Egyetemi doktori (PhD) értekezés tézisei

Médiaminőség-vizsgálat hálózati forgalom veszteségmentes analízisével

Monitoring media quality based on lossless traffic evaluation

Skopkó Tamás

Témavezetők:

Orosz Péter, Ph.D.

Almási Béla, Ph.D. †

Debreceni Egyetem

Természettudományi Doktori Tanács

Informatikai Tudományok Doktori Iskola

Debrecen, 2016.

Tartalomjegyzék – Contents

1 A doktori értekezés előzményei és célkitűzései

Az internetet, mint általános célú adattovábbító infrastruktúrát, egyre több és több alkalmazás használja. Az alkalmazások felhasználói ma nem pusztán az elégséges működést, hanem a minél jobb felhasználói élményt (pl. folyamatos működés, gyors válaszidő) is elvárják. Ez az igény még inkább jelen van a fizetős szolgáltatások esetében. Ezért az ipari szereplők, mind a hálózati infrastruktúra üzemeltetői, mind az alkalmazások szolgáltatói törekednek a magas szintű felhasználói élmény biztosítására. Az egyre összetettebb hálózatok aktuális és jövőbeli igényeknek is megfelelő folyamatos hangolásához és fejlesztéséhez a szolgáltatóknak szükségük van olyan eszközökre, amelyek mérhető paraméterek segítségével naprakész áttekintő képet adnak az infrastruktúra és a szolgáltatás állapotáról. A mutatók folyamatos követésével pedig hosszútávú trendek vázolhatók fel, amelyek a fejlesztési stratégiát is meghatározhatják.

A Quality of Service (QoS, szolgáltatási minőség) a publikus internet elterjedésével párhuzamosan fejlődő alapvető eszközrendszer, amely a hálózati teljesítménymutatók mérésére és meghatározására szolgál. Olyan metrikák gyűjteménye, amelyet a korszerű hálózati infrastrukturális aktív eszközök képesek mérni. A QoS mutatók közül a legfontosabbak az útvonali késleltetés (latency), érkezési ingadozás (jitter), csomagvesztés (packet loss), átviteli ráta (throughput), csomagátrendeződés (reordering) és duplikáció (duplication). Az internetszolgáltató (Internet Service Provider, ISP) saját eszközeit monitorozva a QoS metrikákat rendszeresen lekérdezi és tárolja. A passzív megfigyelésen túl a metrikákhoz határértékeket állapíthat meg, amelyeket meghaladva lépéseket tehet a hálózat erőforrásainak átcsoportosítására. Egy alkalmazás szolgáltatója szintén képes

QoS mutatókat mérni, de jellemzően csak azokon a végpontokon, ahol a saját alkalmazásai is futnak.

A QoS eszközrendszer már a kezdetek óta a halózatokkal kapcsolatos kutatások egyik fő fókuszpontja. A témakörben született publikációk rámutattak, hogy a szolgáltató hálózatában pusztán a metrikák értékeinek javulása nem feltétlenül eredményezi, hogy az alkalmazások minősége is érezhetően javulni fog. Sokáig a hálózati infrastruktúra üzemeltetők az akkut csomagvesztésre a hálózati sávszélesség növelésével reagáltak, ám a beruházástól remélt minőségi javulás sok esetben elmaradt.

A felhasználó nézőpontját feltárni próbáló vizsgálati módszereket Quality of Experience (QoE, érzeti minőség) fogalom alatt tartjuk számon. Ez a szintén régóta aktív kutatási terület kezdetben elsősorban szubjektív vizsgálati módszereket jelentett. Ezek a módszerek a felhasználó megkérdezésével és a leadott értékelések statisztikai összesítésével adtak képet, rendszerint egy adott szolgáltatás érzeti minőségéről. Mivel a szubjektív értékelés elvégzése meglehetősségen körülményes és költséges, ráadásul az azonnali visszacsatolás nagyon nehezen oldható meg, a QoS–QoE összekapcsolását megkísérlő kutatási területek létező és fontos szolgáltatói igényeket is kielégíthetnek. Ezek a módszerek a QoS mérhető paramétereit próbálják megfeleltetni a QoE érzeti minőségi értékeinek.

Kutatásaim célkitűzése szolgáltatások és médiaátvitel érzeti minőségét becslő módszerek, ill. ezek alapjául szolgáló metrikák létrehozása. A módszerek kidolgozásához adatgyűjtés és elemzés szükséges. Az adatgyűjtést a hálózati forgalom monitorozásával és a forgalom elkapásával végzem. A veszteségmentes forgalom-megfigyelés és -gyűjtés munkám során mindvégig fontos szempontként szerepelt. A begyűjtött forgalmi mintázatok hatékony analíziséhez ugyanis lényeges a megfelelő felbontású és részletességű adatsorok megléte. Eleinte az elérhető céleszközök híján olyan módszerekre volt szükség, amelyekkel a hozzáférhető,

jellemzően általános célú architektúrákon is végezhető, lehetőleg veszteségmentes hálózati forgalmi mérés. A mérések egyik fontos eleme az időbélyegzés, amely a forgalmi mintázat rekonstrukciója során fontos szerepet játszik. Az adott környezet és sávszélesség mellett kielégítő pontosságú időbélyegzés megvalósítása kihívást jelentett. Mindemellett a végponti időbélyegzés többlet erőforrásigénye negatívan befolyásolta az elemezni kívánt rendszer működését is.

1.1 Optimalizált szoftveres időbélyegzés

Szoftveres időbélyegzés során az időbélyeg a rendszeridő lekérdezése után kerül a csomagokra. Egy rendszerben az időt szolgáltató alrendszerek általában egyszerű órajel számlálók, amelyek önmagukban nem szolgáltatnak pontos időt. Ezeket az órajel forrásokat a frekvenciájuk ismeretében időméréshez is tudjuk használni. A pontos idő követéséhez szükség van egy referencia időforráshoz történő (egyszeri, rendszeres vagy folyamatos) szinkronizálásra is. Így a tényleges időbélyegzés két fő részműveletre bontható: i.) az órajelforrás lekérdezése (clock read) és ii.) a kapott nyers delta ciklusszám UTC időre történő alakítása (clock conversion). Az óraforrás jellemzően valamilyen hardver komponens, amelyhez a hozzáférés a megfelelő illesztőfelületen, meghajtóprogramon, esetleg processzor regiszteren keresztül történik. Minél rövidebb az óraforrás hozzáférési ideje és ez a hozzáférési idő minél kisebb szórást mutat, annál alkalmasabb a pontos időbélyegzéshez való felhasználásra. A ciklusszámláló frekvenciája pedig meghatározza az időmérési pontosságot, ezáltal az időbélyegét is.

Az időbélyegzés tekintetében fontos metrika annak felbontása (Timestamping Resolution). Minden helyi órajelforrást jellemez annak felbontása, vagyis az időforrásban nyilvántartott számláló információ (pl. regiszter) ábrázolási pontossága, finomsága (granularity). A helyi óraforrás és a kiolvasott érték tárolására

szolgáló adatszerkezet ábrázolási felbontásának együttese határozza meg az előállított időbélyeg felbontását. A felbontás jellemzi, milyen finomságban tudjuk előállítani az időbélyeget. Az időbélyeg felbontása függ az óraforrás felbontásától, az ahhoz való hozzáférési időtől, a kapott értéket tároló adatszerkezet kapacitásától és az átalakítás pontosságától.

