

Applying neural network based algorithms in communication technology

című értekezés tézislevele

Tisza Dávid

Témavezetők:

Dr. János Levendovszky

dr. Oláh András



Pázmány Péter Katolikus Egyetem
Információs Technológiai és Bionikai Kar
Roska Tamás Műszaki és Természettudományi
Doktori Iskola

Budapest, 2018

1. Technológiai motivációk

A kommunikációs és szenzoros technológia fejlődésének köszönhetően számos olyan alkalmazási jelenik meg, amely képes a felhasználói oldalon a mobilitás és flexibilitás kielégítésére. Ezek a felhasználási területek egyre komplexebb rendszereket kívánnak meg, így a rendszer háttérében működő hálózati elemeknek is egyre intelligensebben kell tudni viselkedni és alkalmazkodni a felsőbb rétegek által támasztott követelményekhez.

Tipikusan ilyen területek az **IoT** alapú alkalmazások, szenzor hálózatokkal végzett monitorozó illetve beavatkozó rendszerek [1], a multimédia folyamatokat továbbító szolgáltatások de ide sorolhatjuk a háttér hálózatokra épülő „felhő alapú” szolgáltatásokat [37] is. Ezek a szolgáltatásoknak egy közös jellemzője, hogy egy előírt szolgáltatási minőséget (**QoS**) kívánnak biztosítani, ezzel egy időben viszont adott erőforrásokat csak korlátozottan használhat a rendszer [42]. Ezek az erőforrás korlátok a vezeték nélküli rádiós technológiákban hatványozottan jelennek meg.

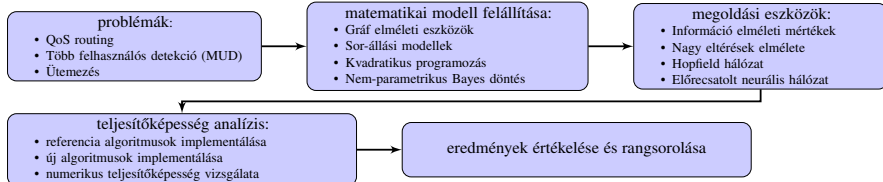
A jelenleg használt hálózati technológiák réteges szerveződésűek és legnagyobb többsége csomagkapcsolt kommunikációs rendszerként működik. Azokban a hálózatokban, ahol nincs külön csatorna a jelzési(signaling) és kontroll folyamatokhoz az előírt minőség biztosítása újabb erőforrásokat igényel. Ennek következtében a felhasználók számára használható kapacitás tovább csökken. Azok a széles körben használt, tradicionális csomagkapcsolt hálózati struktúrák amiket „best effort” működés módra terveztek, általában nem használnak ilyen külön csatornát. Ilyen protokollok például az OSI szabvány által definiált [14] **PHY** rétegben az Ethernet, a 802.11 vagy a hálózati rétegben az **IP**. Annak ellenére, hogy a tervezésük alapján közvetlenül nem támogatják a **QoS**-t [29, 17], nagyon elterjedtek a megbízhatóságuk és az átjárhatóságuk miatt [17].

A csomagkapcsolt jellegből adódóan a hálózat működésének alapvető kérdései közé tartozik, hogy az adat csomagokat milyen útvonalon, milyen ütemezéssel juttatja el a hálózat a nyelőhöz [35, 17, 25, 6, 5]. Rádiós környezetben ugyanakkor a fizikai és adatkapcsolati rétegben a protokolloknak újabb kényszereket kell figyelembe venni a közös hozzáférési közeg használata miatt. Ezek eredményeképpen a nyitott kérdések amelyeket a disszertáció válaszokat keres:

- Hogyan lehet előírt minőségű útvonal és úthálózat választást elérni egy csomagkapcsolt hálózatban (QoS unicast, multicast routing),
- Hogyan lehet hatékony, jól párhuzamosítható algoritmust adni a hálózatban felmerülő ütemezési problémákra (scheduling)
- Hogyan lehet közel optimálisan teljesítő, jól párhuzamosítható algoritmust adni a vezeték nélküli adatkapcsolati rétegben felmerülő több felhasználót egyszerre kiszolgáló detekciós problémákra (Multi User Detection)

2. Alkalmazott kutatási módszertan

A disszertáció a vizsgált problémákat a következő módszerekkel vizsgálja:



A disszertáció által vizsgált problémákat és a felhasznált algoritmikus eszköztár összefoglalóját a 1-es táblázat tartalmazza.

<i>technológiai problémák</i>	<i>modell formalizálás és probléma típus</i>	<i>elméleti teljesítőképesség</i>	<i>felhasznált módszerek, algoritmusok</i>	<i>kapcsolódó tézisek</i>
Unicast routing	véletlen link-leírók és link meghirdetési sémák (LAS) alkalmazásával a probléma visszavezetése additív illetve „legszűkebb keresztmetszet” típusú mértékekre	A választott útvonal a kezdő és végpont közt egy előírt minőségi kritérium (QoS) teljesítésére képes	Sor-állási modellek, Markov modulált Poisson eloszlások, gaussi közelítés, nagy eltérések elmélete,	I.1. I.2.
Link meghirdetési séma optimalizálása	információ elméleti mértékek (Link Entrópia és signaling Entrópia) alkalmazásával kényeszeres optimalizáció	A megtervezett LAS használatával a hálózat állapot hirdetésének az adatforgalma egy előírt sávszélesség alatt marad, miközben egy meghatározott küszöböt nem lép túl a linkekről ismert bizonytalanság mértéke	információ elméleti mértékek, kimerítő keresés, általános nemlineáris kényeszeres optimalizációs algoritmusok	I.4.
Multicast routing	véletlen link-leírók alkalmazásával kényeszer vezérelt gráf minimális Steiner fa (CGMST) keresés és a (CGMST) keresési probléma UBQP megfogalmazása	A választott úthálózatban a kezdő és végpontok között garantáltak az előírt minőségű adat folyamok, miközben egy meghatározott kényeszer szerint (energia fogyasztás, teljes felhasznált sávszélesség) minimális a választott úthálózat.	Gaussi közelítés, Nagy eltérések elmélete, Hopfield neurális hálózat, bináris kvadratikus programozási eljárások	I.3.

- folytatás a következő oldalon -

táblázat 1 – folytatás az előző oldalról

<i>technológiai problémák</i>	<i>modell formalizálás és probléma típus</i>	<i>elméleti teljesítőképesség</i>	<i>felhasznált módszerek, algoritmusok</i>	<i>kapcsolódó tézisek</i>
Multi User Detection	UBQP vagy nem-parametrikus bayesi detekció	Az UBQP egzakt megoldása a probléma optimális megoldása. A kidolgozott UBQP megoldók jól párhuzamosíthatók, szuboptimális megoldást találnak gyors konvergencia idő mellett. A nem-parametrikus modell esetén a javasolt algoritmus tetszőlegesen megközelítheti az optimális döntés minőségét, a komplexitás növelésének rovására.	Új hipergráfon értelmezett dimenzió redukción ill. addíción alapuló algoritmus, Hopfield neurális hálózat, bináris kvadratikus programozási eljárások, előrecsatolt mesterséges neurális hálózat (FFNN), logaritmikusan keresés	II.1, III.1
Ütemezés (Scheduling)	UBQP	Az UBQP egzakt megoldása a probléma optimális megoldása. A kidolgozott megoldók szuboptimális megoldást találnak gyors konvergencia idő mellett	Hopfield hálózat, bináris kvadratikus programozási eljárások	II.1

1. táblázat. Vizsgált problémák, felhasznált algoritmikus eszköztár és kapcsolódó tézisek

3. Új tudományos eredmények összefoglalása

A disszertáció három téziscsoportban mutatja be a vizsgált problémákat és a javasolt megoldási módszereket:

- I) Az első téziscsoportban a csomagkapcsolt hálózatokban előforduló előírt minőségű útvonal és úthálózat keresésének problémáit foglalom össze. Ebben a csoportban vizsgálom továbbá a „link scaling” problémát és az optimalizálásra javaslok egy eljárást információ elméleti mértékek alapján.
- II) A második téziscsoportban pedig a kommunikációs technológiában előforduló problémák közül a többfelhasználós detekciót és az ütemezés problémáját kvadratikus bináris programozási feladatként írom le és egy Hopfield hálózat alapú szuboptimális, jól párhuzamosítható eljárás családot mutatok be.
- III) A harmadik téziscsoportban a vezeték nélküli kommunikációban a fizikai rétegben használt többfelhasználós detekciós problémára adok egy nem-parametrikus Bayes döntéssel összemérhető teljesítményű előrecsatolt neurális hálózat alapú algoritmust

I. téziscsoport - unicast és multicast útvonal választás véletlen link-leírók esetén

Az első téziscsoport a csomagkapcsolt kommunikációs hálózatokhoz mutat be új neurális hálózatokon illetve információ elméleti mértékeken alapuló algoritmusokat az útvonal választási és hirdetési problémákhoz.

A szerző ide vonatkozó publikációja „Multicast Routing in Wireless Sensor Networks with Incomplete Information” [45] címmel jelent meg.

Csomagkapcsolt hálózatokban ahol nincs külön csatorna fenntartva a signaling folyamatoknak a hálózaton belüli információ hirdetés további erőforrásokat kíván. Ez a falszanáló számára hasznos sávszélességet csökkenti, miközben az a használt szolgáltatások igénylik hálózat felőli sebességet és megbízhatóságot. Ezek az egymásnak ellentmondó feltételek teremtik meg a QoS útvonal választás problémáját. Többek között az IoT világban az eszközök fogadnak és küldhetnek is információt, sőt relay eszközként is funkcionálhatnak. Egy nagy csoportja ezen eszközöknek elemről működik, miközben elvárjuk tőlük a megbízható adat továbbítást. (pl. az intelligens otthon környezetben a felhasználó nem szabad, hogy elem cserére legyen kényszerítve gyakran vagy mezőgazdasági területre telepített szenzor hálózat, ahol az elem csere kivitelezhetetlen.) Másrészt olyan alkalmazásoknál, ahol az energia felhasználás csökkentése nem kritikus (pl. az automotív iparban, az intelligens otthonban egy hűtő esetén vagy egy intelligens gyártósornál) más fajta megbízhatósági kritériumoknak kell megfelelnie a hálózatnak. Többek közt ilyenek például a robusztusság, redundancia vagy a hatékony sávszélesség felhasználás. Ezekben a hálózatokban vagy a

peer-to-peer multimédia alkalmazásokban is gyakran előforduló kommunikációs séma, amikor egy információ forrás egy vagy több célponthoz azonos információt akar eljuttatni (Unicast ill. Multicast adatforgalom). A korlátozott hálózati erőforrások és a csomagkapcsolt jelleg miatt (alapvetően elterjedt IP tervezési döntései következtében) az egyik fő kihívás, hogy hogyan lehet egy adott QoS-t (késletetés, garantált sávszélesség, stb.) biztosítani az adatforgalom számára az információ forrás és a fogadó fél között. Egy adatfolyam QoS feltételeinek biztosítása a hálózat szempontjából történhet olyan utak illetve útházatok kiválasztásával, ahol a pont-pont(ok) közti késletetés, vagy sávszélesség követelmények nem lépnek túl egy előírt statisztikai határt.

A tradicionális csomagkapcsolt hálózatok amik nem úgy lettek tervezve eredetileg, hogy QoS-t legyenek képesek biztosítani általában nem rendelkeznek külön signaling csatornával. Ilyen protokollok többek közt a PHY rétegben az Ethernet, a 802.11 vagy a hálózati rétegben az IP. Annak ellenére, hogy nem a QoS garantálására lettek tervezve [29, 17], mégis rendkívül elterjedt a használatuk az egyszerűségük, megbízhatóságuk és az együttműködési képességük miatt [17]. Így szükséges kiterjeszteni ezeket a protokollokat, hogy támogatni tudják a QoS adattovábbítást. Az egyik fő kihívás ezekben a hálózatokban, hogy hogyan tudunk végpontok közti előírt késletést biztosítani [35, 17, 25, 6, 5] a tradicionális minimális ugrásszám alapú útvonal választáshoz képest. A „manifold optimalizációs kritérium” [15] miatt már unicast esetben is a QoS útvonal választás feladat egy sokkal nehezebb matematikai feladat mint a tradicionális, ugrásszám alapú útvonal választás [15]. Az előírt minőségű útvonal választás beintegrálására ezen hálózatokba több tanulmány létezik. Például az IP hálózatokra [17], vagy a klasszikus „IP over ATM” rendszere [29], Több „QoS aware” útvonalválasztó algoritmus is létezik, amik különböző szinteken kezelik a problémát, például inter-domain szinten [16], MPLS alapú hálózati részekben [28], az AS-ek határain [35].

