti országok gazdasági fejlődését, közvetlenül előmozdítja saját anyagi érdekeit is."

Az első modern úttörvény megalkotásával jelentős mértékben járult hozzá a gazdasági élet fejlesztéséhez. Kötelezően bevezette a német helyett a magyar nyelvű levelezést a vasútnál. Az államvasutak kielégítő jövedelmezősége feleslegessé tette ennek állami támogatását. Baross ugyanis megreformálta a vasúti személydíjszabást. A zónarendszer behozásával a tengerentúl figyelmét is Magyarországra vonta. Ezáltal egyrészt az egymásra utalt vidék forgalmát is növelte, másrészt a legtávolabbi helyeket is a fővároshoz fűzte. A személyszállítási áraknak e merész leszállítása Barossnak már régi álma volt. A magyar állam pénzügyi viszonyai azonban nem engedték meg, hogy a vasút ne legyen jövedelmező, ezért Baross kénytelen volt a legmagasabb árakat elfogadni a zóna-tarifarendszerben. Ez a vasutak jövedelmét úgy megsokszorozta, hogy a magyar vonatokat európai színvonalra tudták fejleszteni.

Baross kimagasló közgazdasági érzékével egész jelentőségében felismeri a posta és a távíró egyesítésének feltétlen szükségességét és gazdasági előnyeit. Átlátja, hogy az évről évre hiánnyal záró távírószolgálat a posta üzleti eredményeire támaszkodva és műszakilag is a kor színvonalára emelve, a jobb szolgáltatással hasznot hozó üzemmé emelhető. Mellékgondolata volt, hogy az egyesítés az államháztartásban is megtakarításokat fog eredményezni. A vasminiszter még 1887-ben rendkívül gyorsan végrehajtotta a két intézmény egyesítését, és ezzel alapot teremtett a távíró és távbeszélő fejlesztéséhez. Az elgondolás helyességét az első év üzleti eredménye is igazolta.

1887 őszén a budapesti távbeszélő-hálózat, majd a vidéki hálózat megváltására került sor. Bár a budapesti hálózatot a régi tulajdonosoknak mindjárt bérbe is adták az állam felügyeleti jogának biztosításával, mégis a megváltás a hálózatok fejlődésére jótékony hatással volt. A bizalom a távbeszélő-hálózat irányába erősödött, ami maga után vonta az előfizetők számának rohamos növekedését. Az egyesítés végrehajtásakor az országos távíróhálózat műszakilag kezdetleges fokon állt, nem szólva a gyermekkorát élő távbeszélőről. Baross felismerte a gyors hírközlést biztosító távíró és távbeszélő rendkívül nagy jelentőségét, és elhatározta, hogy ezeket minden eszközzel korszerű műszaki színvonalra fejleszti. Első miniszteri tevékenységei közé tartozott, hogy két postamérnök állásra 1887 tavaszán pályázatot hirdetett azzal a célzattal, hogy a távírónak a postával való egyesítése és a távbeszélő-hálózatok megváltása után szakemberek vezetésével, állami kezelésben indulhasson meg a felsorolt intézmények tervszerű műszaki üzemvitele és fejlesztése. A Közmunka- és Közlekedésügyi Minisztérium 22.119/1887. V. 3. számú ügyirata szerint: „A posta és távírda üzleti és műszaki szolgálatának magasabb színvonalra emelése ... czéljából elhatároztam, hogy a vezetésem alatt álló ministérium posta és távírda szakosztályában 3 – f. évi október 1-jétől kezdve pedig minden egyesített posta és távírda igazgatóságnál 1-1 műegyetemet végzett műszaki közeget alkalmazok – Baross."

A mérnökök feladata volt a postaszolgálat ágazataiban adódó általános műszaki kérdések szakszerű megoldásán kívül a távíró és távbeszélő fejlődésének éber figyelemmel kísérése és értékelése is. A tanulmányok irányának és keretének biztosítása végett a közlekedésügyi miniszter részletesen előírta a fejlesztési tervet, melynek célkitűzései:

- A távíróhivatalok számának növekedése.
- A vonalhálózatok bővítése.
- A megnövekedett forgalom miatt újabb Hughes gépek üzembe állítása.
- Közvetlen összeköttetések teremtése a fontosabb bel- és külföldi gócok között.