Amennyiben egy 1 Gbit/s kapacitású Ethernet kapcsolat forgalmát szeretnénk elemezni, úgy a legkisebb bruttó 72 byte méretű csomagokból akár 1 488 096 is érkezhet másodpercenként. Ez azt jelenti, hogy 672 ns-onként érkezhet egy új csomag. A forgalmi mérést szerver oldalon végezve még kisebb érkezési időközökkel számolhatunk a jelenleg elterjedt 10 Gbit/s sávszélességű kapcsolatokon. Vagyis a teljeskörű vizsgálathoz nem elegendő a 10^{-6} másodperces skála, 10^{-9} másodperces felbontásra volna szükség, továbbá olyan adatszerkezetre, amely képes a 10^{-9} másodperc felbontású idő ábrázolására (pl. 32+32 bit) [10].

Az időbélyegzési precizitás (Timestamping Precision) metrikája a mérési eredmény szórásának feleltethető meg. Ezt az óraforrásból származtatott idő egy referenciának tekintett időforrással való együttfutása határozza meg. Az óraforrás kiolvasásának költsége is mutathat szórást, mivel az ütemező és a megszakításkezelés is hatással van rá. Az időbélyegzés precizitása így az óraforrás precizitása és a hozzáférés költségének szórása.

A harmadik időbélyegzési metrika az időbélyeg pontossága (Timestamping Accuracy, offset), amely az előállított abszolút időbélyeg helyi referencia időforráshoz képesti eltérését jellemzi. Minél kisebb ez az eltérés, annál inkább a megjelölni kívánt esemény bekövetkezésének valós idejéhez közeli időbélyegeket állítunk elő. A referencia óraforrás jellemzően egy, a rendszeren kívüli óra. Felbontása praktikusan legalább a helyi óraforrásával megegyező.

A metrikák ismeretében az időbélyegek akkor állnak legközelebb az ábrázolni kívánt tényleges időpontokhoz, ha minél kisebb az időbélyegzési folyamathoz szükséges idő. A hatékony időbélyegzés kellően nagy felbontás mellett az órajelforrást alacsony elérési idővel és az elérési idő minimális szórásával kérdezi le, valamint alakítja a lekérdezett nyers időértéket rendszeridővé.

Végrehajtási kontextusok: Az operációs rendszerek rendszerfolyamatai megkülönböztetett környezetben (kernel kontextusban), privilegizálva futnak a hatékony működés érdekében. Csomag érkezésekor megszakítás generálódik, a végrehajtás egy kitüntetett módba, megszakítási kontextusba vált. A rendszer ebben az állapotban van addig, amíg a kapcsolódó műveleteket (pl. a csomagok regisztrálása) el nem végezte. Megszakítási kontextusban az adott processzormagon minden más folyamat végrehajtása szünetel. Hatékonysági okokból célszerű, ha minél rövidebb ideig tartózkodik a rendszer ebben a kizárólagos állapotban. Felhasználói végrehajtási módban a kernelen kívüli, felhasználói folyamatok futnak, amelyek ütemezéséért a rendszermag ütemezője a felelős. Fontos tényező, hogy a kernel kontextusai és a felhasználói végrehajtási mód közötti adatmozgatás extra időbeli költséggel jár.

Csomagfeldolgozási kapacitás: Minden rendszer rendelkezik fizikai (pl. számítási kapacitás) korlátokkal. Az egyre fejlődő hálózati technológiákkal párhuzamosan egyre nagyobb linksebességek állnak rendelkezésre. A megnövekedett sávszélesség természetesen csak a rendelkezésre álló számítási teljesítmény mellett szolgálható és aknázható ki. Általános célú architektúrákban ez a számítási teljesítmény megoszlik a rendszermag és az alkalmazások között. A rendszert számítási kapacitásának határán használva az óraforrás elérési idejének szórása megnövekszik, ezért az időbélyegzés pontossága romolhat, valamint az általa felemésztett erőforrás ronthatja az

alkalmazás, illetve az igénybe vett szolgáltatás minőségét, a QoE-t. Amint a csomagok beérkezési intenzitása meghaladja a feldolgozó kapacitás határát, átmenetileg a pufferek segítenek áthidalni a szűk keresztmetszetet a feldolgozás késleltetésével. A pufferek betelése után érkező csomagok viszont elvesznek. Ily módon a csomagfeldolgozás egy forgalomalakítóhoz (traffic shaper) hasonlóan működik, amelyben a limitáló tényező az adott rendszerre jellemző folyamatos csomagfeldolgozó kapacitás. Az ezt meghaladó beérkezési intenzitás mellett a forgalom időben késleltetést szenved, a pufferkapacitást meghaladva a működés csomagvesztésbe torkollik [3].

Bár általános célú operációs rendszerekben a csomagfeldolgozásban egy hálózati csatolóhoz több processzormag nem rendelhető, csomagfeldolgozás közben a további processzormagok egyéb feladatokat viszont futtathatnak. A hatékonyság növelése érdekében ezek a processzormagok bevonhatók bizonyos csomagfeldolgozási részműveletek elvégzésébe is.

A mobil eszközök feldolgozási kapacitása erősen limitált. A munkaállomásoknál jóval kisebb teljesítményű központi feldolgozó egységnek (Central Processing Unit, CPU, processzor) számos folyamatot kell futtatnia. Ugyanakkor a több processzormagos technológia már ezekben a készülékekben is elérhetővé vált. Az időbélyegzés folyamatát érdemes a több processzormagos környezet adta lehetőségekkel kihasználni oly módon, hogy az időbélyegzés fázisai közül azt, amely nem szerves része a helyi óraforrás kiolvasásának, áthelyezhetjük a végrehajtás egy későbbi, jól párhuzamosítható szakaszába.

Az általam kidolgozott módszer a helyi óraforrásból származó érték abszolút idejű ábrázolásra történő konverzióját áthelyezi egy utófeldolgozási szálba, amelyet több processzormagos rendszerben alacsony hozzáférési idejű (pl. regiszter alapú) helyi óraforrással kombinálva hatékonyan csökkenthető a

megszakítási kontextusban töltött idő [12]. Emellett a konverziós művelet leginkább tétlen processzormagokon történő végrehajtásával növekszik a csomagfeldolgozó kapacitás és csökken a csomagvesztés lehetősége.

1.2 Rate Control Transport Protocol

K+F projekt keretében 100 Gbit/s sávszélességű szolgáltatói (mag-)hálózatok aggregált forgalmának monitorozására képes eszköz megalkotásában vettem részt. Az eszköz egyik fő funkciója a forgalom hardverrel gyorsított veszteségmentes monitorozása. A megcsapolt forgalmat szűrve és folyamokra bontva, azokat 1 vagy 10 Gbit/s áteresztőképességű kapcsolatokon adatgyűjtő- és feldolgozó szerverek felé továbbítja. Mivel a szerverek a forgalom további, mélyebb szintű elemzésével, statisztika készítéssel is foglalkoznak, feldolgozási oldalon a hardveres gyorsítás nem választható. Az általános célú operációs rendszerrel működő gépek, illetve az azokon futó adatgyűjtő, feldolgozó alkalmazások felé az előszűrt byte-folyamokat veszteségmentesen kellett továbbítani.

A monitoring eszközt FPGA alapú hardver segítségével valósítottuk meg. Számos korláttal kellett számolni a tervezés és fejlesztés során. Ezek egyike volt, hogy minél több funkciót kellett az FPGA chipben elhelyezni, annál több logikai kaput kellett „feláldozni”. Bár egyre nagyobb kapacitású és teljesítményű FPGA IC-k érhetők el, a rendelkezésre álló kapacitás mindig véges és az áruk az integrált logikai elemek számával hozzávetőleg exponenciálisan emelkedik. Ráadásul a komplexitás növekedésével egyre nehezebb a megtervezett áramkör implementálásakor az optimális órajel-frekvencia megtartása. Ezért fontos, hogy az alkalmazott transzport protokoll erőforrás-takarékos és minél egyszerűbb legyen. Az említett projekt kezdetén felvetődött a Transmission Control Protocol (TCP) alkalmazása. Mivel hálózati interfészenként példányosítani kellett

a transzport protokollt, annak sokszoros erőforrásigénye miatt alternatív protokoll alkalmazásának és kifejlesztésének igénye vetődött fel.

