

Csomagkapcsolt hálózatok kommunikációs protokolljainak optimalizálása neurális algoritmusokkal



Karlócai Balázs

Pázmány Péter Katolikus Egyetem

Információs Technológiai Kar

Multidiszciplináris Műszaki Tudományok Doktori Iskola

Tézisfüzet a Ph.D. disszertációhoz

2013. tavasz

Témavezető: Dr. Levendovszky János MTA Dr.

1. Bevezetés

Napjainkban a nagy adatátviteli sebesség igénye, ugyanakkor a limitált erőforrások újabb és újabb kihívások elé állítják a kommunikációs technológiákat. Az elmúlt tíz év hálózati fejlődését vizsgálva az tapasztalható, hogy a legnagyobb kihívás a minőségi kommunikáció, a Quality of Service (QoS) biztosítása véges erőforrások (pl. energia, processzási képesség, sávszélesség) mellett. Ez általánosan a hálózat tervezését és optimális működtetését, mint kényszeres optimalizálási feladatot veti fel: pl. vezeték nélküli szenzorhálózatok esetén keressük azt az előírt minőségű útvonalat, amely minimális energiafogyasztást igényel a csomag továbbadásában résztvevő node-októl. Hasonló feladatok megfogalmazhatóak mind az Internet Protokoll (IP), mind a vezeték nélküli szenzorhálózatok (Wireless Sensor Network - WSN)[1], mind a testközeli vezeték nélküli hálózatok (Wireless Body Area Network - WBAN)[2, 3] terén. Ezek közül a fő problémák a következők:

- Az elmúlt évek alatt a multimédiás kommunikáció (VoIP, Video streaming) nagy teret hódított a teljes internetes forgalomban, azonban ezek a szolgáltatások előírt minőséget igényelnek. Ez hangsúlyozza a routereken működő csomagosztályozás feladatának fontosságát, mert különböző szolgáltatási minőséghez tartozó csomagokat különböző módokon kell kiszolgálni, és a kiszolgáláshoz tartozó akciókat real-time módon kell elvégezni. Így az IP esetében nagyon fontos kérdéskörre vált a csomagosztályozás (packet classification - PC) [4], mely alapja többek között a QoS szolgáltatásnak is [5]. Tekintettel arra, hogy az Internet központi routereiben kiemelkedően fontos a csomagosztályozást elvégezni, ez könnyen szűk keresztmetszetet képezhet a kommunikációban. A csomagok priorizálására az IPv6 (Internet Protocol version 6) már specifikációjában is lehetőséget biztosít, itt a címinformáció mellett a csomagfejlécben QoS-jellegű, valamint a származási helyre jellemző adatok találhatóak. Így lehetővé válik fire-wall és QoS alkalmazások megvalósítása is. Ez a probléma arra egyszerűsíthető, hogy hogyan lehet a csomagok fejlécében felismert információ alapján gyors akciót vagy osztályba sorolást elvégezni. Ez algoritmikusan is mélyebb feladatokhoz vezet, amelyek tradicionálisan az ún. "computational geometry" tárgykörébe esnek (lásd [6]).

- A vezeték nélküli szenzorhálózatok esetében kiemelkedően fontos az energiafelhasználás optimalizálása. Ismert az, hogy a Rayleigh fading-modell [7] alapján milyen valószínűséggel történik a csomag sikeres vétele adott energiájú küldés és adott távolság esetén. Mivel adott megbízhatóságú direkt (egy hop-os) csomagtovábbítás a bázisállomástól távolra nagy energiákat igényelne [8], ezért Multi-Hop kommunikációs modellt kell használnunk [9], amelyben a küldő node és a bázisállomás között több "relay" (közvetítő) node helyezkedik el. Itt viszont felmerül a kérdés, hogy hogyan lehet minimalizálni azokat a küldési energiákat a Multi-Hop láncon belül, amelyek az aztán adott megbízhatóságot (a csomagnak a bázisállomásra történő adott valószínűségű megérkezését) eredményezik. A minimális energia egyértelműen az élettartam meghosszabbítását jelenti [10, 11].
- A nagy sávzélességű, különböző QoS paraméterű hozzáférési hálózatok megjelenésével ezeknek a hálózatoknak a kiszolgálását végző eszközök, node-ok új kihívásokat támasztanak a méretezés és teljesítőképesség terén [12, 13]. Ezeknek az rendszereknek közös tulajdonsága, hogy fa-szerű topológiába szerveződnek, ahol a belső node-ok további kapacitási korlátokkal rendelkeznek, valamint hogy a shared-bus architektúrának köszönhetően a rendelkezésre álló kapacitás az aktuális feltöltési és letöltési forgalmon múlik [14]. Ezért a legfontosabb cél egy olyan új, többnode-os méretező módszer kifejlesztése, amely képes kezelni további kapacitási korlátokat, valamint a kétirányú forgalom hatását. Ezen módszer segítségével képesek vagyunk megállapítani, hogy ténylegesen hány node-ra és milyen link kapacitásokra van szükség ahhoz, hogy egy adott számú felhasználót megadott cellavesztési valószínűség mellett kiszolgáljunk.

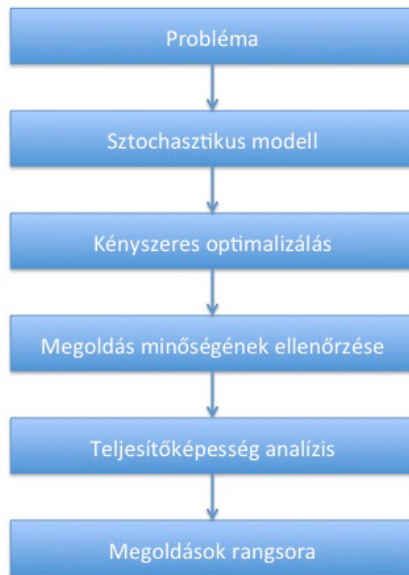
A fenti, látszólag különálló csomagkapcsolt hálózati tématerületeket a kombinatorikus optimalizálás igénye kapcsolja össze.

2. Alkalmazott módszerek

A következőkben bemutatásra kerülő tézisek mindegyikénél az alább leírt módszereket alkalmaztam a tudományos eredményeimhez:

- felállítottam a sztochasztikus modelleket
- optimalizálási módszereket alkalmaztam az adott kommunikációs rendszer teljesítőképességének, mint célfüggvénynek az optimalizálására
- az optimalizálásra numerikus vagy analitikus módszereket használtam
- a kapott megoldás minőségét szimulációkkal vizsgáltam, amelyekkel az egyes kommunikációs protokollok között rangsort sikerült felállítani

A kutatás módszertanát az 1. ábra illusztrálja:



1. ábra. Kutatási módszertan

A fenti módszerekhez használt apparátus a következő elemeket tartalmazta:

- valószínűség-számítás
- sztochasztikus folyamatok
- kombinatorikus optimalizálás
- randomizált szimulációk

3. Technológiai Motiváció

A dolgozat három fő témakörének részletes technológiai motivációit az alábbi alfejezetekben tárgyalom.

Ahhoz, hogy a felhasználók csomagjai a lehető leggyorsabban és legkisebb hibával célba érjenek, a routereken a csomagok hibamentes feldolgozására és továbbításra alkalmas algoritmusokat kell telepíteni.

Ezek az algoritmusok a csomagok fejlécének analízise alapján eldöntik, hogy a router melyik output portjára kell továbbítani (address lookup), illetve hogy milyen egyéb processzálás szükséges (pl. csomagok szűrése tűzfal alkalmazása esetén, vagy prioritásos csomagkezelés adott QoS kritériumoknak eleget téve)[5].

3.1. IPv6 és a csomagklasszifikáció

Napjainkban a routerek működésének nagy része pusztán egy továbbító funkció, melynek irányát a beérkező adatsomag fejléce határozza meg, ugyanakkor az IPv6-ban ennél jóval bonyolultabb csomagosztályozásra is szükség van. A feladat algoritmikus kihívása nemcsak a megvalósítandó csomagosztályozási feladatok sokrétűségében, hanem ennek a sebességében is rejlik, hiszen szélessávú adatfolyamokon kell végrehajtani.

Az elmúlt években az Internethez hozzáférő felhasználók száma, valamint az ezen felhasználók által igénybevett sávszélesség drámai mértékben megnőtt [15]. A küldött és fogadott adatok mennyisége és minősége számtalan szolgáltatás biztosításának lehetőségét igényli a routerektől. Hiszen nem mindegy, hogy egy videókonferencia adatait, egy mp3-letöltést, vagy egy email-továbbítást milyen sorrendben és prioritással kezel a router; a végrehajtás sorrendje eltérő lehet a beérkezett igények sorrendjétől, holott a router feladata elsődlegesen annyi, hogy egyre közelebb és közelebb kerüljön a csomag a címzetthez.

Ahhoz, hogy a routerek el tudják dönteni egy csomagról vagy beérkezett igényről, hogy milyen speciális módon kell kezelni, szükség van a célállomás címén kívül többlet-információra. Ez jelentősen lelassíthatja a továbbítást, hiszen nagyobb mennyiségű adatot kell a routernek feldolgoznia. Egy másik nehézség abból fakad, hogy a felmérések szerint az Interneten áthaladó csomagok 75%-ának mérete kisebb, mint egy átlagos TCP-

(Transmission Control Protocol)- csomag (522 byte), és döntő részük csupán 40-50 byte nagyságú [16]. Tekintve, hogy a használt sávszélesség több gigabites léptékűre nőtt, és a router által használt memória elérési ideje is szab egy időkeretet, jól látható, hogy a routerben dolgozó algoritmusnak szűk időérés alatt kell precíz döntést hoznia az áthaladó csomag jövőjét illetően. Új megoldás szükségességét a következő példával illusztrálnám: vegyünk IP-csomagokat 40 byte méretben, a router portjait vegyük 10Gbit/s sebességűnek, és számoljunk 10 porttal. Ezzel a felállással a legrosszabb esetben 3.2 nsec alatt kell döntést hoznunk a csomag jövőjét illetően. A DRAM átlagos elérési sebessége is ebben a tartományban található, 3-5 nsec. Ez jelzi, hogy szükséges gyors csomagklasszifikációs algoritmusokat használni a megfelelő átvitel biztosításának érdekében, mivel gyorsabb hardverre egyre kevésbé lehet számítani.

