

**RĪGAS TEHNISKĀ UNIVERSITĀTE**  
Elektronikas un Telekomunikāciju fakultāte  
Telekomunikāciju institūts

**Jans JEĻINSKIS**

Doktora studiju programmas „Telekomunikāciju un datoru tīkli” doktorants

**FAZILOĢIKAS BALSTĪTA PIEKĻUVES  
VADĪBA MPLS-TE/GMPLS TĪKLOS**

**Promocijas darbs**

Zinātniskais vadītājs  
Dr.sc.ing., profesors  
**G.LAUKS**

**Rīga 2011**

## Anotācija

Šī promocijas darba mērķis bija izveidot jaunu fazi-CAC projektēšanas metodi un novērtēt konstruētā fazi-CAC risinājuma pielietojuma iespējas MPLS-TE (GMPLS) trafika pārraides/vadības sistēmā, modificējot RSVP-TE protokola darbību, šādā veidā nodrošinot dinamisko un selektīvo LSP iestatīšanu. Balstoties uz darbā definēto fazi-CAC fazi izvedumu sistēmas (FIS) modeli un MPLS-TE tīkla modeli, kā arī vairāku lielu starptautisko sakaru pakalpojumu sniedzēju SLA, ar heuristisku pētījuma metodi tiek atrasta fazi izvedumu sistēma, kas nodrošina vēlamu fazi-CAC veikspēju. FIS heuristiskai izvēlei tiek izmantots MPLS-TE modeļa simulācijas rīks. Darba izstrādes gaitā tika izveidots MPLS-TE eksperimentāls tīkls uz Cisco 2800 sērijas maršrutētāju bāzes, maršrutētāja saskarnes un vadības modulis, fazi-CAC kontroles modulis RSVP-TE protokola vadībai, kā arī trafika ģenerācijas un analīzes moduļi.

Rezultātā ir pierādīta fazi-CAC efektivitāte salīdzinājumā ar klasisko sliekšņa-CAC algoritmu, kas tiek pielietots RSVP-TE vadībai MPLS-TE tīklos.

Promocijas darbs balstās uz eksperimentāliem rezultātiem fazi-CAC pielietošanā RSVP-TE protokola vadībai MPLS-TE tīklā.

Darba apjoms – 174 lpp, 90 attēli, 20 tabulas, 2 pielikumi un 148 literatūras avoti.

## Annotation

This thesis aimed to create a new fuzzy-CAC design method, as well to evaluate the possible application of built fuzzy-CAC solution in the MPLS-TE (GMPLS) traffic transmission/management system, modifying the RSVP-TE protocol, thus providing a dynamic and selective LSP setup. Based on the defined fuzzy-CAC fuzzy inference system (FIS) model and the MPLS-TE network model, as well as several major international telecommunications service provider SLAs, by using the heuristic research method, FIS is found, which ensures the desired fuzzy-CAC performance. For FIS heuristic selection MPLS-TE model simulation tool is used. Within the framework of this thesis, MPLS-TE experimental network was made on the basis of a Cisco 2800 series routers, as well as the router interface and control modules, fuzzy-CAC control module for RSVP-TE protocol management, traffic generation and analysis modules.

As the result, fuzzy-CAC effectiveness is proved, if compared to the classic threshold-CAC algorithm that is currently being used in RSVP-TE for MPLS-TE network management.

The doctoral thesis is based on the experimental results of fuzzy-CAC application to the RSVP-TE protocol, while managing traffic in MPLS-TE network. The amount of work is 174 pp., 90 figures, 20 tables, 2 appendices and 148 sources of literature.

## Saīsinājumu saraksts

ACAC	ARROWS Admission Control	ARROWS slēguma vadība
ASON	Automatically Switched Optical Network	Automātiski komutējama optiskais tīkls
ATM	Asynhronous Transfer Mode	Asinhronās pārsūtīšanas režīms
BADD	Basic Defuzzification Distributions	Pamata defazifikācijas izkārtojums
B-ISDN	Broadband Integrated Services Digital Networks	Platjoslas Integrētais pakalpojumu cipartīkls
CAGR	Compound Annual Growth Rate	Salikta Gada Pieauguma Intensitāte
CBR	Constant Bit Rate	Konstants Bitu Ātrums
CDD	Constraint Decision Defuzzification	Ierobežotā lēmuma defazifikācija
CLI	Command-line interface	Komandrindas interfeiss
COG	Center of gravity	Massas centrs
CR-LDP	Constraint based Label Distribution Protocol	Ierobežojumu balstīts izdales protokols
DiffServ	Differentiated services	Diferencētie pakalpojumi
DSR	Dynamic Source Routing	Dinamiska avota maršrutēšana
EQM	Extended quality method	
fBm	Fractional Brownian Motion	Frakcionālā Brauna kustība
FAOM	Fuzzy Adaptive Optimized Marking	Fazi adaptīvā optimizētā marķēšana
FCD	Fuzzy clustering defuzzification	Fazi klasteru defazifikācija
FERM	Fuzzy explicit rate marking	Fazi precīzās intensitātes marķēšana
FM	Fuzzy mean	Fazi vidējais
FOM	First of maximum	Pirmais no maksimuma
GLSD	Generalized level set defuzzification	Vispārinātā līmeņa kopas defazifikācija
GMPLS	Generalized Multiprotocol Label Switching	Vispārināta daudzprotokolu iezīmju komutācija
ICAC	Interference Admission Control	Interferences pieejas vadība
ICMP	Internet Control Message Protocol	Interneta vadības ziņojumu protokols
ICOG	Indexed center of gravity	Indeksētais masas centrs
IETF	Internet Engineering Task Force	Interneta tehniskā uzdevumgrupa
IntServ	Integrated Services	Integrētie pakalpojumi

IP	Internet Protocol	Interneta Protokols
IPDV	IP Packet Delay Variation	IP pakešu aizturu variācija
ISP	Internet Service Provider	Interneta piekļuves sniedzējs
ITU-T	International Telecommunication Union -Telecommunication Standardization Sector	Starptautiskās Telesakaru apvienības komiteja
LAN	Local Area Network	Lokālais datoru tīkls
LDP	Label Distribution Protocol	Iezīmju izdales protokols
LER	Label Edge Router	Robežapgabala iezīmes komutējamā ceļa maršrutētājs
LOM	Largest of maximum	Lielākais no maksimuma
LSP	Label Switched Path	Pēc iezīmēm komutējams ceļš
LRD	Long-range dependency	Ilgatkārība
LSR	Label Switch Router	Iezīmes komutējamā ceļa maršrutētājs
MeOM	Mean of maxima	Vidējais no maksimuma
MIB	Management Information Bases	Vadības informācijas bāze
MIMO	Multiple Input Multiple Output	Daudziedadu un daudzizvadu sistēma
MISO	Multiple Input Single Output	Daudziedadu un viena izvada sistēma
MOM	Medium of maximum	Maksimuma vidējais
MPLS	Multiprotocol Label Switching	Daudzprotokolu iezīmju komutācija
NGN	Next Generation Network	Nākamās paaudzes tīkls
NPCAC	Nonpredictive call admission control	Neprognozējamā izsaukumu pieejas vadība
P/Invoke	Platform Invocation Services	Platformas izsaukuma serviss
P2P	Peer-to-Peer	Vienādranga tīklošana
QM	Quality Method	Kvalitātes metode
QoS	Quality of Service	Servisa kvalitāte
RBWB	Residual Band With Blocking	Atlikušā caurlaides josla ar bloķēšanu
RCOM	Random choice of maximum	Maksimuma gadījuma izvēle
RED	Randon Early Detection	Gadījuma agrā detektēšana
RPCAC	Received power-based call admission control	Uz saņemto jaudu balstīta izsaukuma pieejas vadība
RTSP	Real Time Streaming Protocol	Reāllaika straumēšanas protokols
RTT	Round trip time	Aprite laiks
RTTMON	Round-Trip Time Monitor	Aprite laika monitors
SLA	Service Level Agreements	Līgums par pakalpojumu līmeni
SLIDE	Semi-linear defuzzification	Daļējilineārā defazifikācija

SMR	Split multipath routing	Sadalīta daudzceļu maršrutēšana
SOM	Smallest of maximum	Mazākais no maksimuma
TCP	Transmission Control Protocol	Pārraides vadības protokols
UDP	User Datagram Protocol	Lietotāja datagrammu protokols
VBR	Variable Bit Rate	Mainīgs bitu ātrums
VoD	Video on demand	Video pēc pasūtījuma
VoIP	Voice over IP	IP balss pārraide
WFM	Weighted fuzzy mean	Svērtais fazi vidējais
WoB	Without Blocking	Bez bloķēšanas

## Saturs

IEVADS .....	8
Disertācijas zinātniskais jauninājums .....	15
Pētījumu sfēra un ierobežojumi .....	16
Pielietotās pētījumu metodes .....	22
Svarīgākās definīcijas un autora pieņēmumi .....	24
1. Faziloģika mūsdienu trafika vadīšanā .....	26
1.1. Mūsdienu sakaru sistēmu vadības problēmas .....	26
1.2. Faziloģikas pamatkonceptija .....	31
1.3. Ieskats faziloģikas attīstībā .....	32
1.4. Faziloģikas izvedumu sistēmas uzbūve .....	34
1.4.1. Piederības funkciju deklarācijas .....	34
1.4.2. Ieejas mainīgo fazifikācija .....	38
1.4.3. Pamata faziloģiskās operācijas un izvedumi .....	39
1.4.4. Faziloģikas izvedumu sistēmas tipi .....	40
1.4.5. Izvada mainīgo defazifikācija .....	41
1.5. Faziloģikas pielietojums telekomunikācijās .....	44
1.5.1. Faziloģikas pielietojums maršrutēšanā ar QoS nodrošinājumu .....	44
1.5.2. Pārslodzes kontrole un AQM, TCP/IP tīklos ar faziloģiku .....	45
1.6. Fazi – CAC algoritmi. Esošās realizācijas .....	46
1.6.1. Fazi-CAC ATM sakaru sistēmās .....	46
1.6.2. Fazi-CAC CDMA sakaru sistēmās .....	47
1.6.3. Fazi-CAC GPRS sakaru sistēmās .....	49
1.6.4. Fazi-CAC MPLS sakaru sistēmās .....	50
2. Jauns fazi-CAC risinājums MPLS-TE/GMPLS sakaru sistēmām .....	54
2.1. MPLS-TE sakaru sistēmas atsauču modelis .....	54
2.1.1. MPLS-TE sakaru sistēmas modeļa pamatelementu definīcijas .....	56
2.1.2. MPLS-TE sakaru sistēmas modeļu raksturojošo parametru definīcijas .....	58
2.1.3. MPLS-TE eksperimentālā tīkla modeļa notikumu kopa .....	61
2.2. Fazi-CAC atsauču modelis .....	62
2.2.1. Fazi-CAC modeļa pamatelementi .....	62
2.2.2. Fazi-CAC ieejas parametru kopa .....	63
2.2.3. Fazi-CAC modeļa izejas parametru kopa .....	65
2.3. Fazi-CAC modeļa ieejas parametru fazifikācija .....	66
2.4. Fazi-CAC modeļa IF-THEN kārtulu zināšanu bāze .....	66

2.5.	Fazi-CAC modeļa stāvokļu kopa .....	68
2.6.	Pielietotā fazi-CAC pseidokods .....	69
2.7.	Fazi-CAC lēmējsistēmas uzbūves heuristiskā izvēle .....	70
2.7.1.	Fazi-CAC veiktspējas mērķu definīcija .....	71
2.7.2.	QoS parametru degradācijas ierobežojumu definīcijas .....	74
2.7.3.	Heuristiskās pētījuma metodes apraksts .....	80
2.7.4.	MPLS-TE sakaru sistēmas modeļa simulācijas vide fazi-CAC lēmējsistēmas ātrai prototipēšanai .....	81
2.7.5.	Ieejas parametru fazifikācijas līkņu definīcijas .....	86
2.7.6.	Ieejas un izejas parametru lingvistisko mainīgo piederības funkciju deklarācijas .....	87
2.7.7.	Pielietotā IF-THEN kārtulu zināšanu bāze .....	93
2.7.8.	Defazifikācijas metodes izvēle .....	98
2.7.9.	Fazi izvedumu sistēmas veida izvēle .....	101
2.7.10.	Fazi-CAC lēmuma pēcapstrāde .....	105
2.7.11.	Papildu resursu rezervēšanas koeficienta vērtības izvēle .....	107
3.	Fazi-CAC risinājuma eksperimentālā pārbaude .....	108
3.1.	Praktiskā fazi-CAC algoritma realizācija .....	108
3.1.1.	Praktiskās fazi-CAC realizācijas MPLS-TE eksperimentālā tīkla apraksts .....	108
3.1.2.	Eksperimenta gaita .....	114
3.1.3.	Tīkla QoS parametru mērījumi ar Cisco IOS IP SLA .....	116
3.1.4.	Analizējamo parametru apskats .....	118
3.1.5.	Eksperimentu scenāriju apraksts .....	120
3.1.6.	Eksperimenta rezultātu attēlojums .....	121
3.2.	Eksperimenta rezultāti – optimistiskais scenārijs .....	122
3.3.	Pesimistiskais scenārijs – rezultāti .....	134
4.	Fazi-CAC adaptācijas mehānismi .....	147
4.1.	Piederības funkciju deklarāciju modifikācijas .....	148
4.2.	Lēmuma pieņemšanas sliekšņa modifikācijas .....	157
5.	Secinājumi .....	161
	LITERATŪRA .....	166
	Pielikumi .....	175

## IEVADS

Sakaru tīkli attīstās ļoti strauji un tehnoloģiskā ziņā kļūst aizvien sarežģītāki, lai nodrošinātu nemītīgi augošo tīklu lietotāju pieprasījumu pēc resursu izdales. Tīkla prasības pēc adaptācijas spējas pie trafika uzliesmojumiem un neprognozējamajiem tīkla resursu pieprasījumiem, kā arī ļoti striktas sakaru sesiju drošības un drošuma prasības no lietotāju aplikāciju puses rada aizvien jaunus izaicinājumus trafika vadības jomā. Mūsdienu sakarus lielākoties nodrošina interneta tīkla servisu konverģences spēja [109] [55]. Tā, no vienas puses, ļauj sniegt ļoti plašu pakalpojumu spektru lietotājiem caur vienu globālu tīmekli, bet no otras puses, pieprasa arī implementēt tīkla attīstības redzējumā jaunus trafika vadības paņēmienus. Šiem paņēmieniem ir jānodrošina atsevišķiem servisa tipiem specifiskas QoS garantijas, kur katram servisam ir iespēja „diktēt” tīklam pavisam atšķirīgas apkalpošanas prasības un saņemt „gals-gals” sakaru sesijas attiecīgu apkalpošanu.

Trafika ģenerējošās multimediju aplikācijas ar katru dienu aizvien vairāk kļūst jutīgākas pret tīkla veiktspēju raksturojošiem parametriem, tādiem kā stabili pieejama caurlaides josla, trīce un pakešu aiztures tīklā. Tiešsaistes pakalpojumi kļūst par miljonu cilvēku ikdienu un milzīgi skaitļošanas centri un datu glabātavas pārņem skaitļošanas funkcijas no lietotāju personāliem datoriem. Tas noved pie atgriešanās pie kādreiz esošās centralizētās skaitļošanas, kad pie lielas ESM tika pieslēgti termināļi, un iekšējais tīkls nodrošināja informācijas ievadu un izvadu [79], [43]. Mūsdienu tiešsaistes servisi iekļauj sevī ne tikai e-pasta sūtījumus, bet arī tiešsaistes darbu ar teksta dokumentiem un grafiskiem failiem. Šādu pakalpojumu piedāvā, piemēram, Google docs, kas pielieto lielas jaudas serverus datu apstrādei un glabāšanai. Lietojot šāda tipa pakalpojumus, ātrs tīkla savienojums nodrošina tiešsaistes darba iespējas ar parastajiem ofisa dokumentiem caur pārlūkprogrammas interfeisu. Arī video un grafiskos failus lietotāji aizvien biežāk izvēlas glabāt publiski pieejamās vietnēs, kur to ir iespējams piedāvāt apskatei citiem tīkla lietotājiem, izmantojot daudzus sociālos tīklos radītās sasaites starp neskaitāmiem lietotājiem. Šai tendencei pamatā ir strauja ātrgaitas piekļuves tīklu attīstība uz tādu tehnoloģisku risinājumu kā FTTx un PON [17], [14], [15], [37], [50], [16], [21] pamata, kas dos iespēju pārnest skaitļošanas jaudas uz lielo centrālo serveru „pleciem”, bet ātrgaitas tīkla pieslēgums nodrošinās tiešsaistes piekļuvi pie vēlamā satura.

Nākamais aspekts, kas ir jāņem vērā, – interneta tīkls kļūst aizvien „impulsīvāks” – milzīgas tīkla caurlaides spējas un galiekārtu skaitļošanas kapacitātes kopā ar nemitīgi augošajiem datu pārraides ātrumiem un interneta pieejamību galalietotājiem spēj tīklu padarīt ļoti ievainojamu – tas ne tikai vienā acumirklī izplata datorvīrusu infekcijas, bet arī momentāni reaģē uz sociāliem notikumiem pasaulē. Kā piemēru varētu minēt nesenu pasauleslavena mūziķa Maikla Džeksona nāvi 2009. gada jūnijā, kad interneta meklētāja Google vadības algoritmi bija spiesti atslēgt lielu daļu servisu, jo pārdzīvoja īstu lietotāju „uzbrukumu” ar meklēšanas operāciju pieprasījumiem un tīklā nekontrolējami pieauga tiešsaistes audio un video satura apjomu sūtījumi [68], [69], [67]. Tika pārslogots arī Twitter mikroblogošanas tīkls, atsakot saviem lietotājiem piekļuvi pie servisiem. CNN ziņu serveri bija tuvu kraham [56].

Arī pavisam neseni notikumi, tādi kā prinča Viljama un Keitas Midltones kāzas 2011. gada 26. aprīlī un Osamas bin Ladenas nāve 2011. gada 2. maijā, izraisīja negaidīti straujus trafika plūsmu uzliesmojumus tīmeklī, momentāni pārslogojot Google meklētāju un Youtube tiešsaistes video servisu.

Pat nepieminot neparedzamus hakeru uzbrukumus tīklā vai vīrusu inficētu datoru mēstuļu nekontrolējamo izsūtīšanu, šodien varam sastapties ar citiem, vairāk eksotiskiem trafika uzliesmojuma cēloņiem. Piemēram, Eiropvīzijas laikā, kad miljoniem cilvēku momentāni ieslēdz „peer-to-peer” tīklu aplikācijas un sāk gandrīz režģtīkla režīmā sūtīt viens otram mūziķu uzstāšanos ierakstus vai arī sāk pārslogot tiešsaistes video serverus ar neskaitāmiem pieslēgumu mēģinājumiem, tad runāt par šādu notikumu paredzēšanu un tīkla adekvātiem lēmumiem, pārdalot resursus starp lietotāju aplikācijām, būtu ļoti grūti.

Attīstot šo domu tālāk, gribētos pieminēt vēl dažus faktus, kas parāda tīmekļa esošo stāvokli un tajā notiekošo procesu dinamiku. Interneta lietotāju skaits 2011. gada sākumā jau pārsniedza 1,96 miljardus cilvēku, kas savās aktivitātēs tikai 2010. gada laikā radīja tīklā 21.4 miljonus jaunu interneta vietņu, kuru kopējais skaits jau tuvojās 260 miljoniem. Pērn tika nosūtīti arī 107 triljoni e-pasta sūtījumu, no kuriem vidēji 89,1 % bija surogātpasta sūtījumi, kas sastāda apmēram 262 miljardus mēstuļu dienā! Tikai viens tiešsaistes video serviss Youtube apkalpo vairāk nekā divus miljardus video failu dienā, no kuriem vairāk nekā 100 miljoni tiek skatīti, pielietojot mobilas iekārtas. Tās pašas dienas laikā vairāk nekā 200 000 datoru kļūst par t.s. botu tīklu „zombijiem” un izsūta surogātpastu, kas satur t.s. „Trojas kodus” un vīrusus [48], [117].

Kaut arī tīmekļa resursi šķiet milzīgi, ir jāatzīst, ka globāls tīkls savā būtībā ir viegli ievainojams un veseli tā apgabali momentāni var tikt degradēti, kad tīkla elementi tiek pārslogoti ar ģenerētu datu negaidītiem uzliesmojumiem vai vīrusu nekontrolētās izplatīšanās gadījumā.

Viss iepriekš minētais dod pamatu secināt, ka jebkāda tipa trafika pieprasījuma pat ilglaicīgas prognozes, kas balstās uz pieņēmumu par trafika stacionāro dabu, agri vai vēlu var dot negatīvu efektu, jo mūsdienu tīklos prognozēšanas pamatā bieži tiek izmantoti vēsturiski tīkla uzvedības dati un trafika modeļi, kas būtībā neņem vērā trafika dziļi nestacionāro dabu [18], [19].

Pieņemsim situāciju, kad tīkla resursu pārdales algoritmi, balstoties uz prognozēšanas rezultātiem, organizē linku rezervēšanu, maršrutēšanas ceļu preferenču tabulas izmaiņas utt. Tas viss tiek darīts ar pieņēmumu par trafika stacionāro dabu, kur tendences un to apvienojums sezonālītātē deklarē tīkla uzvedības pamatus un lēmumu pieņemšanas politiku izmaiņas. Šāda sistēma ir stabila tikai līdz brīdim, kad viss notiek saskaņā ar plānu un trafiks neizrāda negaidītus uzliesmojumus vai aktivitātes pazemināšanos. Šinī situācijā tīkla veiktspēja nopietni degradēsies, jo visi efektivitātes kritēriji tika pakārtoti prognozējamajām vērtībām.

Tīklu vadība, kas impulsīvi reaģē uz lietotāju uzvedības šabloniem un izrāda nestacionāru uzvedību, ir ļoti sarežģīts uzdevums. Tas prasa pavisam jaunu tīkla vadības koncepciju.

Par vienu no mūsdienu sakaru tīklu vērā ņemamiem apdraudējumiem uzskata arī elementāru tīkla linku neoptimālu utilizāciju, kas ir tieši saistīta ar problēmu par tīkla resursu pārdalīšanu. Mūsdienu sakaru tīklu posmi ļoti bieži ir vai nu pārslogoti, radot grūtības attiecīgu QoS parametru uzturēšanai mūsdienu aplikācijām, vai arī atrodas minimālās noslogotības stāvoklī, radot zaudējumus tīkla uzturētājiem [99], [92], [6], [84], [115]. Notiek arī plašas diskusijas par tīkla resursu optimālu vadību tīklā [116] [38]. Faktiski linku neoptimālās noslogotības jautājumi ir cieši saistīti ar trafika „uzvedības” nenoteiktību, jo vadīt trafika resursu sadali nenoteiktības apstākļos ir ļoti sarežģīts uzdevums.

Tātad, lai nodrošinātu tīkla resursu dinamisku vadību, tīkla adaptācijas mehānismu darbību un sistēmas kopēju robustumu, ir jāmaina tīkla vadības politika, kas pagaidām lielākoties balstās uz TCP/IP protokolu saimes un papildus pielieto AQM algoritmus. Kā vienu no progresīvākajām mūsdienu trafika vadības tehnoloģijām varētu nosaukt aizvien plašāku popularitāti iekarojošo MPLS risinājumu, kas spēj implementēt arī, piemēram,

DiffServ vai IntServ vadību. Šodien tiek attīstīta arī ITU-T vīzija par GMPLS tīkliem, kas jau savā pamatā ir balstīta uz ideju par inteligentu un dinamisku tīkla resursu vadību, nodrošinot LSP iestatīšanu pēc galalietotāja aplikācijas pieprasījuma. Iepriekš aprakstīto risinājumu pamatā ir ideja par trafika plūsmu vadību, balstoties uz iepriekš definētām QoS klasēm. QoS klašu izvirzīto prasību nodrošināšanai plūsmas tiek vadītas pa dinamiski iestatītiem ceļiem, kur katrs links nodrošina attiecīgu resursu apjomu un kvalitāti. Šādi risinājumi, pagaidām ir nākotnes vīzijā, un šobrīd notiek aktīvs pētniecisks darbs, kur tiek meklēti mehānismi, kas būtu spējīgi nodrošināt dinamisku tīkla resursu vadību, vienlaicīgi nodrošinot servisiem nepieciešamo QoS līmeni tīklā.

Katrā ziņā globālu un efektīvu izmaiņu pamatā var būt vienīgi kompleksā tīkla vadības politikas maiņa, kas balstītos uz esošās situācijas analīzi un globālā tīmekļa attīstības nākotnes vīzijām.

Tagad pievērsīsimies jautājumiem, kas tieši skar trafika vadības pamatelementus – lēmuma pieņemšanas stratēģijai un pašiem lēmēj mehānismiem.

Jebkādas izmaiņas tīklā notiek vai nu uz ekspertu zināšanām balstītiem lēmumiem, kuras tiek realizētas tīklu projektēšanas un plānošanas gaitā, vai arī uz tīklā implementētu lēmējiekārtu ātri veiktajiem lēmumiem. Pirmajā gadījumā tiek lemts par ilgtspējīgu tīkla infrastruktūras attīstību, un šādu lēmumu pamatā ir ilgtermiņa attīstības vīzijas, vēstures datu analīze, prognozes un dažādu attīstības scenāriju simulācijas. Otrajā gadījumā tie ir tādi lēmumi kā CAC kontrole, ātra maršrutēšanas ceļu izvēle, dinamisku tuneļu izveide caur dažādu arhitektūru tīkliem, piem., LSP caur MPLS vai nākotnē caur GMPLS.

Izejot no iepriekš aprakstīta lēmumu pieņemšanas scenārija, balstoties uz laiku, kādā tiek analizētas tīkla stāvokļu izmaiņas un tiek pieņemti lēmumi par izmaiņām tīklā, trafika vadības lēmumus nosacīti varētu sadalīt 3 kategorijās.

1. Tiešsaistes lēmumi – sākot no sekundes daļām un beidzot ar padsmītos sekunžu – tiek pieņemti, lai nodrošinātu tiešsaistes trafika vadības uzdevumus efektīvai tīkla resursu pārdalei starp lietotājiem – CAC kontrole, maršrutēšanas ceļu izvēle plūsmām utt.
2. Fona lēmumi – minūtes un minušu desmiti – tiek pieņemti, balstoties uz tīkla darbības fonā saņemtajiem datiem par tīkla uzvedību un trafika dabu – vidējā termiņa lēmumi par rezervēšanas operācijām linkos, maršrutēšanas ceļu preferencēm nākamajiem savienojumiem utt.

3. Ilgtermiņa lēmumi – stundas, dienas, mēneši – lielas nozīmes tīkla infrastruktūras izmaiņas gan topoloģiskajā, gan arī loģiskajā plaknē – apakštīklu uzbūve, jaunie fiziskie un loģiskie savienojumi vai rezervētie resursi ar lielu kapacitāti utt.

Tagad atgriezīsimies pie NGN tīklu attīstības iespējamā scenārija, kur par pamatu tiek izvirzīta koncepcija par katras komunikācijas sesijas individuālo apkalpošanu. Šāda veida plūsmu apkalpošana būs, sākot no pieslēguma izveides, kas balstās uz striktu pieslēgumam nepieciešamo parametru izpildi, un beidzot ar biržai līdzīgo pakalpojumu iegādi uz nepieciešamo sesijas laiku vai pieprasīto resursu daudzumu. Galveno lomu šajā jautājumā spēlēs tieši tiešsaistes lēmumi, nodrošinot efektīvu tīkla resursu pārdali starp lietotājiem. GMPLS tīklu gadījumā, kuriem par alternatīvu mūsdienu sakaru sistēmās varam izvēlēties MPLS-TE tīklus, runa ietu par CAC kontroli LSP iestatīšanas laikā.

Ņemot vērā visu iepriekš minēto, šodien īpaši aktuāls kļūst jautājums par LSP plūsmu vadību MPLS-TE tīklu gadījumā. Mūsdienu MPLS tīklos LSP tiek iestatīti pārsvarā manuāli, rezervējot resursus ar RSVP-TE protokola palīdzību. Trafika maršrutēšana notiek pa iepriekš iestatītiem LSP, balstoties uz iepriekš definētām QoS klasēm. Kad konekcijas laikā resursi linkā tiek rezervēti ar RSVP-TE protokolu, CAC vadība notiek balstoties uz klasisko sliekšņa-CAC algoritmu, kas dod pieslēgumu, ja linkā brīvo resursu skaits pārsniedz pieprasīto resursu apjomu, un atsaka pieslēgumu, ja pieprasīto resursu apjoms ir mazāks par linkā esošajiem brīvajiem resursiem. Šāda algoritma darbību varam attēlot ar šādu lēmuma pieņemšanas kārtulu (1):

$$CAC_c(Bw_{rq}) = \begin{cases} Atļaut, & ja Bw_{rq} \leq Bw_{aw} \\ Atteikt, & ja Bw_{rq} > Bw_{aw} \end{cases} \quad (1)$$

kur

- $CAC_c(Bw_{rq})$  – ir klasiskā sliekšņa CAC atbilde,
- $Bw_{rq}$  – ir pieprasītās caurlaides joslas platums [Mb/s],
- $Bw_{aw}$  – ir pieejamās joslas platums linkā [Mb/s].

Šī pieeja principā izslēdz citu QoS raksturojošo parametru analīzi lēmuma pieņemšanas laikā un balstās vienīgi un pieejamās caurlaides joslas platumu. Sakaru sesijas QoS monitorings maršrutētājā pēc dažām minūtēm iespējams likvidēs iestatīto LSP tuneli, ja sesijas QoS prasības netiks izpildītas. Tas savukārt novedīs pie atkārtotiem konekcijas

mēģinājumiem un, iespējams, atkārtotiem atteikumiem, kā arī nekvalitatīviem sakariem LSP tuneļa dzīves laikā. Tam par iemeslu ir vēl viena elementa trūkums mūsdienu trafika vadības paradigmā: praktiski **netiek veikta proaktīva resursu vadības politika**. Visas darbības tīkla trafika vadības plaknē vadās pēc reakcijas principa – tiek detektēti neatbilstoši QoS parametri, tādi kā pakešu zudumi, aizture un trīce, un tad tiek pieņemti lēmumi par tālākām rīcībām. Tas viss noved pie nepieciešamības atgriezties pie jautājuma par trafika vadības paradigmu, kas savā būtībā, šodien ir **balstīta uz stacionārā trafika vīziju un reaktīvo darbības veidu**.

Šādā situācijā faziloģika būs daudzsološs pretendents efektīvajam lēmumu pieņemšanas nodrošinājumam, postulējot nenoteiktības apstākļus kā neatņemamu sistēmas sastāvdaļu un nodrošinot robustu lēmējmehānisma darbību. Šāda tīklu attīstības vīzija var balstīties tikai uz ātriem un efektīviem maršrutēšanas un CAC risinājumiem, kas ir spējīgi darboties ātri mainīgajā vidē un pieņemt lēmumus nenoteiktības apstākļos.

Tagad sīkāk pievērsīsimies jautājumam par RSVP-TE protokola pielietojuma aktualitāti MPLS-TE sakaru tīklos. Kā jau tika minēts iepriekš, LSP iestatīšana pētāmajā MPLS-TE tīklā var notikt ar RSVP-TE protokola palīdzību. RSVP-TE ir detalizēti aprakstīts IETF RFC3209, un papildināts RFC5151. Pieņemot CAC lēmumu, tas ļauj iestatīt LSP tuneli MPLS-TE tīklā, ņemot vērā tādu ierobežojošos parametrus kā pieejamā caurlaides josla un precīzs lēkumu skaits. 2003. gada februārī, IETF MPLS darba grupa nolēma atteikties no alternatīvā CR-LDP (RFC3212) signalizācijas protokola attīstības un koncentrējās vienīgi uz RSVP-TE protokola attīstību (RFC3473), jo 2002. gada jūnijā publicētais pārskats par pirmajām GMPLS realizācijām parādīja, ka no 22 risinājumiem 21 pielietoja RSVP-TE protokolu, un tikai 1 lietoja CR-LDP [147]. Tātad RSVP-TE protokola attīstība un modifikācijas ir attiecināmas arī uz GMPLS tīkliem, kas ir definēti kā NGN tīkla pamata tehnoloģija.

Mūsdienu pētījumos CAC vadībai ir veltīta liela uzmanība, kas skar telekomunikāciju jomu [60], [61], [58], [73], [146], [100], [5], [143]. Arī MPLS tīklu vadības CAC kontrole ir apskatīta dažos nesenos pētījumos [39], [4]. Tomēr faziloģikas pielietojums CAC kontrolei MPLS tīklos, pēc šīs disertācijas autora ziņām, ir piedāvāts tikai vienā pētījumā [87], kas tiks sīkāk izanalizēts nodaļā 1.6.4.

Visas iepriekš aprakstītās problēmas motivēja šī promocijas darba autoru pievērsties mūsdienu trafika vadības problēmai, kā rezultātā tapa šī disertācija. Tā ir veltīta faziloģikas pielietojumam CAC uzdevuma risinājumam NGN tīklos ar MPLS-TE/GMPLS vadības plakni, pielietojot RSVP-TE protokolu.