Számos transzport protokollt megvizsgáltam, alkalmasak-e a cél környezetben történő felhasználásra. A legtöbbjük a TCP-hez hasonlóan komoly pufferelési igénnyel rendelkeznek. Másoknál a hatékonyság, túlzott összetettség vagy egyéb okok, – pl. a veszteségmentes továbbítás hiánya – voltak a kizáró tényezők.

A mérőrendszerhez olyan transzport protokoll létrehozása volt a célom, amely minimális erőforrás felhasználás mellett képes a veszteségmentes továbbításra. A mérőeszköz több (akár 10 darab) fizikai hálózati csatolón keresztül továbbítja a mérési adatokat. Az architektúra adottságaiból fakadóan ezeket a fizikai kapcsolatokat számos egyéb funkció mellett független példányokként kell implementálni. Ezért az erőforrás-takarékosság kiemelkedő szempont volt a tervezéskor. Ez a protokoll akár általánosabb célokra is alkalmazható lehet, pl. beágyazott és egyéb alacsony erőforrásokkal bíró rendszerekben.

A lehetséges torlódás és csomagvesztés egyaránt bekövetkezhet a továbbítást végző infrastruktúrán és a végponton is. Vezeték nélküli kapcsolatoknál a továbbító közeg kiszámíthatatlan: zaj, interferencia bármikor felléphet, az emiatt előforduló csomagvesztés a fizikai rétegben bekövetkezett átviteli hiba miatt történik. Vezetékes hálózatokban fellépő vesztés a hálózati forgalom torlódásából, valamely eszköz pufferének telítődésétől következik be. Dedikált célra létesített hálózatok (pl. adatközpontokban, mérőrendszerekben, blokkolásmentes adatátviteli kapcsolókban (non-blocking switch) megfelelő tervezés mellett ezektől mentesek lehetnek. Ilyen környezetben csak a végponton következhet be csomagvesztés.

Egy általános célú operációs rendszerekkel rendelkező architektúrákban szintén pufferekeresztül vezet a csomagok útja a fogadó alkalmazásig. Az első puffer a hálózati csatoló saját

puffere, amelybe a fizikai rétegen (Physical Layer, PHY) keresztül érkezik a csomag. Ez a puffer rendszerint minimális méretű, de jól tervezett meghajtóprogrammal alkalmas arra, hogy a következő csomag érkezéséig áthidaljon, amíg az aktuális csomag bemásolódik a rendszermemóriába (Random Access Memory, RAM). A további pufferek már az operációs rendszer és a csomagfeldolgozó alrendszer részeit képezik. A legtöbb esetben ezen pufferek méretei hangolhatók, ami biztosítja, hogy impulzusszerűen érkező forgalom esetén is elég kapacitást lehessen allokálni az átmeneti tároláshoz. A beérkező csomagokat az operációs rendszernek fel kell dolgoznia: a csomag az őt érintő protokollok fejléceinek ellenőrzésén megy át, amely egyfelől segít kiszűrni a hibás vagy a rendszer és alkalmazásai számára érdektelen csomagokat, másfelől a megfelelő alrendszer felé irányítható tovább az információ. Ezek a feladatok a csomagok beérkezési intenzitásának növekedésével egyre nagyobb terhet rónak a rendszerre. Amennyiben a beérkező csomagok feldolgozására nem áll rendelkezésre elegendő számítási kapacitás, a pufferek telítődése után érkező csomagok elvesznek. A végpontokon a pufferkapacitások mellett tehát fontos mutató a csomagfeldolgozó alrendszer teljesítménye is.

1.3 Opus hangkódoló alapú VoIP minőségbecslés

Az internet elterjedésével párhuzamosan megjelentek azok a szolgáltatások, amelyek publikus, illetve heterogén hálózatokon biztosítanak IP-feletti (Voice over IP, VoIP) hangtovábbítást. A nagyobb elérhető számítási kapacitás és sávszélesség nyitotta új lehetőségek az általános architektúrák szoftverkörnyezetéből fakadó rugalmasságának köszönhetően gyorsan fejlődésnek indultak a hangkódolók is, amelyek adott sávszélesség mellett sokkal jobb minőségre képesek, mint a távközlési szolgáltatók által támogatott hagyományos hangkódolók. Az Opus kódoló,

köszönhetően a fontos ipari szereplők támogatásának, rövid időn belül Internet Engineering Task Force (IETF) szabvánnyá vált.

Kutatásomat motiválta, hogy az Opus kódolót VoIP környezetben nem vizsgálták korábbi publikációkban. Léteznek olyan minőségbecslő módszerek, amelyek a forrásanyag ismeretében nagy biztonsággal képesek a dekódolt anyag érzeti minőségének becslésére. Ezeket referencia alapú (Full Reference, FR) módszereknek hívjuk. A gyakorlatban rendszerint az eredeti hanganyag nem hozzáférhető. A forrásanyag rögzítése számos kérdést felvet. Hogyan oldható meg a tárolás, a rögzített és a dekódolt anyag szolgáltatóhoz történő megbízható és transzparens eljuttatása? A rögzített anyag felhasználása továbbá jogi aggályokat is felvet. Ezért a referencia alapú módszerek a szolgáltatók számára csak ritkán minősülnek gyakorlatban alkalmazható eszköznek. Ha az eredeti anyag teljes egészében nem, de annak bizonyos jellemzői rendelkezésre állnak, csökkentett referenciájú (Reduced Reference, RR) módszereket alkalmazhatunk a becsléshez. A gyakorlatban ez a VoIP forgalom egy részének, pl. a csomag fejlécek elkapásával történhet. Ez az út még mindig elég távol esik a szolgáltatói felhasználás lehetőségeitől. Kutatásaim ezért a referencia nélküli (No Reference, NR) módszerek felé irányultak, amelyek mérhető metrikák alapján segítik az érzeti minőség becslését.

Számos kutatás irányult a QoS metrikák és a QoE közötti összefüggések keresésére. Önállóan egy-egy metrika (csomagvesztés, késleltetési ingadozás) jellemzően nem mutatott kellő korrelációt az érzeti minőséggel, ezért az újabb kutatások már inkább a metrikák kombinációival foglalkoznak.

2 Az értekezés új tudományos eredményei

I. téziscsoport: Szoftveres időbélyegzés tehermentesítéssel. Fontos tényező, hogy az időbélyeg tényleges felhasználása a csomagban hordozott hasznos teher feldolgozásánál később történik meg. Ez idő alatt a már lekérdezett óraforrásból származó érték konverziója a felhasználásig késleltetve is elvégezhető. A késleltetett vagy nem kritikusan leterhelt környezetben történő végrehajtást a szakirodalom tehermentesítő (offloading) technikának hívja. Az általam bemutatott módszerben – e technikára alapozva – a csomagfeldolgozás kritikus szakaszában, a megszakítási kontextusban csak az óraforrás lekérdezése és a kapott érték eltárolása történik meg. A konverzió tehermentesített esetben felhasználói végrehajtási módban zajlik, amely a csomagfeldolgozási képesség javulását eredményezi az időbélyegzés használata mellett.