A QoS alapú útvonalválasztó protokolloknak ismerniük kell a hálózat forgalmi állapotát. Ezt az információt áramoltatni kell az útvonalválasztó csomópontok közt (signaling). A dolgozat olyan csomagkapcsolt hálózatokkal foglalkozik, ahol a signaling folyamatok ugyanazt az erőforrást használják, mint amit a felhasználók számára hasznos adat A hálózatban minél távolabb van két hálózat rész, annál pontatlanabb ismerettel rendelkeznek egymás állapotáról. Ezért kiemelkedő ezeknek, a hálózat működtetése számára fontos adatfolyamoknak a pontos, ugyanakkor erőforrás szempontjából minimális áramoltatása. Azért, hogy a signaling folyamat erőforrás igényeit csökkentsék, a QoS érzékeny protokollok rendszerint olyan technikákat használnak, amivel a hálózat állapotára vonatkozó információt igyekeznek összevonni statisztikai paraméterekbe (aggregáció, illetve LAS bevezetése). Ez az információ aggregáció egy újabb bizonytalansági tényezőt visz a rendszerbe amellet, hogy az információ forrásokat keltő felhasználók is véletlen módon ingadozó adatfolyamokat generál-

nak. Mindezek mellett a tradicionális útvonalválasztó protokollok (pl. **OSPF**, **BGP**) hierarchikus módon is működnek, így az egyes alhálózatok egy-egy aggregált csomópontként látszanak a felsőbb hierarchiai szinteken. Egy alhálózatról ezért a felsőbb szintek csak statisztikai aggregált információkat ismernek. Így az alhálózatokon belüli véletlen forgalmi fluktuáció újabb bizonytalanságot vezet be a felsőbb szintek számára a hálózat aktuális állapotára vonatkozólag. Ezeket a mértékeket a hálózat periodikusan hirdeti az útvonalválasztó csomópontok között.

Ezek indokolják, hogy az egyes csomópontokon a forgalom állapotát valószínűségi változókkal modellezzük [11, 40, 24]. A fent leírt jelenségek adnak okot, hogy a tradicionális minimális ugrásszám alapú útvonal választás helyett, a feladatot statisztikai és információ elméleti mértékek alapján definiáljuk. Így a feladat olyan útvonalak illetve útvonal hálózatok kiválasztása, amelyek egy előírt valószínűségi kritériumot elérve biztosítják az adatfolyamok számára a **QoS** kritériumokat [11, 40, 39, 23, 12].

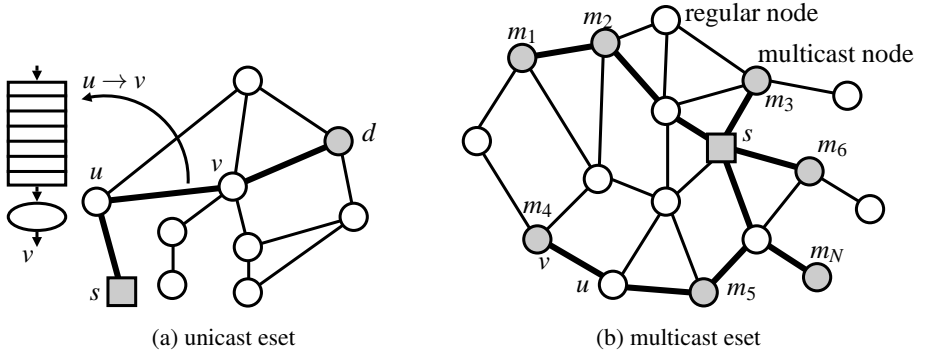
A multicast eset felfogható az előző probléma kiterjesztéseként, ami a probléma **GSMT** megfogalmazásához vezet. Már az egyszerűbb **GSMT** probléma is NP-nehez komplexitású determinisztikus él leírók esetén [20]. A **CGSMT** probléma még nehezebb feladat, ahol a minimális súlyú fa keresése egyidejűleg történik egy kényszer által előírt előírt minőség biztosításával [21]. Ezért szeretnénk felhasználni a **HNN** [34] előnyös tulajdonságait, amely ugyan várhatóan szuboptimális megoldást szolgáltat, azonban a megoldás minősége várhatóan jó lesz. Az algoritmus sebessége ugyan akkor a hálózat összekötöttségével arányos, mivel egy-egy neuron egy összeköttetést reprezentál a hálózatban.

A **QoS** ezzel együtt viszont erősen függ hálózat állapotáról elérhető információ-tó bizonytalanságától, amit az állapothirdetésre(Link State Advertisement - LSA) használt felosztásrács határoz meg. Ezt a felosztásrácsot hívjuk „Link Advertisement Scheme”-nek (LAS). Ennek a felosztásrácsnak a beállítását nevezzük „link scaling”-nek, amit ha alkalmasan állítunk be, akkor ez mindamelllett hogy csökkenti a jelzési sávszélességet az információ bizonytalanságát sem csökkenti. Minél finomabban tagolt ez a felosztásrács, a mért paraméter annál gyakrabban ugrál az egyes intervallumok közt, hirdetést generálva. A gyakori hirdetések miatt az elérhető információ pontos lesz, segítve ezzel jó minőségű útvonalak megtalálását. Azonban a hirdetések gyakorisága miatt a hálózat a felhasználók számára hasznos kapacitásból használ fel egyre többet a saját jelzési folyamatára. Ezek miatt a LAS optimális beállítása egy hálózatüzemeltetési fontos kérdés, amit „optimal link scaling (OLS)”-nek neveznek. Az OLS, ahogy látható egy kényszeres optimalizálási probléma, ahol a hálózat teljesítményét szeretnénk maximalizálni mindamelllett, hogy a jelzési sávszélességet egy előírt küszöbszint alatt tartjuk.

A kommunikációs hálózat modellezésére így egy gráfot használhatunk:

$$G(V, E, \delta_{(u,v)}, F_{(u,v)}(x)),$$

ahol V a hálózatban szereplő csomópontokat, E az összeköttetéseket, $\delta_{(u,v)}$ az egyes összeköttetések link-leíróit és $F_{(u,v)}(x)$ az egyes link-leírók eloszlás függvényeit jelentik.



1. ábra. A hálózat gráf modellje

Az unicast esetben a feladat, hogy keressünk egy optimális utat \tilde{R} -t az összes lehetséges útvonal közül $\mathcal{R}^{s \rightarrow d}$, mely teljesíti az adott **QoS** kritériumot („bottleneck” és additív mértékekre):

$$\tilde{R} = \operatorname{argmax}_{R \in \mathcal{R}^{s \rightarrow d}} \mathbb{P} \left(\min_{(u,v) \in R} \delta_{(u,v)} \geq B \right) \quad \tilde{R} = \operatorname{argmax}_{R \in \mathcal{R}^{s \rightarrow d}} \mathbb{P} \left(\sum_{(u,v) \in R} \delta_{(u,v)} < T \right)$$

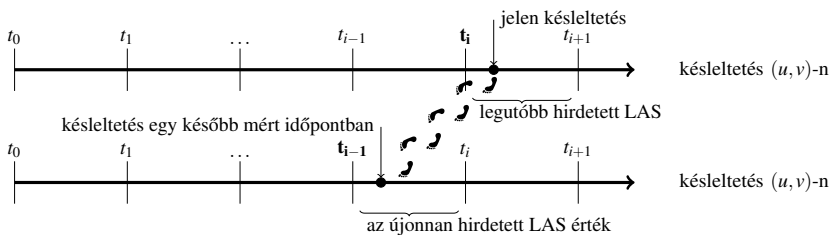
A multicast esetre az információ forrást (tipikusan egy bázis állomás) $s \in V$, a nyelők halmazát $M = \{m_1, m_2, \dots, m_N\} \subset V$ jelöli. Ez esetben a feladat, hogy egy optimális fát találjunk a gráfban \tilde{A} a lehetséges multicast fák közül $\mathcal{A}^{s \rightarrow M}$, amely a legnagyobb valószínűséggel teljesíti az előírt **QoS** kritériumot („bottleneck” és additív mértékekre)

$$\tilde{A}_1 = \operatorname{argmax}_{A \in \mathcal{A}^{s \rightarrow M}} \mathbb{P} \left(\max_{(u,v) \in A} \delta_{(u,v)} < P \right) \quad \tilde{A}_2 = \operatorname{argmax}_{A \in \mathcal{A}^{s \rightarrow M}} \mathbb{P} \left(\max_{\substack{R_{sm} \in \mathcal{R}_T \\ \in \\ R_{sm}}} \sum_{(u,v)} \delta_{(u,v)} < T \right)$$

Megjegyzendő, hogy a multicast eset felírása kissé eltér az unicast esettől, mert itt a link-leírók tipikusan fogyasztást jelölnek pl. egy **WSN** hálózat esetén. Ezért

a „bottleneck” ebben az esetben a hálózatban részt vevő legnagyobb fogyasztású elemről működtetett eszköz. De a következtetések kis módosítással alkalmazhatóak sávzélesség típusú „bottleneck” leírókra is.

A link-leíróról nyerhető információt a linkek küszöbintervallumainak hirdetései-ből nyerhetjük empirikus úton. A legtöbb hálózatban a hálózati elemek valamilyen formában küldenek a saját állapotukról információkat (**LSA**). Például a **PNNI** vagy **OSPF**, de a **WSN** hálózatokban is a proaktív routing protokollok esetében a hálózati elemek hirdetnek magukról információt. Mi az **LSA**-kat kihasználva új minőségi paraméter értékeket hirdetünk a hálózat számára. A késletetést a következőképpen hirdetjük: adott egy nem feltétlenül ekvidisztáns felosztású rács amit a mért hálózati paraméter értékeire illesztünk. A csomópontok felderítő csomagokat küldenek a szomszédjaiknak. Amennyiben a mért érték kikerül abból az intervallumból, amiben előzőleg volt, akkor az adott kapcsolathoz egy új t_i érték fog tartozni, amely minden linkre összesítetten **LSA** csomag formájában meghirdetődik a hálózatban. Az alábbi ábra egy hirdetési eseményt mutat be, ahol a link-leíró késletetés jellegű



2. ábra. **LSA** késletetés típusú link-leírók esetén.

A link késletetéséről, vagy más mért paramétréről tehát nincs egy pontos érték, hanem csak egy intervallum, amelybe beleesik. A küszöb intervallumok halmaza $T = \{t_i, i = 1, \dots, N\}$. Egy t_i alsó intervallum érték akkor hirdetődik meg a k időpillanatban, ha $\delta_{(u,v)}(k) \in (t_i, t_{i+1})$ és $\delta_{(u,v)}(k-1) \notin (t_i, t_{i+1})$. A jelzési folyamat ezért tekinthető egy információ elméleti forrásnak $\kappa_{(u,v)}$ ami a $t_i, i = 0, \dots, Z-1$ szimbólumokat küldheti $P_i, i = 0, \dots, Z-1$ valószínűségekkel. Ezeket a valószínűségeket a véletlen szerűen változó forgalom határozza meg.

A **LAS** optimalizálása újabb tárgya a minőség javításnak abban az értelemben, hogy még tovább lehessen csökkenteni és jobban ki lehessen használni a signaling sáv-zélességet (az irodalomban **OLS**). Minél kisebbek ezek a hirdetendő intervallumok, annál kisebb a határozatlanság a link állapota felől. Ez, mivel a minőségi bizonytalanságot csökkenti, így a routing minőségét javítja. Ezt a két mennyiséget felírhatjuk két

információ elméleti mérték (link entrópia és jelzési entrópia) segítségével:

$$\text{link entrópia: } H(\delta_{(u,v)} | \nu_{(u,v)})$$

$$\text{signaling entrópia: } H(\kappa_{(u,v)} | \chi_{(u,v)})$$

A link entrópia annak a mértéke, hogy egy (u, v) linkről milyen bizonytalan az információ, ha tudjuk, hogy $\nu_{(u,v)} = i$, azaz a t_i szimbólum lett hirdetve az (u, v) linkre. A jelzési entrópiában $\kappa_{(u,v)}$ jelenti azt az információ elméleti forrást, ami a $\{t_1, t_2, \dots, t_N\}$ halmazból küld szimbólumokat, és $\chi_{(u,v)}$ pedig azt, hogy az információ elméleti forrás aktív-e. Ezzel a két entrópia értékkel felül lehet becsülni a signaling folyamatok sávszélességi igényeit.

I.1. TÉZIS (unicast útvonal választás véletlen link-leírókkal Gaussi közelítés alapján). *Megfogalmaztam egy transzformációt a link-leírókra, azzal a feltételezéssel, hogy a leírók normális eloszlást követnek m és $\tilde{\sigma}_{(u,v)} = \sqrt{m_{(u,v)}}$ paraméterekkel, valamint hogy a LAS felosztási rács $m = \frac{t_{i+1} + t_i}{2}$. Ebben az esetben*

1. Állítás. *ha $\delta_{(u,v)}$ normális eloszlású $\tilde{\sigma}_{(u,v)} = \sqrt{m_{(u,v)}}$ paraméterekkel, akkor az ARII megoldása*

$$\tilde{R} = \operatorname{argmax}_{R \in \mathcal{Q}^{s \rightarrow d}} \mathbb{P} \left(\sum_{(u,v) \in R} \delta_{(u,v)} < T \right)$$

egyenértékű a következő feladat megoldásával

$$\tilde{R} = \operatorname{argmin}_R \sum_{(u,v) \in R} m_{(u,v)}$$

ami tradicionális legrövidebb út kereső algoritmusokkal(pl. Bellman-Ford) polinom időben megoldható.

A fenti feltételek esetén az ARII feladatot átfogalmaztam tradicionális determinisztikus SPR feladattá.