3.2. Megbízható csomagtovábbítás vezeték nélküli hálózatokban

Az elmúlt években a vezeték nélküli szenzorhálózatok alkalmazása széles körben elterjedt, ezért, tekintettel a szenzorok limitált energiakészletére, szükségessé vált olyan kommunikációs protokollok kifejlesztése, amelyek energiafelhasználás-érzékenyek [8]. Ez a kérdéskör különösen előtérbe került az olyan helyzetekben, ahol az elemek feltöltése nem lehetséges. Erre jó példa a testbe implantált szenzor [17], amelynél a legfontosabb az, hogy a meglévő energiakészletet minél tovább használhassuk. Tekintettel arra, hogy a szenzorban a legnagyobb energiafelhasználása a rádióadónak van, az új kommunikációs protokollok leginkább a küldési energiára optimalizálnak [18]. Ez a feltétel különösen igaz az ún. multihop-kommunikációban, ahol az egyes csomópontok, annak érdekében, hogy a bázisállomásra juttassák az információjukat, igénybe veszik a többi szenzort is. A sikeres csomagtovábbítás valószínűsége egy adott fading modell segítségével írható le (pl. Rayleigh fading) [7], így a multihop-kommunikáció során csomagvesztés léphet fel [9]. A feladat tehát olyan útvonalválasztó algoritmus kifejlesztése, amely garantálja, hogy az útvonalon a csomagvesztés valószínűsége egy előírt értéknél kisebb, ugyanakkor a csomagtovábbításban résztvevő node-ok energiafogyasztása minimális (a hálózat élettartamát maximalizáljuk) [10, 11].

Jelenleg az energia-tudatosságot a következő protokollok biztosítják:

- Energy Conserving Routing[19];
- LEACH [10];
- PAMAS [20].

A LEACH eredetileg valóban 2000-ben lett publikálva, de azóta (2005-2009) több továbbfejlesztést publikáltak hozzá [? ?]. A PAMAS algoritmus a legkorábbi (1998) algoritmus az említettek közül, ennek csak történetiség szempontjából van jelentősége. A ECR a LEACH eredeti verziójával egy évben jelent meg, ebben az irányban kevesebb továbblépés történt.

Az említett algoritmusok az energia-optimalitás mellett nem garantálnak előírt megbízhatóságot, ezért kihívás olyan új protokollok tervezése, amely az energiatudatosság mellett az előírt megbízhatóságot is garantálják.

3.3. Hálózat-dimenzionálás csomagkapcsolt hálózatokban

Csomagkapcsolt kommunikációs protokollok esetében a beengedett forgalom maximalizálása mellett nagy fontossággal bír az előírt szolgáltatási minőség biztosítása. Ez konkrét hálózati technológiák (pl. ATM) esetén a hálózattervezést (dimenzionálást) egy kényszeres optimalizálási feladatra képezi le, ahol a cél minimális hardverigényű hozzáférési architektúrák tervezése adott cellavesztési valószínűség mellett. Ezért a tézis eredményei a csomagkapcsolt hálózatok hozzáférési architektúrájának komplexitását optimalizálja a minimális költségű hardver megvalósítása érdekében. Ugyanakkor az architektúrának az előírt cellavesztési valószínűséget kell biztosítania. A feladat tehát olyan dimenzionálási eljárás kidolgozása, amely képes az előírt felhasználói igények hozzáféréseinek hatékony menedzselésére.

Téziscsoport 1

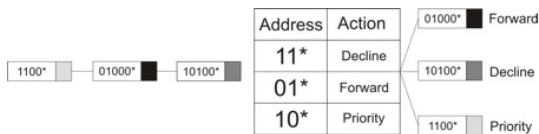
4. Csomagklasszifikáció IP-hálózatokban

4.1. A csomagosztályozás formális modellje - packet classification-táblák

A csomagosztályozás feladata az, hogy a beérkező fejlécek alapján a csomaghoz kapcsolódó akciókat (tevékenységeket) jelöljön ki, pl. firewall esetén tiltott címről érkező csomag eldobása, VoIP esetén prioritás biztosítása. Ez a feladat a következőképpen formalizálható:

- q_1, \dots, q_D jelöli a csomag fejlécekében megfigyelhető bináris stringeket;
- adott a fejléceken értelmezett logikai függvényeknek egy f_1, \dots, f_L halmaza, valamint minden egyes függvényhez egy A_1, \dots, A_L akció;
- találjuk meg a lehető leggyorsabban azokat a logikai függvényeket i_1, \dots, i_l , amelyekre $f_{i_j}(q_1, \dots, q_D) = TRUE, j = 1, \dots, l$
- ezek közül kiválasztva a legnagyobb prioritásút $v \in i_1, \dots, i_l$, hajtsuk végre A_v -t.

Az előző definíció alapján a PC működését a következő ábra szemlélteti:



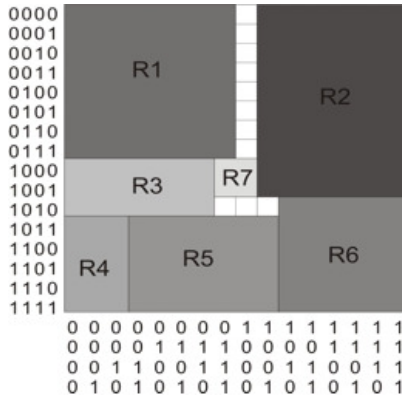
2. ábra. A PC-táblák működése

A gyakorlatban a logikai függvények általában maszkolásként jelennek meg, ezért teljesülésük igazából illeszkedést ("matching") jelent. A legnagyobb prioritása a leghosszabb illeszkedésnek van. Egy gyakorlati PC táblát az 4.1 sz. táblázat szemléltet.

Rule	F1	F2
R_1	00*	00*
R_2	0*	01*
R_3	1*	0*
R_4	00*	0*
R_5	0*	1*
R_6	*	1*

1. táblázat. PC-tábla felépítése

Az előzőeknek megfelelően a PC mint geometriai feladat is interpretálható (lásd [10, 11, 12]). Jelölje Q_i azt a halmazt, amely az A_i akcióhoz tartozik, nevezetesen $Q_i = \{q_1, \dots, q_D : f_i(q_1, \dots, q_D) = TRUE\}$. Ez megfigyelve a csomagfejléctet q_1, \dots, q_D , ez meghatároz egy D dimenziós pontot $r = (q_1, \dots, q_D)$, ahol minden koordináta a megfelelő bináris formában adott. A feladat a lehető leggyorsabban megtalálni azt a Q_i halmazt, amely tartalmazza r -et, azaz $r \in Q_i$. Ez alapján az A_i akció elvégezhető.



3. ábra. A PC képfeldolgozási módszerrel

A 3. ábrán egy byte geometriai reprezentációja található. Jól látható, hogy egy szabály egy területet definiál, egy beérkező IP pedig egy pontot. Feladatunk a pont helyzetének meghatározása a síkon. Ennek a problémának kiterjedt irodalma van (lásd [13, 14, 15]).

4.2. A csomagklasszifikáció mint képfeldolgozási feladat – csomagklasszifikáció Cellular Neural Network (CNN) segítségével

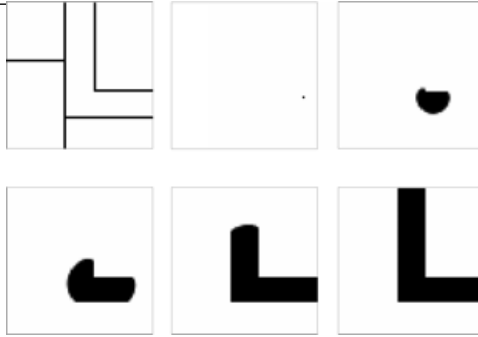
Mint ahogy az előzőekben már szó esett róla, a csomagklasszifikáció felfogható egy geometriai feladatként, ahol a beérkező csomag fejlécmezeje által meghatározott pontnak egy adott szabállyal fémjelzett halmazba való tartozását kell polinomiális komplexitásban kimutatni. Azaz, ha a szabálycsomagokat a fentiek szerint egy geometriai ábrába rendezzük, ahol minden terület egyértelműen meghatároz egy szabályt, akkor csak azt kell kiolvasnunk, hogy a beérkező cím melyik területre esett. Így a probléma egy képfeldolgozási feladatként is interpretálható: ki kell "színezni" azt a halmazt, amelybe a kurrens csomag fejléce által meghatározott pont esik. Ez az interpretáció azért lényeges, mert amennyiben valós idejű képfeldolgozásra nyílik lehetőség (pl. egy CNN-hálózattal), a csomagklasszifikáció processzálási ideje jelentősen felgyorsítható (ami az egyik szűk keresztmetszete a jelenlegi routing technológiának, és így az IPv6 alapú hálózati kommunikációnak is).

Tézis 1.1 *Bizonyítottam, hogy a csomagklasszifikációt CNN-architektúrával megoldható a következő template-ek segítségével*

A fenti probléma megoldható egy CNN-architektúrával [21, 22, 23, 24], ahol az egyes képpontok (fekete/fehér) állapotának egy neuron bináris kimenete felel meg, valamint a beérkező csomag fejlécének megfelelő pontból hullámokat triggerelünk, amik az adott tartomány határáig "kiszínezik" a régiót (azaz a régióban fekvő összes neuron kimenete aktívvá válik a stacionér állapotban). Mivel a hullámterjedésnek megfelelő tranziens ideje a mikroszekundumok tartományába esik, a csomagklasszifikáció műveleti ideje is ebben a tartományban mozog, ami lehetővé tenné a Mbit/sec-os sebességű csomagfeldolgozást a routerekben. A megoldás intuitív képét a 4. ábra mutatja.

Ahhoz, hogy a csomagklasszifikációt CNN-re lehessen implementálni, a következő feladatokat kell megoldani:

- a szabályok egyértelmű leképezése egy 2D-s térbe (még akkor is, ha több mint kettő fejlécmezőt kell vizsgálni);
- a szabályoknak megfelelő halmazkontúrok elhelyezése a CNN-en;

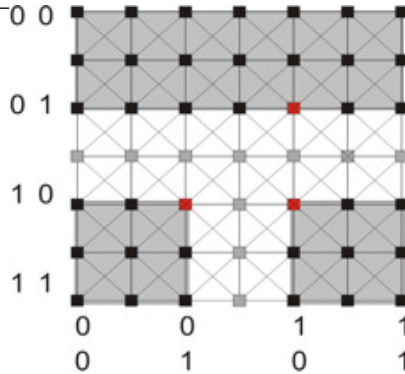


4. ábra. A hullámterjedés egy egyszerű szabályhalmazon ($\tau = (0, 3, 6, 12, 20)$)

- a csomagfejlécnek mint pontnak "elhelyezése" a raszteren;
- hullámok generálása adott template szerint;
- a "beszínezett" rész kiolvasása.