I.1. tézis: *Új szoftveres időbélyegzési módszert dolgoztam ki, amely az óraforrást valós idejű késleltetéssel (megszakítási kontextusban) olvassa ki, míg a konverziós lépéseket áthelyezi késleltetéstűrő felhasználói végrehajtási módba [4].*

I.2. tézis: *Kimutattam, hogy az időbélyegzés konverziós fázisának késleltetéstűrő végrehajtási módba történő áthelyezésével jelentős csomagfeldolgozási teljesítménynövekedés érhető el, amely a csomagfeldolgozást érintő erőforrások szaturációjakor a csomagvesztés arányát csökkenti [3].*

I.3. tézis: *Az időbélyegzés óraforrás kiolvasó elemi műveletének megszakítási kontextusban tartása és a konverzió célirányos késleltetése az időbélyegzési precizitást növeli.*

A téziscsoport igazolása kontrollált laboratóriumi környezetben végrehajtott mérésekkel történt. A tehermentesített időbélyegző módszert Linux környezetben implementáltam.

II. téziscsoport: A Rate Control Transport Protocol (RTCP) egy zárt hurkú szabályzóra épül, amelynek feladata a küldési intenzitás hangolása a fogadóoldal teljesítménymutatói alapján. A szabályzófüggvény bemenő paraméterei időről időre mintavételezéssel állnak elő: jelen esetben az érintett pufferek telítettsége és a csomagfeldolgozást végző processzormag kihasználtsága. A protokoll ezek alapján egy, az adott paraméterekre jellemző maximális küldési rátát számol. Előnye az alacsony összetettség, hiszen küldő oldali pufferre nincs szükség (így akár közvetlen hardveres implementáció is lehetséges), valamint a gyors visszacsatolás. Ugyanakkor elkerüli a folyamatos nyugtázást, amit eseményvezérléssel helyettesít.

Az esetleges csomagvesztés elkerülhető a kritikus erőforrások folyamatos monitorozásával, valamint a csomagfolyam útjába eső pufferek megfelelő hangolásával, a mintavételezési periódusok és a visszacsatolási késleltetés áthidalásával. A mintavételezés állandó periódusidővel történik. Mintavételkor az algoritmus a pufferek telítettségét és a processzormagok kihasználtságát kérdezi le. Ezek a paraméterek a vezérlőfüggvény bemenetét fogják képezni.

A protokoll küldő oldala kezdő lépcsőfokként a küldendő byte-folyam csomagokba szervezését végzi. A folyamot a hatékonyság érdekében maximális átviteli egység (Maxmimum Transmission Unit, MTU) méretű csomagokba célszerű szervezni. Ha az alacsony késleltetés is szempont, egy várakozási küszöbérték segítségével biztosítható, hogy minden csomag véges határidővel feldolgozásra kerüljön. A csomagok kiküldése az aktuálisan érvényben lévő küldési ráta szerint történik. Ez a mindenkori ráta

egy rátaszabályzó (rate control) csomag érkezésekor bírálódik felül. A csomagok felépítésére és tárolására, illetve a byte-folyam kívánt küldési rátához szabására egy kisebb méretű egyszerű várakozási sor (First In First Out, FIFO) is elegendő.

A vételi oldal szerepe az erőforrás monitorozás és visszacsatolás. Ez egy időkritikus folyamat, ezért biztosítani kell, hogy a végrehajtás pontosan kerüljön a folyamatra. A mintavételezés során a kiolvasott metrikák (pufferállapot és csomagfeldolgozó processzormag terheltsége) a vezérlőfüggvényük segítségével elállítja az aktuális körülmények között teljesíthető maximális csomagrátát. Az effektív kívánt csomagráta ezek minimuma lesz. Amennyiben a számított érték különbözik az előző ciklusban meghatározottól, azt egy szabályzó csomagban küldjük át. Ha a kiszámolt ráta a korlátozás feloldását követeli meg, akkor a maximális csomagrátát küldjük ki a vezérlőcsomagban. A maximális csomagrátát induláskor a linksebességből származtatjuk. A protokoll a rendelkezésre álló erőforrások változása miatti küldési ráta változtatásakor egy konstanspár segítségével biztosítja az aszimmetrikus működést az oszcilláció elkerüléséhez.

II.1. tézis: *Kidolgoztam az RCTP torlódás megelőző transzport protokollt, amely a TCP-hez viszonyítva alacsonyabb erőforrásigény mellett képes – akár vonali rátán is – elkerülni a csomagvesztést [1].*

II.2. tézis: *A szabályzófüggvények vizsgálatával kimutattam, hogy az 5 és 30% között lineáris karakterisztikájú függvény biztosítja — 0,2-es büntető, 0,1-es jutalmazó konstansokkal alkalmazva — a leghatékonyabb szabályzást az RCTP protokoll számára. Ezen túl rámutattam, hogy a rendszerparaméterek megfelelő megválasztásával a vonali ráta megközelítéséhez nem szükségszerű a mintavételezési periódusidő csökkentése, bár a*

maximális átviteli teljesítmény gyorsabban elérhető a rövidebb mintavételezési periódusidővel.

III. téziscsoport: Az Opus (IETF RFC 6716) hangkódoló VoIP környezetben zajlott beszédhang-átviteli képességeinek vizsgálata során olyan QoS-QoE összefüggéseket tapasztaltam, amelyre referencia nélküli minőségbecslő módszer építhető. Az eljárás első fázisában elvégeztük a különböző forrásanyagok hálózati hibák emulálása melletti minőségelemzését. A szubjektív értékelések begyűjtése után polinomiális regresszió segítségével kerül megállapításra az első fázisban alkalmazott QoS paraméterek és a kapott QoE értékelések közötti kapcsolat fokszámának megállapítása. Korreláció analízis segítségével állnak elő az adott fokszám melletti polinomokban alkalmazandó együtthatók.

Az első fázis eredményei (a csomagvesztés és érkezési ingadozás kapcsolata a szubjektív minőségérzet átlagos élménypont skálán (Mean Opinion Score, MOS) történő kifejezésével) egy 3-tengelyú grafikonon ábrázolhatók. Polinomiális regresszió segítségével meghatározható az illeszkedő felületet leíró kétváltozós polinom.

Az utolsó mozzanat elsődleges célja annak vizsgálata, hogy általános érvényű-e a második fázisban meghatározott polinomok és együtthatók alkalmazásakor várható korreláció. Ez további hanganyagok esetében elvégzett értékelésekkel történik, miközben az első fázisban ismertetett mérési környezet változatlan marad. Az eljárás végső célja egy lehetőleg alacsony fokszámú becslőfüggvény készítése volt, amely bemenő paraméterként objektív mért értékeket kapva a MOS skálán becsli meg a QoE értékét.

III.1. tézis: *Az Opus hangkódolóra vonatkozóan VoIP környezetben végzett mérések analitikus kiértékelésekor QoS-QoE korreláció*

tekintetében a következőket állapítottam meg. A csomagérkezési ingadozás esetében a QoS-QoE kapcsolat lineáris, a csomagvesztés esetében pedig másodfokú összefüggés van. A kombinált QoS paraméterek esetén a másodfokú összefüggés szintén jól leírja a korrelációt [7].

III.2. tézis: *A III.1. tézis összefüggései alapján olyan VoIP QoE-becslő módszert hoztam létre, amely megbízhatóan jelzi előre az Opus-alapú VoIP alkalmazás minőségét a csomagvesztés és érkezési ingadozás QoS metrikákból származtatva. A módszer kis számításigényű, mivel alacsony fokszámú polinomokon alapul [2].*

2.1 Az eredmények alkalmazása

Az I. téziscsoport kapcsán bemutatott tehermentesített időbélyegzési módszer segítségével az általános célú architektúrákon a szoftveres időbélyegzési pontosság növelhető. Továbbá a több processzormaggal rendelkező környezetben az időbélyegzési művelet konverziós lépéseinek a csomagfeldolgozás kritikus időszakából későbbi végrehajtási fázisba áthelyezése növeli az adott csomópont csomagfeldolgozási kapacitását. Ezáltal az erőforrásokban szűkösebb, jellemzően mobil végponti eszközökön hatékonyabb QoS mérés valósítható meg, segítve ezzel a hálózati infrastruktúra vagy az alkalmazások fejlesztőinek, üzemeltetőinek munkáját.

