



FACULTATEA DE ELECTRONICĂ, TELECOMUNICAȚII ȘI
TEHNOLOGIA INFORMAȚIEI

Ing. Andrei Bogdan Rus

TEZĂ DE DOCTORAT

CALITATEA SERVICIILOR PRIN TEHNICI CROSS-LAYER PENTRU VIITORUL INTERNET

QUALITY OF SERVICE THROUGH CROSS-LAYER TECHNIQUES FOR THE FUTURE INTERNET

**Conducător științific,
Prof.dr.ing. Virgil DOBROTĂ**

Comisia de evaluare a tezei de doctorat:

- PREȘEDINTE: - Prof.dr.ing. **Marina Țopa** - decan, Facultatea de Electronică,
Telecomunicații și Tehnologia Informației, Universitatea Tehnică din
Cluj-Napoca;
- MEMBRI: - Prof.dr.ing. **Virgil Dobrotă** - conducător științific, Universitatea Tehnică din
Cluj-Napoca;
- Prof.dr.ing. **Tatiana Rădulescu** - referent, Universitatea "Politehnica" din
Bucuresti;
- Prof.dr.ing. **Radu Vasii** - referent, Universitatea „Politehnica” din
Timișoara;
- Prof.dr.ing. **Aurel Vlaicu** - referent, Universitatea Tehnică din Cluj-Napoca.

Susținerea publică a tezei de doctorat: 25 februarie 2011
Aula „Alexandru Domșa”
Universitatea Tehnică din Cluj-Napoca
Strada Constantin Daicoviciu nr. 15

Mulțumiri

Doresc să adresez mulțumirile cuvenite tuturor celor care, din punct de vedere profesional, direct sau indirect, prin dezbaterile și sugestiile oferite, au contribuit la elaborarea acestei teze.

Mulțumesc domnului prof.dr.ing Virgil DOBROTĂ pentru îndrumarea, încurajările și suportul continuu oferit de-a lungul întregii perioade de pregătire a doctoratului. De asemenea, le mulțumesc și colegilor din colectivul Unified Communications Laboratories: conf.dr.ing. Daniel ZINCA, șl.dr.ing. Tudor Mihai BLAGA, as.dr.ing. Cristian Mihai VANCEA, ing. Gabriel LAZĂR, ing. Melinda BARABAS, ing. Georgeta BOANEA și ing. Sabin SĂRMAȘ pentru întreaga colaborare.

De asemenea mulțumesc dr. K.K. RAMAKRISHNAN, IEEE Fellow, Distinguished Member of Technical Staff, și dr. Jacobus E van der MERWE, ambii de la AT&T Labs Research, S.U.A. pentru discuțiile și ideile valoroase schimbate în cadrul conferințelor IEEE LANMAN. Domnilor prof.dr.ing Aurel VLAICU, prorector al Universității Tehnice din Cluj-Napoca și conf.dr.ing Bogdan ORZA aș dori să le mulțumesc pentru sugestiile primite cu ocazia participării în comisia de evaluare a rapoartelor de doctorat. Mulțumirile se îndreaptă și către colaboratorii cu care am lucrat în diverse proiecte de cercetare: conf.dr.ing Zsolt POLGAR și ing. Zsuzsanna KISS.

Deoarece o parte din munca prezentată în aceasta teză a fost efectuată în cadrul proiectului FP7-ICT-2007-1-216041-4WARD, finanțat de către Uniunea Europeană, aș dori să îi mulțumesc coordonatorului de proiect Henrik ABRAMOWICZ (EAB) și colaboratorilor: Giorgio NUNZI (NEC), Dominique DUDKOWSKI (NEC), Rolf STADLER (KTH) și Alberto GONZALEZ (KTH). Am avut șansa să lucrez în cadrul aceluiași proiect cu colegi de la Siemens PSE Brașov și Siemens PSE Cluj-Napoca. Mulțumiri speciale se îndreaptă către Frank-Uwe ANDERSEN și Cornel PAMPU, la vremea respectivă angajați ai companiei Nokia Siemens Networks Germania.

Nu în ultimul rând, sunt recunoscător părinților și surorii mele pentru suportul oferit. De asemenea îi mulțumesc soției Anca RUS pentru încurajări și înțelegere.

Ing. Andrei Bogdan RUS
Februarie 2011

Cuprins

Capitolul 1	Introducere	6
1.1	Soluții QoS standardizate.....	6
1.1.1	Integrated Services (IntServ)	6
1.1.2	Differentiated Services (DiffServ)	7
1.2	Prezentarea tehnicilor Cross-Layer.....	7
1.3	O privire de ansamblu asupra conceptului de management distribuit in rețea (In-Network Management)	8
1.4	Motivația tezei.....	8
Capitolul 2	Clasificarea și evaluarea mecanismelor Cross-Layer QoS.....	10
2.1	Motivație	10
2.2	Necesitatea tehnicilor Cross-Layer	10
2.3	Tipuri de semnalizări Cross-Layer existente	10
2.4	Metode de implementare a semnalizărilor Cross-Layer.....	11
2.5	Clasificarea implementărilor Cross-Layer	13
2.6	Concluzii.....	13
Capitolul 3	Proiectarea și implementarea preliminară a unei arhitecturi Clean-Slate pentru Cross-Layer CLQ	14
3.1	Motivație	14
3.2	Principii de proiectare a modulului de Cross-Layer QoS (CLQ).....	14
3.3	Principii de implementare a modulului Cross-Layer QoS	16
3.4	Concluzii.....	17
Capitolul 4	Metodologii propuse pentru măsurarea parametrilor QoS	18
4.1	Motivație	18
4.2	Măsurarea parametrului Întârziere dus-întors (RTT - Round-Trip Time).....	18
4.3	Măsurarea parametrului Latență (OWD – One-Way Delay).....	19
4.4	Măsurarea parametrului Rată de transfer disponibilă (ATR – Available Transfer Rate)...	19
4.5	Estimarea parametrului Probabilitate de eroare pe bit (BER – Bit Error Rate)	20
4.6	Concluzii.....	21
Capitolul 5	Implementarea și evaluarea tehnicilor Cross-Layer pentru controlul congestiei	22
5.1	Motivație	22
5.2	Arhitectura modulului de Cross-Layer QoS	22
5.3	Folosirea serviciilor modulului CLQ de către modulul NC pentru controlul congestiei	22
5.4	Rezultate experimentale	23
5.5	Concluzii.....	25
Capitolul 6	Implementarea și evaluarea CLQ pentru rutarea conștientă de QoS.....	26
6.1	Motivație	26
6.2	Utilizarea parametrilor QoS.....	26
6.3	Utilizarea mecanismului CLQ pentru rutare	26

6.4	Testarea performanțelor soluției propuse	27
6.5	Concluzii.....	30
Capitolul 7 Implementarea și evaluarea modului CLQ pentru routere MPLS.....		31
7.1	Motivație	31
7.2	Calculul metricii compozite	31
7.3	Teste	32
7.4	Concluzii.....	34
Capitolul 8 Îmbunătățirea algoritmului lui Dijkstra cu CLQ		35
8.1	Motivație	35
8.2	O privire de ansamblu asupra algoritmului lui Dijkstra clasic.....	35
8.3	Modificări aduse algoritmului lui Dijkstra.....	36
8.4	Testarea performanțelor algoritmului lui Dijkstra modificat.....	37
8.5	Concluzii.....	38
Capitolul 9 Viitorul Internet de obiecte și dispozitive inteligente		39
9.1	Motivație	39
9.2	Entități prevăzute în viitorul Internet	39
9.2.1	Obiecte inteligente de informație	39
9.2.2	Obiecte inteligente de tip resursă	40
9.2.3	Echipament inteligent.....	40
9.3	Scenariu propus – Cooperarea dintre IIO și SD.....	41
9.4	Concluzii.....	42
Capitolul 10 Contribuții la calitatea serviciilor prin tehnici Cross-Layer pentru viitorul Internet		43
10.1	Sumarul contribuțiilor.....	43
10.2	Remarci finale	46
10.3	Premii obținute.....	46
10.4	Publicații personale	47
Bibliografie selectivă.....		50

Capitolul 1 Introducere

În ziua de azi Internetul este considerat ca fiind o entitate importantă a vieții cotidiene, oferind noi posibilități de comunicare între oameni și facilitând accesul la informație. Pentru a satisface cererile tot mai complexe ale utilizatorilor, noi servicii apar pe piață în fiecare zi. Un lucru important de menționat este acela că cele mai populare sunt cele ce presupun transmisii de date audio/video în timp real. Acestea necesită menținerea parametrilor de calitate a serviciilor QoS într-un anumit domeniu, pentru a garanta un grad ridicat de satisfacție a utilizatorului. Deși cercetarea focalizată pe studiul mecanismelor QoS este prezentă în lumea rețelisticii de un timp îndelungat, încă nu se cunoaște o implementare la scară largă a acestora în rețelele de producție. De aceea, utilizatorul primește în continuare servicii de tip Best Effort (se încearcă atingerea unui grad cat mai ridicat al serviciilor însă fără a oferi nici o garanție cu privire la un nivel minim). Unul din motivele datorită căruia s-a ajuns în această situație este faptul că activarea și optimizarea mecanismelor QoS necesită o oarecare experiență și înțelegere a tuturor proceselor implicate. Din această cauză, operatorii preferă să supradimensioneze rețeaua, abordare care duce la o creștere nejustificată a costurilor.

O aplicație nou apărută dar care devine tot mai populară în ziua de azi este cea cunoscută sub numele de Cloud Computing. Aceasta presupune distribuirea unui set de resurse pe un grup de echipamente, astfel încât acestea să fie disponibile unui număr ridicat de utilizatori conectați la distanță. Cu ale cuvinte este vorba de acele aplicații care sunt accesibile prin Internet. Studiile arată că mecanismele de QoS vor juca un rol foarte important în garantarea succesului acestor noi servicii. În [Xio09] se spune despre lumea comutației de pachete, că există o legătură strânsă între numărul maxim de utilizatori, gradul minim garantat de satisfacție și numărul maxim de servicii suportate. Importanța mecanismelor QoS în Cloud Computing este ilustrată de asemenea în [Cao09], unde se propune o nouă arhitectură ce suporta garantarea de servicii QoS într-un domeniu de tip Cloud Computing.

Dacă noțiunea de QoS este considerată a fi un indicator obiectiv a performanței rețelei din punct de vedere tehnic, noțiunea de QoE (Quality of Experience – calitatea serviciilor din punctul de vedere al utilizatorului) poate fi definită ca și indicatorul cel mai relevant din perspectiva celui ce primește respectivul serviciu. Cu alte cuvinte, QoE este un parametru centrat pe experiența percepută de către individ, nefiind focalizat pe tehnologie [Dro08]. Astfel, operatorii trebuie să execute diverse teste pentru a verifica dacă se asigură un nivel ridicat de satisfacție a utilizatorilor, indicat de către parametrul QoE. Toate aceste măsuri sunt necesare dacă se dorește asigurarea succesului într-o lume a telecomunicațiilor unde concurența este acerbă.

1.1 Soluții QoS standardizate

În această secțiune se vor prezenta două soluții standardizate care asigură implementarea mecanismelor QoS în rețea.

1.1.1 Integrated Services (IntServ)

IntServ este considerat a fi prima încercare standardizată de includere a mecanisme QoS în arhitectura unei rețele cu comutație de pachete [Bra94]. Acesta a fost emis de către organismul internațional IETF (Internet Engineering Task Force). Principala caracteristică a acestei abordări este faptul că se încearcă garantarea unui set de parametri QoS pentru fiecare flux de date în parte, utilizând protocolul RSVP (Resource Reservation Protocol – protocol de rezervare a resurselor) [Rad07]. Acesta permite accesul unui nou flux de date în rețea doar dacă există destule resursele disponibile pentru a suporta cerințele adiționale de QoS. Astfel, mecanismul RSVP este unul fezabil

datorită faptului că reușește garantarea strictă a parametrilor QoS, datorită faptului că un flux adițional va fi primit în rețea doar în cazul în care resursele disponibile permit acest lucru. În acest fel, se reușește asigurarea nivelului de QoS corespunzător pentru fiecare micro flux de date. Pe lângă avantajele menționate anterior, există și dezavantaje. Cel mai important este legat de scalabilitatea soluției deoarece este necesară implementarea unui mecanism de semnalizare dedicat fiecărui flux existent în rețea. Deoarece acest număr poate fi foarte ridicat în special în nodurile centrale, procesarea și menținerea informațiilor de control a fluxului nu este o cerință ușor de realizat. Astfel de operații pot afecta simțitor performanțele routerelor datorită puterii ridicate de procesare necesare. De aceea, activarea mecanismelor IntServ se recomandă doar în partea de acces a rețelei unde scalabilitatea nu este o problemă atât de mare datorită faptului că există un număr redus de fluxuri de date.

1.1.2 Differentiated Services (DiffServ)

Recunoscând limitările de scalabilitate ale soluției IntServ, organismul IETF a dezvoltat o altă soluție cunoscută sub numele de DiffServ (Differentiated Services – servicii diferențiate) [Rad07]. Această abordare încearcă să elimine problema scalabilității caracteristice primei soluții. Astfel se introduce conceptul de servicii diferențiate care definește un set de clase de trafic. În acest sens, cerințele de QoS sunt mapate pe un set de clase de prioritate, definite în standard. Acesta prevede servicii de tip premium și servicii asigurate.

Cel mai important avantaj al acestei soluții este scalabilitatea îmbunătățită [Bla98], datorită faptului că informația de stare corespunzătoare fiecărui flux este memorată doar în routerele marginale ale domeniului, în timp ce nodurile centrale vor controla doar fluxurile agregate. Există totuși o limitare ce constă în faptul că mecanismele DiffServ nu dețin o imagine de ansamblu asupra întregii rețelei, având ca și consecință faptul că deciziile luate nu sunt întotdeauna cele optime.

1.2 Prezentarea tehnicilor Cross-Layer

În ziua de azi, când serviciile în timp-real devin din ce în ce mai populare, abordarea clasică în ceea ce privește mecanismele de garantare a parametrilor QoS nu mai este satisfăcătoare deoarece acești parametri trebuie menținuți într-un domeniu strict, astfel încât utilizatorul să experimenteze o calitate ridică a serviciilor primite. Activarea tehnicilor Cross-Layer (ce încalcă regulile impuse de stivele de protocoale OSI și TCP/IP) este soluția cea mai atractivă, permițând protocoalelor de comunicare să schimbe între ele informații de management și control, pentru optimizarea deciziilor luate în timpul funcționării. Astfel, protocoalele de comunicare aflate la straturile superioare își vor putea adapta deciziile ținând cont de parametrii de trafic corespunzători straturilor inferioare. În lucrarea [Gra05] se spune despre aceste tehnici Cross-Layer că sunt o modalitate de adaptare a protocoalelor proiectate pentru rețelele cablate astfel încât să funcționeze optim în rețelele radio.

În abordarea tradițională (cea care implementează stiva TCP/IP), semnalizarea între protocoalele de comunicații ce rulează la diferite nivele se face prin intermediul unor interfețe standardizate aflate între straturi adiacente. Ca și alternativă, tehnicile Cross-Layer permit comunicarea directă între diverse protocoale, indiferent de nivelul la care acestea rulează, folosind orice tip de interfață (vezi Figura 1-1). O altă caracterizare interesantă vizavi de tehnicile Cross-Layer poate fi întâlnită în [Sri05] unde acestea sunt văzute ca și metode de proiectare a protocoalelor de comunicare astfel încât să se elimine constrângerile caracteristice modelelor de referință OSI sau TCP/IP.

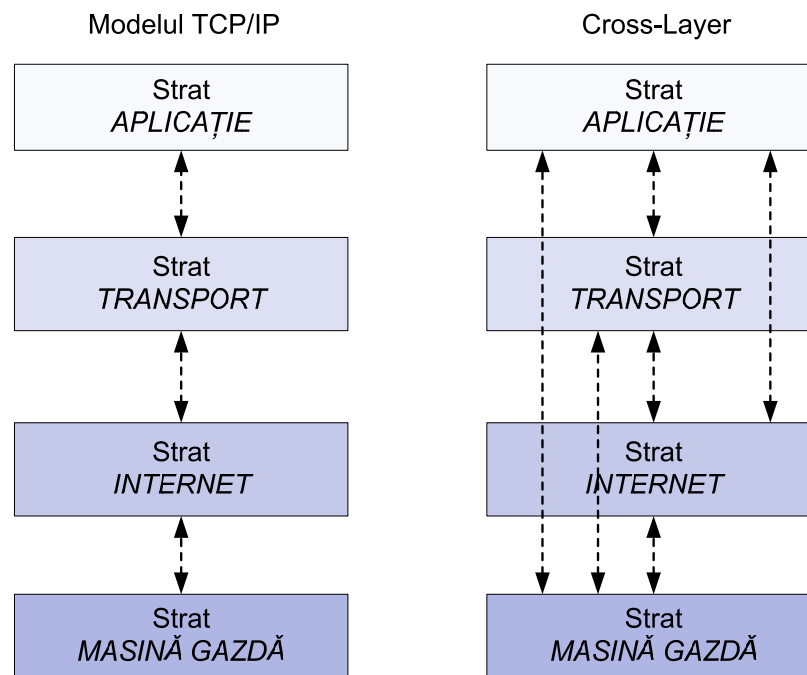


Figura 1-1 Moduri de interacțiune între protocoale în cazul modelelor TCP/IP și Cross-Layer

1.3 O privire de ansamblu asupra conceptului de management distribuit în rețea (In-Network Management)

Conceptul de management distribuit în rețea (INM – In-Network Management) a fost propus și dezvoltat în cadrul proiectului de cercetare FP7-4WARD, ce are ca și obiectiv principal propunerea unei noi arhitecturi care să poată fi utilizată în Internetul viitorului. INM presupune distribuirea operațiilor de gestiune a rețelei, în toate nodurile care o compun, astfel încât acestea să fie aduse cât mai aproape de serviciile oferite. Una dintre provocările principale o reprezintă identificarea arhitecturii optime care să permită atingerea tuturor obiectivelor propuse. Beneficiul major al acestei abordări este suportul intrinsec al proceselor de auto-gestionare a resurselor disponibile.

1.4 Motivația tezei

În abordarea clasică, parametrii QoS într-o rețea de comutație de pachete sunt monitorizați la nivelul stratului Rețea. În acest sens, o etapa inițială în cadrul acestei teze a presupus proiectarea și studiul unui instrument de emulare a diverselor comportamente caracteristice rețelelor de tip WAN (Wide Area Network – rețele de largă răspândire). Emularea diverselor comportamente era realizată prin variația unui set de parametri QoS la nivelul stratului Rețea, așa cum apare descris în [Rus08a]. În respectiva lucrare, aplicația prezentată controlează următorii parametri QoS: întârziere, procent de pachete pierdute și procent de pachete duplicate. Emulatorul propus folosește două module disponibile în kernel-ul Linux: iptables and tc. În urma investigațiilor efectuate, concluzia rezultată specifică faptul că deciziile luate la nivelul stratului Rețea nu sunt cele optime. Astfel, pentru a contracara aceste dezavantaje, am luat în considerare studiul tehnicilor Cross-Layer (vezi Figura 1-2). De asemenea, majoritatea activităților ce se desfășoară prin internet: eÎnvățare, eBusiness, eGuvernare etc. au nevoie de o infrastructură modernă care să fie capabilă să garanteze un anumit nivel de calitate a serviciilor. Până în prezent, acest vis a rămas încă neîmplinit deoarece soluțiile clasice nu reușesc să facă față cerințelor actuale [Vas06]. Mecanismele propuse în această teză au ca și scop garantarea automată a parametrilor QoS în funcție de cerințele fiecărui serviciu în parte.

La trei luni de la începerea tezei, a început colaborarea în cadrul proiectului FP7-4WARD cu diverși parteneri din Europa, atât din industrie (Ericsson EAB, Nokia Siemens Networks, NEC etc.) cât și din mediul academic (Technion, Fraunhofer, Swedish Institute of Computer Science, Waterford Institute of Technology etc.). În acest cadru am avut posibilitatea de a contribui la primele cercetări ce au vizat viitorul Internet, în cadrul unui grup dedicat tematicii In-Network Management în care contribuția principală a fost proiectarea mecanismului Cross-Layer QoS (CLQ).

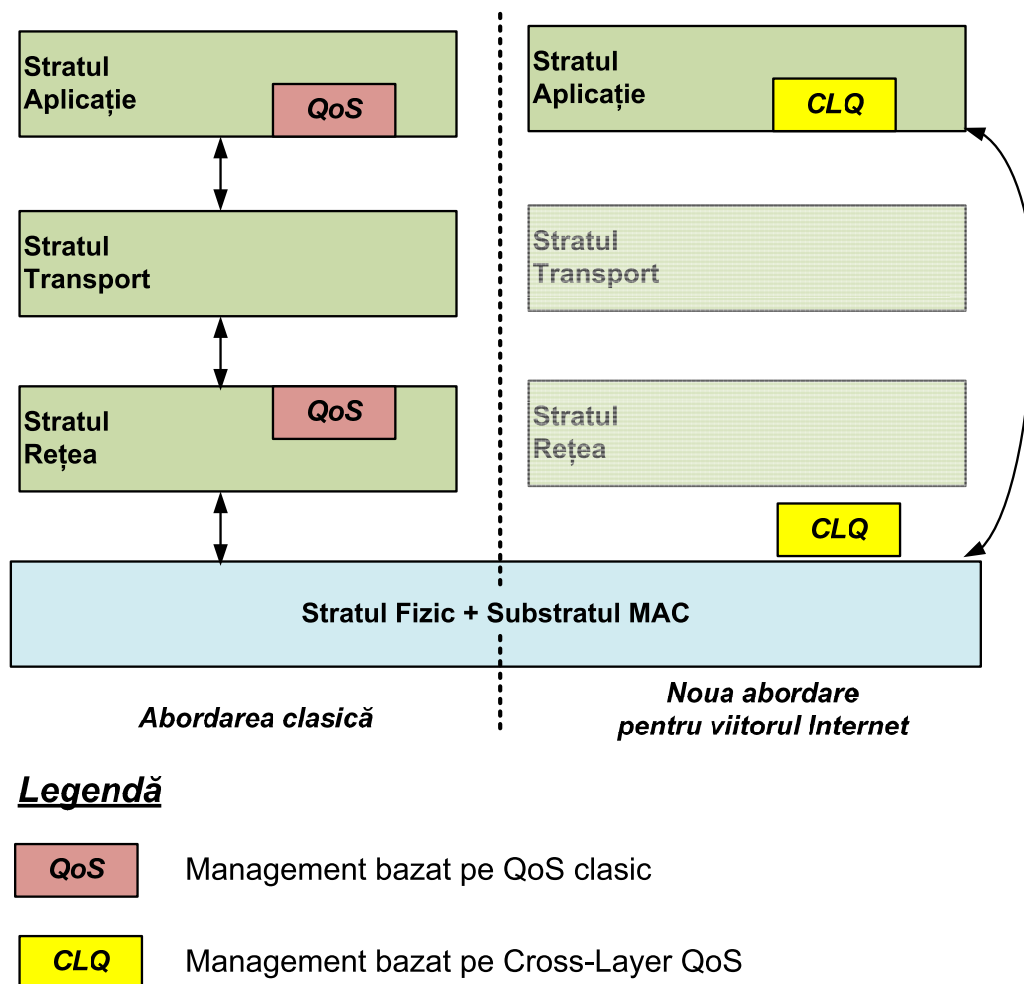


Figura 1-2 Comparație între abordarea clasică și Cross-Layer

Această teză reflectă atât puncte de vedere comune cu partenerii din proiectul FP7-4WARD cât și viziuni proprii, mai ales privind fazele următoare ale viitorului Internet. Activitatea a fost de asemenea validată prin articole acceptate de prestigioase conferințe cum ar fi: IEEE LANMAN 2010, IEEE SOFTCOM 2009, IEEE SOFTCOM 2010, ITU-T KALEIDOSCOPE 2010.