A probléma CNN-es megvalósítása a következő geometriai leképezésen alapul: A jobb átláthatóság érdekében celláink számozását $C(i, j) = C(0, 0)$ -tól kezdjük. A kép pontjait két halmazra szeparáljuk, minden $u_{ij} : (i \wedge j = \text{odd})$ pixel tartalmazza a bejövő IP-csomagot, míg a többi $y_{ij} : (i \vee j = \text{even})$ pixel tartalmazza a szabályokat. A bejövő IP-cím egyértelműen meghatároz egy darab cellát a leképezésen, míg egy szabályhalmaz egy területet definiál. Feladatunk annak a területnek a kiolvasása, mely tartalmazza azt a cellát, amelyet az IP-cím meghatároz. Ennek a kézenfekvő módja – figyelembe véve a CNN előnyeit – az, hogy az IP-cím által meghatározott cella hullámokat indít, amelyek átszínezik a szomszédos cellákat, egészen addig, amíg a szabályok által meghatározott határfelületet el nem érik [25]. Minden körbehatárolt területen definiálunk egy $r_t \in u_{ij} (i \wedge j = \text{odd})$ referenciapontot. A referenciapont átszíneződéséből tudjuk egyértelműen kiolvasni, hogy melyik terület aktív.

Ennek a megoldásnak az alkalmazásánál a következő problémába ütközünk: ahhoz, hogy a csomagklasszifikációt egy chipen végezzük, szükségünk van a chipen belül az összes IPv6-os cím kétszeresének megfelelő számú cellára. A legnagyobb CNN-chip jelenleg



5. ábra. A referenciapontok elhelyezése a CNN területén

128X128 méretű (ACE16K, QEye), ugyanakkor a megoldási javaslat a CNN méretére vonatkozó igénye $2^{\lceil W/2 \rceil} \times 2^{\lceil W/2 \rceil}$. Tehát kénytelenek vagyunk a címfelismerést szétdarabolni, például byte-okra. Ehhez két dolgot kell tennünk:

- A bejövő IP-címet byte-okra daraboljuk, így egy chip minden byte-ja egyértelműen meghatároz egy cellát. Ezek fogják indukálni a színhullámokat.
- A meglévő szabályainkat is feldaraboljuk. Egy szabály egy területet fed le a chip felületén, kivéve, ha a szabály egy byte-nál hosszabb (pl.:10101010111*); ez esetben ugyanis az első chipen ez a szabály is csupán egy cellát fog jelölni, a következőn pedig egy területet.

A szabályok átvitele geometriai síkra a következő módon lehetséges:

Az n -ik bytehoz tartozó CNN-chipen kialakítjuk a következő struktúrát minden olyan R_k -val, amely értelmezve van azon a byte-on:

$y_{i,j} = 1$, ha \exists olyan $p, q \in \{-1, 0, 1\}$ hogy $i + p$ és $j + q$ páros és $IP_{i+p,j+q} \in R_k[n]$, és $IP_{i-p,j-q} \notin R_k[n]$, egyébként $y_{i,j} = -1$.

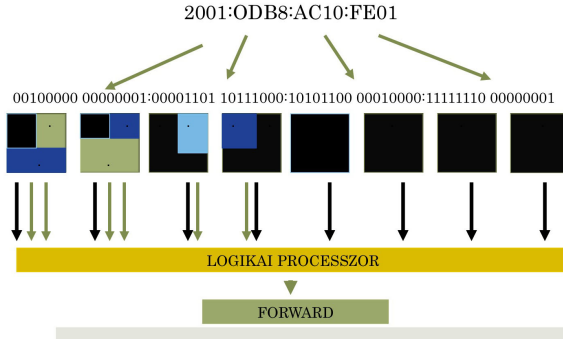
Ahol $IP_{i,k}$ tartalmazza az értelmezett bejövő IP-csomagok bináris értékét a 32x32 cella méretű CNN felületén, a következő módon: $IP_{i,k} = [i(1) j(1) i(2) j(2) i(3) j(3) i(4) j(4)]$ nyolcbites értéket, ahol az $i(x)$ bináris értékének vett x -ik bitjét jelenti.

Egy olyan CNN-template-re van tehát szükség, amely ezt a feladatot valósítja meg [26, 27, 28]. A szükséges A,B templateket keresési algoritmusokkal sikerült elérni, amelyeket megfelelőségi vizsgálattal ellenőriztem, így ha a szabályokat a bemenetre rakjuk ($y_{i,j}$), és a beérkező pixelt a kezdeti állapotba ($u_{i,j}$), valamint a CNN szélén lévő pixeleket fehérre állítjuk (Boundary=-1), akkor a következő változókkal pont a feladatot valósítja meg:

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -8 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad Z_0 = -2 \quad (1)$$

4.2.1. Nagyobb szabályrendszerek esetén a szabályok diszjunkt területté való leképezése

Az architektúra tervezésekor fontos szempont volt, hogy minden esetet lefedjen a módszer. A klasszifikációs szabályok nagy része átfedő területeket hoz létre a CNN chip felületén. Ennek kiküszöbölésére az átfedő területeket önálló diszjunkt halmazokká bontottuk. A transzformáció után a logikai processzor feladata lesz egyszerű szabálytáblák alapján megtalálni a megfelelő sorszámú szabályt. A CNN futási feltételeként bármely referenciapont 1-re váltását (aktiválását) adjuk meg. A végső döntést végzi el a logikai processzor, aminek a bemenetére kötjük a 6 CNN-t (6. ábra). A logikai processzor mind a 6 CNN-től vár egy eredményt, majd a rendelkezésre álló szabályhalmaz szerint logikai úton eldönti, hogy melyik szabály teljesült.



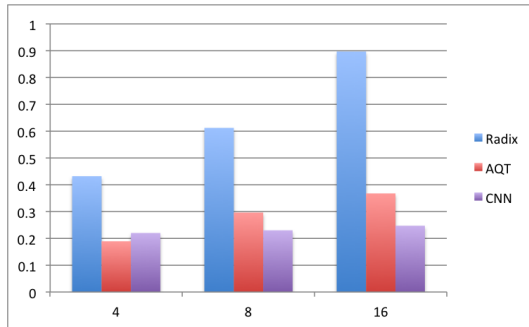
6. ábra. Az architektúra logikai felépítése

4.3. Numerikus eredmények

A numerikus eredmények, az összehasonlíthatóság és a modell vizsgálhatósága kedvéért, szimulációkkal készültek. Ezeket a Martin Haengi, Java nyelven íródott CNN toolkitjéne segítségével sikerült megalkotni. A teljesség kedvéért, valamint a működőképesség igazolása miatt az algoritmust egy valós CNN-en (egy ACE16K rendszeren) is megvalósítottam.

A szimulációhoz a környezetet a következő módon alkottam meg: 80479 csomagot vettem át az ohioi Wilbreforce University Argus nevű szerveréről, amely 2 percnyi kommunikációnak felelt meg. A szerveren 1219 darab szabály volt aktív ebben a pillanatban. Az egyes algoritmusok közül a Radix Trie-t, valamint az AQT-t szimulációs programként megvalósítottam, majd ugyanazon a számítógépen, ugyanazon a hálózati topológián lefuttattam őket.

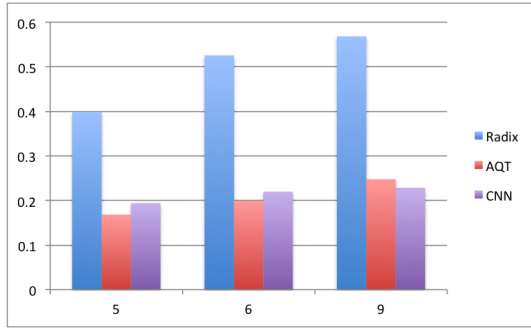
A következő eredmények egyrészt a szimulált környezetben változó paraméterű szimulált csomagfolyamok hatását, másrészt a leírt valós csomagfolyam teljesítőképességét mutatja be.



7. ábra. Különböző szabályszerűségű teljesítőképesség (klasszifikációs idő - ms) azonos csomagforrás-paraméterek mellett. Látható, hogy a CNN-es megoldás, több szabály esetén a kis területek miatt hamarabb tüzel

Látható, hogy a Radix érzékeny a szabályszerűságra. Ugyanakkor elmondható, hogy az AQT abban az esetben, ha DSP-n (Digital Signal Processor) implementáljuk, a gyors szorzásnak és a párhuzamosságának köszönhetően a szabályszerűságra sokkal kevésbé érzékeny, mint a Radix. A pontos hatást az algoritmus esetében egyéb környezeti körülmények befolyásolják.

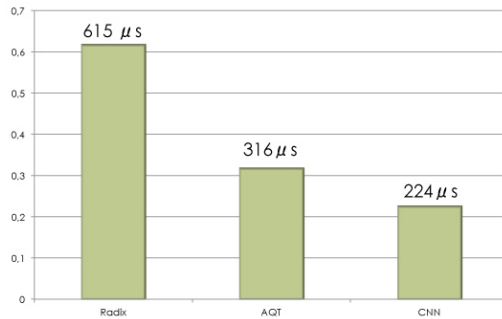
A szimuláció során az architektúrákat megfelelően modellezve építettem be a szimulációs környezetbe. Az AQT esetén egy TTT (Token Triggered Threading)[? ?] alapú multithread DSP implementációt vizsgáltam meg, és ez alapján alakítottam ki a szimulációs környezetet, amelyet megvizsgáltam abból a szempontból, hogy milyen műveleteket végez el párhuzamosan, és ezeket a szimulációban is párhuzamosan végzettként tekintetem. Ez az oka annak, hogy a DSP erős párhuzamossági képessége a szimulációban is megjelenik.