I.2. TÉZIS (unicast útvonal választás véletlen link-leírók esetén rekurzív eljárással). *Eljárásokat adtam csomagkapcsolt hálózatokban használható útvonal választási eljárásra ami egy előírt minőségű útvonalat tud találni adott valószínűségi kritériummal az 1. és 2. algoritmusokban.*

Az algoritmusok azon alapulnak, hogy a véletlen link-leírókat a nagy eltérések elmélete felhasználásával áttranszformálják determinisztikus mértékekre. A transzformációt a 2 állítás tartalmazza:

2. Állítás. *A log-moment generáló függvény felhasználásával*

$$\mu_{(u,v)}(s) = \ln \mathbb{E} \left(\exp (s\delta_{(u,v)}) \right) = \ln \int_{-\infty}^{\infty} \exp (sx) dF_{(u,v)}(x), \quad (3.1)$$

vagy diszkrét valváltozó esetén

$$\mu_{(u,v)}(s) = \ln \mathbb{E} \left(\exp (s\delta_{(u,v)}) \right) = \ln \sum_{i=1}^{\infty} \exp (sx_i) p_i, \quad (3.2)$$

az ARII megoldása ekvivalens a következő függvény minimalizálásával

$$\tilde{R} = \operatorname{argmin}_R \sum_{(u,v) \in R} \mu_{(u,v)}(\hat{s}) \quad (3.3)$$

ahol az optimális \hat{s} paraméter a következő:

$$\hat{s} = \inf_s \sum_{(u,v) \in \tilde{R}} \mu_{(u,v)}(s) - sT. \quad (3.4)$$

A két algoritmus a következő:

Algoritmus 1 Exhaustive-s algoritmus

Input: $G(V, E, \delta_{(u,v)}, F_{(u,v)}(x)), src, dst$

Definiáljunk egy felosztás rácsot $\mathcal{S} = \{s_i, s_i > 0, i = 1, \dots, M\}$.

for all $i = 1, \dots, M$ **do**

válasszunk egy $s_i \in \mathcal{S}$ és keressünk egy utat R_i egy tradicionális **SPR** algorit-mussal a következő mértékekkel: $\mu_{(u,v)}(s_i) := \ln \mathbb{E}(\exp(s_i \delta_{(u,v)}))$.

A választott R_i út alapján számoljuk ki a hozzá tartozó

$$\hat{s}_i = \text{Solve} \left[\sum_{(u,v) \in \tilde{R}_i} \frac{d\mu_{(u,v)}(s)}{ds} = T, s \right],$$

és számoljuk ki a hozzá tartozó küszöbértéket

$$B_i := \exp \left(\sum_{(u,v) \in R_i} \mu_{(u,v)}(\hat{s}_i) - \hat{s}_i T \right).$$

end for

Majd keressünk utat ami a minimális küszöbértékhez tartozik

$$\tilde{R}_j : j = \underset{i}{\operatorname{argmin}} B_i.$$

Output: \tilde{R}_j a választott út src és dst közt

Algoritmus 2 The Recursive Path Finder - s Finder algoritmus

Input: $G(V, E, \delta_{(u,v)}, F_{(u,v)}(x)), src, dst$

válasszunk egy $s \leftarrow$ pozitív kezdőértéket

repeat

rendeljük minden $(u, v) \in E$ linkhez a $\mu_{(u,v)}(s)$ mértéket.

keressünk útvonalat **SPR** algorit-mussal $\tilde{R}(s)$ a választott s -hez.

A választott \tilde{R} útvonalhoz határozzuk meg \tilde{s} -t

$$\tilde{s} = \frac{\mu'^{-1} \left(T - \sum_{(u,v) \in R} a_{(u,v)} \right)}{|R|}.$$

$s \leftarrow \tilde{s}$.

until $\tilde{R}(\tilde{s}) \neq \tilde{R}(s)$

Output: \tilde{R}_j a választott út src és dst közt

Teljesítőképesség analízis: Ahhoz, hogy az algoritmusok teljesítőképessége összehasonlítható legyen, először definiálok egy mértéket mellyel két út (R_a és R_e) minősége hasonlítható össze, ahol mindkét út src -ből indul és dst -be érkezik. Ez a mérték egy arányszám két valószínűség közt, hogy R_e (a kimerítő keresés által talált útvonal) és R_a (a vizsgált algoritmus által választott útvonal) milyen valószínűséggel teljesítik az előírt minőségi kritériumot. Ehhez egy segéd mérték:

$$\eta(R_a, T) := \mathbb{P} \left(\sum_{(u,v) \in R_e} \delta_{(u,v)} < T \right) - \mathbb{P} \left(\sum_{(u,v) \in R_a} \delta_{(u,v)} < T \right), \quad (3.5)$$

Mivel R_e a lehető legjobb út, abban az értelemben, hogy ez az út teljesíti a legnagyobb valószínűséggel a T QoS kritériumot, ezért $\eta(R_a, T) \geq 0$. Ez a mérték a lehető legjobb útvonalhoz képesti teljesítmény csökkenést méri egy adott T QoS értékre.

A normalizált görbék alatti terület a következő:

$$\chi(R_a) = \int_0^{T_{\max}} \eta(R_a, T) dT \Big/ \int_0^{T_{\max}} \mathbb{P} \left(\sum_{(u,v) \in R_e} \delta_{(u,v)} < T \right) dT, \quad (3.6)$$

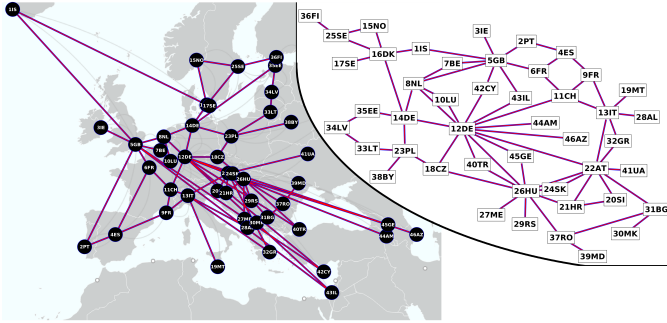
ahol T_{\max} a vizsgált legnagyobb QoS érték. χ tehát az útvonal minőségét számszerűsíti a legjobb útvonalhoz képest T értékétől függetlenül. Könnyű belátni, hogy az a legrosszabb útvonal (R_0), ami nem tud teljesíteni semmilyen $T \leq T_{\max}$ értéket csak 0 valószínűséggel. Ennek a mérőszáma $\chi(R_0) = 1$. A legjobb útvonal pedig $R_a = R_e$ amihez a $\chi(R_e) = 0$ érték tartozik.

Hogy általánosabb, az algoritmusokra jellemző teljesítmény értékeket kapjak, egy gráf halmazon is futtattam az algoritmusokat, nemcsak egy konkrét topológián. Ebben a véletlenszerűen generált elemekből álló gráf halmazban minden gráfnak 10 csomópontja volt, legalább 3-as kardinalitással és minden csomópont egy komponenshez tartozott. Az összes lehetséges utat egy G gráfban (minden lehetséges kezdő src -ből minden dst végpontba, $src \neq dst$) $\mathcal{R}(G)$ -vel jelölöm.

Ezen a halmazon értelmezett teljesítményt a következőképpen $\chi(G)$ -vel és χ_e -vel jelölöm, melyek:

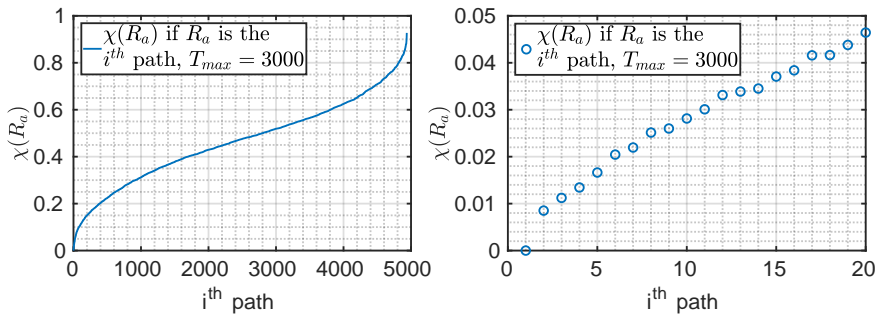
$$\chi(G) := \frac{1}{|\mathcal{R}(G)|} \sum_{R \in \mathcal{R}(G)} \chi(R) \quad \chi_e := \frac{1}{|\mathcal{G}|} \sum_{G \in \mathcal{G}} \chi(G) \quad (3.7)$$

Az algoritmusok teljesítménye a kimerítő keresés teljesítményéhez lett mérve. A vizsgált algoritmusok: „OSPF”, „Gaussian approximation”, „exhaustive-s”, „recursive-s”. Az „OSPF” az egyszerű legrövidebb út kereső algoritmus a meghirdetett LAS értékét használva metrikának. Míg a többi az általam előzőleg javasolt algoritmusok.



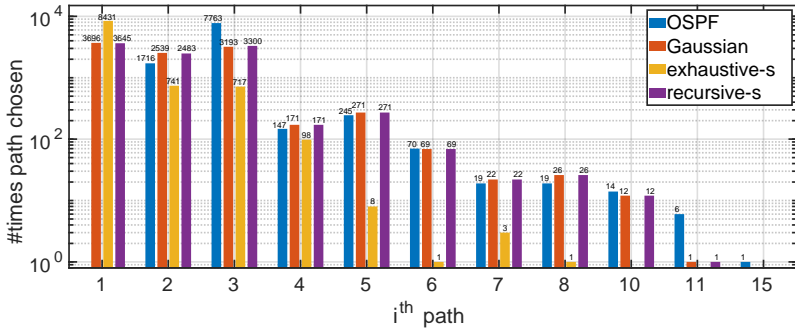
3. ábra. GEANT topológia a javasolt algoritmusok teljesítőképességének vizsgálatára.

A GEANT topológiát használva Izlandot „IISL” választottam forrás csomópontnak *src* és Romániát „37RO” *dst* cél csomópontnak, mert e két állomás közt változatos útvonalak léteznek. Az összes útvonal *src*-ből *dst*-be megcímkézhető egy indexszel (1-től 4937-ig) növekvő sorrendben $\chi(R_a)$ szerint. A felcímkézett utakat és a hozzájuk tartozó mértéket a 4-es ábra mutatja. Az általam választott és szimulált link-leírókat

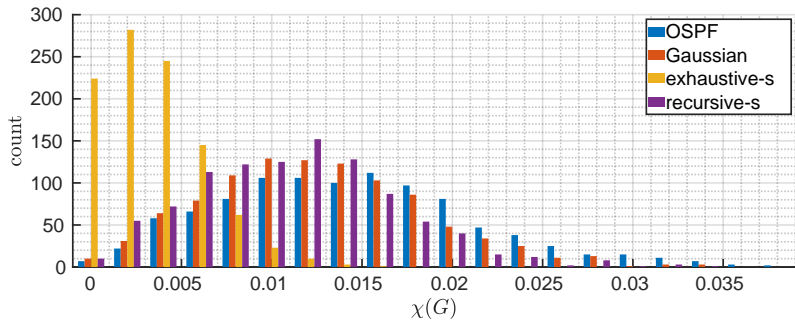


4. ábra. Útvonal index vs χ , $T_{max}=3000$. A bal oldali ábra mutatja mind a 4937 utat, míg a jobb az első 20-at mutatja nagyítva.

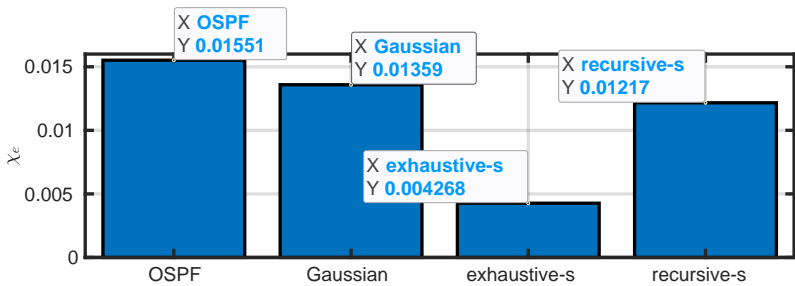
használva mértem a hálózatot. Amikor az OSPF megtalálta a legjobb útvonalat (#1-es útvonal) az összes többi is. A használt link-leíró eloszlásokra a relatív gyakorisága annak, hogy az OSPF nem a legjobb utat választotta $P = 0.4148$ volt. Tehát közel 40%-ban létezett volna egy jobb útvonal. 10000 mintapontból álló statisztikára vizsgáltam a teljesítőképességet, amikor az OSPF nem a legjobb utat választotta. Ebben az esetben a relatív gyakorisága, hogy melyik algoritmus melyik útvonalat választja a 5-ös, a gráf halmazra számolt teljes mérték χ_e pedig a 7-es ábrán látható.



5. ábra. választott útvonalak relatív frekvenciája 10^4 minta esetén, amikor az OSPF nem a legjobb utat (#1) választotta a GEANT topológián



6. ábra. A gráf halmaz esetén az egyes G gráfon mért teljesítmények ($\chi(G)$) eloszlása.



7. ábra. A gráf halmazon számolt teljesítmény értékek.