## **Darba mērķis un uzdevumi.**

Izejot no visa iepriekš teiktā, šī promocijas darba **mērķis bija izveidot jaunu fazi-CAC projektēšanas metodi, kas novērš minētos trūkumus, kā arī novērtēt konstruētā fazi-CAC pielietošanas iespējas MPLS-TE (GMPLS) trafika pārraides/vadības sistēmā ar RSVP-TE protokolu, nodrošinot dinamisko LSP iestatīšanu.**

**Lai sasniegtu minēto mērķi bija nepieciešams atrisināt šādus uzdevumus:**

1. Analizējot literatūru, atrast efektīvus risinājumus sistēmās ar nenoteiktību un proaktīvu vadību un piemērot tos konkrētajam uzdevumam.
2. Noteikt mūsdienu trafika vadības paradigmas ierobežojumus.
3. Definēt MPLS-TE un Fazi-CAC modeli eksperimentāliem pētījumiem, kas būtu par pamatu simulācijas shēmas un eksperimentālā tīkla realizācijai, kā arī noteikt MPLS-TE QoS ierobežojošos faktorus un to robežas, pamatojoties uz esošo SLA (angl. Service Level Agreement) datiem un izvēlētā modeļa īpatnībām.
4. Pielietojot modeļi ietvertos ierobežojumus, ar heuristisko pieeju konstruēt fazi izvedumu sistēmu, kas nodrošina selektīvu LSP iestatīšanu un QoS parametru saglabāšanu definēto robežu ietvaros.
5. Izveidot eksperimentālu MPLS-TE sakaru sistēmas tīklu un simulācijas vidi, kā arī veikt klasiskā sliekšņa-CAC un fazi-CAC algoritmu praktiskās realizācijas identiskos apstākļos ar mērķi veikt turpmāko rezultātu salīdzinošu analīzi.
6. Izanalizēt iespējamās fazi-CAC adaptācijas mehānismus, tādus kā ieejas parametru lingvinistisko mainīgo deklarāciju izmaiņas un fazi-CAC atbildes pēcapstrāde.

## **Aizstāvamās tēzes.**

1. Uz faziloģiku balstīts CAC algoritms, salīdzinot ar klasisko sliekšņa-CAC algoritmu, vadot RSVP-TE resursu rezervēšanu MPLS-TE tīklā (arī GMPLS nākotnē), spēj nodrošināt maksimālo vidējo linka utilizāciju pie uzdotiem QoS parametru degradācijas ierobežojumiem.
2. Salīdzinot ar klasisko sliekšņa-CAC algoritmu, uz faziloģiku balstīts CAC risinājums spēj nodrošināt selektīvu trafika plūsmu izvēli LSP tuneļu izveidei MPLS-TE tīklos (arī GMPLS nākotnē) un palielināt augstu QoS klašu plūsmu pieslēgumu skaitu, vienlaicīgi nodrošinot plūsmu apkalpošanai attiecīgus QoS parametrus.

3. Pielietojot MPLS-TE (arī GMPLS nākotnē) tīklos fazi-CAC algoritmu klasiskā sliekšņa-CAC algoritma vietā, ir iespējams samazināt tādu plūsmu pieslēgumus, kas neatbilst linka QoS nodrošinājuma iespējām LSP iestatīšanas brīdī, tādā veidā mazinot pieslēgto datu plūsmu priekšlaicīgus atslēgumus QoS parametru neatbilstības dēļ.
4. Salīdzinot ar klasisko sliekšņa-CAC algoritmu, uz fazilōģiku balstīts CAC risinājums spēj dod iespēju tiešsaistes režīmā nodrošināt ātru un efektīvu trafika vadības politikas maiņu MPLS-TE maršrutētājos (arī GMPLS nākotnē), adaptējoties pie mainīgajiem tīkla apstākļiem.

## DISERTĀCIJAS ZINĀTNISKAIS JAUNINĀJUMS

Darba zinātniskais jauninājums ir MPLS-TE sakaru sistēmas RSVP-TE protokola CAC vadības projektēšanas metode, kura balstās uz faziloģikas bāzes.

Autora izstrādātais fazi-CAC risinājums dod iespēju iestatītām LSP tuneļu plūsmām nodrošināt šādus vēlamos QoS parametrus: pakešu aizture, pakešu aizturu trīce un pakešu zudumi. Pēc promocijas darba autora ziņām, tas ir pirmais uz faziloģiku balstīts CAC algoritms un risinājums LSP iestatīšanai MPLS-TE tīklos, kurā tiek pielietotas šādas veida fazi-CAC ieejas un izejas parametru fazikopas un zināšanu bāze IF-THEN kārtulu formā. Šī pētījumu rezultātā, pēc autora ziņām, pirmo reizi tika radīta veikspējīga un izvirzītām prasībām atbilstoša fazi-CAC praktiskā realizācija MPLS-TE sakaru sistēmas eksperimentālā tīklā. Šis promocijas darbs ir pirmais pētījums par fazi-CAC pielietojumu efektivitāti MPLS-TE sakaru sistēmā, un tas ietver ne tikai teorētiskus un simulācijas pētījumus, bet arī eksperimentāli pārbauda fazi-CAC algoritma efektivitāti pakešu līmenī. Šim nolūkam tika izveidots MPLS-TE sakaru sistēmas eksperimentāls tīkls.

### Jauniegvums

- Ir izstrādāta jauna fazi-CAC vadības projektēšanas metode, nodrošinot MPLS-TE (GMPLS) tīklos RSVP-TE protokola vadību ar fazi-CAC.
- Ir atrasta optimāla (pēc utilizācijas kritērija un ierobežojumiem) fazi-CAC FIS struktūra, kas ietver sevī
  - fazi-CAC ieejas un izejas parametru lingvinistiskos mainīgos;
  - lingvinistisko mainīgo piederības funkciju tipus un parametrus;
  - IF-THEN kārtulu zināšanu bāzi;
  - ieejas parametru fazifikācijas līkņu tipus un parametrus;
  - izejas parametra defazifikācijas metodes.

- Balstoties uz šiem pētījumiem ir izstrādātas rekomendācijas katra FIS struktūrelementa pielietojuma apgabaliem.
- Ir piedāvāts oriģināls risinājums fazi-CAC atbildes stāvokļu kopai, kas satur 3 dažādas atbildes:
  - atteikt LSP iestatīšanu;
  - atļaut LSP iestatīšanu;
  - atļaut LSP iestatīšanu ar papildu resursu rezervēšanu.

Dotais risinājums ļauj izveidot t.s. „drošības spilvenu” augstās prioritātes LSP plūsmām linkā, kas ir tieši proporcionāls augstās prioritātes LSP plūsmām linkā.

- Ir atrastas likumsakarības starp RSVP-TE rezervētiem resursiem linkā, fazi-CAC atteikto plūsmu pieslēgumu skaitu un piedāvātā fazi-CAC risinājuma FIS atbildes pēcapstrādes sliekšņa vērtību. Šie dati ļauj veikt momentānās fazi-CAC politikas korekcijas atkarībā no apkārtējās vides izmaiņām.
- Izstrādātā projektēšanas metode ļauj realizēt adaptīvus algoritmus attiecībā pret ārējās vides izmaiņām.
- Ar fazi-CAC, selektīvi iestatot LSP tuneļus, linka utilizācija, salīdzinot ar sliekšņa-CAC, pazeminās, vienlaicīgi saglabājot QoS parametrus definētās robežās.

### **Darba praktiskā vērtība**

- Ir izveidots fazi-CAC FIS modelis, kā arī ir veikta izveidotā modeļa verifikācija un validācija.
- Ir izveidota praktiskā fazi-CAC realizācija RSVP-TE protokola vadībai MPLS-TE tīklā, kas ir eksperimentāls produkts.

- Eksperimentālais MPLS-TE sakaru sistēmas tīkls ar RSVP-TE protokola faziloģikas balstītu CAC kontroli, dod tehnoloģisku pamatu plaša loka pētījumiem par fazi-CAC pielietojumu, saglabājot vēlamos QoS parametrus.
- Uz eksperimentālā MPLS-TE sakaru tīkla bāzes, pielietojot vairāku maršrutētāju fazi-CAC vadību ar attiecīgu starpprogrammatūru (piem. JADE), var realizēt vairāku fazi-CAC aģentu sadarbību multiaģentu slēgumā.
- Eksperimentālais tīkls un tam piesaistītais programmatiskais nodrošinājums tiek pielietots bakalaura un maģistra darbu izstrādei.
- Ir izveidota metodika MPLS-TE maršrutētāja darbības korekcijai, nodrošinot RSVP-TE protokola vadību ar fazi-CAC fazi izvedumu sistēmu (FIS).

### **Darba rezultātu aprobācija:**

Fazi-CAC darbības efektivitāte ir pārbaudīta attiecībā pret RSVP-TE pielietoto klasisko sliekšņa-CAC algoritmu gan plūsmu gan pakešu līmenī, pierādot tā spēju nodrošināt selektīvu LSP iestatīšanas politiku un proaktīvu trafika vadības paradigmu.

### **Promocijas darba galvenie rezultāti tika prezentēti šādās konferencēs:**

1. Electronics and Electrical Engineering, Lithuania, Kaunas, 21. – 23. maijs, 2007
2. RTU 48. starptautiskā zinātniskā konference 12. oktobris, 2007.
3. Electronics and Electrical Engineering, Lithuania, Kaunas, 20. – 22. maijs, 2008.
4. Electronics and Electrical Engineering, Lithuania, Kaunas, 13. – 15. maijs, 2009.
5. IEEE Region 8 International Conference on Computational Technologies in Electrical and Electronics Engineering, Russia, Irkutsk, Listvjanka, 11. – 15. jūlijs, 2010.
6. Electronics and Electrical Engineering, Lithuania, Kaunas, 18. – 20. maijs, 2010.
7. 18th Telecommunications Forum TELFOR 2010, Serbia, Belgrade, 23. – 25. novembris, 2010.

8. Electronics and Electrical Engineering, Lithuania, Kaunas, 17. – 19. maijs, 2011.
9. 11th International Conference on Telecommunications, ConTEL 2011, Graz, Austria, 15. – 17. jūnijs, 2011.
10. 12th IEEE International Conference on High Performance Switching and Routing, Spain, Cartagena, 4. – 7. jūlijs, 2011.

**Kopumā ir publicēti 14 zinātniski raksti dažādos zinātniskos izdevumos**

1. Jeļinskis J., Lauks G. Optimal Policy for LSP Control in MPLS Networks // Electronics and Electrical Engineering, LIETUVA, Kaunas, 21.-23. maijs, 2007. 65–68. lpp
2. Jeļinskis J. Heuristic algorithm for optimal LSP set up policy in MPLS Networks // RTU zinātniskie raksti. 7. sēr., Telekomunikācijas un elektronika. - 7. sēj. (2007), 23.–26. lpp.
3. Jeļinskis J., Lauks G. Approximation of Internet Traffic Using Robust Wavelet Neural Networks // Electronics and Electrical Engineering 6(86), Lithuania, Kaunas, 20.-22. maijs, 2008. 81–84. lpp.
4. Jeļinskis J., Lauks G. Data Mining for Managing Intrinsic Quality of Service in MPLS // Electronics and Electrical Engineering 5(85), Lithuania, Kaunas, 20.-22. maijs, 2008. 33–36. lpp.
5. Jeļinskis J. Heuristic algorithm for Robust Approximate Hurst Parameter Estimation with Wavelet Analysis and Neural Networks // Scientific Journal of RTU. 7. series., Telekomunikācijas un elektronika. - 8. vol. (2008), 12–15. lpp.
6. Bobrovs V., Jeļinskis J., Ivanovs Ģ., Lauks G. Research of Traffic Management in FTTx Optical Communication Systems // Latvian Journal of Physics and Technical Sciences . - 2. (2009) 41. –55. lpp.
7. Lauks G., Jeļinskis J. Metamodelling of Queuing Systems Using Fuzzy Graphs // Electronics and Electrical Engineering Nr. 4(92) ISSN 1392-1215, Electronics and Electrical Engineering, Lithuania, Kauņa, 13.–15. maijs, 2009. 61–64. lpp.
8. Jeļinskis J., Lauks G. Detection of Trends of Internet Traffic Using Sequential Patterns // Electronics and Electrical Engineering Nr. 4(93) ISSN 1392-1215, Electronics and Electrical Engineering, Lithuania, Kauņa, 13.–15. maijs, 2009. 3–6. lpp.
9. Jeļinskis J., Lauks G. Fuzzy Approach for QoS Aware Application Driven Traffic Control in GMPLS Networks // IEEE Region 8 International Conference on Computational Technologies in Electrical and Electronics Engineering, Volume I, Krievija, Irkutska, Listvjanka, 11.–15. jūlijs, 2010. 199.–203. lpp.

10. Jeļinskis J., Rutka G., Lauks G. Fuzzy-CAC for LSP Setup in GMPLS Networks // Electronics and Electrical Engineering 2010, Nr.5(101), Lithuania, Kauņa, 18.-20. maijs, 2010. 31. –34. lpp.
11. Jeļinskis J., Lauks G. Fuzzy CAC Adaptation for Effective Traffic Control in GMPLS Networks // 18th Telecommunications Forum TELFOR 2010, Serbija, Belgrade, 23.-25. novembris, 2010. - 10.-13. lpp.
12. J. Jelinskis, A. Skrastins, G. Lauks. Practical Fuzzy-CAC Realization for Effective Traffic Engineering in MPLS-TE Network // Electronics and Electrical Engineering. – Kaunas: Technologija, 2011. – No. 4(110), 30.–34. lpp.
13. J. Jelinskis, A. Skrastins, G. Lauks. Fuzzy-CAC based Traffic Management in MPLS-TE Networks //11th International Conference on Telecommunications, ConTEL 2011, Graz, Austria, 15.–17. jūnijs, 2011, 389.–395. lpp.
14. J. Jelinskis, A. Skrastins, G. Lauks. Fuzzy-CAC Driven MPLS-TE Realization // 12th IEEE International Conference on High Performance Switching and Routing, Spain, Cartagena, 4. – 7. jūlijs, 2011, 146 – 150. lpp.

Promocijas darba izstrādes laikā promocijas darba rezultāti tika izmantoti vairāku projektu pētījumos

1. Trafika apvienošanas izpēte „IP virs WDM” tīklos. Projekta Nr.: ZP2008/16 (2008/2009)
2. Valsts pētījumu programmas „Informācijas Tehnoloģijas projekts” nr. 5 „Jaunas elektronisko sakaru tehnoloģijas” novirziens – elektrosakaru sistēmu drošums un drošība. RTU projekts Nr V7408.1. (2007, 2008, 2009)
3. Trafika vadības izpēte „dzīsla līdz mājai” optiskajā sakaru sistēmā. Projekta Nr.: ZP2007/13 (2007/2008)
4. Nr. 05.1649 „Optimālās maršrutizācijas algoritmi uzklātos tīklos” (2007, 2008)

## PĒTĪJUMU SFĒRA UN IEROBEŽOJUMI

Pētījuma metodoloģija rada ierobežojumus pētījumu rezultātiem. Pirmkārt, pētījuma rezultāti, kas tika iegūti pētījuma laikā, balstās uz pētāmās MPLS-TE sakaru sistēmas izvēlēto atsauču modeli un autora piedāvāto fazi-CAC fazi izvedumu sistēmas (FIS) atsauču modeli. FIS tiks atrasta heuristiskās atlasē pētījumu metodes gaitā, izvēloties fazi-CAC FIS parametru vērtības. Otrkārt, pētījuma rezultātā iegūtie dati tiks piesaistīti specifiskajiem ieejas parametru vērtību vektoriem, kas pēc būtības ir speciāls gadījums sintētisko datu ievadam.

Kā nākamo ierobežojumu var pieminēt MPLS-TE eksperimentālā tīklā pielietotās reālās datu plūsmas, kas izmanto tīklu, vadot LSP pieslēgumus ar sliekšņa-CAC un fazi-CAC algoritmiem. Datu plūsmu specifiskās īpašības, piemēram, uzliesmojošā daba, rezultātā dos tām raksturīgu pakešu apkalpošanas laika atkarību no pētāmā tīkla izmantošanas. Rezultātā tiks radīta šī konkrētā tīkla un datu plūsmu apvienojumam specifiska fazi izvedumu sistēma.

Tomēr, iepriekš pieminētie ierobežojumi nenovērš iespēju formēt vispārinātus secinājumus, jo pielietotās likumsakarības ir konsekventas plašā trafika izpētes jautājumu lokā. Piemēram, atļaujot datu plūsmas konekciju ar augstām QoS prasībām, kad tās nevar tikt nodrošinātas, klienta SLA noteikumi netiek izpildīti un kopējā tīkla veiktspēja degradē, jo šāda plūsma ir jāslēdz. Tas dos rezultātā pārrautu sakaru sesiju un atkārtotus konekcijas pieprasījumus. Šādi pieņēmumi ir spēkā plašajā tīklu tehnoloģiju un tehnoloģisku risinājumu lokā. Kaut gan šī pētījuma rezultāti nevar garantēt pilnīgi universālus rezultātus, tie dod atbildes, kas var tikt vispārinātas attiecībā uz daudziem CAC problēmu risinājumu veidiem MPLS-TE, kā arī GMPLS tīklos nākotnē.

## PIELIETOTĀS PĒTĪJUMU METODEDES

Šis pētījums ir sadalīts vairākās fāzēs. Pirmā fāze ietver sevī literatūras studijas, kas dod pārskatu par esošo trafika vadības paradigmu telekomunikācijās, kā arī faziloģikas pielietojuma analīzi dažādās telekomunikāciju jomās un esošo fazi-CAC algoritmu apskatu.

Otrā fāze ietver MPLS-TE sakaru sistēmas eksperimentālā maketa atsauču modeļa izveidi un fazi-CAC algoritma fazi izvedumu sistēmas atsauču modeļa izveidi. Šajā pētījuma fāzē autors arī definē piedāvātā fazi-CAC algoritma darbības efektivitātes kritērijus.

Ar nolūku atrast izvirzītiem darbības efektivitātes kritērijiem optimālo vai tam tuvu fazi-CAC algoritma lēmējsistēmas uzbūvi, autors pielieto heuristisko atlasē, mēģinājumu un kļūdu pētniecisko metodi, rezultātā izvēloties nepieciešamos parametrus, tādus kā:

- ieejas parametru lingvinistisko mainīgo piederības funkciju deklarācijas;
- ieejas parametru fazifikācijas likumsakarības;
- fazi izvedumu sistēmas izvēle;
- izejas parametra defazifikācijas metodes izvēle;
- IF-THEN kārtulu zināšanu bāzes deklarācija;
- fazi-CAC pēcapstrādes veida un parametru izvēle.

Šim nolūkam autors pielieto programmatisko MPLS-TE sakaru sistēmas tīkla modeļa realizāciju MATLAB vidē, ka arī fazi-CAC algoritma realizāciju, balstītu uz *Fuzzy Logic Toolbox*<sup>TM</sup> bibliotēkas, kas arī ir *MATLAB*<sup>®</sup> tehniskās skaitļošanas vides rīka sastāvdaļa.

Nākamajā pētījumu fāzē promocijas darba autors izveido RTU ETF Telekomunikāciju institūta, Maršrutēšanas laboratorijā, MPLS-TE sakaru sistēmas eksperimentālo tīklu, ka arī izveidoto programmatisko fazi-CAC vadības rīku. Fazi-CAC vadības programmatiskais rīks ietver sevī fazi-CAC vadības bloku, visas nepieciešamās saskarnes maršrutētāja vadībai, trešās puses utilitprogrammu vadības sistēmas un trafika analīzei nepieciešamos programmatūras rīkus. Uz izveidotā eksperimentālā MPLS-TE sakaru sistēmas tīkla tiek veikta promocijas darba autora piedāvātā fazi-CAC algoritma darbības salīdzinoša analīze ar klasisko sliekšņa-CAC algoritmu, kas ir standarta RSVP-TE protokola darbības pamatā. Šajā fāzē autora piedāvātais algoritms tiek salīdzināts ar sliekšņa CAC algoritmu, balstoties uz iepriekšnoteiktajiem eksperimentu scenārijiem.

Pēdējā pētījuma fāzē, ar nolūku noskaidrot autora piedāvātā fazi-CAC algoritma iespējamās adaptācijas spējas mainīgajiem apstākļiem, algoritma parametru vērtības tiks modificētas, mainot fazi-CAC fazi izvedumu sistēmas uzbūvi un analizējot fazi-CAC darba efektivitāti, veicot pēcapstrādes lēmumu modifikācijas.

## SVARĪGĀKĀS DEFINĪCIJAS UN AUTORA PIENĒMUMI

Šajā nodaļā dotās disertācijas autors strikti definē (pēc autora domām) svarīgākos jēdzienus un pieņēmumus, kas ir izmantoti šajā pētījumā. Tas ir vajadzīgs, lai nevarētu rasties domstarpības un dažādas interpretācijas, ejot cauri autora sniegtajiem pētījuma metodikas aprakstiem un rezultātu novērtējumiem, kā arī lai lasītājam būtu pilnīgi skaidra dotā promocijas darba autora motivācija, lietojot konkrētus jēdzienus vai arī izvēloties kādas noteiktas pieejas problēmas risinājumiem.

### **CAC vadība**

CAC vadības definīcija [ITU I371,1995] ir šāda: „CAC ir darbību kopa, ko veic tīkls savienojuma izveides fāzē (vai savienojuma laikā), lai nodrošinātu virtuāla kanāla savienojuma vai virtuāla ceļa savienojuma atteikumu vai akceptēšanu”

### **Nenoteiktība**

Šajā promocijas darbā autors lieto jēdzienu „nenoteiktība”, kas tiek pārsvarā izmantots datu pārraides tīklu un interneta trafika raksturojumos. Tā kā vārds „nenoteiktība” var tikt pielietots ļoti plašā definīciju spektrā, gribētos strikti ierobežot šī jēdziena interpretāciju šī pētījuma ietvaros.

Autors ir izvērtējis daudzas šī vārda definīcijas un kā vislabāk atbilstošu šī pētījuma mērķiem ir izvēlējis definīciju no mūsdienu ekonomiskās vārdnīcas B. Rajzberga, L. Lozovska un E. Starodubceva redakcijā [148]. Šī definīcija autora tulkojumā skan šādi:

Nenoteiktība – zināšanu trūkums par apstākļiem, kuros notiks ekonomiskā darbība, zema notikumu prognozēšanas pakāpe. Nenoteiktība ir saistīta ar plānošanas un lēmuma pieņemšanas risku, kā arī ar darbību riskiem visos ekonomiskās sistēmas līmeņos.

Attiecinot šo jēdzienu definīciju uz mūsdienu sakaru sistēmām, kas ir šī pētījuma sfēras ietvaros, varam sasaistīt vārdus „ekonomika” un „sakaru tīkls” vai arī „ekonomika” un „trafiku raksturojoši parametri”.

Te gribētos vēl uzsvērt, ka ar vārdu nenoteiktība nekādā mērā netiek domātas kļūdas! Autors vārdu „nenoteiktība” drīzāk lieto, lai uzsvērtu grūti prognozējamo vērtību izkliedi, kuru varētu piedēvēt mērāmajam, kas varētu būt gan trafiku, gan tīkla elementu raksturojošs parametrs vai parametru kopa.

## Fazi-CAC algoritma ieejas testa datu kopas vērtību ģenerators izvēle

Ir svarīgi pamatot ieejas datu kopas vērtību ģenerators izvēli, kas tiek pielietots, veidojot fazi izvedumu sistēmas ieejas parametru vērtību trases failus. Šie trases faili tiek pielietoti autora piedāvātā fazi-CAC un klasiskā sliekšņa CAC algoritma testēšanai, gan veicot simulācijas, gan pārbaudot fazi-CAC darbību MPLS-TE sakaru sistēmas eksperimentālā tīklā.

Autoram bija vajadzīgs trafika pieprasījuma modelis, kas vienlaicīgi reprezentētu mūsdienu trafikam piemītošu ilgatkarību un nestacionaritāti. Ir jāpiemin dažas svarīgākās īpatnības, kādām būtu jāpiemīt trafika pieprasījuma modelim, kurš apmierinātu šī pētījuma prasības. Ir svarīgi, lai gadījuma procesa modelis būtu ar nestacionāro raksturu, kā arī lai šāda modeļa viens no formāliem parametriem tieši raksturotu Hersta parametru [28], kas ir LRD mērs. Tā kā mūsdienu trafiku raksturo arī LRD [80], [59], [20], [11], [45], [64], [113], [96], to vajadzētu ietvert autora izvēlētajā modelī.

Šādiem nosacījumiem lieliski atbilst frakcionētas Brauna kustības modelis jeb saīsināti fBm [angl. Fractional Brownian motion]. Tas ir nestacionārs, un viens no tā parametriem ir H jeb Hersta parametrs, kas ļauj uzdot ilgatkarības pakāpi modelētajai datu laika rindai [86]. Arī daudzas literatūrā atrastās atziņas par fBm procesa lielisko atbilstību mūsdienu trafika modelēšanai [145], [93], [101], [104] veicināja autora lēmumu par labu šī modeļa pielietojumam, modelējot ieejas datus autora piedāvātā fazi-CAC algoritma testēšanai. Tika izvēlēts fBm ilgatkarīgo procesu ģenerators [8], kura realizācija bija pieejama kā *fBm* definēta funkcija Matlab skaitļošanas vidē.

# 1. FAZILOĢIKA MŪSDIENU TRAFIKA VADĪŠANĀ

## Nodaļas kopsavilkums

Šinī nodaļā promocijas darba autors sniedz īsu ieskatu mūsdienu sakaru sistēmu vadības paradigmā, uzsvērt tās vājās un stiprās puses un norādot faziloģikas pielietojuma iespējamus plusus un mīnus. Tālāk tiek sniegts ieskats faziloģikas pamatkonceptijā un dots faziloģikas kā zinātnes attīstības īss apraksts, kā arī ir norādītas faziloģikas pielietojuma svarīgākās sfēras un spilgtākie piemēri. Tiek dots arī faziloģikas lēmējsistēmas pamatelementu detalizēts apraksts:

- lingvinistisko mainīgo piederības funkcijas;
- pamata faziloģiskās operācijas;
- ievada parametru fazifikācijas paņēmieni;
- izvada parametru defazifikācijas metodes.

Nodaļas nobeigumā promocijas darba autors dod īsu literatūras apskatu par faziloģikas pielietojumu telekomunikāciju dažādās sfērās, īpašu uzmanību veltot esošajiem faziloģikas pielietojumu piemēriem CAC uzdevumu risinājumos.

### 1.1. Mūsdienu sakaru sistēmu vadības problēmas

Telekomunikāciju sistēmas savā būtībā ir tiešsaistes sakaru tīkli, kas nodrošina drošu datu transportu no punkta A līdz punktam B. Tomēr ir jāpatur prātā viens svarīgs un nozīmīgs fakts – visas sakaru sesijas notiek nepārtraukti mainīgajos apstākļos. Apskatot kaut vai fizisko slāni, kur saskaramies, piemēram, ar elektromagnētiskiem traucējumiem, vai tīkla slāni ar nenoteiktu maršrutēšanas protokolu darbību, vai arī transportēšanas slāni ar plūsmu vadību un TCP protokola loga lieluma nemitīgām fluktuācijām, un beidzot lietojumu slāni ar nenoteiktu lietotāju uzvedību un lietojumu QoS prasībām – itin visur pastāv nenoteiktība, kas nedod iespēju pieņemt lēmumus strikti determinētos apstākļos. Šādā situācijā faziloģika ir lielisks pretendents efektīvajam lēmumu pieņemšanas nodrošinājumam, pieņemot nenoteiktības apstākļus kā neatņemamu sistēmas sastāvdaļu un nodrošinot robustu lēmēj mehānisma darbību.

Faktiski mūsdienu trafika vadības paradigma savā pamatā balstās uz pieņēmumu par trafika stacionāro dabu. Tas dod pamatu daudzām trafika vadības pieejām, kuras pielieto trafika prognozes datus. Būtu jāsaprot, ka trafiku nevis uzskata par stacionāru, bet gan tam mākslīgi uzdod šādas īpašības, fokusējoties uz sezonālītātēm, tendencēm un abstrahējoties no nevēlamajiem uzliesmojumiem. Rezultātā trafika raksturs tiek „nogludināts” un daudzas problēmas tiek ignorētas – iespējams prognozēt tīkla uzvedību, jebkura veida neprognozētie notikumi tīklā tiek deklarēti kā neraksturīgi un ar tiem tiek galā pēc principa „ja kaut kas notiks – tad arī jārikojas!”. Šajā pieejā būtībā nav proaktīvās rīcības, jo ar pašu pieņēmumu par trafika stacionāro dabu mēs abstrahējamies no tā īstās uzvedības, it īpaši ātro tiešsaistes lēmumu gadījumā, tādu kā CAC kontrole. Ilgtermiņa prognozēm, bez šaubām, ir svarīga loma tendenču veidošanā, kas dod pamatu ilgtermiņa izmaiņām tīklu arhitektūrā un to vadības vadlīnijās. Savukārt, runājot par īstermiņa prognozēšanu, iespējams, būtu vērtīgi apskatīt arī citas trafika vadīšanas iespējas, kas neizsvītro no trafika raksturojošajiem parametriem tā nestacionāro dabu.

Turpinot pieminēšu arī ļoti respektējamo globālo trafika prognozi, ko sniedz Cisco, kā daļu no t.s. VNI jeb Cisco Visual Networking Index. Šī globālā IP trafika prognoze dod ļoti detalizēto 5 gadu trafika attīstības vīziju tīmeklī, kas ir sadalīta pa trafika ģenerējošajām aplikācijām, ģeogrāfisko trafika avotu novietojumu un trafika transportēšanas īpatnībām, tādām kā mobilo un fiksēto tīklu trafiks. Tiek sniegts arī prognozēšanas metodoloģijas pamatojums, ar ko var iepazīties sīkāk pēc dotās norādes [44].

1.1. tabula

Globāls lietotāju interneta trafiks 2009–2014

<b>Lietotāju interneta trafiks, 2009-2014</b>							
	2009	2010	2011	2012	2013	2014	CAGR 2009-2014
Katrā no apakšsegmentiem (PB/mēnesī)							
Failu apmaiņa	4,091	5,075	6,197	7,492	9,125	11,340	23%
Interneta video	2,776	4,725	7,718	11,026	14,838	19,468	48%
Interneta video uz TV	107	263	711	1,502	2,686	4,075	107%
Web/Dati	1,688	2,273	3,006	3,930	4,933	6,134	29%

Video zvani	83	128	199	284	407	599	48%
Tiešsaistes spēles	63	86	120	167	226	307	37%
VoIP	122	134	141	144	145	146	4%

Dažas nodaļas no 2010. gada 2. jūlija prognozes īpaši saistīja dotā darba autora uzmanību. Kā var redzēt no tabulas 1.1. tad pēc Cisco VNI prognozes video trafika īpatsvars interneta tīmeklī kļūs dominējošs. Tas jau tuvāko divu gadu laikā pārspēs mūsdienu līderi – failu apmaiņu, kas pārsvarā balstās uz P2P trafiku. Interneta video, gan klasiskajā nozīmē, gan arī straumēšanai tieši uz TV ierīcēm, kā arī video zvani, iespējams, dos vislielāko pieaugumu pasaules ģenerētajam trafikam. Tas savukārt nozīmē, ka kardināli tiks izmainīts gan trafika raksturs, gan tā apjomi tīmeklī. Video trafika uzliesmojošais raksturs jo vairāk pastiprina kopējā tīkla trafika nestacionāro dabu un paceļ jaunā virsotnē jautājumu par lēmumu pieņemšanu trafika vadības uzdevumiem. Kas tad notiks ar tehnoloģisko pusi un algoritmiem, kuri tiek pamatā lietoti trafika vadīšanai tīklā?

Tabulās 1.2, 1.3 un 1.4 ir attēlotas trafika pieauguma prognozes tieši video trafika segmentā. Tabulā 1.2 ir doti dati par prognozējamo video trafika pieaugumu, kas tiktu skatīts uz parastajiem personālajiem datoriem. Tabulā 1.3 varam redzēt video trafiku, kas, iespējams, tiks novirzīts uz TV ierīcēm, un beidzot tabula 1.4 parāda trafiku, ko ģenerēs klasiskie TV raidījumu veidotāji un tālāk ISP piedāvās saviem klientiem sava apakštīkla robežās.

Būtu jāpiemin arī tas, ka šīs prognozes neņem vērā P2P trafiku, bet gan lejupielādējamo vai arī tiešsaistē skatāmo video saturu. Tas savukārt nozīmē, ka tuvākajā nākotnē tīklos dominēs QoS prasīgās sakaru sesijas, kuru caurlaides joslas prasības sniegsies no dažiem desmitiem līdz pat dažiem simtiem Mb/s. Jāņem vērā arī tas, ka video plūsmām ir stipri izteikta uzliesmojoša daba, kas padara to vadīšanu tīmeklī par jo vairāk sarežģītu uzdevumu [10], [9], [42].

1.2. tabula

## Globāls lietotāju interneta Video-to-PC trafiks, 2009-2014

<b>Lietotāju Video-to-PC trafiks, 2009-2014</b>							
	2009	2010	2011	2012	2013	2014	CAGR 2009-2014
Katrā no apašsegmentiem (PB/mēnesī)							
Short Form	717	997	1,415	2,011	2,754	3,727	39%
Long Form SD	1,132	1,707	2,286	2,534	2,687	3,286	24%
Long Form HD	405	989	2,195	3,541	5,068	6,432	74%
Long Form 3D	0	2	13	43	106	226	259%
Internet Video-Live TV	285	449	702	1,033	1,456	1,928	47%
Internet PVR	99	218	357	628	1,026	1,600	74%
Ambient Video	137	364	750	1,236	1,740	2,270	75%

1.3. tabula

## Globāls lietotāju interneta Video-to-TV trafiks, 2009-2014

<b>Lietotāju interneta Video-to-TV, 2009-2014</b>							
	2009	2010	2011	2012	2013	2014	CAGR 2009-2014
Katrā no apašsegmentiem (PB/mēnesī)							
Gaming Consoles Streaming Video	47	112	277	488	776	1,063	86%
Internet-Enabled TVs	1	5	27	108	316	686	285%
Internet-Enabled non-SP STBs	3	10	25	49	88	141	110%
Internet-Enabled SP STBs	47	118	343	799	1,417	2,062	113%
PC-TV Connections	5	11	24	40	61	84	73%
Placeshifting-Standalone	3	7	15	18	28	40	67%

## Globāls lietotāju IP trafiks, 2009–2014

Lietotāju IP trafiks, 2009–2014							
	2009	2010	2011	2012	2013	2014	CAGR 2009-2014
Katrā no apašsegmentiem (PB/mēnesī)							
IPTV VoD	329	504	714	948	1,209	1,463	35%
Cable VoD	2,277	3,176	4,534	6,147	7,850	9,411	33%
Katrā no apašsegmentiem (PB/mēnesī)							
IPTV SD VoD Traffic	250	334	391	397	386	428	11%
IPTV HD VoD Traffic	79	169	323	549	821	1,031	67%
Cable SD VoD Traffic	1,309	1,558	1,758	1,850	1,874	2,200	11%
Cable HD VoD Traffic	945	1,551	2,601	3,906	5,237	6,039	45%
Hybrid IP Cable SD VoD Traffic	12	31	59	92	122	174	69%
Hybrid IP Cable HD VoD Traffic	10	36	115	296	608	982	148%
IPTV 3D VoD Traffic	0	0	1	1	3	5	183%
Hybrid IP Cable 3D VoD Traffic	0	0	0	0	1	2	330%
Cable 3D VoD Traffic	0	0	2	4	8	13	169%

Balstoties uz šo prognožu datiem, rodas jautājums par tehnoloģisko risinājumu, kas būtu spējīgs vadīt šādas trafika plūsmas.