Az 1888. évi XXXI. törvény megalkotása nagy lépéssel vitte előre a távíró, és főképpen a távbeszélő elterjedését. E törvény a távíró-, távbeszélő- és villamosjelző-berendezések építését az állam fenntartott jogának minősíti, és kimondja, hogy az ingatlantulajdonosok a hálózatok támszerkezeteinek felállítását ingatlanaikon kötelesek ellenszolgáltatás nélkül tűrni. 1890-ben a miniszter elrendeli a légvezetékek átépítését fali tartókról tetőtartókra, ezenkívül a telefonközpontokban az áttérést a szám szerinti hívásra.

A műszaki szolgálat csak kifogástalan anyagokkal valósítható meg. A felhasználandó anyagok minőségét szakszerű anyagvizsgálattal kell megállapítani. A vizsgálatok rendszeres végzésére 1891-ben mérőszobát rendeznek be a főposta épületében (49 100/1890. számú kereskedelemügyi miniszteri rendelet). Vezetésére Kolossváry Endre postafőmérnök kapott megbízást. Ez a volt Posta Kísérleti Állomás csírája. Az elsőként felvett 12 mérnök közül 3 a minisztériumban szervezett

posta és távírda szakosztályon kap beosztást. A későbbiekben közülük előbb Kolossváry Endre, majd Vater József lett a PKI elődintézményének, a Kísérleti Állomásnak az igazgatója. Baross a 6297/1891. számú ker. min. rendeletével elrendeli a főposta épületében elhelyezett mérőszobák működését, és szabályozza a kísérleti állomás munkakörét. A minisztérium 1892 szeptemberében rendelte meg a Posta és Távírda Gazdászati Hivatalánál a Kísérleti Állomás bélyegzőjét a következő felirattal:

„M. Kir. posta- és távírda Kísérleti Állomás Budapesten."

A halál dicsősége és működése csúcspontján találta: 1892 márciusában a vaskapui szabályozás műveleteinek megvizsgálásának alkalmával meghűlt, és rövid betegség után, mellhártyagyulladásban meghalt (1892. május 8-án). Az utolsó nagyszabású alkotások és tervek közül, melyek élete utolsó napjaiban foglalkoztatták, a Vaskapu szabályozása volt neki a legkedvesebb, a legtöbbet gondolt erre s ezt szerette volna még befejezni, s azért sokszor maga is ellátogatott a Vaskapuhoz.

Emlékszobrát a budapesti Keleti pályaudvar előtt 1898. november 20-án leplezték le.

Irodalom:

1. Petrovics László: Bellusi Baross Gábor volt magyar kereskedelmi miniszter élete (1892).
2. Postamérnöki szolgálat 50 éve, Magyar Távközlési Vállalat, 2. kiadás, az eredeti alapján, Budapest, 1990.
3. PKI 100 éve, Távközlési Könyvkiadó, Budapest, 1991.
4. Révai Nagy Lexikona, II. köt., 639. old.

Hírek

Világ-csúcstalálkozó az információ társadalomról

Az ENSZ közgyűlése határozatában üdvözölte a világ-csúcstalálkozó megrendezését az információs társadalomról. A két részben, 2003-ban Genfben és 2005-ben Tunéziában az ENSZ főtitkárának védnöksége alatt megrendezendő csúcstalálkozó a digitális megosztottság kérdéseivel foglalkozik. Kofi Annan főtitkár utalt arra, hogy a világ lakosságának többsége még nem húzott hasznot az információs és távközlési technikából, ami pedig, „ha jól irányítják és használják ki, akkor társadalmi, gazdasági és kulturális életüket minden szempontból magasabb szintre tudja felemelni".

A rendezvény megszervezésével az ITU-t bízták meg. Felkérték a nemzetközi közösséget, hogy önkéntes adományukkal töltsék fel az e célra létesített külön pénzalapot, aminek fő feladata a fejlődő és különösen a legkevésbé fejlett országok megfelelő részvételének biztosítása.

További részletek a www.itu.Internet/newsroom/press_releases/2002/UNGA-res-e.html címen találhatók. (H. Gy.)