Capitolul 2 Clasificarea și evaluarea mecanismelor Cross-Layer QoS

2.1 Motivație

Acest capitol este dedicat prezentării unei imagini de ansamblu asupra tehnicilor Cross-Layer a căror scop este de îmbunătățire a calității serviciilor într-o rețea cu comutație de pachete. Deoarece această teză se axează pe propunerea și testarea mai multor implementări Cross-Layer existente, cititorul trebuie familiarizat în prealabil cu noțiunile care vor fi vehiculate în restul lucrării.

2.2 Necesitatea tehnicilor Cross-Layer

Abordarea clasică, în care protocoalele de comunicații funcționează independent unul de celălalt, este folosită cu succes în cazul aplicațiilor care nu presupun transmisii de date în timp real și care funcționează în rețele cablate. Dacă în schimb ne îndreptăm atenția asupra serviciilor în timp real, sau asupra aplicațiilor ce rulează în rețele radio, care au caracteristici variabile în timp, abordarea clasică s-a dovedit a fi nesatisfăcătoare. În acest sens, un anumit grad de adaptabilitate la condițiile actuale din rețea este necesară pentru a optimiza execuția aplicațiilor de nivel înalt sau a diverselor protocoale de comunicații, folosind informații primite de la straturile inferioare. De asemenea, tehnicile Cross-Layer pot fi folosite și pentru a optimiza energia consumată de echipamente, îmbunătățind astfel autonomia acestora.

2.3 Tipuri de semnalizări Cross-Layer existente

În literatura de specialitate se pot găsi diverse tipuri de semnalizări Cross-Layer schimbate între protocoale aflate la nivele diferite. O clasificare a acestora a fost făcută în [Sri05] și este de asemenea ilustrată în Figura 2-1.

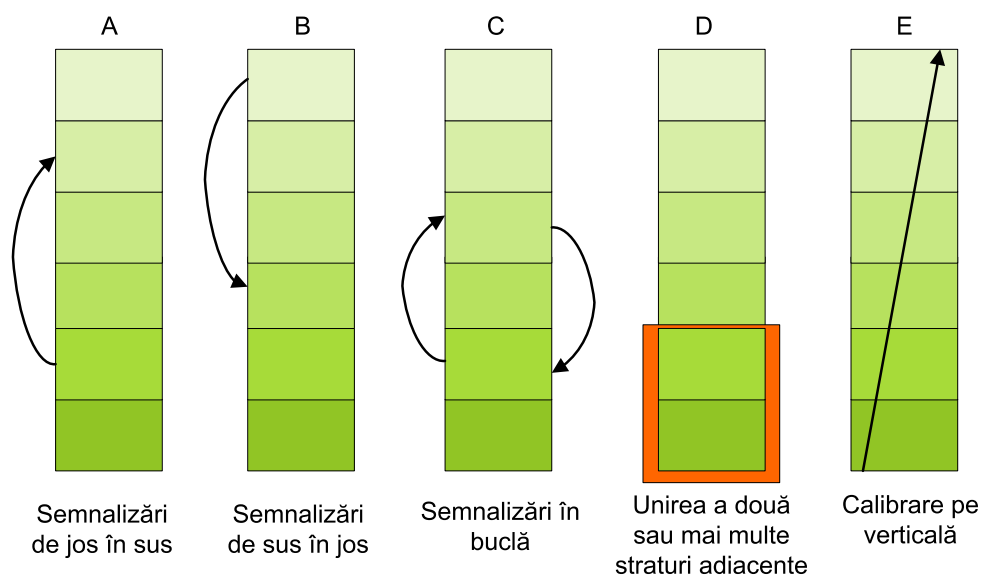


Figura 2-1 Tipuri de semnalizări Cross-Layer existente [Sri05]

A. Semnalizări Cross-Layer de jos în sus

Acest tip de semnalizare Cross-Layer este folosit atunci când un protocol ce rulează la un nivel superior necesită indicatori de performanță caracteristici straturilor inferioare, în timpul rulării. Pentru implementarea acestui mecanism, este necesară o nouă interfață de comunicare care să permită comunicarea eficientă între cele două straturi (vezi Figura 2-1 A).

B. Semnalizări Cross-Layer de sus în jos

Semnalizările Cross-Layer de sus în jos sunt utilizate atunci când este necesară ajustarea parametrilor unui protocol ce rulează la un strat inferior în funcție de informațiile extrase de la un protocol/aplicație ce rulează la un strat superior. Pentru a facilita comunicarea directă între straturi, este nevoie de crearea unei interfețe dedicate. Acest model de semnalizare este ilustrat în Figura 2-1 B. O aplicație interesantă în care s-ar putea folosi o astfel de abordare este aceea în care stratul Legături de Date necesită informații în timp real asupra parametrilor QoS ce trebuie respectați de către un flux de date provenit de la stratul Aplicație. Astfel, datele cele mai sensibile la întârzieri vor fi stocate într-o coadă de așteptare cu prioritate mai mare.

C. Semnalizări Cross-Layer în buclă

Semnalizările în buclă presupun un schimb continuu de informații între două protocoale ce rulează la straturi diferite. Acest schimb de date este făcut în vederea optimizării continue a deciziilor luate de către cele două în timpul funcționării. O ilustrare grafică a acestui tip de semnalizare poate fi studiată în Figura 2-1 C.

D. Unirea a două sau mai multe straturi adiacente

O altă tehnică de tip Cross-Layer ce poate fi implementată este aceea în care două sau mai multe straturi vecine pot fi unite, rezultând astfel un super-strat ce va conține suma tuturor serviciilor oferite de straturile inițiale. O reprezentare grafică a acestei metode este ilustrată în Figura 2-1 D. Un avantaj al acestei abordări este acela că nu necesită crearea de noi interfețe de comunicare. Astfel, schimbul de date cu restul stivei de protocoale se va face folosind interfețele standard corespunzătoare stratului inferior și celui superior, existente în arhitectura originală

E. Calibrare pe verticală

Această metodă, prezentată în Figura 2-1 E, presupune ajustarea comportamentului unui anumit protocol de comunicații în funcție de parametrii tuturor protocoalelor ce rulează la straturile inferioare. Motivul pentru utilizarea unei astfel de tehnici este acela că există o dependență strânsă între performanțele transmisiei sesizate la un anumit nivel și comportamentul protocoalelor inferioare. Calibrarea pe verticală poate fi făcută atât static cât și dinamic. Prima varianta presupune comunicarea indicatorilor specifici protocoalelor inferioare înainte de începerea transmisiei, sau chiar mai devreme, în faza de proiectare. În cazul calibrării dinamice, ajustarea parametrilor fiecărui protocol se face în timpul funcționării acestuia.

2.4 Metode de implementare a semnalizărilor Cross-Layer

Semnalizările Cross-Layer prezentate anterior pot fi implementate în diverse aplicații folosind varii metode. Acestea se pot grupa în câteva categorii ce vor fi discutate în cadrul acestui

subcapitol. Trei dintre ele apar în lucrarea [Sri05] și sunt ilustrate în Figura 2-2, iar a patra a fost extrasă din lucrarea [Kha06] și este ilustrată în Figura 2-3.

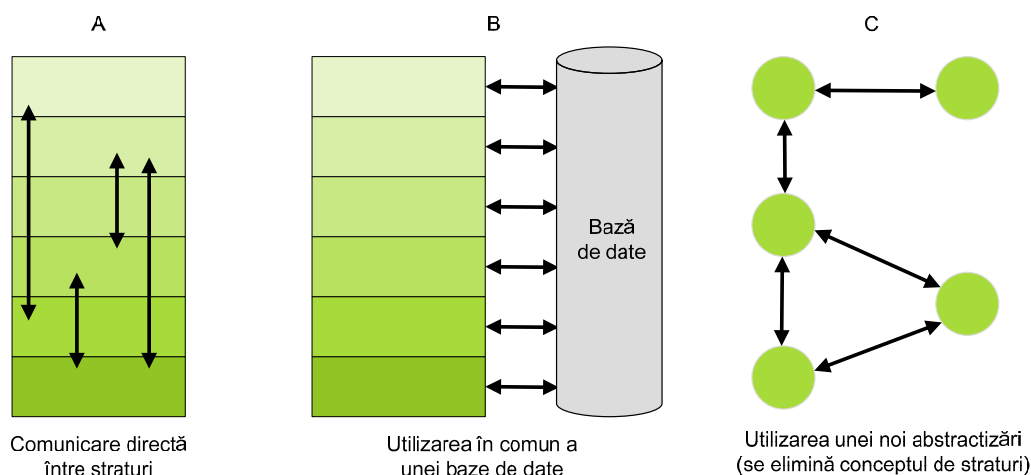


Figura 2-2 Metode folosite pentru implementarea semnalizărilor Cross-Layer [Sri05]

A. Comunicare directă între straturi

Această metodă de implementare este prezentată în Figura 2-2 A și permite comunicarea directă între straturi. Această abordare este optimă atunci când se implementează o metodă de semnalizare ce presupune crearea de noi interfețe între nivele diferite ale stivei de protocoale. Astfel, variabilele specifice unui anumit protocol vor fi făcute disponibile altora în timpul funcționării.

B. Folosirea în comun a unei baze de date

Cea de-a doua metodă de implementare a semnalizărilor Cross-Layer este prezentată în Figura 2-2 B și presupune folosirea în comun a unei baze de date, în care fiecare protocol ce rulează la diverse nivele, poate stoca sau citi date. Astfel, de fiecare dată când un protocol de comunicații colectează un set de indicatori specifici respectivului nivel, aceștia vor fi trimiși către baza de date respectivă.

C. Folosirea unei noi abstractizări

În Figura 2-2 C avem ilustrată o a treia metoda de implementare ce presupune renunțarea la noțiunea de strat și utilizarea unei noi abstracțiuni. Astfel, entitățile vor putea schimba informații între ele fără nici o restricție. Această metodă oferă o anumită flexibilitate proiectantului atât pe perioada implementării cât și în timpul execuției protocolului creat. Există totuși un dezavantaj și anume faptul că această metodă necesită o re-proiectare a întregii arhitecturi a Internetului. Astfel de abordări pot fi găsite în diverse proiecte, cum ar fi: ANA, 4WARD sau Ambient Networks.

D. Utilizarea unui modul dedicat de optimizare

Cea de-a patra și ultima metodă ce poate fi utilizată pentru implementarea tehnicilor Cross-Layer este ilustrată în Figura 2-3, fiind propusă de către autorii din [Kha06]. Ideea principală constă în utilizarea unui modul specializat de optimizare a tuturor protocoalelor de comunicare din stivă. Acest modul citește un set de parametri caracteristici fiecărui nivel și rulând un algoritm de optimizare, generează mai multe seturi (pentru fiecare protocol controlat) ce vor îmbunătăți performanțele întregului sistem.

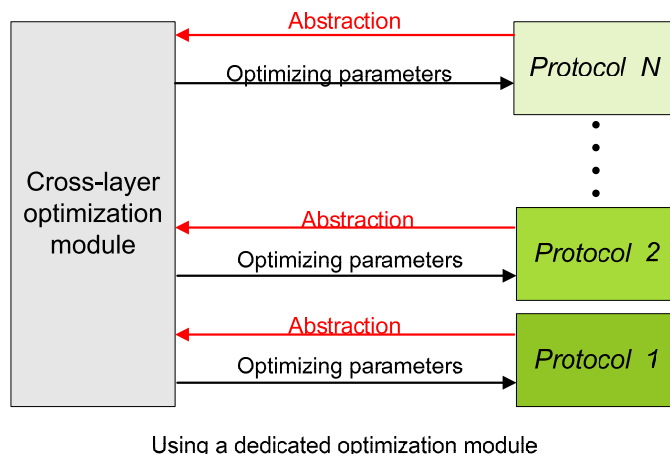


Figura 2-3 Metode de implementare a semnalizărilor Cross-Layer [Kha06]

2.5 Clasificarea implementărilor Cross-Layer

Cercetarea focalizată pe studiul mecanismelor Cross-Layer s-a maturizat de-a lungul timpului, ajungând să cunoască un set vast de implementări. Pentru asigurarea succesului acestor tehnici și în viitor, este necesară o clasificare a acestora din câteva puncte de vedere.

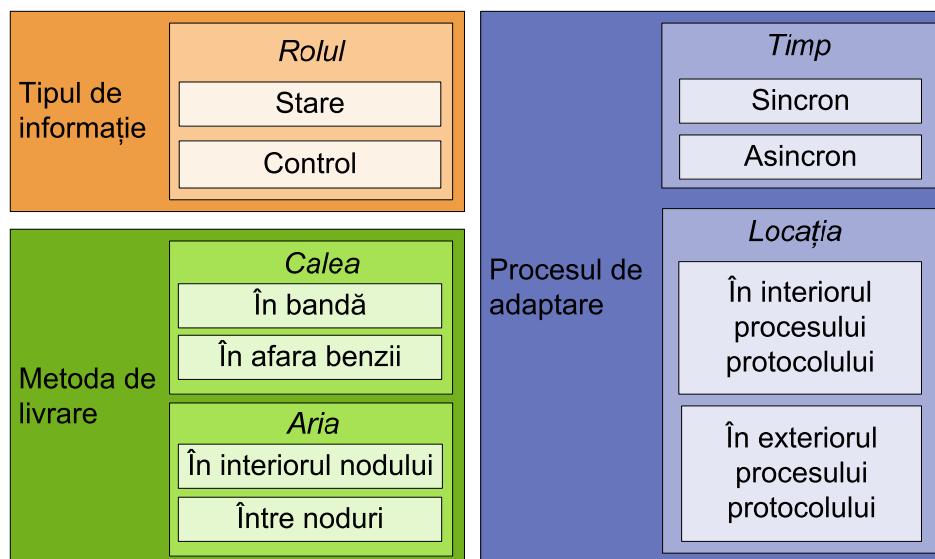


Figura 2-4 Clasificarea implementărilor Cross-Layer

2.6 Concluzii

În acest capitol s-a făcut o prezentare de ansamblu asupra tehnicilor Cross-Layer existente la ora actuală, ilustrând moduri în care acestea pot îmbunătăți calitatea serviciilor transmise prin intermediul rețelelor cu comutație de pachete. Ar fi o greșeală să susțin faptul că trebuie să se renunțe complet la modelele OSI și TCP/IP deoarece acestea nu au fost proiectate să facă față cerințelor actuale. Deși performanțele acestora nu sunt optime, ele pot fi îmbunătățite prin implementarea unora din tehnicile Cross-Layer descrise anterior. Chiar dacă pe viitor arhitectura Internetului va cunoaște o schimbare semnificativă, tehnicile Cross-Layer vor fi în continuare necesare deoarece acestea vor permite semnalizări nerestricționate între toate entitățile incluse.

Capitolul 3 Proiectarea și implementarea preliminară a unei arhitecturi Clean-Slate pentru Cross-Layer CLQ

3.1 Motivație

În acest capitol se face o descriere a modului în care se pot insera capabilități QoS într-o rețea ce implementează o arhitectură de management distribuit (INM). Partea principală a acestei secțiuni este dedicată ilustrării conceptelor de proiectare a modului de Cross-Layer QoS (CLQ). Acesta este astfel conceput încât să se integreze perfect într-o arhitectură ce regândește întreaga filozofie a Internetului [Nun09], oferind servicii de QoS prin tehnici Cross-Layer, tuturor entităților aflate în același nod sau în noduri diferite. Un alt aspect foarte important de menționat este faptul că modulul CLQ este caracterizat de următorul set de atribute: auto-configurare, interoperabilitate, compozabilitate și extensibilitate. În cea de-a doua parte a acestei secțiuni se ilustrează o implementare preliminară al modulului Cross-Layer QoS.

3.2 Principii de proiectare a modulului de Cross-Layer QoS (CLQ)

Previțiunile legate de rețeaua viitorului susțin faptul că aceasta va fi o infrastructură ce va oferi acces la o gamă variată de servicii: servicii de date, servicii de telefonie, transmisii video de înaltă definiție, transmisii video 3D etc. Deoarece nu toate tipurile de fluxuri au aceleași restricții de trafic, ele trebuie clasificate și tratate diferit atunci când sunt transportate în rețea. Ținta principală este de a reuși menținerea unui anumit nivel de QoS și QoE (Quality of Experience – calitatea serviciului experimentată de către utilizator), astfel încât să se respecte contractul încheiat între operator și client. Astfel, soluția pentru satisfacerea tuturor acestor cerințe este implementarea unui modul CLQ într-o arhitectura INM astfel încât toate aceste mecanisme QoS să se poată auto-configura fără intervenția unui operator uman.

În acest sens, modulul CLQ va conține două tipuri de interfețe: o interfață prin care vor fi transportate diverse servicii (ORG) și o interfață folosită pentru colaborarea între entități de același rang (COLL). De asemenea, acest modul va implementa trei tipuri de capabilități de management..

- *Monitorizarea resurselor de la nivelul fizic*: facilitează execuția de măsurători efectuate între nodurile direct conectate;
- *Calculul unei metrici compozite*: un indicator global ce abstractizează performanțele actuale a rețelei va fi calculat. Acesta înglobează o serie de parametri de trafic măsurați/primiți direct de la echipamentul de comunicații;
- *Tehnici Cross-Layer pentru QoS*: se referă la implementarea de tehnici de optimizare a unui protocol ce rulează într-un anumit strat, folosind informații caracteristice altui strat.

Interacțiunea dintre modulul de Cross-Layer QoS și alte entități specifice viitorului internet este prezentată în

Figura 3-1. Ținând cont de faptul că rețeaua viitorului va trebui să se poată autogestiona fără intervenția unui operator uman, modulul INM CLQ trebuie să poată fi caracterizat de următoarele proprietăți:

- *Auto-configurare*: modulul trebuie să fie capabil să pornească și să își determine automat parametrii optimi de funcționare;
- *Interoperabilitate*: CLQ trebuie să aibă posibilitatea de a comunica și cu alte entități, indiferent de producător;
- *Compozabilitate*: acest modul trebuie să fie poată crea la cerere noi servicii, prin combinarea mai multor servicii existente.
- *Extensibilitate*: modulul trebuie să suporte eventuale extinderi ale funcționalităților sale.

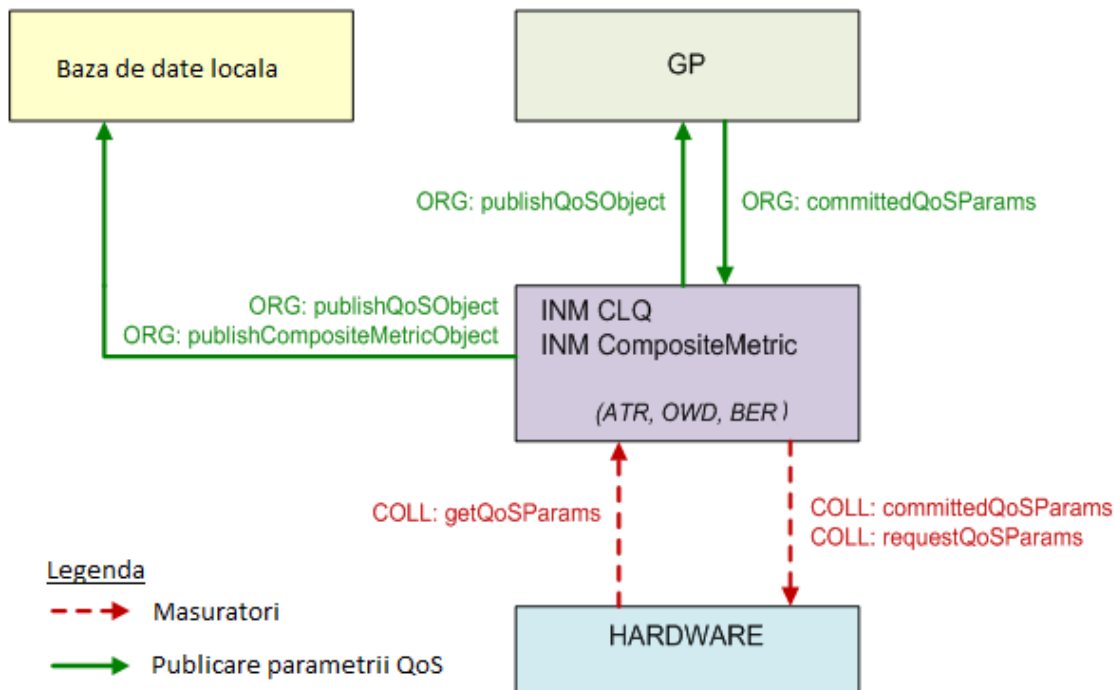


Figura 3-1 Interacțiunea dintre modulul CLQ și alte entități din interiorul unui nod

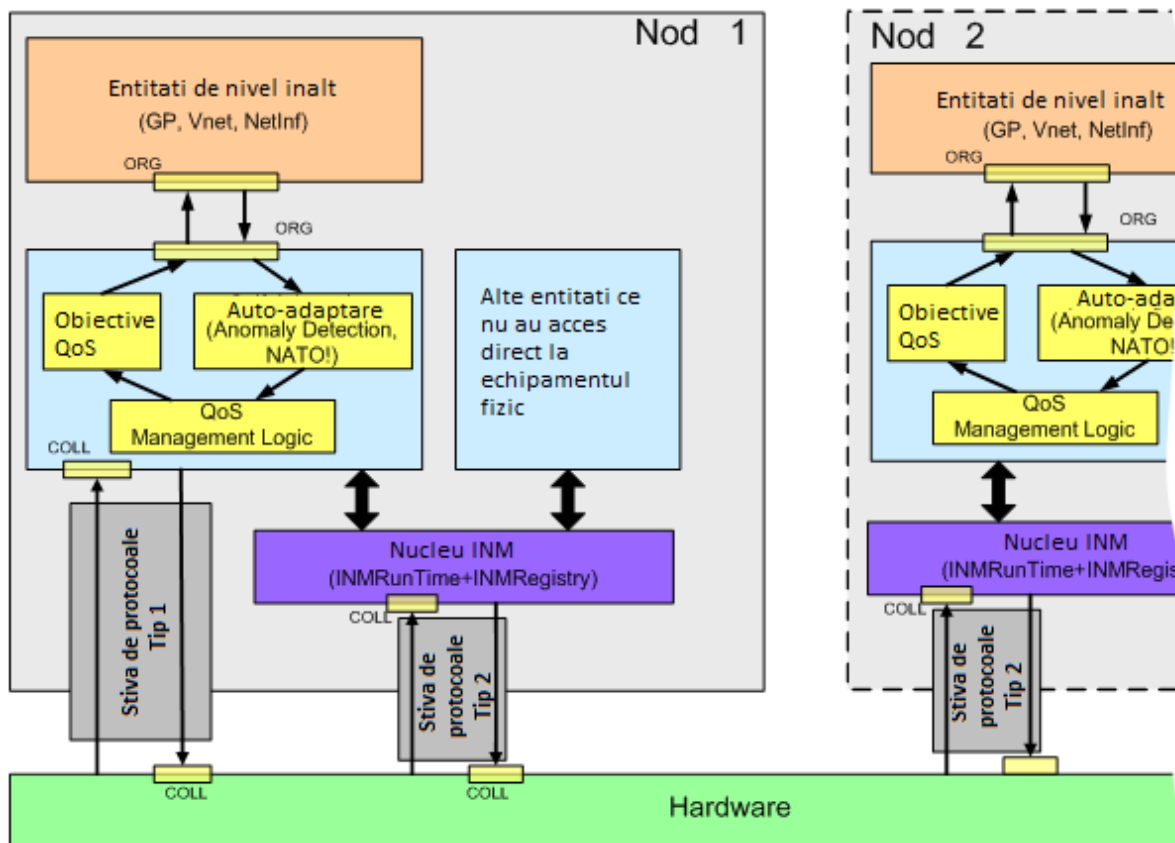


Figura 3-2 Integrarea modulului de Cross-Layer QoS în arhitectura INM

Modulul CLQ folosește două tipuri de stive de protocoale, așa cum apare în Figura 3.2. Stiva de tip 1 este dedicată comunicării directe cu echipamentul fizic (hardware). Aceasta este folosită pentru următoarele operații:

- a) Operații Cross-Layer de jos în sus folosite pentru efectuarea de măsurători deasupra substratului MAC. În acest sens se transmit probe personalizate de măsură pentru a monitoriza starea unei legături între două noduri adiacente;
- b) Operații Cross-Layer de sus în jos utilizate pentru alocarea globală a resurselor disponibile. De asemenea acestea pot ajuta la transmiterea datelor utilizatorilor, din nod în nod, în cazul în care protocoalele de transmisie experimentează probleme de funcționare.