I.3. TÉZIS (multicast útvonal választás véletlen link-leírók esetén **HNN**-el). *Definiáltam egy algoritmust, amivel szuboptimális megoldás adható a multicast útvonal keresési problémára véletlen link-leírók esetén:*

Algoritmus 3 optimális fa keresése end-to-end kritérium mellett

Input: $G(V, E, \delta_{(u,v)}, F_{(u,v)}(x))$, $\kappa = 1$, $T > 1$ *src, m*

válasszunk G , $\kappa = 0$, $T > 1$

repeat

 A = keressünk fát HNN-el(G, κ, T)

if A találtuk **then**

 csökkentsük κ

else

 növeljük κ

end if

until nincs szignifikáns minőség javulás

Output: A a multicast fa *src* és *m* közt.

Az eljárás a véletlen link-leírókat transzformálja determinisztikus mértékekre a nagy eltérések elmélete segítségével:

$$\tilde{A}_2 : \operatorname{argmin}_{A \in \mathcal{A}} \sum_{(u,v) \in A_E} C_{uv},$$

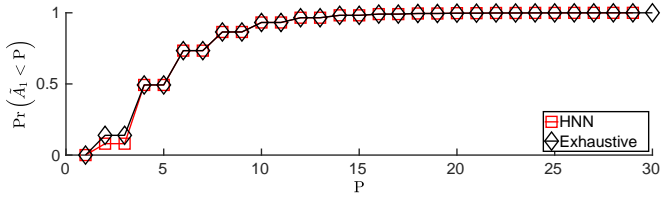
$$\text{s.t. } \mu_{R_{src,m}}(s) < \ln(\kappa) + s \cdot T,$$

*A transzformált problémáról belátható hogy egy **Constrained Graph Steiner Minimal Tree (CGSMT)** probléma, aminek nem ismert polinomiális idejű optimális megoldása, azonban az általam javasolt algoritmus **HNN** segítségével „elég jó” megoldásokat talál. A hálózat paramétereit a disszertáció 2.4.3-es „Approximation by HNN” című fejezete tartalmazza, ami összefoglalva:*

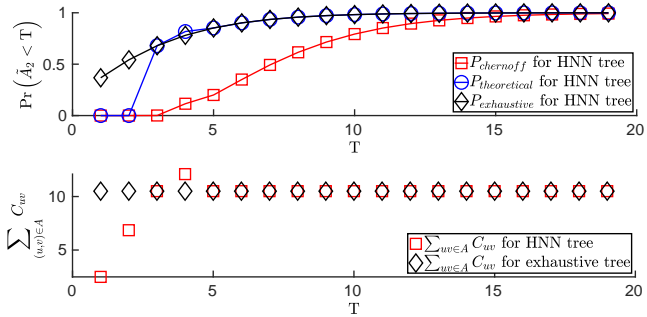
$$\mathcal{E}(\mathbf{y}) = \alpha_1 \left(2\mathbf{y}^{tr} \mathbf{b}^{(1)} \right) + \alpha_2 \left(\mathbf{y}^{tr} \mathbf{W}^{(2)} \mathbf{y} + 2\mathbf{y}^{tr} \mathbf{b}^{(2)} \right),$$

Teljesítő képesség analízis: Az \tilde{A}_1 és az \tilde{A}_2 célfüggvényeket a javasolt algoritmusokkal és kimerítő kereséssel is kiértékeltem a gráfokon a következő paraméterekkel: a hálózat mérete $N = 8$, a Rayleigh fading-gel terhelt kommunikációs csatorna paramétereit tipikus beltéri értékek mellett szimuláltam: $\gamma = 3$, $g = 1$, $\theta = 10$, $s^2 = 1$. A csomópontok pozíciói véletlen szerűen i.i.d. egyenletes eloszlás szerint lettek generálva az egység alapterületen. 3 multicast célpont lett véletlen szerűen lettek kiválasztva. A teljesítő képessége az analitikusan számolt eloszláshoz illetve a kimerítő keresés eredményéhez képest is ábrázolva vannak, hogy az egyes lépések hatásai láthatóak

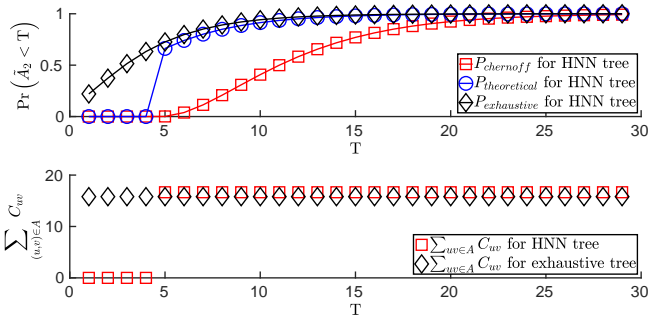
legyenek.



8. ábra. Egy tipikus \tilde{A}_1 célfüggvény.



(a) Optimálshoz közeli \tilde{A}_2 célfüggvény



(b) Egy tipikus \tilde{A}_2 célfüggvény

9. ábra. A multicast fa kereső algoritmus teljesítőképessége additív link-leírókra

I.4. TÉZIS (optimizing link scaling using MMPP). Modellt adtam az eredeti kényszeres optimalizálási problémára, ami a hálózat állapotának „pontos ismerete”, annak a hatása (*Link Entropy*) és a hozzá tartozó jelzési sávszélesség (*Signaling Entropy*) közt teremti meg a kapcsolatot. Ezzel a modellel számíthatóak a fenti információ elméleti mértékek egy előírt valószínűség mellett:

$$\min_{\Delta t} H(\delta_{(u,v)} | v_{(u,v)}(\Delta t))$$

$$s.t. H(\kappa_{(u,v)}(\Delta t) | \chi_{(u,v)}(\Delta t)) < \alpha$$

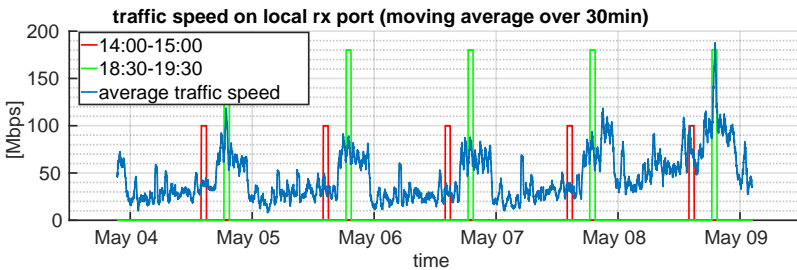
A modell számítható, mert a link-leírók dinamikáját MAP/M/I folyamatokkal modelleztem:

$$H(\kappa_{(u,v)}(\boldsymbol{\pi}, \mathbf{D0}, \mathbf{D1}, \mu, \Delta t) | \chi_{(u,v)}(\Delta t))$$

$$H(\delta_{(u,v)}(\boldsymbol{\pi}, \mathbf{D0}, \mathbf{D1}, \mu) | v_{(u,v)}(\boldsymbol{\pi}, \mathbf{D0}, \mathbf{D1}, \mu, \Delta t))$$

Következésképpen az kívánt egyensúly beállítható a modell kiértékelésével és a megfelelő munkapont beállításával.

Teljesítő képesség analízis: A forgalmi modelljeimet a publikus DISCO adathalmaz méréseire alapoztam, melyet a „Measurement-lab” szervezet készít [30]. Az „mlab1.dub01.measurement-lab.org” szerverhez csatlakoztatott switch értékeivel dolgoztam. 2016 júniusa óta az M-Lab nagy idő felbontású (10 másodperces) switch telemetria adatokat gyűjt az M-lab szervereinek forgalmáról. A „Bytes received by the machine switch port” és a „Unicast packets received by the machine switch port” mezőket használtam a forgalmi modelljeim statisztikájának alapjául. Azzal a



10. ábra. A forgalom napi szintű önhasonlósága az mlab1.dub01.measurement-lab.org szerverről. Mozgó átlaggal képzett idősor 2018. május 4-9 közti időszakra.

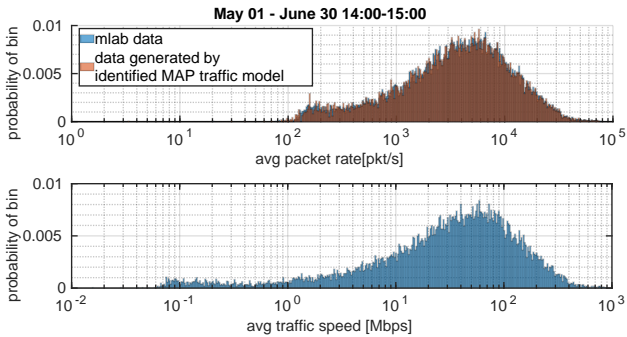
feltételezéssel éltem, hogy egy órán belül a forgalom kellően statikusan viselkedik,

a napi fluktuációkhoz mérten. Erre alapozva két napi intervallumot választottam amiből készítettem a statisztikát a forgalmi modellekhez.

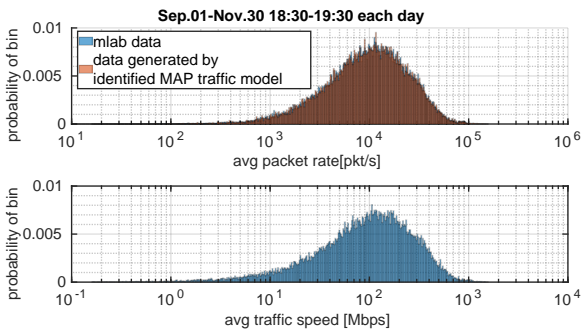
Egy átlagos forgalmi terhelést: 14:00-15:00 között minden napra 2018. máj. 1. és jún. 30. közt. Az átlagos csomag ráta 6066.341pps, std:6865.61pps, max:75944.03pps és az átlagos sebesség 62Mbps, std:78.28Mbps, max:883.66Mbps volt.

Egy intenzívebb forgalmi terhelésre: 18:30-19:30 között minden napra 2018. szept. 1. és nov. 30. közt. Ekkor az átlagos csomag ráta 13613.81pps, std:12278.3pps, max:137180.7pps és az átlagos sebesség 143.5Mbps, std:141.5Mbps, max:1605.45Mbps volt.

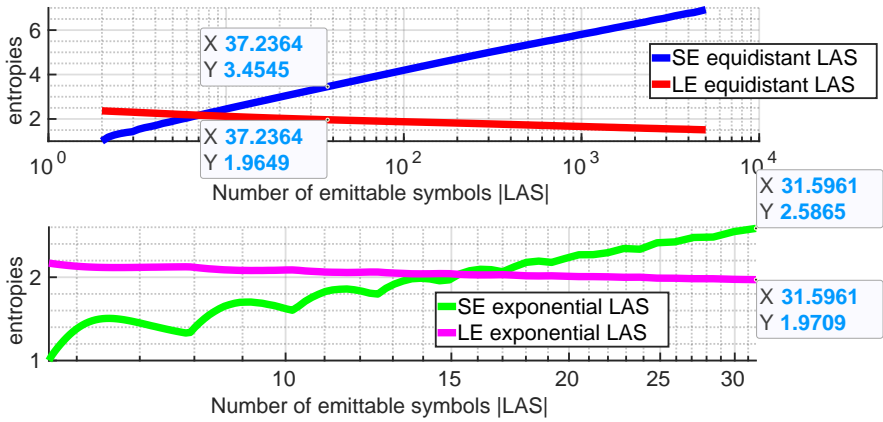
16 állapotú MAP struktúrát használtam mindkét esetben, hogy a forgalmat modellezzem. A modellekkel generált forgalom, azonos statisztikával rendelkezik a valós forgalmi adatokkal. A MAP struktúra tehát kellően komplex, hogy a valós adatokat leképezze. A 13-as és 14-es ábrán a két információ elméleti mérték (**SE**, **LE**) látható a link-leírók felett értelmezett felosztás rács elemeinek számának függvényében.



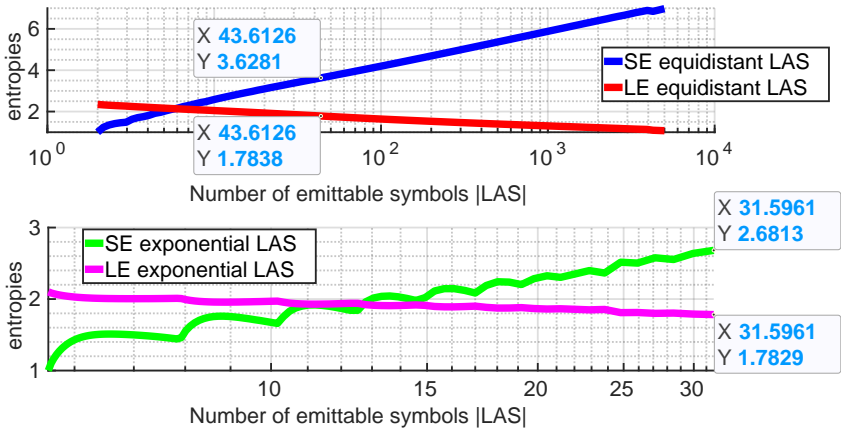
11. ábra. Forgalmi statisztika 14:00-15:00 2018. máj. 1. - jún. 30.



12. ábra. Forgalmi statisztika 18:30-19:30 2018. szept. 1. - nov. 30.



13. ábra. Információ elméleti mértékek az átlagos forgalmi esetre.



14. ábra. Információ elméleti mértékek az második forgalmi esetre.