A tehermentesített időbélyegzési technika alkalmazásával elérhetővé váló erőforrások lehetővé teszik, hogy a csomagfeldolgozási útvonalon több ponton is helyezhessünk el időbélyeget, amely az operációs rendszer és az alkalmazások fejlesztőinek munkáját segítheti a protokollrétegek átfogóbb monitorozásával.

A II. téziscsoportban ismertetett RCTP protokoll ugyan egy dedikáltan aggregált forgalmat monitorozó rendszer számára készült, de a Machine To Machine (M2M), valamint az intelligens

épületgépészet is olyan mérőrendszerek telepítését teszik szükségessé, amelyeknek számos pontról, akár nagy mennyiségű adat veszteségmentes begyűjtését kell megvalósítaniuk. Ezekre az alkalmazásokra jellemző, hogy a generált adatforgalom egyik irányban, a szenzortól (probe) az adatfeldolgozó irányban domináns. Továbbá, mivel nagy számú mérőeszközt kell telepíteni, fontos szempont azok energiahatékonysága, ami elsősorban alacsony összetettségű hardver alkalmazásával valósítható meg.

A III. téziscsoportban alkalmazott háromfázisú módszerrel akár tetszőleges kódolót és azonos hangkerettípusokat alkalmazó VoIP alkalmazás számára is létrehozható olyan becslőfüggvény, amely mérhető QoS értékek alapján referencia nélkül képes az érzeti minőség (QoE) becslésére a MOS skála szerint. Mivel alacsony fokszámú polinomokkal is magas korrelációt mutattak a vizsgálat során kapott becsült értékek a szubjektív értékelésekkel, a módszer akár hardverben történő implementációban is alkalmazható, pl. gerinchálózati eszközökben történő valós idejűhöz közeli becslés megvalósításához.

1 Background and objectives of the dissertation

An increasing number of novel applications rely on the public Internet to transport specific data to the end users. Nowadays, users of the applications do not only expect acceptable functionality but the best possible user experience (e.g. smooth operation, responsivity). In the case of paid services this expectation is even higher. Therefore industrial partners like network infrastructure operators and service providers also pursue higher and higher level of user experience. Network operators need tools reporting objective metrics to get an up-to-date overview of the infrastructure and services. It enables them to tune and enhance their more and more complex infrastructure or service. Service providers can also benefit from the by continuous monitoring of the indicators. Thereby long-term trends can be traced that even influence development strategies.

Quality of Service (QoS) is a generic toolset parallel evolving with the Internet to measure and determine network performance. Modern network backbone devices are able to collect the most important QoS metrics, like network path delay (latency), deviation of inter-arrival time (jitter), packet loss, throughput, packet reordering and duplication. Besides constant passive monitoring an Internet Service Provider (ISP) can also define limits for the monitored metrics. Network resources can be rearranged when these limits are exceeded. Provider of an Internet application is also able to measure QoS but typically only at endpoints running the application.

QoS is in the focus of network research since the beginning. Publications pointed out that simply enhancing the metrics aforementioned not necessarily result in better user experience.

Quality of Experience (QoE) summarizes evaluation methods taking the users aspect into account. This research area is also

active for a long time. It initially described subjective evaluation methods. These procedures are based on user opinions submitted to statistical analysis. The results help to create a view of the experienced quality of the corresponding application. Since executing subjective evaluations is expensive and circumstantial, moreover immediate feedback is not trivial, researches correlating QoS and QoE are targeting serious demands of providers. These methods try to map measured objective QoS metrics to QoE.

My research is focused creating new assessment methods for service and media transfer quality, as well as creating new metrics. Collecting network traffic patterns with no loss has always been a fundamental aim. An effective analysis must be based on data of adequate resolution and accuracy. Since dedicated devices were inaccessible at the time, in the early phase of our research we were in great need of methods capable of lossless traffic capturing on generic computing architectures. Timestamping is one of the most important factors of the measurements. It plays essential role during the reconstruction of the traffic pattern. At the time it was also challenging to create adequate resolution timestamps for the given environment and bandwidth. Moreover extra resource demand of the timestamping affected the performance of the monitored systems negatively.

1.1 Optimized software timestamping

Software based timestamps are put onto packets after querying the system time. Time counter subsystems are simple cycle counters available in most systems. These counters do not output time directly and can be used for time calculation based on their frequencies. They had to be synchronized to a reference clock source (at once or periodically). Hereby timestamping can be split into two main phases: querying the cycle counter (clock read) and converting the raw counter delta value to wall clock time (clock

conversion). The clock source is typically a hardware component accessible through a proper interface or device driver, or through a processor register. The lower its access time and deviation the more feasible for timestamping. Frequency of the cycle counter determines the precision of timekeeping as well as of timestamping.

One of the timestamping metrics is Timestamping Resolution. All local clock sources can be characterized by their resolution (granularity) based on the representation of data structure (e.g. register). Overall timestamp resolution is determined together by the resolution of the read out value and the data structure used for storing the timestamp. Resolution describes the accuracy of the generated timestamps. Timestamp resolution depends on the clock source resolution, data structure size, clock access time and conversion precision.

If a 1 Gbit/s Ethernet link has to be analyzed, even 1 488 096 of the smallest sized packets (72 Bytes gross) may arrive in every seconds. This means that in every 672 ns a new Ethernet frame is received. At traffic measurements on servers using common 10 Gbit/s network links even shorter inter-arrival can be expected. Thus for proper analysis on this link capacity the 10^{-6} second scale is not adequate, 10^{-9} scale is required. The data structure holding the timestamps also has to represent time of resolution at 10^{-9} (e.g. 32+32 bits) [10].

Timestamping Precision can be correlated with the tracking precision of a reference clock source. Since scheduler and interrupt handlers of the operating system affect clock reading, its cost also shows some deviance. Timestamping precision is also affected by the precision of the clock source and the deviation of clock source access time.

Timestamping Accuracy or offset is the delta between the generated absolute timestamp and the reference clock source. The less the difference is the more close timestamps to the real

time values are generated. The reference clock is typically located outside of the measurement system and its resolution is higher of equal to the local clock source.

A proper timestamping method offers timestamps generated using a clock source accessed as fast as possible with a deviation as less as possible, and with a resolution adequate for the actual bandwidth. This can ensure timestamps being as close as possible to the real time moments.

Execution contexts: For security reasons operating system processes are running in privileged environment called kernel context. When a packet arrives an interrupt is being raised. Execution gets into a special state called interrupt context. The system stays in this context until the related operations (e.g. registering packets) are finished. In interrupt context execution of any other processes is suspended on the affected processor core. For performance reasons it is optimal not being for too much time in this exclusive state. Any other processes (typically user space applications) not related to the kernel are running in user space context and are coordinated by the kernel scheduler. An important factor is that data exchange between kernel and user space contexts require extra time.

Packet processing performance: Systems are constrained by physical limits (e.g. computing capacity). Evolution of networking technologies result in higher and higher link capacities. A link capacity can be fully utilized by a system with enough computing power. In generic computing architectures processing power is shared among the kernel and user space processes. A system running at the limits of processing power the deviation of the clock source access time gets higher. Therefore timestamping precision degrades. Furthermore, quality of the service or application also degrades by the higher resource demand of timestamping. When incoming packets intensity exceeds processing capacity, buffers help to temporarily overcome the

bottleneck by delaying the packet processing. Packets arriving after getting the buffers full are lost. Therefore packet processing behaves similarly to a traffic shaper. The limiting factor is the continuous processing capacity of the system. Intensity of packets temporarily exceeding the limit causes delayed forwarding of them. A long-term state will conclude into packet loss [3].