Cea de-a doua stivă de protocoale poate include oricare dintre protocoalele clasice de comunicare și este necesară pentru publicarea informațiilor de management altor entități cu care colaborează modulul CLQ și care se pot afla în același nod sau în alte noduri ale rețelei [Gon10].

Dacă o anumită entitate (ex. GP – Generic Path – entitate dedicată transmisiei fiabile a datelor în viitorul Internet proiectat în cadrul proiectului 4WARD) necesită impunerea unui set de parametri QoS către infrastructura rețelei, modulul INM CLQ va primi respectiva informație pe interfața ORG, iar acesta va transmite mai departe un mesaj de management în format XML direct către echipamentul fizic (se consideră că pe viitor acesta va fi capabil să interpreteze astfel de mesaje de control) folosind interfața COLL. Implementând aceste operații, funcționalitatea echipamentelor fizice va fi controlată astfel încât să răspundă necesităților existente.

3.3 Principii de implementare a modului Cross-Layer QoS

Această secțiune este dedicată ilustrării unei implementări preliminare a modului INM CLQ care va fi caracterizat doar de un subset al proprietăților prezentate anterior în acest capitol. Motivul pentru o astfel de abordare constă în faptul că primul obiectiv este demonstrarea fezabilității conceptului, urmând ca o dată stabilit acest lucru, în cea de-a doua fază să se treacă la o implementare completă a acestuia.

Datorită faptului că un anumit utilizator ar putea fi interesat de evoluția a diverși parametri monitorizați în rețea, în fiecare nod se poate activa funcționarea unui modul de interfață grafică care să prezinte la cerere graficul actualizat în timp real a parametrilor de interes. Datele prezentate de către modulul de interfață grafică (GUI – Graphical User Interface), sunt prelevate dintr-o bază de date locală în care informațiile sunt populate de către modulul CLQ. De menționat este faptul că aceste date pot reprezenta informații monitorizate pe legături care conțin atât nodul curent cât și alte legături din rețea. O ilustrare a unei astfel de arhitecturi poate fi analizată în Figura 3-3.

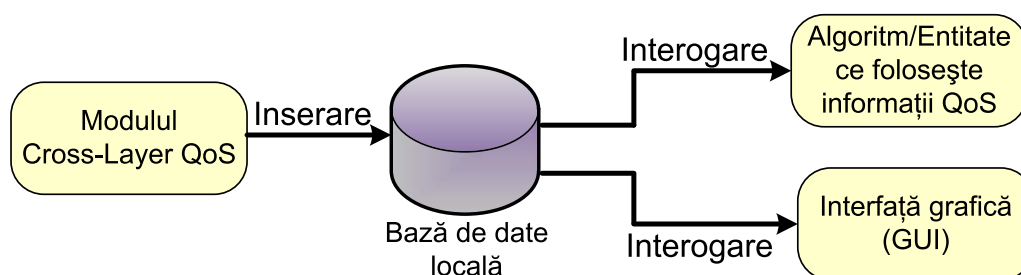


Figura 3-3 Arhitectura de comunicare între CLQ, GUI și alte entități

Pe lângă modul de colaborare ilustrat mai sus, orice entitate poate beneficia de informații oferite de către modulul INM CLQ cu ajutorul tehnicii cunoscută sub numele de Publisher-Subscriber (furnizor de informații - abonat), ilustrat în Figura 3-4. Astfel respectiva entitate se va înscrie în lista de abonați a modului CLQ, iar în momentul în care parametrii furnizați de către acesta vor fi actualizați, va fi informat în timp real de către acesta. Astfel, abonatul va putea interoga noile valori [Nun10].

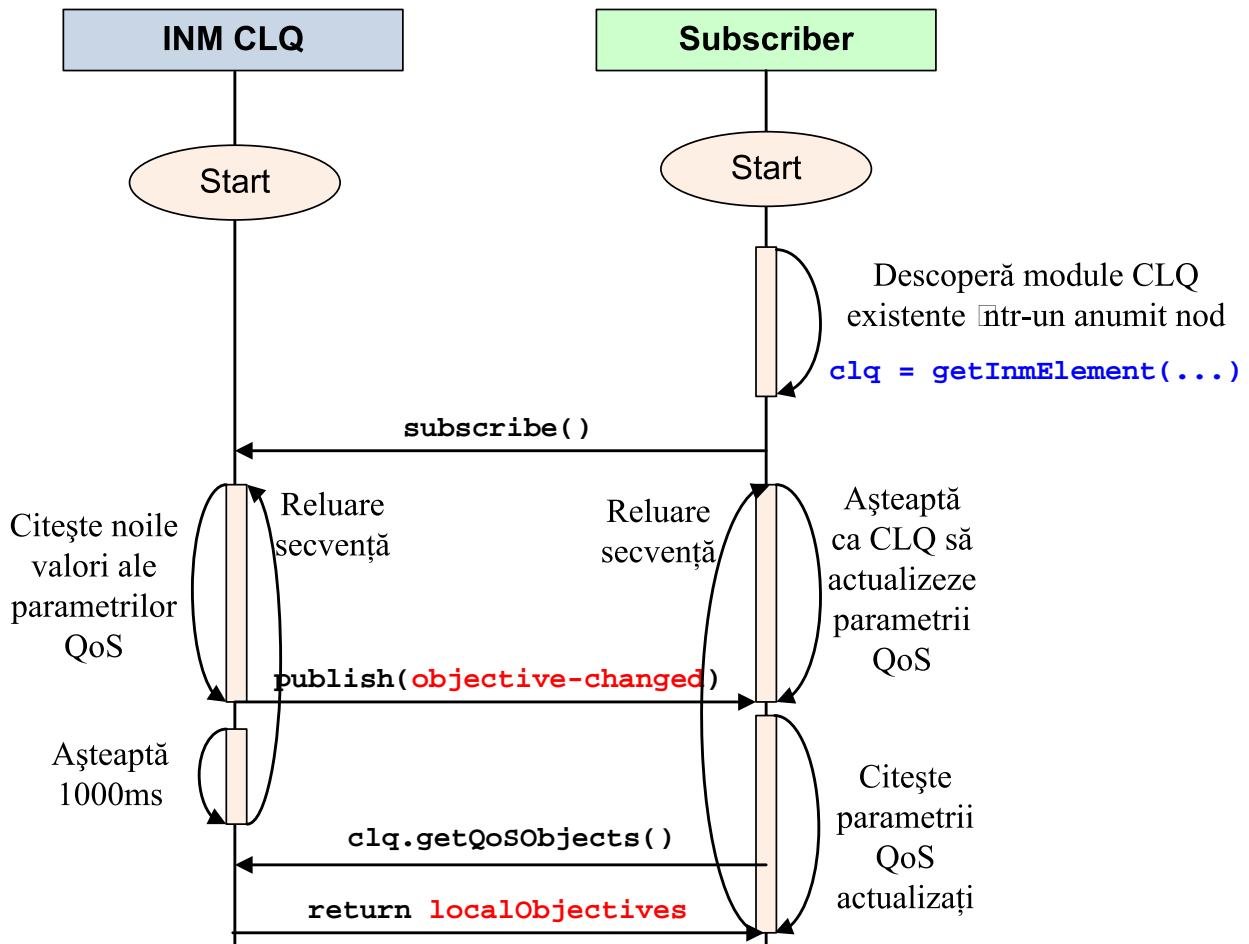


Figura 3-4 Comunicarea de tip Publisher-Subscriber între INM CLQ și o alte entitate

3.4 Concluzii

Munca prezentată în acest capitol este focalizată în principal pe o serie de metode de includere a mecanismelor de calitate a serviciilor (QoS) într-o arhitectură ce implementează un management distribuit în rețea. Principala trăsătură a modului de INM CLQ este capacitatea acestuia de a oferi în timp real informații legate de parametrii de trafic monitorizați în rețea. Bazat pe aceste informații s-ar putea calcula un indicator global al stării actuale a rețelei.

Desigur că au existat mai multe proiecte care să se ocupe de studiul mecanismelor de tip Cross-Layer pentru viitorul Internet (ex. Ambient Networks, ANA, SENSEI) dar ideile prezentate în acest capitol se axează pe o implementare care să fie compatibilă cu proiectul 4WARD. Astfel, modulul CLQ a fost proiectat de la zero, compatibil cu o arhitectură de management distribuit și care să permită colaborarea cu toate entitățile prevăzute a fi active în Internetul viitorului.

Capitolul 4 Metodologii propuse pentru măsurarea parametrilor QoS

4.1 Motivație

Primul tip de semnalizări Cross-Layer implementat de către modulul CLQ este cel de jos în sus, în care parametrii de la straturi inferioare (ex. stratul Fizic + substratul MAC = Hardware) sunt făcuți disponibili protocoalelor sau altor entități ce rulează la straturi superioare. Astfel, în acest capitol vor fi ilustrate a unui set de utilitare ce au ca scop efectuarea de măsurători în timp real a parametrilor de trafic. Principala contribuție inclusă în această secțiune este cea de proiectare și implementare reală programelor de măsură. De menționat este faptul că aceste aplicații vor fi apelate atunci când informațiile de trafic nu vor putea fi interogate direct de la echipamentul fizic.

4.2 Măsurarea parametrului întârziere dus-întors (Round-Trip Time – RTT)

Când se dorește efectuarea de măsurători de întârziere într-o topologie în care nodurile nu au ceasurile interne sincronizate, parametrul ce se pretează a fi monitorizat este întârzierea dus-întors. Deoarece scopul este de a monitoriza parametrii de trafic cât mai aproape de echipament, probele de măsură au fost transmise deasupra substratului MAC. Pentru aceasta s-a folosit un tip special de socket și anume Raw Socket. Astfel, probe personalizate ce au structura ilustrată în Figura 4-1, au putut fi transmise pe legăturile monitorizate.

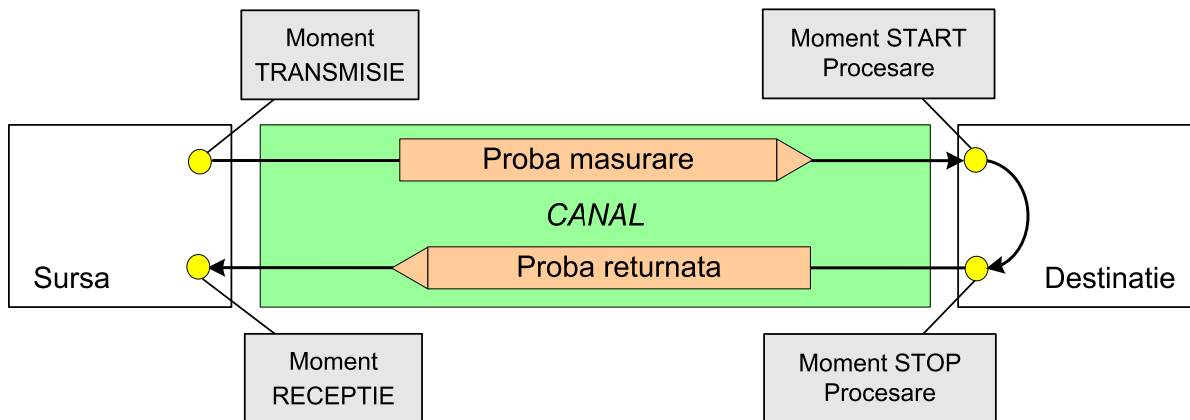


Figura 4-1 Măsurarea întârzierii dus-întors

Proba trimisă pe canalul monitorizat este ilustrată în Figura 4-2.

6 octeti	6 octeti	2 octeti	8 octeti	8 octeti	8 octeti	22 octeti	4 octeti
Adresa MAC Dest.	Adresa MAC Src.	Tip	Moment TRANSMISIE	Moment START Procesare	Moment STOP Procesare	Octeti Nefolositi	CRC32
Payload							

Figura 4-2 Proba de măsură a întârzierii dus întors

Chiar și în cazul în care echipamentele sursă și destinație nu au ceasurile interne sincronizate, pe lângă calculul întârzierii dus întors (excluzând timpul de procesare la destinație), se poate estima valoarea latenței corespunzătoare unei căi folosind timpii ilustrați mai sus. Procedura constă în eliminarea timpului destinat procesării la nodul destinație și împărțirea rezultatului la 2, așa cum poate fi observat în următoarea formulă.

- T_s_SENT = Moment TRANSMISIE = momentul în care a fost trimisă proba de la sursă
- $T_s_START_Proc$ = Moment START Procesare = momentul în care proba a ajuns la destinație
- $T_s_STOP_Proc$ = Moment STOP Procesare = momentul în care proba a fost emisă de către nodul destinație
- $T_s_RECEIVED$ = Moment RECEPȚIE = momentul la care proba a ajuns înapoi la nodul sursă

$$Delay = \frac{(T_s_RECEIVED - T_s_SENT) - (T_s_STOP_Proc - T_s_START_Proc)}{2} [s] \quad (4.1)$$

4.3 Măsurarea parametrului Latență (One-Way Delay – OWD)

În cazul în care nodurile din topologia monitorizată au ceasurile interne sincronizate (folosind de exemplu un echipament de sincronizare prin GPS), cel mai potrivit parametru temporar ce se poate monitoriza este latența (OWD – One Way Delay). Astfel, pe ambele noduri implicate în măsurătoare se va rula un program specializat ce va implementa procedura din Figura 4-3 și va utiliza probe personalizate, a căror structură este ilustrată în Figura 4-4.

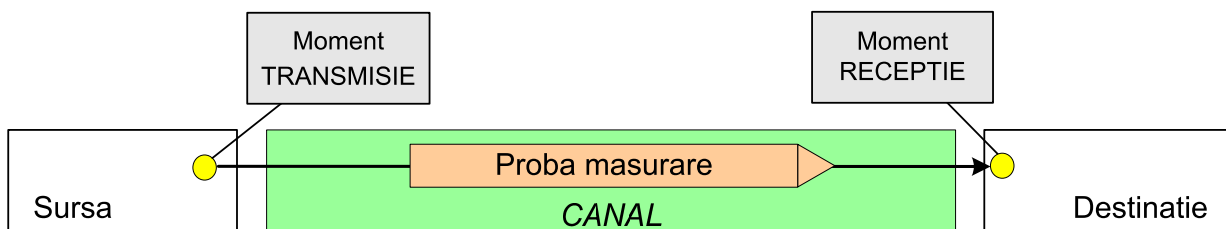


Figura 4-3 Procedura de măsurare a latenței

6 octeti	6 octeti	2 octeti	8 octeti	38 octeti	4 octeti
Adresa MAC Dest.	Adresa MAC Src.	Tip	Moment TRANSMISIE	Octeti Nefolositi	CRC32
			Payload		

Figura 4-4 Probă pentru măsurarea latenței

De îndată ce proba este primită la destinație, se marchează momentul sosirii și se calculează latența făcând diferența între momentul curent și momentul transmisiei.

4.4 Măsurarea parametrului rată de transfer disponibilă (Available Transfer Rate – ATR)

Pentru monitorizarea parametrului ATR se folosesc tehnici pasive de măsură. Procedura implementată constă în următorii pași:

1. se monitorizează traficul recepționat pe o anumită interfață
2. știind valoarea capacității canalului, se face diferența dintre capacitate și cantitatea de trafic existent pe interfața monitorizată, calculându-se astfel rata de transfer disponibilă pe respectivul link.

4.5 Estimarea parametrului Probabilitate de eroare pe bit (Bit Error Rate – BER)

Atunci când se transmit date pe canale radio, un parametru ce influențează direct calitatea serviciilor este BER. Este importantă cunoașterea acestui indicator deoarece exprimă într-un mod foarte sugestiv calitatea mediului de transmisie. Dacă se cunoaște raportul semnal zgomot (SNR) împreună cu tipul de modulație folosit, utilizând modele matematice corespunzătoare plus un set de aproximații putem estima care ar fi valoarea parametrului BER. Tehnologiile considerate în acest capitol sunt 802.11 a/b/g și xDSL. Pentru alte tehnologii existente la nivelul stratului fizic, modele matematice corespunzătoare trebuie utilizate.

Analizând standardul 802.11 a/b/g, se poate observa faptul că pentru fiecare rată de transmisie se utilizează modulații diferite (vezi Tabelul 4-1). Ratele de transfer considerate sunt cele care folosesc tehnica de transmisie OFDM (Orthogonal frequency-division multiplexing).

Modulația	Rata de transfer [Mbps]
BPSK	1.5, 6, 9
QPSK	3, 4.5, 11, 12, 18
16-QAM	24, 36
64-QAM	27, 48, 54

Tabelul 4-1 Modulații folosite de către tehnologia 802.11 a/b/g

Modelul matematic utilizat folosește câteva aproximații. Prima constă în considerarea zgomotului alb (Additive White Gaussian Noise – AWGN) ca fiind singurul factor ce perturbă canalul. Cea de-a doua aproximare constă în faptul că vom considera că pe fiecare subpurtătoare OFDM avem același raport semnal zgomot și este egal cu raportul semnal zgomot pentru întreg simbolul OFDM. Deoarece parametrul de intrare este raportul semnal zgomot (SNR), va trebui să îl folosim în modelele de calcul utilizate în continuare. Astfel voi exprima raportul E_b/N_0 (necesar pentru prelucrarea modelelor matematice) ca o funcție de SNR.

$$\frac{E_b}{N_0} = \frac{10^{SNR/10}}{n} \quad (4.2)$$

După procesarea modelelor matematice din [Rap02], am ajuns la următoarele expresii de calcul a parametrului BER pentru fiecare sub-canal OFDM în funcție de modulație:

$$BER_{BPSK} = \frac{1}{2} \operatorname{erfc}\left(\sqrt{10^{SNR/10}}\right) \quad (4.3)$$

$$BER_{QPSK} = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{10^{SNR/10}}{2}} \right) \quad (4.4)$$

$$BER_{N-QAM} = \frac{2 \left(1 - \frac{1}{\sqrt{N}} \right) \cdot \operatorname{erfc} \left(\sqrt{\frac{3 \cdot 10^{SNR/10}}{2 \cdot (N-1)}} \right)}{\log_2 N} \quad (4.5)$$

Cea de-a doua tehnologie considerată pentru care se propune un model de estimare a parametrului BER este xDSL (vezi Figura 4-5).

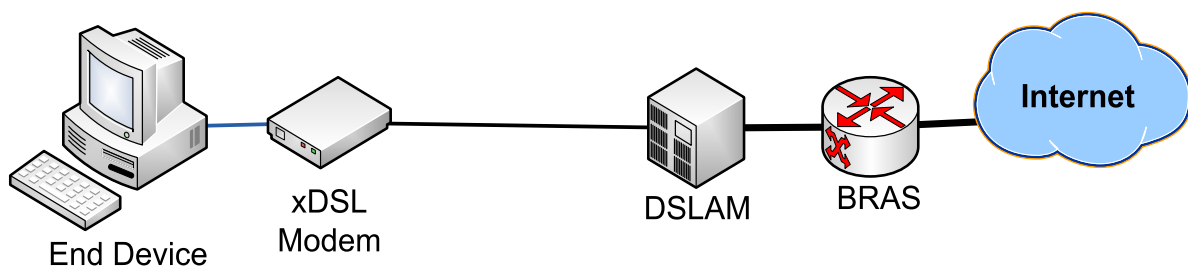


Figura 4-5 Topologie xDSL generică

Transmisia de date pe o conexiune xDSL este făcută pe un set de sub-canale cunoscute sub numele de tonuri. Pe fiecare ton se utilizează în funcție de caracteristicile canalului un anumit tip de modulație. Setul de modulații din care se vor alege cele folosite, este următorul: BPSK, QPSK, 8-QAM, 16-QAM, 32-QAM, 64-QAM, 128-QAM și 256-QAM. Pentru estimarea parametrului BER pentru întreaga transmisie se pot folosi următoarele formule matematice:

$$BER_{xDSL} = 1 - P_0 \quad (4.6)$$

$$P_0 = \prod_{i=1}^T (1 - BER_{tone_i}) \quad (4.7)$$

Unde BER_{tone_i} este rata de eroare pe bit calculată pentru tonul i corespunzător transmisiei xDSL, calculată folosind una dintre ecuațiile (4.3), (4.4) or (4.5). P_0 este probabilitatea ca să nu avem eroare pe nici un ton xDSL.

4.6 Concluzii

În acest capitol au fost propuse un set de utilitare destinate monitorizării parametrilor QoS. Aceștia au fost mășurați folosind atât metode active (OWD și RTT) cât și pasive (ATR și BER). Tehnologiile considerate au fost: Ethernet, xDSL și 802.11 a/b/g. Deși la prima vedere aceste utilitare par a fi doar niște soluții particularizate, ele pot fi generalizate pentru a lucra cu diverse tehnologii deoarece scopul principal a fost efectuarea de măsurători deasupra substratului MAC. Din același motiv, contribuția prezentată în acest capitol rămâne valabilă chiar și pe viitor când se preconizează că se va renunța la stiva TCP/IP.

Capitolul 5 Implementarea și evaluarea tehnicilor Cross-Layer pentru controlul congestiei

5.1 Motivație

În această secțiune se va prezenta o implementare a unui modul Cross-Layer QoS pentru controlul congestiei într-o rețea cu comutație de pachete. Această soluție este potrivită atunci când nu se mai pot adăuga resurse adiționale în rețea. Implementarea constă în activarea unor module CLQ dedicate monitorizării parametrilor QoS. Această informație este mai apoi procesată pentru a se determina starea actuală a rețelei, indicând astfel apariția congestiei. Odată cu detecția acestui fenomen, modulul CLQ va controla activarea unui mecanism de codare a rețelei, cunoscut sub numele de Network Coding (NC), care este de fapt o tehnică de rutare distribuită utilizând tehnici de colaborare.

5.2 Arhitectura modului de Cross-Layer QoS

Modulul Cross-Layer QoS (CLQ) propus în această secțiune folosește atât semnalizări de jos în sus, cât și de sus în jos, schimbate între echipament și alte entități, transportând atât semnalizări de stare cât și de control, transmise în bandă cât și în afara benzii, iar procesul de adaptare este asincron. Modulul CLQ va funcționa continuu pentru a monitoriza infrastructura, acceptând de asemenea și cereri din partea unor module ce rulează la nivele superioare. În cazul semnalizărilor **de jos în sus**, tehnicile Cross-Layer vor colecta informații de stare specifice legăturilor fizice. Acestea vor fi obținute direct de la echipamentul fizic unde este posibil, sau de la programe specializate atunci când este cazul, utilizând atât tehnici de măsurare pasive cât și active. În cazul semnalizărilor **de sus în jos**, tehnicile Cross-Layer sunt folosite pentru controlul direct a diverselor entități: module specializate în transportul fiabil a traficului, echipamentul fizic etc.