8. ábra. A klasszifikációs idő (ms) a csomagforrás hosszát változtatva (leghosszabb prefix - W) azonos szabálysám és csomagszám mellett

Jól látszik, hogy a hagyományosnak mondható Radix-módszer miért nem alkalmas IPv6-os hálózatok számára. A Radix a csomagforrás-hossz függősége miatt komoly hátrányt jelent a hosszabb fejléc kiválasztásakor. Ebben az esetben láthatjuk, hogy a CNN mutatja a legjobb eredményeket.

Végül a valós csomagfolyamon elért eredmények:



9. ábra. A klasszifikációs idő (ms) az egyes algoritmusokkal az Ohio-i egyetem szerveréről szedett csomagfolyammal

Téziscsoport 2

5. Vezetéknélküli szenzorhálózatok

A vezetéknélküli szenzorhálózatok élettartamának maximalizálása az energiafelhasználás optimalizálásával jelenleg is kutatások tárgya. Ezen munkák [29, 30, 31, 32, 33, 34] közös jellemzője, hogy általában az optimális energiaszint beállítását egy folytonos skálán keresik. Azonban a fizikai megvalósítás szempontjából a szenzorok adóenergiája egy diszkrét halmazból választható. Ennek megfelelően az optimális küldési energia meghatározását pusztán a node által használható diszkrét értékeken kell elvégezni. Ez viszont egy új típusú kombinatorikus optimalizálási feladathoz vezet.

A második téziscsoportban három új, egymáshoz hasonló módszert dolgoztam ki, melyek mindegyike alacsony komplexitású műveletekkel képes olyan új útvonalakat választani, amelyek energiája minimális a node-ok diszkrét energiakészletéből választva. Ezek a megoldások szuboptimálisak ugyan, de a Leach-protokollhoz [10, 35] képest energiatakarékosabb útvonalválasztást tesznek lehetővé. A javulást két módon értelmezve

- egyrészt előírt megbízhatóságot garantálva (a csomag előírt valószínűséggel érkezik meg a bázisállomásra)
- másrészt a megbízhatósági kényszer mellett kisebb energiával juttatjuk a csomagokat a bázisállomásra.

5.1. Modell és probléma felvetés

Célom egy olyan útvonalat találni, amely az energia szempontjából minimális, ugyanakkor a csomag megérkezési valószínűsége a bázisállomásra egy meghatározott érték felett marad. A hálózatot egy kétdimenziós gráf modellezi, melynek éleit v jelöli, ezek az élek a node-ok közötti rádiós csomagátvitelt reprezentálják. Minden élhez tartozik egy valószínűség, amely a küldő és fogadó node közötti sikeres csomagátvitel valószínűsége, amely a két

node közötti távolság (d) és az adóteljesítmény (G) függvénye (Ψ). Ezt Rayleigh-fading modell alapján (2) írja le,

$$P_{i_j, i_{j+1}} = \Psi(d_{i_j, i_{j+1}}, G_{i_j, i_{j+1}}) = e^{-\theta * \sigma^2 * d_{i_j, i_{j+1}}^\alpha / G_{i_j, i_{j+1}} - G_0} \quad (2)$$

ahol

- θ az érzékenységi küszöb,
- σ a zaj energiája,
- P_r annak a valószínűségét definiálja, hogy az adott csomag a csomagtovábbítás során nem veszik el,
- d a távolság,
- G az adóenergia,
- G_0 a készülék belső konstans energiaigénye.

Látható, hogy ez a valószínűség az adó energiafüggvénye, amely egy diszkrét halmazból veszi fel az értékeit. Ha a csomag egy adott útvonalon halad végig, akkor a linkátvitelek függetlenségét feltételezve a sikeres bázisállomásba-érkezés valószínűsége a következő:

Az egyes R útvonalakat úgy írom le, hogy az tartalmazza azt a V vektort, amelynek tartalma az útvonalban szereplő v -k, azt az E vektort, amely az egyes élekhez tartozó energiaértékeket adja, valamint a megfelelő élekhez tartozó távolságot.

$$R(V, E, d_{\in V}) \quad (3)$$

Így tulajdonképpen célom megtalálni az optimális R_{opt} -ot.

Modellemben olyan környezetet választottam, ahol a csomagok megérkezési valószínűsége vezeték nélküli csomagtovábbítás esetében leírható egy olyan függvénnyel (Ψ), amelynek paraméterei a távolság (d) és a küldési energia (g). Későbbiekben lehetőségem lesz bármilyen terjedési modellt alkalmazni a Ψ függvényre, például a legtöbbet

alkalmazott Rayleigh Fading-modellt.

$$P_{u,v} = \Psi(d_{u,v}, g) \quad (4)$$

A probléma alapfeltétele az, hogy az útvonalon végighaladó csomag teljes megérkezési valószínűsége egy általunk, illetve a hálózat tervezője által definiált kritikus érték (ϵ) felett maradjon, így egy produktummal kifejezhetjük ezt a feltételt a következőképpen:

$$\prod_{u,v} \Psi(d_{u,v}, g) \geq 1 - \epsilon \quad (5)$$

Ezt a feltételt átvisszük összeges formára, így a következő logaritmikus skálázást fogjuk kapni:

$$\sum_{(u,v) \in R} -\lg(\Psi(d_{u,v}, g)) \leq -\lg(1 - \epsilon) \quad (6)$$

Tekintettel arra, hogy a jelenlegi alapfeltevésünk szerint az energiaértékek diszkrét értékeket (G_1, G_2, \dots, G_N) vehetnek fel, a probléma célfüggvénye az, hogy kényszeres optimalizálási feladatként minimalizáljuk a küldési energiát, ahol a kényszert a 6 jelenti.

$$\min(g), g \in \{G_1, G_2, \dots, G_N\} \quad (7)$$

Ezzel a célfüggvényt additív optimalizálási feltétellé alakítottam, úgy hogy az u értékhez negatív logaritmikus értékeket rendeltem.

Tézis 2.1 *Megmutattam, hogy minimális energiájú, adott megbízhatóságú algoritmus megoldható Bellmann-Ford-algoritmussal, így a feladat polinomiális időben kivitelezhető.*

A megoldás során kihasználom azt az alapfeltételt, hogy a csomópontok csak diszkrét energiaértékeket vehetnek fel, így a folytonos energiaszint-optimalizálás helyett 8-10 diszkrét energiaszintet kell vizsgálni. Ezzel a feltétellel gyorsan találhatunk az energia szempontjából optimális útvonalat. A feladatot a következő iterációs algoritmus oldja meg:

1. Lépés

Első fázisban az energiát a beállítható minimumértéknek veszem, és így keresek egy optimális útvonalat; a csomagtovábbítás teljes valószínűségére (7. képlet) vonatkozó kritériumot figyelmen kívül hagyom. Így a feladat a következőképpen alakul:

$$g_0 \rightarrow R_0 : \min_{(u,v) \in R} -\lg(\Psi(d_{u,v}, g)) \quad (8)$$

Ez a minimum-keresés egy Dijkstra- vagy egy Bellmann-Ford-algoritmus segítségével polinomiális időn belül véghez vihető.

2. Lépés

Az iterációs lépés második fázisa az, hogy megvizsgálom, hogy a kapott érték vajon teljesíti-e a fent leírt feltételt, és ennek alapján közelítem a megoldást.

$$\sum_{(u,v) \in R} -\lg(\Psi(d_{u,v}, g)) \leq -\lg(1 - \epsilon) \quad (9)$$

Ha teljesíti, akkor

$$g_1 := g_0 \quad (10)$$

Ha nem, akkor

$$g_1 := g_0 + 1 \quad (11)$$

Ez a módszer addig emeli az energiaszinteket, amíg a feltételnek szabott kritériumnak eleget nem tesz. Így tulajdonképpen egy költséghatékony és gyors módszerhez jutok.

Tézis 2.2 *A hálózati topológiát figyelembe véve egy módosított algoritmust adtam a teljesítőképesség további javítására.*

Az előbb bemutatott módszer egy egyenletesen elosztott hálózat esetén nagyon jó eredményt hoz, ugyanakkor nem veszi figyelembe azt, hogy a hálózat csomópontjai nem egyenletesen vannak elszórva a síkon, csomókat alkothatnak.

A másik probléma, hogy azt sem veszi figyelembe, hogy a bázisállomáshoz közeli node-ok szűk keresztmetszetté, azaz olyan csomóponttá alakulnak, amelyen minden forgalom

keresztülhalad, így gyorsan elfogy az energiája. Ezért módosítjuk a módszert úgy, hogy figyelembe vesszük az energiaszinteket is. Mindig a legjobban töltött csomópont energiaszintjét növeljük, egészen addig, amíg vagy a feltételt nem teljesíti, vagy az energiaszintje nem különbözik túlságosan az alapszinttől.

Ezért definiálunk egy n egész számot, amely azt határozza meg, hogy mennyire növelhetjük az egyes node-ok energiaszintjét. Az alapenergiaszintet akkor emelem, ha már nem tudok emelni a node-ok energiáin. Figyeljük meg, hogy $n = 0$ esetében ez az algoritmus azonos az előzővel.

Az algoritmus a következő:

Algorithm 1 G - iterációs algoritmus

$GlobalG := 0$

loop

$G_{1..N} := GlobalG$

repeat

if Required Probability satisfied **then**

return G

end if

$G_x := G_x + 1$ where $G_x < GlobalG$

until exists $G_{1..N} < GlobalG$

$GlobalG := GlobalG + 1$

end loop

A módszert a szimuláció során G néven említtem.

Tézis 2.3 *Az algoritmust úgy fejlesztettem tovább, hogy a kevés energiájú csomópontokat kizártam, így performancia-javulást tudtam elérni.*

A második módosítás, egy másik megközelítése ugyanennek a problémának, hogy az energiaszinteket is figyelembe vesszük. A módosítás lényege, hogy kizárjuk a csomagtovábbítás folyamatából azokat a csomópontokat, amelyek átlag alatti energiaszinttel

rendelkeznek. Ezek a csomópontok csak akkor küldenek csomagot, ha azok rajtuk generálódtak.

Algorithm 2 MinG - Kevés energiájú csomópontok kizárása

loop

$T :=$ average value of $A_{1..N}$

$I = \emptyset$

for all Nodes **do**

if $A(Node) > T$ **then**

$I+ = Node$

end if

end for

RUN $G(I)$

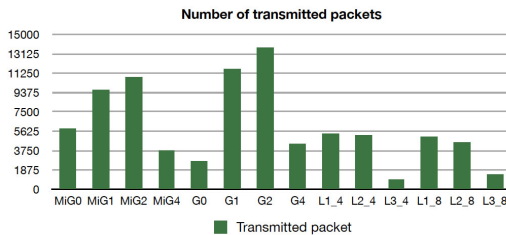
end loop

Ezzel a módszerrel nem terheljük az erőseket, hanem védjük a gyengéket.