Although in generic computing architectures only a single processor core can be associated to process the traffic of a single Network Interface Card (NIC) or packet queue, other cores can be dedicated to running the remaining tasks. To increase performance, idle processor cores can be involved into certain operations of packet processing.

Processing power of mobile devices is strongly limited. Their Central Processing Unit (CPU) is less powerful compared to desktop and server computers while they also have to run several processes. Multicore processor technology is also available in the mobile and embedded devices. Part of timestamping operation that is not closely related to the local clock source access may be postponed until a later phase of execution.

My method relocates conversion to absolute time into a post processing thread. If combined with a fast access (e.g. register-based) clock source, the time spent in interrupt context can be significantly decreased [12]. Besides dedicating conversion steps to otherwise idle cores, packet processing performance improves and the probability of packet loss will be lower.

1.2 Rate Control Transport Protocol

I was taking part in a recently finished R&D project. Its aim was to develop a network analysis device for 100 Gbit/s network backbones. One of the most important functions of the device was hardware accelerated lossless network monitoring. It filters the traffic tapped and splits it into flows. These filtered flows are transmitted towards collector and processing nodes via 1 or

10 Gbit/s network links. The endpoint nodes are generic computing architectures running software for deeper analysis of the traffic, as well as creating statistics. Therefore hardware acceleration is not applicable on these nodes. Despite lossless data transmission of pre-filtered byte streams was mandatory.

We responded to the challenge with an FPGA based hardware. We faced many limitations during the design and implementation. Many functions had to be implemented within the FPGA chip. More and more logical units had to be utilized for the modules. However there are high capacity FPGA available, they are always limited in processing capacity. Their price increases exponentially by the number of integrated logical units. Another challenge is maintaining the desired clock frequency. The more complex the FPGA design is the more difficult is to keep a certain frequency as data lines are getting longer. Thus it is essential for the applied transport protocol to be as simple and economic for resources as possible. At the start of the development using Transmission Control Protocol (TCP) was an option. The protocol had to be instantiated to every interface. This implied its multiple demand on resources and thus consuming too many logical units at the expense of other modules of the FPGA design.

I inspected a number of alternate transport protocols as well but none of them fulfilled the requirements for the project. Many of them involved high demand of sending buffering. Others were too complex or had various limitations preventing the implementation in the FPGA environment. Those facts led me to design and develop a new transport protocol.

The desired transport protocol for the measurement system was the minimal use of resources (computing, buffering) while being able to transmit data without loss. The measurement device can be equipped even with 10 network interfaces for the data transfer. Due to the architectural limitations these physical links had to be implemented by independent instances along with

many capture, filtering and monitoring functions on the same chip. Thus low resource demand was an important factor during the design phase. There can be taken advantage of the new protocol in simple, resource limited hardware, like sensors and embedded systems.

Packet congestion and loss can occur on the forwarding infrastructure or at the communication endpoints. Wireless transmission is unpredictable: noise, interference are always to count with. Failures in the physical layer caused by the aforementioned factors are transmission errors. There are complex protocols like TCP to deal with it. In wired networks packet losses are caused by congestion in some of the network devices. Typically when a buffer is becoming full. In dedicated networks (in data centers, measurement systems, non-blocking switches) proper design can assure that no packets are lost within the packet forwarding infrastructure.

Data path of packets to the receiving application also leads through buffers on architectures with generic operating systems. Arriving from the Physical Layer (PHY) the first buffer the packet faces with is the buffer in the NIC itself. Although its size is small, together with a proper NIC driver this buffer can hold the data without loss while the packet is being copied into the Random Access Memory (RAM). The next buffers are parts of the operating system and its packet processing subsystem. In most cases the size of these buffers can be adjusted. This ensures that enough temporal space can be allocated even for longer bursts of incoming packets. The arriving packets had to be processed by the operating system first: protocol headers in the packets are parsed to filter out suspicious or damaged packets or the ones not being the target of the current node. During the processing the extracted information may be forwarded to other subsystem. These tasks burden the system more and more when the incoming traffic intensity increases. When there is not enough computing power

to process the incoming traffic and the buffers get full, further arriving packets are lost. Thereby among buffer usage packet processing performance is also an important metric.

1.3 VoIP QoE prediction based on the Opus voice codec

With the spread of the Internet applications providing voice transmission over the Internet Protocol (Voice over IP, VoIP) on public, heterogeneous networks are also emerging. Perspectives opened by increasing computing capacity and bandwidth joined with the flexibility of the generic software architecture, voice codecs are also catalyzed. Nowadays they generally offer a higher quality voice transfer at a specific bandwidth than the traditional voice codecs used by mobile and landline providers. Owing to the support of the industrial partners the Opus codec became an Internet Engineering Task Force (IETF) standard in a short time.

My research was motivated by the Opus not being inspected in VoIP applications before. A number of quality prediction methods using the original material have already been existing. These are called Full Reference (NR) methods. Unfortunately in a real life service environment original material is not available or not applicable. Recording and using voice material raises questions of storage space, transfer method among the also important privacy reasons. Thus FR methods are tools rarely applicable for providers. If only parts of the original material (e.g. metadata) are available, Reduced Reference (RR) methods can be chosen for the QoE prediction. In practice the input of an RR method can be part of the traffic (e.g. packet headers) and/or any metadata (like codec parameters). This still falls far from the need of an application or network provider. Therefore my research focus was the No Reference (NR) methods that only rely on measurable metrics (e.g. QoS parameters) for QoE prediction.

A number of researches were aimed at correlating QoS metrics with QoE. Individual metrics (packet loss, jitter) only showed poor correlation with the subjective quality. Thus newer researches focus on the combination of metrics.

2 New results

Thesis group I.: An offloaded software timestamping method. An important fact is that generated timestamps are used not right after the packets got captured. During the time period between clock read and analysis of the timestamp itself conversion of raw value can be executed anytime. Calculation done in a delayed or at a not extremely loaded time moment is called offloading technique. In this method only the cycle counter is queried and the value is stored during the packet processing of interrupt context. Conversion of delta values are performed in user space context. It improves packet processing performance with timestamping enabled particularly on multicore systems with one or more idle cores.

Thesis I.1.: *I worked out a new software-based timestamping method, that queries clock source with real-time latency (in the interrupt context), while conversion operations are relocated into the delay-tolerant user space context [4].*

Thesis I.2.: *I pointed out that relocating the conversion phase of timestamping to delay-tolerant execution context improves packet processing significantly. This processing power gain helps to decrease packet loss ratio when system resources are saturated at a high system load [3].*

Thesis I.3.: *Keeping the clock source read operation in interrupt context and targeted delaying of the conversion improves timestamping precision.*

Every thesis of Thesis group I was evaluated in a controlled laboratory environment with measurements executed in a programmed way. I implemented the offloaded software timestamping method in Linux.

Thesis group II.: Rate Control Transport Protocol (RTCP) is based on closed loop control. Its purpose is to control sending intensity depending on the performance metrics of the reception side. Input parameters of the control function are sampled periodically. For RCTP these metrics are utilization of the buffers and load of processor core performing the packet processing. The protocol calculates a sending rate limit based on the input parameters. Its advantage is low complexity and resource demand, since there is no need for large sending buffers, as well as fast feedback. It substitutes continuous acknowledgement of the sent data by event control.