O altă caracteristică a modului CLQ este aceea de elaborare a unui indicator global a stării actuale a rețelei denumit metrică compozită (CM). Formula utilizată pentru calculul acestui parametru este următoarea:

$$CM = \frac{k_0}{ATR[bps]} + \frac{OWD[s]}{k_1} + k_2 \times BER \quad (5.1)$$

unde $k_0 = 10^9$ [bps], $k_1 = 10^{-5}$ [s] și $k_2 = 10^{12}$. CM poate fi utilizată ca și criteriu de activare a mecanismului de codare a traficului din rețea.

5.3 Folosirea serviciilor modului CLQ de către modulul NC pentru controlul congestiei

Tehnicile de codare a traficului din rețea (NC – Network Coding) sunt o metodă de a face față congestiei prin combinarea de fluxuri de date ce partajează aceeași legătură [Ahl00], [Koe03]. Topologia cea mai simplă în care se pot evidenția mecanismele NC este cea de tip fluture reprezentată în Figura 5-1. Dacă nodul R1 are de trimis date către nodul R4 și nodul R2 trimite date către R3 în același timp, iar în rețea nu există congestie, fluxurile de date vor folosi căile: R1-R5-R6-R4 și respectiv R2-R5-R6-R3 fără probleme. De asemenea, este posibilă transmisia fiabilă chiar

și în condiții de congestie pe legătura comună (R5-R6) dacă cele două fluxuri sunt codate folosind operatorul XOR și se trimit de asemenea fluxuri adiționale pe legăturile R1-R3 și R2-R4.

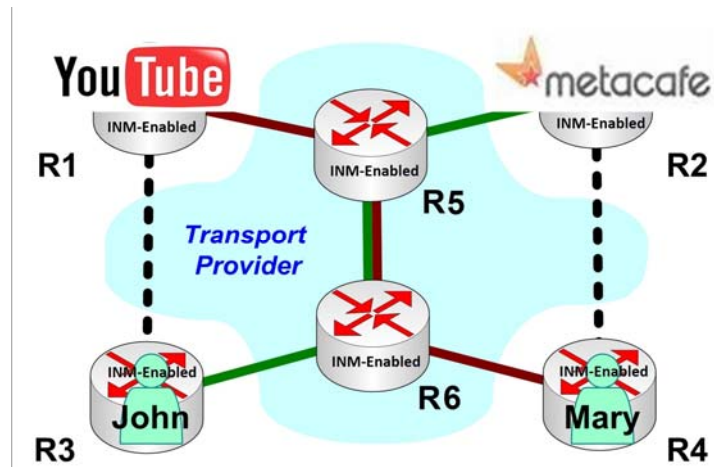


Figura 5-1 Topologie fluturi asimetrică cu șase noduri: fără NC

Chiar și în cazul celor mai simple tehnici NC, există probleme de rezolvat. Acestea sunt legate de sincronizarea fluxurilor și găsirea momentului propice activării/ dezactivării acestora. Cea de-a doua operație este necesară eficientizării modului de utilizare a resurselor rețelei. Pentru a rezolva aceste provocări, este necesară implementarea unui proces de codare dinamic care are următoarele caracteristici:

- Codarea se face doar asupra pachetelor care generează congestia la un moment dat. Dacă pe o anumită cale nu se transmit date, pachetele celuilalt flux vor fi transmise necodat.
- Dacă lungimea pachetelor nu este aceeași, se va coda doar partea din pachet egală cu dimensiunea minimă dintre cele două.

5.4 Rezultate experimentale

Două fluxuri video sunt transmise pe căile: R1-R5-R6-R4 și R2-R5-R6-R3 de către utilitarele VLC ce rulează pe nodurile R1 și R2. Acestea vor folosi în comun legătura R5-R6 care poate experimenta congestie în cazul în care fluxurile de date transmise depășesc capacitatea canalului.

Evaluarea performanțelor mecanismului implementat se bazează pe evaluarea următorului set de parametri măsurat pentru fiecare flux video:

- a) Rata de transfer măsurată în timp real;
- b) Numărul de pachete pierdute măsurat per sesiune.

Arhitectura sistemului implementat pentru efectuarea testelor este ilustrată în Figura 5-2. Trei cazuri de test au fost considerate după cum urmează:

- *Cazul 1 (nu există congestie)*: deoarece rata de transfer disponibilă pe legătura R5-R6 este destul de mare pentru a suporta traficul existent, calitatea fluxurilor video la destinație este foarte bună. Forma fluxului video ce traversează rețeaua prin secțiunea R2-R5-R6-R3 poate fi vizualizată în Figura 5-3.
- *Cazul 2 (există congestie dar mecanismul NC nu este activat)*: pe legătura partajată R5-R6 apare congestie datorită faptului că rata de transfer disponibilă este sub valoarea necesară transportării celor două fluxuri video în paralel. Astfel, ambele noduri destinație R3 și R4 experimentează o calitate scăzută a fluxului video recepționat. Modul în care fluxul video este degradat poate fi vizualizat în Figura 5-4.

- *Cazul 3 (există congestie iar mecanismul NC este activat):* conexiunea R5-R6 experimentează congestie ca și în cazul anterior dar mecanismul de NC a fost activat de către modulul CLQ. Rezultatul poate fi vizualizat în Figura 5-5.

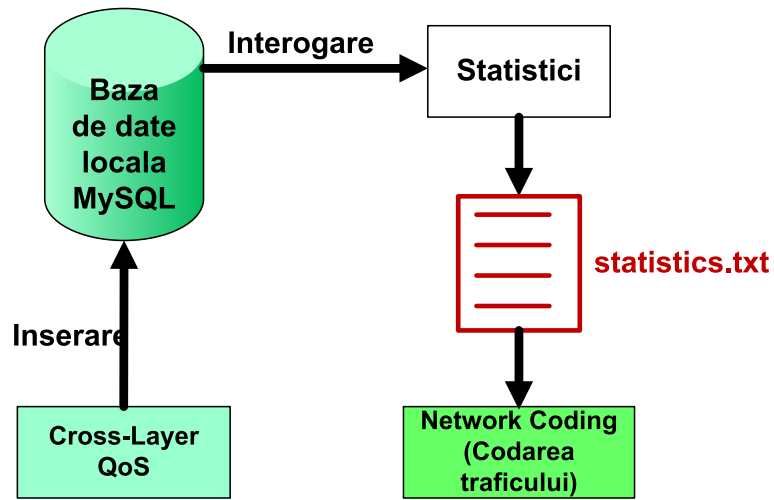


Figura 5-2 Arhitectura sistemului implementat pentru evitarea a congestiei

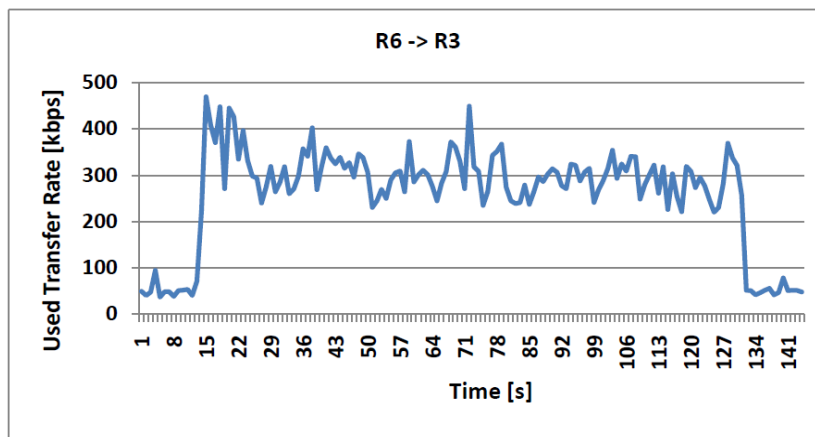


Figura 5-3 Forma fluxului video pe legătura R6-R3 în cazul 1
(Copyright © 2010 IEEE [Rus10a])

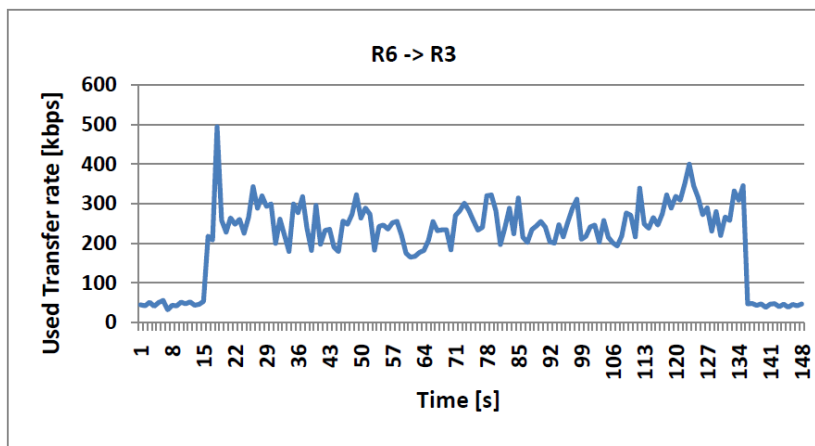


Figura 5-4 Fluxul video recepționat de nodul R3 în cazul 2
(Copyright © 2010 IEEE [Rus10a])

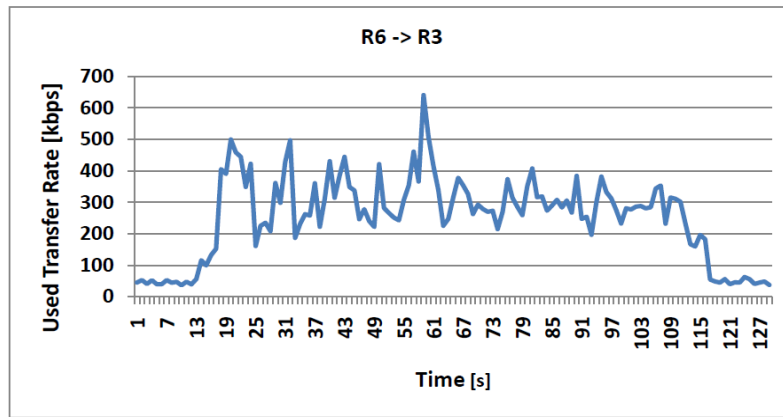


Figura 5-5 Fluxul video recepționat de nodul R3 în cazul 3
(Copyright © 2010 IEEE [Rus10a])

Măsurătorile prezentate în Figura 5-6 ilustrează faptul că o dată ce se implementează în rețea un mecanism CLQ împreună cu un mecanism de NC, se obține o scădere a numărului de pachete pierdute, de la aproximativ 20% până la o valoare sub 1% așa cum apare și în Figura 5-6.

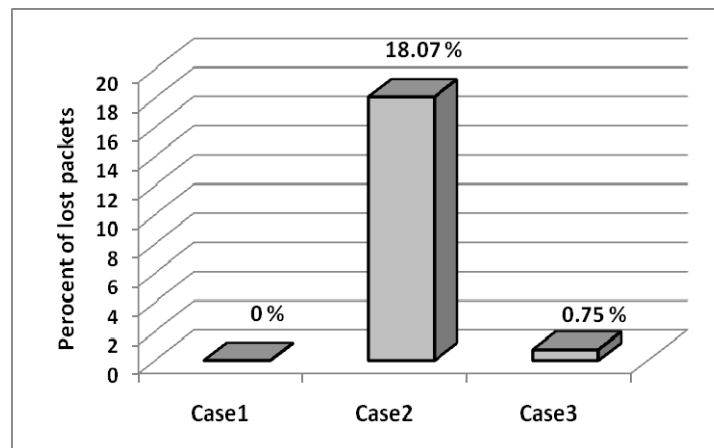


Figura 5-6 Procentul de pachete pierdute în cele trei cazuri implementate
(Copyright © 2010 IEEE [Rus10a])

5.5 Concluzii

Din punctul de vedere a calității experimentate de către utilizatorul final, atunci când în rețea s-au activat mecanismele de evitare a congestiei, performanțele măsurate au fost net superioare. Datorită implementării modulului de NC, fluxurile video transmise au fost limitate la formatul QCIF (rezoluție = 176x144). O altă caracteristică a acestei implementări este aceea că modulul de Cross-Layer QoS a executat măsurători continue deasupra substratului MAC la un interval de o secundă, astfel încât să se permită un timp de reacție scurt la orice eveniment neașteptat.

Prototipul prezentat în acest capitol este unul ce se adresează operatorilor de telecomunicații în momentele în care nu este posibilă adăugarea de noi resurse în infrastructura gestionată, sau când o altă soluție mai complexă (ex. rutare bazată pe QoS) nu este disponibilă.

Capitolul 6 Implementarea și evaluarea CLQ pentru rutarea conștientă de QoS

6.1 Motivație

Un dezavantaj comun protocoalelor de rutare din ziua de azi este faptul că nu țin cont de starea actuală a rețelei atunci când determină calea optimă spre o destinație. Deși acest lucru este considerat a nu fi necesar, în deosebi în rețelele fixe, este o eroare să facem abstracție de variabilitatea canalelor de comunicații deoarece acestea își pot schimba caracteristicile în funcție de traficul existent. Indicatorii QoS considerați în acest capitol, datorită faptului că ilustrează într-un mod obiectiv starea rețelei, sunt: rata de transfer disponibilă (Available Transfer Rate - ATR) și latența (One-Way Delay - OWD). Aceștia vor fi utilizați pe post de parametri de intrare de către protocolul de rutare folosit în rețea.

6.2 Utilizarea parametrilor QoS

Conceptul de utilizare/conștientizare a parametrilor QoS trebuie interpretat ca și capacitatea unei entități/protocol de a sesiza starea actuală a rețelei utilizând un set de indicatori QoS: ATR și OWD. Aceștia caracterizează într-un mod obiectiv performanțele diverselor legături din topologia monitorizată. Abordarea aleasă constă în următorii pași:

1. Măsurarea permanentă a legăturilor dintre oricare două noduri;
2. Publicarea rezultatelor către toate nodurile domeniului;
3. Recepția măsurătorilor publicate de alte entități.

Astfel, performanțele globale ale unei căi dintre o sursă și o destinație pot fi determinate utilizând informațiile referitoare la toate legăturile care o compun [Rus10d]. Pentru a implementa cele de mai sus, este nevoie ca în fiecare nod din domeniu să fie activat un modul Cross-Layer QoS (CLQ). Acesta este compus din două entități (vezi Figura 6-1): un modul specializat în efectuarea de măsurători (Modul măsurare CLQ) și un modul dedicat publicării și recepționării de informații QoS (Modul de publicare CLQ).

6.3 Utilizarea mecanismului CLQ pentru rutare

Pentru implementarea operațiilor specifice rutării, a fost nevoie de implementarea a unui set de două module: Modul de detecție a rutelor și un Manager tabelă de rutare (vezi Figura 6-1). Modulul de detecție a căilor are următoarele funcții: descoperirea topologiei corespunzătoare domeniului în care acesta rulează și determinarea căilor existente între două noduri oarecare. Această informație este trimisă mai departe celui de al doilea modul (Manager tabelă de rutare) care are rolul de a determina cea mai bună cale dintre toate posibile. Criteriul de decizie este metrica compozită (Composite Metric – CM) a cărei expresii poate fi studiată în ecuația (6.1) și care încorporează parametrii ATR și OWD.

$$CM = \frac{k_0}{ATR[bps]} + \frac{OWD[s]}{k_1} \quad (6.1)$$

Constantele k au următoarele valori: $k_0 = 10^9$ [bps], $k_1 = 10^{-5}$ [s]. După o analiză detaliată a expresiei anterioare, se observă faptul că valoarea metricii compozite este minimă atunci când rata de transfer disponibilă este de 1 Gbps, iar latența are valoarea de 10 microsecunde. Astfel, ruta care este caracterizată de valoarea minimă a metricii este considerată a fi ruta cea mai bună. De asemenea același modul va manipula înregistrările existente în tabela de rutare, prin operații de ștergere și adăugare de rute.

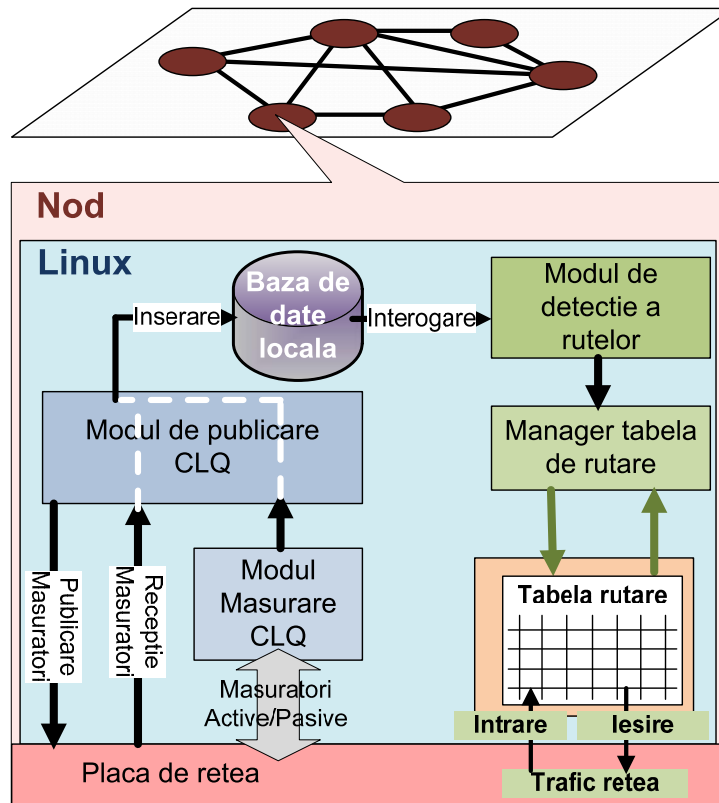


Figura 6-1 Arhitectura sistemului de rutare conștientă de QoS

Întregul proces de: măsurare, publicare parametri QoS, recepție parametri QoS, detecție a rutelor optime și actualizare a tabelii de rutare, este repetat în fiecare secundă. De menționat că acest interval poate fi modificat dacă este cazul, în funcție de dinamicitatea tehnologiei curente. Astfel, în cazul unei rețele hibride ce conține și conexiuni radio, datorită dinamicității acestui mediu, intervalul între două iterații poate fi micșorat. Pe de altă parte, dacă rețeaua are o comportare stabilă în timp, intervalul între două iterații consecutive poate fi mărit astfel încât să se eficientizeze utilizarea resurselor de calcul corespunzătoare nodurile rețelei.

6.4 Testarea performanțelor soluției propuse

În această secțiune este prezent un scenariu în care se pot evidenția avantajele implementării rutării conștiente de QoS (vezi Figura 6-2). Actorii principali ai scenariului considerat sunt următorii: Sursa A, Destinația A, Ferma de servere și Destinația B. Scenariul constă în următorii pași:

1. În prima fază, Sursa A trimite un flux de date de 40 Mbps către Destinația A.
2. La un moment dat, un al doilea flux de date agregat, de 80 Mbps pornește dinspre Ferma de servere către Destinația B.

În cazul în care rutarea în respectivul domeniu se face cu ajutorul protocoalelor clasice (ex. RIP sau OSPF), se va încerca transmisia ambelor fluxuri pe calea R1-R2-R5. Ținând cont de faptul

că ruta utilizată este compusă din secțiuni FastEthernet (100 Mbps) iar traficul ce se încearcă a fi transmis are o valoare totală de 120 Mbps, cele două fluxuri vor experimenta fenomenul de congestie. Consecința va fi o degradare semnificativă a datelor recepționate, datorită faptului că unele secțiuni de rețea sunt supra utilizate în timp ce altele rămân nefolosite. În cazul în care se activează în rețea un protocol de rutare conștient de QoS, efectul va fi unul benefic deoarece se va încerca evitarea congestie prin găsirea de noi căi ce pot fi utilizate pentru transmisia datelor.

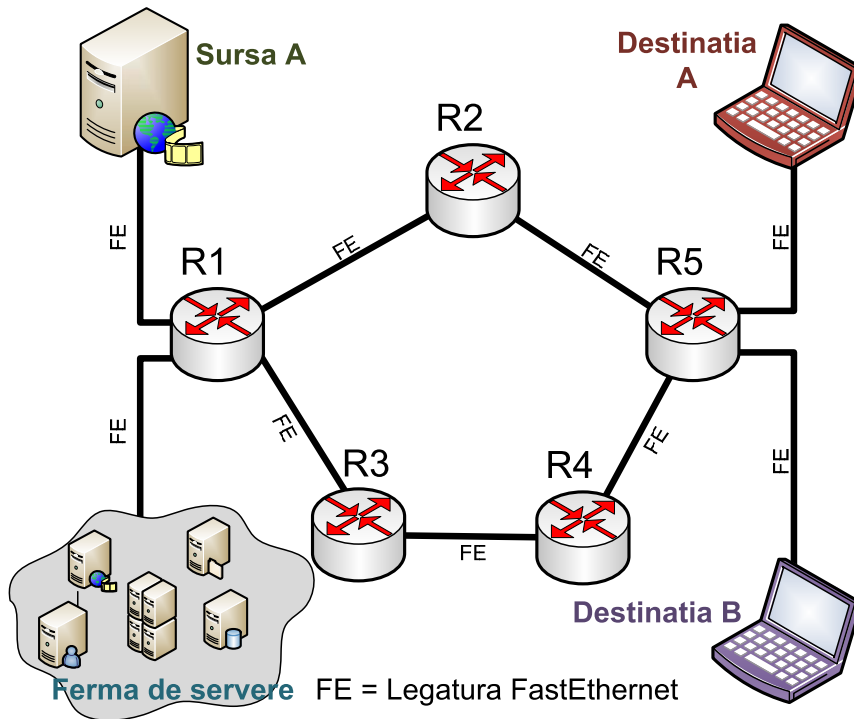


Figura 6-2 Scenariul utilizat pentru ilustrarea avantajelor rutării conștiente de QoS

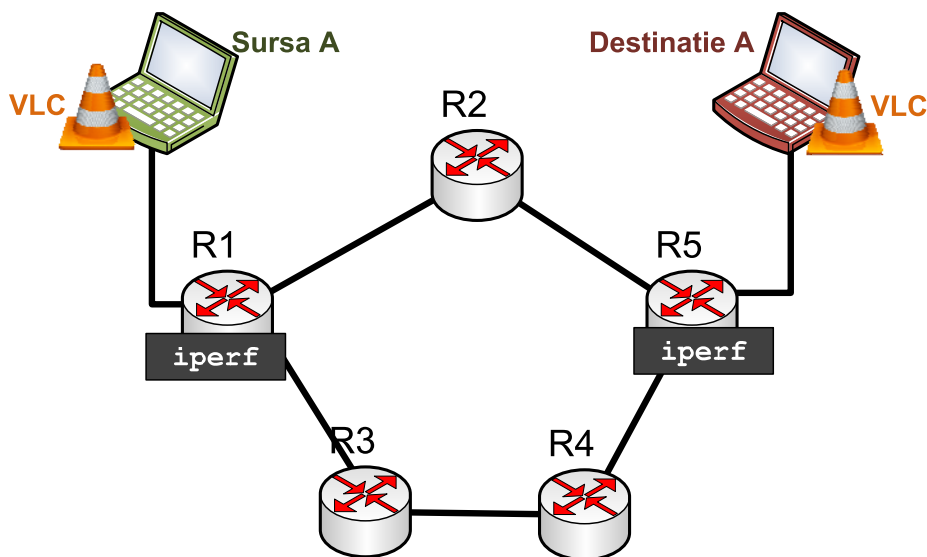


Figura 6-3 Arhitectura de test utilizată

Deoarece nu am avut la dispoziție o Fermă de servere, topologia de test a diferit într-o oarecare măsură față de cea din scenariul prezentat anterior, după cum este ilustrat și în Figura 6-3. Fluxul video transmis de la Sursa A către Destinația A a fost generat cu ajutorul utilitarului VLC. Fluxul de date ce ar fi trebuit să circule de la Fermă de servere la Destinația B a fost emulat cu ajutorul

utilitarului *iperf* între nodurile R1 și R5. Etapele din cadrul testului au fost similare cu cele corespunzătoare scenariului considerat, cu mențiunea că fluxul video dintre Sursa A și Destinația A a fost mult mai mic (în jur de 200 kbps) iar cel generat de *iperf* a fost mai mare (aproximativ 99.9 Mbps).