A módszert a szimuláció során MinG néven említtem.

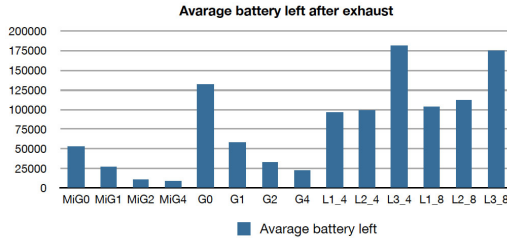
5.2. Teljesítőképesség-analízis

A következő ábrán láthatjuk azt, hogy hány ezer csomagot tudunk továbbítani a hálózatokban, míg azok elérték a lemerülési kritériumot.



10. ábra. A továbbított csomagok száma (x1000) a hálózat lemerüléséig a különböző algoritmusok használatával

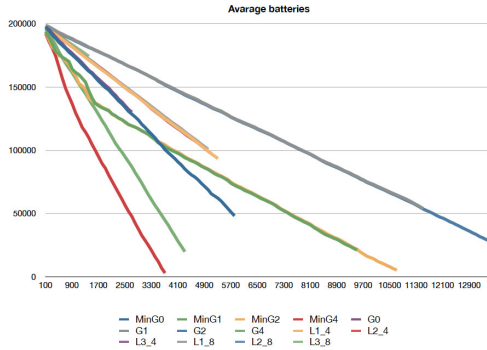
Jól látható, hogy a fix-G algoritmusaink többszörös eredményt mutatnak. Az eredmények azonban azt is megmutatják, hogy ha túl nagyra választjuk a fix-G algoritmusunkhoz az n paramétert, akkor a performancia visszaesik.



11. ábra. A lemerülést követően a hálózatban maradt átlagenergia

Ezen az ábrán az látható, hogy a MinG algoritmusok valóban arra törekednek, hogy teljesen egyenletesen osszák el az energia-terhelést, hiszen ezzel az algoritmussal maradt a legkevesebb felesleges energia a hálózatban. A 10. ábrával összevetve azonban látszik, hogy a fő célt – a csomagok továbbítását – mégsem ez a módszer érte el a legjobban.

Ha megfigyeljük, látható, hogy nem feltétlenül az a hálózat teljesít jobban, amelyekben kevesebb energia marad a végére, hiszen a G2-algoritmussal használt hálózatban majdnem háromszor annyi energia maradt, mint a MinG4-ben, amely viszont már alig hárommillió csomag elküldése után lemerült, ellentétben a másikkal, amelyek több mint 13 millió csomagot tudott továbbítani.



12. ábra. Az átlag energiaértékek hálózatonként

A 12. ábrán látható, hogy hogyan alakul a legtöbb energiával, illetve a legkevesebbel bíró node sorsa. Látszik, hogy a MinG algoritmusok, miután észrevették, hogy vészesen fogy az egyik node energiája, nem hagyták, hogy azok a csomópontok részt vegyenek a kommunikációban, és ezért a minimum energia kisebb szögben csökkent egy idő után. Ezen kívül az is megállapítható, hogy azon hálózatok, amelyekben a max node energia elég magas, még rendelkeznek tartalékokkal.

5.3. Az elért eredmények jelentősége

Összességében elmondható, hogy az algoritmus segítségével elérhető, hogy

- a hálózat csomópontjai később merüljenek le
- a hálózat csomópontjai nagyjából egyszerre merüljenek le
- a hálózat kommunikációja egészen a lemerülésig az elvárt minőséget éri el

Az egyszerű, sima G verzió a gyors konvergencia érdekében egyszerre emeli a linkek energiaértékeit, így az optimálistól távoli kezdőállapotból is gyorsan halad az optimum felé,

de az elért megoldás minősége csak közelíti a globális optimumot. A második algoritmus nem egyszerre emeli a linken energiaértékeit, így lassabb a konvergenciája, de jobb minőségű eredményt kapunk.

Így az elért eredmények például egy épületmonitorozási feladat esetén nagyban csökkenti a szenzorhálózat fenntartási költségeit, de például egészségügyi monitorozás esetén (idősek otthona, kórház) a kritikusnak számító adattovábbítási minőséget is garantálja.

Téziscsoport 3

6. Hálózattdimenzionálás csomagkapcsolt hálózatokban

Ebben a téziscsoportban az Internet-eléréshez szükséges különböző felhasználói szinteket biztosító Hierarchical Admission Module (HAM) méretezéséhez mutatok be új algoritmusokat.

Az új algoritmusok biztosítják, hogy a felhasználók különböző szintű Internet-hozzáférése adott minőségben, de minimális hardver komplexitással történjen. Ezért célom a HAM optimális topológiájának kialakítása, ami előírt QoS biztosítása mellett minimális hardverkomplexitást eredményez. Az optimális architektúra megtalálásához a lehetséges topológiák és linkkapacitások teljes állapotterét kell végignézzük, hogy megtaláljuk az optimális architektúrát. Ez a méretezést egy optimalizációs problémává alakítja, amelyre mostantól NCAP-ként (Node and Capacity Arrangement Problem) hivatkozunk. A tervezést két eszköz segítségével végezhetjük el: a Nagy Eltérések Elméletének alkalmazásával megbecsüljük a forgalom farokeloszlását, valamint kombinatorikus optimalizációs algoritmusokat használunk.

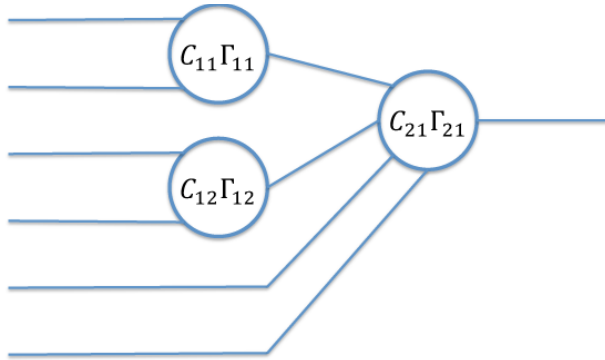
6.1. A modell

Tipikus Internet-eléresi esetben a szolgáltató a felhasználókat a következő 3 forgalomosztályba sorolja:

- Internet Access1;
- Internet Access2;
- Voice over ADSL.

A forgalmi osztályokhoz a felhasználók tekinthetők ON/OFF forrásnak, rendre m_1, m_2, m_3 átlaggal és h_1, h_2, h_3 csúcsértékekkel. A $\gamma_1, \gamma_2, \gamma_3$ QoS-paraméter azt határozza meg, hogy mekkora cellavesztést (Cell Loss Probability - CLP) engedünk meg.

A fentieknek megfelelően a méretezés várt kimenetelét az alábbi ábra mutatja:



13. ábra. A méretező-algoritmus kimeneteként kapott struktúra

Az ábra alapján a HAM node-ok egy halmazának tekinthető, amelyek fa-topológiában vannak elhelyezve. A méretező-algoritmus meghatározza a topológiát, valamint minden node-hoz linkkapacitásokat és QoS-paramétereket rendel. Ez alapján a HAM formálisan a következő módon írható le:

$$HAM = \{V, E, C, \Gamma\}, \quad (12)$$

ahol a V a csúcsokat, E az éleket jelenti, amíg C és Γ a kapacitások és QoS-paraméterek mátrixait jelöli a következő módon:

$$C_{kj} = C_j(k) \quad (13)$$

a kapacitása a j . node-nak a k -ik rétegben.

$$\Gamma_{kj} = \gamma_j(k) \quad (14)$$

az elvárt QoS-érték a j . node-nak a k -ik rétegben.

Megjegyzendő, hogy amikor a HAM sok node-ból áll össze, a cellák bármelyik eszközön elveszhetnek, ezért ilyenkor szigorúbb CLP-követelményt kell előírni az egyes

6. HÁLÓZATDIMENZIONÁLÁS CSOMAGKAPCSOLT HÁLÓZATOKBAN

node-okon. Ennek eredményeként valóban kihívás az aggregált CLP-érték node-okra vonatkozó dekompozíciója.

A következő jelöléseket fogjuk használni a HAM leírásához:

- forgalomsztályok: $i = 1, \dots, M$;
- rétegek a fatopológiában: $k = 1, \dots, K$;
- node-ok a k -ik rétegben: $l = 1, \dots, L_k$;
- engedélyezési vektor a j -ik node-hoz a k -ik rétegben: $\mathbf{n}^j(k)$, ahol az $n_i^j(k)$ komponens az i osztálybeli források számát jelöli;
- az engedélyezhető vektorok egy halmazát Admission Set (AS)-nek nevezzük, amely a forgalomvektorokat tartalmazza az adott node-hoz a fatopológiában, a következők szerint:

$$\text{AS} = \{ \mathbf{n}^l(k) \forall l = 1, \dots, L_k \forall k = 1, \dots, K \}; \quad (15)$$

- a bemeneti forgalom-állapotvektort a következőképpen írjuk le:

$$\mathbf{v}(1) = (\mathbf{n}^1(1), \mathbf{n}^2(1), \dots, \mathbf{n}^{L_1}(1)) \quad (16)$$

Megjegyzendő, hogy az AS a bemeneti állapotvektorral összefüggésben áll, mégpedig olyanformán, hogy mindegyik bemeneti állapotvektor felbontható egy AS-re, méghozzá a következő formában:

A bemeneti állapotvektor dekompozíciója:

A bemeneti állapotvektor $\mathbf{v}(1) = (\mathbf{n}^1(1), \mathbf{n}^2(1), \dots, \mathbf{n}^{L_1}(1))$ dekompozícióját egy AS-ra, a folyamatábra alapján, a következőképpen definiáljuk:

$$n_i^l(k) = \sum_{j \in A_l} n_i^j(k-1), \quad (17)$$

6. HÁLÓZATDIMENZIONÁLÁS CSOMAGKAPCSOLT HÁLÓZATOKBAN

ahol A_l azon *node*-ok halmaza $k - 1$. rétegben, amelyek kapcsolatban állnak l -ik *node*-dal a k . rétegben.