Ir pilnīgi skaidrs, ka šādu trafika apjomu transportu spēj nodrošināt tikai ātrie optiskie tīkli. To vadības plakne, pamatojoties uz ITU-T vīziju par NGN tīkliem, būs GMPLS (RFC 3471) ar RSVP-TE (RFC3473). ITU-T savu izvēli pamato ar nepieciešamību pēc pilnīgi dinamiskās sakaru sesiju vadības, kuras vadītu pašas aplikācijas, diktējot savas QoS prasības [54]. GMPLS ar RSVP-TE tiek piedāvāts arī kā konekciju vadības platforma ASON tīkliem (G.7713.2). Tomēr pastāv problēma ar RSVP-TE protokola CAC algoritmu, kas atbild par jauno LSP iestatīšanu. Tas būtībā ir klasiskā sliekšņa–CAC realizācija, kas pieņem lēmumu par resursu piešķiršanu, balstoties tikai uz pieejamās caurlaides joslas platumu.

Rezumējot, viss iepriekš minētais dod šādu ainu – nākotnes video trafika dominante radīs pavisam jaunas prasības trafika vadības jomā un svarīgāka loma būs tieši ātrajam sesiju uzstādīšanas procesam optiskajos tīklos ar garantēto QoS parametru saglabāšanu jauniestatītajai sakaru sesijai. Tātad faktiski noteicoša loma būs LSP dinamiskajai iestatīšanai GMPL tīklos. Ņemot vērā trafika neparedzamo raksturu īsos laika nogriežņos un nepieciešamību pieņemt lēmumu tiešsaistes režīmā, ir jāpieāvā kāds jauns risinājums trafika vadīšanai. Tam ir jābūt spējīgam darboties nenoteiktības apstākļos ar uzliesmojošo trafiku un jāspēj nodrošināt QoS parametru analīzi CAC lēmuma pieņemšanas momentā.

Šādu uzdevumu risinājumiem lieliski piemērota ir faziloģika, kas spēj nodrošināt ātru lēmuma pieņemšanu, abstrahējoties no precīziem ieejas datiem, kas atspoguļo fiziskus mērījumus. Fazi izvedumu sistēmas darbībai pietiek ar aptuvenu lingvinistisko vērtību skalu. Pielietojot lingvinistiskus mainīgos, fazi izvedumu sistēmas loģiku var balstīt uz kompakto zināšanu bāzi IF-THEN kārtulu veidā bez nepieciešamības veidot masīvas nosacījumu datubāzes. Visas šīs un citas faziloģikas īpašības tiks sīkāk izskatītas šī darba turpmāk dotajās nodaļās, pierādot faziloģikas pielietojuma efektivitāti CAC uzdevumu risinājumos mūsdienu trafika vadības CAC uzdevumos.

## **1.2. Faziloģikas pamatkonceptija**

Faziloģika faktiski ir klasiskās loģikas paplašinājums, kur tiek apskatīta iespēja darboties ar daļējās patiesības jēdzienu. Faziloģika mūsdienās aizvien plašāk kļūst par vienu no veiksmīgākajiem tehnoloģiskajiem risinājumiem komplekso adaptīvo vadības sistēmu izstrādē. Tā faktiski ir modelēšanas metodoloģija, kas var tikt pielietota, lai risinātu reālās dzīves problēmas. Fazikopu teorija, kas ir pamatā faziloģikas koncepcijai, atgādina cilvēka spriešanas jeb lēmumu pieņemšanas taktiku un pielieto aptuvenu jeb daļēju informāciju par sistēmas stāvokli un ieejas mainīgajiem. Tā tika pamatā pielietota, lai matemātiski attēlot nenoteiktību un aptuvenu informāciju par apskatāmajiem parametriem, kā arī lai piedāvātu rīkus, kuras būtu spējīgi darboties ar neprecizitātēm, kas ir sastopamas ļoti daudzās inženierijas uzdevumu problēmās [35].

Galvenā faziloģikas īpatnība ir spēja pārnest ieejas parametru veidoto momentāno ainu uz izejas parametra vērtībām, balstoties uz IF-THEN kārtulām. Faziloģika var palīdzēt risināt problēmas, kuru pamatā ir matemātiskā sarežģītība. Tā strādā ar fazi terminiem, fazi kopām un fazi operācijām un ļauj pieņemt lēmumus, balstoties uz fazi noteikumiem, kas ir aprakstīti ar IF-THEN kārtulām [75].

**Faziloģikas granularās skaitļošanas pamatā ir ideja, ka vairumā gadījumu, pieņemot lēmumu, nav nepieciešami precīzi fizisko vērtību dati un pietiek ar aptuvenu lingvinistisko vērtību skalu.**

Izsmelšu informāciju par faziloģikas teorētiskajiem pamatiem, tās matemātisko aparātu un plašiem pielietojuma aspektiem ir iespējams iegūt no daudzajām tās pamatlicēja profesora *Lotfali Askar-Zadeh* zinātniskajām publikācijām [119 - 142].

### **1.3. Ieskats faziloģikas attīstībā**

Faziloģikas koncepciju pirmais piedāvāja profesors *Lotfali Askar-Zadeh*, kura 1965. gadā publicētais radošais darbs „*Fuzzy Sets*” [129] kļuva par savdabīgo atskaites punktu faziloģikas jomā. Pati ideja ir diezgan vienkārša: ja vajag noteikt kāda mainīgā piederību noteiktai vērtību skalai, tad jebkāda veida spriedumi nesīs vairāk informācijas, ja mums būs nepārtraukta piederības gradācija. Faziloģikas spriedumi ir bāzēti uz prof. *Zadeh* piedāvāto nesavietojamības principu, kas nosaka, ka augsta precizitāte principā nav saderīga ar augstu sarežģītības pakāpi. Prof. *Zadeh* redzējums bija šāds: sistēmas sarežģītība un precizitāte, ar kādu tā var tikt analizēta atrodas loģiskajā pretstatā. Zinātnieks apgalvoja – kopš reālās pasaules uztvere savā būtībā atbilst neskaidru kopu īpašībām, tad pastāv pamatots iemesls lai ieviestu šo koncepciju arī skaitļošanas algoritmos. No pašas faziloģikas teorijas aizsākšanas to pavadīja daudzi strīdi. Angļu valodā runājošajās valstīs vārdam „*fuzzy*” kādreiz bija diezgan neviennozīmīga asociācija ar kaut ko nenoteiktu, neprecīzu, izplūdušu, kam bija pat negatīva nozīme. Bet jau 1990. gadā vārds „*fuzzy*” Japānā tika godalgots kā jauns vārds, kas apzīmēja jaunu intelekta izpausmi skaitļošanā. Pašlaik vārds „*fuzzy*” uzvarēja cīņu par savu pastāvēšanu un deva vārdu neskaitāmiem komerciālajiem produktiem, kas strādā un ikdienā pierāda faziloģikas darbību skaitļošanas tehnikas jomā.

Faziloģikas sistēmas 80. gados tika lielā mēra ignorētas gan Eiropā, gan ASV, toties Japāna ļoti aktīvi iesaistījās faziloģisko sistēmu izpētē un attīstībā, un vislielāko interesi par tām izrādīja *Seiji Yasunobu* un *Soji Miyamoto* no koncerna *Hitachi*. 1985. gadā viņi piedāvāja simulācijas eksperimentu rezultātus, kas viennozīmīgi pierādīja faziloģisko vadības sistēmu pārākumu, lai nodrošinātu *Sendai* industriālā kravu pārvadāšanas dzelzceļa vadību. Viņu idejas tikušas īstenotas, un 1987. gadā, kad dzelzceļš tika atvērts, faziloģikas sistēmas tika izmantotas, lai kontrolētu vilcienu sastāvu paātrinājumu, bremzēšanu un apstāšanos.

1987. gadā notika arī cits svarīgs notikums, kas palīdzēja paātrināt faziloģikas attīstību pasaulē. Kad šajā gadā Tokijā notika starptautiska faziloģikas pētnieku sanāksme, Japānas zinātnieks *Takeshi Yamakawa* demonstrēja faziloģikas iespējas, rādot „apgrieztā svārsta” eksperimentu. Tā ir klasiskā vadības problēma, kurā mehānisms mēģina noturēt nelielu apgriezti novietotu svārstu, kas ir piestiprināts ar kustīgu šarnīru vertikāli, kustoties uz priekšu un atpakaļ. Skatītāji bija sajūsmā, kad uz svārsta tika novietota glāze ar ūdeni un pat dzīvā pele. Sistēma spēja saglabāt stabilitāti visos šajos gadījumos. Vēlāk *Takeshi Yamakawa* organizēja vienu no pirmajām specializētajām faziloģikas izpētes laboratorijām, ko nosauca „*Fuzzy logic systems institute*” (FLSI) un ļoti veiksmīgi realizēja savus daudzus patentus šajā jomā. [32]

Japāņu zinātniekus ļoti aizrāva fazi sistēmu iespējas un tieši šajā valstī tika radītas pirmās faziloģikas pielietojamas iespējas gan industriālās mašīnās, gan arī daudzos plaša patēriņa produktos. 1988. gadā Japānā tika nodibināta „*Laboratory for International Fuzzy Engineering*”, saīsinājumā *LIFE*, kas apvienoja 48 kompāniju pūliņus ar nolūku nodrošināt kvalitatīvu faziloģikas iespējamo pielietojumu izpēti. [30] Pēc tam japāņu kompānijas laida pasaulē ļoti plašu preču klāstu, kas izmantoja faziloģiku vadības procesu nodrošināšanai, tajā skaitā arī tādas ikdienišķas lietas kā veļas mazgājamās mašīnas, gaisa kondicionētāji, fotokameras ar automātisko attēla fokusēšanu un pat rīsu vārīšanas mašīnas. Tāds autobūves gigants kā *Nissan*, sāka pielietot faziloģiku dzinēju kontroles blokus un kruīza kontroles vadībā. Japānas Seismoloģijas institūts pielieto faziloģiku zemestrīču prognozēšanas sistēmās, bet *Hitachi* mūsdienās pielieto faziloģikas kontrolierus metro vadības sistēmās.

Arī Eiropā un ASV faziloģika ar laiku guva atzinumu un klajā nāca daudzi produkti, kas izmantoja faziloģikas vadītus kontrolierus, tomēr Japāna bija un paliek faziloģikas izmantošanas pionieris.

Pašlaik Eiropā darbojas *European Society for Fuzzy Logic and Technology (EUSFLAT)* [34], kas par saviem mērķiem izvirzīja faziloģikas ideju un ar to saistītu priekšmetu izplatību un veicināšanu, kā arī, lai nodrošinātu ideju apmaiņas platformu starp zinātniekiem un inženieriem, kas strādā faziloģikas jomā.

Nobeigumā gribētos pieminēt, ka pat mūsu kaimiņvalstī Polijā mūsdienās ražotie granulu apkures katli pielieto uz faziloģiku balstītus kontrolierus gaisa padeves regulēšanai degkamerā.

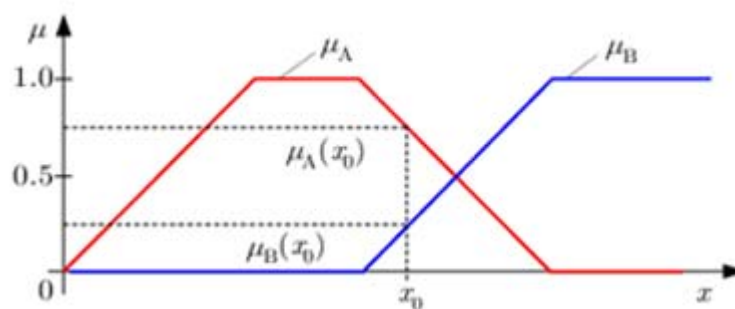
## 1.4. Faziloģikas izvedumu sistēmas uzbūve

Šajā nodaļā tiks apskatīti fazi izvedumu sistēmas svarīgākie aspekti, tādi kā piederības funkciju deklarāciju tipi, ieejas mainīgo fazifikācija, faziloģikas pamata loģiskās operācijas, fazi izvedumu sistēmu tipi, kā arī izvada mainīgo defazifikācija.

### 1.4.1. Piederības funkciju deklarācijas

Katram ieejas parametra lingvinistiskajam mainīgajam tiek deklarēta unikāla piederības funkcija. Piederības funkcija  $\mu_A(x)$  apraksta  $x$  elementu, no bāzes kopas  $X$ , piederības pakāpi pie fazi-kopas  $A$ , līdz ar to  $\mu_A(x)$  var attēlot plašu funkciju spektru. Visbiežāk tiek pielietotas vienkāršas funkcijas, kas bāzējas uz lineāro funkciju salikumiem, kā, piemēram, trīsstūra un trapecveidīgās.

Piederības funkcijas  $\mu_A(x)$  piederības pakāpe  $\mu_A(x_0)$  apraksta kāda noteikta fazi-kopas elementa  $x = x_0$  piederības pakāpi pie fazikopas  $A$ . Šī vērtība ir robežās  $[0,1]$ . Protams, ka  $x_0$  var vienlaicīgi piederēt arī pie kādas citas fazi-kopas  $B$ , kur  $\mu_B(x_0)$  raksturo  $x_0$  piederības pakāpi pie  $B$ . Šāds gadījums ir uzrādīts attēlā 1.1.



1.1. att.  $x_0$  piederības pakāpes pie kopām  $A$  un  $B$ .  $\mu_A(x) = 0.75$  un  $\mu_B(x_0) = 0.25$

Faktiski piederības funkcijas attēlo grafisko reprezentāciju piederības līmeni pie kāda no ieejas parametriem. Pašas IF-THEN kārtulas pielieto ieejas parametru piederības funkcijas kā „svara koeficientus”, lai noteikt to ietekmi uz fazi izvada vērtībām.

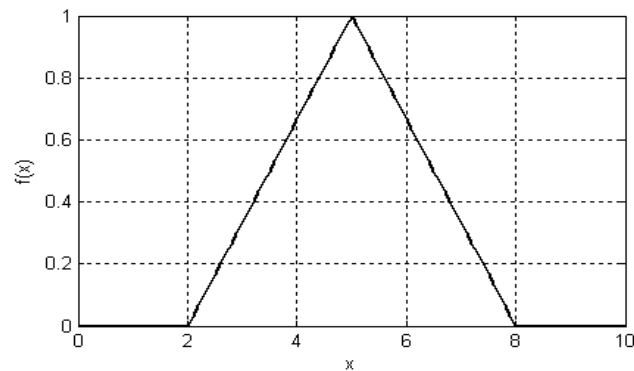
Pārsvārā tiek pielietotas trīsstūra formas piederības funkcijas, bet pielieto arī Gausa funkciju, trapecveidīgo, vispārinātu zvanveida funkciju un sigmoidālo funkciju. Ir iespējams pielietot arī sarežģītākās formas piederības funkcijas, bet tas tomēr prasa daudz lielākas

papildu skaitļošanas jaudas. Turpmāk ir dotas klasisko faziloģikas piederības funkciju definīcijas un attēlojumi.

### Trīsstūra piederības funkcija

Trīsstūra piederības funkcija tiek aprakstīta ar trim parametriem  $\{a,b,c\}$ ,  $a < b < c$  un tiek definēta šādi:

$$\mu_{Trīsstūra}(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (1.1)$$

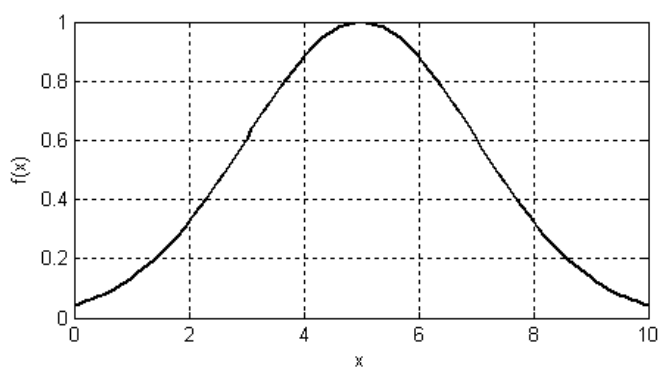


1.2. att. Trīsstūra piederības funkcija

### Gausa piederības funkcija

Gausa piederības funkcijai ir divi parametri:  $\bar{x}$  – kas ir atkarīgs par funkcijas centru, un  $\sigma$  – kas ir atkarīgs par funkcijas platumu. Gausa piederības funkcija tiek definēta šādi:

$$\mu_{Gausa}(x) = \exp\left(-\left(\frac{x-\bar{x}}{\sigma}\right)^2\right) \quad (1.2)$$

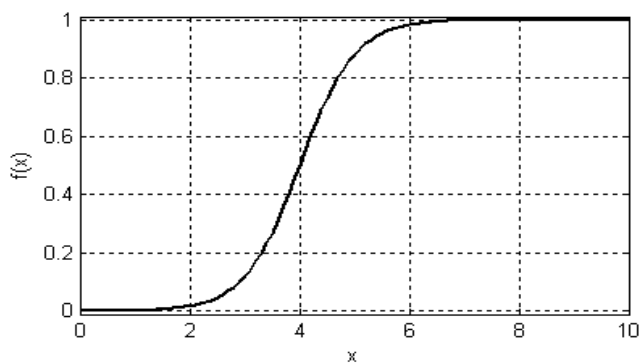


1.3. att. Gausa piederības funkcija

### Sigmoidālā piederības funkcija

Sigmoidālā funkcija tiek aprakstīta ar diviem parametriem:  $a$  – kurš ir atbildīgs par slīpumu krustošanas punktā,  $x=c$

$$\mu_{sigm}(x) = \frac{1}{1 + \exp[-a(x-c)]} \quad (1.3)$$

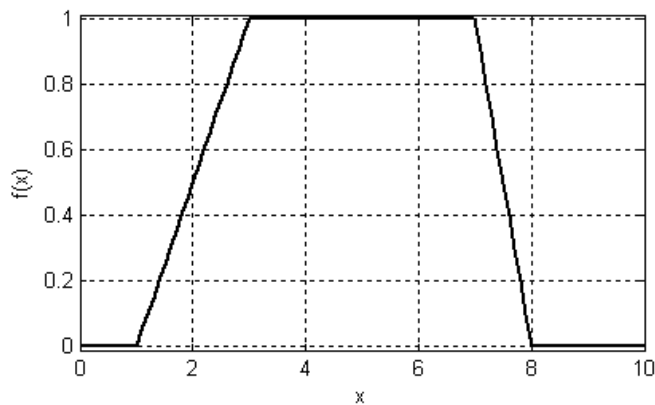


1.4. att. Sigmoidālā piederības funkcija

### Trapecveida piederības funkcija

Trapecveida piederības funkcija tiek aprakstīta ar četriem parametriem  $\{a,b,c,d\}$ ,  $a < b < c < d$  un tiek definēta šādi:

$$\mu_{Trap}(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & d \leq x \end{cases} \quad (1.4)$$

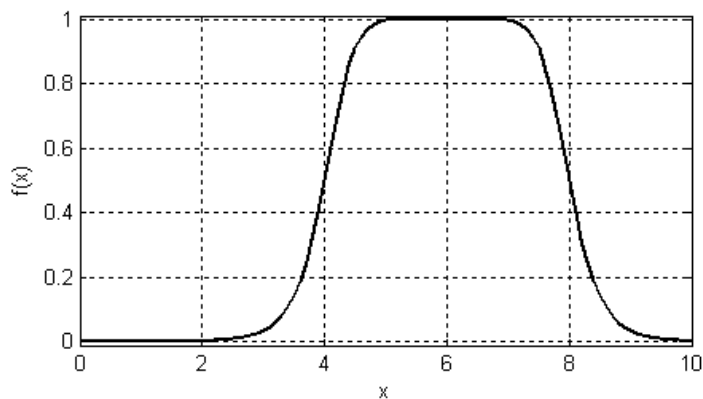


1.5. att. Trapecveida piederības funkcija

### Vispārināta zvanveida piederības funkcija

Vispārināta zvanveida piederības funkcija tiek aprakstīta ar trim parametriem:  $a$  – nosaka funkcijas platumu,  $c$  – nosaka funkcijas centru un  $b$  – nosaka funkcijas slīpumu. Funkcija tiek definēta šādi:

$$\mu_{Zvanv}(x) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (1.5)$$



1.6. att. Vispārināta zvanveida piederības funkcija

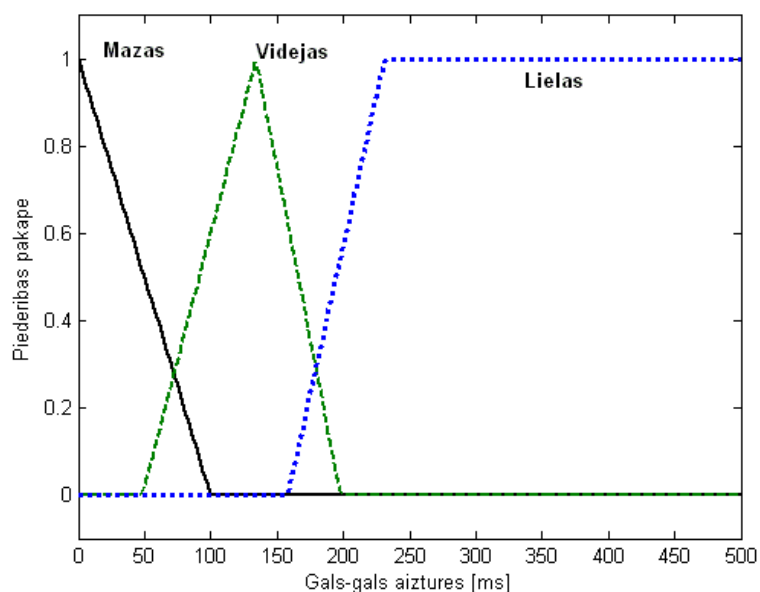
### 1.4.2. Ieejas mainīgo fazifikācija

Fazifikācija ir noteiktu ieejas parametru vērtību pārveidošanas process par fazi-kopu lingvinistisko mainīgo piederības pakāpēm. Piederības funkcijas tiek pielietotas, lai piesaistītu lingvinistisko mainīgo ar to piederības pakāpi pie noteiktas fazikopas. Rezultātā reālās skalārās vērtības tiek pārvērstas par fazi-vērtībām. Fazifikācija visbiežāk notiek, pielietojot vienkāršas formas piederības funkcijas, tādās kā trīsstūra vai Gausa. Sīkāks piederības funkciju apskats ir dots šī promocijas darba nodaļā 1.4.1.

Reālo vērtību fazifikācija notiek, balstoties uz intuīciju, pieredzi, kā arī analizējot ar vēlamo sistēmas darbību saistītas IF-THEN kārtulu kopas un ieejas vērtības. Nav noteiktu procedūru, kas nodrošinātu optimālu fazifikācijas procesu [102]. Tas savukārt nozīmē, ka fazifikācijas laikā var tikt ieviesta noteikta deva subjektīvās lēmēj mehānisma konstruktora izpratnes par vadāmās sistēmas pielietoto mainīgo robežvērtībām.

Piemēram, definējot linka aiztures tiešsaites aplikāciju vajadzībām, tādām kā VoIP vai VoD, jēdziens „mazas” vai „lielas” aiztures būs pavisam citas nekā, ja tādās pašas lingvinistisko mainīgo deklarācijas tiks attiecinātas uz SMTP vai HTTP trafiku.

Iztēlosimies fazi-CAC algoritma realizāciju, kas IF-THEN kārtulu zināšanu bāzē dod vēlamo lēmējsistēmas reakciju tiešsaites trafikam ar augstu jutību attiecībā uz pakešu aizturēm tīklā. Tas varētu tikt attiecināms, piemēram, uz VoIP aplikācijām. Ņemot vērā, ka VoIP paketes ir ļoti jutīgas attiecībā uz „gals-gals” aizturēm, kas ir augstākas par 150 ms, šajā piemērā šādas vērtības fazifikācijas procesā būtu vēlams deklarēt kā fazikopas „lielas aiztures” centrālo atskaites punktu, gadījumā ja varam lēmuma pieņemšanas laikā operēt ar pakešu aptuveni zināmajām „gals-gals” aizturēm konkrētajā savienojuma maršrutā. Tālāk, jau IF-THEN kārtulu zināšanu bāzes ietvaros, ir iespējams deklarēt arī toleranci attiecībā uz lielākām aizturēm, kaut vai ņemot vērā klientu vēlmi lietot balss sakarus ar lielākām aizturēm par zemākām izmaksām. Pat tikai divi pakešu ceļi no zemes uz satelītu un atpakaļ var ienest ap 500ms paketes aizturi, bet daudzu aplikāciju, tādu kā *Skype* lietotāji ir ar mieru paciest arī sliktākus sakaru kvalitātes radītājus apmaiņā pret zemām pakalpojuma izmaksām. Tātad šādu aplikāciju sakaru sesiju vadībā var deklarēt citu, daudz pielaidīgāku attiecībā uz aizturu vērtībām, fazikopas „lielas aiztures” piederības funkciju.



1.7. att. Fazificētas pakešu „gals-gals” aizturu vērtības

Attēlā 1.7 varam redzēt fazificētos pakešu „gals-gals” aizturu lingvinistiskos mainīgos, kas savu piederību pie deklarētām fazikopām „mazas”, „vidējās” un „lielas” definē ar trīsstūra un lineāro piederības funkciju „lielo” aizturu gadījumā.

### 1.4.3. Pamata faziloģiskās operācijas un izvedumi

Balstoties uz faziloģikas pamatlicēja prof. *Zadeh* piedāvātajām standarta faziloģikas loģisko operāciju definīcijām, varam izšķirt šādas operācijas.

1. NEGĀCIJA (nolieguma kritērijs) : patiesība ( $N\bar{E} x$ ) = 1.0 – patiesība(x)

$$\mu_{n\bar{e}A}(x) = 1 - \mu_A(x) \quad (1.6.)$$

2. UN (krustošanās jeb minimuma kritērijs) : patiesība(x un y) = minimums(patiesība(x), patiesība(y))

$$\mu_{A\cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad (1.7.)$$

3. VAI (apvienošanas jeb maksimuma kritērijs) :  $\text{patiesība}(x \text{ vai } y) = \text{maksimums}(\text{patiesība}(x), \text{patiesība}(y))$

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad (1.8.)$$

Uz iepriekš minētajām faziloģikas loģiskām operācijām balstās fazi izvedumu sistēmas darbība, nodrošinot IF-THEN kārtulu zināšanu bāzes nosacījumu izpildi, pielietojot ieejas parametru lingvinistisko mainīgo vērtības.

#### 1.4.4. Faziloģikas izvedumu sistēmas tipi

No izvedumu procesa viedokļa, ir iespējams izšķirt divus pamata Fazi Izvedumu Sistēmas (FIS) tipus:

- a. *Mamdani* tipa izvedumu sistēma [66];
- b. *Takagi-Sugeno-Kang* izvedumu sistēma [103].

Galvenā atšķirība starp *Mamdani* un *Takagi-Sugeno* fazi izvedumu sistēmām ir dažāda tipa izvada piederības funkciju definīcijas. *Sugeno* tipa fazi sistēmas satur vai nu lineārās izvada piederības funkcijas, vai konstantes. *Mamdani* fazi izvedumu sistēma var saturēt jebkura tipa izvada piederības funkciju definīcijas. Pielietojuma ziņā *Mamdani* tipa izvedumu sistēmas ir daudz plašāk pielietotas, jo tās nodrošina pietiekošu rezultātu ar salīdzinoši vienkāršo struktūru, kā arī, pateicoties tam, ka tā pielieto intuitīvi viegli saprotamu IF-THEN kārtulu zināšanu bāzi [51]. *Takagi-Sugeno-Kang* (jeb saīsināti *Takagi-Sugeno*) izvedumu sistēmas gadījumā, IF-THEN kārtulu nosacījumu sekas nav definētas kā fazi vērtības, un tas noved pie intuitīvi saprotamās fazi sistēmas interpretācijas zuduma. Jāņem vērā arī tas, ka *Mamdani* FIS var tikt pielietots pa tiešo MIMO sistēmu uzbūvē, kas satur vairākus izvada argumentus, kā arī MISO sistēmu uzbūvē, kas var saturēt tikai vienu izvada parametru. Turpretim *Takagi-Sugeno* FIS var tikt pielietots, tikai veidojot MISO fazi sistēmas.

Salīdzinot *Mamdani* un *Takagi-Sugeno* FIS varam izcelt dažas svarīgākas nianšes

- No skaitļošanas viedokļa *Takagi-Sugeno* FIS ir efektīvākas, jo, salīdzinot ar *Mamdani* FIS pielietoto komplicēto defazifikāciju, *Takagi-Sugeno* FIS pielieto svērto vidējo attiecībā uz izvada vērtībām.
- *Mamdani* FIS ir vienkāršāka un dod iespēju pielietot intuitīvi viegli saprotamu IF-THEN kārtulu zināšanu bāzi, ko var veidot, pielietojot lingvinistiski vienkārši izteiktas ekspertu zināšanas.

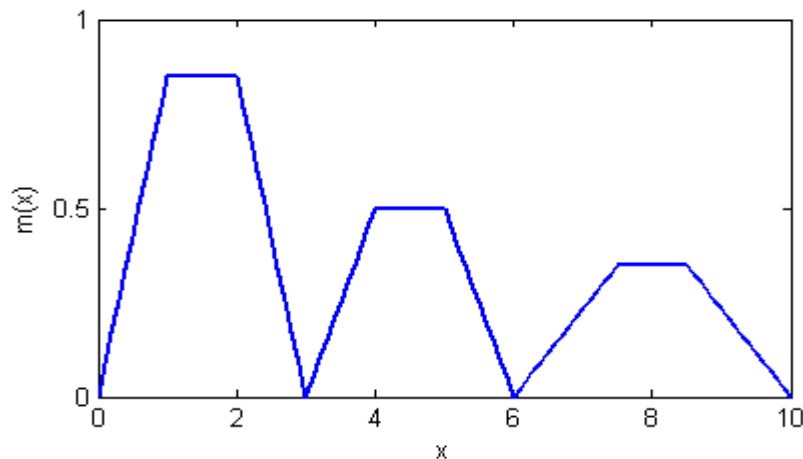
### 1.4.5. Izvada mainīgo defazifikācija

Defazifikācija ir ļoti svarīga operācija fazikopu teorijā. Defazifikācija nodrošina fazikopas informācijas transformāciju skaitliskajā informācijā. Šī darbība kopā ar fazifikāciju ir kritiski svarīga, lai konstruētu korektu fazi izvedumu sistēmu, jo abas šīs operācijas nodrošina saikni starp fazikopu un reālo skalāro vērtību definīcijas apgabaliem [94].

Tā kā fazi izvads tiek konstruēts, uzliekot visu individuālo IF-THEN kārtulu rezultātus vienu otram pa virsu jeb pielietojot faziloģisko apvienošanas operāciju, tad rezultātā iegūtais fazi izvads attēlo vairāku aktīvo kārtulu rezultātus izvadu fazikopu veidā.

Pastāv daudzas defazifikācijas metodes, tādas kā RCOM, FOM, MeOM, BADD, GLSD, ICOG, SLIDE, FM, WFM, QM, EQM, CDD, FCD [52] [63] un citas, bet šajā darbā tiks apskatīti tikai pieci vispopulārākie defazifikācijas paņēmieni: COG, Bisector, MOM, SOM un LOM.

Lai apskatītu defazifikācijas metožu ietekmi uz defazificētās vērtības izvadu, izmantosim iespējamo fazi lēmēj mehānismu ar trim trīsstūra izvada piederības funkcijām izvadā (Att. 1.8).