Rezultatul obținut în cazul în care s-a utilizat un protocol clasic de rutare (care nu ține cont de starea actuală a rețelei) poate fi vizualizat în Figura 6-4. Se observă că la secunda 42, rata de transfer a fluxului video scade brusc. Acest fenomen este datorat congestiei care apare datorită celui de-al doilea flux de date emis în topologia de test. Din punctul de vedere al utilizatorului, acest fenomen este sesizat printr-o scădere semnificativă a calității transmisiei video, ajungând ca imaginea primită să fie indescifrabilă.

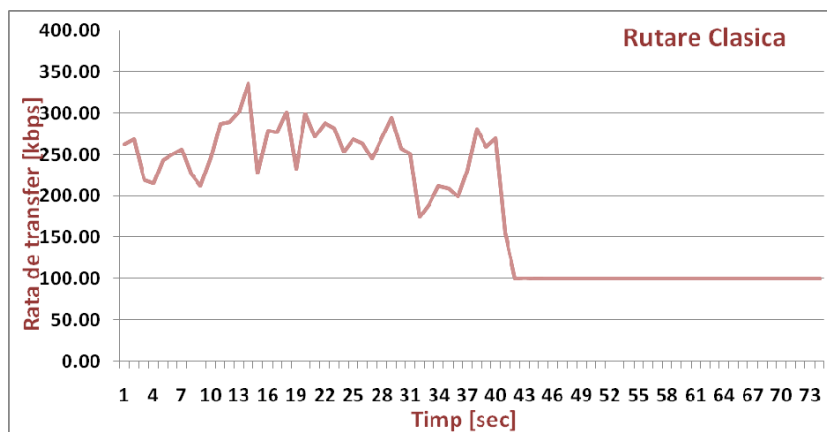


Figura 6-4 Forma fluxului de date dintre Sursa A și Destinația A atunci când se implementează rutarea clasică

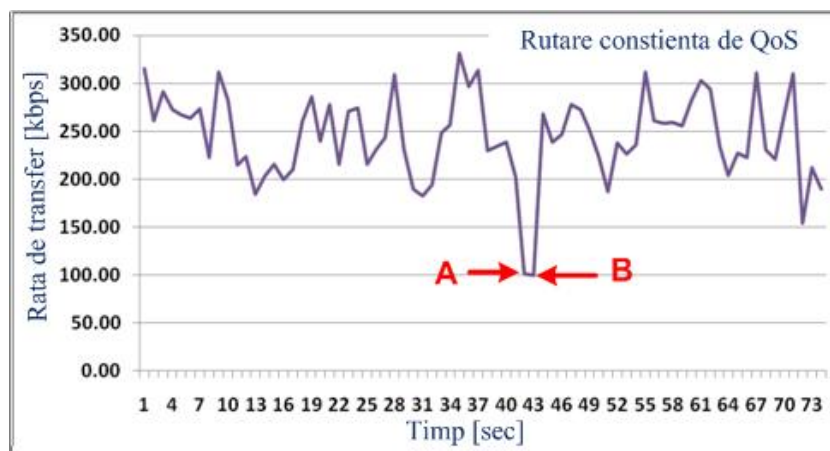


Figura 6-5 Forma fluxului de date dintre Sursa A și Destinația A atunci când se implementează rutarea bazată pe QoS

Dacă repetăm același scenariu de test dar de această dată activând în rețea un protocol de rutare care este conștient de parametrii QoS, observăm o îmbunătățire semnificativă a rezultatelor (vezi Figura 6-5). Dacă la secunda 42 (momentul marcat în grafic cu A) se observă o scădere bruscă a ratei de transfer corespunzătoare fluxului video dintre Sursa A și Destinația A, de îndată ce congestia este detectată de către mecanismul CLQ, protocolul de rutare conștient de QoS va lua măsurile necesare eliminării congestiei. În acest caz, traficul emis de *iperf* a fost re-rutat pe calea R1-R3-R4-R5 (vezi Figura 6-3). Consecința acestui fapt este marcată în graficul de mai jos cu B, care este momentul în care rata de transfer a fluxului video își revine la forma inițială. Din punctul de vedere al utilizatorului, s-a observat timp de o secundă înrăutățirea bruscă a imaginii după care calitatea la recepție și-a revenit la normal.

6.5 Concluzii

În acest capitol sunt prezentate avantajele utilizării parametrilor de trafic (măsurați deasupra substratului MAC) atunci când se iau decizii de rutare. Astfel, rezultatul este un mecanism de rutare conștient de QoS ce folosește informații primite de la un modul CLQ. Pentru a determina cea mai bună cale pe care să se ruteze traficul, se va utiliza un indicator global denumit Metrică compozită ce înglobează informații despre rata de transfer disponibilă și latența caracteristică unei anumite rute.

Ca și dezvoltări ulterioare se are în vedere implementarea unui utilitar soft care să măsoare rata de transfer disponibilă utilizând tehnici active de măsură în loc de cele pasive, așa cum se face în prezent. Astfel se prevede creșterea acurateții rezultatelor obținute. Motivul pentru care măsurarea activă nu a fost o prioritate este faptul că s-a dorit demonstrarea fezabilității ideii, urmând ca în cea de-a doua fază să optimizăm întregul sistem. O a doua problemă care trebuie rezolvată pe viitor este legată de fenomenul de oscilație a rutelor care apare atunci când două sau mai multe căi disponibile către aceeași destinație au performanțe aproximativ egale. Soluția implementată temporar a constat în utilizarea unor praguri de tip histerezis reducând astfel semnificativ problema. Pe viitor, trebuie studiate mai multe astfel de mecanisme astfel încât să se utilizeze cel mai eficient dintre ele.

Capitolul 7 Implementarea și evaluarea modulului CLQ pentru routere MPLS

7.1 Motivație

MPLS (Multiprotocol Label Switching) este un protocol întâlnit în ziua de azi în special în nucleele rețelelor gestionate de operatorii de telecomunicații. Acesta este caracterizat prin faptul că utilizează un set de etichete atribuite pachetelor de date, pentru a realiza comutația acestora. Un avantaj major este acela că oferă servicii specifice comutației de circuite: securitate, calitatea serviciilor. Datorită simplității, fiabilității și flexibilității care îl caracterizează, tehnologia MPLS se pretează a fi folosită atunci când este necesar a fi implementate procese de inginerie a traficului astfel încât să se garanteze utilizarea eficientă a resurselor disponibile în rețea [Kan05]. Capitolul curent propune inserarea unui modul de Cross-Layer QoS (CLQ), astfel încât să se automatizeze procesele de inginerie a traficului (mă voi axa pe găsirea căii optime pe care să se transmită fluxurile de date).

7.2 Calculul metricii compozite

Parametrii QoS ce caracterizează obiectiv un canal de comunicație sunt utilizați pentru calculul unui indicator global denumit Metrică Compozită (CM – Composite Metric). Acesta oferă o imagine de ansamblu asupra performanței actuale din rețea, fiind folosit pentru detecția anumitor deficiențe de performanță. Astfel, având această informație, orice algoritm poate lua decizia ce mai bună la un moment dat. Ecuația folosită pentru calculul CM este următoarea:

$$CM = \frac{10^{10}}{RATE[bps]} + \frac{DLY[sec]}{10^{-6}} + 10^{10} * BER \quad (7.1)$$

Unde:

- RATE = rata de transfer disponibilă pe canal
- DLY = latența (întârzierea într-un singur sens)
- BER = rata de eroare pe bit

În aplicația de față, metrica compozită este utilizată pentru a detecta calea optimă dintre o sursă și o destinație într-un domeniu MPLS. Analizând ecuația de mai sus se poate observa că o metrică mică este caracteristică unei căi mai performante. Astfel, analizând două sau mai multe căi disponibile, se va alege calea cu metrica minimă.

O altă contribuție inclusă în acest capitol se referă la utilizarea mecanismului de distribuție a etichetelor pentru publicarea metricilor compozite în întregul domeniu. Astfel, cei 20 de biți ai unei etichete MPLS vor fi utilizați astfel (vezi Figura 7-1):

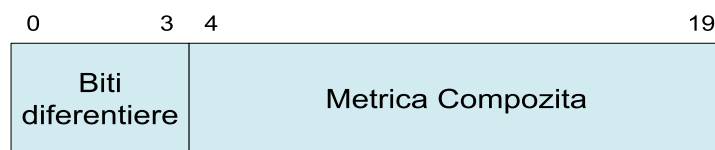


Figura 7-1 Structura internă a unei etichete MPLS

Deoarece eticheta MPLS poate fi aleasă aleator, am decis utilizarea acesteia pe post de container specializat în transportul metricii compozite. Această metrică va caracteriza legătura pe care respectivul pachet circulă. Datorită faptului că am alocat doar 16 biți pentru codarea CM, domeniul în care aceasta poate lua valori este: 0 ... 65535. Primii patru biți ai etichetei au fost rezervați pentru a face diferențierea între etichetele calculate pe legături ce au aceleași performanțe. De asemenea, dacă un pachet va conține mai multe etichete (transportând aceeași informație CM) trebuie să existe un set de biți utilizați pentru diferențierea acestora.

7.3 Teste

Pentru a cuantifica îmbunătățirile aduse de propunerea prezentată în acest capitol (adăugarea unui mecanism CLQ într-un domeniu MPLS), o serie de simulări au fost efectuate în OMNET++. Topologia de test a fost compusă din cinci routere (R1, R2, R3, R4, R5), un nod sursă (Host 1) și un nod destinație (Host 2) – vezi Figura 7-2.

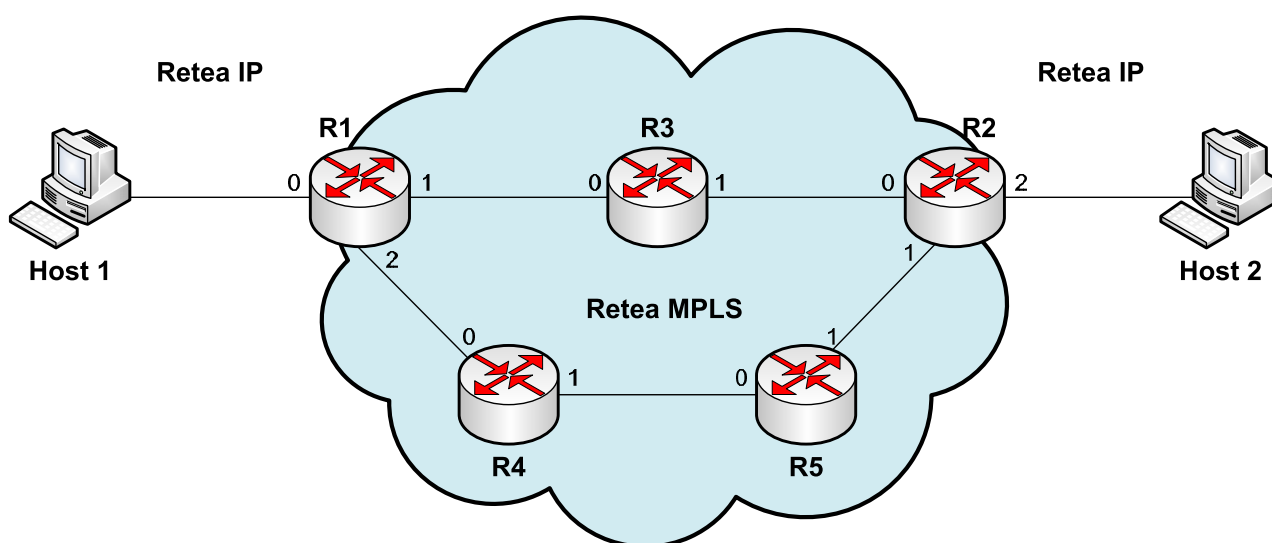


Figura 7-2 Topologia de test simulată în OMNET++

Studiind topologia de mai sus, se poate observa că între echipamentul sursă (Host 1) și echipamentul destinație (Host 2) există două căi disponibile (Host1-R1-R3-R2-Host2 și Host1-R1-R4-R5-R2-Host2). Scenariul de test constă în emiterea unui trafic ICMP generat de către o aplicație de tip *ping*, de la Host 1 la Host2. O dată cu începerea simulării, ne așteptăm să observăm faptul că datele sunt transmise prin domeniul MPLS utilizând calea optimă (cea care are performanțele cele mai bune din punct de vedere a parametrilor de trafic măsurați: rată de transfer disponibilă, latență și probabilitate de eroare pe bit).

Pentru a evalua cât mai obiectiv îmbunătățirile aduse de noua abordare comparativ cu varianta clasică, s-a folosit același scenariu de test în ambele teste (Test 1 și Test 2) și s-a evaluat parametrul întârziere dus întors măsurat cu ajutorul traficului ICMP. Se va observa în graficele ce urmează, faptul că atunci când se activează mecanismul CLQ, întârzierea măsurată este mai mică decât în abordarea clasică (în care se consideră doar caracteristicile teoretice ale legăturilor), datorită faptului că se asigură un grad de adaptabilitate a mecanismului de rutare, la starea actuală a rețelei.

Test 1 a constatat în simularea domeniului MPLS în care s-a activat mecanismul de Cross-Layer QoS în timp ce Test 2 este scenariul în care s-a utilizat abordarea clasică care nu ține cont de parametrii de trafic. Pe parcursul ambelor scenarii, s-a înrăutățit pentru un timp calea Host 1-R1-R3-R2-Host 2 pe care circulă inițial traficul ICMP. Astfel, atunci când CLQ este activat, deciziile luate vor ține cont de această înrăutățire (doar în cazul scenariului Test 1) și vor utiliza calea Host

1-R1-R4-R5-R2-Host 2 care prezintă performanțe superioare. Rezultatele celor două scenarii de test sunt ilustrate atât în Figura 7-3 cât și în Figura 7-4.

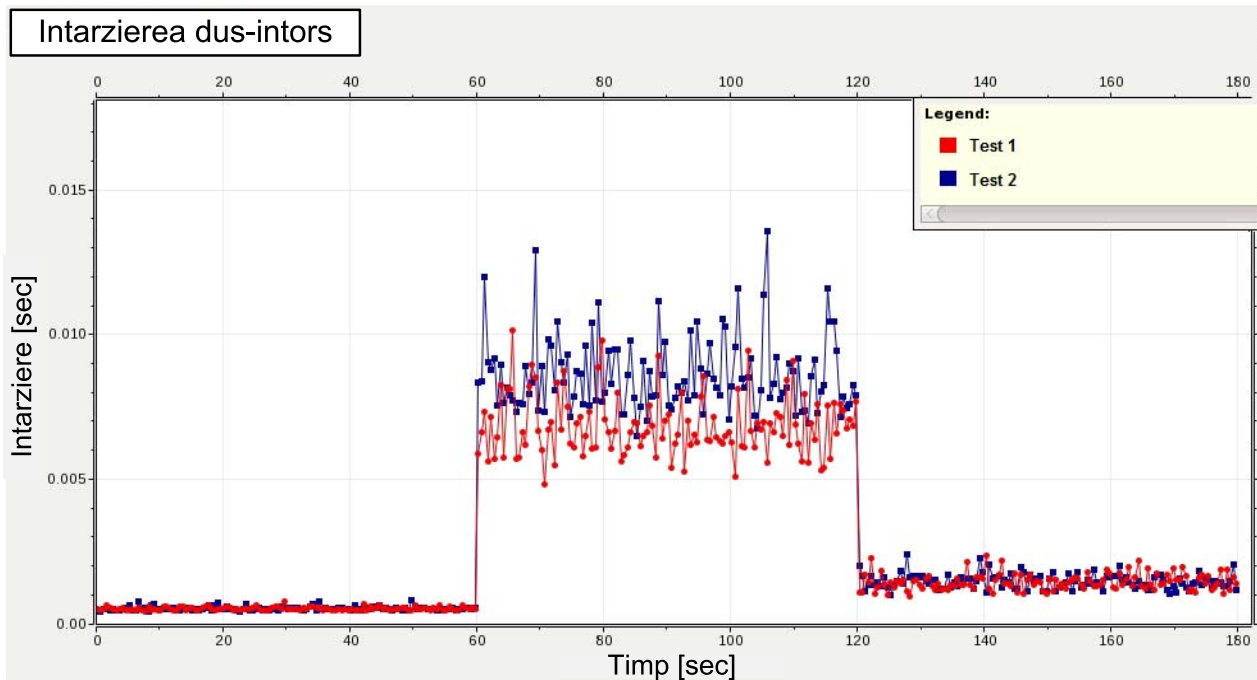


Figura 7-3 Întârzierea dus-întors măsurată între Host 1 și Host 2 folosind trafic ICMP

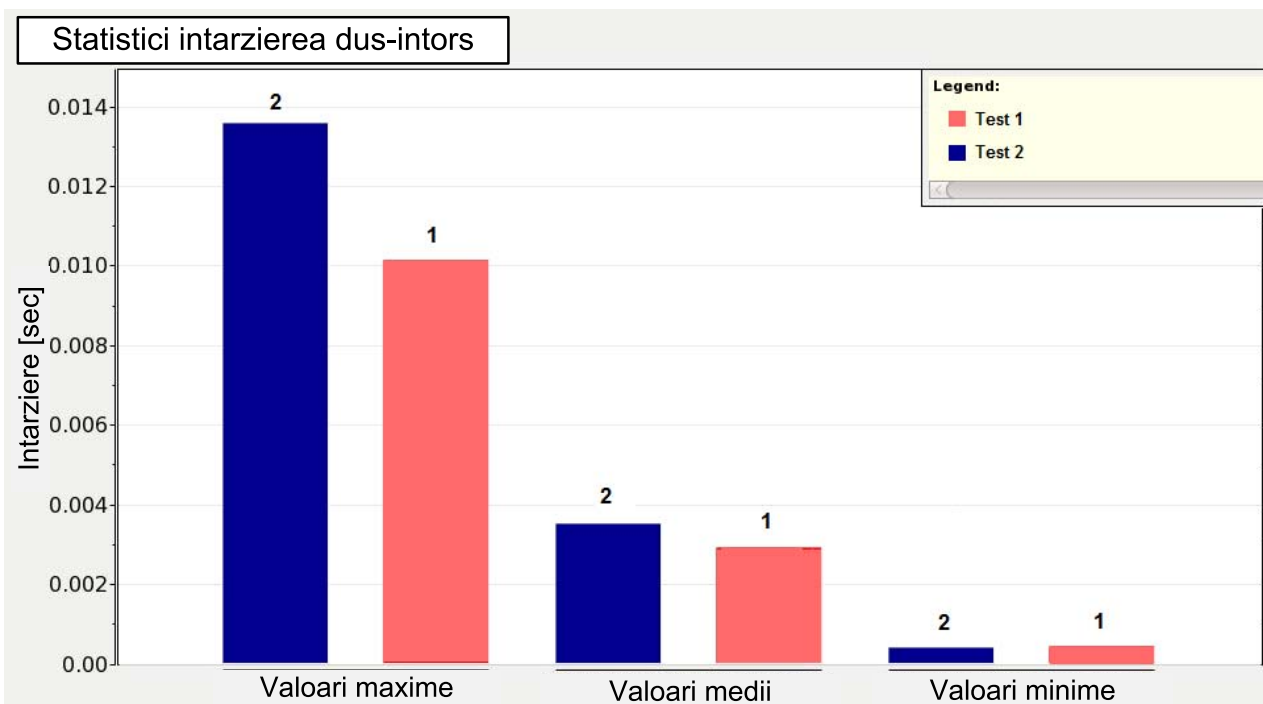


Figura 7-4 Comparație între valorile maxime, medii și minime ale întârzierii dus întors în cazul celor două teste

În Figura 7-3 sunt ilustrate graficele măsurătorilor în cazul variantei cu CLQ activat (Test 1 – graficul roșu) cât și în cazul clasic (Test 2 – graficul albastru). Înainte de secunda 60 și după secunda 120 se observă faptul că graficul întârzierii în cele două cazuri este aproximativ la fel deoarece calea aleasă pentru rutarea pachetelor MPLS este aceeași (Host 1-R1-R3-R2-Host 2). De îndată ce această cale este înrăutățită (secundele 61 - 119), mecanismul CLQ detectează acest lucru

și schimba ruta, folosind varianta Host 1-R1-R4-R5-R2-Host 2. De aceea, se poate observa o scădere a întârzierii comparativ cu cazul clasic în care deteriorarea performanțelor căii inițiale este total transparentă. În Figura 7-4 se face o comparație între cele două abordări, utilizând un set de statistici extras din rezultatele obținute și ilustrate în Figura 7-3.

7.4 Concluzii

Acest capitol prezintă o metodă de îmbunătățire a calității serviciilor într-o rețea MPLS prin utilizarea unui mecanism adaptiv rutare. Abordarea propusă ține cont de situația reală din rețea, caracterizată de către un set de parametri măsurați și făcuți publici de către mecanismul Cross-Layer QoS: rata de transfer disponibilă, latența și probabilitatea de eroare pe bit. Astfel, ținând cont de aceste informații, mecanismul de rutare își va optimiza deciziile în timp real.

Pentru validarea soluției propuse, au fost efectuate în OMNET++ o serie de simulări în care s-au implementat atât mecanismul propus cât și cel clasic, iar rezultatele au fost ilustrate în graficele prezentate anterior. Testele efectuate au arătat că întârzierea cap-la-cap pentru varianta cu CLQ poate fi mai mică cu 15% ... 25% față de rezultatele obținute în rutarea MPLS clasică. În câteva cuvinte putem spune că s-a reușit implementarea preliminară a unui mecanism automatizat de inginerie a traficului, într-un domeniu MPLS. Acest lucru se datorează implementării în fiecare nod a unui modul de tip Cross-Layer QoS.

Ecuția utilizată în această secțiune pentru calculul metricii compozite (ecuația (7.1)) folosește un set particular de valori pentru constantele utilizate pentru normalizarea parametrilor rată de transfer disponibilă, latență și probabilitate de eroare pe bit. Acest lucru este datorat faptului că din punct de vedere temporar, această contribuție a fost prima din cadrul perioadei de lucru la doctorat, iar formula metricii a fost într-o formă incipientă, cunoscând pe parcurs o transformare la forma ilustrată în ecuația (5.1). Motivul pentru care experimentele nu au fost repetate s-a datorat faptului că deciziile luate și rezultatele testelor nu ar fi diferit, singura modificare ar fi fost vizibilă analizând valoare instantanee a parametrului CM.

Capitolul 8 Îmbunătățirea algoritmului lui Dijkstra cu CLQ

8.1 Motivație

Protocoalele de rutare sunt una dintre cele mai importante entități ce pot influența calitatea serviciilor primite de către utilizatorul final [Mac06], [Med07]. Din nefericire, în ziua de azi majoritatea protocoalelor clasice utilizează informații statice, atunci când iau decizii de rutare a pachetelor, spre o destinație oarecare. Pe de altă parte, pentru a alege întotdeauna calea optimă, este nevoie de a ține cont de mai mult de un singur parametru de trafic. Astfel, în această secțiune se vor considera parametri: rată de transfer disponibilă (ATR), latența (OWD) și probabilitatea de eroare pe bit (BER).