Egyértelmű, hogy a fenti definíció tekinthető egy $V \rightarrow AS$ -leképezésnek, ahol a bemenet a bemeneti állapotvektor $\mathbf{v}(1)$, és a kimenetet nevezzük $AS(\mathbf{v}(1))$ -nek.

A HAM-t reprezentáló adatstruktúra a következőképpen néz ki: A HAM mint egy gráf $G\{V, E, \mathbf{C}, \Gamma\}$ teljes mértékben leírható a \mathbf{C} és a Γ mátrixokkal. Ezen mátrixok segítségével mind a topológia és hozzákapcsolódó kapacitási beállítások $\{C_l(k), l = 1, \dots, L_k, k = 1, \dots, K\}$, mind a QoS-paraméterek $\{\gamma_l(k), l = 1, \dots, L_k, k = 1, \dots, K\}$ visszaállíthatóak.

A topológiát egy \mathbf{G} mátrix írja le:

$$G_{kl} = \begin{cases} 1 & \text{ha van node a } k\text{-ik réteg } l\text{-ik pozíciójában} \\ 0 & \text{egyébként} \end{cases} \quad (18)$$

A HAM QoS-beállításait egy adott \mathbf{G} topológia esetében Γ^G mátrixszal jelöljük, ahol a kl elem a k -ik rétegben található l -ik *node* QoS-paraméterét jelenti. Ha a k -ik réteg l -ik pozíciójában nincsen *node*, akkor a $\Gamma_{kl}^G = 0$, ami azt jelenti, hogy ha $G_{kl} = 0$, akkor $\Gamma_{kl}^G = 0$. Továbbá feltételezzük, hogy a QoS lehetséges értékei egy diszkrét $\gamma_1, \dots, \gamma_V$ halmazt alkotnak. Ezért a $— = \{\Gamma_{min}, \dots, \Gamma_{max}\}$ halmaz tartalmazza a lehetséges QoS-mátrixokat. A Γ_{min}^G mátrixok $\Gamma_{kl} = \text{Min}$ -nek, míg a Γ_{max}^G mátrixot $\Gamma_{kl} = \text{Max}$ -nak definiáljuk, ahol *Min* és *Max* korábban meghatározott értékek. Ebben a formában a megfelelő QoS-séma kiválasztásához a méretező algoritmusnak minden egyes *node*-hoz ($l = 1, \dots, L_k$ and $k = 1, \dots, K$) végig kell mennie a $G_{kl} \in (\text{Min}, \text{Max})$ intervallumon.

A HAM kapacitás-hozzárendelését a \mathbf{C} mátrixszal fejezzük ki. Megjegyzendő, hogy ha $G_{kl} = 0$, akkor $C_{kl} = 0$, ami azt jelenti, hogy a topológiában kapacitás csak létező *node*-hoz rendelhető. A \mathbf{G} topológiához tartozó lehetséges kapacitás-mátrixot \mathbf{C}^G -vel jelöljük (ahol $C_{ij}^G \in \{C_1, \dots, C_R\}$). Ezek a mátrixok $\mathcal{C}^G = \{\mathbf{C}_{min}^G, \mathbf{C}_2^G, \dots, \mathbf{C}_{max}^G\}$ diszkrét teret

6. HÁLÓZATDIMENZIONÁLÁS CSOMAGKAPCSOLT HÁLÓZATOKBAN

alkotnak, ahol $C_{min}^G : C_{ij} = C_1 G_{ij} \forall i, j$ a minimális kapacitású hálózati topológiát jelenti, míg $C_{max}^G : C_{ij} = C_R G_{ij} \forall i, j$ ugyanezen topológia maximális kapacitású node-jait. Tekintettel arra, hogy a lehetséges kapacitások száma véges, a programozó a \mathcal{C}^G halmazt bármilyen szabály alapján rendezni tudja. (Jelen adaptációban a rendezési séma az elemek összegétől és a megfelelő mátrixon belüli indexek rangjától függ.)

Ezen adatstruktúrák segítségével a méretező-algoritmus teljes egészében leírható.

Tézis 3.1 *Az optimális HAM eléréséhez kifejlesztettem egy egynode-os optimalizációs algoritmust.*

Egynode-os méretező algoritmus

Rendelkezésre áll egy $\mathcal{C} = \{C_1, \dots, C_R\}$ $C_1 < C_2 < \dots < C_R$ diszkrét kapacitáshalmaz, egy bemeneti forgalom, amit $\mathbf{n} = (n_1, \dots, n_M)$ forgalmi konfigurációs vektor fejez ki, valamint egy γ CLP-szint, mint QoS-paraméter.

Legyen $C := C_1$ és $r := 1$.

Számoljuk ki a logaritmikus momentumot generáló $\mu_i(s)$ $i = 1, \dots, M$ függvényeket.

- 1. Határozzuk meg $s_{opt} : \inf_s \sum_{i=1}^M n_i \mu_i(s) - sC$*
- 2. Nézzük meg, hogy $\sum_{i=1}^M n_i \mu_i(s_{opt}) < s_{opt}C - \gamma$ teljesül-e.*
- 3. Ha IGEN, térjünk vissza C -vel, ha NEM, legyen $r := r + 1$ és térjünk vissza az 1. lépéshez.*

Ezzel az algoritmussal megtalálhatjuk a minimális kapacitású C_{min} -t, ami elég hatékony ahhoz, hogy biztosítson az \mathbf{n} terhelési vektorhoz megfelelő kapacitást γ QoS-szinthez.

6.2. Új többnode-os dimenzionálási algoritmus

A dimenzionálási feladat célja, hogy találjunk egy olyan topológiát, megfelelő kapacitásokkal, amely egy az egész rendszerre vonatkozó QoS feltételt teljesít a megfelelő terhelés-vektorra. Ezért megadunk egy leképezést (Ψ) bemeneti terhelés-vektor $v(1)$, a QoS γ paraméter és a $G_{opt}(V, E, C, \Gamma)$ között. Így az optimalizálási probléma formálisan a következőképpen írható le:

$$G_{opt} \{V, E, C, \Gamma\} = \Psi(\mathbf{v}(1), \gamma); \quad (19)$$

ahol

$$G_{opt}(V, E, C, \Gamma) : \min_{G(V, E, C, \Gamma)} \sum_{k=1}^K L_k. \quad (20)$$

Észrevehető, hogy számos Γ mátrix kielégíti a QoS feltételeket, így nem csak a kapacitásokat, hanem a QoS paramétereket is érdemes jól beállítani, hogy a node-ok között legyen osztva. Ez a gondolat adja a rekurziós algoritmus alapötletét: Kezdjünk egy minimális konfigurációval, $G\{V, E, C, \Gamma\}$, és nézzük meg, hogy teljesíti-e a QoS kritériumokat. Ha nem, akkor bővítjük a konfigurációt egészen addig, amíg a $v(1)$ bemeneti vektorra a QoS ill. a CLP követelmények teljesülnek. Tekintettel arra, hogy a minimális konfigurációval indítjuk az iterációt, az optimális megoldást fogjuk megkapni.

Tézis 3.2 *Az optimális HAM eléréséhez kifejlesztettem egy többnode-os optimalizációs algoritmust.*

Többnode-os méretező-algoritmus kiegyensúlyozott terheléshez

Az első réteg minden node-jának bemeneténél rendelkezésre áll egy

$$\mathbf{v}(1) = (\mathbf{n}^1(1), \mathbf{n}^2(1), \dots, \mathbf{n}^{L^1}(1)) \quad (21)$$

forgalomkonfiguráció, és egy \mathbf{T} logikai változókat tartalmazó mátrix, melynek minden T_{kl}

eleme azt jelzi, hogy a k -ik réteg l -ik node-ján levő helyi QoS-kritériumnak az adott node megfelel-e.

Az U változó azt jelöli, hogy az egészében vett QoS-kritériumnak megfelel-e.

1. Legyen $\mathbf{G} = \mathbf{G}_{min}$;
2. Dekomponáljuk a $\mathbf{v}(1)$ -et $AS(\mathbf{v}(1)) = \{\mathbf{n}^l(k), l = 1, \dots, L_k, k = 1, \dots, K\}$ AS-sé;
3. Legyen $\mathbf{C}^G := \mathbf{C}_{max}^G$;
4. Legyen $\mathbf{\Gamma}^G := \mathbf{\Gamma}_{max}^G$;
5. Számoljuk ki a $s_{l\ opt}(k)$ -t megoldva a

$$s_{l\ opt}(k) : \sum_{i=1}^M n_i^l(k) \frac{d\mu_i(s)}{ds} = C_l(k) \quad \forall l = 1, \dots, L_k \quad k = 1, \dots, K$$

egyenletet.

6. Ha

$$T_{lk} = \begin{cases} \text{IGAZ} & \text{if } \sum_{i=1}^M n_i^l(k) \mu_i(s_{l\ opt}(k)) < s_{l\ opt}(k) C_l(k) - \gamma_l(k) \\ \text{HAMIS} & \text{if } \sum_{i=1}^M n_i^l(k) \mu_i(s_{l\ opt}(k)) > s_{l\ opt}(k) C_l(k) - \gamma_l(k) \end{cases} \quad (22)$$

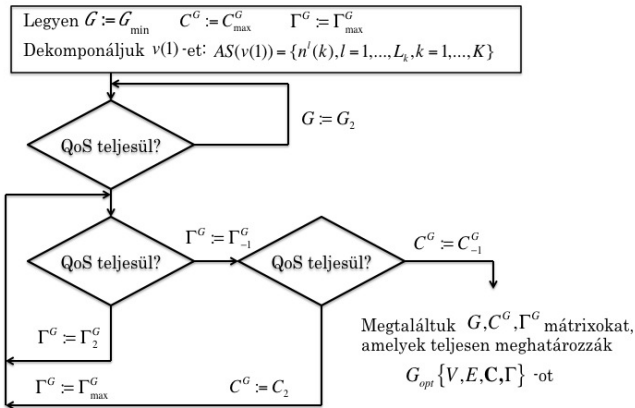
7. Számoljuk ki az $U := \bigcap_{k=1}^K \bigcap_{l=1}^{L_l} T_{kl}$ értéket.
8. Ha $U = \text{HAMIS}$, növeljük a topológiát azáltal, hogy beállítjuk $\mathbf{G} := \mathbf{G}_2$ -t, visszatérünk a 3. lépéshez és addig ismételjük ezt, amíg $U = \text{IGAZ}$
9. Ha $U = \text{IGAZ}$, akkor találtunk topológiát, de azzal lazítottunk a QoS-követelményeken, hogy másik $\mathbf{\Gamma}^G \in -G$ -t választunk a $\mathbf{\Gamma}^G := \mathbf{\Gamma}_2$ beállításával, visszatérünk az 5. lépéshez és addig ismételjük ezt, amíg $U = \text{HAMIS}$ vagy $\sum_{k=1}^K \sum_{l=1}^{L_k} e^{-\gamma_l(k)} > e^{-\gamma}$ majd visszatérünk $\mathbf{\Gamma}^G$ előző értékéhez.