❖ ❖ ❖

A Sun Microsystems Sun Cobalt QubeTM 3 szerverkészüléke az otthoni hálózatokat és a kisvállalatokat célozza meg, lehetőséget biztosít az otthoni irodák és kisvállalatok számára, hogy működtessék saját weblapjukat, hálózatban üzemeltessék gépeiket, továbbá megvédjék számítástechnikai környezetüket az internet veszélyeitől (például a számítógépes kalózoktól vagy a vírusoktól).

A veszélyek elleni védelem biztosítása érdekében a szerver beépített tűzfalat tartalmaz, és támogatja a virtuális magánhálózatokat (VPN) is. A tűzfal konfigurálható a személyes adatok védelmére, ugyanakkor a távoli telephelyeken tevékenykedő szakemberek számára a vállalati hálózat biztonságos elérését gyakorlatilag bármilyen Windows gépről.

A Sun Cobalt Qube 3 szerverberendezés általános célú szervek képességeit kínálja, böngészőalapú grafikus felhasználói felülete (GUI), előre telepített és konfigurált alkalmazásai, beállítható hálózati szolgáltatásai megkönnyítik üzemeltetését és felügyeletét.

Hírek

A Siemens innovatív termékekkel jelenik meg az idei CeBIT 2002 kiállításon. Valamennyi terméket, a mobileszközöktől és az ehhez kapcsolódó infrastruktúráktól kezdve az üzleti életben alkalmazott elektronikus és mobilmegoldásokon át a vállalatok és hálózati kiszolgálók számára értékesített IP konvergenciarendszerekig és a „Next Generation Network" nevű hálózati megoldásig a befektetések gyors megtérülése és a magas szintű hozzáadott érték szempontjainak figyelembevételével optimalizáltak. A fő hangsúly az IP alapú konferenciatechnológiákon, valamint a széles sávú hozzáférést biztosító, és a fényhálózati technológiák termékcsaládjain van.

A Gamma wave a 868 MHz-es frekvenciasávon működő épületfelügyeleti rendszer adó- és vevőfunkcióra is alkalmas. Ezáltal nagyobb biztonság érhető el, például az állapotinformációk átvitele biztos. A Gamma wave révén olyan távműködtetés és fogyasztási adatrögzítés valósítható meg, mint például a világítás kapcsolása, fényerő szabályozása vagy az árnyékoló szerkezetek mozgatása.

A Delta line szerelvénycsaládot a kapcsolók és dugaszolóaljzatok harmonikus aránya, tiszta kontúrok és szokatlanul lapos kivitel jellemzik. A Delta line nagy szilárdságú, halogénmentes anyagból készül.

❖ ❖ ❖

Az IDC a UNIX piac jövőjéről

A Sun Microsystems az elmúlt 18 negyedév alatt vezeti a UNIX szerverek piacát. Mind több vállalat, képviselve az iparágak összességét, választja működéskritikus vállalati rendszerek kiépítését.

Az IDC hiszi, hogy a UNIX-szerverek továbbra is a középkategóriás szerverpiac gerince marad, platformot biztosítva a webes, valamint a létfontosságú üzleti alkalmazásokhoz. Arra alapozva, hogy sok éven át sikeresen alkalmaztak Unix-szervereket a komoly kihívásokat támasztó, létfontosságú alkalmazások terén, és e rendszerek továbbra is szolgálják a világ legnagyobb vállalatait. Az internet is Unix-szerverekre és Unix-os hálózati technológiákra épült. Ahogy a gazdaság helyreáll, és a nagy cégek újra növekvő mértékben költenek az informatikára, a Unix-szervereladások megugranak 2003-ban.

❖ ❖ ❖

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM Országos Műszaki Információs Központ és Könyvtár 2001

Szélesebb körben támogatja a Sun a Linuxot

A Sun bővítette a Linux operációs rendszer támogatását. Először is, a Sun a Linux operációs rendszer teljes implementációját fogja szállítani. Másodszor, jelentősen bővíti a Sun Cobalt Linusot futtató célszervereinek családját, sőt, egy új, általános célú, Linux/x86 alapú számítógépcsaláddal jelenik meg. Végül, intenzívebben kíván részt venni a Linux közösség életében azáltal, hogy ingyenesen biztosítja a Solaris (tm) operációs környezetszoftver egyes kulcselemeit.