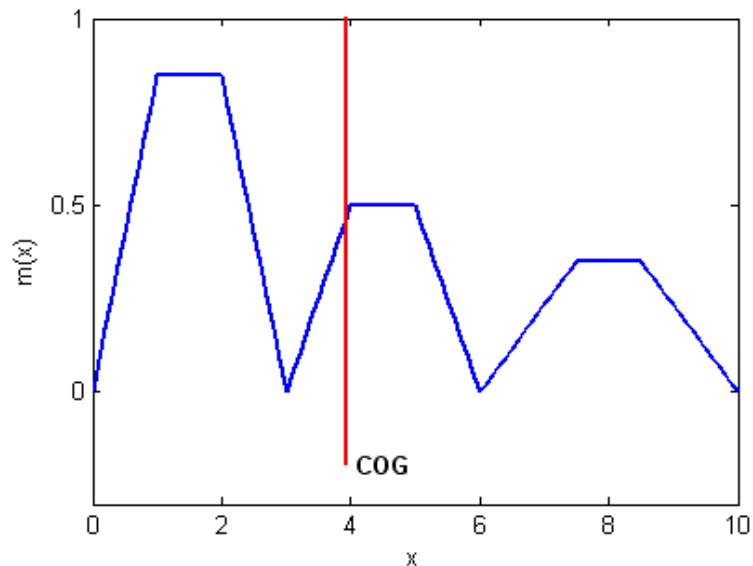


1.8. att. Fazi izvedumu sistēmas trīs izvada fazikopas

1. **COG** jeb gravitācijas centra metode, kas tiek saukta arī par centroida defazifikācijas metodi. Šī metode nosaka virsmas centra punktu zem līknes. Ja iztēlojamies, ka virsma zem līknes ir plātne ar vienādu blīvumu, COG izvads būtu punkts, uz kura virsma atrodas līdzsvara stāvoklī (Att. 1.9). COG rezultātu varam izteikt ar sekojošu izteiksmi (1.9.):

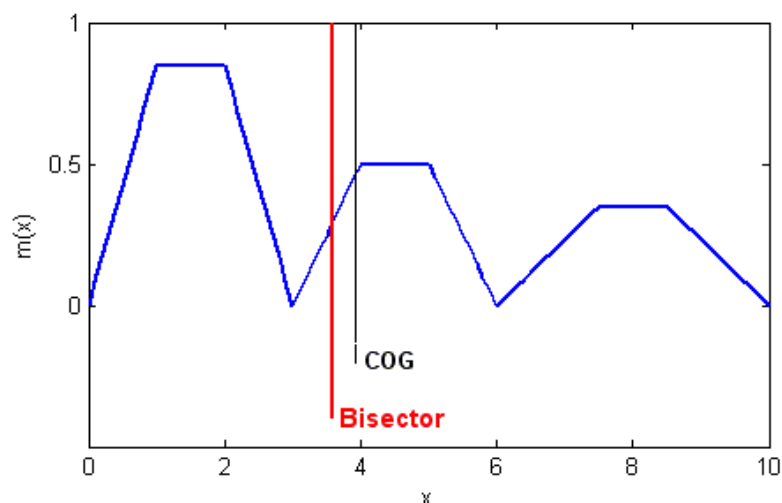
$$z_0 = \frac{\int_{\min}^{\max} x \mu(x) dx}{\int_{\min}^{\max} \mu(x) dx}, \quad (1.9.)$$

kur  $z_0$  ir defazificētā skalārā vērtība un min un max ir defazificējamās virsmas  $x$  ass robežas, kurās tiek apskatītas  $\mu(x)$  izvada fāziskopu piederības funkcijas.



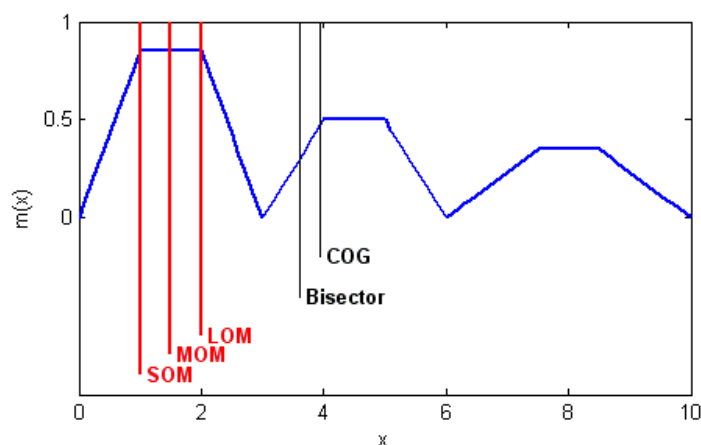
1.9. att. COG metodes atgrieztā skalārā defazificētā vērtība

2. **Bisector** defazifikācijas metode. Dotais defazifikācijas paņēmieni par skalāro vērtību nosaka  $x$  vērtību, pie kādas vertikāli vērsta līnija sadalītu izvada piederības funkciju kopējo virsmu divos reģionos ar vienādām virsmām. Šī vērtība ir bieži vienāda ar COG metodes atgriezto vērtību (Att. 1.10).



1.10. att. Centroid un COG metožu atgrieztās skalārās defazificētās vērtības

3. **MOM** (*Middle of maximum*), **SOM** (*Smallest of maximum*) un **LOM** (*Largest of maximum*) metodes nosaka vidējo, mazāko un lielāko vērtību no lielākās sakopotās piederības funkcijas. Šinī piemērā sakopotai piederības funkcija ir plato, un tāpēc MOM, SOM un LOM skalārās vērtības ir atšķirīgas (Att. 1.11). Gadījumā, ja sakopotā piederības funkcija satur kādu unikālu maksimumu, tad doto defazifikācijas metožu rezultāti sakrītīs.



Att. 1.11. Centroid, COG, SOM, MOM un LOM metožu atgrieztās skalārās defazificētās vērtības

Defazifikācijas tehnikai, kas spētu saglabāt visu fazi atbilžu sniegto informāciju, ir jāņem vērā visu fazikopu rezultāti. Šādu rezultātu sniedz, piemēram, t.s. centroida jeb COG

defazifikācijas metode, kas defazifikācijas rezultātā atgriež virsmas centra punktu zem piederības funkciju līknes, un kā defazificētā vērtība tiek pielietota  $X$  koordināta.

## 1.5. Faziloģikas pielietojums telekomunikācijās

Šajā nodaļā autors ļoti īsi un konspektīvi pievērsīsies telekomunikāciju sistēmu jomām, kas jau tika apskatītas citu autoru pētījumos un parāda faziloģikas pielietojuma plašās iespējas un efektivitāti sakaru sistēmu trafika vadības uzdevumu jomā.

### 1.5.1. Faziloģikas pielietojums maršrutēšanā ar QoS nodrošinājumu

Maršrutēšanas uzdevumi, saglabājot nepieciešamos plūsmas QoS parametrus, ir pamats ātrgaitas tīklu efektīvai darbībai, nodrošinot multimediju datu sūtīšanu starp tīkla elementiem. Šiem uzdevumiem ir divi mērķi:

- atrast maršrutus, kas sader ar QoS ierobežojumiem;
- nodrošināt efektīvu tīkla resursu izmantošanu.

Lai nodrošinātu iepriekš minēto mērķu sasniegšanu, ir jārisina komplekss uzdevums. Tam vienlaicīgi jāietver vairāki ierobežojumi, pieņemot lēmumus par maršrutēšanas ceļa izvēli.

Faziloģikas pielietojums maršrutēšanas uzdevumu risinājumā ļauj vienlaicīgi manipulēt ar vairākiem ieejas parametriem un, rēķinoties ar vairākiem ierobežojumiem vienlaicīgi, darīt to ļoti vienkāršā un intuitīvā veidā [90].

Šī promocijas darba autora atrastajā literatūrā ir piedāvāti ne tikai jauni algoritmi, bet arī esošo maršrutēšanas algoritmu modifikācijas ar faziloģikas palīdzību, kaut gan faktiski visus atrastos risinājumus vieno iespēja vienlaicīgi pielietot vairākus ieejas parametrus, pieņemot lēmumu par vēlamo maršrutēšanas ceļu un balstīt lēmumu pieņemšanas loģiku uz maza izmēra IF-THEN kārtulu zināšanu bāzes.

Piemēram, *Susan Rea* un *Dirk Pesch* savā pētījumā [85] piedāvā DSR maršrutēšanas algoritma uzlabojumus trafika vadībai ad-hoc tīklos. *Mirabedini, S.J et al.* [70] iedvesmoja skudru algoritma potenciāls, kas tika papildināts ar faziloģikas iespējām, radot FLAR adaptīvo maršrutēšanas algoritmu. Autoru rezultāti uzrāda ievērojamus uzlabojumus, salīdzinot FLAR ar AntNet2.0 un OSPF.

*Alandjani un Johnson* 2003. gada pētījumā [3] arī pievēršas bezvadu ad-hoc tīklu tēmai un piedāvā jaunu maršrutēšanas shēmu, kas, pielietojot faziloģiku, nodrošina diferencētu resursu iedalīšanu, ņemot vērā trafika svarīgumu un tīkla stāvokli. Autori piedāvā redundanti sūtīt svarīgākās paketes paralēli, izmantojot divus neatkarīgus sūtīšanas ceļus, ar nolūku paaugstināt svarīgu sakaru sesiju drošumu. Mazsvarīgs trafiks, pēc autoru piedāvātās shēmas, var tikt speciāli apspiests avota līmenī, lai nodrošinātu selektīvu trafika vadību ad-hoc tīklā. Autori salīdzina sevis piedāvāto uz faziloģikas balstīto maršrutēšanas algoritmu ar DSR un SMR bezvadu maršrutēšanas algoritmu.

Vēl viens interesants darbs ir veltīts faziloģikas pielietojumam ad-hoc tīklu vadībā. *Natheh Essam, Khatun Sabira un Jantan Adzan* 2006. gada publikācijā [74] apskata AODV maršrutēšanas protokola iespējamo modifikāciju. Autori, pielietojot faziloģisko lēmējmehānismu ART parametra dinamiskajai noteikšanai, nosaka laiku, uz kādu maršruts maršrutēšanas tabulā var tikt definēts kā aktīvs. Klasiski ART ir statiskā vērtība. Faziloģiskais interfeiss pieņem lēmumus, balstoties uz datiem par raidītāju un starpmezgliem, šādā veidā uzlabojot pakešu zudumu un aizturu radītājus.

### **1.5.2. Pārslodzes kontrole un AQM, TCP/IP tīklos ar faziloģiku**

Faziloģika tiek pielietota arī pārslodzes kontrolē. Piemēram, *Hamed, Yaghmaee un Tabatabaee* 2006. gada publicētajā pētījumā [76] ir piedāvāta uz faziloģiku balstīta TCP realizācija. Autori piedāvā t.s. lēnā starta pārblīves izvairīšanās algoritma vietā pielietot uz faziloģikas balstītu risinājumu, kas pielieto datus par esošā pārblīves loga izmēru un lēnā starta sliekšņa līmeni, kā arī par pēdējo un iepriekšējiem RTT. Autoru piedāvātais algoritms nosaka pārblīves loga izmēru, pielietojot fazi lēmējsistēmu. Rezultātā linka resursi tiek izmantoti optimāli, palielinot kopējo caurplūdes ātrumu.

2005. gada pētījumā, autori *Jin Wu un Karim Djemame* [114] piedāvā uz faziloģikas balstīto AQM algoritmu FAOM. NS2 simulācijas vidē tiek salīdzināti FAOM un plaši pielietots RED. Rezultātā autori parāda, ka FAOM interaktīvo multimediju aplikāciju gadījumā nodrošina daudz labāku „gals-gals” veiktspēju.

*C. Chrysostomou, A. Pitsillides, L. Rossides, A. Sekercioglu* 2003. gada pētījumā [25] piedāvā populārā RED algoritma modifikāciju ar faziloģiku – Fuzzy-RED. Šī algoritma pamatā ir ideja par RED definētā maksimālā un minimālā rindas robežlīmeņa maiņu pret

dinamiskajām sliekšņa vērtībām, kas ir atkarīgas no tīkla stāvokļa un tiek izskaitļotas ar faziloģikas kontroliera palīdzību, pielietojot kompakto IF-THEN zināšanu bāzi. Autori izpēta Fuzzy-RED algoritmu, pielietojot homogēnas un heterogēnas trafika plūsmas un dažādas noslodzes pakāpes. Rezultātā tiek parādīts, ka Fuzzy-RED nodrošina labākus utilizācijas rādītājus plašā noslodzes diapazonā.

## **1.6. Fazi-CAC algoritmi. Esošās realizācijas**

Faziloģikas pielietojums CAC kontrolē piedāvā jaunus un daudzsoļus risinājumus trafika vadības jomā. Tomēr Fazi-CAC algoritmu piedāvājumi ir publicēti salīdzinoši nedaudzās zinātniskās publikācijās, kas pamatā skar ATM un mobilo sakaru tīklu tematiku. Turpmāk tiks doti īsi ieskati dažos, pēc šī promocijas darba autora ieskata, interesantākajos fazi-CAC algoritmu piedāvājumos.

### **1.6.1. Fazi-CAC ATM sakaru sistēmās**

Visplašākais CAC risinājumu spektrs, kas balstīts uz faziloģikas interfeisu, tiek pārstāvēts tieši ATM sakaru sistēmās.

Kā vienu no pēdējiem pētījumiem šajā jomā varētu pieminēt *France Cheong*, un *Richard Lai* 2007. gada publikāciju [24], kas apskata faziloģikas iespējamo pielietojumu CAC algoritma uzlabojumiem ATM komutatoram tīklu veiktspējas palielināšanai. Autori piedāvā interesantu risinājumu ABR plūsmu kontrolei ar nolūku palielināt tīkla linku utilizācijas rādītājus darbam ar uzliesmojošām MPEG plūsmām. ATM piedāvā četras svarīgākās ATM slāņa servisu klases: CBR, VBR, ABR un UBR, kurām ir atšķirīgas QoS prasības. CBR un VBR plūsmas tiek pievienotas, balstoties uz esošiem CAC algoritmiem, bet ABR plūsmas aizpilda atlikušos esošos resursus linkā. ABR plūsmām ir ļoti zems jutīgums attiecībā uz aizturēm tīklā. Tā kā MPEG plūsmu uzliesmojošās dabas dēļ ir neiespējami paredzēt linka pieejamo resursu apjomus, lai tos izmantotu ABR plūsmām, autori piedāvā papildināt CAC algoritmu ar faziloģikas kontrolieri. Tas paralēli vada ABR plūsmas ATM tīklā, ar nolūku sasniegt labākus linku utilizācijas rezultātus. Kā ieejas parametrus autori izmanto ABR šūnu zudumus un zudumu izmaiņu dinamiku, un izvadā saņem vadības signālus ABR šūnu sūtīšanas ātruma izmaiņām. Rezultātā autori iegūst stabilas vērtības ABR šūnu bufera

izmēram arī nopietno pārslodžu gadījumos un ātru kontroliera reakciju, kas ir īpaši svarīga, darbojoties ar uzliesmojošo MPEG trafiku.

Kā nākamo pētījumu varētu pieminēt ļoti interesantu *Lachlan Andrew* un *Kuan Su-Hsien* [62] darbu, kura autori piedāvā modificēt jau esošo FERM algoritmu [81]. FERM savā oriģinālā veidā būtībā nevar nodrošināt robustu un efektīvu ABR trafika vadību ATM sakaru sistēmās lielu RTT gadījumos. FERM algoritms izrēķina katra trafika avota maksimālo datu caurlaides ātrumu un uz šo datu pamata pieņem lēmumu par ABR plūsmas pieslēgumu vai tās noraidīšanu. Fazi kontrolieris aktualizē savu lēmumu ik pēc katra kontroles laika intervāla. Lielu RTT esamība darbā ar ABR plūsmām padara algoritma darbu nestabilu, jo kontroles laiki ir fiksēti un ABR plūsmu caurlaides ātrumu pieauguma un mazinājuma tempi arī ir vienādi. Autori piedāvā divas pieejas ar nolūku modificēt fazi loģikas ABR plūsmu CAC kontrolieri. Pirmā pieeja skar IF-THEN kārtulu modifikācijas, kas tiek mainītas, lai nodrošinātu lēzenu ABR plūsmu ātrumu pieaugumu un vajadzības gadījumā ļoti ātru samazinājumu. Otrs paņēmieni skar fazifikācijas stadiju, kad ieejas parametros tiek ievadīti dati par bufera stāvokli. Tad vienkāršu fazifikācijas noteikumu izmaiņu rezultātā tiek palielināta vai arī pretēji – mazināta šī lingvinistiskām mainīgā nozīme, nemainot IF-THEN kārtulu datu bāzi. Šādā veidā tiek sasniegts līdzīgs efekts, nodrošinot lēnāku kontroliera reakciju dodot atļauju slēgt jaunas ABR plūsmas, un pretēji – paātrinot ABR plūsmu izslēgšanas ātrumu.

Ir vērts apskatīt vēl vienu rakstu, kas apskata faziloģikas pielietojumu ATM sistēmu vadības uzdevumu risinājumos, kas neskar tiešā veidā CAC vadību, bet gan ļoti radniecisku problēmu – buferatmiņas vadību.

*Constance Lightner*, 2006. gada rakstā [65] piedāvā uz faziloģikas bāzēto kontrolieri, kas būtu spējīgs izvietot pienākošās šūnas buferī, ņemot vērā šūnas sniegto informāciju par tās QoS prasībām, „gals-gals” aizturēm un informāciju par pārblīvēm tīklā. Vadoties pēc autora sniegtiem rezultātiem, šis algoritms spēj efektīvi minimizēt šūnu zudumu rādītājus ATM komutatoros.

### **1.6.2. Fazi-CAC CDMA sakaru sistēmās**

Kā vienu no spilgtākajiem piemēriem varētu minēt dažus pētījumus par iespējamo fazi-CAC pielietojumu CDMA tīklu resursu vadībā. Piemēram, *Jun Ye*, *Xuemin Shen* un *Jon W. Mark* piedāvā interesantu risinājumu [53], kur sākumā tiek iegūts mobilās stacijas pieprasītās

efektīvās caurlaides joslas platums un informācija par tās mobilitāti. Vēlāk tiek pieņemts lēmums par mobilās stacijas savienojuma akceptēšanu vai noraidīšanu, balstoties uz tās sniegtiem pieprasījuma datiem un sistēmas resursu pieejamību. Lai demonstrētu fazi-CAC algoritma efektivitāti, pētījumā tiek uzrādīti šādi skaitliski rezultāti:

- jaunu izsaukumu bloķēšanas varbūtība;
- aktīvu izsaukumu bloķēšanas varbūtība pārslēgšanas laikā uz citu bāzes staciju;
- aktīvā izsaukuma bloķēšanas varbūtība pastāvīgā pieslēgumā pie bāzes stacijas;
- resursu utilizācija.

Autori salīdzina piedāvātā FCAC algoritma efektivitāti ar RPCAC un NPCAC algoritmiem. FCAC parāda lieliskus rezultātus uzturot zemas jauno izsaukumu bloķēšanas varbūtības vērtības, pieaugot balss izsaukumu pienākšanas intensitātei, kā arī esošo savienojumu pārtraukšanas varbūtības vērtības, pieaugot balss zvanu pienākšanas intensitātei. FCAC uzrādīja arī vislabākos rezultātus salīdzinājumā ar RPCAC un NPCAC resursu utilizācijā, pieaugot datu savienojumu intensitātei.

Vēl viens fazi-CAC algoritma piedāvājums CDMA tīkla vadībai ir atrodams *Paolu Dini* un *Roberto Cusani* veiktajā 2005. gada pētījumā [33], kur tiek piedāvāts modificēts uz mērījumiem bāzēts CAC risinājums, kas pamatā analizē interferences/signāla attiecību. Tā kā fazilōģika ir spējīga darboties ar vairākiem ieejas parametriem vienlaicīgi, autori piedāvā ieejas parametrus papildināt ar pārslodzes raksturojošo parametru. Fazi-CAC ieejas parametrs tiek definēts ar krustojošām piederības funkcijām: „ir pārblīve” un „nav pārblīves”, kā arī ar pieejamo resursu raksturojošo parametru, kas tiek uzdots ar trim daļēji pārklātām piederības funkcijām: „mazi”, „vidēji” un „lieli”. Rezultātā tiek analizētas četras atbildes: „noraidīt”, „viegli noraidīt”, „viegli akceptēt”, „akceptēt”. Pētnieki simulācijas gaitā salīdzina piedāvāto fazi-CAC algoritmu ar ICAC un ACAC algoritmiem kā arī ar situāciju, kad CAC kontrole netiek pielietota. Tiek izmantoti trīs trafika modeļi, kas raksturo balss pārraidi (Puasona sadalījums), datu pārraidi, un fona trafiku (Pareto sadalījums). Rezultāti parāda, ka, analizējot sistēmas darbību gan ar fona trafiku, gan bez tā, fazi-CAC shēma dod labākus rezultātus attiecībā uz balss datu pārraidi. Tiek nodrošināta zemāka esošā izsaukuma nomešanas vai jaunā izsaukuma bloķēšanas varbūtība. Te gan būtu jāpiebilst, ka autori izmantoja konstantā ātruma trafika avotus ar nolūku ģenerēt fona trafiku, neparedzot trafika plūsmu uzliesmojumus katram avotam atsevišķi. Rezultāti parāda, ka, pielietojot fazi-CAC shēmu, izdevies nostabilizēt linku utilizācijas līmeni pie aptuvenas vērtības 0,9, salīdzinot ar citu

analizējamo CAC risinājumu sniegto utilizāciju pie vērtības  $\approx 0,8$ , kur trafika plūsmas palielinājums tika sasniegts, tieši pateicoties datu trafika pieaugumam linkā.

### 1.6.3. Fazi-CAC GPRS sakaru sistēmās

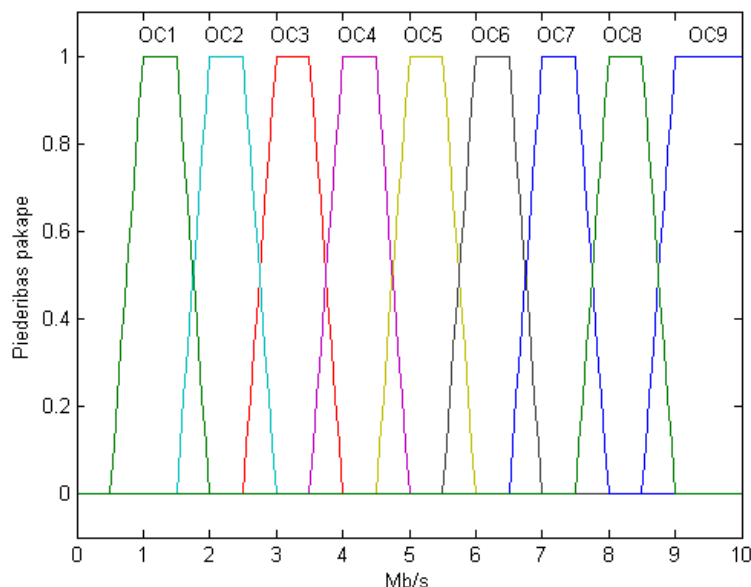
Fazi-CAC atrada savu pielietojumu arī GPRS/EGPRS mobilās datu pārraides pētījumos. *Doru Todinca, Ștefan Holban, Philip Perry un John Murphy* [107] 2004. gada pētījumā apskata iespējamo fazi-CAC algoritma realizāciju, kas būtu spējīga nodrošināt ne tikai efektīvāko linku utilizāciju, bet arī vienlaicīgi garantētu katram lietotājam vai aplikācijas grupai atšķirīgus QoS savienojuma parametrus. Autori piedāvā mainīt CAC algoritmu darbības paradigmu mobilajos tīklos, kas primāri balstās tikai uz tīkla noslogotību vai linka stāvokli no lietotāja līdz bāzes stacijai. Tie piedāvā pievērsties CAC problēmai vairāk kompleksi, apvienojot iepriekš minēto ar datu pārraidei svarīgajiem QoS kritērijiem, tādiem kā pakešu aiztures tīklā, pieslēguma prioritāte un dažādu trafiku paveidu raksturojošie QoS līmeņi (interaktīvās aplikācijas, e-pasts utt.). Kā vienu no iemesliem faziloģikas pielietošanai autori min nespēju risināt tik komplekso uzdevumu ar klasiskajiem CAC algoritmiem, kas savā būtībā ir uz sliedzīna-CAC bāzētas algoritmu modifikācijas, kā arī ar nespēju definēt šādas komplicētības uzdevuma CAC problēmu ar pietiekami kompaktu, skaitļošanai derīgu matemātisku aprakstu. Aprakstītais fazi-CAC algoritms ir samērā vienkāršs un, pēc autoru nostādnes, var tikt nepieciešamības gadījumā definēts ar lielāko ieejas parametru skaitu. Tas faktiski ņem vērā tīkla noslogotības stāvokli, kuru definē ar 3 trīsstūra piederības funkcijām: „augsti noslogots”, „vidēji noslogots” un „maz noslogots”. Tas pielieto arī lietotāju savienojuma prioritātes, kas tiek definētas līdzīgi: „augsta”, „vidēja” un „zema”. Prioritātes tiek izrēķinātas, pielietojot attiecīgus svāra koeficientus lietotāju trafika ģenerācijas laikā un balstās uz trim trafika kategorijām: „tiešsaistes trafiks”, „WWW trafiks” un „fona trafiks”. Fazi-interfeisa izvadā tiek analizēts mainīgs sliedzīns un, balstoties uz iepriekšnoteikto sliedzīna vērtību, tiek pieņemts lēmums par lietotāja konekcijas akceptēšanu vai arī tās noraidīšanu. Rezultāti parāda, ka fazi-CAC shēma ļauj elastīgi kontrolēt linka noslogotību, pielietojot tikai dažus lingvinistiskus mainīgos un nelielu skaitu IF-THEN kārtulu. Faziloģikas risinājums ļauj ņemt vērā vairākus ieejas parametrus ar nolūku saglabāt nepieciešamos pieslēguma QoS parametrus.

#### 1.6.4. Fazi-CAC MPLS sakaru sistēmās

Šī promocijas darba autora izpētītajā literatūrā tika atrasts tikai viens piemērs fazi-CAC algoritmam MPLS tīkla vadības plaknē. Tas ir R. Resende, S. Rossi, A. Yamakami, L. Bonani un E. Moschim 2003. gadā publicēts darbs [87]. Autori izvirza mērķi savam algoritmam – palielināt konekciju skaitu un vienlaicīgi minimizēt pārslodzes MPLS tīklā augstas slodzes gadījumā. Faktiski tiek piedāvāts komplicēts MPLS trafika kontroles mehānisms, kas ir balstīts uz 3 moduļiem un ietver sevī arī CAC kontroli. Pirmais modulis nodrošina ceļa atrašanu ar zemāko utilizācijas līmeni, un tā darbība balstās uz *Dijkstra* ceļa meklēšanas algoritmu [118]. Otrais modulis pieņem lēmumu par to, vai akceptēt aplikācijas konekciju ar pieprasīto caurlaides joslu, vai arī piedāvā aplikācijai citu caurlaides joslu atkarībā no tā, kādu resursu daudzumu ir iespējams saņemt no tīkla. Trešais modulis ir atbildīgs par LSP iestatīšanu no trafika avota līdz trafika saņēmējam.

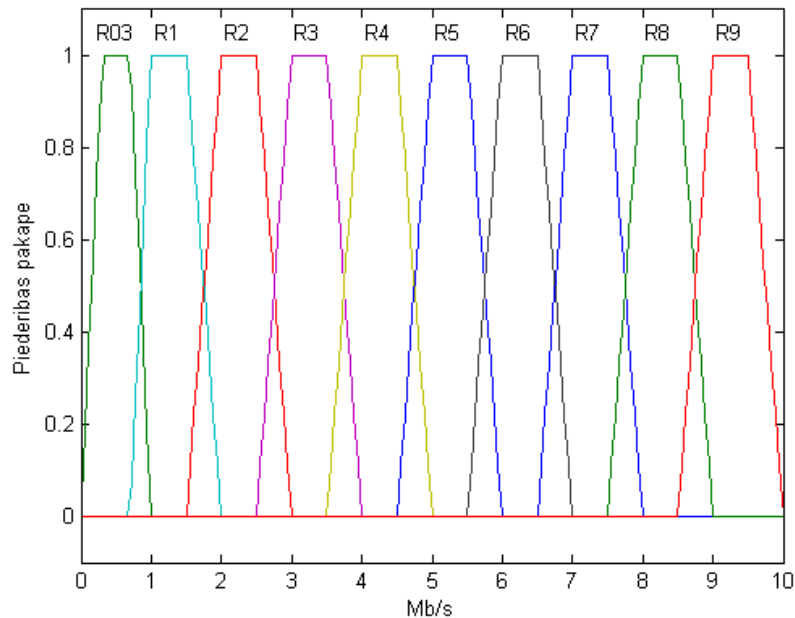
Faktiski šajā pētījumā mūs interesē otrā moduļa uz faziloģiku balstītā kontroliera uzbūve, kas pieņem lēmumu par CAC kontroli. Autori piedāvā pielietot FIS ar diviem ieejas parametriem:

- a. „*Link occupation*” jeb linkā aizņemtās caurlaides joslas platums, ko apraksta ar 9 lingvinistisko mainīgo piederības funkcijām (Att. 1.12.)



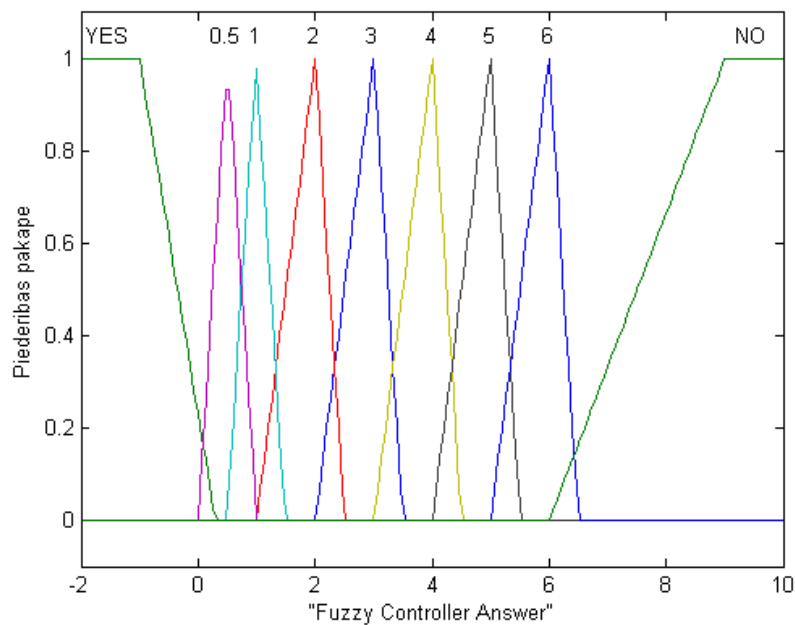
1.12. att. Lingvinistisko mainīgo piederības funkcijas ieejas parametram „*Link occupation*”

„Demand request” jeb aplikācijas pieprasītās caurlaides joslas platums, ko apraksta ar 10 lingvistisko mainīgo piederības funkcijām (Att. 1.13)



1.13. att. Lingvistisko mainīgo piederības funkcijas ieejas parametram „Demand request”

Algoritma FIS risinājumam ir viens izejas parametrs „Fuzzy Controller Answer”, ko apraksta ar 9 lingvistisko mainīgo piederības funkcijām (Att. 1.14)



1.14. att. Lingvistisko mainīgo piederības funkcijas izejas parametram „Fuzzy Controller Answer”

Autori piedāvā pielietot IF-THEN kārtulu zināšanu bāzi, kas atkarība no linkā aizņemtās caurlaides joslas platuma un aplikācijas pieprasītās caurlaides joslas platuma vai nu dod konekcijas akceptu, vai arī to noraida, jeb dod iespējamo caurlaides joslas platuma piedāvājumu aplikācijai, ko tā savukārt var pieņemt vai arī noraidīt (Tabula 1.5)

1.5. tabula

Fazi kontroliera IF-THEN kārtulu zināšanu bāze.

„Link Occupation”	„Demand Request”									
	R03	R1	R2	R3	R4	R5	R6	R7	R8	R9
OC1	YES	YES	YES	YES	YES	YES	YES	YES	YES	NO
OC2	YES	YES	YES	YES	YES	YES	YES	YES	6Mbps	NO
OC3	YES	YES	YES	YES	YES	YES	YES	6Mbps	6Mbps	NO
OC4	YES	YES	YES	YES	YES	YES	5Mbps	5Mbps	5Mbps	NO
OC5	YES	YES	YES	YES	YES	4Mbps	4Mbps	4Mbps	4Mbps	NO
OC6	YES	YES	YES	YES	3Mbps	3Mbps	3Mbps	3Mbps	3Mbps	NO
OC7	YES	YES	YES	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	2Mbps	NO
OC8	YES	YES	500Kbps	500Kbps	500Kbps	500Kbps	500Kbps	500Kbps	500Kbps	NO
OC9	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO

Algoritma darbība pārbaudei tika pielietots NS-2 diskreto notikumu tīkla simulators versijā ns-2.lh7a, pielietojot MPLS simulācijas moduli [1], kas ļāva pielietot LDP un CR-LDP protokolus LSP iestatīšanai. Autori pielietoja VBR un CBR plūsmas uz UDP bāzes.

Rezultātā, izmantojot autoru piedāvāto risinājumu, tiek sasniegts vidējo pakešu zudumu līmenis zem 5%, kas dod 35% pakešu zudumu samazinājumu salīdzinājumā ar RBWB algoritmu. Pētījuma autori uzrāda bloķēto savienojumu samazinājumu plašajā pieslēgumu intensitātes diapazonā.

Dotajā risinājumā pētījuma autori pielieto vienkāršu divu ieeju un vienas izejas Mamdani tipa fazi izvedumu sistēmu, kas ņem vērā vienīgi pieprasītās un pieejamās caurlaides joslas platumu. Netiek ņemti vērā nekādi citi QoS kritēriji, tādi kā pakešu aiztures vai pakešu zudumi linkā. Šādas sistēmas priekšrocība ir tās vienkāršums. Tomēr autoru ideja par caurlaides joslas piedāvājumu aplikācijai atbildē uz CAC lēmuma pieprasījumu prasa kādu speciāli veidotu atpakaļsaītes mehānismu. Tam būtu jānodrošina arī atbilde aplikācijas

pusē pēc CAC lēmuma saņemšanas no autoru piedāvātā CAC algoritma. Turklāt pētījuma autoru piedāvāto ieejas parametru lingvinistisko mainīgo definīcijas rezultātā dod 81 IF-THEN lielu kārtulu zināšanu bāzi. Analizējot pielietotās trapecveida piederības funkcijas, var secināt, ka dotā fazi-lēmējsistēma kļūst ļoti līdzīga klasiskās loģikas risinājumam, kur piederības pakāpe pie kādas vērtību kopas ir aprakstīta ar bināro piederības funkciju. Pēc promocijas darba autora domām, lai izmantotu visu faziloģikas potenciālu, fazi izvedumu sistēma CAC kontrolei MPLS tīklos ir jāveido, paplašinot ieejas parametru skaitu. Tas būtu nepieciešams, lai ietvertu lēmuma pieņemšanas loģikā vairākus QoS parametrus. Arī definējot mazāku skaitu lingvinistisko mainīgo, var izveidot daudz kompaktāku un intuitīvi saprotamu IF-THEN kārtulu zināšanu bāzi, ko būtu iespējams vienkārši modificēt, pielāgojot fazi izvedumu sistēmas darbību mainīgajiem tīkla apstākļiem.