Possible packet loss can be avoided by constant monitoring of the critical system resources, prudent tuning of packet buffers and sampling frequency to compensate the latency of the transmission. Monitoring is done by using a fixed frequency. At sampling the algorithm queries buffer utilization and processor core load. These parameters will be fed into the control function.

As the first phase in sender side of the protocol, packets are organized into a byte stream. The stream will be sent in Maximum Transmission Unit (MTU) sized packets to achieve the best performance. If low latency is also a requirement, a waiting threshold can assure that packets will be processed and sent out in a finite time. Packets are sent using the enforced sending rate. This rate is always updated when a new rate control packet is received from the reception side. To construct, store and organize packets into a byte stream a smaller sized First In First Out (FIFO) is sufficient.

Monitoring and feedback are tasks of the reception side. This is time sensitive, and thus it is important to execute the operations with minimal latency. At sampling the control function calculates maximal achievable transmission rate based on the queried metrics (buffer states and system loads). The effective enforced

sending rate will be the minimum of the calculated rates of all metrics. As long as the calculated rate value is not equal to the current rate, it will be sent to the sender side in a rate control packet. When the calculated rate is not limited, the protocol sends the maximum sending rate out in the control packet. The maximum rate is calculated at start based on the link capacity. When a limitation is enforced a constant b_1 is used as weight to decrease the rate. When a limitation is cancelled, constant b_2 is used as weight to increase the rate. These weights help to stabilize transmission rate and avoid oscillation.

Thesis II.1.: *I worked out a new transport protocol called Rate Control Transport Protocol (RCTP) aiming at congestion avoidance. This protocol puts significantly lower demand on resources compared to TCP, while it is able to transmit packets to the destination even at line rate without loss. [1].*

Thesis II.2.: *During the inspection of control functions I pointed out that control function doing a linear control between 5% and 30% using 0.2 as penalty constant and 0.1 as rewarding constant ensures the best performance for RTCP. Moreover I pointed out that by specifying the appropriate system parameters using higher sampling frequency is not necessary. However convergence to the line rate can be achieved faster when using a higher sampling rate.*

Thesis group III.: During the inspection of the Opus audio codec (IETF RFC 6716) in VoIP environment I discovered the correlation properties of QoS and QoE. Based on the results I designed a method to construct a no-reference prediction method. During the process source materials were degraded by emulating network anomalies in a biased way. After the measurement session degraded materials were subjectively evaluated. Polynomial regression was applied to calculate the polynomials

and their coefficients. During correlation analysis the relation of polynomials for a certain degree and subjective evaluations where determined. The aim of the last phase was to evaluate the generality of the polynomials and their coefficients from the second phase. The same measurement scenario was applied on different source materials. The final aim of the research was to construct low degree polynomials with QoS parameters as input determining QoE with a high reliability.

Thesis III.1.: *During the analysis of the measurement of Opus codec in VoIP environment I determined the following facts on QoS-QoE correlation. There is a linear relationship between QoS parameter jitter and QoE, there is a quadratic relationship between QoS parameter packet loss and QoE. The correlation of combined QoS parameters and QoE can be described using a quadratic relation [7].*

Thesis III.2.: *I constructed a No Reference estimation method based on low degree polynomials for VoIP applications using the Opus voice codec to reliably predict QoE. The input parameters of the prediction function are the jitter and packet loss QoS metrics [2].*

2.1 Application of the results

Timestamping method introduced through Thesis group I can help the generic computing architectures to enhance timestamping accuracy. Moreover architectures with multiple processor cores the conversion phase of timestamp generation can be relocated from the critical interrupt context to a user space context for later processing. Offloading the conversion improves packet processing capacity. Hardware with low resources (typically mobile or embedded devices) a higher performance of QoS measurement can be achieved using the new timestamping

method. More accurate timestamping and better packet processing performance can help the application developers and providers.

The freed resources by using offloaded timestamping method series of timestamps can be generated at different points of packet processing path. This can help the operating system and application developers by better monitoring of the protocol stack.

Although RTCP protocol introduced in Thesis group II is dedicated for a monitoring system to be operated in core networks, Machine to Machine (M2M) and intelligent buildings require monitoring systems transmitting large amount of measurement data. The information has to be collected from many locations possibly without loss. In these applications data transfer is generally dominant in one direction from the sensors (probe) to the collector. Since large amount of measurement devices to install, their energy efficiency is also an important factor that can be more easily achieved by using low complexity hardware.

The three-phase method introduced in Thesis group III can be used to create NR QoE prediction function for VoIP application based on any codec using uniform frame types. The function's inputs can be QoS metrics and the output is generated on the MOS scale. Since predicted values showed a high correlation with subjective evaluations even using low degree polynomials, the Opus-based function could be implemented in hardware as well to be deployed in core network devices for QoE prediction in real-time manner.

Nyilvántartási szám: DEENK/39/2016.PL
Tárgy: PhD Publikációs Lista

Jelölt: Skopkó Tamás
Neptun kód: KBT3J6
Doktori Iskola: Informatikai Tudományok Doktori Iskola

A PhD értekezés alapjául szolgáló közlemények

Idegen nyelvű tudományos közlemény(ek) hazai folyóiratban (1)

1. Orosz, P., **Skopkó, T.**, Varga, M.: RCTP: A Low-complexity Transport Protocol for Collecting Measurement Data.
Infocommun. J. 6 (3), 28-36, 2014. ISSN: 2061-2079.

Idegen nyelvű tudományos közlemény(ek) külföldi folyóiratban (3)

2. Orosz, P., **Skopkó, T.**, Nagy, Z., Lukovics, T.: A No-reference Voice Quality Estimation Method for Opus-based VoIP Services.
1942-2601 7 (1-2), 12-21, 2014.

3. **Skopkó, T.**: Loss Analysis of the Software-based Packet Capturing.
CJECE. 5 (1), 107-111, 2012. ISSN: 1844-9689.

4. Orosz, P., **Skopkó, T.**: Performance Evaluation of a High Precision Software-based Timestamping Solution for Network Monitoring.
Int. J. Adv. Softw. 4 (1/2), 181-188, 2011. EISSN: 1942-2628.

Magyar nyelvű konferencia közlemény(ek) (1)

5. Orosz P., **Skopkó T.**: Adatcsomagok nagyfelbontású időbélyegzése osztott erőforrású környezetben.
In: XI. ENELKO - XX SzámOkt Nemzetközi Energetikai-elektrotechnikai és Számítástechnika és Oktatási Konferencia : Szatmárnémeti, Románia, 2010.10.07-2010.10.10. [Erdélyi Magyar Műszaki Tudományos Társaság], Kolozsvár, 192-197, 2010.

Idegen nyelvű konferencia közlemény(ek) (6)

6. Orosz, P., **Skopkó, T.**: Multi-threaded Packet Timestamping for End-to-End QoS Evaluation.
In: 8th International Conference on Systems and Networks Communications, ICSNC 2013, October 27-31, 2013, Venice, Italy : Proceeding. Ed.: Renzo Davoli, Josef Noll, [s.n.], [s.l.], 94-99, 2013. ISBN: 9781612083056

7. Orosz, P., **Skopkó, T.**, Nagy, Z., Lukovics, T.: Performance Analysis of the Opus Codec in VoIP Environment Using QoE Evaluation.
In: 8th International Conference on Systems and Networks Communications, ICSNC 2013, October 27-31, 2013, Venice, Italy : Proceeding. Ed.: Renzo Davoli, Josef Noll, [s.n.], [s.l.], 89-93, 2013. ISBN: 9781612083056

8. Orosz, P., **Skopkó, T.**, Imrek, J.: A NetFPGA-based Network Monitoring System with Multi-layer Timestamping: Rnetprobe.
In: Proceedings of the 15th International Telecommunications Network Strategy and Planning Symposium. [s.n.], [s.l.], 319-324, 2012.