A linuxos alkalmazások bármely Solaris alapú rendszeren futtathatók. Ehhez beépített Linux-kompatibilitással szerelik fel a legfontosabb programozási felületeket, parancsokat, segédprogramokat és felhasználói környezeteket. Készül egy Linux kompatibilitás-biztosító eszköz (LinCAT) is, amely egyszerűsíti a Linux-alkalmazások a futtathatóságának ellenőrzését.

❖ ❖ ❖

Az Ericsson és a Tele2 fix-telefonhálózatra vonatkozó ENGINE-szerződést írt alá

A norvég Tele2 az ENGINE-nel jelenlegi áramkörkapcsolt hálózatát nyílt, rétegzett, csomagkapcsolt hálózattá alakítják. Az Ericsson szállítja majd mind a hardvert, mind a szoftvert.

Az ENGINE hálózatot Oslóban, Bergenben, Trondheimben és Bodø-ban helyezik üzembe.

Az új hálózat révén megkezdhető az egyszolgáltatásos hálózat többszolgáltatásos, IP alapú hálózatokká alakítása.

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM * Oktatási Műszaki Információs Központ és Könyvtár 2001

Contents

Preface to the February Issue 1

SOFTWARE TECHNOLOGY

András Kovács and Bálint Pete
Testing of IPv4-IPv6 Migration Technologies 3

Zoltán Andriska, Gábor Bátori, Antal Wu-Hen-Chang and dr. Gyula Csopaki
Automatic Testing Selection on the Basis of Formal Specifications 11

Roland Gecse, Zoltán J. Szabó, Tibor Csöndes
Timed Finite Automatic Based Testing Method 19

DEVELOPMENT OF NETWORKS

Péter Füzesi, Attila Vidács
Game Theory Approach to Network Dimensioning Questions 25

E. Grampín, J. Rubio, N. Vardalachos, A. Galis, J. Serrat
Implementation Issues of Authorization based Network Managing Systems 31

Ladányi Zsuzsanna, Szász András
Monitoring and Billing on DiffServ Networks 37

Biró Péter, Lója Krisztina, Sütő Márton
Designing and Testing Distributed Frequency Allocation Algorithms 43

Juhász Ákos, Ulrich Ferenc, Dr. Eged Bertalan, Kubinszky Ferenc
Wireless Network Performance Testing 49

TELECOMMUNICATION POLICY

Mrs. Imréné Kozma
The Future of Hungarian Televising 57

Gyula Horváth
Crisis Survival 61

László Sipos
Conference on „Quality Assurance in Nuclear Energetics" – Improvement of Security Culture by Efficient Quality Assurance 65
Tamás Roska Member of the Academy – Winner of the 2002's Bolyai Prize 67

Havaska Beatrix Nagy
Gábor Baross – The Iron Minister 69

Szerkesztőség
HTE Budapest V., Kossuth L. tér 6–8.
Tel.: 353 1027, Fax: 353 0451
e-mail: hte@mtesz.hu

Hirdetési árak:
1/1 (205 x 290 mm) 4C 120 000 Ft + áfa
Borító 3 (205 x 290 mm) 4C 180 000 Ft + áfa
Borító 4 (205 x 290 mm) 4C 240 000 Ft + áfa

Cikkek eljuttathatók az alábbi címre is
BME Mikrohullámú Híradástechnikai Tanszék
Budapest XI., Goldmann Gy. tér 3.
Tel.: 463 1559, Fax: 463 3289
e-mail: zombory@mht.bme.hu

Előfizetés
HTE Budapest V., Kossuth L. tér 6–8.
Tel.: 353 1027, Fax: 353 0451
e-mail: hte@mtesz.hu

2001-ES ELŐFIZETÉSI DÍJAK
Hazai közületi előfizetők részére
1 évre bruttó 30 000 HUF

Hazai egyéni előfizetők részére
1 évre bruttó 6 000 HUF

Subscription rates for foreign subscribers
12 issues 150 USD, single copies 15 USD

www.hte.hu

Felelős kiadó: MÁTÉ MÁRIA
Lapmenedzser: Dankó András

Design by: Kocsis és Szabó Kft.
HU ISSN 0018-2028

Printed by: Regiszter Kft.