## **2. JAUNS FAZI-CAC RISINĀJUMS MPLS-TE/GMPLS SAKARU SISTĒMĀM**

### **Nodaļas kopsavilkums**

Šajā nodaļā tiks sniegts pētāmā MPLS-TE tīkla atsauču modelis un fazi-CAC fazi izvedumu sistēmas atsauču modelis. Tiks definēti fazi-CAC fazi izvedumu sistēmas veiktspējas mērķi un to sasniegšanai pieļaujamo QoS parametru degradāciju ierobežojumu definīcijas un to pamatojumi. Tiks dots heuristiskās atlasēšanas pētījumu metodes apraksts, kā arī ātrai fazi-CAC lēmējsistēmas prototipēšanai nepieciešamās MPLS-TE sakaru sistēmas modeļa simulācijas vides apraksts.

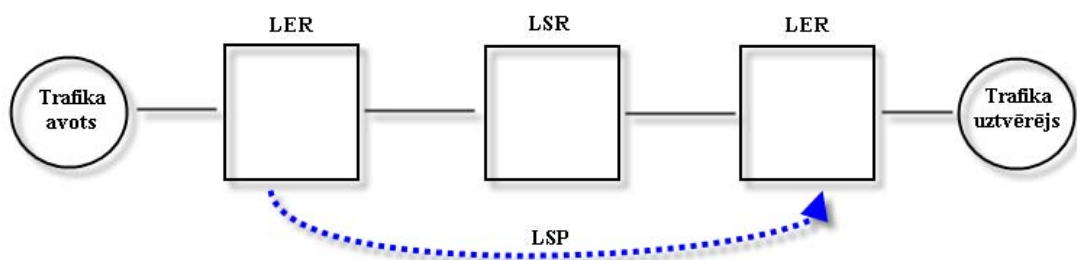
Rezultātā tiks uzdotas ieejas parametru fazifikācijas līknes un promocijas darba autora piedāvātā fazi-CAC risinājuma fazi izvedumu sistēmas ieejas un izejas parametru lingvinistisko mainīgo piederības funkciju deklarācijas. Tiks uzrādīta arī izvēlēta IF-THEN kārtulu zināšanu bāze, kā arī sniegtas fazi-CAC atbildes pēcapstrādes metodes, pašas fazi sistēmas izvedumu tipa izvēles pamatojums un iegūtās fazi-CAC lēmumu virsmas.

### **2.1. MPLS-TE sakaru sistēmas atsauču modelis**

Šeit tiks formulēts pētāmās MPLS-TE sakaru sistēmas modelis, uz kura bāzes tiks veidots eksperimentālais MPLS-TE tīkls un simulācijas vide. Tiks arī pētīts autora piedāvātais uz faziloģiku bāzēts fazi-CAC algoritma risinājums RSVP-TE protokola darbības atbalstam.

Šī promocijas darba autors pilnīgi abstrahējās no maršrutēšanas uzdevumiem, visi autoru interesējošie pētījumi var tikt veikti pēc vienkāršotas shēmas, kas nodrošina LSP iezīmes komutējamā ceļa iestatīšanas iespējas starp divām tīkla iekārtām.

LSP iestatīšana tiek ierosināta robežapgabala iezīmes komutējamajā maršrutētājā (LER), tad LSP šķērso vienu vai vairākus iezīmes komutējamus maršrutētājus (LSR) MPLS apgabala iekšienē, un izbeidzas citā LER. Rezultātā minimālais nepieciešamais maršrutētāju skaits ir 3 – divi LER un viens LSR (Att. 2.1). Tie nodrošinās divus pilnvērtīgus lēkumus maģistrālajā apgabalā un divus piekļuves tīkla apgabala lēkumus.



2.1. att. 2 LER un viena LSR slēgums MPLS-TE sakaru sistēmas eksperimentālā maketa izveidei

LSP iestatīšana pētāmajā MPLS-TE tīklā notiek ar RSVP-TE protokola palīdzību. RSVP-TE ir detalizēti aprakstīts IETF RFC3209 un papildināts RFC5151. Tas ļauj iestatīt MPLS-TE tīklā LSP tuneli, ņemot vērā tādas ierobežojošos parametrus, kā pieejamā caurlaides josla un precīzs lēkumu skaits, pieņemot CAC lēmumu. 2003. gada februārī IETF MPLS darba grupa nolēma atteikties no alternatīvā CR-LDP (RFC3212) signalizācijas protokola attīstības un koncentrējās vienīgi uz RSVP-TE protokola attīstību (RFC3469), jo 2002. gada jūnijā publicēts pārskats par pirmajām GMPLS realizācijām parādīja, ka no 22 risinājumiem 21 pielietoja RSVP-TE protokolu un tikai 1 lietoja CR-LDP [147].

Kā jau tika minēts, RSVP-TE protokols spēj pieņemt CAC lēmumu, balstoties uz pieejamās caurlaides joslas platumu. Tas nozīmē, ka tā darbība pamatā balstās uz klasisko sliekšņa CAC algoritmu un var tikt izteikta ar šādu lēmuma pieņemšanas kārtulu (2.1.):

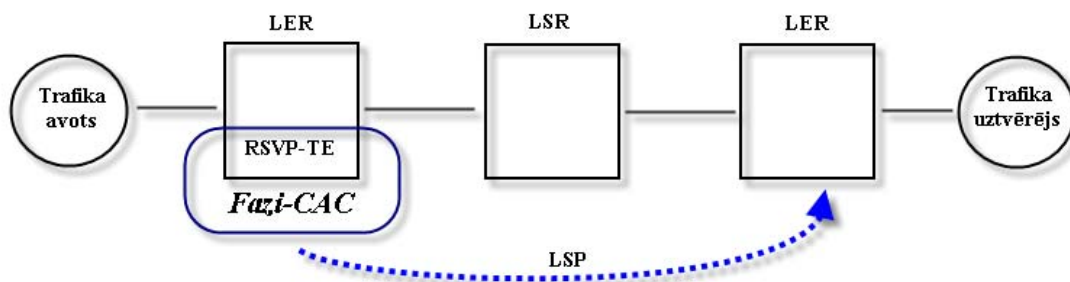
$$CAC_c = \begin{cases} Atļaut, & ja \ Bw_{rq}(t_i) \leq Bw_{aw}(t) \\ Atteikt, & ja \ Bw_{rq}(t_i) > Bw_{aw}(t) \end{cases} \quad (2.1.)$$

kur **CAC<sub>c</sub>** – klasiskā sliekšņa CAC atbilde,

**Bw<sub>rq</sub>(*t<sub>i</sub>*)** – pieprasītās caurlaides joslas platums lēmuma pieņemšanas momentā *t<sub>i</sub>* un ir izteikts [Mb/s],

un **Bw<sub>aw</sub>(*t<sub>i</sub>*)** – pieejamās joslas platums linkā lēmuma pieņemšanas momentā *t<sub>i</sub>* un ir izteikts [Mb/s].

Promocijas darba autora piedāvātais fazi-CAC algoritms vadīs RSVP-TE CAC lēmuma pieņemšanas politiku un nodrošinās vairāku QoS parametru izvērtēšanu lēmuma pieņemšanas momentā, nodrošinot selektīvo LSP iestatīšanu (Att. 2.2)



2.2. att. 2 LER un viena LSR slēgums MPLS-TE sakaru sistēmas eksperimentālā maketa izveidei

### 2.1.1. MPLS-TE sakaru sistēmas modeļa pamatelementu definīcijas

#### 1. $T_a$ – Trafika avots.

Trafiku ģenerējoša aplikācija, kas nodrošina pakešu plūsmu ar iepriekš definētu caurlaides ātrumu. Praktiskā MPLS-TE tīkla realizācijā tika pielietota *Bit-Twist* utilītprogramma. *Bit-Twist* [13] ir vienkārša, bet jaudīga, uz *libpcap* [105] bibliotēkas balstīta utilītprogramma, kas nodrošina pakešu ģenerāciju tīklā. Tā tika projektēta kā *tcpdump* [105] papildinājums. Ar *Bit-Twist* caur tīkla interfeisu ir iespējams ģenerēt tīklā *tcpdump* trases failus (.pcap failus). *Bit-Twist* arī ļauj mainīt pakešu galveņu datus, kā arī ierakstītā *tcpdump* trases faila „atskaņošanas” ātrumu. Eksperimentālajā daļā uz praktiskā MPLS-TE tīkla tika pielietots 50MB liels FTP sakaru sesijas *tcpdump* trases fails, kas tika modificēts, nomainot nosūtītāja un saņēmēja mašīnu IP adreses, un tika, pastāvīgi atkārtojot, palaists tīklā ar definēto vidējo caurlaides ātrumu. Vidējais caurlaides ātrums tika uzdots, balstoties uz  $Bw_{rq}(t)$  caurlaides joslas pieprasījumu LSP iestatīšanas laikā, rezervējot resursus ar RSVP-TE protokolu.

2.  $T_u$  – Trafika uztvērējs.

Praktiskajā MPLS-TE eksperimentālā tīkla realizācijā tas ir fiziskais sakaru sistēmas interfeiss, kas nodrošina pakešu uztveri.

3.  $LER_{in}$  – robežapgabala iezīmes komutējama maršrutētājs trafika avota pusē.

Šajā darbā tika pielietots Cisco 2800 sērijas maršrutētājs MPLS-TE konfigurācijā. LSP iestatīšanai tika pielietots RSVP-TE protokols. Ar doto maršrutētāju praktiskajā maketa realizācijā tiek nodrošināta saite, kas vada RSVP-TE protokolu, balstoties uz fazi-CAC algoritma lēmumiem.

4.  $LSR$  – iezīmes komutējama maršrutētājs.

Šī Cisco 2800 sērijas maršrutētāja loma ir nodrošināt pilnvērtīgu MPLS-TE apgabala eksistenci eksperimentālajā MPLS-TE tīkla shēmā. Tas nodrošina iezīmes nomaiņu, sūtot paketes MPLS-TE apgabalā starp  $LER_{in}$  un  $LER_{out}$ .

5.  $LER_{out}$  – robežapgabala iezīmes komutējama maršrutētājs trafika uztvērēja pusē.

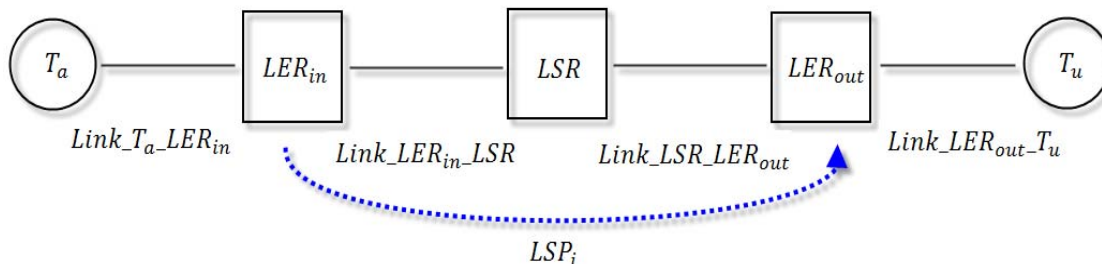
$LER_{out}$  nodrošina pēdējās iezīmes noņemšanu un paketes nosūtīšanu uz  $T_u$ .

6.  $Link_{T_u-LE_{R_{in}}}$  – 100 Mb/s links starp trafika avotu un robežapgabala iezīmes komutējamo maršrutētāju trafika avota pusē.

7.  $Link_{LE_{R_{in}}-LSR}$  – 10 Mb/s links starp robežapgabala iezīmes komutējamo maršrutētāju trafika avota pusē un iezīmes komutējamo maršrutētāju.

Faktiski šī linka QoS parametru izpēte ir eksperimentālā MPLS-TE tīkla pamatā. Dotais 10 Mb/s links nodrošina „pudeles kakla” posmu eksperimentālajā MPLS-TE maketa shēmā un dod iespēju veikt klasiskā sliekšņa CAC, un šī promocijas darba autora piedāvātā fazi-CAC algoritma salīdzinošu QoS analīzi apstākļos, kad links ir pārslogots.

8. *Link\_LSR\_LER<sub>out</sub>* – 10 Mb/s links starp iezīmes komutējamo maršrutētāju un robežapgabala iezīmes komutējamo maršrutētāju trafika uztvērēja pusē.
9. *Link\_LER<sub>out</sub>\_T<sub>u</sub>* – 100 Mb/s links starp robežapgabala iezīmes komutējamo maršrutētāju trafika uztvērēja pusē un trafika uztvērēju.



2.3. att. MPLS-TE sakaru sistēmas eksperimentālā maketa pamatelementu slēguma shēma

### 2.1.2. MPLS-TE sakaru sistēmas modeļus raksturojošo parametru definīcijas

1. Pētāmā *Link\_LER<sub>in</sub>\_LSR* linka caurlaides spēja –  $Bw_l$

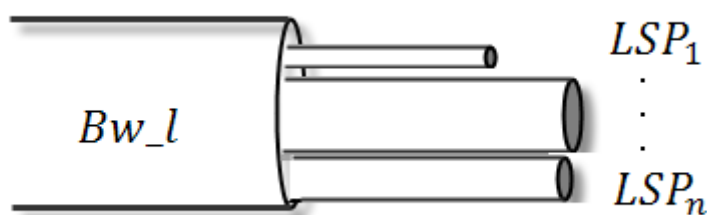
$$Bw_l = \{1, 2, 3, \dots\}, \quad (2.2.)$$

kur  $Bw_l$  ir izteikts Mb/s un ir pētāmā *Link\_LER<sub>in</sub>\_LSR* linka caurlaidspēja. Praktiskajā MPLS-TE tīkla slēgumā tā ir vienāda ar 10 Mb/s.

2. Aktīvais iezīmju komutējamais ceļš –  $LSP_i$

$$LSP_i \in R_+; LSP_i \in (0; Bw_l] \quad (2.3.)$$

MPLS-TE eksperimentālā tīkla shēmas ietvaros, kā arī simulācijas vidē  $LSP_i$  tiek definēts kā  $i$ -tajam LSP, ar RSVP-TE protokolu rezervēto resursu daudzums pie nosacījuma, ka  $LSP_i$  ir aktīvs. Rezervēto resursu daudzums ir izteikts Mb/s. Aktīvie  $LSP_i$  līdz  $LSP_n$  veido kopējo RSVP-TE rezervēto resursu daudzumu pētāmajā linkā un nosaka linka vidējo utilizācijas līmeni. (Att. 2.4)  $LSP$ , kas tika slēgti, tiek izslēgti no linka vidējā utilizācijas līmeņa kalkulācijas.



2.4. att. LSP tuneļi sakaru kanālā

**3.  $util(t_i)$  jeb pētāmā  $Link\_LER_{in\_LSR}$  linka izskaitļotā utilizācija CAC lēmuma pieņemšanas momentā  $t_i$**

$$util(t_i) = 1 - \frac{Bw_l - \sum_1^{n_{t_i}} LSP_i}{Bw_l} \quad (2.4.)$$

$$util(t_i) \in [0; 1], \quad (2.5.)$$

kur

$n_{t_i}$  – aktīvo  $LSP$  skaits CAC lēmuma pieņemšanas momentā  $t_i$ ;

$\sum_1^{n_{t_i}} LSP_i$  – CAC lēmuma pieņemšanas momentā  $t$  aktīvo  $LSP$  rezervēto resursu summa, kas izteikta Mb/s.

$util(t_i)$  nosaka pētāma  $Link\_LER_{in\_LSR}$  linka utilizāciju CAC lēmuma pieņemšanas momentā  $t$ . Aktīvo  $LSP$  rezervēto resursu summa dod pētāmā linka utilizācijas vērtību, kas

apraksta tikai RSVP-TE rezervētos resursus. Ņemot vērā trafika uzliesmojošo raksturu, kā arī šī promocijas darba autora piedāvātā fazi-CAC algoritma darbības īpatnībās, kas definē papildu resursu rezervēšanu, nomērītā linka utilizācijas var atšķirties no RSVP-TE rezervēto resursu veidotās utilizācijas.

**4.  $util_r(t_i)$  jeb pētāmā  $Link\_LER_{in\_LSR}$  linka nomērītā utilizācija laika momentā  $t_i$**

$$util_r(t_i) = 1 - \frac{Bw_l - Bw_n(t_i)}{Bw_l} \quad (2.6.)$$

$$util(t_i) \in [0; 1], \quad (2.7.)$$

kur  $Bw_n(t_i)$  – laika momentā  $t_i$  nomērītās pakešu plūsmas caurlaides ātrums MPLS-TE eksperimentālā tīkla trafika uztvērēja  $T_a$  interfeisā, kas izteikts Mb/s.

**5. Pakešu zudumi  $LER_{in}$  robežapgabala iezīmes komutējamajā maršrutētājā trafika avota pusē starp CAC lēmumu pieņemšanas momentiem  $t - 1$  un  $t$**

$$Drops_l(t_i) \in [0; N_p(t_i)], \quad (2.8.)$$

kur  $N_p(t)$  – pakešu skaits, kas tika nosūtītas laika nogrieznī no 0 līdz  $t_i$ .  $Drops_l(t_i)$  mērvienība ir [paketes].

**6. Vidējās pakešu aiztures starp  $LER_{in}$  un  $LER_{in}$  robežapgabala iezīmes komutējamajiem maršrutētājiem starp CAC lēmumu pieņemšanas momentiem  $t_i$  un  $t_{i-1}$**

$$Delay_l(t) \in [0; t_i - t_{i-1}], \quad (2.9.)$$

$Delay_l(t_i)$  mērvienība ir [ms].

7. Pakešu aizturu, starp  $LER_{in}$  un  $LER_{in}$  robežapgabala iezīmes komutējamajiem maršrutētājiem, vidējā trīce starp CAC lēmumu pieņemšanas momentiem  $t_i$  un  $t_{i-1}$

$$Jitter_l(t) \in [0; t_i - t_{i-1}] \quad (2.10.)$$

$Jitter_l(t)$  mērvienība ir [ms].

### 2.1.3. MPLS-TE eksperimentālā tīkla modeļa notikumu kopa

MPLS-TE eksperimentālā tīkla modeļa notikumu kopa sastāv no diviem pamata notikumiem, kas apraksta LSP tuneļa iestatīšanu un LSP tuneļa slēgšanu.

1. **LSP\_setup** – tiek iestatīts jauns  $LSP_i$  tunelis starp  $LER_{in}$  un  $LER_{out}$ , ar RSVP-TE protokola palīdzību rezervējot resursus linkos  $Link_{LER_{in}, LSR}$  un  $Link_{LSR, LER_{out}}$ .

Pozitīvā CAC lēmuma gadījumā, LSP\_setup notikums nozīmē RSVP-TE resursu rezervēšanu.

2. **LSP\_drop** – tiek slēgts esošais  $LSP_i$  tunelis starp  $LER_{in}$  un  $LER_{out}$ , atbrīvojot no linkiem  $Link_{LER_{in}, LSR}$  un  $Link_{LSR, LER_{out}}$  ar RSVP-TE protokola palīdzību rezervējos resursus.

LSP\_drop notikums nozīmē ik pēc n-tiem CAC lēmumiem esošā  $LSP_i$  tuneļa starp  $LER_{in}$  un  $LER_{out}$  rezervēto resursu atbrīvošanu un  $LSP_i$  slēgšanu.

## **2.2. Fazi-CAC atsauču modelis**

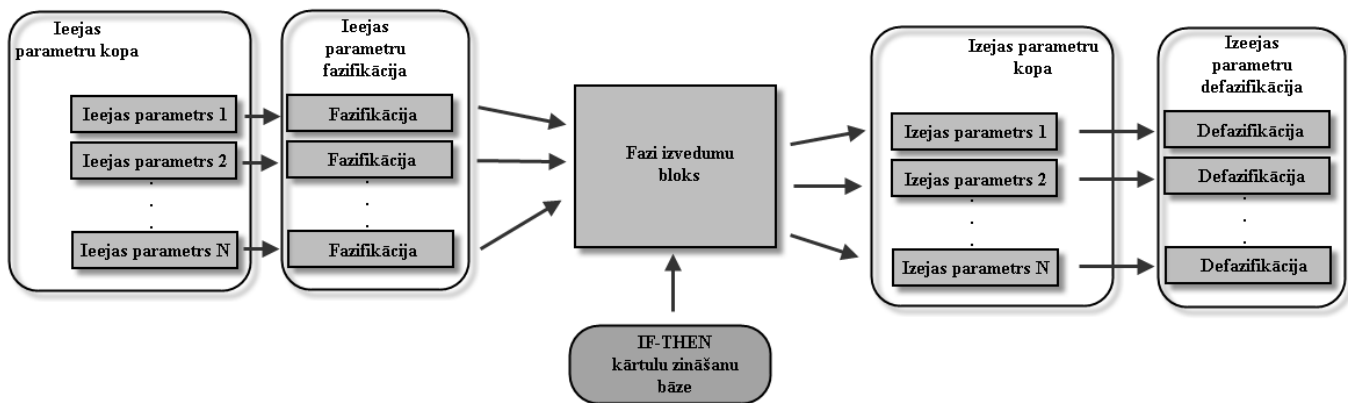
Šeit tiks formulēts fazi-CAC atsauču modelis, uz kura bāzes tiks veidota fazi-CAC risinājuma fazi izvedumu sistēma, kas tiks pielietota eksperimentālajā MPLS-TE tīklā un simulācijas vidē, kur savukārt tiks pētīts autora piedāvātais uz faziloģiku bāzēts CAC risinājums.

Fazi-CAC atsauču modeļa pamatā ir standarta fazi izvedumu sistēmas shēma, kas spēj pārnest ieejas parametru veidoto momentāno ainu uz izejas parametra vērtībām, balstoties uz IF-THEN nosacījumu kārtulām. Fazi-CAC atsauču modelis kalpo par pamatu konkrētai fazi izvedumu sistēmas realizācijai, kas tiks pielietota salīdzinošajā analīzē ar klasisko sliekšņa CAC algoritmu.

### **2.2.1. Fazi-CAC modeļa pamatelementi**

Šī pētījuma autora piedāvātā fazi-CAC algoritma darbība balstās uz fazi izvedumu sistēmu, kas ļauj pārnest ieejas parametru vērtības uz izvada parametra vērtībām, pielietojot zināšanu bāzi IF-THEN kārtulu veidā, kā arī fazi loģiskās operācijas. Tātad šī promocijas darba piedāvātā fazi-CAC mehānisma atsauču modelis satur 6 pamatelementus (Att. 2.5)

1. Ieejas parametru kopa.
2. Ieejas parametru fazifikācijas modulis.
3. Fazi izvedumu bloks.
4. IF-THEN kārtulu zināšanu bāze.
5. Izejas parametru mainīgo kopa.
6. Izejas parametru defazifikācijas modulis.



2.5. att. Fazi-CAC atsauču modeļa shēma

Fazi-CAC mehānisma pamatelementu sīkāks izklāsts ir dots turpmākajās darba nodaļās.

### 2.2.2. Fazi-CAC ieejas parametru kopa

Šī promocijas darba autora piedāvātais fazi-CAC risinājums pielieto 3 ieejas parametrus. Parametru vērtības, kas tiek turpmāk definētas, ir uzrādītas kā fazi-CAC ieejas parametru vērtības pēc fazifikācijas procedūras. Fazifikācijas līknes tiks definētas nodaļā 2.7.5 pēc fazi-CAC ierobežojošo parametru vērtību definīcijām. Ieejas parametru kopas, kā arī lingvinistisko mainīgo piederības funkciju definīciju pamatojums tiks dots nodaļā 2.7.6.

1. Pieprasītās un pieejamās caurplūdes joslas platumu attiecība CAC lēmuma pieņemšanas momentā  $t_i$ , ko apzīmēsim ar  $Bw_r(t_i)$ ,

$$Bw_r(t_i) = \frac{Bw_{aw}(t_i)}{Bw_{rq}(t_i)}, \quad (2.11.)$$

kur,  $Bw_{aw}(t_i)$  – pieejamais joslas platumu linkā CAC lēmuma pieņemšanas momentā un  $Bw_{rw}(t_i)$  – pieprasītās caurlaides joslas platumu CAC lēmuma pieņemšanas momentā, kas ir izteikti Mb/s.

$$Bw_{aw}(t_i) = Bw_l - Bw_{rw}(t_i) , \quad (2.12.)$$

kur savukārt

$$Bw_l = \{1, 2, 3, \dots\} , \quad (2.13.)$$

un apzīmē pētāmā linka caurlaides spēju izteiktu Mb/s.

Savukārt

$$Bw_{rw}(t_i) \in [0, Bw_l] , \quad (2.14.)$$

$$Bw_{rw}(t_i) \in R_+ / 0 \leq Bw_{rw}(t_i) \leq Bw_l , \quad (2.15.)$$

$$Bw_{rw}(t_i) = \sum_1^{n_{t_i}} LSP_i , \quad (2.16.)$$

kur  $n_{t_i}$  – aktīvo *LSP* skaits CAC lēmuma pieņemšanas momentā  $t_i$ ;

$Bw_{rw}(t_i)$  apzīmē RSVP rezervētos resursus linkos *Link\_LER<sub>in</sub>\_LSR* un *Link\_LSR\_LER<sub>out</sub>* CAC lēmuma pieņemšanas momentā  $t_i$ . Aktīvie  $LSP_i$  līdz  $LSP_n$  veido kopējo RSVP-TE rezervēto resursu daudzumu pētāmajos linkos un ir izteikti [Mb/s].

Savukārt

$$Bw_{rq}(t_i) \in [0, Bw_l] , \quad (2.17.)$$

$$Bw_{rq}(t_i) \in R_+ / 0 \leq Bw_{rq}(t_i) \leq Bw_l , \quad (2.18.)$$

un apzīmē pieprasītās joslas platumu CAC lēmuma pieņemšanas momentā  $t_i$ , ko nosaka galaiekārtas aplikācija, un ir izteikts [Mb/s].

2. QoS klase, jeb līmenis, ko pieprasa aplikācija CAC lēmuma pieņemšanas momentā  $t_i$ . Apzīmē ar  $QoS_c(t_i)$ , kur

$$QoS_c(t_i) \in [0,10], \quad (2.19.)$$

$$QoS_c(t_i) \in R_+ / 0 \leq QoS_c \leq 10, \quad (2.20)$$

$QoS_c(t_i)$  vērtības tiek ģenerētas ar fBm ģeneratoru, kas ir aprakstīts turpmāk dotajā literatūras avotā [8].

3. Pakešu aiztures starp  $LER_{in}$  un  $LER_{out}$  – momentānās pakešu aizturu vērtības CAC lēmuma pieņemšanas momentā, starp  $LER_{in}$  un  $LER_{out}$  robežapgabala iezīmes komutējamajiem maršrutētājiem, kuras apzīmēsim ar  $Delay_l(t_i)$ , kur

$$Delay_l(t_i) \in [0,10], \quad (2.21)$$

$Delay_l(t_i)$  vērtību apgabals tiek deklarēts intervālā [0,10] pēc fazifikācijas procedūras.

### 2.2.3. Fazi-CAC modeļa izejas parametru kopa

Autora piedāvātais fazi-CAC risinājums pielieto vienu izejas parametru, kas nosaka fazi-CAC lēmumu.

**Fazi-CAC atbilde** – fazi izvedumu bloka atgrieztā vērtība pēc defazifikācijas, ko apzīmēsim ar  $CAC_f(t_i)$ ,

$$CAC_f(t_i) \in [0,10], \quad (2.22.)$$

$$CAC_f(t_i) \in \mathbb{R}_+ / 0 \leq CAC_f(t_i) \leq 10. \quad (2.23.)$$

### 2.3. Fazi-CAC modeļa ieejas parametru fazifikācija

#### 1. $Bw_r(t_i)$ fazifikācija.

$Bw_r(t_i)$  netiek papildus fazificēts, bet gan ievadīts fazi izvedumu sistēmā tiešā veidā.

#### 2. $QoS_c(t_i)$ fazifikācija.

$QoS_c(t_i)$  vērtības tiek deklarētas intervālā  $[0,10]$ , ņemot vērā to turpmāko ērto izmantošanu bez nepieciešamības papildus pielietot fazifikācijas procedūru.  $QoS_c(t_i)$  vērtības tiek ievadītas fazi izvedumu sistēmā tiešā veidā.

#### 3. $Delay_l(t_i)$ fazifikācija.

Praktiskajā MPLS-TE tīkla realizācijā nomērītās aizturu vērtības tiks fazificētas, iekļaujot ieejas parametrus intervālā  $[0,10]$ . Fazifikācijas līkne ir uzrādīta nodaļā 2.7.5.

### 2.4. Fazi-CAC modeļa IF-THEN kārtulu zināšanu bāze

Fazi-CAC modeļa zināšanu bāze tiek definēta kā IF-THEN kārtulu kopa, kur katra atsevišķa kārtula ir izteikta šādā veidā:

**IF-** (*Nosacījums*) **AND/OR-** (*Nosacījums*<sub>2</sub>) **AND/OR-** (*Nosacījums*<sub>3</sub>)

..... **AND/OR-** (*Nosacījums<sub>N</sub>*)

**THEN** - (*Sekas*),

kur *Nosacījums<sub>i</sub>* – apgalvojums par kāda noteikta fazi sistēmas ieejas parametra vērtības piederību pie šī ieejas parametra konkrētā lingvinistiskā mainīgā vērtību kopas. Piederības pakāpe tiek noteikta ar dotā lingvinistiskā mainīgā piederības funkciju. Savukārt *Sekas* nosaka fazi sistēmas izejas parametra vērtības piederību pie šī izejas parametra konkrētā lingvinistiskā mainīgā vērtību kopas, kas arī tiek aprakstīta ar kādu konkrētu piederības funkciju.

Piemēram, pieņemsim, ka fazi lēmējsistēma pielieto divus ieejas parametrus – *Delays* un *QoS* un vienu izejas parametru – *CAC\_decision*. Pieņemsim arī, ka ieejas parametri *Delays* un *QoS*, katrs tiek definēti ar 3 lingvinistiskiem mainīgiem, un *CAC\_decision* pielieto 2 lingvinistiskos mainīgos:

*Delays* – {*mazas*, *vidējas*, *lielas*}

*QoS* – {*real\_time*, *vidējais*, *best\_effort*}

*CAC\_decision* – {*at\aut*, *aizliegt*}

Tad šādas fazi lēmējsistēmas iespējamās IF-THEN kārtulas izskatītos šādi:

1. **IF-** (*Delays ir "mazas"*) **AND-** (*QoS ir "real\_time"*)  
**THEN-** (*CAC\_decision ir "at\aut"*)
2. **IF-** (*Delays ir "lielas"*) **AND-** (*QoS ir "real\_time"*)  
**THEN-** (*CAC\_decision ir aizliegt*)
3. **IF-** (*Delays ir "lielas"*) **AND-** (*QoS ir "best\_effort"*)

**THEN- (*CAC\_decision* ir "atļaut")**

**Utt.**

Šādi definēti nosacījumi veido IF-THEN kārtulu zināšanu bāzi, kas tiks pielietota fazi-CAC fazi izvedumu sistēmas uzbūvē. IF-THEN kārtulu zināšanu bāzes izvēles pamatojums un tās definīcija ir dota nodaļā 2.7.7.

## **2.5. Fazi-CAC modeļa stāvokļu kopa**

Šī promocijas darba autors piedāvā pielietot fazi-CAC algoritmam 3 stāvokļus, kas nosaka CAC algoritma reakcijas uz ieejas parametru vērtību kopu.

- a. Atļaut LSP iestatīšanu.
- b. Atteikt LSP iestatīšanu.
- c. Atļaut LSP iestatīšanu ar papildu resursu rezervēšanu.

Klasiskajā CAC uzdevuma vadībā tiek operēts ar diviem CAC lēmumiem, kas definē vai nu pozitīvo atbildi, vai arī negatīvo. Šī promocijas darba autors piedāvā modificēt CAC kontroles uzdevumu un sevis piedāvātajā fazi-CAC algoritma realizācijā operē ar 3 iepriekš minētajiem fazi-CAC algoritma stāvokļiem.

Šādas izvēles pamatā ir nepieciešamība nodrošināt mūsdienu trafika plūsmas ar pieprasītām QoS parametru garantijām, arī gadījumā, kad plūsmu raksturs ir uzliesmojošs. Uzliesmojumu pīķu brīžos datu sūtīšanas intensitāte pārsniedz esošo tīklu resursu sliekšni, rezultāta novedot pie QoS parametru degradācijas, kas nav pieļaujami.

Iztēlosimies situāciju, kad maršrutētājs saņem jaunu LSP tuneļa izveides pieteikumu no aplikācijas ar prasību rezervēt noteiktu resursu daudzumu linkā  $R_1$  [Mb/s]. Ja aplikācijas QoS prasības ir zemas un links satur pietiekamu daudzumu brīvo resursu, tad pozitīvā CAC atbilde liek RSVP-TE rezervēt pieprasīto caurlaides joslas platumu  $R_1$  noteiktajā tīkla apgabalā.