II téziscsoport - heurisztika és alkalmazása az **UBQP** megoldására hipergráf alapú reprezentáció segítségével

A téziscsoportban a kommunikációs technológiákban előforduló **UBQP** problémákhoz mutatok be új neurális hálózat alapú megoldókat. Az **UBQP** fontos szerepet tölt be az optimalizáció területén, mert számos célfüggvény megfogalmazható kvadratikus alakban. A kommunikációs technológiákban a számos probléma, úgy mint a **MUD**, „machine scheduling” visszavezethető az úgy nevezett **UBQP** kombinatorikus optimalizálási problémára. Összefoglaló található az alkalmazásokról [19]-ben.

„Cloud computing” környezetben és az **IoT** paradigmában az egyik központi probléma a feladatok megfelelő ütemezése [27, 49], a feladatok hatékony elosztása, melyek a skálázhatóságot és a teljesítményt befolyásolják. Számos megközelítés létezik [38, 48, 41] - általában metaheurisztikák - amely algoritmusok részeként valamilyen kényszeres optimalizálási problémára keresnek megoldást. Ez a lépés általában megfogalmazható **UBQP**-ként, így a javasolt algoritmusok alkalmazhatóak ezekre a problémákra. Egy friss összefoglaló található a **IaaS** cloud computing környezetben használt ütemező és terhelés megosztó algoritmusokról [38, 48, 41].

Továbbá olyan problémák, amik lineáris vagy kvadratikus kényszereket tartalmaznak, szintén megfogalmazhatóak **UBQP** problémaként, mint ahogy az bemutatásra [18] és alkalmazásra [3, 31, 47], [7, 26] került. Általános esetben az **UBQP** NP nehéz probléma [9], de léteznek olyan esetek, amikor egzaktul megoldható polinomiális idejű algoritmusokkal [3, 32, 33, 2]. Ezek ellenére és éppen ezért szükség van gyors, megfelelő minőségű és skálázható algoritmusokra, amik közel optimális megoldást nyújtanak, amikor a probléma mérete miatt az egzakt megoldás nem valósítható meg. A téziscsoport olyan algoritmusok bemutatásával foglalkozik, amik egy hipergráf alapú reprezentáción keresnek dimenzió növeléssel vagy csökkentéssel közel optimális megoldásokat. Az algoritmusokat a következő három alkalmazási területen teszteltem:

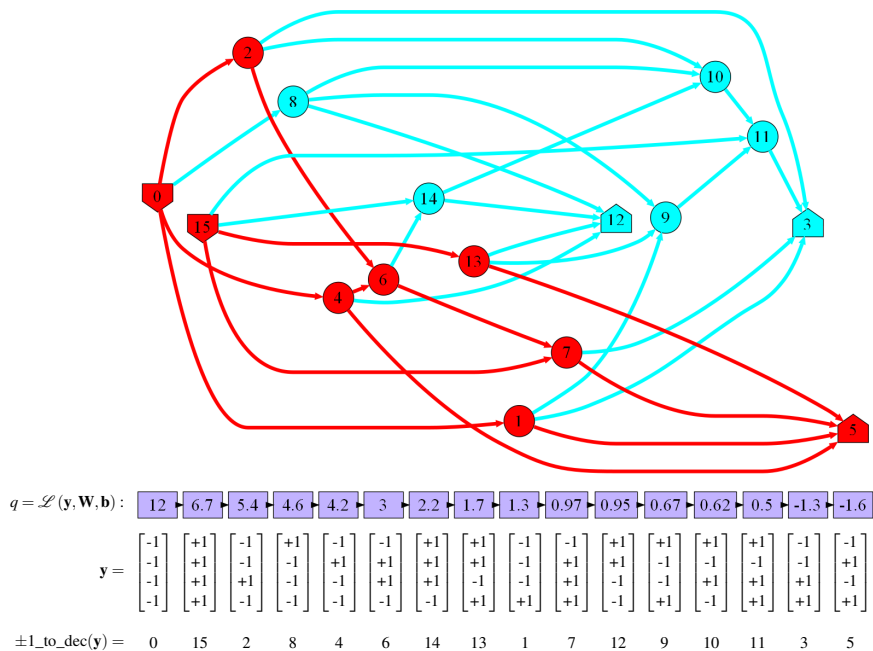
az ORLIB által definiált **UBQP** teszt feladatok; multiuser detekció - **MUD**; ütemezés - scheduling.

A szerző ide vonatkozó folyóirat cikke „Novel algorithms for quadratic programming by using hypergraph representations” [44] címmel jelent meg.

Az **UBQP** egy kvadratikus kombinatorikus optimalizálási probléma, ahol a változó vektor elemei y két diszkrét értéket vehetnek fel. Ezek általában a $+1$, -1 értékek.

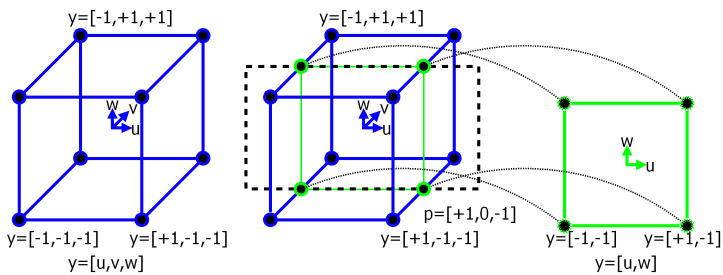
$$\begin{aligned}\mathcal{L}(\mathbf{y}, \mathbf{W}, \mathbf{b}) &:= \mathbf{y}^T \mathbf{W} \mathbf{y} - 2\mathbf{y}^T \mathbf{b}, \\ \mathbf{y} &\in \{\pm 1\}^N, \mathbf{b} \in \mathbb{R}^N, \mathbf{W} \in \mathbb{R}^{N \times N} \\ \mathbf{y}_{opt} &= \min_{\mathbf{y} \in \{\pm 1\}^N} \mathcal{L}(\mathbf{y}, \mathbf{W}, \mathbf{b})\end{aligned}$$

A változó vektor \mathbf{y} egy konkrét értéke (állapot) megfeleltethető egy N dimenziós hiperkocka egy csúcának. Ennek a hiperkockának a csúcspontjait megfeleltethetjük egy gráf csomópontjaival, és a csomópontokhoz hozzárendelhetjük az adott állapot célfüggvény értékét, így egy csúc súlyozott gráfot kapunk. Ha a hiperkockát el-

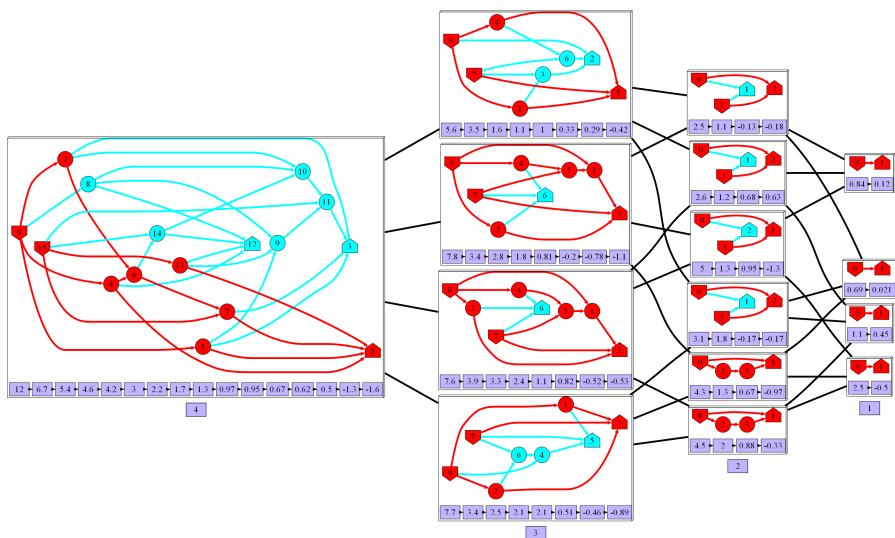


15. ábra. Egy 4 dimenziós UBQP keresési tere - 4 dimenziós hiperkocka csúcjai

metsszük egy adott dimenzióan áthaladó hipersíkkal, egy $N - 1$ dimenziós hiperkockát kapunk, amivel hasonlóképpen eljárhatunk mint az előbb, így kapunk egy újabb fele akkora méretű gráfot. Ha ezt a módszert rekurzívan alkalmazzuk minden lehetséges kombinációban, egy hipergráfot kapunk. Ezen a hipergráfon mint állapottéren végzünk keresést, így a keretrendszer lehetővé teszi a már meglévő UBQP megoldók használatát.



16. ábra. A 2. dimenzió „eldobása” metszéssel a 3 dimenziós „hiper”kockából



17. ábra. A 4 dimenziós UBQP hipergráfja

II.1. TÉZIS (Hipergráf alapú heurisztikus algoritmus család az UBQP megoldására). *Algorithm 4*-ban definiáltam egy hipergráf alapú, könnyen párhuzamosítható algoritmus családot, amely szuboptimális megoldást ad az UBQP problémára.

Az algoritmusok az eredeti keresési teret egy hipergráf reprezentációra vetítik és HNN alapú belső megoldót használnak. Négy különböző példányát definiálom az algoritmus családnak, mely közül kettő dimenzió csökkentést, kettő dimenzió „építést” használ. A Táblázat 3 foglalja össze a bemutatott algoritmusok működési módjait. A részletes leírás megtalálható a disszertáció E függelékében a - „Description and performance characterization of the new UBQP related algorithms” cím alatt.

3. táblázat categorization of the algorithms

	greedy	opportunistic
dim. reducer	L01	D01
dim. adder	DA02	DA01

Az eljárások teljesítőképességét három különböző problémahalmazon vizsgálom: a standard ORLIB által definiált **UBQP** teszthalmazon, egy ütemezési problémán, és egy szimulált **MUD** problémán. Bemutatom, hogy a javasolt eljárások közel optimálisan teljesítenek a vizsgált infokommunikációs problémákon.

Algoritmus 4 az algoritmus család pszeudokódja

```
1: function INNER_SOLVER( $\mathbf{W} \in \mathbb{R}^{k \times k}$ ,  $\mathbf{b} \in \mathbb{R}^k$ ,  $\mathbf{y}(\text{init}) \in \{\pm 1\}^k$ )
2:   a kedvenc BQP minimalizáló algoritmusod
3:   return  $\mathbf{y} \in \{\pm 1\}^k$ 
4: end function
5: function  $\Psi(u \in V_H, \mathbf{y} \in u_V)$ 
6:   válasszuk ki  $\hat{u} \in V_H$  ▷ a következő hipernode választása
7:   válasszuk ki  $\hat{\mathbf{y}} \in \hat{u}_V$  ▷ a választott hipernodeban a kezdő állapot
8:   return  $\hat{u}, \hat{\mathbf{y}}$ 
9: end function

10:  $\hat{u} \leftarrow u(\text{init}) \in V_H$  ▷ a kezdő hipernode választása
11: válasszunk egy  $\hat{\mathbf{y}} \in u(\text{init})_V$  ▷ kezdő állapot a hipernode-ban
12: repeat
13:   legyen  $L(\mathbf{W}, \mathbf{b}, \mathbf{y})$  a célfüggvény
14:    $u \leftarrow \hat{u}$  és  $\mathbf{y} \leftarrow \hat{\mathbf{y}}$ 
15:    $\mathbf{W}, \mathbf{b} \leftarrow$  paraméterek  $u = G(V, E, Q(\mathbf{W}, \mathbf{b}))$ 
16:   if FUTTASSUK_E_AZ_INNER_SOLVERT( ) then
17:      $\mathbf{y}^* \leftarrow$  INNER_SOLVER( $\mathbf{W}, \mathbf{b}, \mathbf{y}$ )
18:   else
19:      $\mathbf{y}^* \leftarrow \mathbf{y}$ 
20:   end if
21:    $\hat{u}, \hat{\mathbf{y}} \leftarrow \Psi(u, \mathbf{y}^*)$  ▷ a következő hipernodeba ugunk
22: until STOP_CRIT( )
```

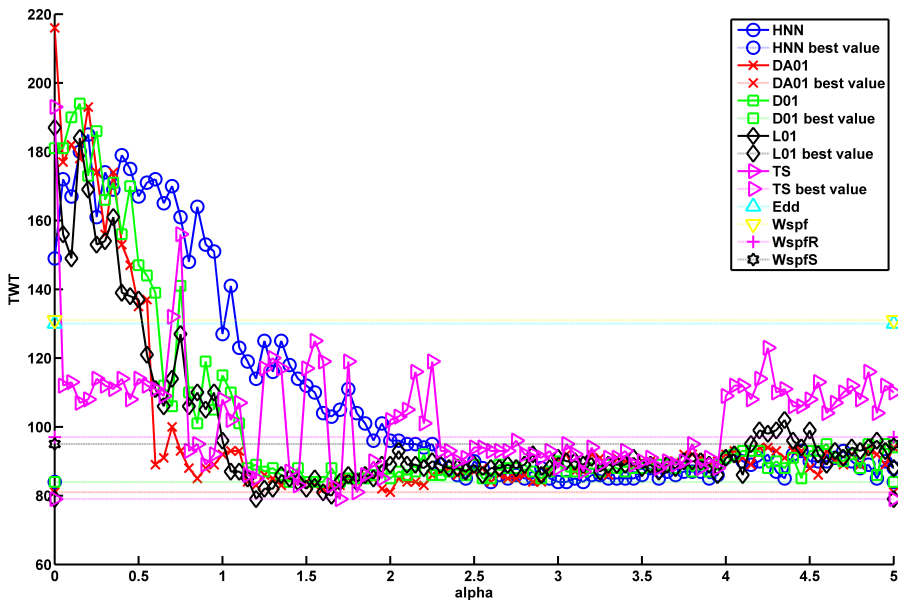
Teljesítő képesség analízis: Az algoritmusok teljesítő képességét más referencia algoritmusokkal egyetemben az ORLIB teszt halmazra a következő táblázat foglalja össze. A javasolt algoritmusok mellett a következő tradicionális **UBQP** megoldók is szerepelnek:

- „HNN” - a diszkrét idejű, diszkrét értékű Hopfield hálózat.
- „1-opt **LS**” - a 1-opt lokális keresés.
- „BLS” Beasley [3] által javasolt algoritmus, ami egy 1-opt lokális keresésen alapul.
- „BTS” a taboo search algoritmus szintén Beasley-től [3].
- „DDT” a **DDT** algoritmus (Boros, Hammer, and Sun [4])
- „SDR” egy **SDR** típusú algoritmus randomizálás nélkül (Luo et al. [26]).
- „SDR with randomization” mint az előző csak randomizálással [26].