9. Orosz, P., **Skopkó, T.**, Imrek, J.: Performance Evaluation of the Nanosecond Resolution Timestamping Feature of the Enhanced Libpcap.
In: The Sixth International Conference on Systems and Networks Communications, ICSNC 2011, October 23-29, 2011 - Barcelona, Spain. [s.n.], [s.l.], 1-6, 2011.

10. Orosz, P., **Skopkó, T.**: Timestamp-resolution Problem of Traffic Analysis on High Speed Networks.
In: International Conference on Applied Informatics (8)(2010.01.27-2010.01.30)(Eger)Proceedings of the 8th International Conference on Applied Informatics : January 27-30, 2010, Eger. Ed.: by Attila Egri-Nagy et al., Eszterházy K. College, Eger, 237-245, [2011]. ISBN: 9789639894723

11. Orosz, P., **Skopkó, T.**: Software-Based Packet Capturing with High Precision Timestamping for Linux.
In: The Fifth International Conference on Systems and Networks Communications : ICSNC 2010. [IEEE Computer Society], [Los Alamitos, CA], 381-386, 2010. ISBN: 9780769541457

További Közlemények

Idegen nyelvű közlemény(ek) külföldi folyóiratban (1)

12. **Skopkó, T.**, Orosz, P.: Investigating of the Precision of the TSC-based Packet timestamping. *CJECE. 4* (1), 117-122, 2011. ISSN: 1844-9689.

Idegen nyelvű konferencia közlemény(ek) (1)

13. Orosz, P., **Skopkó, T.**, Nagy, Z., Varga, P., Gyimóthi, L.: A Case-Study on Correlating Video QoS and QoE.
In: Proceedings of the 2014 IEEE Network Operations and Management Symposium (NOMS). Ed.: Hanan Lutfiyya, Piotr Cholda, Institute of Electrical and Electronic Engineers, Red Hook, NY, 1-5, 2014. ISBN: 9781479909124
DOI: http://dx.doi.org/10.1109/NOMS.2014.6838399

A DEENK a Jelölt által az iDEa Tudóstérbe feltöltött adatok bibliográfiai és tudománymetriai ellenőrzését a tudományos adatbázisok és a Journal Citation Reports Impact Factor lista alapján elvégezte.

Debrecen, 2016.02.19.

DEBRECENI EGYETEM KENÉZY ÉLETTUDOMÁNYI KÖNYVTÁRA EGYETEMI ÉS NEMZETI KÖNYVTÁR

Registry number: DEENK/39/2016.PL
Subject: Ph.D. List of Publications

Candidate: Tamás Skopkó
Neptun ID: KBT3J6
Doctoral School: Doctoral School of Informatics

List of publications related to the dissertation

Foreign language scientific article(s) in Hungarian journal(s) (1)

1. Orosz, P., **Skopkó, T.**, Varga, M.: RCTP: A Low-complexity Transport Protocol for Collecting Measurement Data.
   Infocommun. J. 6 (3), 28-36, 2014. ISSN: 2061-2079.

Foreign language scientific article(s) in international journal(s) (3)

2. Orosz, P., **Skopkó, T.**, Nagy, Z., Lukovics, T.: A No-reference Voice Quality Estimation Method for Opus-based VoIP Services.
   1942-2601 7 (1-2), 12-21, 2014.

3. **Skopkó, T.**: Loss Analysis of the Software-based Packet Capturing.
   CJECE. 5 (1), 107-111, 2012. ISSN: 1844-9689.

4. Orosz, P., **Skopkó, T.**: Performance Evaluation of a High Precision Software-based Timestamping Solution for Network Monitoring.
   Int. J. Adv. Softw. 4 (1/2), 181-188, 2011. EISSN: 1942-2628.

Hungarian conference proceeding(s) (1)

5. Orosz P., **Skopkó T.**: Adatcsomagok nagyfelbontású időbélyegzése osztott, erőforrású környezetben.
   In: XI. ENELKO - XX SzámOkt Nemzetközi Energetikai-elektrotechnikai és Számítástechnika és Oktatási Konferencia : Szatmárnémeti, Románia, 2010.10.07-2010.10.10. [Erdélyi Magyar Műszaki Tudományos Társaság], Kolozsvár, 192-197, 2010.

Foreign language conference proceeding(s) (6)

6. Orosz, P., **Skopkó, T.**: Multi-threaded Packet Timestamping for End-to-End QoS Evaluation.
In: 8th International Conference on Systems and Networks Communications, ICSNC 2013, October 27-31, 2013, Venice, Italy : Proceeding. Ed.: Renzo Davoli, Josef Noll, [s.n.], [s.l.], 94-99, 2013. ISBN: 9781612083056

7. Orosz, P., **Skopkó, T.**, Nagy, Z., Lukovics, T.: Performance Analysis of the Opus Codec in VoIP Environment Using QoE Evaluation.
In: 8th International Conference on Systems and Networks Communications, ICSNC 2013, October 27-31, 2013, Venice, Italy : Proceeding. Ed.: Renzo Davoli, Josef Noll, [s.n.], [s.l.], 89-93, 2013. ISBN: 9781612083056

8. Orosz, P., **Skopkó, T.**, Imrek, J.: A NetFPGA-based Network Monitoring System with Multi-layer Timestamping: Rnetprobe.
In: Proceedings of the 15th International Telecommunications Network Strategy and Planning Symposium. [s.n.], [s.l.], 319-324, 2012.

9. Orosz, P., **Skopkó, T.**, Imrek, J.: Performance Evaluation of the Nanosecond Resolution Timestamping Feature of the Enhanced Libpcap.
In: The Sixth International Conference on Systems and Networks Communications, ICSNC 2011, October 23-29, 2011 - Barcelona, Spain. [s.n.], [s.l.], 1-6, 2011.

10. Orosz, P., **Skopkó, T.**: Timestamp-resolution Problem of Traffic Analysis on High Speed Networks.
In: International Conference on Applied Informatics (8)(2010.01.27-2010.01.30)(Eger)Proceedings of the 8th International Conference on Applied Informatics : January 27-30, 2010, Eger. Ed.: by Attila Egri-Nagy et al., Eszterházy K. College, Eger, 237-245, [2011]. ISBN: 9789639894723

11. Orosz, P., **Skopkó, T.**: Software-Based Packet Capturing with High Precision Timestamping for Linux.
In: The Fifth International Conference on Systems and Networks Communications : ICSNC 2010. [IEEE Computer Society], [Los Alamitos, CA], 381-386, 2010. ISBN: 9780769541457

List of other publications

Foreign language scientific article(s) in international journal(s) (1)

12. **Skopkó, T.**, Orosz, P.: Investigating of the Precision of the TSC-based Packet timestamping. *CJECE. 4* (1), 117-122, 2011. ISSN: 1844-9689.

Foreign language conference proceeding(s) (1)

13. Orosz, P., **Skopkó, T.**, Nagy, Z., Varga, P., Gyimóthi, L.: A Case-Study on Correlating Video QoS and QoE.
In: Proceedings of the 2014 IEEE Network Operations and Management Symposium (NOMS). Ed.: Hanan Lutfiyya, Piotr Cholda, Institute of Electrical and Electronic Engineers, Red Hook, NY, 1-5, 2014. ISBN: 9781479909124
DOI: http://dx.doi.org/10.1109/NOMS.2014.6838399

The Candidate's publication data submitted to the iDEa Tudóstér have been validated by DEENK on the basis of Web of Science, Scopus and Journal Citation Report (Impact Factor) databases.

19 February, 2016