Tagad modelēsim situāciju, kad aplikācijas QoS prasības ir ļoti augstas, tīkla QoS parametru vērtības ir daudz augstākas nekā aplikācijas pieprasītais QoS līmenis. Turklāt resursu daudzums linkā ir daudzkārt lielāks nekā aplikācijas pieprasītā caurlaides josla. Šādos gadījumos šī promocijas darba autors piedāvā fazi-CAC izvedumu sistēmas IF-THEN kārtulu zināšanu bāzē paredzēt noteiktās „sekas”, kas virzītu CAC atbildi: „Atļaut LSP iestatīšanu ar papildu resursu rezervēšanu.” Šajā gadījumā aplikācijas resursu prasība tiktu apmierināta ar RSVP-TE rezervējot lielāku resursu daudzumu, nekā ir prasīts:  $R_1 \cdot k$ , kur  $k$  ir papildu rezervēšanas koeficients. Tas nosaka, cik reizes vairāk resursu tiks rezervēts pie dotās fazi-CAC atbildes. Šāds papildus resursu rezervēšanas notikums līdzinās specifiskajam „drošības spilvenam”. Tas ļaus saglabāt QoS parametrus no degradācijas arī uzliesmojošā trafika gadījumā, turklāt šāda veida pasargāšana būtu tieši proporcionāla augstā līmeņa QoS plūsmām linkā. Gadījumā, ja LSP tuneļu pieprasījumiem ir zemas QoS prasības, arī RSVP-TE rezervēto resursu daudzums sakrītīs ar aplikāciju pieprasīto resursu daudzumu. Papildus rezervēšanas koeficienta  $k$  vērtība tiks definēta nodaļā 2.7.11.

Tāpat fazi-CAC izvedumu sistēmas modeļa stāvokli nosaka fazi-CAC izejas parametra  $CAC_f(t_i)$  vērtība, ko savukārt izrēķina fazi-CAC algoritma fazi izvedumu sistēma.

Tā kā  $CAC_f(t_i)$  vērtība pēc defazifikācijas ir robežās no 0 līdz 10, ir jāizvēlas kādas noteiktas sliekšņu vērtības „ $s_1$ ” un „ $s_2$ ”, kas nodrošinātu  $CAC_f(t_i)$  pārbaudi attiecībā uz to vērtību. Tāpat fazi-CAC atbildes  $CAC_f(t_i)$  vērtības pēcapstrāde nodrošinās tās pārbaudi attiecībā uz sliekšņu vērtību, kas atšķirs LSP slēguma atteikumu, atļauju un atļauju ar papildu resursu rezervēšanu. Fazi-CAC atbildes  $CAC_f(t_i)$  pēcapstrādes sliekšņu „ $s_1$ ” un „ $s_2$ ” vērtības ir definētas nodaļā 2.7.10.

## 2.6. Pielietotā fazi-CAC pseidokods

Ņemot vērā visas iepriekš definētās ieejas parametru vērtības un fazi-CAC stāvokļu kopu, šī promocijas darba autora pielietotā fazi-CAC darbība var tikt attēlota ar tālāk dotu pseidokodu:

**START fazi-CAC**

FOR  $i = 0$  TO  $n$  STEP 1

    READ  $Bw_{rq}(t_i)$

```

READ  $Bw_{aw}(t_i)$ 
READ  $Delay_l(t_i)$ 
READ  $QoS_l(t_i)$ 
CALCULATE  $Bw_r(t_i)$ 
  CALCULATE  $CAC_f(t_i)$ 
  IF  $CAC_f(t_i) < 5 - s_1$ 
    THEN  $f_{CAC} = \text{„Atteikt”}$  AND  $LSP_i = 0$ ;
  ESLE IF  $CAC_f(t_i) > 5 - s_1$  AND  $CAC_f(t_i) < 5 + s_2$ 
    THEN  $f_{CAC} = \text{„Atļaut”}$  AND  $LSP_i = Bw_{rq}(t_i)$ 
  ESLE IF  $CAC_f(t_i) > 5 + s_2$ 
    THEN  $f_{CAC} = \text{„Atļaut ar papildu rezervēšanu”}$  &&  $LSP_i = Bw_{rq}(t_i) \cdot k$ 
  END IF
NEXT  $t_i$ 
AND FOR
END fazi-CAC

```

Fazi-CAC algoritms balstās uz konkrētās fazi izvedumu sistēmas, kuras parametru izvēle un pamatojums tiks sniegti turpmāk dotajās promocijas darba nodaļās.

## 2.7. Fazi-CAC lēmējsistēmas uzbūves heuristiskā izvēle

Šī promocijas darba autora piedāvātais fazi-CAC algoritms RSVP-TE protokola vadībai MPLS-TE sakaru sistēmas realizācijā balstās uz fazi-CAC fazi izvedumu sistēmas modeli, kura pamata elementi, kā arī ieejas un izejas parametru kopas tika definētas nodaļā 2.2.

Iepriekš minētajā nodaļā netika dota  $Delay_l(t_i)$  ieejas mainīgā fazifikācijas līknes deklarācija, izejas parametru defazifikācijas paņēmiena definīcija, kā arī ieejas parametru lingvinistisko mainīgo definīcijas un to izvēles pamatojumi. Netika definēta arī autora piedāvātā fazi-CAC risinājuma IF-THEN kārtulu zināšanu bāze. Šīs darbības var tikt izpildītas pēc fazi-CAC veiktspējas mērķu definēšanas, nosakot konkrētās QoS parametru

degradācijas robežas. Šinī nodaļā autors definēs iepriekš minētos mērķus, kā arī definēs vēlamās fazi izvedumu sistēmas parametru vērtības.

Pirmām kārtām ir jādefinē fazi-CAC veikspējas mērķis, kas ir jāsasniedz, meklējot optimālo vai arī tam tuvu risinājumu, izvēloties fazi-CAC lēmējsistēmas veidu (*Mamdani, Takagi-Sugeno*), fazi-CAC lēmējsistēmas modelim nepieciešamos ieejas parametru lingvinistisko mainīgo piederības funkciju deklarācijas, IF-THEN kārtulu zināšanu bāzi, kā arī iepriekš minēto parametru deklarācijas.

### 2.7.1. Fazi-CAC veikspējas mērķu definīcija

Turpmāk tiek definēti vēlamie mērķi fazi-CAC efektīvai darbībai, ar nolūku izvēlēties optimālo vai arī tam tuvu risinājumu fazi-CAC fazi izvedumu sistēmas izveidē. Promocijas darba autors definē 3 pamata mērķus, uz kuru sasniegšanu ir jātiecas, meklējot vēlamo fazi-CAC lēmējsistēmas uzbūvi ar nolūku nodrošināt tās optimālo veikspēju.

1. **Nodrošināt maksimālo vidējo linka utilizāciju pie uzdotiem QoS parametru degradācijas ierobežojumiem.**

$$\forall t_i : \max \frac{\sum_1^n util(t_i)}{n} \quad (2.24.)$$

$$Delay\_l(t_i) / 0 \leq Delay\_l(t_i) \leq Delay\_l(t_i)_{MAX}$$

$$Drops\_l(t_i) / 0 \leq Drops\_l(t_i) \leq Drops\_l(t_i)_{MAX}$$

$$Jitter\_l(t_i) / 0 \leq Jitter\_l(t_i) \leq Jitter\_l(t_i)_{MAX}$$

kur

- $n$  – ir eksperimenta laikā veikto CAC lēmumu skaits,
- $util(t_i) \in [0,1]$  un ir pētāmā linka utilizācija, izteikta kā RSVP-TE rezervētie resursi CAC lēmuma pieņemšanas momentā  $t_i$ ,
- $Delay\_l(t_i)$  ir pētāmā testa tīklā nomērītās pakešu aiztures CAC lēmuma pieņemšanas momentā,
- $Delay\_l(t_i)_{MAX}$  ir maksimālās pieļaujamās pakešu aiztures pētāmajā testa tīklā CAC lēmuma pieņemšanas momentā,

- $Jitter\_l(t_i)$  ir ir pētāmā testa tīklā nomērītās trīces vērtības CAC lēmuma pieņemšanas momentā,
- $Jitter\_l(t_i)_{MAX}$  ir maksimālās pieļaujamās trīces vērtības pētāmajā testa tīklā CAC lēmuma pieņemšanas momentā,
- $Drops\_l(t_i)$  ir pētāmā testa tīklā nomērītie pakešu zudumi CAC lēmuma pieņemšanas momentā,
- $Drops\_l(t_i)_{MAX}$  ir maksimāli pieļaujami pakešu zudumi pētāmajā testa tīklā CAC lēmuma pieņemšanas momentā.

Šī mērķa sasniegšanai ir jāapzinās visi iespējami MPLS-TE linku aprakstošie stāvokļi, kas būtībā ir  $util(t_i)$ ,  $Delay\_l(t_i)$ ,  $Jitter\_l(t_i)$  un  $Drops\_l(t_i)$  vērtību kombinācijas (Tabula 2.1)

2.1. tabula

MPLS-TE linku aprakstoši stāvokļi vienam mērījumam

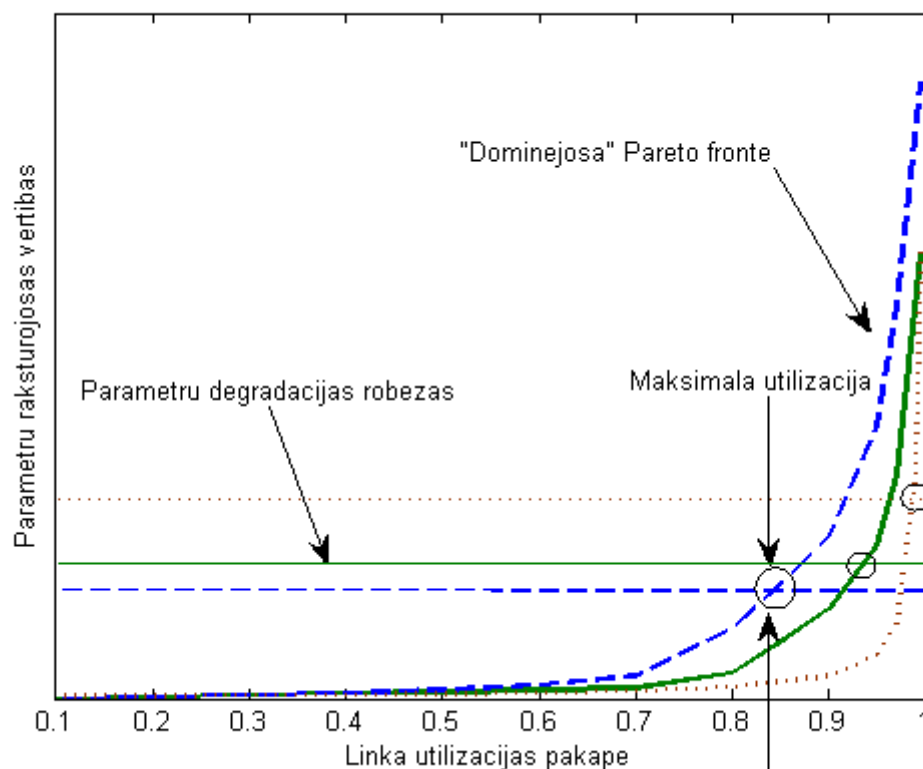
$t_i$	$util(t_i)$	$Delay\_l(t_i)$	$Jitter\_l(t_i)$	$Drops\_l(t_i)$
1	U1	Z1	J1	D1
2	U2	Z2	J2	D2
...	...	...	...	...
n	Un	Zn	Jn	Dn

Veicot pietiekami lielu skaitu linka noslodzes mērījumu, varam izskaitļot vidējās  $Delay\_l(t_i)$ ,  $Jitter\_l(t_i)$  un  $Drops\_l(t_i)$  vērtības konkrētajam testa tīklam.

Rezultātā iegūstam savdabīgās risinājumu *Pareto* frontes (Att. 2.6) parametru kombinācijām:

- $util(t_i)$  un  $Delay\_l(t_i)$ ;
- $util(t_i)$  un  $Jitter\_l(t_i)$ ;
- $util(t_i)$  un  $Drops\_l(t_i)$ .

Mērījumu rezultātā tiks definēta „dominējošā” *Pareto* fronte, kas noteiks konkrēto risinājuma punktu, definējot maksimālo linka utilizācijas pakāpi (RSVP-TE rezervēto resursu daudzumu), kas ir pieļaujama, lai tīkla darbības laikā QoS parametru vērtības robežas paliek definēto vērtību apgabalā. (Att. 2.6)



2.6. att. Iespējamo risinājumu Pareto frontes

Jāpiezīmē, ka šāda tipa risinājums būs unikāls katrai konkrētajai situācijai un to ietekmēs trafika uzliesmojošā daba. Piemēram, stipri uzliesmojošā fona trafika klātbūtnē,  $(util(t_i) - Delay_l(t_i))$  „dominējošās” risinājumu Pareto frontes vietu var ieņemt  $(util(t_i) - Drops_l(t_i))$  vai  $(util(t_i) - Jitter_l(t_i))$ .

2. Nodrošināt selektīvu trafika plūsmu izvēli LSP tuneļu izveidei, ar nolūku palielināt augstas QoS klases plūsmu pieslēgumu skaitu pie uzdotiem QoS parametru degradācijas ierobežojumiem.

Šī mērķa pamatā ir fazi-CAC selektīvā LSP iestatīšanas daba, kas spēj nodrošināt lielāku skaitu jaunu LSP pieslēgumu, operējot ar mazākiem caurlaides joslu pieprasījumiem, saglabājot QoS parametru vērtības pieļaujamās degradācijas robežās.

### **3. Mazināt tādu plūsmu pieslēgumus, kas neatbilst linka QoS nodrošinājuma iespējām LSP iestatīšanas brīdī.**

Šis mērķis ir tieši pakārtots otrajam mērķim un izriet no fazi-CAC spējas selektīvi pieslēgt jaunas LSP plūsmas, rezervējot resursus linkā ar RSVP-TE protokola palīdzību. Fazi-CAC fazi izvedumu sistēmas zināšanu bāze IF-THEN kārtulu veidā kopā ar FIS ieejas un izejas parametru struktūru nodrošina fazi-CAC spēju nepieļaut tādu LSP pieslēgumu akceptēšanu, kas neatbilst FIS darbības loģikā definētajām QoS parametru vērtībām.

#### **2.7.2. QoS parametru degradācijas ierobežojumu definīcijas**

Šajā nodaļā ir uzrādīts pamatojums izvēlēto QoS parametru degradācijas ierobežojumu izvēlei. Šīm nolūkam par pamatu tiks apskatītas QoS jutīgāko mūsdienu multimediju aplikāciju prasības un ierobežojumi, ka arī dažu lielāko maģistrālo tīklu SLA definētas QoS parametru vērtības.

QoS „prasīgo” aplikāciju saime pārsvarā sastāv no tiešsaites aplikācijām, kas nodrošina multimediju pakalpojumu piegādi gala lietotājiem. Balss un video datu pārraide mūsdienās kļūst aizvien populārākas un, balstoties uz *Cisco Visual Networking Index* sniegtajām prognozēm, tieši interneta video un interneta video uz TV, ka arī video zvani un tiešsaites spēles veidos tuvāko 4 gadu trafika pieauguma stūrakmeni. Šīs aplikācijas nodrošinās ikgadējo trafika apjomu pieaugumu pat 107% apjomā video trafika gadījumā un 48% – video zvanu gadījumā.

Sākumā multimediju aplikācijas varam nosacīti iedalīt divās daļās:

- aplikācijas, kas pielieto TCP protokolu trafika transportam;
- aplikācijas, kas pielieto UDP paketes.

Vairākums straumēšanas video internetā pielieto RTSP protokolu, nodrošinot „gals-gals” datu transportu ar TCP palīdzību. Šajā gadījumā praktiski visi pakešu zudumi tiks eliminēti, pateicoties TCP protokola darbībai, toties rezultātā parādīsies ievērojami trīces

uzliesmojumi, kas degradēs video un skaņas kvalitāti, ja trīces buferi gala lietotāja ierīcē nespēs tos kompensēt.

Apskatīsim kaut vai, piemēram, uz TCP balstītu IPTV servisu. Video signāla kvalitāte ir atkarīga no IP pakešu plūsmas, kas pienāk konstantā ātrumā un nodrošina datu pārraidi bez pakešu zudumiem. Dekoderis klienta pusē tiek sasinchronizēts ar koderi IPTV datu centrā, un jebkādas pakešu starpatnākšanas laiku svārstības jeb trīce var slikti iespaidot video attēla kvalitāti, izsaucot mirgošanu uz ekrāna un nepārtrauktā video attēla pārtraukumus. Tam par iemeslu ir pakešu atnākšanas laiku nevienlīdzība, kad novēloti atnākušās paketes jau nepaspēj iestāties pakešu rindā un noved pie attēla degradācijas. Savukārt tās paketes, kas atnāk ātrāk, pārpilda buferi, un radušies pakešu zudumi dod to pašu negatīvo rezultātu.

Šo efektu cenšas minimizēt, pielietojot lielas ietilpības atmiņas buferus IPTV lietotāju gala iekārtās. Tomēr ir jāatceras, ka, kaut arī, pielietojot lielākos, buferus ir iespējams samazināt trīces ietekmi uz video signāla kvalitāti, tie ienes ļoti ievērojamas papildu aiztures.

Gadījumā, ja IPTV pielieto UDP datagrammas, tas kļūst īpaši jutīgs attiecībā uz aizturēm, ko izraisa pārslogotie serveri, maršrutēšanas process un tīkla pārslodzes, laikā, kad datu paketes šķērso tīkla segmentus. Jebkuri pakešu zudumi tieši iespaido video attēla un skaņas kvalitāti.

Vajag izvērtēt vēl vienu svarīgu multimediju aplikāciju sadalījuma veidu:

- tiešsaites multimediju aplikācijas, kas nodrošina reālā laika divvirzienu komunikāciju;
- multimediju aplikācijas, kas nodrošina vienvirzienu video un audio datu pārraidi.

Pirmajā gadījumā tās ir tiešsaites divvirzienu komunikāciju aplikācijas, kas nodrošina VoIP un video konferenču pakalpojumus. Otrajā gadījumā tās ir video un audio straumēšanas aplikācijas, kas pielieto multimediju satura serverus un datu pārraides vidi pie gala lietotāja.

Tieši tiešsaites divvirzienu komunikācijas aplikācijas ir visjutīgākās attiecībā pret QoS parametru degradāciju [57], [95].

VoIP ir ļoti jutīgs attiecībā uz pakešu zudumiem. Pielietojot vismazāk kompresējošo G.711 kodeku, pat 1% pakešu zudumi var ievērojami pasliktināt VoIP sarunas kvalitāti, bet citi kodeki ar lielāko kompresijas pakāpi ir vēl jutīgāki attiecībā uz pakešu zudumiem. Pēc *Cisco* sniegtiem datiem, populārais G.729 kodeks, lai nodrošinātu sakaru sesiju bez jūtamiem runas signāla traucējumiem, prasa pakešu zudumu ierobežojumus daudz mazākus par 1% [40].

Pieminēsim arī daudzlietotāju tiešsaites spēles, kuras mūsdienās visbiežāk lieto UDP datagrammas. UDP atļauj sūtīt lielus daudzumus sinhronizācijas datu, kas ir īpaši jutīgi attiecībā uz pakešu aizturēm. TCP gadījumā pakešu pārsūtīšanas ienestās aiztures var dramatiski ietekmēt daudzlietotāju tiešsaites spēles veiktspēju tīklā, toties pakešu zudumi nav tik kritiski, jo daudzi dati tiek atkārtoti pārsūtīti pēc neilga laika perioda. Daudzlietotāju tiešsaites spēļu gadījumā neatmaksājas pieprasīt pārsūtīt zudušās paketes, jo daudz efektīvāk ir vienkārši sagaidīt nākamos datus un atjaunot spēlei nepieciešamos datus izmantojot pēdējās saņemtās paketes. TCP tiek pielietots reti [23] un visbiežāk tikai tiešsaites teksta veida komunikācijas nodrošināšanai starp lietotājiem. Tātad daudzlietotāju tiešsaites spēļu gadījumā lietotāju aplikācijas ir daudz jutīgākas attiecībā uz pakešu aizturēm nekā uz pakešu zudumiem.

Tagad, noskaidrojot jutīgo multimediju aplikāciju QoS prasības, apskatīsim dažu lielo maģistrālo tīklu operatoru definētos servisa līmeņa līgumus jeb SLA. Turpmāk ir doti dati par diviem Eiropas maģistrālā tīkla SLA, ko pārstāv NNT Communication – NNT Europe [77] un Sprint Network [97]. Pēdējie piedāvā atšķirīgus SLA MPLS tīkliem. Turpmāk ir publicēti dati par ASV maģistrālo tīklu SLA, ko pārstāv Internap [46], Qest [82] un Verio [110].

## **1. NNT Communication – NNT Europe SLA**

- Pakešu aiztures Eiropas tīklā – 35ms vai mazāk.
- Pakešu zudumi Eiropas maģistrālajā tīklā – 0,3% vai mazāk.
- Vidējā trīce Eiropas maģistrālajā tīklā – 0,5ms vai mazāk, nepārsniedzot 10ms vairāk kā 0,1% mēneša kalendārajā laikā.

## **2. Sprint Network SLA**

SprintLink SLA.

- Pakešu aiztures Eiropas maģistrālajā tīklā – 45ms vai mazāk.
- Pakešu zudumi Eiropas maģistrālajā tīklā – 0,3% vai mazāk.
- Vidējā trīce Eiropas maģistrālajā tīklā – 2 ms vai mazāk.

Global MPLS SLA.

- Pakešu aiztures Eiropas maģistrālajā tīklā – 45ms vai mazāk.
- Pakešu zudumi Eiropas maģistrālajā tīklā – 0,1% vai mazāk.
- Vidējā trīce Eiropas maģistrālajā tīklā – 2 ms vai mazāk.

### 3. Internap SLA

- Pakešu aiztures ASV maģistrālajā tīklā – 45ms vai mazāk.
- Pakešu zudumi ASV maģistrālajā tīklā – 0,3% vai mazāk.
- Vidēja trīce ASV maģistrālajā tīklā – 0,5 ms vai mazāk.

### 4. Qwest SLA

- Pakešu aiztures ASV maģistrālajā tīklā – 42ms vai mazāk.
- Pakešu zudumi ASV maģistrālajā tīklā – 0,1% vai mazāk.
- Vidēja trīce ASV maģistrālajā tīklā – 2 ms vai mazāk.

### 5. Verio SLA

- Pakešu aiztures ASV maģistrālajā tīklā – 50 ms vai mazāk.
- Pakešu zudumi ASV maģistrālajā tīklā – 0,1% vai mazāk.
- Vidēja trīce ASV maģistrālajā tīklā – 0,25 ms vai mazāk, nepārsniedzot 10ms vairāk nekā 0,1% mēneša kalendārajā laikā.

Praktiski visu pakalpojumu sniedzēju SLA ir ļoti līdzīgi un ierobežo maģistrālā tīkla pakešu aiztures  $\approx 40\text{ms} - 50\text{ms}$  līmenī, pakešu zudumus  $\approx 0,1\% - 0,3\%$  līmenī un trīci  $\approx 0,5\text{ ms} - 2\text{ ms}$  līmenī ar atrunu, ka ir pieļaujami īslaicīgi trīces uzliesmojumi līdz pat 10ms.

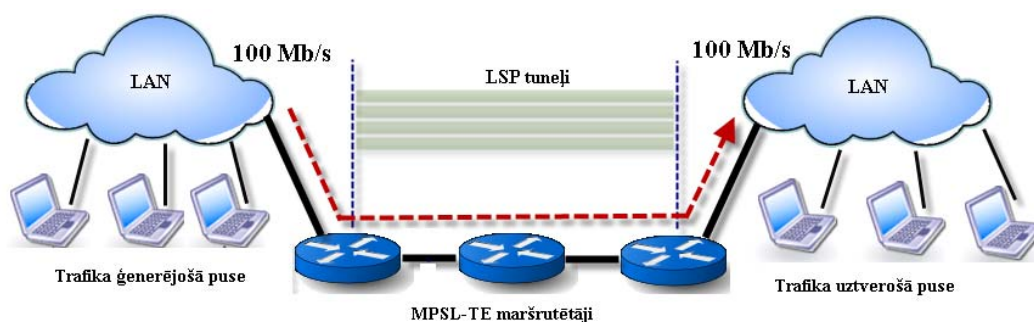
Tagad, noskaidrojot multimediju aplikāciju QoS parametru ierobežojumus un lielāko maģistrālo tīklu operatoru SLA ietvaros definētās QoS prasības, varam definēt nepieciešamos QoS ierobežojumus ar nolūku palielināt linka utilizāciju, pielietojot fazi-CAC algoritmu, un vienlaicīgi saglabāt nepieciešamās QoS parametru prasības.

Apskatītie SLA ierobežojumi pārsniedz pat visjutīgāko multimediju aplikāciju, tādu kā VoIP, iepriekš apskatītās QoS prasības, tāpēc šī darba autors definēs paša piedāvātā fazi-CAC algoritma pieskaņošanai nepieciešamās QoS parametru degradācijas robežas uz apskatīto pakalpojumu sniedzēju SLA vidējām svērtām vērtībām (Tabula. 2.2).

Elektronisko pakalpojumu sniedzēju SLA vidējās svērtās vērtības

Pakešu aiztures	Pakešu zudumi tīklā	Vidēja trīce
45 ms	0,2 %	1 ms

Tomēr pastāv pamatots jautājums par šī darba autora MPLS-TE eksperimentālā tīkla parametru izvēles salīdzināmību ar SLA datiem, kas ir iegūti no maģistrālo tīklu pakalpojumu sniedzējiem. Šī promocijas darba autora MPLS-TE eksperimentālā tīkla vienkāršota shēma ir redzama attēlā 2.7.



2.7. att. Fazi-CAC algoritma pārbaudes eksperimentālā MPLS-TE tīkla slēguma shēma

Kā redzams no attēla 2.7, šī slēguma shēma faktiski paredz tikai 2 lēkumus maģistrālā MPLS-TE tīkla robežās, kas rosina jautājumu par lielo maģistrālo tīklu pakalpojumu sniedzēju sniegto SLA datu pielietojumu šim testa tīklam.

Promocijas darba autors ir atradis datus par 3 lielo pakalpojumu sniedzēju – *Verizon*, *AT&T* un *Sprint Networks* topoloģijas īpašībām [31], kā arī par vidējo lēkumu skaitu šajos tīklos, veidojot savienojumus. Kā var redzēt no tabulām 2.3, 2.4 un 2.5, tad vidējais lēkumu skaits caur lielo maģistrālo tīklu ir  $\approx 2,7$ .

*Verizon* tīkla topoloģijas dati

<i>Mezglu skaits</i>	<i>Posmu skaits</i>	<i>Vidējais lēkumu skaits</i>
38	75	2,77

2.4. tabula

*AT&T* tīkla topoloģijas dati

<i>Mezglu skaits</i>	<i>Posmu skaits</i>	<i>Vidējais lēkumu skaits</i>
25	41	2,80

2.5. tabula

*Sprint Networks* tīkla topoloģijas dati

<i>Mezglu skaits</i>	<i>Posmu skaits</i>	<i>Vidējais lēkumu skaits</i>
25	56	2,66

SLA datu korektajam pielietojumam, definējot QoS parametru ierobežojumus pētāmajam MPLS-TE tīklam (PMT), ir iespējams reducēt lielo maģistrālo tīklu SLA datus, ņemot vērā lēkumu skaita atšķirības:

$$\frac{\text{Maģistrālo tīklu SLA ierobežojumi}}{\text{PMT ierobežojumi}} = \frac{\text{Maģistrālo tīklu vidējais lēkumu skaits}}{\text{Lēkumu skaits PMT}} \quad (2.25.)$$

$$\frac{45 \text{ ms}}{\text{PMT pakešu aiztures}} = \frac{2,7}{2} \Rightarrow \text{PMT pakešu aiztures} = 33 \text{ ms} \quad (2.26.)$$

$$\frac{45 \text{ ms}}{\text{PMT pakešu zudumi}} = \frac{2,7}{2} \Rightarrow \text{PMT pakešu zudumi} = 0,15 \% \quad (2.27.)$$

$$\frac{45 \text{ ms}}{\text{PMT trīce}} = \frac{2,7}{2} \Rightarrow \text{PMT trīce} = 0.7 \text{ ms} \quad (2.28.)$$

Šādā veidā nonākam pie fazi-CAC fazi izvedumu sistēmas parametru izvēles QoS ierobežojumiem, definējot vēlamo fazi-CAC algoritma uzvedību eksperimentālajā MPLS-TE sakaru sistēmas tīklā. (Tabula 2.6)

Fazi-CAC fazi izvedumu sistēmas QoS parametru ierobežojumi

Eksperimentālā MPLS-TE tīkla <b>pakešu aiztures</b>	Eksperimentālā MPLS-TE tīkla <b>pakešu zudumi</b>	Eksperimentālā MPLS-TE tīkla <b>vidēja trīce</b>
33 ms	0,15 %	0,7 ms

### 2.7.3. Heiristiskās pētījuma metodes apraksts

Optimālā vai tam tuva šī promocijas darba piedāvātā fazi-CAC izvedumu sistēmas risinājuma atrašana ir daudzu kritēriju uzdevums. Tas šajā konkrētajā gadījumā, izrietot no promocijas darba autora piedāvātā un definēta MPLS-TE sakaru sistēmas modeļa un fazi-CAC sistēmas modeļa, balstās uz

- fazi-CAC ieejas un izejas parametru lingvinistisko mainīgo skaitu,
- piederības funkciju definīciju atrašanu,
- IF-THEN kārtulu zināšanu bāzes atrašanu,
- fazi sistēmas tipa atrašanu,
- ieejas parametru fazifikācijas līkņu atrašanu,
- izejas parametra defazifikācijas metodes atrašanu.

Ir jāatrod arī piemērotais fazi-CAC algoritma atbildes pēcapstrādes paņēmieni, kas dotu iespēju definēt striktus CAC lēmumus.

Apskatot optimizācijas uzdevumam izvirzītos mērķus, varam redzēt, ka tas ietver plašu fazi izvedumu sistēmas un MPLS-TE tīkla analīzi, kas izpaužas gan kā linka vidējās utilizācijas  $\frac{\sum_1^i util(t_i)}{i}$  maksimizēšana, gan arī kā augstas prioritātes plūsmu selektīvā pieslēgšana, neizejot ārpus definētām QoS parametru degradācijas robežām. Arī plūsmu pieslēgumu skaita samazinājums, kas neatbilst iepriekš definētiem QoS kritērijiem, dod MPLS-TE sakaru sistēmai ieguvumu. Tas ļauj samazināt iestatīto LSP tuneļu slēgšanas gadījumu skaitu, ar turpmākajiem to atjaunotās iestatīšanas mēģinājumiem, rezultātā samazinot pārrauto sakaru sesiju skaitu, kā arī resursu rezervēšanai nepieciešamās servisa informācijas daudzumu tīklā.

Ņemot vērā uzdevuma komplikētību, promocijas darba autors ir izvēlējis heiristisko atlases, mēģinājumu un kļūdu pētījuma metodi, rezultātā nonākot pie optimumam tuvākā

risinājuma. Izvēlētā heuristiskā atlasē pētījuma metode balstās uz sākuma rupjo fazi-CAC lēmējsistēmas modeļa vairāku realizāciju kopas, kas tika izveidota un pārbaudīta, pamatojoties uz promocijas autora izveidoto MPLS-TE sakaru sistēmas vienkāršu simulācijas modeli Matlab vidē. Simulācijas modelis tiks aprakstīts turpmāk. Labākie no modeļa realizācijas variantiem tika modificēti un papildināti, līdz tika sasniegti rezultāti, kas bija vistuvākie izvērītiem Fazi-CAC modeļa veikspējas mērķiem.

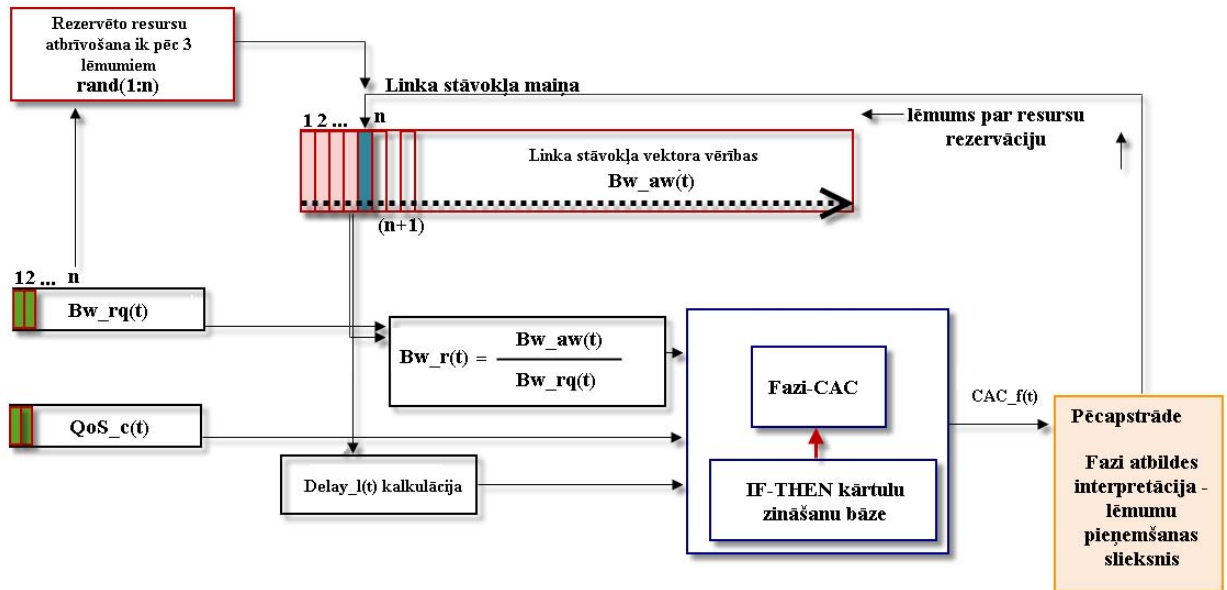
Definēsim interesējošos fazi-CAC modeļa parametrus, kuru vērtību atrašana ir nepieciešama iepriekš definēto fazi izvedumu sistēmas veikspējas mērķu sasniegšanai.