4. táblázat Performance comparison of BTS vs DA01

		E-solution		best solution		relative freq of best solution		mean run time		mean run time until best solution found	
dim=50 K=10000	prob\alg	BTS	DA01	BTS	DA01	BTS	DA01	BTS	DA01	BTS	DA01
	1	2160	2055,4	2160	2160	1	0,1052	232,23	1	232,2286	9,5057
	2	3658	3588,6	3658	3658	1	0,1316	251,05	1	251,0545	7,59878
	3	4650	4680,4	4650	4778	1	0,4924	240,36	1	240,3645	2,03087
	4	3472	3400,7	3472	3472	1	0,1275	234,53	1	234,5311	7,84314
	5	4152	4098,5	4152	4152	1	0,7053	234,99	1	234,9933	1,41784
	6	3842	3823,6	3842	3842	1	0,5405	239,71	1	239,7059	1,85014
	7	4588	4535,1	4588	4588	1	0,2746	241,11	1	241,107	3,64166
	8	4222	4195,2	4222	4222	1	0,8435	233,71	1	233,7129	1,18554
	9	3862	3829,9	3862	3862	1	0,6273	228,99	1	228,9912	1,59413
10	3496	3450,8	3496	3496	1	0,2163	236,26	1	236,264	4,62321	
dim=100 K=10000	prob\alg	BTS	DA01	BTS	DA01	BTS	DA01	BTS	DA01	BTS	DA01
	1	7910	7631	7910	7910	1	0,0039	65,814	1	65,8142	256,41
	2	11178	11030	11178	11178	1	0,0192	75,619	1	75,61899	52,0833
	3	12956	12875	12956	12956	1	0,1854	73,601	1	73,60066	5,39374
	4	10606	10493	10606	10606	1	0,0722	70,157	1	70,15745	13,8504
	5	8994	8777,2	8994	8996	1	0,0173	70,659	1	70,6588	57,8035
	6	10470	10362	10470	10486	1	0,011	71,521	1	71,5213	90,9091
	7	9980	9877,6	9980	10030	1	0,0182	72,419	1	72,41868	54,9451
	8	11380	11240	11380	11380	1	0,1357	71,742	1	71,74218	7,3692
	9	11340	11246	11340	11340	1	0,269	79,918	1	79,91803	3,71747
10	12438	12348	12438	12438	1	0,032	79,959	1	79,95852	31,25	
dim=500 K=1000	prob\alg	BTS	DA01	BTS	DA01	BTS	DA01	BTS	DA01	BTS	DA01
	1	116526	114780	116526	116532	1	0,001	13,642	1	13,6418	1000
	2	128678	127801	128678	128678	1	0,002	14,043	1	14,0425	500
	3	131084	130013	131084	131084	1	0,006	13,294	1	13,2944	166,667
	4	129794	128646	129794	129784	1	0,001	13,209	1	13,2092	1000
	5	125008	123859	125008	125062	1	0,001	11,694	1	11,6937	1000
	6	121868	120189	121868	121868	1	0,001	13,336	1	13,3355	1000
	7	122730	121163	122730	122756	1	0,001	13,989	1	13,989	1000
	8	123454	121958	123454	123428	1	0,001	13,811	1	13,8106	1000
	9	121622	120026	121622	121668	1	0,001	13,889	1	13,8885	1000
10	130900	130219	130900	131374	1	0,01	13,865	1	13,8647	100	

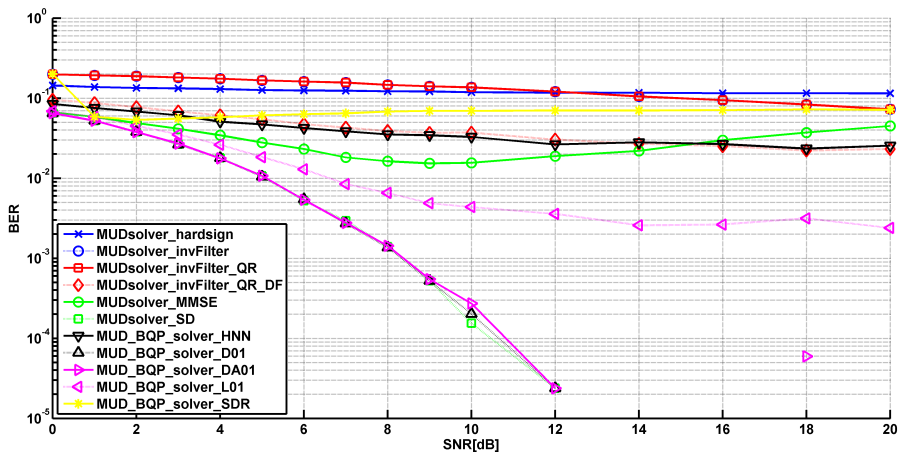
Az ütemezési probléma esetén tradicionális algoritmusokkal hasonlítottam össze a javasolt eljárásokat, úgy mint a „Earliest Deadline first”, „Weighted Shortest Processing Time first” és variációi. A teljesítőképeség mértéknek a „Total Weighted Tardiness”-t választottam, ami a feladatok átlagos késleltetésével van arányban. Ebben az esetben az algoritmusok által elért legjobb **TWT** eredmények HNN:84, DA01:81,



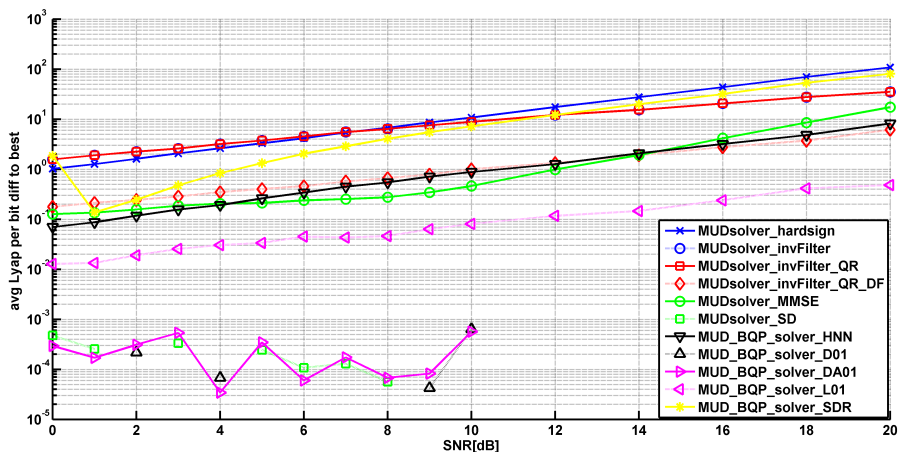
18. ábra. TWT teljesítő képesség az α heurisztikus paraméter függvényében $J = 10$ esetre

D01:84, L01:79, TS:79, EDD:130, WSPT:131 WSPTR:97, WSPTS:95. Erre az esetre az L01 és a Taboo Search algoritmusok teljesítenek a legjobban kissé jobb eredményt elérve a többi általam javasolt algoritmus mellett.

A MUD problémára a következő algoritmusokkal hasonlítottam össze a javasolt algoritmusokat [22]: „threshold” egyszerű szignum detektor az illesztett szűrők után Az „invFilter” a generalizált „Zero-Forcing” detektor, ami a csatorna interferencia kiegyenlítésére törekszik, így minimalizálva a felhasználók által okozott interferenciát A „QR” és a „DF” változatai az algoritmusoknak a QR faktorizációt és a döntés visszacsatolást is alkalmazzák. Az „MMSE” a négyzetes hibát minimalizáló detektor, míg az „SD” a „sphere detector” algoritmust jelöli. A következő ábrák egy nem telített csatornára mutatják be az algoritmusok teljesítő képességét: $M = 7$ felhasználó kommunikált egyszerre az alábbi szimulációkban. Látható, hogy a javasolt algoritmusok közel teljesítenek az elméleti határhoz, ahogy az „SD” is, míg a hagyományos detektorok erre nem képesek. Az ábráról látható, hogy a kvadratikusan célfüggvény valóban jól karakterizálja a bithibaarányt, mivel a BER és a kvadratikusan célfüggvény érték együtt mozog. Látható, hogy a „D01” és az „SD” teljesít a legjobban, de a „DA01” is a közelükben teljesít. Megjegyzendő, hogy a „DA01” algoritmus komplexitása jóval alacsonyabb, mint a „D01”-é, hiszen az előbbi opportunistikus heurisztikát



19. ábra. BER teljesítő képesség 7 felhasználóra



20. ábra. A választott megoldások átlagos kvadratikus célfüggvény értékei a 7 felhasználós szimulációra

használ, míg az utóbbi egy „mohó” algoritmus.

A „sphere detector” szintén az elméleti határhoz közeli megoldásokat szolgáltat, azonban a komplexitása jóval magasabb, mint a „DA01”-e. Ezen felül a javasolt algoritmusok természetüknél fogva jól párhuzamosíthatóak.

III táziscsoport - nem-parametrikus közel optimális teljesítı k pess g  detektor **Feed Forward Neural Networks** seg ts g vel

Extensive simulations and the corresponding numerical analysis demonstrate that the proposed algorithms yield near optimal performance on real channel models (COST-207).

Ebben a t ziscsoportban  j k dol si technik t mutatok, amivel nem-parametrikus neur lis h l zat alap  detektor  p thet  zajjal terhelt mintakeres si feladatra. Ez az alapvet  probl ma számos m rn ki feladatban el fordul, mint p ld ul „big data” minta illeszt si feladatokban, automatiz lt biztons gtechnikai  s megfigyel si elj r sokban, hang  s k pfelismer sben, tartalom alap  keres sben vagy a t bbfelhaszn l s hozz f r si technik kban **MUD** el fordul  probl m kban. Az elj r st a **MUD** probl m n mutatom be, de az elj r s k nnyen kiterjeszthet  m s  ltal nosabb probl m ra is. Az **FFNN** alap  detekci s elj r s a fel ll tott modellben tetsz leges pontoss ggal param ter mentesen reproduk lja a Bayes d nt st. Ehhez egy line ris k dol si elj r st  s egy hozz  tartoz  **FFNN** strukt r t adok.

A szerz  ide vonatkoz  foly irat cikke „Multi-user detection using non-parametric Bayesian estimation by feed forward neural networks” [43] c mmel jelent meg.

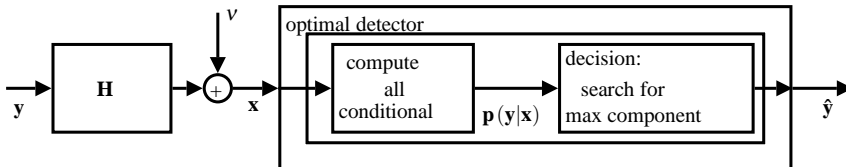
Az optim lis bayesi MAP d nt st  ltal nos estben a k vetkez k ppen fogalmazhatjuk meg a modell alapj n [46]:

$$\hat{\mathbf{y}} = f_{opt}(\mathbf{x}) = \underset{\mathbf{y} \in \{\pm 1\}^L}{\operatorname{argmax}} \mathbb{P}(Y = \mathbf{y} | X = \mathbf{x}),$$

Ugyanakkor a **MAP** d nt s ekvivalens azzal, hogy megkeress k a maxim lis  rt k  komponenszt a k vetkez  vektorb l: $\mathbf{p}(\mathbf{y}|\mathbf{x})$. Vegy k észre, hogy direkt m don hozz f rni a vektorhoz exponenci lis sz m t si komplexit st ig nyel, mivel $N = 2^L$.