10. Ha $U = \text{IGAZ}$, akkor találtunk topológiát, de azzal csökkentjük a kapacitást, hogy másik $C^G \in \mathcal{C}^G$ -t választunk a $C^G := C_2$ beállításával, visszatérünk a 4. lépéshez és addig ismételjük ezt, amíg $U = \text{HAMIS}$, majd visszatérünk C^G előző értékéhez.

11. Visszatérünk a G, C^G, Γ^G mátrixokkal, amelyek teljesen meghatározzák $G_{\text{opt}} \{V, E, C, \Gamma\}$ -ot.

Látható, hogy az algoritmus az optimális HAM-t úgy adja vissza, hogy nem csak a node-ok száma minimális (megtalálva a legkisebb topológiát), hanem ezzel együtt a hozzá tartozó kapacitás és QoS-beállítások is. Így megtalálhatjuk a legkevésbé szigorú feltételeket, melyekkel egy minimális topológiájú hálózat kiegyensúlyozott hálózati forgalmat tud kezelni egy meghatározott teljes QoS mellett.

A leírt algoritmus folyamatábrája a következő ábrán látható:



14. ábra. A többnode-os optimaizációs algoritmus folyamatábrája

6.3. Numerikus eredmények

A következő táblázatok egy adott tipikus forgalmi-terhelési konfiguráció (a forgalmi-terhelési konfiguráció az adott forgalmi osztályban lévő felhasználók számával fejezhető ki) mellett megtalált optimális topológiákat mutatja be:

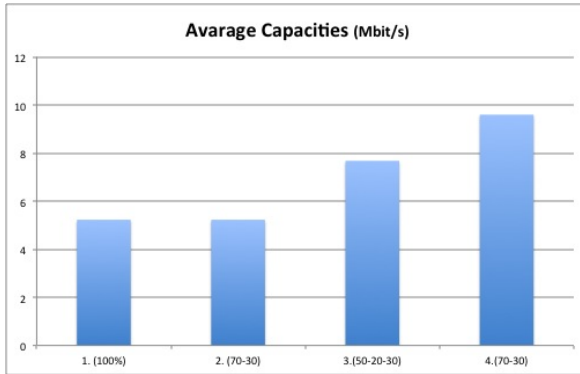
1. 100% Internet Access1 users: $\mathbf{n} = (3000, 3000, 0, 0, 0, 0)$;
2. 70% Internet Access1 users + 30% Voice over DSL users :
 $\mathbf{n} = (2100, 2100, 0, 0, 900, 900)$;
3. 50% Internet Access1 users + 20% Internet Access2 users + 30% Voice over DSL users: $\mathbf{n} = (1500, 1500, 600, 600, 900, 900)$
4. 70% Internet Access2 users + 30% Voice over DSL users :
 $\mathbf{n} = (0, 0, 2100, 2100, 900, 900)$

A 2. táblázat a többnode-os kiegyensúlyozott terheléses optimalizáló algoritmus segítségével elkészített eredményeket mutatja be.

	eszközsám	QoS-paraméter $((\gamma_1, \gamma_2))$	kapacitás (C_1, C_2)
1.(100%)	1. réteg 12 2. réteg 1	1. réteg 7.895 2. réteg 9.215	5.23 Mbit/s 12.21 Mbit/s
2.(70-30)	1. réteg 12 2. réteg 1	2. réteg 7.895 2. réteg 9.215	5.23 Mbit/s 12.21 Mbit/s
3.(50-20-30)	1. réteg 12 2. réteg 1	1. réteg 7.895 2. réteg 9.215	6.98 Mbits/s 19.2 Mbit/s
4.(70-30)	1. réteg 12 2. réteg 1	1. réteg 7.895 2. réteg 9.215	8.72 Mbit/s 22.69 Mbit/s

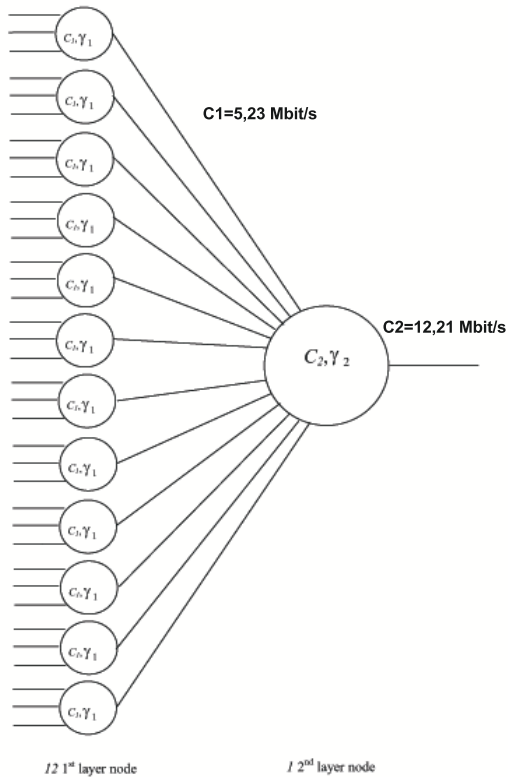
2. táblázat. Forgalom-terhelési konfiguráció mellett megtalált optimális topológiák kiegyensúlyozott terheléses algoritmussal (1-es módszer)

Látható, hogy a különböző kimeneti vektorok esetében csak a kapacitás és a QoS beállítások különböznek.



15. ábra. Átlagkapacitások a különböző forgalomkonfigurációkhoz

A 15. ábrán látható az átlagos kapacitás igény $(\frac{1}{\sum_{k=1}^K L_k} \sum_{k=1}^K \sum_{l=1}^{L_k} C_l(k))$ a négy különböző esetben. Észrevehető, hogy a negyedik esetben a legnagyobb az átlagos kapacitás. Az oszlopdiagramból egyértelműen leolvasható, hogy a 4. konfiguráció jelenti a legszigorúbb kapacitás követelményeket.



16. ábra. Kiegyensúlyozott terheléses algoritmus segítségével készített optimális hálózati topológia

A fenti eredmények megadják az adott forgalmi konstellációhoz tartozó minimális HAM architektúrát (topológia, kapacitás és belső QoS értékek). Az új algoritmus minden más forgalmi konfigurációra is képes elvégezni a dimenzionálást.

7. Konklúziók

A következőkben az elért eredményeket tárgyalom az említett témák és a jelenlegi technológiák szűk keresztmetszetei alapján. A 3. táblázat ezeket mutatja be, hozzárendelve a tárgyalt megoldásokat.

	Csomagklasszifikáció	Vezetéknélküli csomagtovábbítás	Hálózatdimenzionálás
Csomagtovábbítás-minőségi feltételek biztosítása	Tézis 1.1	Tézis 2.1	Tézis 3.1 Tézis 3.2
Energiafelhasználás		Tézis 2.2 Tézis 2.3	
Forgalom-optimalizálás	Tézis 1.1		Tézis 3.2

3. táblázat. Tárgyalt témák, illetve a hozzájuk kapcsolódó szűk keresztmetszetek az elért eredmények Téziseinek jelzésével

Összességében elmondható, hogy a csomagkapcsolt hálózatok szűk keresztmetszeteire vonatkozóan a következő, az alábbi alfejezetekben részletezett, akadályokat sikerült áttörni.

7.1. Csomagklasszifikáció CNN-alapú technológiával

A jelenleg használatos IPv6-os routing-technológiát sikerült az általam készült módszerrel javítani, ennek segítségével az Interneten nyújtott - főleg multimédiás - szolgáltatások minősége javítható.

A dolgozat alapján az új algoritmus működőképes, elvégzi a csomagklasszifikáció feladatát. Az algoritmust szimulációval teszteltem, és vizsgáltam a teljesítőképességét.

A CNN-nel történő klasszifikáció technológiai nyitás egy teljesen új rendszer felé. A tesztelés és az algoritmus megírása bebizonyította, hogy egy működőképes és grafikai és

párhuzamossági tulajdonságának köszönhetően egy gyors módszert jelent a CNN-alapú megoldás .

A teljesítképeség-analízis során rangsort állítottam fel az algoritmusok között. A neurális és statisztikai alapú algoritmusok jól teljesítenek, és megfelelő hardveres támogatás mellett mindegyik egy használható módszer. A konstrukció egy mért routing táblában valós adatokkal 0,224 ms alatt végezte el a klasszifikációt, míg az AQT 0,316 ms és Radix 0,615 ms alatt teljesítette ezt a feladatot. A CNN-es megoldás így majdnem 3x-os sebességnövekedést tud elérni.

Az eredmények segítségével új felépítésű csomagtovábbítási eszközök kialakítása valósulhat meg, amelyek megfelelő csomagtovábbítási paramétereket tudnak teljesíteni.

7.2. Vezetéknélküli csomagtovábbítás útvonalválasztása

Új algoritmust hoztam létre, amely az útvonalkeresést egy energiaoptimalizálással ötvözi, úgy, hogy figyelembe veszi a valóságos csomópontok diszkrét energiakioldási lehetőségeit is. A szimuláció során belátható, hogy az algoritmus alkalmas arra, hogy csomagokat továbbítsunk, valamint képes arra, hogy a csomópontokra a küldési információt továbbítsa. A szimulációs eredményeket tekintve azt láttuk, hogy az algoritmus a hagyományos Leach-protokollal szemben átlagosan 1.8x olyan jól teljesít. Azt is megfigyelhetjük, hogy az esetek 90%-ban teljesít jobban, és azon esetekben, amikor a csomópontok "cluster"-esednek, tud a Leach előnyt szerezni.