1. Katra ieejas parametra lingvinistisko mainīgo skaits un lingvinistisko mainīgo piederības funkciju deklarācijas.
2. Izejas parametra lingvinistisko mainīgo skaits un lingvinistisko mainīgo piederības funkciju deklarācijas.
3. Ieejas parametru fazifikācijas līkņu definīcijas.
4. Izejas parametra defazifikācijas metodes izvēle.
5. IF-THEN kārtulu zināšanu bāzes definīcija.
6. Fazi izvedumu sistēmas veids – *Mamdani, Tagaki-Sugeno*.
7. Iespējamais izejas parametra pēcapstrādes veids.

#### **2.7.4. MPLS-TE sakaru sistēmas modeļa simulācijas vide fazi-CAC lēmējsistēmas ātrai prototipēšanai**

Ar nolūku veikt fazi-CAC fazi izvedumu sistēmas modeļa iepriekš uzdotu parametru izpēti, kas nodrošinātu definēto mērķu sasniegšanu, tika izveidota nodaļā 2.1. definētās MPLS-TE sakaru sistēmas modeļa simulācijas realizācija Matlab vidē. Tika simulētas pētāmā linka diskreto stāvokļu izmaiņas brīžos, kad lēmums par jauno LSP iestatīšanu tika pieņemts, balstoties uz fazi-CAC algoritma atbildes.

Simulācijas algoritma blokshēma ir attēlota turpmāk (Att. 2.8). Simulācijas vides pirmkods ir dots pielikumā Nr. 1.



2.8. att. MPLS-TE sakaru sistēmas modeļa simulācijas algoritma blokskāme

$Bw_{rq}(t_i)$  un  $QoS_c(t_i)$  vērtību vektori tika ģenerēti ar fBm ģeneratoru [8]. Ir jāatrod  $Delay_l(t_i)$  vērtības, simulējot MPLS-TE tīkla darbību ar RSVP-TE protokola fazi-CAC algoritma balstītu darbību.

Kā izriet no nodaļā 2.1. definētā MPLS-TE eksperimentālā tīkla atsauču modeļa, varam secināt, ka  $Delay_l(t_i)$ , pirms defazifikācijas praktiski sastāvēs no pakešu apkalpošanas laiku summas maršrutētājos  $LER_{in}$ ,  $LSR$  un  $LER_{out}$ . Tātad, zinot, ka maršrutētāju  $LER_{in}$ ,  $LSR$  un  $LER_{out}$  apkalpošanas ātrums ir 10Mb/s, vidējais  $T_a$  ģenerēto pakešu garums ir 510 baiti, bufera izmērs 40 paketes un apkalpošanas secība ir FIFO, ir jāatrod maršrutētāju pakešu apkalpošanas laika atkarība no maršrutētāju noslogotības jeb linka utilizācijas pakāpes  $util(t_i)$ .

Sākumā salīdzināsim M/M/1 rindošanas modeli ar nolūku noteikt vidējo paketes apkalpošanas laika atkarību no maršrutētāja noslogotības pakāpes jeb utilizācijas.

Pielietosim šādus izejas datus:

$\mu = 10\text{Mb/s}$  – vidējais apkalpošanas ātrums;

$\lambda$  – vidējais pakešu atnākšanas ātrums;

$\rho = \frac{\lambda}{\mu}$  – trafika intensitāte jeb linka utilizācija;

$T = \frac{1}{\mu - \lambda}$  – kopējais paketes gaidīšanas laiks maršrutētājā jeb vidējā pakešu aizture;

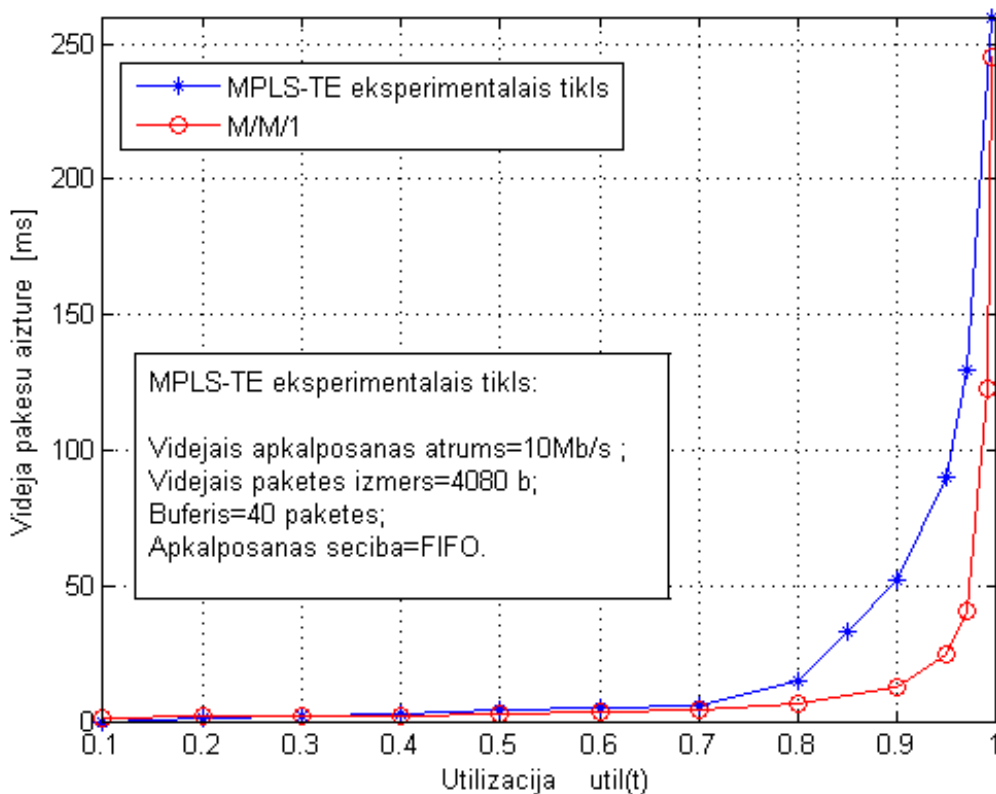
Vidējais paketes izmērs = 510 Baiti = 4080 biti.

Rezultātā ir iegūta M/M/1 rindošanas modeļa vidējās pakešu aiztures līkne atkarībā no tīkla utilizācijas pakāpes. Attēlā 2.9 ir parādītas vērtības summārām pakešu aizturēm  $LER_{in}$ ,  $LSR$  un  $LER_{out}$  maršrutētājos.

Lai definēt konkrētam MPLS-TE eksperimentālajam tīklam raksturīgu  $Delay_l(t_i)$  atkarību no  $util(t_i)$ , mērījumu rezultātā tika atrasta tālāk dotajā attēlā 2.9, līdzās M/M/1 modeļa līknei parādīta sakarība.

Ar nolūku nodrošināt simulācijas vides atbilstību eksperimentālā MPLS-TE tīkla parametriem, par simulācijas vides pamatu  $Delay_l(t_i)$  aprēķinātām vērtībām pirms fazifikācijas tika pieņemta MPLS-TE tīklam raksturīgā pakešu apkalpošanas laika likumsakarība.

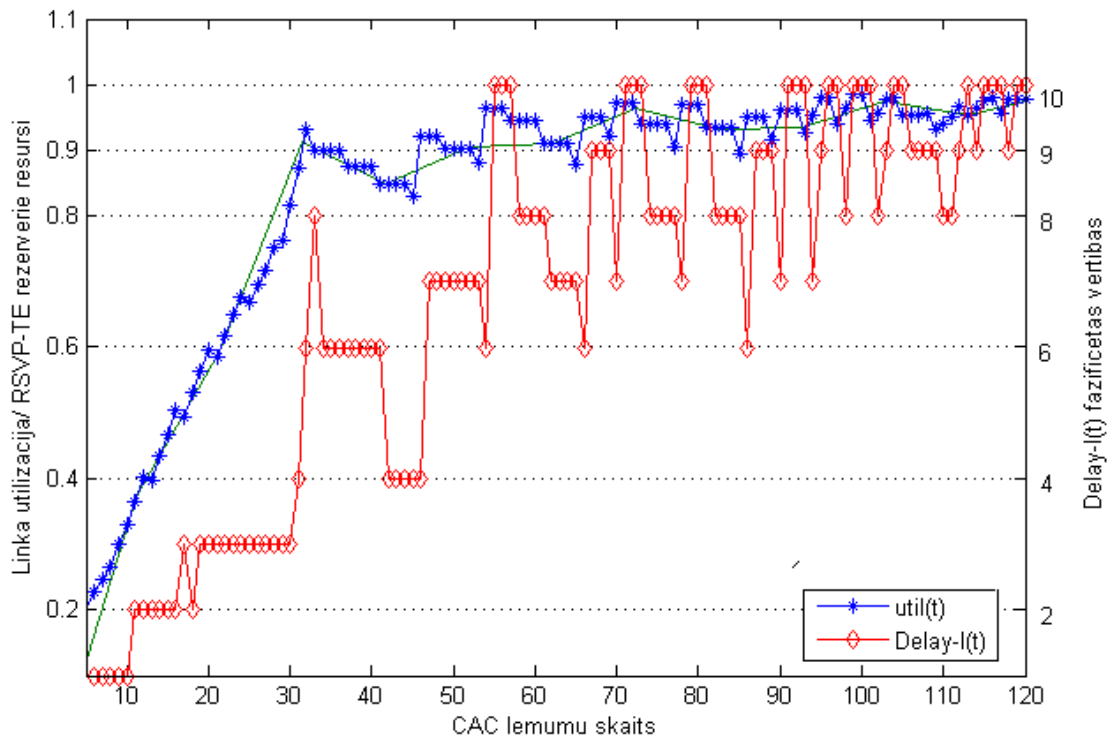
Zinot  $Bw_{aw}(t_i)$  vērtības un aprēķinātās  $Delay_l(t_i)$  vērtības, ir iespējams nodrošināt simulācijas vides rezultātu atbilstību konkrētā MPLS-TE eksperimentālā tīkla pakešu apkalpošanas laikam. Tas savukārt dod iespēju fazi-CAC fazi izvedumu sistēmas darbību simulācijas vides ietvaros pietuvināt reālā MPLS-TE eksperimentālā tīkla darbībai.



2.9. att. MPLS-TE eksperimentālā tīkla pakešu aizturu atkarība no linka utilizācijas

MPLS-TE sakaru sistēmas modeļa simulācijas vide tika pielietota ar nolūku ātri atlasīt labākos fazi-CAC lēmējsistēmas risinājumus, kas pēc turpmākām lingvinistisko mainīgo deklarāciju izmaiņām, kā arī IF-THEN kārtulu zināšanu bāzes modifikācijām tika pārbaudītas uz praktiskā MPLS-TE sakaru sistēmas eksperimentālā tīkla. MPLS-TE sakaru sistēmas modeļa simulācijas vide deva iespēju ātri pārbaudīt lielu skaitu fazi-CAC lēmējsistēmas risinājumu, virzoties uz iepriekš definētiem mērķiem.

MPLS-TE sakaru sistēmas tīkla simulācijas realizācija deva iespēju analizēt RSVP-TE rezervētos resursus linkā, vienlaicīgi sekojot līdzī arī  $Delay_l(t_i)$  vērtībām (Att. 2.10)  $Delay_l(t_i)$  fazifikācijas līkne ir dota nodaļā 2.7.5.

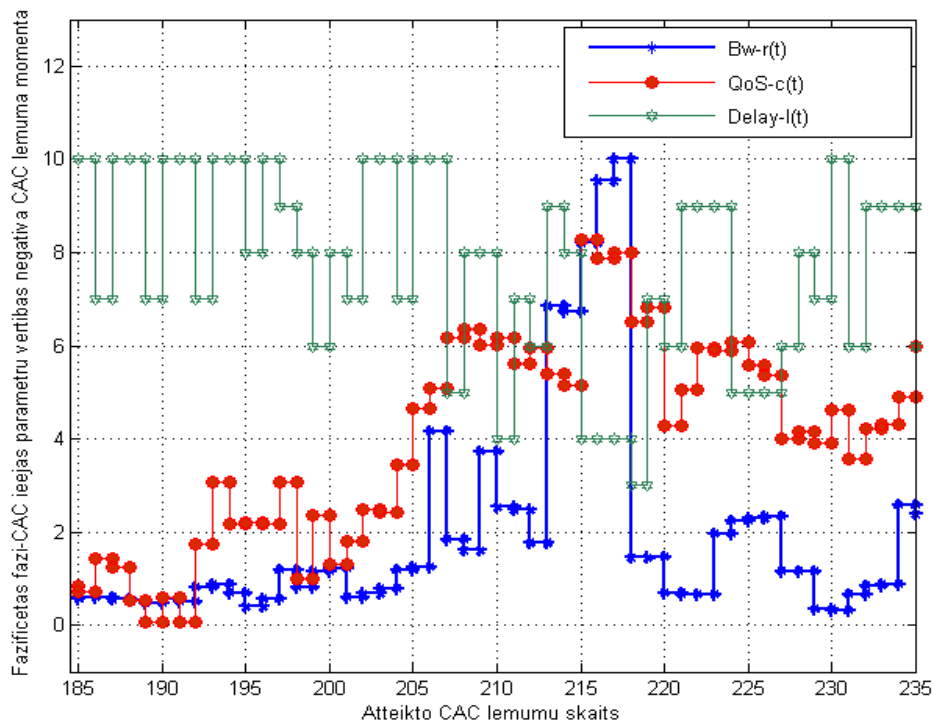


2.10. att. RSVP-TE rezervētie resursi un  $Delay_l(t_i)$  vērtības

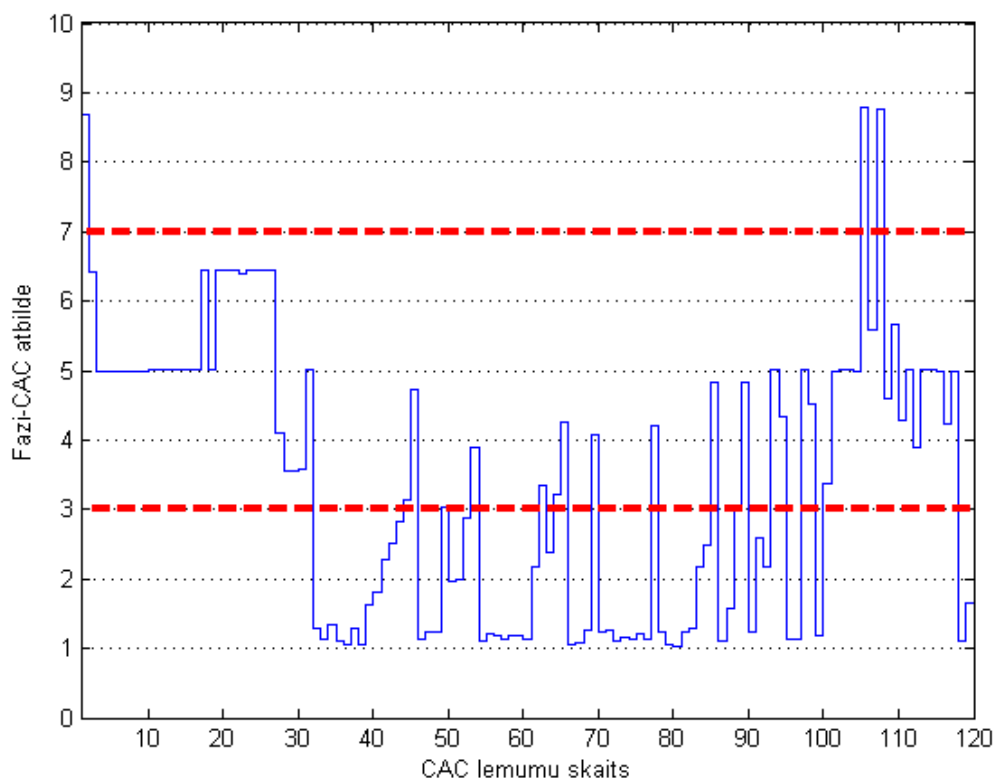
Ar simulācijas vides veiktajiem pētījumiem bija ērti analizēt ieejas parametru vektoru vērtības iestatīto un noraidīto LSP gadījumā (Att. 2.11), kā arī izvērtēt fazi lēmējsistēmas atbildes pēcapstrādes sliekšņu vērtības ar nolūku iegūt fazi-CAC sistēmas veikspējas parametrus pēc iespējas tuvākus izvirzītajiem mērķiem (Att. 2.12).

Zemāk dotajā attēlā 2.11 ir parādīts neliels, 50 CAC lēmumu izgriezums no reģistrētajām fazi-CAC fazi izvedumu sistēmas ieejas parametru vērtībām, kas parāda doto parametru vērtības negatīvā fazi-CAC lēmuma gadījumā. No attēla ir skaidri redzama selektīvā fazi-CAC lēmumu pieņemšanas politika, kad arī gadījumos, kad  $Bw_r(t_i)$  vērtības ir lielas, fazi-CAC lēmumi ir negatīvi. Šāda lēmuma iemesls ir augstas  $QoS_c(t_i)$  vērtības ar vienlaicīgi augstām  $Delay_l(t_i)$  vērtībām. Sliekšņa-CAC algoritms pieslēgtu jaunas LSP plūsmas visos gadījumos, kad  $Bw_r(t_i) \geq 1$ , pilnīgi ignorējot aplikācijas augstās QoS prasības un lielas pakešu aizturu vērtības tīklā.

Attēlā 2.12 ir parādīta fazi-CAC algoritma  $CAC_f(t_i)$  atbilde un simetriski  $CAC_f(t_i)$  pēcapstrādes sliekšņi.



2.11. att. Fazi-CAC fazi izvedumu sistēmas ieejas parametru vērtības negatīvo CAC lēmumu gadījumā



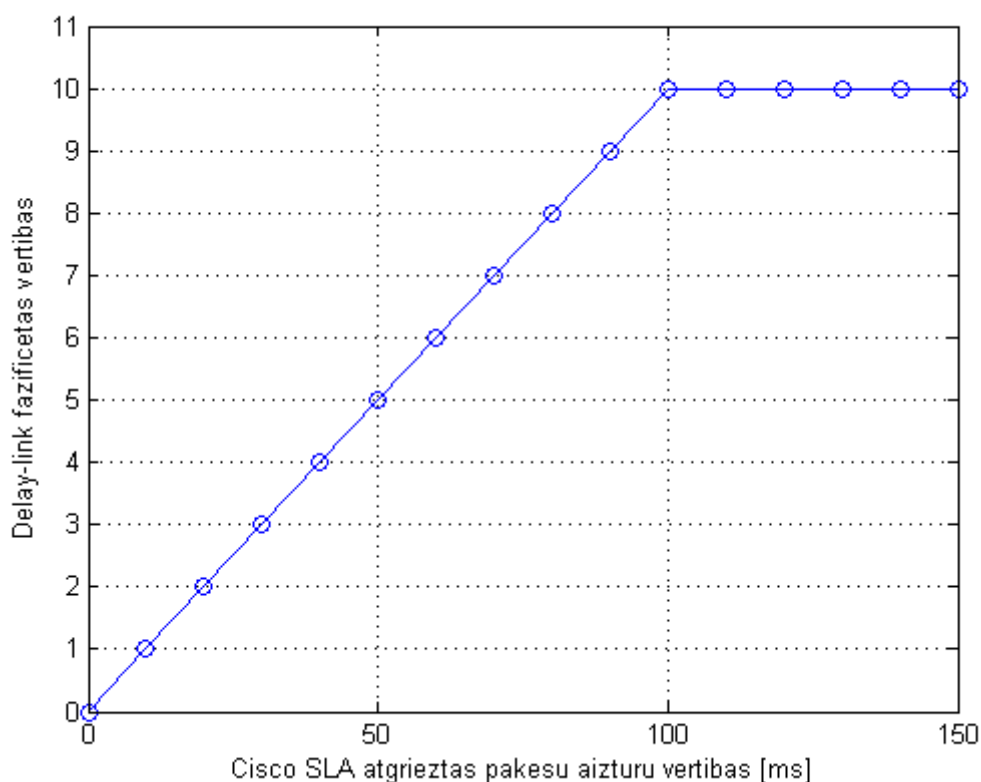
2.12. att. Fazi-CAC atbilde un simetriska sliekšņa vērtība

### 2.7.5. Ieejas parametru fazifikācijas līkņu definīcijas

Kā jau tika minēts, vienīgais promocijas darba piedāvātā fazi-CAC modeļa ieejas parametrs, kas tiks pirms pielietošanas papildus fazificēts, ir  $Delay_l(t_i)$ .  $Delay_l(t_i)$  vērtības iegūšana pirms defazifikācijas ir aprakstīta nodaļā 3.1.3, pie MPLS-TE sakaru sistēmas eksperimentālā tīkla apraksta.

$Delay_l(t_i)$  parametra vērtību fazifikācijas robežas tika noteiktas empīriski, analizējot eksperimentālās MPLS-TE sakaru sistēmas maketa aizturu dinamiku, pakāpeniski palielinot slodzi un nosakot brīdi, kad buferatmiņas pārpildīšanas dēļ tīklā konstatēja pakešu zudumus. Pakešu zudumu sākuma robeža pakešu aizturu izteiksmē [ms] tika definēta kā maksimālā piederības funkcijas vērtība.

Šāda robeža tika konstatēta pie  $\approx 100$  ms šai konkrētajai MPLS-TE sakaru sistēmas praktiskās realizācijas eksperimentālā tīkla konfigurācijai. Rezultātā, fazificēto  $Delay_l(t_i)$  mainīgā vērtību robežas nosaka funkcija, kas parādīta attēlā 2.13.



2.13. att.  $Delay_l(t_i)$  ieejas parametra fazifikācijas līkne

### 2.7.6. Ieejas un izejas parametru lingvinistisko mainīgo piederības funkciju deklarācijas

Heiristiskās atlasēs rezultātā tika izvēlēta fazi-CAC modeļa realizācija ar šādām ieejas un izejas parametru lingvinistisko mainīgo piederības funkciju deklarācijām.

Iespējamie faziloģikas lingvinistisko mainīgo piederības funkciju tipi tika apskatīti nodaļā 1.4.1, un šajā algoritmā sākotnēji tika pieņemts lēmums pielietot visvienkāršāko piederības funkciju veidu – trijstūra piederības funkcijas. Tās prasa mazāk skaitļošanas jaudas nekā citas piederības funkcijas un tām nav izteiktu platu. Tas savukārt attālina šādu piederības funkciju definīciju no klasiskās bivalentās loģikas un ļauj izmantot visas faziloģikas priekšrocības.

Autora piedāvātajā algoritmā ieejas un izejas parametriem tika pielietotas šādas lingvinistisko mainīgo piederības funkcijas:

1.  $Bw_r(t_i)$  ieejas parametra gadījumā tika pieņemts, ka tā stāvokļus aprakstīs 4 lingvinistiskie mainīgie (Att. 2.14):

a.  $Bw_r(t_i)$  ir „nepieņemams”, ko apraksta šāda piederības funkcija  $B_n(x)$ :

$$B_n(x) = \begin{cases} 1, & x \leq 0 \\ \frac{2-x}{2}, & 0 \leq x \leq 2 \\ 0, & 2 \leq x \end{cases} \quad (2.29)$$

b.  $Bw_r(t_i)$  ir „slikts”, ko apraksta šāda piederības funkcija  $B_s(x)$ :

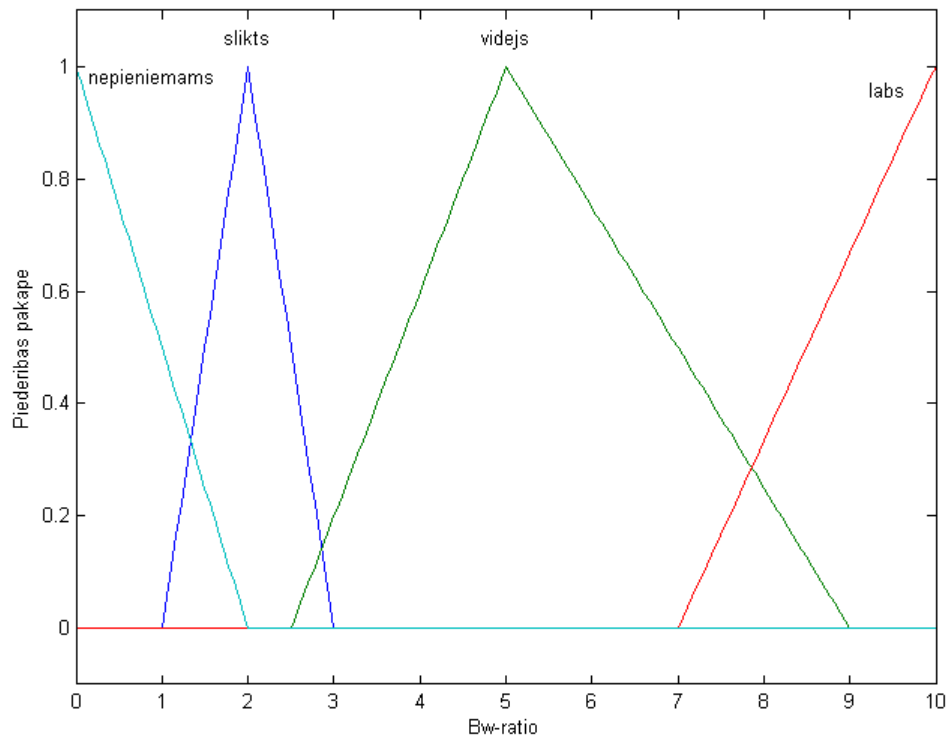
$$B_s(x) = \begin{cases} 0, & x \leq 1 \\ x - 1, & 1 \leq x \leq 2 \\ 3 - x, & 2 \leq x \leq 3 \\ 0, & 3 \leq x \end{cases} \quad (2.30.)$$

c.  $Bw_r(t_i)$  ir „vidējs”, ko apraksta šāda piederības funkcija  $B_v(x)$ :

$$B_v(x) = \begin{cases} 0, & x \leq 2,5 \\ \frac{x-2,5}{2,5}, & 2,5 \leq x \leq 5 \\ \frac{9-x}{4}, & 5 \leq x \leq 9 \\ 0, & 9 \leq x \end{cases} \quad (2.31)$$

d.  $Bw_r(t_i)$  ir „labs”, ko apraksta šāda piederības funkcija  $B_l(x)$ :

$$B_l(x) = \begin{cases} 0, & x \leq 7 \\ \frac{x-7}{3}, & 7 \leq x \leq 10 \\ 1, & 10 \leq x \end{cases} \quad (2.32.)$$



2.14. att.  $Bw_r(t_i)$  ieejas parametra lingvinistisko mainīgo trījstūra piederības funkcijas

**2.  $Delay_l(t_i)$  ieejas parametra gadījumā tika pieņemts, ka to aprakstīs 3 lingvinistiskie mainīgie (Att. 2.15):**

a.  $Delay_l(t_i)$  aiztures ir „mazas”, ko apraksta šāda piederības funkcija  $D_m(x)$ :

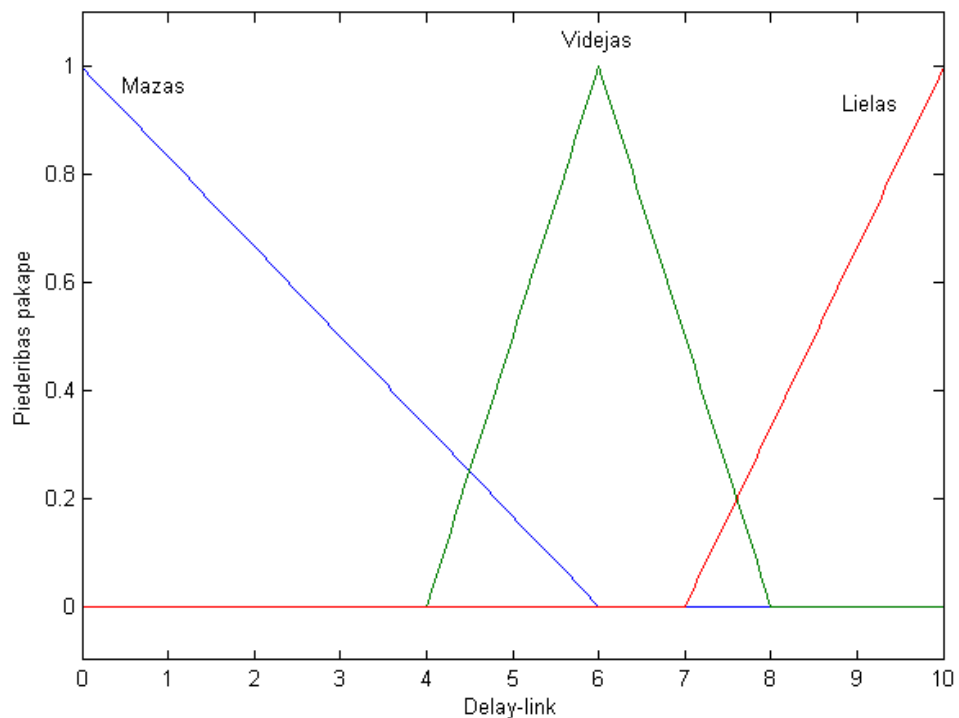
$$D_m(x) = \begin{cases} 1, & x \leq 0 \\ \frac{6-x}{6}, & 0 \leq x \leq 6 \\ 0, & 6 \leq x \end{cases} \quad (2.33.)$$

b.  $Delay_l(t)$  aiztures ir „vidējas”, ko apraksta šāda piederības funkcija  $D_v(x)$  :

$$D_v(x) = \begin{cases} 0, & x \leq 4 \\ \frac{x-4}{2}, & 4 \leq x \leq 6 \\ \frac{8-x}{2}, & 6 \leq x \leq 8 \\ 0, & 8 \leq x \end{cases} \quad (2.34.)$$

c.  $Delay\_l(t_i)$  aiztures ir „lielas”, ko apraksta šāda piederības funkcija  $D_l(x)$ :

$$D_l(x) = \begin{cases} 0, & x \leq 7 \\ \frac{x-7}{3}, & 7 \leq x \leq 10 \\ 1, & 10 \leq x \end{cases} \quad (2.35.)$$



2.15. att.  $Delay\_l(t_i)$  ieejas parametra lingvinistisko mainīgo trijstūra piederības funkcijas

**3.  $QoS\_c(t_i)$  ieejas parametra gadījumā tika pieņemts, ka to aprakstīs 3 lingvinistiskie mainīgie (Att. 2.16):**

- a.  $QoS_c(t_i)$  ir „best effort”, jeb zema līmeņa, ko apraksta šāda piederības funkcija  $Q_b(x)$ :

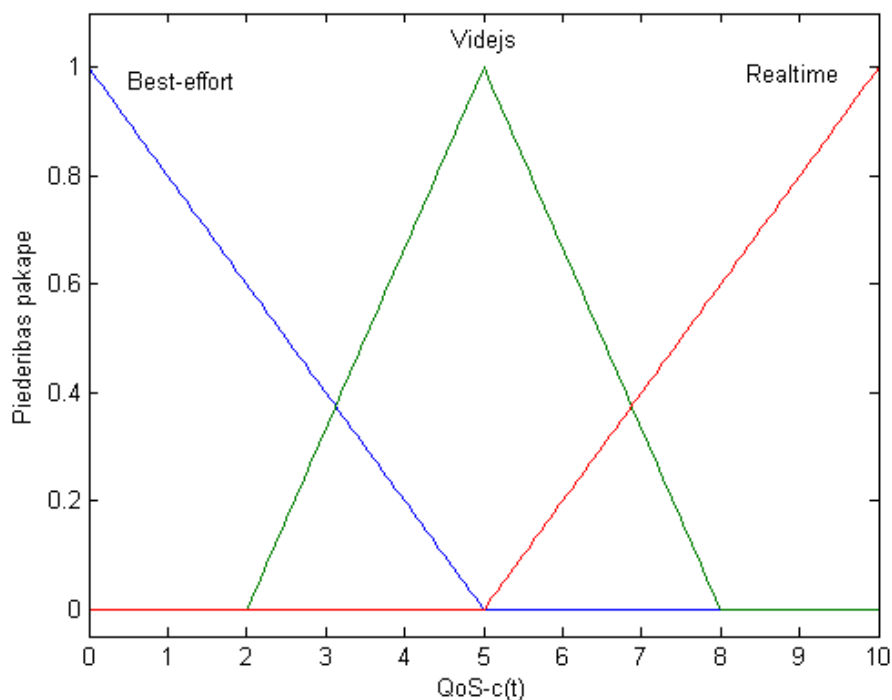
$$Q_b(x) = \begin{cases} 1, & x \leq 0 \\ \frac{5-x}{5}, & 0 \leq x \leq 5 \\ 0, & 5 \leq x \end{cases} \quad (2.36.)$$

- b.  $QoS_c(t_i)$  ir vidējs, ko apraksta šāda piederības funkcija  $Q_v(x)$ :

$$Q_v(x) = \begin{cases} 0, & x \leq 2 \\ \frac{x-2}{3}, & 2 \leq x \leq 5 \\ \frac{8-x}{3}, & 5 \leq x \leq 8 \\ 0, & 8 \leq x \end{cases} \quad (2.37.)$$

- c.  $QoS_c(t_i)$  ir t.s. „realtime” jeb ar augstāko prioritāti, ko apraksta šāda piederības funkcija  $Q_r(x)$ :

$$Q_r(x) = \begin{cases} 0, & x \leq 5 \\ \frac{x-5}{5}, & 5 \leq x \leq 10 \\ 1, & 10 \leq x \end{cases} \quad (2.38.)$$