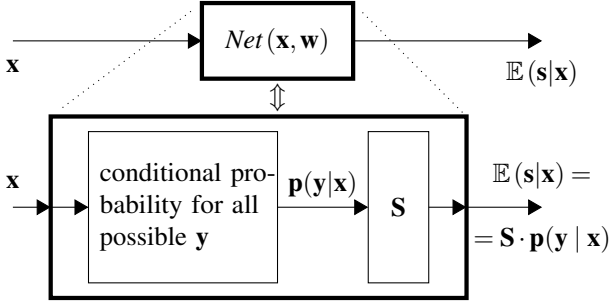
$$\hat{\mathbf{y}}_{opt} = \mathbf{y}^{(i)} : i = \underset{n \in 1 \dots N}{\operatorname{argmax}} p_n(\mathbf{y}|\mathbf{x})$$

Az elm leti exponenci lis komplexit s  detektor a k vetkez   br n l that 

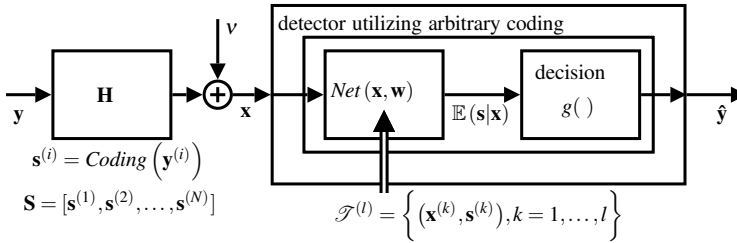


21.  bra. az elm leti exponenci lis komplexit s  optim lis detekrot blokkdiagramja

III.1. TÉZIS (nem-parametrikus detekció intervallum felezéses kódolási eljárással **FFNN** segítségével). *Definiáltam egy lineáris kódolási sémán és **FFNN** alapuló, könnyen párhuzamosítható nem-parametrikus detektort a **MUD** probléma megoldására, amely optimális teljesítményt nyújt az alábbi feltétel teljesülése esetén:*



22. ábra. az FFNN és a kódolási stratégia együttműködése



23. ábra. a detektor blokkdiagramja tetszőleges lineáris kódolás esetén

$$j = \max_{i \in 1 \dots N} p(\mathbf{s}^{(i)} | \mathbf{x})$$

if $j \in E$

$$\sum_{k \in E} p(\mathbf{s}^{(k)} | \mathbf{x}) > \sum_{i \in \{1 \dots N\} \setminus E} p(\mathbf{s}^{(i)} | \mathbf{x}).$$

A következőben megadom az intervallum felezésen alapuló lineáris kódolás kódszavait amit a tanító halmaz előállítására lehet használni:

$$S_{i,j} = s_i^{(j)} = \text{sgn} \left(-\sin \left(2\pi \cdot 2^{(i-1)} \cdot \frac{j}{N+1} \right) \right), \quad i = 1 \dots L, \quad j = 1 \dots N$$

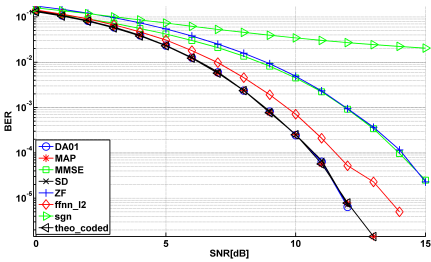
és a kis komplexitású döntő függvény, ami a hálózat kimenetén megjelenő jelből állítja elő a döntési eredményt:

$$\hat{\mathbf{y}} = \text{sgn}(\mathbb{E}(\mathbf{s}|\mathbf{x})).$$

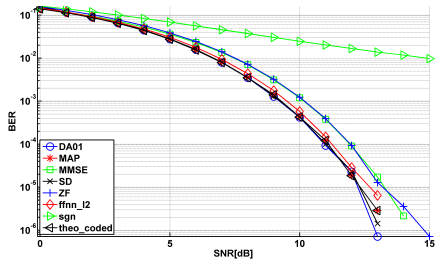
Szimulációs eredményeken mutatom be, hogy a javasolt algoritmus közel optimálisan teljesít a vizsgált problémákra, miközben nincs szüksége a csatorna előzetes ismeretére.

Teljesítő képesség analízis: Összehasonlításnak a javasolt algoritmusok mellett feltüntettem a következő detektorokat is: MMSE, ZF, SDR valamint az elméleti határt. A szimuláció a COST 207 [8] csatorna modelleket használja. Négy különböző típusú modell, „COST207 Hilly Terrain 6 tap alternative”, „COST207 Rural Area 6 tap”, „COST207 Typical Urban 12 tap” és „COST207 Bad Urban 12 tap” kerül alkalmazásra.

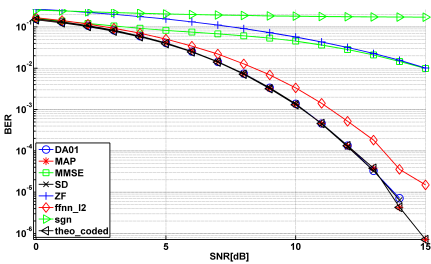
A következő ábrán látható az algoritmusok teljesítő képessége $M = 7$, $K = 2$ és $L = 14$ esetre. A képekről látszik, hogy a javasolt algoritmus, az elméleti határ közelében teljesít. Az elméleti határtól való eltérést az magyarázza, hogy az eljárás egyfelől támaszkodik a tézisben megfogalmazott feltétel meglétére, valamint a hálózat tanulási hibája asszimptotikus értelemben tűnik el, azaz az FFNN nem tudta hibátlanul reprodukálni a feltételes valószínűségeket a tanítóhalmaz alapján. Ezzel egyidőben nem sokkal marad alul a legjobban teljesítő **SD**-hez képest. Megjegyzendő, hogy míg az **SD** egy parametrikus detektor a csatorna tökéletes ismeretével, a javasolt algoritmus nem. Az „FFNN I2” algoritmus kissé teljesít rosszabbul, mint az optimális **MAP** döntés.



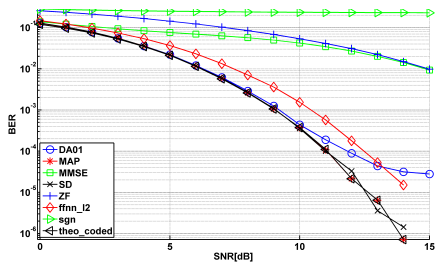
(a) BER a „COST 207 Hilly Terrain 6 tap alternative” csatorna esetén



(b) BER a „COST 207 Rural Area 6 tap” csatorna esetén

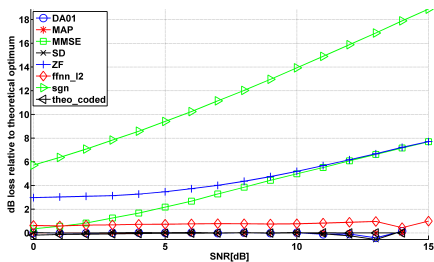


(c) BER a „COST 207 Typical Urban 12 tap” csatorna esetén

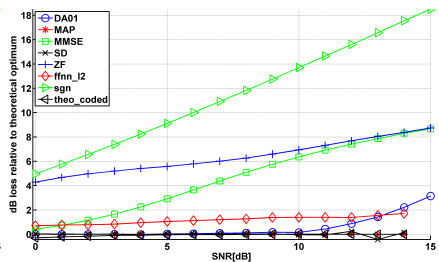


(d) BER a „COST 207 Bad Urban 12 tap” csatorna esetén

24. ábra. teljesítő képesség görbék a $L = 14$ esetre négy különböző csatorna modellre



(a) SNR veszteség görbe $L = 10$ paraméterre a „COST 207 Bad Urban 12 tap” csatorna esetén



(b) SNR veszteség görbe $L = 14$ paraméterre a „COST 207 Bad Urban 12 tap” csatorna esetén

4. Összefoglalás és záró gondolatok

A disszertációmban a következő kérdésekre adtam választ:

Csomag kapcsolt hálózatokban hogyan lehet hatékonyan olyan útvonalat vagy fát kiválasztani, amelyek teljesítenek egy adott QoS értéket bottleneck vagy additív típusú link-leírók esetén. Ezt a hálózat forgalmi működéséről gyűjtött statisztika felhasználásával érem el úgy, hogy a meglévő modelleket olyan alakúra transzformálom, hogy azok könnyen illeszthetők legyenek a tradicionális útvonal választó algoritmusokhoz. Továbbá az útválasztó eljárások hatékonysága tovább növelhető az általam javasolt „link scaling” eljárással úgy, hogy a hálózat kapacitásából valamennyit a jelzési folyamatra használunk fel. Ennek a mértékét információ elméleti mértékekkel írtam le.

Az **UBQP** probléma megoldására adtam hatékony és közel optimális eljárat. Az **ICT** alkalmazások hatékony működése gyakran erre a problémára vezethető vissza, úgy mint ütemezési feladatok kommunikációs hálózatokban (**IoT** vagy cloud környezetben), erőforráselosztás és **MUD** a vezeték nélküli kommunikációs technikákban. A problémára párhuzamosan és hatékonyan végrehajtható eljárásokat mutattam be, melyek mind a tisztán előre csatolt (Feed Forward) és a visszacsatolt (Recurrent type neural networks) neurális hálózatokat használják. A feladatot energia optimalizálási problémára fogalmaztam át, valamint az előrecsatolt neurális hálózatok általános taníthatóságát használtam ki, hogy a problémára tanítás után megoldást adjanak.

A mintafelismerési feladat hatékonyan kezelhető az előrecsatolt neurális hálózatok segítségével és egy általam javasolt lineáris kódolás segítségével. Bemutattam, hogy a javasolt algoritmus hatékonyan talál közelítő megoldást a **MUD** problémára. Az eljárás azonban más automatizálási feladatra is alkalmazható, többek között megfigyelő rendszerek, a beszéd felismerés vagy a tartalom alapú keresés problémáinál.

Numerikus példákkal támasztom alá a disszertáció állításait, hogy a javasolt eljárások jó minőségű megoldást adnak és hatékonyan végrehajthatóak.

A disszertáció sok felmerülő kérdésre nem ad választ, azonban kellő részletességgel mutatja be a környezetet, hogy további kutatás alapjává válhasson. A disszertáció ezen felül megfogalmaz egy jól alkalmazható egységes numerikus keretrendszert, amellyel más algoritmusok minősége is összehasonlíthatóvá válik. Egy lehetséges kiterjesztése az itt leírt algoritmusoknak az, hogy a jelenleg használt neurális hálózatok helyére „deep-learning” alapú algoritmusokat választunk megvizsgálva az ebből adódó teljesítmény növekedést vagy veszteséget. Az disszertáció általános célú algoritmusokat mutat be, amelyek további, a cél alkalmazáshoz tartozó módosításokat igényelnek egy konkrét feladat esetén. Továbbá fontos megjegyezni, hogy a disszertáció írásakor még nem léteznek azok a fizikai architektúrák, amelyek lehetővé tennék a beágyazott alkalmazást, de ez a munka iránymutatást nyújthat ebben a kérdésben is.

5. Köszönetnyilvánítás

Elsőként szeretnék köszönetet mondani a témavezetőmnek, *Dr. Levendovszky Jánosnak*, †*prof. Roska Tamásnak*, *Dr. Csurgay Árpádnak*, *Nyékyné dr. Gaizler Juditnak*, *Dr. Szolgay Péternek*, a doktori iskolánk jelenlegi és egykori vezetőinek, *Martin Haenggi-nek* valamint *prof. Kolombán Gézának* a tanácsaikért, bátorításukért és építő kritikájukért, amik mind lendületet adtak az eddigi bejárt utamhoz.

Külön köszönet jár *dr. Oláh Andrásnak*, *Vizi Péternek*, *dr. Németh Kornélnak*, *Pásztor Jánosnak*, *Somogyi Józsefnek*, *Süttő Dánielnek* és *Kauker Dávidnak* a támogatásukért, kitartásukért és segítségükért, ami nélkül kétség kívül a jelen dolgozat nem jött volna létre.

Szítén szeretnék köszönetet mondani egykori doktorandusz társaimnak és a munkatársaimnak az értékes társaságukért és támogatásukért, többek között *Bojársky Andrásnak*, *dr. Treplán Gergelynek*, *dr. Tornai Kálmánnak*, *dr. Józsa Csabának*, *dr. Reguly Istvánnak*, *Koller Miklósnak*, *Juhász Imrének*, valamint *Abonyi Gábornak*, *Porázik Péternek*, *Kosztolánczi Tamásnak*, *Fehér Attilának*, *Tibély Marcellnak*, *Nagy Máténak*, *Varjasi Ferencnek*, *Dvorzsák Alexandernek* és *Györki Milánnak*.

Köszönöm *Vida Tivadarné Katainak* a pontos és figyelmes adminisztratív háttérrel, amellyel megajándékozott a doktori éveim során és azon túl.

Szeretnék még köszönetet mondani a gazdasági osztály és a tanulmányi osztály valamennyi tagjának, a dékáni hivatal és az informatikai csoport minden tagjának a munkájukért, mely olykor talán nem annyira látványos, de kevésbé sem értéktelen.

Szeretném megköszönni az *SP CEE Scholarship* program tagjainak és alapítóinak a barátságát, vezetését és pénzügyi támogatását.