Contribuția prezentată în acest capitol constă în îmbunătățirea algoritmului lui Dijkstra (variantea cu surse multiple și o singură destinație), folosit de către protocolul OSPF (Open Shortest Path First), astfel încât să se optimizeze deciziile de rutare. Modificările propuse încearcă să corecteze deficiențele mecanismelor de rutare din ziua de azi, astfel încât să se țină cont de dinamicitatea parametrilor QoS măsurați deasupra substratului MAC.

8.2 O privire de ansamblu asupra algoritmului lui Dijkstra clasic

Varianta algoritmului lui Dijkstra considerat în această secțiune este cea care rezolvă problema surselor multiple și o singură destinație, utilizând arborele de rutare (Spanning Tree). Acesta include cele mai scurte căi (din punct de vedere a costului minim) de la fiecare nod sursă către același nod destinație. În continuare se va face o scurtă descriere a variantei clasice, utilizându-se următoarele notații:

- D_i : costul căii de la nodul sursă i la nodul destinație d
- P : mulțimea nodurilor i pentru care s-a calculat cea mai scurtă cale
- Q : mulțimea nodurilor pentru care nu s-a calculat încă cea mai scurtă cale
- d_{ij} : distanța unei legături între două noduri direct conectate i și j

În cazul protocolului OSPF, costul unei legături dintre două noduri direct conectate este (D_{ij}) cu condiția ca $D_{ij} \geq 0$. Formula utilizată pentru calculul costului este:

$$D_i = D_{ij} + D_j = \frac{10^9 [bps]}{C_{ij} [bps]} + D_j \quad (8.1)$$

unde: $C_{ij} \geq 0$ este capacitatea teoretică a legăturii, dintre nodurile i și j . Costul total a unei căi (de la un nod oarecare i la destinație – D_i), calculat în cazul protocolului OSPF ce utilizează algoritmul Dijkstra clasic, este egal cu suma tuturor costurilor corespunzătoare legăturilor ce compun calea respectivă.

$$D_i = \sum_{j,k} D_{jk} = \sum_{j,k} \frac{10^9}{C_{jk}} \quad (8.2)$$

Pentru o prezentare cât mai intuitivă, în cele ce urmează va fi ilustrată schema logică a variantei clasice a algoritmului lui Dijkstra (variantea cu surse multiple și o singură destinație) – vezi Figura 8-1

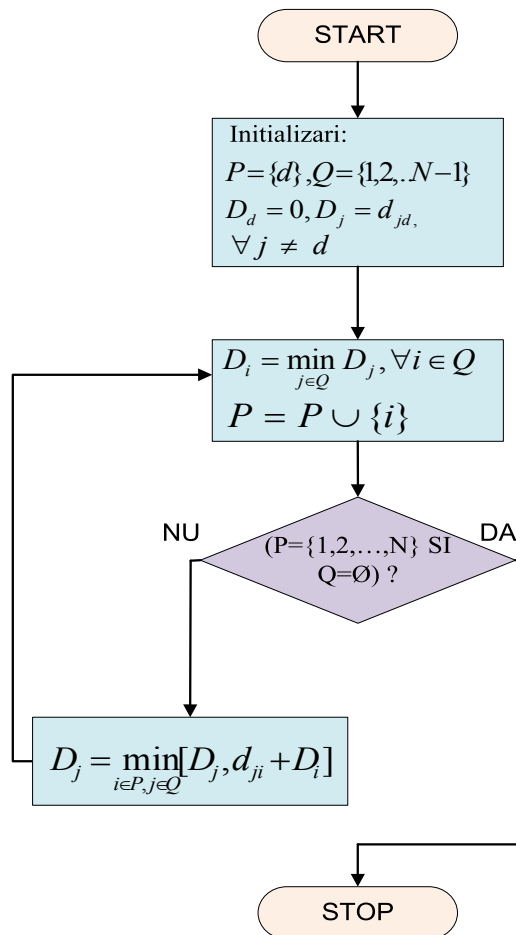


Figura 8-1 Algoritmul lui Dijkstra clasic

8.3 Modificări aduse algoritmului lui Dijkstra

În această secțiune voi face o prezentare a modului în care algoritmul Dijkstra a fost modificat, astfel încât să utilizeze informațiile oferite de către modulul Cross-Layer QoS (CLQ). Cele trei modificări aduse variantei clasice ale algoritmului sunt următoarele:

- Utilizarea ratei de transfer disponibilă, corespunzătoare unei legături dintre două noduri direct conectate. Acest indicator va înlocui parametrul corespunzător capacității teoretice a canalului;
- Utilizarea a doi parametri QoS adiționali (latența – OWD și probabilitatea de eroare pe bit – BER) pentru calculul unei metrici compozite ce va înlocui parametrul cost, utilizat în abordarea clasică;
- O dată ce s-a calculat arborele Spanning Tree, algoritmul Dijkstra va rula din nou (după un interval de timp setat în funcție de dinamicitatea infrastructurii) cu un nou set de parametri QoS primiți de la modulul CLQ.

În timpul procedurii de identificare a căii optime dintre o sursă și o destinație, se vor lua în considerare un set de parametri QoS (ATR, OWD și BER) ce indică într-un mod obiectiv performanța legăturii. Valorile acestora nu vor fi folosite ca atare, ci vor fi utilizate pentru calculul unui indicator global de performanță, denumit Metrică Compozită (Composite Metric – CM). Rolul parametrului CM este același ca și cel pe care îl are costul unei căi în cazul protocolului OSPF care utilizează varianta clasică a algoritmului lui Dijkstra. Formula utilizată pentru calculul parametrului CM este prezentată în ecuația (8.7). Se observă faptul că pentru calculul valorii finale corespunzătoare metricii compozite, este luată în considerare valoarea calculată global, pe întreaga cale, corespunzătoare fiecărui parametru QoS considerat (ATR_T , OWD_T și BER_T). Trebuie ținut cont

de faptul că fiecare dintre cei trei parametri QoS au o compozabilitate (mod în care se calculează valoarea respectivului parametru pe toată ruta) diferită. În cele ce urmează va fi ilustrat modul de calcul a valorii totale a parametrilor QoS considerați.

- Rata totală de transfer disponibilă (ATR_T) a unei căi este egală cu valoarea minimă dintre toate ratele de transfer disponibile pe legăturile ce compun calea respectivă.

$$ATR_T = \min_{i,j}(ATR_{ij}) \quad (8.3)$$

Unde ATR_{ij} =rata de transfer disponibilă între nodurile i și j ;

- Valoarea totală a latenței unei căi (OWD_T) este egală cu suma latențelor corespunzătoare tuturor legăturilor ce compun calea respectivă.

$$OWD_T = \sum_{i,j} OWD_{ij} \quad (8.4)$$

Unde OWD_{ij} =latența căilor dintre nodurile i și j ;

- Valoarea totală a probabilității de eroare pe bit (BER_T) corespunzătoare unei căi, ia în considerare probabilitatea de eroare pe bit a fiecărei legături ce compune calea respectivă:

$$BER_T = 1 - P_T \quad (8.5)$$

$$P_T = \prod_{i,j} (1 - BER_{ij}) \quad (8.6)$$

Unde BER_{ij} =probabilitatea de eroare pe bit a legăturii dintre nodurile i și j ;

P_T =probabilitatea de a nu avea erori pe întreaga cale.

Costul (D_j) calculat pentru întreaga cale, va agrega într-o singură metrică, denumită metrică compozită (CM), valorile calculate global a celor trei parametri QoS considerați. Formula folosită este ilustrată mai jos [Rus10a]:

$$D_j = CM = \frac{K_0}{ATR_T} + \frac{OWD_T}{K_1} + K_2 * BER_T \quad (8.7)$$

Unde: $K_0 = 10^9 [bps]$ $K_1 = 10^{-5} [s]$ $K_2 = 10^{12}$. Valorile constantelor au fost alese astfel încât metrica să fie minimă pentru o legătură cu capacitatea de 1 Gbps, o latență de 10 μs și o probabilitate de eroare pe bit egală cu 10^{-12} .

8.4 Testarea performanțelor algoritmului lui Dijkstra modificat

Pentru a demonstra beneficiile aduse de varianta îmbunătățită a algoritmului lui Dijkstra, s-au efectuat o serie de teste cu ajutorul simulatorului OMNET++. Topologia utilizată este compusă dintr-un număr de 13 routere. Scopul principal a acestor simulări este de a verifica dacă varianta îmbunătățită a algoritmului ține cont de variabilitatea întregii topologii, fapt ce este total transparent pentru algoritmul clasic. Pentru evidențierea avantajelor, rezultatele obținute vor fi comparate cu cele corespunzătoare simulărilor în care s-a utilizat algoritmul clasic pe aceeași topologie de test. Indicatorul de performanță ales este întârzierea cap-la-cap (End-to-End Delay – EED) măsurată pe un flux video ce este transmis prin topologia de test.

Simulările efectuate au luat în considerare mai multe scenarii în care au fost variați pe rând parametrii OWD, ATR și BER iar în ultimul scenariu, s-au modificat cei trei parametri în același timp, pe legăturile vizate a fi înrăutățite.

Analizând Figura 8-2 se observă faptul că parametrul EED are valori mai mari (întârziere mărită este un indicator al performanțelor inferioare corespunzătoare unei căi) în cazul variantei clasice ale algoritmului. De asemenea, datorită probabilității mari de eroare a căii utilizate, pe graficul afișat se pot observa mai multe subțieri, care ilustrează faptul că au existat pachete pierdute. Toate aceste fenomene se traduc print-o calitate scăzută a fluxului video recepționat de către

utilizatorul final. Astfel, putem spune că din punct de vedere a parametrului EED, s-a observat o îmbunătățire de 10 ori a performanțelor, atunci când s-au înrăutățit parametrii a două dintre legăturile folosite, față de rezultatele obținute în cazul algoritmului lui Dijkstra clasic. Bine înțeles există un preț ce trebuie plătit atunci când se activează soluția propusă. Pentru ca toate nodurile să fie informate cu privire la starea actuală a rețelei, indicatori de performanță măsurați pe toate legăturile trebuie publicate tuturor nodurilor, de către modulul de CLQ. Astfel, ținând cont de faptul că măsurătorile QoS erau publicate de fiecare din cele 13 noduri o dată pe secundă, traficul redundant transmis în topologia de test a avut o valoare totală de 192 kbps. Consider că acest lucru nu este un dezavantaj major ținând cont de beneficiile evidențiate prin testele efectuate.

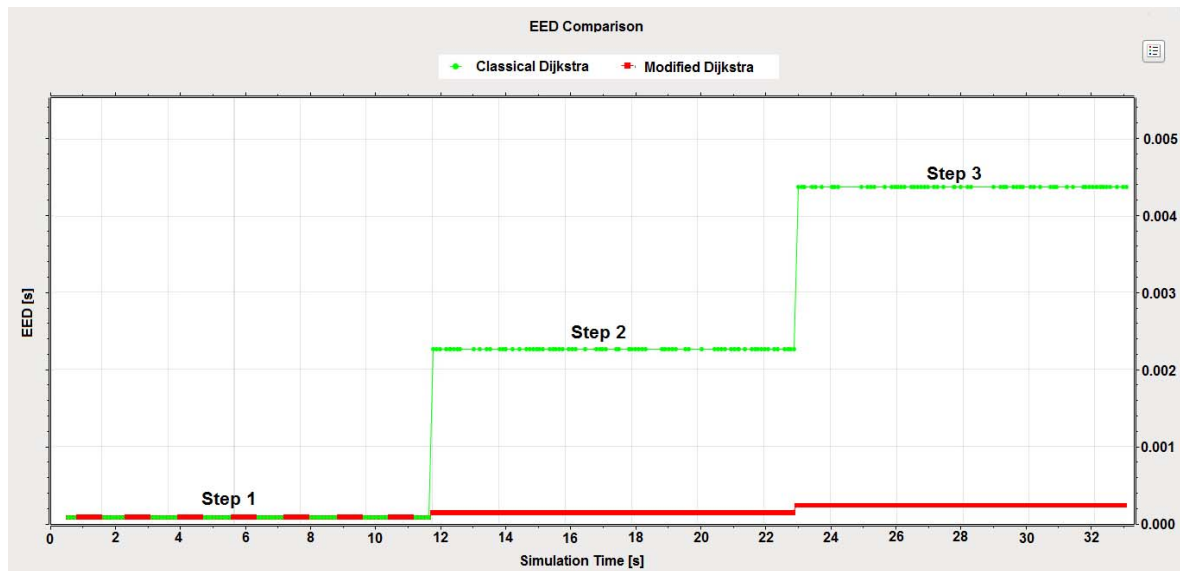


Figura 8-2 Comparație între performanțele algoritmului lui Dijkstra clasic (graficul cu verde) și algoritmul Dijkstra modificat (graficul cu roșu)

8.5 Concluzii

Capitolul de față prezintă o metodă de modificare a algoritmului Dijkstra prin utilizarea de informații primite de la un modul CLQ astfel încât să se îmbunătățească calitatea serviciilor primite de către utilizatorul final, printr-o anumită topologie de rețea. Ideea de bază constă în utilizarea unui set de parametri QoS actualizați în timp real, atunci când se construiește arborele Spanning Tree (ST). Astfel, de fiecare dată când apar probleme în anumite secțiuni ale rețelei, acestea vor fi evitate datorită faptului că ST va fi reactualizat.

Contribuțiile principale ale acestui capitol sunt următoarele:

- Utilizarea parametrului rată de transfer disponibilă va înlocui valoarea capacității teoretice a canalului;
- Propunerea unei metrici compozite (actualizate permanent) ce va înlocui indicatorul cost calculat pe fiecare cale în cazul algoritmului lui Dijkstra clasic;
- Parametrii latență și probabilitate de eroare pe bit vor fi de asemenea considerați atunci când se va calcula metrica compozită corespunzătoare unei căi;
- Determinarea valorilor globale ale parametrilor QoS înainte de a fi utilizate în calculul metricii compozite;
- Execuția algoritmului va fi reluată de fiecare dată când se va primi un set actualizat de parametri QoS, astfel încât să se adapteze arborele ST la orice anomalie ce poate apărea în rețea.

Capitolul 9 Viitorul Internet de obiecte și dispozitive inteligente

9.1 Motivație

De la începuturi, Internetul a fost proiectat a fi o infrastructură specializată în a oferi căi de comunicare între echipamente. De atunci și până în ziua de azi, modul de accesare a informației a rămas în esență același (entitatea care are nevoie de anumite date trebuie să specifice adresa exactă a echipamentului pe care acestea sunt stocate, înainte de a le accesa). Această paradigmă s-a dovedit a nu fi cea optimă atunci când informația este stocată de exemplu pe mai multe echipamente (tehnologii de tip Cloud computing – nor informațional). Soluția o reprezintă utilizarea unei noi abordări, unde rețeaua este văzută ca o infrastructură centrată pe informație [Ohl09]. Astfel noua arhitectură va oferi căi de comunicație între obiecte de informație și entități ce au nevoie de acestea. Capitol de față este rezultatul experienței dobândite de pe urma contribuțiilor prezentate anterior. Astfel, în această secțiune voi propune un Internet al viitorului unde multe din îmbunătățiri se vor baza pe serviciile oferite de modulul de Cross-Layer QoS (CLQ).

9.2 Entități prevăzute în viitorul Internet

În cele ce urmează voi face o prezentare a entităților prevăzute a fi implementate în viitorul Internet.

9.2.1 Obiecte inteligente de informație

Dacă până recent tendința era de a muta o parte din inteligența rețelei către marginile acesteia (echipamentele finale), în această secțiune se propune inserarea unei oarecare inteligențe și în containerele ce transportă informația între diverse entități. Un astfel de container va fi abstractizat printr-un obiect inteligent de informație (Intelligent Information Object - **IIO**). Această abordare ar avea ca și avantaj o reducere a traficului de semnalizare, schimbat între diverse echipamente în abordarea clasică, datorită faptului că IIO vor avea capacitatea de a decide singure care sunt deciziile cele mai eficiente ce trebuiesc luate într-o situație oarecare. Informațiile necesare pentru ca entitățile IIO să ia decizii optime, vor fi oferite în fiecare nod de către un modul CLQ.

Arhitectura internă a unui IIO prevede trei secțiuni:

1. Secțiunea *Proprietăți*: conține o listă de caracteristici ale obiectului;
2. Secțiunea *Cod binar*: reprezintă codul binar care oferă inteligență respectivului obiect;
3. Secțiunea *Date*: este zona în care vor fi stocate datele transportate în rețea.

O reprezentare intuitivă a arhitecturii unui IIO este ilustrată în Figura 9-1.

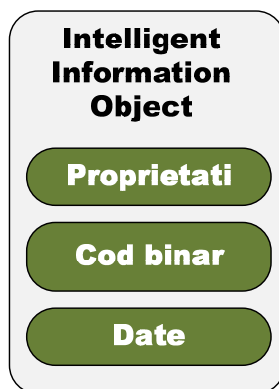


Figura 9-1 Structura internă a unui IIO

Prima secțiune, Proprietăți va include următoarele informații:

- Identificatori ai sursei și destinației;
- Constrângeri: include un set de parametri QoS ce trebuie respectați de către datele transportate. În funcție de aceștia se va alege calea optimă spre destinație;
- Alte proprietăți ce pot fi adăugate ulterior.

Secțiunea Cod binar conține de fapt inteligența respectivului obiect. Prin execuția codului inclus, entitatea IIO va fi capabilă să ia decizii optime în ceea ce privește cea mai bună cale spre o destinație. O listă generală de acțiuni ce pot fi executate de un astfel de cod binar este următoarea:

- Interogarea parametrilor QoS corespunzătoare căilor disponibile;
- Compararea performanțelor fiecărei căi și alegerea celei optime (variază în funcție de constrângerile specificate în secțiunea Proprietăți);
- Dacă nu există nici o rută disponibilă, IIO se va întoarce în nodul anterior de unde va urma o nouă cale;
- Verificarea faptului că acțiunile ce vor fi executate nu sunt în contradicție cu măsurile de securitate impuse de către managerul rețelei.

Secțiunea Date este cea care conține datele utilizatorului ce trebuie transportate în deplină siguranță, menținând totodată un anumit nivel cerut de calitate a serviciilor. Respectiva informație poate fi transportată în diverse formate: date de tip text/XML, date video, date audio etc.

9.2.2 Obiecte inteligente de tip resursă

Deoarece obiectul este entitatea centrală din Internetul viitorului, ar fi util abstractizarea tuturor resurselor unei rețele cu ajutorul acestor obiecte inteligente de tip resursă (Intelligent Resource Objects – IRO). Aceste abstracțiuni vor conține informații ce fac referire la resursele fizice a respectivei rețele: rata de transfer, probabilitate de eroare pe bit, putere de procesare, capacitate de stocare etc. Motivul pentru care aceste obiecte de tip resursă trebuie să fie și inteligente este datorat necesității de a permite colaborarea între astfel de obiecte, în vederea creării unor rețele virtuale cu performanțe particularizate. O dată abstractizate toate resursele rețele, acestea pot fi concentrate la nevoie în diverse secțiuni ale acesteia, unde există pentru o perioadă finită de timp o supra utilizare, astfel încât să se satisfacă nevoile curente ale utilizatorilor. Pentru a implementa aceste cerințe, structura internă a unui astfel de obiect trebuie să fie în concordanță cu cea prezentată în Figura 9-2.

- ID echipament: identificator al echipamentului căruia îi corespunde respectiva resursă;
- Nume parametru: numele parametrului care este abstractizat de către IRO;
- Valoarea+Unitate măsură: valoarea și unitatea de măsură corespunzătoare parametrului abstractizat;
- Cod binar: codul executabil care reprezintă inteligența respectivului obiect.

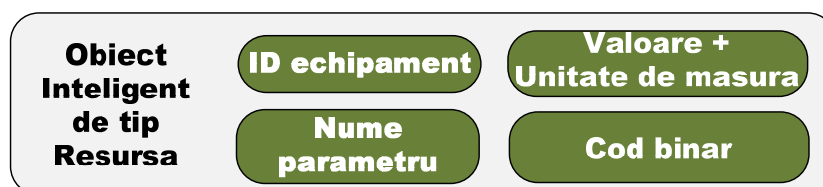


Figura 9-2 Structura internă a unui IRO

9.2.3 Echipament inteligent

Echipamentul inteligent (Smart Device – SD) este echipamentul fizic, conectat la rețea. Caracteristicile acestuia sunt abstractizate de către un set de obiecte inteligente de tip resursă (IRO). Un alt atribut important este faptul că un SD va fi capabil să interacționeze securizat cu orice obiect

inteligent de informație (IIO) oferindu-i acestuia un mediu special în care să execute acțiuni specifice: auto-adaptare, auto-gestionare sau auto-rutare. Pentru interacțiunea cu IIO, echipamentele inteligente trebuie să activeze execuția unui modul de tip CLQ care va extrage din rețea parametrii de trafic corespunzători legăturilor existente. O altă proprietate ce trebuie implementată de către orice SD o reprezintă suportul nativ pentru virtualizare, deoarece se vehiculează că pe viitor aceasta va fi o cerință sigură în lumea rețelisticii.

9.3 Scenariu propus – Cooperarea dintre IIO și SD

Una dintre problemele majore în rețelele de comunicație din ziua de azi o reprezintă fenomenul de pierderi de pachete, ce apare în timpul diverselor transmisii. Două dintre motivele apariției acestui fenomen sunt: congestia ce poate apărea în routere sau erorile de pe canalul de comunicație. În Figura 9-3 este prezentată o comparație între modul în care fenomenul de congestie influențează atât rețeaua clasică cât și rețeaua viitorului (cea care utilizează entitățile IIO și SD).

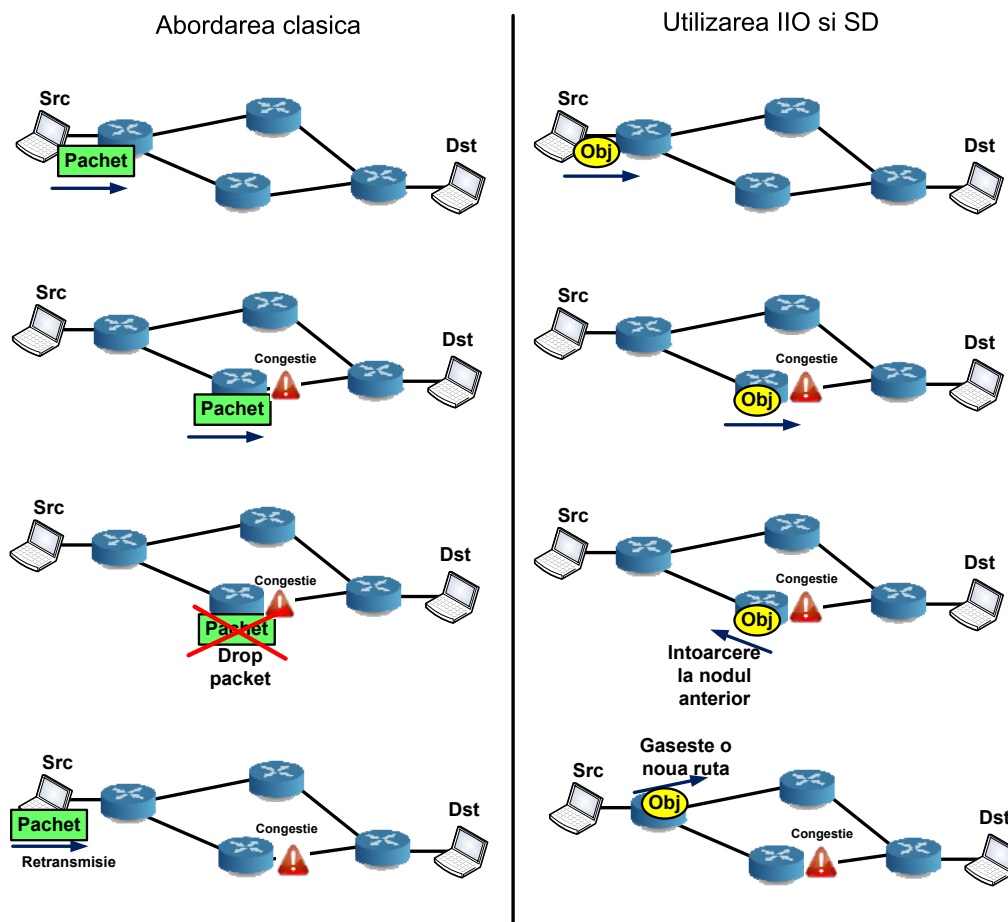


Figura 9-3 Comparație între abordarea clasică și utilizarea entităților IIO și SD

În primul pas se emite un pachet (abordarea clasică) de la o sursă la o destinație. Analog, în cazul noii abordări se emite de la sursă un IIO ce transportă datele de la un nod sursă la un nod destinație. În calea lor, atât pachetul de date cât și IIO ajung într-un nod ce experimentează congestie. Efectul este următorul: pachetul de date va fi șters iar sursa va fi anunțată prin diverse mecanisme că trebuie să retransmită respectivul pachet (rețeaua clasică) în timp ce obiectul inteligent de informație (IIO) se va întoarce la un nod anterior și va încerca găsirea altei căi spre destinație fără să mai implice sursă într-o nouă retransmisie (paradigma propusă). Pentru a informa

entitatea IIO cu privire la performanțele căilor disponibile spre o destinație, SD trebuie să extragă parametrii QoS de la un modul CLQ activ în fiecare nod.