Az eredmények segítségével biológiai területen életminőség-javítás érhető el, ipari folyamatok monitorozásában költséghatékonyság és jobb szervezethez érhető el, javíthatja a minőséget; lakás vagy irodaház monitorozása esetében pedig energiahatékonyságot, jobb környezetet tudunk kialakítani.

7.3. Hálózatdimenzionálás csomagkapcsolt hálózatokban

Csomagkapcsolt hálózatok tervezése esetén kiemelkedően fontos, hogy adott szolgáltatásminőséget minimális hardvereszközök segítségével valósítsunk meg. Ezt a hálózattervezési

feladatot egy kényszeres optimalizálási feladatként lehet értelmezni, ahol a cellavesztési valószínűség meghatározott.

A téziscsoportban olyan iterációs algoritmusokat mutattam be, amelyek az egyes felhasználói forgalom-terhelési konfigurációs kezdeti feltételek mellett megoldják a feladatot úgy, hogy a cellavesztési valószínűségi kritériumot a teljesítik.

Sikerült az adott felhasználói forgalmat minimális kapacitású linkekkel megfelelő minőségben kiszolgálni, ahol a link kapacitás 12.21 MBit/sec.

7.4. Összefoglalás

A fenti eredmények átfogják a csomagkapcsolt hálózati kommunikáció legfontosabb korlátait, és ezen korlátok feloldására szolgáló algoritmusokat mutatnak be:

- a router-technológiában fontos real-time csomagklasszifikációs feladatok CNN-alapú gyors megoldása;
- energiahatékony útvonalválasztási protokollok kidolgozása energiában limitált vezeték nélküli szenzoriális hálózatok számára;
- minimális komplexitású hardver tervezése adott forgalom kiszolgálására internet-hozzáférési modulokban.

A felsorolás alapján a disszertáció új eredményekkel járult hozzá a csomagkapcsolt hálózatok teljesítőképességének növeléséhez (ezeknek számszerű értékei a 7.1, 7.2 és a 7.3 fejezetben találhatóak).

A szerző publikációi

- [1] J. Levendovszky and B. Karlócai, „Dimensioning hierarchical admission architect,” *Periodica Polytechnica*, 2012 - in progress.
- [2] B. Karlócai, A. Bojársky, and J. Levendovszky, „Energy aware routing protocols for wireless sensor networks using discrete transmission energies,” in *2011 International Joint Conference of IEEE TrustCom-11/IEEE ICSS-11/FCST-11*. IEEE, 2011, pp. 1704–1707.
- [3] J. Levendovszky, B. Karlócai, and A. Bojársky, „Novel cnn based packet classification algorithms for networking,” *Periodica Polytechnica*, 2011 - in progress.
- [4] J. Levendovszky, A. Bojársky, B. Karlócai, and A. Oláh, „Energy balancing by combinatorial optimization for wireless sensor networks,” *WSEAS Transactions on Communications*, vol. 7, no. 2, pp. 27–32, 2008.
- [5] A. Bojársky and B. Karlocai, „Új neurális alapú csomagklasszifikációs algoritmusok fejlesztése és tesztelése ipv6 hálózatok számára,” in *Országos Tudományos Diákköri Konferencia, Informatika szekció, Konferenciakiadvány*, vol. 1, 2005, p. 76.

Hivatkozások

- [1] J. Yick, B. Mukherjee, and D. Ghosal, „Wireless sensor network survey,” *Computer networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [2] E. Jovanov, A. Milenkovic, C. Otto, P. De Groen, B. Johnson, S. Warren, and G. Taibi, „A wban system for ambulatory monitoring of physical activity and health status: applications and challenges,” in *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*. IEEE, 2006, pp. 3810–3813.
- [3] B. Latre, B. Braem, I. Moerman, C. Blondia, E. Reusens, W. Joseph, and P. Demeester, „A low-delay protocol for multihop wireless body area networks,” in *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*. IEEE, 2007, pp. 1–8.
- [4] D. Taylor, „Survey and taxonomy of packet classification techniques,” *ACM Computing Surveys (CSUR)*, vol. 37, no. 3, pp. 238–275, 2005.
- [5] J. y. N. Hui, *Switching and traffic theory for integrated broadband networks*. Boston: Kluwer Academic Publishers, 1990.
- [6] M. De Berg, O. Cheong, and M. Van Kreveld, *Computational geometry: algorithms and applications*. Springer-Verlag New York Inc, 2008.
- [7] M. Hasna and M. Alouini, „End-to-end performance of transmission systems with relays over rayleigh-fading channels,” *Wireless Communications, IEEE Transactions on*, vol. 2, no. 6, pp. 1126–1131, 2003.
- [8] S. Kumar, „Sensor networks: evolution, opportunities, and challenges,” *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, 2003.
- [9] J. Levendovszky and B. Hegyi, „Optimal statistical energy balancing protocols for wireless sensor networks,” *WSEAS Transactions on Communications*, pp. 689–694, 2007.

- [10] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, „Energy-efficient communication protocol for wireless microsensor networks,” in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*. IEEE, 2002, pp. 10–pp.
- [11] D. Puccinelli and M. Haenggi, „Wireless sensor networks: applications and challenges of ubiquitous sensing,” *Circuits and Systems Magazine, IEEE*, vol. 5, no. 3, pp. 19–31, 2005.
- [12] A. Bache, L. BUILLOU, H. Layec, B. LORIG, and Y. Matras, „Rcp, the experimental packet-switched data transmission service of the french ptt: History, connections, control,” in *Proceedings of the Third International Conference on Computer Communication (ICCC)*, 1976, pp. 3–6.
- [13] R. Tucker, J. Baliga, R. Ayre, K. Hinton, and W. Sorin, „Energy consumption in ip networks,” in *Optical Communication, 2008. ECOC 2008. 34th European Conference on*. Ieee, 2008, pp. 1–1.
- [14] A. Pras, L. Nieuwenhuis, R. van de Meent, and M. Mandjes, „Dimensioning network links: a new look at equivalent bandwidth,” *Network, IEEE*, vol. 23, no. 2, pp. 5–10, 2009.
- [15] A. Lööf and H. Seybert, „Internet usage in 2009-households and individuals,” *Eurostat Data in focus*, vol. 46, p. 2009, 2009.
- [16] R. Sinha, C. Papadopoulos, and J. Heidemann, „Internet packet size distributions: Some observations,” *USC/Information Sciences Inst.[Online]*. Available: <http://netweb.usc.edu/~rsinha/pkt-sizes>, 2007.
- [17] D. Gough, L. Kumosa, T. Routh, J. Lin, and J. Lucisano, „Function of an implanted tissue glucose sensor for more than 1 year in animals,” *Science Translational Medicine*, vol. 2, no. 42, pp. 42ra53–42ra53, 2010.
- [18] A. Goldsmith and S. Wicker, „Design challenges for energy-constrained ad hoc wireless networks,” *Wireless Communications, IEEE*, vol. 9, no. 4, pp. 8–27, 2002.

-
- [19] J. Chang and L. Tassiulas, „Energy conserving routing in wireless ad-hoc networks,” in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1. IEEE, 2000, pp. 22–31.
- [20] S. Singhand and C. Raghavendra, „Pamas: Power aware multi-access protocol with signalling for ad hoc networks,” *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 3, 1998.
- [21] L. Chua and L. Yang, „Cellular neural networks: Theory,” *Circuits and Systems, IEEE Transactions on*, vol. 35, no. 10, pp. 1257–1272, 1988.
- [22] —, „Cellular neural networks: Applications,” *Circuits and Systems, IEEE Transactions on*, vol. 35, no. 10, pp. 1273–1290, 1988.
- [23] T. Roska, G. Paziienza, and C. Wu, „Cellular wave computing via nanoscale chip architectures,” *International Journal of Circuit Theory and Applications*, 2010.
- [24] T. Roska, L. Belady, and M. Ercsey-Ravasz, „Cellular wave computing in nanoscale via million processor chips,” *Cellular Nanoscale Sensory Wave Computing*, p. 5, 2009.
- [25] T. Roska, C. Wu, and L. Chua, „Stability of cellular neural networks with dominant nonlinear and delay-type templates,” *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 40, no. 4, pp. 270–272, 1993.
- [26] L. Chua and T. Roska, „The cnn paradigm,” *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 40, no. 3, pp. 147–156, 1993.
- [27] T. Roska and L. Chua, „The cnn universal machine: An analogic array computer,” *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 40, no. 3, pp. 163–173, 1993.
- [28] T. Roska, „Cellular wave computer architectures in a new era of computing 15 years later,” *International Journal of Circuit Theory and Applications*, vol. 36, no. 5-6, pp. 523–524, 2008.

-
- [29] P. Pereira, A. Grilo, F. Rocha, M. Nunes, A. Casaca, C. Chaudet, P. Almström, and M. Johansson, „End-to-end reliability in wireless sensor networks: survey and research challenges,” in *EuroFGI Workshop on IP QoS and Traffic Control*. Citeseer, 2007, pp. 67–74.
- [30] C. Wan, A. Campbell, and L. Krishnamurthy, „Psfq: a reliable transport protocol for wireless sensor networks,” in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. ACM, 2002, pp. 1–11.
- [31] F. Zhao, J. Shin, and J. Reich, „Information-driven dynamic sensor collaboration for tracking applications,” *IEEE Signal processing magazine*, vol. 19, no. 2, pp. 61–72, 2002.
- [32] J. Levendovszky, G. Kiss, and L. Tran-Thanh, „Energy balancing by combinatorial optimization for wireless sensor networks,” *Performance Modelling and Analysis of Heterogeneous Networks*. River Publishers, Aalborg, Denmark, pp. 169–182, 2009.
- [33] J. Levendovszky, A. Bojársky, B. Karlócai, and A. Oláh, „Energy balancing by combinatorial optimization for wireless sensor networks,” *WSEAS Transactions on Communications*, vol. 7, no. 2, pp. 27–32, 2008.
- [34] J. Levendovszky, A. Olah, G. Treplan, and L. Tran-Thanh, „Reliability-based routing algorithms for energy-aware communication in wireless sensor networks,” *Performance Models and Risk Management in Communications Systems*, pp. 93–126, 2011.
- [35] G. Ran, H. Zhang, and S. Gong, „Improving on leach protocol of wireless sensor networks using fuzzy logic,” *Journal of Information & Computational Science*, vol. 7, no. 3, pp. 767–775, 2010.