Legmélyebb köszönetem szüleimnek, testvéreimnek és a családom többi tagjának jár: *Tisza Kálmánnak*, †*Tisza Kálmánnénak* szül. *Adorján Katalin*, *Kálmánnak* és *Leventének*, †*Gergelyné Busa Ildikó* egykori tanítóimnak: †*Gutai Jánosnak*, †*Kovaliczky Máriának* és középiskolai tanáraimnak, név szerint *Szabó Kálmánné Katalinnak* és *Szeitzné Viski Erikának*, hogy olyan biztos alapot adtak számomra amire biztosan építhettem az évek során.

References

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. “Wireless sensor networks: a survey”. In: *Computer Networks* 38.4 (2002), pp. 393–422. ISSN: 1389-1286. DOI: [http://dx.doi.org/10.1016/S1389-1286\(01\)00302-4](http://dx.doi.org/10.1016/S1389-1286(01)00302-4). URL: <http://www.sciencedirect.com/science/article/pii/S1389128601003024>.
- [2] F. Barahona. “A solvable case of quadratic 0-1 programming”. In: *Discrete Applied Mathematics* 13.1 (1986), pp. 23–26.
- [3] J.E. Beasley. “Heuristic algorithms for the unconstrained binary quadratic programming problem”. In: *London, UK: Management School, Imperial College* (1998).
- [4] E. Boros, P. Hammer, and X. Sun. “The DDT method for quadratic 0-1 minimization”. In: *RUTCOR Research Center, RRR* 39 (1989), p. 89.
- [5] Shigang Chen and Klara Nahrstedt. “An overview of quality of service routing for next-generation high-speed networks: problems and solutions”. In: *IEEE Network Magazine, Special Issue on Transmission and Distribution of Digital Video* 12.6 (Nov. 1998), pp. 64–79. ISSN: 0890-8044. DOI: [10.1109/65.752646](https://doi.org/10.1109/65.752646).
- [6] Shigang Chen and Klara Nahrstedt. “On finding multi-constrained paths”. In: *Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on*. Vol. 2. IEEE. June 1998, pp. 874–879. DOI: [10.1109/ICC.1998.685137](https://doi.org/10.1109/ICC.1998.685137). URL: <http://citeseer.nj.nec.com/chen98finding.html>.
- [7] M.O. Damen, H. El Gamal, and G. Caire. “On maximum-likelihood detection and the search for the closest lattice point”. In: *Information Theory, IEEE Transactions on* 49.10 (2003), pp. 2389–2402. ISSN: 0018-9448. DOI: [10.1109/TIT.2003.817444](https://doi.org/10.1109/TIT.2003.817444).
- [8] M. Failli. *Digital land mobile radio communications COST 207*. Tech. rep. European Commission, 1989.
- [9] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Vol. 174. New York, NY, USA: W.H.Freeman & Co Ltd, 1979. ISBN: 0716710447.
- [10] R. Gold. “Optimal binary sequences for spread spectrum multiplexing (Corresp.)” In: *Information Theory, IEEE Transactions on* 13.4 (1967), pp. 619–621. ISSN: 0018-9448. DOI: [10.1109/TIT.1967.1054048](https://doi.org/10.1109/TIT.1967.1054048).
- [11] Roche A Guérin and Ariel Orda. “QoS routing in networks with inaccurate information: theory and algorithms”. In: *IEEE/ACM Transactions on Networking (TON)* 7.3 (1999), pp. 350–364.
- [12] Roch Guérin and Ariel Orda. “Computing shortest paths for any number of hops”. In: *Networking, IEEE/ACM Transactions on* 10.5 (2002), pp. 613–620.

ISSN: 1063-6692. DOI: <http://dx.doi.org/10.1109/TNET.2002.803917>.

- [13] Simon Haykin and Michael Moher. *An Introduction to Analog and Digital Communications*. Wiley, 2006. ISBN: 0471432229.
- [14] *ISO/IEC standard 7498-1:1994 - OSI/ISO 7 layer model*. [http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip).
- [15] J.M. Jaffe. “Algorithms for finding paths with multiple constraints”. In: *Networks* 14.1 (1984), pp. 95–116.
- [16] Mirosław Kantor, Piotr Chołda, and Andrzej Jajszczyk. “Least Cost Routing (LCR) solution for inter-domain traffic distribution”. In: *Telecommunication Systems* (2011). 10.1007/s11235-011-9606-1, pp. 1–13. ISSN: 1018-4864. URL: <http://dx.doi.org/10.1007/s11235-011-9606-1>.
- [17] John Klinecicz, James Schmitt, and Richard Wong. “Incorporating QoS into IP Enterprise Network Design”. In: *Telecommunication Systems* 20 (1 2002). 10.1023/A:1015441400785, pp. 81–106. ISSN: 1018-4864. URL: <http://dx.doi.org/10.1023/A:1015441400785>.
- [18] G.A. Kochenberger, F. Glover, B. Alidaee, and C. Rego. “A unified modeling and solution framework for combinatorial optimization problems”. In: *OR Spectrum* 26.2 (2004), pp. 237–250.
- [19] Gary A. Kochenberger, Fred Glover, Bahram Alidaee, and Cesar Rego. “Solving Combinatorial Optimization Problems Via Reformulation and Adaptive Memory Metaheuristics”. In: *Frontiers of Evolutionary Computation*. Ed. by Anil Menon. Vol. 11. Genetic Algorithms and Evolutionary Computation. Springer US, 2004, pp. 103–113. ISBN: 978-1-4020-7524-7. DOI: [10.1007/1-4020-7782-3_5](http://dx.doi.org/10.1007/1-4020-7782-3_5). URL: http://dx.doi.org/10.1007/1-4020-7782-3_5.
- [20] Vachaspathi P. Kompella, Joseph C. Pasquale, and George C. Polyzos. “Multicast routing for multimedia communication”. In: *IEEE/ACM Trans. Netw.* 1.3 (1993), pp. 286–292. ISSN: 1063-6692. DOI: <http://dx.doi.org/10.1109/90.234851>.
- [21] Vachaspathi Peter Kompella. “Multicast routing algorithms for multimedia traffic”. PhD thesis. La Jolla, CA, USA: University of California at San Diego, 1993.
- [22] E.G. Larsson. “MIMO Detection Methods: How They Work [Lecture Notes]”. In: *Signal Processing Magazine, IEEE* 26.3 (2009), pp. 91–95. ISSN: 1053-5888. DOI: [10.1109/MSP.2009.932126](http://dx.doi.org/10.1109/MSP.2009.932126).
- [23] Whay Chiou Lee. “Spanning tree method for link state aggregation in large communication networks”. In: *INFOCOM’95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings. IEEE*. IEEE. Washington, DC, USA: IEEE Computer Society, 1995, pp. 297–302. ISBN: 0-8186-6990-X.

- [24] János Leventovszky, Alpár Fancsali, Csaba Végso, and Gábor Rétvári. “QoS Routing with Incomplete Information by Analog Computing Algorithms”. In: *Quality of Future Internet Services: Second COST 263 International Workshop, QofIS 2001 Coimbra, Portugal, September 24–26, 2001 Proceedings*. Ed. by Mikhail I. Smirnov, Jon Crowcroft, James Roberts, and Fernando Boavida. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 127–137. ISBN: 978-3-540-45412-0. DOI: [10.1007/3-540-45412-8_10](https://doi.org/10.1007/3-540-45412-8_10). URL: http://dx.doi.org/10.1007/3-540-45412-8_10.
- [25] Dean H. Lorenz and Ariel Orda. “QoS routing in networks with uncertain parameters”. In: *IEEE/ACM Transactions on Networking* 6.6 (Dec. 1998), pp. 768–778. ISSN: 1063-6692. DOI: [10.1109/90.748088](https://doi.org/10.1109/90.748088). URL: citeseer.ist.psu.edu/lorenz98qos.html.
- [26] Z. Luo, W. Ma, A.M.C. So, Y. Ye, and S. Zhang. “Semidefinite relaxation of quadratic optimization problems”. In: *Signal Processing Magazine, IEEE* 27.3 (2010), pp. 20–34.
- [27] Nguyen Cong Luong, Dinh Thai Hoang, Ping Wang, Dusit Niyato, Dong In Kim, and Zhu Han. “Data collection and wireless communication in Internet of Things (IoT) using economic analysis and pricing models: A survey”. In: *IEEE Communications Surveys & Tutorials* 18.4 (2016), pp. 2546–2590.
- [28] Haci Mantar. “A scalable QoS routing model for diffserv over MPLS networks”. In: *Telecommunication Systems* 34 (3 2007). 10.1007/s11235-007-9035-3, pp. 107–115. ISSN: 1018-4864. URL: <http://dx.doi.org/10.1007/s11235-007-9035-3>.
- [29] Stéphane Martignoni and Thomas Kühnel. “Extension of Classical IP over ATM to support QoS at the application level”. In: *Telecommunication Systems* 11 (3 1999). 10.1023/A:1019161721356, pp. 291–303. ISSN: 1018-4864. URL: <http://dx.doi.org/10.1023/A:1019161721356>.
- [30] *MeasurementLab homepage*. <https://www.measurementlab.net>. Accessed: 2019.05.14.
- [31] Peter Merz and Kengo Katayama. “Memetic algorithms for the unconstrained binary quadratic programming problem”. In: *Biosystems* 78.1–3 (2004), pp. 99–118. ISSN: 0303-2647. DOI: [10.1016/j.biosystems.2004.08.002](https://doi.org/10.1016/j.biosystems.2004.08.002). URL: <http://www.sciencedirect.com/science/article/pii/S0303264704001376>.
- [32] J.C. Picard and H.D. Ratliff. “A graph-theoretic equivalence for integer programs”. In: *Operations Research* (1973), pp. 261–269.
- [33] J.C. Picard and H.D. Ratliff. “Minimum cuts and related problems”. In: *Networks* 5.4 (1975), pp. 357–370.
- [34] C. Pornavalai, G. Chakraborty, and N. Shiratori. “A neural network approach to multicast routing in real-time communication networks”. In: *ICNP’95*:

- Proceedings of the 1995 International Conference on Network Protocols*. IEEE Computer Society, 1995, p. 332. ISBN: 0-8186-7216-1.
- [35] Chotipat Pornavalai, Goutam Chakraborty, and Norio Shiratori. “Routing with multiple QoS requirements for supporting multimedia applications”. In: *Telecommunication Systems* 9 (3 1998). 10.1023/A:1019160226383, pp. 357–373. ISSN: 1018-4864. URL: <http://dx.doi.org/10.1023/A:1019160226383>.
- [36] John Proakis and Masoud Salehi. *Digital Communications, 5th Edition*. McGraw-Hill Science/Engineering/Math, 2007. ISBN: 0072957166.
- [37] Bhaskar Prasad Rimal, Eunmi Choi, and Ian Lumb. “A taxonomy and survey of cloud computing systems”. In: *INC, IMS and IDC, 2009. NCM’09. Fifth International Joint Conference on*. Ieee. 2009, pp. 44–51.
- [38] Maria Alejandra Rodriguez and Rajkumar Buyya. “A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments”. In: *Concurrency and Computation: Practice and Experience* (2016).
- [39] A. Shaikh, J. Rexford, and K. Shin. “Efficient precomputation of quality-of-service routes”. In: *Proc. Workshop on Network and Operating Systems Support for Digital Audio and Video*. 1998, pp. 15–27. URL: citeseer.ist.psu.edu/article/shaikh98efficient.html.
- [40] Anees Shaikh, Jennifer Rexford, and Kang G Shin. “Dynamics of quality-of-service routing with inaccurate link-state information”. In: *Ann Arbor 1001.CSE-TR-350-97* (1997), pp. 48109–2122. URL: citeseer.ist.psu.edu/shaikh97dynamics.html.
- [41] Sukhpal Singh and Inderveer Chana. “A survey on resource scheduling in cloud computing: Issues and challenges”. In: *Journal of Grid Computing* 14.2 (2016), pp. 217–264.
- [42] Tim Szigeti and Christina Hattingh. *End-to-end qos network design*. Cisco press, 2005.
- [43] **Dávid Tisza**, András Oláh, and János Levendovszky. “Multi-user detection using non-parametric Bayesian estimation by feed forward neural networks”. In: *Telecommunication Systems* (2015), pp. 1–11.
- [44] **Dávid Tisza**, András Oláh, and János Levendovszky. “Novel algorithms for quadratic programming by using hypergraph representations”. In: *Wireless personal communications* 77.3 (2014), pp. 2305–2339.
- [45] **Dávid Tisza**, Péter Vizi, Janos Levendovszky, and András Oláh. “Multicast Routing in Wireless Sensor Networks with Incomplete Information”. In: *Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless), 11th European*. 2011, pp. 1–5.
- [46] Sergio Verdu. *Multiuser Detection*. Cambridge University Press, 1998. ISBN: 0521593735.

- [47] J. Wang. “Discrete Hopfield network combined with estimation of distribution for unconstrained binary quadratic programming problem”. In: *Expert Systems with Applications* 37.8 (2010), pp. 5758–5774.
- [48] Minxian Xu, Wenhong Tian, and Rajkumar Buyya. “A Survey on Load Balancing Algorithms for VM Placement in Cloud Computing”. In: *arXiv preprint arXiv:1607.06269* (2016).
- [49] Changsheng Yu, Li Yu, Yuan Wu, Yanfei He, and Qun Lu. “Uplink Scheduling and Link Adaptation for Narrowband Internet of Things Systems”. In: *IEEE Access* 5 (2017), pp. 1724–1734.