9.4 Concluzii

Ideile prezentate în acest capitol sun pasul logic înainte spre o rețea a viitorului centrată pe informație, unde scopul nu va mai fi conectarea echipamentelor ci oferirea unei căi de comunicație între orice entitate și informația necesară. Unul dintre câștigurile imediate o reprezintă accesul mai rapid la datele dorite și de asemenea o îmbunătățire a experienței utilizatorului. Deoarece vorbim despre o rețea orientată pe obiecte de informație (ce abstractizează toate datele disponibile), o altă idee propusă în această secțiune este adăugarea de inteligență fiecărui obiect în parte, dându-i acestuia o oarecare autonomie în ceea ce privește găsirea căii optime spre destinație.

Desigur că va fi nevoie să se analizeze și să se elimine breșele de securitate care pot apărea o dată cu implementarea acestei paradigme, dar potențialul pe care o astfel de abordare îl are este superior eventualelor dezavantaje. Deoarece în viitor o cerință importantă o va reprezenta virtualizarea, implementarea obiectelor inteligente de tip resursă (IRO) vor îmbunătăți aceste procedee prin implementarea diverselor tehnici de colaborare între acestea.

Capitolul 10 Contribuții la calitatea serviciilor prin tehnici Cross-Layer pentru viitorul Internet

10.1 Sumarul contribuțiilor

În această secțiune voi prezenta o sinteză a tuturor contribuțiilor prezentate în capitolele anterioare

1. Clasificarea și evaluarea mecanismelor Cross-Layer QoS

În cel de-al doilea capitol s-au investigat șase tipuri de arhitecturi Cross-Layer (semnalizări de jos în sus, semnalizări de sus în jos, semnalizări în buclă, unirea a două sau mai multor straturi adiacente și calibrare pe verticală). Aceste arhitecturi pot fi implementate într-un sistem real, utilizând o serie de tehnici: comunicare directă între straturi, utilizarea în comun a unei baze de date în care diverse protocoalele vor stoca variabilele relevante, prin renunțarea completă la noțiunea de strat sau prin folosirea unui modul specializat de optimizare. Studiul arhitecturilor Cross-Layer a avut în vedere în cea de-a doua parte a capitolului, evaluarea soluțiilor existente, din punct de vedere al straturilor traversate, a tipului de informație schimbat (de stare sau de control) și a tipului de semnalizare de management (în banda sau în afara benzii). Ambele metode de localizare a mecanismelor de optimizare au fost de asemenea considerate (în afara sau în interiorul protocoalelor de comunicații), iar din punct de vedere temporar acestea pot fi sincrone sau asincrone.

Contribuția este localizată în: Capitolul 2
Publicații: [Rus08b], [Rus09a]

2. Proiectarea și implementarea preliminară a unei arhitecturi Clean-Slate pentru Cross-Layer QoS

În cadrul acestei contribuții s-a abordat conceptul de management intrinsec (inclus din faza de proiectare în interiorul protocoalelor de comunicație), axat pe calitatea serviciilor, prin proiectarea și implementarea unui modul Cross-Layer ce folosește semnalizarea de jos în sus și de sus în jos pentru transmisia informațiilor de stare și control, atât în bandă cât și în afara benzii. Acest modul lucrează deasupra sub-stratului MAC și propune o structură particulară de mesaje XML, schimbate cu alte entități din noua arhitectură a viitorului Internet (de exemplu Generic Path, NetInf). De asemenea, s-a prezentat diagrama completă a stărilor prin care poate trece modulul de Cross-Layer, utilizând limbajul SDL. Implementarea mecanismului proiectat a fost făcută în fiecare nod din domeniul gestionat, pe care rula sistemul de operare Linux, utilizând limbajul de programare Java, deasupra framework-ului OSGi.

Contribuția este localizată în: Capitolul 3
Publicații: [Rus09b], [Nun09], [Nun10], [Gon10], [Rus10a], [Car10]

3. Metodologii propuse pentru măsurarea parametrilor QoS

Această contribuție include un set de tehnici de măsurare care au ca scop monitorizarea următorului set de parametri QoS: întârzierea dus-întors (RTT), latența (OWD), rata de transfer disponibilă (ATR) și probabilitatea de eroare de biți (BER). Primii doi parametri au fost determinați folosind tehnici de măsurări active utilizând probe particularizate. Rata de transfer disponibilă pe canal s-a determinat folosind metode pasive de măsurare în timp ce probabilitatea de eroare pe bit a fost estimată utilizând modele matematice specifice modulațiilor BPSK, QPSK și N-QAM folosite în tehnologiile 802.11 a,b,g și xDSL.

Contribuția este localizată în: Capitolul 4
Publicații: [Rus09b]

4. Implementarea și evaluarea CLQ pentru controlul congestiei

În situația în care în rețelele congestionate nu se pot adăuga noi resurse, mecanismele de Cross-Layer QoS pot fi folosite împreună cu tehnici de codare cooperative de tip Network Coding (NC), reprezentând o soluție alternativă la rutarea conștientă de QoS. Implementările ce utilizează tehnici de cooperare de tip Network Coding folosesc rata de transfer disponibilă în alte segmente de rețea. Testele efectuate pentru validarea acestei contribuții au constau într-o implementare preliminară la nivelul stratului aplicație a tehnicilor NC. Topologia de test folosită a fost de tip fluture cu 6 noduri, pe care rulează sistemul de operare Linux. S-a dovedit astfel că se poate face față congestiei (fără a o elimina) astfel încât procentul de pachete pierdute să scadă de la aproximativ 20% la sub 1%. Fezabilitatea ideii fiind demonstrată și validată în primul set de experimente, s-a efectuat un al doilea set de teste în care implementarea a fost făcută la nivelul stratului rețea. Concluzia extrasă de pe urma acestei contribuții este următoarea: pentru optimizarea performanțelor se recomandă implementarea acestor mecanisme direct peste substratul MAC.

Contribuția este localizată în: Capitolul 5
Publicații: [Rus10a], [Rus10b], [Pol09a], [Vin10], [Kis10], [Cor11]

5. Implementarea și evaluarea modului CLQ pentru rutarea conștientă de QoS

Mecanismul CLQ propus în cadrul acestei contribuții este utilizat în rutarea dinamică pentru optimizarea protocoalelor de dirijare a pachetelor prin impunerea unei metrici compozite, calculată în timp real. Această abordare presupune determinarea parametrilor QoS: ATR, OWD și BER pentru fiecare din legăturile dintre nodurile incluse în topologia de test. Implementarea s-a făcut în C/C++ sub distribuția Linux Debian. S-a demonstrat faptul că protocoalele de rutare (ex. RIP) pot evita congestia altfel nedetectată prin metodele de rutare clasice. De asemenea, în cazul în care se dorește activarea rutării multi-cale, CLQ poate ajuta la găsirea legăturilor optime din punct de vedere al statisticilor efectuate asupra parametrului ATR.

Contribuția este localizată în: Capitolul 6
Publicații: [Rus10c], [Rus10d], [Boa10]

6. Implementarea și evaluarea modului CLQ pentru routere MPLS

Această contribuție a fost prima realizată, din punct de vedere temporal, iar rolul ei a fost de a se demonstra fezabilitatea ideii de automatizare a ingineriei traficului bazată pe QoS. S-a ales tehnologia MPLS întrucât aceasta conține deja mecanisme de control al calității serviciilor. Parametrii obținuți astfel (ATR, OWD și BER) sunt grupați într-o metrică compozită care este publicată în întreg domeniul MPLS prin intermediul mecanismelor de distribuție a etichetelor. Simularea în OMNET++ a mecanismelor CLQ pe un set de routere MPLS demonstrează faptul că putem avea o îmbunătățire de aproximativ 20% a calității comunicației. Mecanismul poate fi îmbunătățit prin adăugarea unui sistem de management local pentru întreg domeniul, care să păstreze statistici referitoare la starea infrastructurii. Datele de performanță vor rămâne în continuare furnizate de către modulul de CLQ.

Contribuția este localizată în: Capitolul 7
Publicații: [Rus10b], [Bar11]

7. Îmbunătățirea algoritmului lui Dijkstra cu CLQ

O altă aplicabilitate a mecanismului CLQ este legată de eficientizarea algoritmului lui Dijkstra. Astfel, în loc să se oprească rularea algoritmului atunci când arborele Spanning-Tree al celei mai scurte căi a fost obținut, se permite rularea continuă a acestuia. Măsura este necesară pentru a evita conexiunile cu probleme, ținând cont de faptul că monitorizarea parametrilor ATR, OWD și BER este efectuată permanent. S-a demonstrat de asemenea că modificările efectuate sunt mai profunde decât simpla înlocuire a metricii folosite în cazul algoritmului Dijkstra clasic cu o metrică compozită. Astfel, sunt considerați în plus parametrii OWD și BER, ATR nu mai este considerat egal cu capacitatea legăturii, iar costul căii se stabilește luând în calcul cea mai mică valoare a ratei de transfer disponibilă, suma latențelor și BER global. Implementarea în OMNET++ a fost utilizată și pentru validarea formulei de calcul a metricii compozite. S-a demonstrat că întârzierea cap-la-cap obținută atunci când s-a utilizat algoritmul Dijkstra modificat poate fi semnificativ mai mică dacă exista o cale alternativă cu caracteristici superioare. De menționat că algoritmul clasic nu este capabil să detecteze căi de comunicație mai bune, decât în momentul în care se schimbă semnificativ starea legăturii (apar/dispar căi sau se schimbă tehnologia, dar în nici un caz nu poate detecta în timp real variații ale parametrilor QoS).

Contribuția este localizată în: Capitolul 8
Publicații: [Rus10b], [Rus10c]

8. Viitorul Internet de obiecte și dispozitive inteligente

Această contribuție se referă la integrarea informațiilor de calitate a serviciilor și a mecanismelor de rutare, în obiecte de informație (containere dedicate transportului de date ale utilizatorilor) care au capacitatea de a se auto-dirija optim într-o rețea compusă din dispozitive inteligente. S-au propus astfel concepte legate de viitorul Internet, în care se trece de la orientarea rețelelor pe dispozitive, la orientarea acestora pe obiecte de informație. În baza contribuțiilor prezentate anterior se propune ideea conform căreia mecanismul CLQ poate facilita colaborarea între obiectele de informație și dispozitivele inteligente.

Contribuția este localizată în: Capitolul 9
Publicații: [Rot09], [Nun09], [Gon10]

10.2 Remarci finale

Scopul acestei teze a fost evaluarea fezabilității ideii de a utiliza mecanisme de tip Cross-Layer pentru îmbunătățirea calității serviciilor transmise prin rețea. Munca a început cu evaluarea într-o primă fază a mecanismelor Cross-Layer existente în literatura de specialitate.

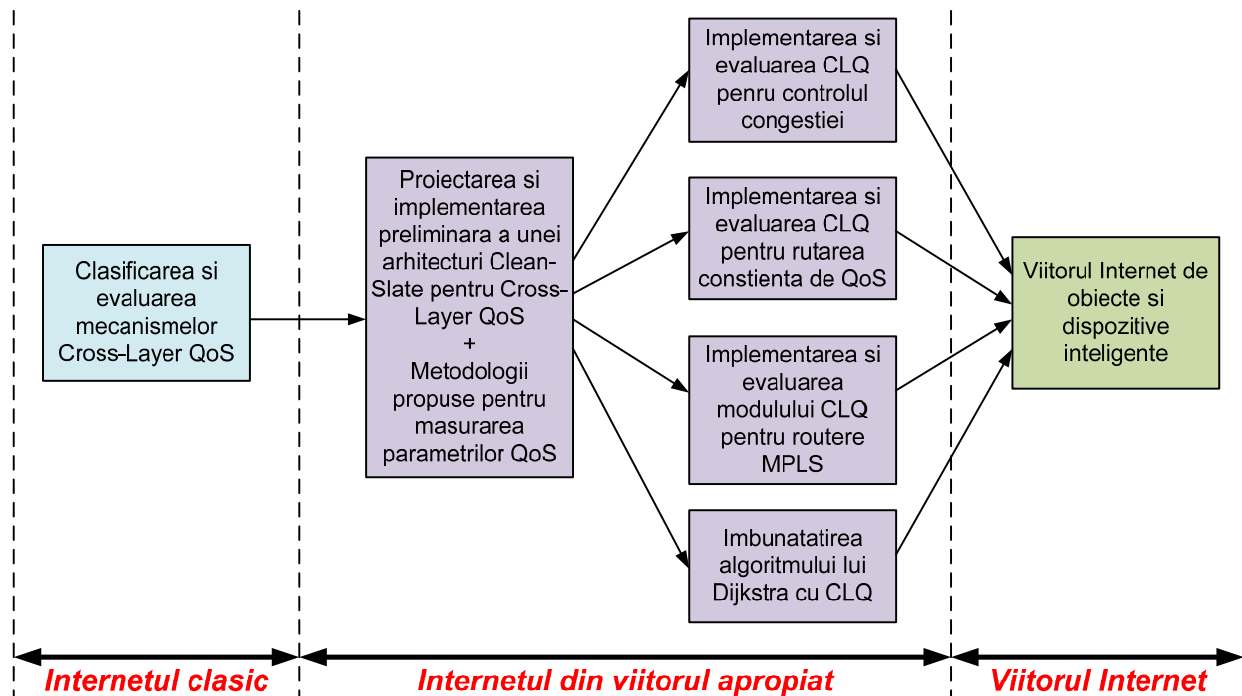


Figura 10-1 Structura acestei teze

În cea de-a doua fază a tezei, după proiectarea unui mecanism de tip Cross-Layer, au fost propuse o serie de aplicații unde s-a testat fezabilitatea ideii și s-au analizat îmbunătățirile aduse. Deoarece aceste aplicații foloseau infrastructura curentă a Internetului, le-am grupat în categoria Internetul viitorului apropiat (vezi Figura 10-1). Ce-a de-a treia și ultima fază a tezei este focalizată pe ilustrarea unor propuneri de schimbare radicală a întregii arhitecturi, astfel încât rețeaua viitorului să fie centrată pe obiecte inteligente de informație, obiecte inteligente de tip resursă și echipamente inteligente.

10.3 Premii obținute

- "Ericsson Awards of Excellence in Telecommunications", categoria Best Papers of Students in the Last Years of Studies, Ericsson Telecommunications Romania, București, Iulie 2007.
- "First Special Prize în cadrul Sesiunii de comunicări științifice în Electronică și Telecomunicații", organizată de către Facultatea de Electronică, Telecomunicații și Tehnologia Informației din cadrul Universității Tehnice din Cluj-Napoca, România, Mai 2007.

10.4 Publicații personale

Carti

- [Cor11] L.M.Correia, H.Abramowicz, M.Johnsson & K.Wünstel (editors), V.Dobrota, Zs.Polgar, **A.B. Rus** (included in list of contributors), “CLQ-Based Testbed used for Generic Path”, Chapter 12 “Prototype Implementations”, pp. 271-276, Architecture and Design for the Future Internet. 4WARD Project. Series: Signals and Communication Technology. 1st Edition, Springer Science + Business Media LLC, 2011, XXIX, 306 p., Hardcover, ISBN: 978-90-481-9345-5

Articole

- [Bar11] M.Barabas, G.Boanea, **A.B.Rus**, V.Dobrota, “Routing Management Based on Statistical Cross-Layer QoS Information Regarding Link Status”, *EuropeComm 2011*, 10-13 May 2011 - Budapest, Hungary, (submitted)
- [Boa10] G.Boanea, M.Barabas, **A.B.Rus** & V.Dobrota, “Design Principles and Practical Implementation of a Situation Aware Multipath Routing Algorithm”, *18th International Conference on Software, Telecommunications & Computer Networks IEEE SOFTCOM 2010*, Split-Bol (Island of Brac), September 23-25, 2010, Croatia, Print ISBN: 978-1-4244-8663-2, INSPEC Accession Number: 11637618, pp.321-325
- [Car10] J.Carapinha, R.Bless, Ch.Werle, K.Miller, V.Dobrota, **A.B.Rus**, H.Grob-Lipski & H.Roessler, “Quality of Service in the Future Internet”, *Proceedings of the 2010 ITU-T Kaleidoscope “Beyond the Internet? - Innovations for Future Networks and Services”*, Pune, India, 13-15 December 2010, 92-61-13171-9/CFP 1038E, S4.4, pp.1-8
- [Kis10] Zs.Kiss, Zs.Polgar, C.Vinti, M.Varga, **A.B.Rus** & V.Dobrota, “Network-Coding Based Congestion Control at Network Layer: Protocol Design and Evaluation”, *International Journal of Computer Networks & Communications (IJCNC)*, Vol.2, ISSN: 0974-9322 [Online]; 0975-2293 [Print], Academy & Industry Research Collaboration Center (AIRCC), 2010
- [Pol09] Zs.Polgar, Zs.Kiss, **A.B.Rus**, G.Boanea, M.Barabas, V.Dobrota, “Preliminary Implementation of Point-to-Multi-Point Multicast Transmission Based on Cross-Layer QoS and Network Coding”, *17th Int.Conf. on Software, Telecommunications & Computer Networks IEEE SOFTCOM 2009*, Split-Hvar, Croatia, September 24-26, 2009, Print ISBN: 978-1-4244-4973-6, INSPEC Accession Number: 10951348, pp.131-135
- [Rus08a] **A.B.Rus**, R.Serral Gracià, J.Domingo-Pascual & V.Dobrota, “WAN Emulator”, *Proceedings of the 7th International Conference “Communications 2008”*, Military Technical Academy, “Politehnica” University of Bucharest, “Electronica 2000” Foundation and IEEE Romania Section, Bucharest, Romania, June 5-7, 2008, pp.225-228, ISBN 978-606-521-008-0, Editura Printech 2008

- [Rus09a] **A.B.Rus**, V.Dobrota, "Overview of the Cross-Layer Paradigm Evolving Towards Future Internet", *ACTA TECHNICA NAPOCENSIS*, Electronics and Telecommunications, ISSN 1221-6542, Vol.50, No.2, 2009, pp.9-14
- [Rus10a] **A.B.Rus**, M.Barabas, G.Boanea, Z.Kiss, Z.Polgar, V.Dobrota, "Cross-Layer QoS and Its Application in Congestion Control", *17th IEEE Workshop on Local and Metropolitan Area Networks LANMAN 2010*, Long Branch, NJ, USA, May 5-7, 2010, ISSN: 1944-0367, Print ISBN: 978-1-4244-6067-0, INSPEC Accession Number: 11416542, Digital Object Identifier: 10.1109/LANMAN.2010.5507149, pp.1-6
- [Rus10c] **A.B.Rus**, V.Dobrota, A.Vedinas, G.Boanea, M.Barabas, "Modified Dijkstra's Algorithm with Cross-Layer QoS", *ACTA TECHNICA NAPOCENSIS*, Electronics and Telecommunications, ISSN 1221-6542, Vol.51, No.3, 2010
- [Rus10d] **A.B.Rus**, M.Barabas, G.Boanea & V.Dobrota, "Implementation of QoS-Aware Virtual Routers", *9th Edition of the International Symposium on Electronics and Telecommunications, ISETC 2010*, Timisoara, Romania, November 11-12, 2010, IEEE Catalog Number: CFP1003L-ART, ISBN: 978-1-4244-8460-7, 978-1-4244-8458-4, IEEE 2010, pp. 161-164
- [Vin10] C.Vinti, M.Varga, Zs.Polgar, **A.B.Rus**, V.Dobrota, "Signaling Protocol for Network Coding-Based Congestion Control at Network Layer", *9th RoEduNet International Conference "Networking in Education and Research"*, ARNIEC/RoEduNet Agency, IEEE Romanian Section, "Lucian Blaga" University of Sibiu, Sibiu, Romania, June 24-26, 2010, ISSN 2068-1038, Print ISBN: 978-1-4244-7335-9, pp. 304-309

Rapoarte de cercetare și livrabile

- [Gon10] A.Gonzales (editor), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., "Management Capabilities", "Network Coding Usage for GP Congestion Control", D-4.3 "In-Network Management Design", *FP7-ICT-2007-1-216041-4WARD- "Architecture and Design for the Future Internet"*, 27 January 2010, Revision: 1.0, pp. 24-25, 75-76, <http://www.4ward-project.eu/index.php?s=Deliverables>
- [Nun09] G.Nunzi, D.Dudkowski (editors), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., "Example of INM Framework Instantiation", D-4.2 In-Network Management Concept, *FP7-ICT-2007-1-216041-4WARD – "Architecture and Design for the Future Internet"*, 31 March 2009, Revision 2.0, pp.39-42, 115-112. <http://www.4ward-project.eu/index.php?s=Deliverables>
- [Nun10] G.Nunzi (editor), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., "Cross-Layer QoS and Generic Path for Real-Time Congestion Control", "INM Cross-Layer QoS", D-4.4 "In-Network Management System Demonstrator", *FP7-ICT-2007-1-216041-4WARD- "Architecture and Design for the Future Internet"*, 30 June 2010, Revision: 1.0, pp. 11-15, 27-31 <http://www.4ward-project.eu/index.php?s=Deliverables>

- [Rot09] R.Roth, F.Wolff, T.Zseby (editors), V.Dobrota, T.M.Blaga, **A.B.Rus** (included in list of contributors) et al., “Selected In-Network management Scenarios and Use Cases, Scenario 3: Home Networks”, D-4.1 “Definitions of Scenarios and Use Cases”, *FP7-ICT-2007-1-216041-4WARD-“Architecture and Design for the Future Internet”*, 31 March 2009, Revision: 2.0, pp. 48-60, <http://www.4ward-project.eu/index.php?s=Deliverables>
- [Wol10] F.Wolff (editor), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., “Reduced integration effort”, D-4.5 “Evaluation of the in-network management approach”, *FP7-ICT-2007-1-216041-4WARD-“Architecture and Design for the Future Internet”*, 11 June 2010, Revision: 1.0, pp. 19-21
<http://www.4ward-project.eu/index.php?s=Deliverables>

Rapoarte de doctorat

- [Rus08b] **A.B.Rus** “Cross-Layer Architectures for Quality of Services” (Arhitecturi cross-layer pentru calitatea serviciilor), Doctoral Research Report 1, *Technical University of Cluj-Napoca*, Romania, December 2008
- [Rus09b] **A.B.Rus** “Design Principles for Quality of Service in the Future Internet”, Doctoral Research Report 2, *Technical University of Cluj-Napoca*, Romania, September 2009
- [Rus10b] **A.B.Rus** “Minimal Implementation of Quality of Services in the Future Internet”, Doctoral Research Report 3, *Technical University of Cluj-Napoca*, Romania, July 2010

Proiecte de cercetarea în care am fost implicat

- FP7-ICT-2007-1 No. 216041 “4WARD – Architecture and Design for the Future Internet”, 2008-2010
- COST 290 “Wi-QoST: Traffic and QoS Management in Wireless Multimedia Networks”, 2004-2008

Bibliografie selectivă

- [Ah100] R.Ahlsvede, N.Cai, S.-Y.R.Li, R.W.Yeung, “Network Information Flow”, IEEE Trans. on Information Theory, 46(4): 1204–1216, 2000
- [Bar11] M.Barabas, G.Boanea, **A.B.Rus**, V.Dobrota, “Routing Management Based on Statistical Cross-Layer QoS Information Regarding Link Status”, *EuropeComm 2011*, 10-13 May 2011 - Budapest, Hungary, (submitted)
- [Bla98] S.Blake, D.Black, M.Carlson, E.Davies, Z.Wang, W.Weiss, “An Architecture for Differentiated Services”, *RFC2475*, December 1998
- [Boa10] G.Boanea, M.Barabas, **A.B.Rus** & V.Dobrota, “Design Principles and Practical Implementation of a Situation Aware Multipath Routing Algorithm”, *18th International Conference on Software, Telecommunications & Computer Networks IEEE SOFTCOM 2010*, Split-Bol (Island of Brac), September 23-25, 2010, Croatia, Print ISBN: 978-1-4244-8663-2, INSPEC Accession Number: 11637618, pp.321-325
- [Bra94] R.Braden, D. Clark, S.Shenker, “Integrated Services in the Internet Architecture: an Overview”, *RFC1633*, June 1994
- [Cao09] B.Q.Cao, B.Li, Q.M.Xia, “A Service-Oriented QoS-Assured and Multi-Agent Cloud Computing Architecture”, *Cloud Computing: First International Conference, CloudCom 2009*, Beijing, 2009
- [Car10] J.Carapinha, R.Bless, Ch.Werle, K.Miller, V.Dobrota, **A.B.Rus**, H.Grob-Lipski & H.Roessler, “Quality of Service in the Future Internet”, *Proceedings of the 2010 ITU-T Kaleidoscope “Beyond the Internet? - Innovations for Future Networks and Services”*, Pune, India, 13-15 December 2010, 92-61-13171-9/CFP 1038E, S4.4, pp.1-8
- [Cor11] L.M.Correia, H.Abramowicz, M.Johnsson & K.Wünstel (editors), V.Dobrota, Zs.Polgar, **A.B. Rus** (included in list of contributors), “CLQ-Based Testbed used for Generic Path”, Chapter 12 “Prototype Implementations”, pp. 271-276, *Architecture and Design for the Future Internet. 4WARD Project. Series: Signals and Communication Technology. 1st Edition*, Springer Science + Business Media LLC, 2011, XXIX, 306 p., Hardcover, ISBN: 978-90-481-9345-5
- [Dob10] V.Dobrota, “Switching and Routing Systems”, Courses 2010, UTCN,2010, <http://el.el.obs.utcluj.ro/scr/>
- [Dro08] D.Drogseth , “Quality of Experience: The Ultimate Collaboration – How Real Deployments are Succeeding and Why”, 2008
- [Gon10] A.Gonzales (editor), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., “Management Capabilities”, “Network Coding Usage for GP Congestion Control”, D-4.3 “In-Network Management Design”, FP7-ICT-2007-1-216041-4WARD-“Architecture and Design for the Future Internet”, 27 January 2010, Revision: 1.0, pp. 24-25, 75-76, <http://www.4ward-project.eu/index.php?s=Deliverables>

- [Gra05] F.Granelli, D.Kliazovich, “Cross-Layering for Performance Improvement in Multi-Hop Wireless Networks”, *Journal of Parallel Architectures, Algorithms and Networks, 8th International Symposium, ISPAN 2005*, Trento University, Italy, Dec. 2005
- [Hes09] B.Hestnes, P.Brooks, S.Heiestad, “QoE (Quality of Experience) – measuring QoE for improving the usage of telecommunication services”, *R&I Research Report, Telenor*, 2009
- [Kan05] Y.H.Kang; J.H.Lee, “The implementation of the premium services for MPLS IP VPNs”, *The 7th International Conference on Advanced Communication Technology, ICACT 2005*, Phoenix Park, Korea, July 2005, pp. 1107 - 1110
- [Kha06] S.Khan, Y.Peng, E.Steinbach, M.Sgroi, W.Keller, “Application-Driven Cross-Layer Optimization for Video Streaming over Wireless Networks”, *IEEE Communications Magazine*, Jan. 2006, pp. 122-130
- [Kis10] Zs.Kiss, Zs.Polgar, C.Vinti, M.Varga, **A.B.Rus** & V.Dobrota, “Network-Coding Based Congestion Control at Network Layer: Protocol Design and Evaluation”, *International Journal of Computer Networks & Communications (IJCNC), Vol.2*, ISSN: 0974–9322 [Online]; 0975-2293 [Print], Academy & Industry Research Collaboration Center (AIRCC), 2010
- [Koe03] R.Koetter, M.Medard, “An Algebraic Approach to Network Coding”, *IEEE/ACM Trans. on Networking*, 11:782–795, 2003
- [Mac06] J.Macfarlane, “Understanding IP Routing in Cisco Systems”, *Wiley Publishing*, 2006
- [Med07] D.Medhi, K.Ramasamy, “Network Routing Algorithms, Protocols, and Architectures”, *Morgan Kauffman Publishers*, 2007
- [Nun09] G.Nunzi, D.Dudkowski (editors), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., “Example of INM Framework Instantiation”, D-4.2 In-Network Management Concept, *FP7-ICT-2007-1-216041-4WARD – “Architecture and Design for the Future Internet”*, 31 March 2009, Revision 2.0, pp.39-42, 115-112.
<http://www.4ward-project.eu/index.php?s=Deliverables>
- [Nun10] G.Nunzi (editor), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., “Cross-Layer QoS and Generic Path for Real-Time Congestion Control”, “INM Cross-Layer QoS”, D-4.4 “In-Network Management System Demonstrator”, *FP7-ICT-2007-1-216041-4WARD-“Architecture and Design for the Future Internet”*, 30 June 2010, Revision: 1.0, pp. 11-15, 27-31
<http://www.4ward-project.eu/index.php?s=Deliverables>
- [Ohl09] B.Ohlman et. al., “D-6.1 First NetInf architecture description”, *FP7-ICT-2007-1-216041-4WARD– “Architecture and Design for the Future Internet”*, 2009
- [Pol09] Zs.Polgar, Zs.Kiss, **A.B.Rus**, G.Boanea, M.Barabas, V.Dobrota, “Preliminary Implementation of Point-to-Multi-Point Multicast Transmission Based on Cross-Layer QoS and Network Coding”, *17th Int. Conf. on Software, Telecommunications & Computer Networks IEEE SOFTCOM 2009*, Split-Hvar, Croatia, September 24–26, 2009, Print ISBN: 978-1-4244-4973-6, INSPEC Accession Number: 10951348, pp.131-135
- [Rad07] T. Radulescu, H.G. Coanda “*QoS in retelele IP multimedia*”, Editura Albastra, Cluj-Napoca, 2007, ISBN 978-973-650-219-4

- [Rap02] T.S.Rappaport, “Wireless Communications: Principles and Practice 2nd Edition”, Prentice Hall Communications Engineering and Emerging Technologies Series, 2002
- [Rot09] R.Roth, F.Wolff, T.Zseby (editors), V.Dobrota, T.M.Blaga, **A.B.Rus** (included in list of contributors) et al., “Selected In-Network management Scenarios and Use Cases, Scenario 3: Home Networks”, D-4.1 “Definitions of Scenarios and Use Cases”, FP7-ICT-2007-1-216041-4WARD-“Architecture and Design for the Future Internet”, 31 March 2009, Revision: 2.0, pp. 48-60, <http://www.4ward-project.eu/index.php?s=Deliverables>
- [Rus08a] **A.B.Rus**, R.Serral Gracià, J.Domingo-Pascual & V.Dobrota, “WAN Emulator”, *Proceedings of the 7th International Conference “Communications 2008”*, Military Technical Academy, “Politehnica” University of Bucharest, “Electronica 2000” Foundation and IEEE Romania Section, Bucharest, Romania, June 5-7, 2008, pp.225-228, ISBN 978-606-521-008-0, Editura Printech 2008
- [Rus08b] **A.B.Rus** “Cross-Layer Architectures for Quality of Services” (Arhitecturi cross-layer pentru calitatea serviciilor), Doctoral Research Report 1, Technical University of Cluj-Napoca, Romania, December 2008
- [Rus09a] **A.B.Rus**, V.Dobrota, “Overview of the Cross-Layer Paradigm Evolving Towards Future Internet”, *ACTA TECHNICA NAPOCENSIS, Electronics and Telecommunications*, ISSN 1221-6542, Vol.50, No.2, 2009, pp.9-14
- [Rus09b] **A.B.Rus** “Design Principles for Quality of Service in the Future Internet”, Doctoral Research Report 2, Technical University of Cluj-Napoca, Romania, September 2009
- [Rus10a] **A.B.Rus**, M.Barabas, G.Boanea, Z.Kiss, Z.Polgar, V.Dobrota, “Cross-Layer QoS and Its Application in Congestion Control”, *17th IEEE Workshop on Local and Metropolitan Area Networks LANMAN 2010*, Long Branch, NJ, USA, May 5-7, 2010, ISSN: 1944-0367, Print ISBN: 978-1-4244-6067-0, INSPEC Accession Number: 11416542, Digital Object Identifier: 10.1109/LANMAN.2010.5507149, pp.1-6
- [Rus10b] **A.B.Rus** “Minimal Implementation of Quality of Services in the Future Internet”, Doctoral Research Report 3, Technical University of Cluj-Napoca, Romania, July 2010
- [Rus10c] **A.B.Rus**, V.Dobrota, A.Vedinas, G.Boanea, M.Barabas, “Modified Dijkstra’s Algorithm with Cross-Layer QoS”, *ACTA TECHNICA NAPOCENSIS, Electronics and Telecommunications*, ISSN 1221-6542, Vol.51, No.3, 2010
- [Rus10d] **A.B.Rus**, M.Barabas, G.Boanea & V.Dobrota, “Implementation of QoS-Aware Virtual Routers”, 9th Edition of the International Symposium on Electronics and Telecommunications, ISETC 2010, Timisoara, Romania, November 11-12, 2010, IEEE Catalog Number: CFP1003L-ART, ISBN: 978-1-4244-8460-7, 978-1-4244-8458-4, IEEE 2010, pp. 161-164
- [Sri05] V.Srivastava, M.Motani, “Cross-Layer Design: A Survey and the Road Ahead”, *IEEE Communications Magazine*, Volume 43, Dec. 2005, pp. 112-119
- [Vas06] R.Vasiu, N.Robu, D.Andone, M.Bucos & M.Onita, “Integration of eLearning in Romanian Technical Universities”, In E. Pearson & P. Bohman (Eds.), *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2006* (pp. 121-126). Chesapeake, VA: AACE.

- [Vin10] C.Vinti, M.Varga, Zs.Polgar, **A.B.Rus**, V.Dobrota, “Signaling Protocol for Network Coding-Based Congestion Control at Network Layer”, *9th RoEduNetInternational Conference “Networking in Education and Research”*, ARNIEC/RoEduNet Agency, IEEE Romanian Section, “Lucian Blaga” University of Sibiu, Sibiu, Romania, June 24-26, 2010, ISSN 2068-1038, Print ISBN: 978-1-4244-7335-9, pp. 304-309
- [Vla08] Aurel Vlaicu (editor coordinator), Radu VasIU, Mugur Mocofan, Bogdan Orza, Tudor Samuila, Nicolae Muresan, Dorin Petreus, Niculaie Palaghita, Cristian Farcas, Virgil Dobrota, Tudor Blaga, Daniel Zinca, Marian Alexandru, Radu Arsinte, Mihai Romanca, Petre Ogrutan, Florin Sandu, Csaba Kertesz, Adrian Nedelcu, Paul Borza, *Intelligent Buildings. IT&C Systems, Technologies and Integrated Solutions*, UT Press Publishers, Cluj-Napoca 2008, ISBN: 978-973-662-397-4 (in Romanian)
- [Wan03] Q.Wang, M.A.Abu-Rgheff, “Cross-Layer Signaling for Next-Generation Wireless Systems”, *IEEE Wireless Communication and Network Conference*, New Orleans, LA, USA, Mar. 2003, pp. 1084-1089
- [Wol10] F.Wolff (editor), V.Dobrota, **A.B.Rus** (included in list of contributors) et al., “Reduced integration effort”, D-4.5 “Evaluation of the in-network management approach”, FP7-ICT-2007-1-216041-4WARD-“Architecture and Design for the Future Internet”, 11 June 2010, Revision: 1.0, pp. 19-21
<http://www.4ward-project.eu/index.php?s=Deliverables>
- [Xav07] Xavi Masip-Bruin et. all, “The EuQoS system: a solution for QoS routing in heterogeneous networks [Quality of Service based Routing Algorithms for Heterogeneous Networks]”, *Communications Magazine, IEEE*, Volume: 45 , Issue: 2, 2007, pp 96-103
- [Xio09] K.Xiong, H.Perros, “Service Performance and Analysis in Cloud Computing”, *2009 World Conference on Services*, Los Angeles, CA, USA, 6-10 July 2009
- [Yuk08] M.Yuksel, K.K.Ramakrishnan, R.D.Doverspike , “Cross-Layer Failure Restoration Techniques for a Robust IPTV Service”, *Proceedings of the 16th IEEE Workshop on Local and Metropolitan Area Networks LANMAN 2008*, Cluj-Napoca, Romania, September 3-6, 2008, pp.49-54, ISBN: 978-1-4244-2027-8
- [Zha05] Q.Zhang, F.Yang, W.Zhu, “Cross-Layer QoS Support for Multimedia Delivery over Wireless Internet”, *EURASIP Journal on Applied Signal Processing*, Jan. 2005, pp. 207-219



**UNIVERSITATEA
TEHNICĂ DIN CLUJ-
NAPOCA,
FACULTATEA DE
ELECTRONICĂ,
TELECOMUNICAȚII
ȘI TEHNOLOGIA
INFORMAȚIEI**



Curriculum vitae

Informații personale

Nume / Prenume	RUS, Andrei Bogdan
Adresă(e)	Str. Colinei, Nr.26, Ap.6, Cluj-Napoca, 400520, România
Telefon(oane)	Mobil: +40 744267970
E-mail(uri)	Bogdan.Rus@com.utcluj.ro
Naționalitate(-tăți)	Română
Data nașterii	28.07.1983
Sex	Masculin

Experiența profesională

Perioada	2007-prezent
Funcția sau postul ocupat	Doctorand cu frecvență
Activități și responsabilități principale	Activități didactice și de cercetare în domeniile: sisteme de comutație și rutare, protocoale pentru Internet
Numele și adresa angajatorului	Universitatea Tehnică din Cluj-Napoca
Perioada	Ian. 2008-Iun.2010
Funcția sau postul ocupat	Cercetător în cadrul proiectului FP7-ICT-2007-1 No. 216041 "4WARD – Architecture and Design for the Future Internet" finanțat de către UE.

Educație și formare

Perioada	Oct. 2007 – Iul. 2009
Diploma obținută	Diplomă Master
Domeniul	Telecomunicații și Tehnologii Multimedia
Numele și tipul instituției de învățământ	Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Universitatea Tehnică din Cluj-Napoca

Perioada **Oct. 2003 – Iul. 2006**
 Diploma obținută Certificat de absolvire a cursului de pregătire a personalului didactic
 Numele și tipul instituției de învățământ Departamentul Pentru Pregătirea Personalului Didactic, Universitatea Tehnică din Cluj-Napoca

Perioada **Oct. 2002 – Iul. 2007**
 Diploma obținută Diplomă de licență - Inginer
 Specializarea Telecomunicații
 Numele și tipul instituției de învățământ Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Universitatea Tehnică din Cluj-Napoca

Studii de specializare

Perioada **Oct. 2010 – Dec. 2008**
 Diploma obținută Curs post-universitar - Cisco Certified Network Associate (CCNA) Security
 Numele și tipul instituției de învățământ Academia locală Cisco, Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Universitatea Tehnică din Cluj-Napoca

Perioada **Oct. 2007 – Iul. 2008**
 Diploma obținută Curs post-universitar - Cisco Certified Network Associate (CCNA), Exploration 4.0
 Numele și tipul instituției de învățământ Academia locală Cisco, Facultatea de Electronică, Telecomunicații și Tehnologia Informației, Universitatea Tehnică din Cluj-Napoca

Activitate didactică

COLABORATOR AL CURSURILOR	PROGRAMUL DE STUDII	ANUL
Protocoale pentru Internet	Telecomunicații	IV
Sisteme de comutație și rutare	Telecomunicații	III

Aptitudini și competențe personale

Limba(i) maternă(e)
 Limba(i) străină(e) cunoscută(e)
 Auto evaluare
Nivel european ()*

Limba română

Limba engleză

Înțelegere				Vorbire				Scriere	
Ascultare		Citire		Participare la conversație		Discurs oral		Exprimare scrisă	
C1	Utilizator experimentat	C1	Utilizator experimentat	C1	Utilizator experimentat	C1	Utilizator experimentat	C1	Utilizator experimentat

Competențe și aptitudini organizatorice

Experiență în coordonarea studenților pe perioada efectuării lucrării de diplomă

Competențe și aptitudini tehnice - Cunoștințe medii/avansat a limbajelor de programare C și Java
 - Cunoștințe avansate de rețelistică: TCP/IP, arhitecturi client-server, QoS, protocoale de rutare
 - Cunoștințe medii de Linux (Fedora Core, Ubuntu, Debian)
 - Cunoștințe de baza în domeniul bazelor de date

Alte competențe și aptitudini - Experiența de lucru în echipe internaționale dobândită cu ocazia proiectului de cercetare FP7-4WARD

Publicații

- [Bar11]** M.Barabas, G.Boanea, **A.B.Rus**, V.Dobrota, "Routing Management Based on Statistical Cross-Layer QoS Information Regarding Link Status", *EuropeComm 2011*, 10-13 May 2011 - Budapest, Hungary, (trimis pentru recenzie)
- [Boa10]** G.Boanea, M.Barabas, **A.B.Rus** & V.Dobrota, "Design Principles and Practical Implementation of a Situation Aware Multipath Routing Algorithm", *18th International Conference on Software, Telecommunications & Computer Networks IEEE SOFTCOM 2010*, Split-Bol (Island of Brac), September 23-25, 2010, Croatia, Print ISBN: 978-1-4244-8663-2, INSPEC Accession Number: 11637618, pp.321-325
- [Car10]** J.Carapinha, R.Bless, Ch.Werle, K.Miller, V.Dobrota, **A.B.Rus**, H.Grob-Lipski & H.Roessler, "Quality of Service in the Future Internet", *Proceedings of the 2010 ITU-T Kaleidoscope "Beyond the Internet? - Innovations for Future Networks and Services"*, Pune, India, 13-15 December 2010, 92-61-13171-9/CFP 1038E, S4.4, pp.1-8
- [Kis10]** Zs.Kiss, Zs.Polgar, C.Vinti, M.Varga, **A.B.Rus** & V.Dobrota, "Network-Coding Based Congestion Control at Network Layer: Protocol Design and Evaluation", *International Journal of Computer Networks & Communications (IJNC)*, Vol.2, ISSN: 0974-9322 [Online]; 0975-2293 [Print], Academy & Industry Research Collaboration Center (AIRCC), 2010
- [Pol09]** Zs.Polgar, Zs.Kiss, **A.B.Rus**, G.Boanea, M.Barabas, V.Dobrota, "Preliminary Implementation of Point-to-Multi-Point Multicast Transmission Based on Cross-Layer QoS and Network Coding", *17th Int.Conf. on Software, Telecommunications & Computer Networks IEEE SOFTCOM 2009*, Split-Hvar, Croatia, September 24-26, 2009, Print ISBN: 978-1-4244-4973-6, INSPEC Accession Number: 10951348, pp.131-135
- [Rus08a]** **A.B.Rus**, R.Serral Gracià, J.Domingo-Pascual & V.Dobrota, "WAN Emulator", *Proceedings of the 7th International Conference "Communications 2008"*, Military Technical Academy, "Politehnica" University of Bucharest, "Electronica 2000" Foundation and IEEE Romania Section, Bucharest, Romania, June 5-7, 2008, pp.225-228, ISBN 978-606-521-008-0, Editura Printech 2008

- [Rus09a]** **A.B.Rus**, V.Dobrota, "Overview of the Cross-Layer Paradigm Evolving Towards Future Internet", *ACTA TECHNICA NAPOCENSIS*, Electronics and Telecommunications, ISSN 1221-6542, Vol.50, No.2, 2009, pp.9-14
- [Rus10a]** **A.B.Rus**, M.Barabas, G.Boanea, Z.Kiss, Z.Polgar, V.Dobrota, "Cross-Layer QoS and Its Application in Congestion Control", *17th IEEE Workshop on Local and Metropolitan Area Networks LANMAN 2010*, Long Branch, NJ, USA, May 5-7, 2010, ISSN: 1944-0367, Print ISBN: 978-1-4244-6067-0, INSPEC Accession Number: 11416542, Digital Object Identifier: 10.1109/LANMAN.2010.5507149, pp.1-6
- [Rus10c]** **A.B.Rus**, V.Dobrota, A.Vedinas, G.Boanea, M.Barabas, "Modified Dijkstra's Algorithm with Cross-Layer QoS", *ACTA TECHNICA NAPOCENSIS*, Electronics and Telecommunications, ISSN 1221-6542, Vol.51, No.3, 2010
- [Rus10d]** **A.B.Rus**, M.Barabas, G.Boanea & V.Dobrota, "Implementation of QoS-Aware Virtual Routers", *9th Edition of the International Symposium on Electronics and Telecommunications, ISETC 2010*, Timisoara, Romania, November 11-12, 2010, IEEE Catalog Number: CFP1003L-ART, ISBN: 978-1-4244-8460-7, 978-1-4244-8458-4, IEEE 2010, pp. 161-164
- [Vin10]** C.Vinti, M.Varga, Zs.Polgar, **A.B.Rus**, V.Dobrota, "Signaling Protocol for Network Coding-Based Congestion Control at Network Layer", *9th RoEduNet International Conference "Networking in Education and Research"*, ARNIEC/RoEduNet Agency, IEEE Romanian Section, "Lucian Blaga" University of Sibiu, Sibiu, Romania, June 24-26, 2010, ISSN 2068-1038, Print ISBN: 978-1-4244-7335-9, pp. 304-309
- [Cor11]** L.M.Correia, H.Abramowicz, M.Johnsson & K.Wüstel (editors), V.Dobrota, Zs.Polgar, **A.B. Rus** (included in list of contributors), "CLQ-Based Testbed used for Generic Path", Chapter 12 "Prototype Implementations", pp. 271-276, *Architecture and Design for the Future Internet. 4WARD Project. Series: Signals and Communication Technology. 1st Edition*, Springer Science + Business Media LLC, 2011, XXIX, 306 p., Hardcover, ISBN: 978-90-481-9345-5