2.16. att.  $QoS_c(t_i)$  ieejas parametra lingvinistisko mainīgo trijstūra piederības funkcijas

4.  $CAC_f(t_i)$  izejas parametra gadījumā tika pieņemts, ka tā stāvokļus aprakstīs 3 lingvinistiskie mainīgie (Att. 2.17):

a.  $CAC_f(t_i)$  ir „atteikt”, ko apraksta šāda piederības funkcija  $C_{att}(x)$ :

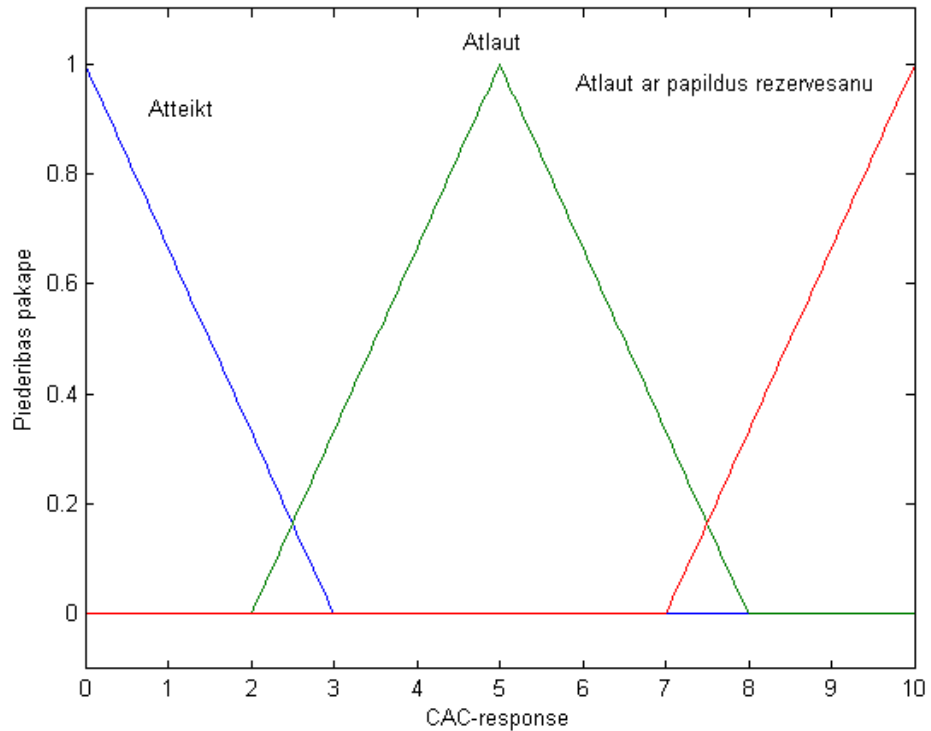
$$C_{att}(x) = \begin{cases} 1, & x \leq 0 \\ \frac{3-x}{3}, & 0 \leq x \leq 3 \\ 0, & 3 \leq x \end{cases} \quad (2.39.)$$

b.  $CAC_f(t_i)$  ir „atļaut”, ko apraksta šāda piederības funkcija  $C_{atl}(x)$ :

$$C_{atl}(x) = \begin{cases} 0, & x \leq 2 \\ \frac{x-2}{3}, & 2 \leq x \leq 5 \\ \frac{8-x}{3}, & 5 \leq x \leq 8 \\ 0, & 8 \leq x \end{cases} \quad (2.40.)$$

c.  $CAC_f(t_i)$  ir „atļaut ar papildu rezervēšanu”, ko apraksta šāda piederības funkcija  $C_{atl_r}(x)$ :

$$C_{atl_r}(x) = \begin{cases} 0, & x \leq 7 \\ \frac{x-7}{3}, & 7 \leq x \leq 10 \\ 1, & 10 \leq x \end{cases} \quad (2.41.)$$



2.17. att.  $CAC_f(t_i)$  izejas parametra lingvinistisko mainīgo trījstūra piederības funkcijas

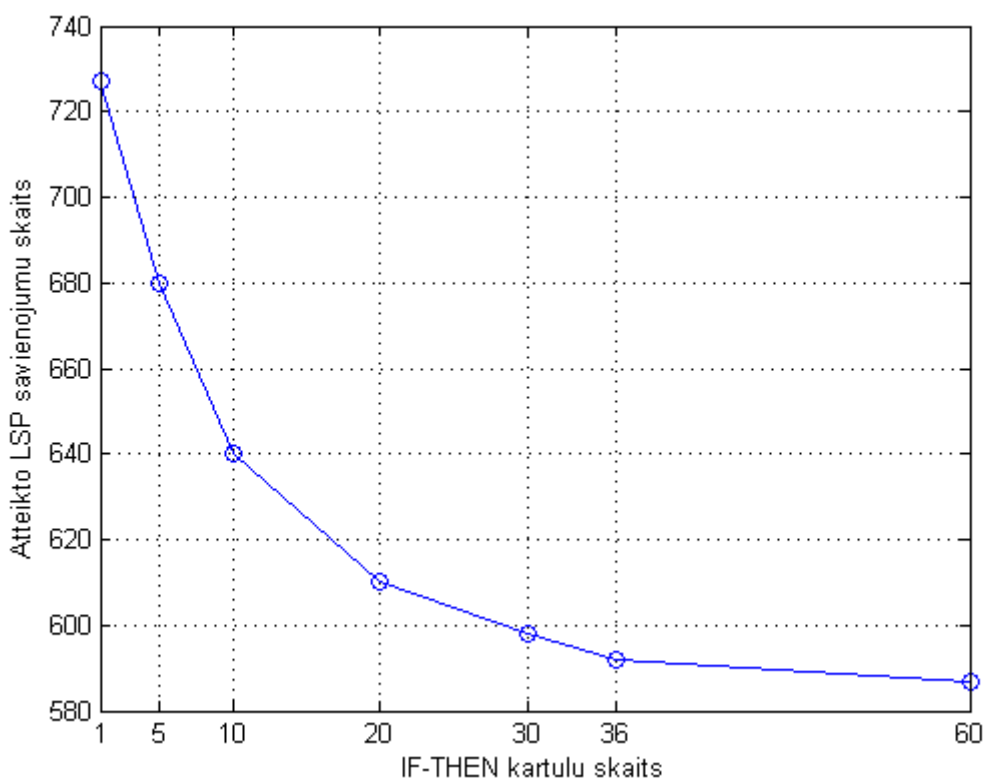
### 2.7.7. Pielietotā IF-THEN kārtulu zināšanu bāze

Fazi-CAC lēmumi ir balstīti uz zināšanu bāzi IF-THEN kārtulu veidā. Fazi-CAC ieejas parametru lingvinistisko mainīgo skaits nosaka arī IF-THEN kārtulu skaitu un kopējo faziloģikas izvedumu sistēmas veiktspēju. Ir loģiski, ka, palielinot ieejas parametru lingvinistisko mainīgo skaitu un rezultātā arī IF-THEN kārtulu skaitu, mēs varam aprakstīt sistēmas stāvokļus daudz precīzāk, tomēr paliek atvērts jautājums par skaitļošanai nepieciešamo laiku, kāds būtu nepieciešams lēmuma pieņemšanai.

Mūsdienu ātrgaitas optisko tīklu kontekstā, tādiem tiešsaistes uzdevumiem kā CAC kontrole lēmuma pieņemšanas laiks ir kritiski svarīgs parametrs. Brīdī, kad cenšamies uzlabot kādu no CAC kontroles veiktspējas parametriem, piemēram, cenšoties samazināt atteikto savienojumu skaitu, ir jāseko līdzi arī skaitļošanai nepieciešamā laika izmaiņām. Tas ir nepieciešams, lai rezultātā nesamazinātu kopējo lēmēj sistēmas veiktspēju, jo tā šajā gadījumā ietver sevī arī lēmuma pieņemšanai nepieciešamo laika nogriezni.

Attēlā 2.18 ir attēlots autora piedāvātā fazi-CAC algoritma realizācijas atteikto savienojumu skata samazinājums 1500 lēmumiem, palielinot pielietoto ieejas parametru lingvinistisko mainīgo skaitu un attiecīgi arī IF-THEN kārtulu skaitu. „Nulles” kārtulu marķējums uz grafika apzīmē klasiskā sliekšņa CAC veikspēju 1500 lēmumiem, un turpmākās atzīmes uz grafika parāda atteikto LSP konekciju samazinājumu, palielinot IF-THEN kārtulu skaitu fazi-CAC izvedumu sistēmas zināšanu bāzē.

Ir skaidri redzams, ka šai konkrētajai fazi-CAC lēmējmehānisma implementācijai IF-THEN kārtulu skaita palielināšana virs noteikta skaita, kurš šajā gadījumā ir 36, noved pie ļoti lēna atteikto LSP savienojumu skaita samazinājuma un dod mazu ieguldījumu kopējā lēmējsistēmas veikspējā.

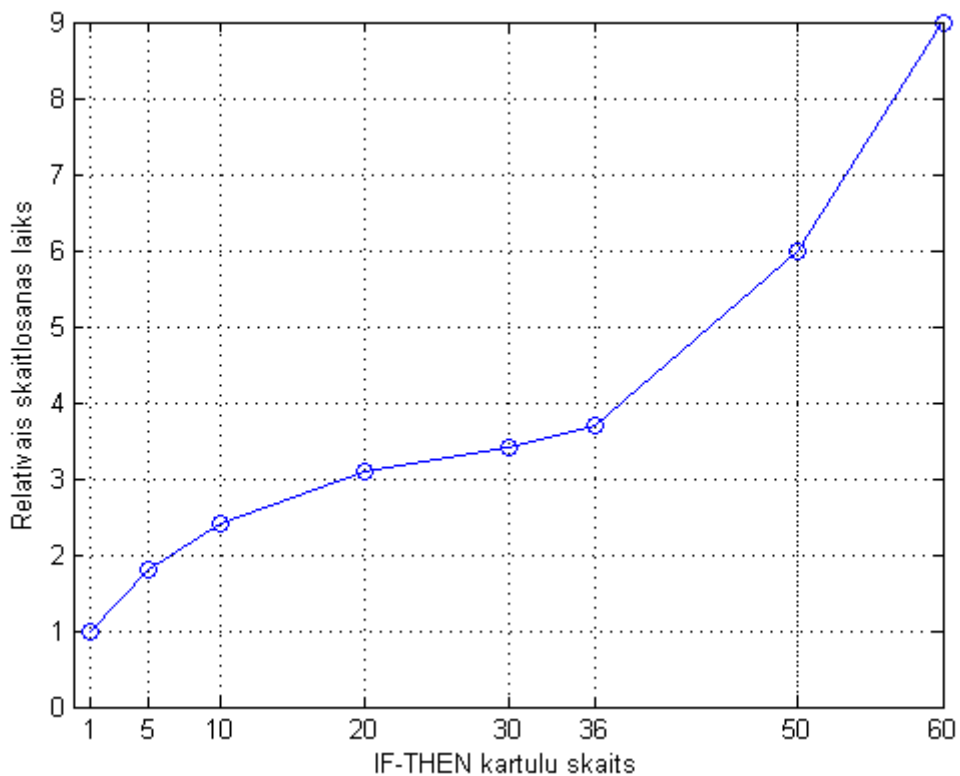


2.18. att. Atteikto LSP savienojumu skaits atkarībā no pielietoto IF-THEN kārtulu skaita fazi-CAC lēmējmehānisma implementācijā 1500 lēmumiem

Toties paralēli lēnajam atteikto LSP konekciju skaita samazinājumam varam novērot ļoti strauju skaitļošanas laika pieaugumu, kas ir nepieciešams lēmuma pieņemšanai, palielinot IF-THEN kārtulu skaitu (Att. 2.19). No simulācijas rezultātiem ir skaidri redzams, ka turpmāka IF-THEN kārtulu palielināšana virs kāda noteikta sliekšņa konkrētai fazi-CAC fazi izvedumu sistēmas implementācijai noved pie strauja skaitļošanas laika pieauguma, kas

paralēli dod salīdzinoši nelielu fazi izvedumu sistēmas veiktspējas uzlabošanu atteikto LSP savienojumu skaita ziņā.

Jāpiebilst, ka laiks ir attēlots kā relatīvs laiks attiecībā pret skaitļošanas laiku 1500 lēmumiem klasiskā sliekšņa-CAC algoritma gadījumā.



2.19. att. Lēmuma pieņemšanai nepieciešamā relatīvā skaitļošanas laika pieaugums atkarībā no pielietoto IF-THEN kārtulu skaita 1500 lēmumiem

Balstoties uz atlasē heuristiskās pētījumu metodes rezultātiem ar nolūku sasniegt nodaļā 2.7.1 definētus fazi-CAC lēmējsistēmas efektivitātes mērķus, tika izvēlēta IF-THEN kārtulu zināšanu bāze, kas sastāv no 36 kārtulām. Tās apkopo šādus loģiskos pieņēmumus, pielietojot iepriekš definētās fazi-CAC lēmējsistēmas ieejas parametru lingvīnistisko mainīgo piederības funkcijas:

- 1. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 2. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „videjas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 3. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 4. **IF**-( $Bw_r(t)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „vidējs”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 5. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „vidējs”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 6. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „vidējs”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 7. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 8. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 9. **IF**-( $Bw_r(t_i)$  ir „slikts”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 10. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 11. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 12. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 13. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „vidējs”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 14. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „vidējs”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut ar papildu rezervēšanu”)
- 15. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „vidējs”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 16. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)

- 17. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 18. **IF**-( $Bw_r(t_i)$  ir „vidējs”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 19. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 20. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 21. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut”)
- 22. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „vidēja”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut ar papildu rezervēšanu”)
- 23. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „vidēja”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut ar papildu rezervēšanu”)
- 24. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „vidēja”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 25. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atļaut ar papildu rezervēšanu”)
- 26. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t)$  ir „atļaut”)
- 27. **IF**-( $Bw_r(t_i)$  ir „labs”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 28. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 29. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t)$  ir „atteikt”)
- 30. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „Best-effort”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 31. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t)$  ir „vidēja”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 32. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „vidēja”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)

- 33. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „vidēja”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 34. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „mazas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 35. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „vidējas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)
- 36. . **IF**-( $Bw_r(t_i)$  ir „nepieņemams”) **AND**-( $QoS_c(t_i)$  ir „Realtime”) **AND**-( $Delay_l(t_i)$  ir „lielas”) **THEN**-( $CAC_f(t_i)$  ir „atteikt”)

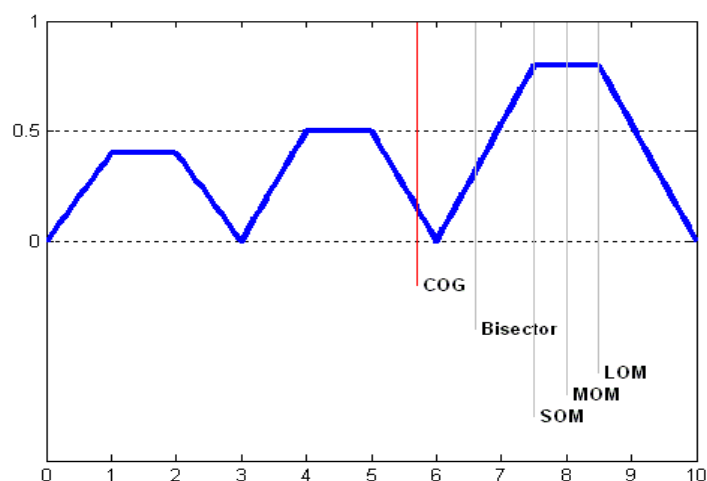
### 2.7.8. Defazifikācijas metodes izvēle

Defazifikācijas metodes izvēle tieši nosaka fazi lēmējmehānisma izvada datu interpretācijas īpašības, kas spriežot pēc nodaļā 1.4.5 dotiem defazifikācijas metožu izvada piemēriem, var būtiski atšķirties. Rodas jautājums par defazifikācijas metodes izvēles pamatojumu.

Tipiski, ka fazi sistēma saturēs daudzas IF-THEN kārtulas, kas pārveido konkrētos ieejas mainīgos fazi izvadā, tātad fazi lēmējmehānisma atbilde tiks attēlota kā piederība pie fazi kopas. Piemēram, šī promocijas darba autora piedāvātais fazi-CAC algoritms operē ar trijiem izvada lēmumiem: „Atļaut LSP iestatīšanu”, „Atteikt LSP iestatīšanu” un „Atļaut LSP iestatīšanu ar papildu resursu rezervēšanu”.

Iztēlojamies situāciju, kad IF-THEN kārtulu zināšanu bāze, pielietojot fazificētās ieejas parametru vērtības, izdod atbildi, kas attēlo šādu izvadu (Att. 2.20):

1. „Atteikt LSP iestatīšanu” – 40%;
2. „Atļaut LSP iestatīšanu” – 50%;
3. „Atļaut LSP iestatīšanu ar papildu resursu rezervēšanu” – 80%.



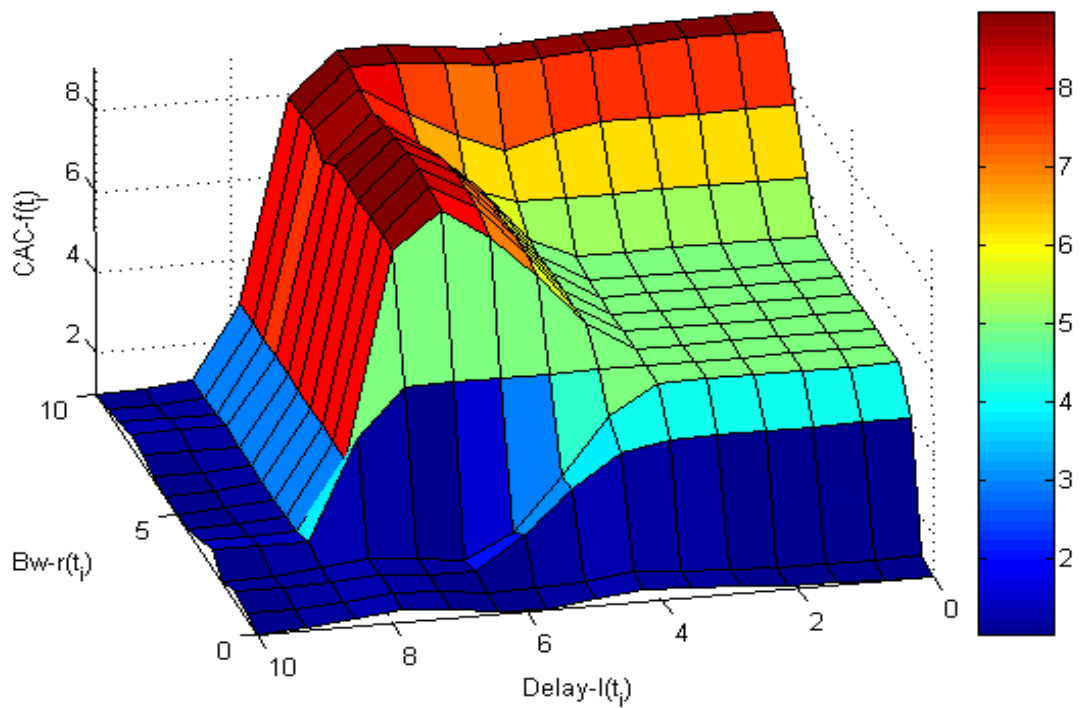
2.20. att. Defazifikācijas metožu COG, Bisector, SOM, MOM un LOM rezultāti

Defazifikācijas procesam šis fazi izvads ir jāreducē uz vienu skalāru vērtību, kas noteiks algoritma darbību. Pats vienkāršākais, bet jāatzīst, ka arī mazāk lietderīgs veids ir izvēlēties fazikopu ar vislielāko piederības pakāpi – šajā gadījumā „Atļaut LSP iestatīšanu ar papildus resursu rezervēšanu”, jo tam ir 80% piederības pakāpe. Šajā gadījumā tiks ignorētas pārējās fazikopas ar mazāku piederības pakāpi, bet šī konkrētā fazikopa tiks pārvērsta par kādu konkrētu skaitli, pielietojot, piemēram, SOM, MOM vai LOM defazifikācijas metodes. Tomēr šīs pieejas problēma ir informācijas zudums, jo visa zināšanu bāzes informācija, kas tika pielietota, lai norādītu arī uz citām iespējamām fazi sistēmas atbildēm, tiks ignorēta.

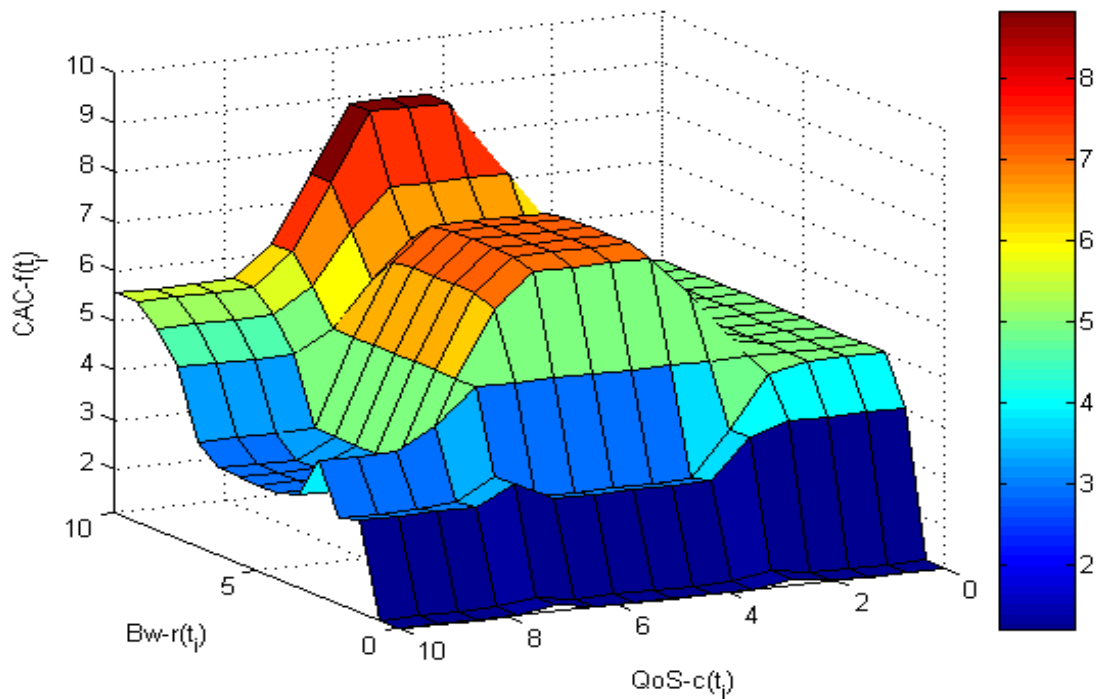
Defazifikācijas tehnikai, kas spētu saglabāt visu fazi atbilžu sniegto informāciju, ir jāņem vērā visu fazikopu rezultāti. Šādu rezultātu sniedz, piemēram, t.s. centroida jeb COG defazifikācijas metode, kas defazifikācijas rezultātā atgriež virsmas centra punktu zem piederības funkciju līknes, un kā defazificētā vērtība tiek pielietota X koordinātā.

Šī promocijas darba fazi-CAC algoritma realizācijā tika pielietota vispopulārākā defazifikācijas metode, kas nosaka t.s. gravitācijas centru (jeb t.s. centroida metode), kura ir sīkāk aprakstīta nodaļā 1.4.5.

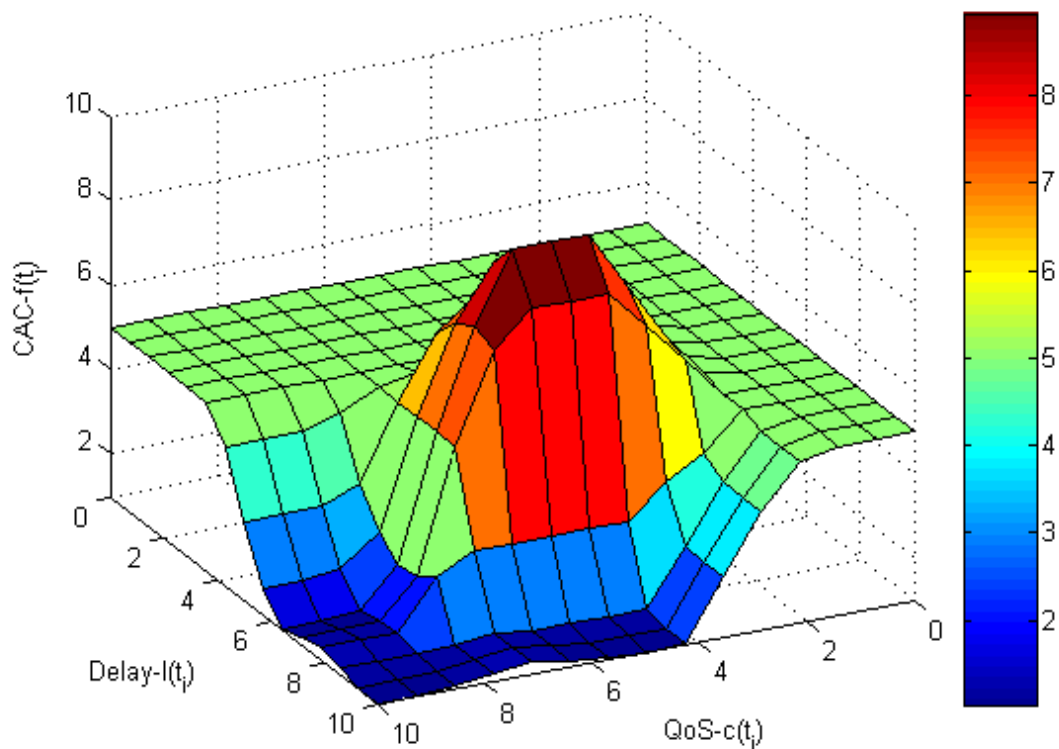
Šinī fazi-CAC algoritma konkrētajā realizācijā, pielietojot gravitācijas centra metodi defazifikācijas procesā, tika iegūtas šādas lēmumu virsmas, kuras apraksta fazi lēmējsistēmas reakciju visām iespējamām ieejas vērtību variācijām, pielietojot iepriekš noteiktas fazi IF-THEN kārtulas (Att. 2.21, 2.22, 2.23)



2.21. att. Fazi-CAC lēmējmehānisma lēmumu virsma  $Delay\_l(t_i)$  un  $Bw\_r(t_i)$  ieejas mainīgo vērtību gadījumā



2.22. att. Fazi-CAC lēmējmehānisma lēmumu virsma  $QoS\_c(t_i)$  un  $Bw\_r(t_i)$  ieejas mainīgo vērtību gadījumā



2.23. att. Fazi-CAC lēmējmehānisma lēmumu virsma  $QoS\_c(t_i)$  un  $Delay\_l(t_i)$  ieejas mainīgo vērtību gadījumā

Iegūtās lēmumu virsmas varētu ļoti ātri un ērti pielietot, veidojot kontrolieri ar matricas tipa ierakstiem loģikas vadībā. Šādā gadījumā lēmumu būtu iespējams pieņemt, balstoties uz uzdotajiem defazificētiem linka stāvokļiem. Tomēr ir jāatzīst arī tas, ka šāda rīcība paralēli fazi-CAC algoritma implementācijai ievērojami mazinās faziloģikas lielisko tiešsaistes optimizācijas īpašību, ko varētu veikt, dinamiski mainot zināšanu bāzi un piederības funkciju deklarācijas.

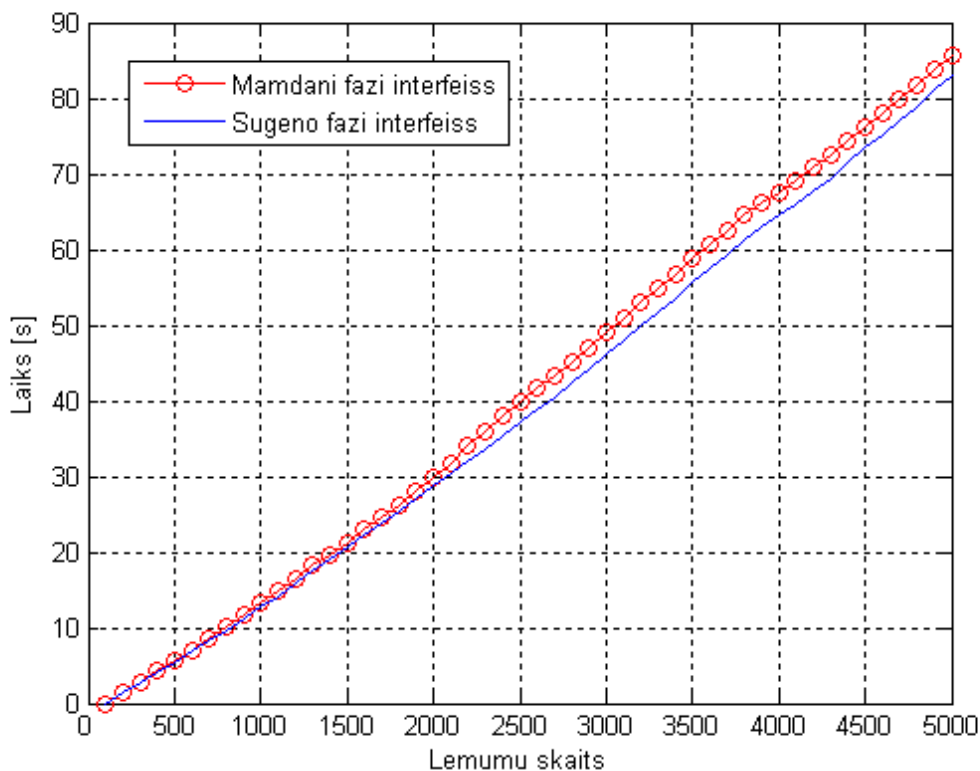
### 2.7.9. Fazi izvedumu sistēmas veida izvēle

No izvedumu procesa viedokļa, ir iespējams izšķirt divus pamata Fazi Izvedumu Sistēmas (FIS) tipus:

- c. *Mamdani* tipa izvedumu sistēma;
- d. *Takagi-Sugeno* izvedumu sistēma.

Ar nolūku izvēlēties fazi-CAC efektīvai darbībai labāku fazi izvedumu sistēmas risinājumu ir nepieciešams apzināties galvenos fazi izvedumu sistēmu raksturojošos parametrus un to ietekmi uz fazi-CAC efektivitāti.

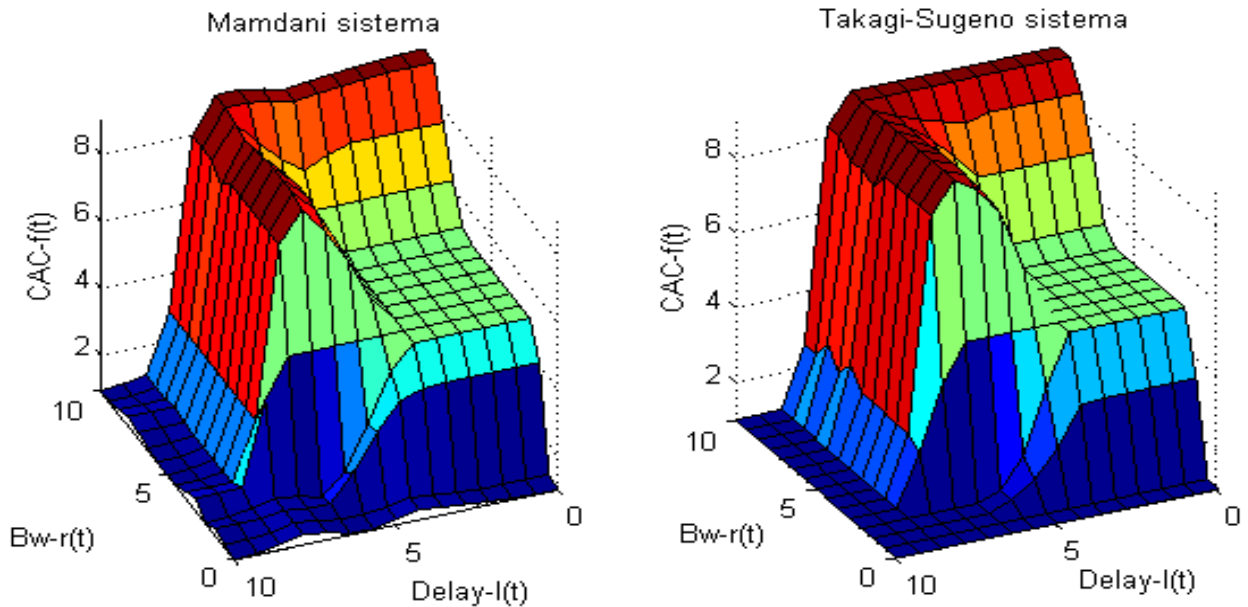
Galvenā atšķirība starp *Mamdani* un *Takagi-Sugeno* fazi izvedumu sistēmām ir dažāda tipa izvada piederības funkciju definīcijas. *Sugeno* tipa fazi sistēmas satur vai nu lineārās izvada piederības funkcijas, vai konstantes. *Mamdani* fazi izvedumu sistēma var saturēt jebkura tipa izvada piederības funkciju definīcijas. Rezultātā, no skaitļošanas viedokļa, *Takagi-Sugeno* metode ir efektīvāka un, iespējams, dos ātrāku reakciju [7]. Ātra skaitļošanas atbilde ir kritiska ātrgaitas komutācijas ierīcēs, jo no tās ir atkarīga kopējā trafika vadības efektivitāte. Lai izvēlētos kādu konkrētu fazi izvedumu sistēmas vaidu, promocijas darba autors ir veicis konkrētās fazi-CAC lēmējsistēmas ātrdarbības salīdzinošu analīzi, kuras rezultāti ir parādīti attēlā 2.24.



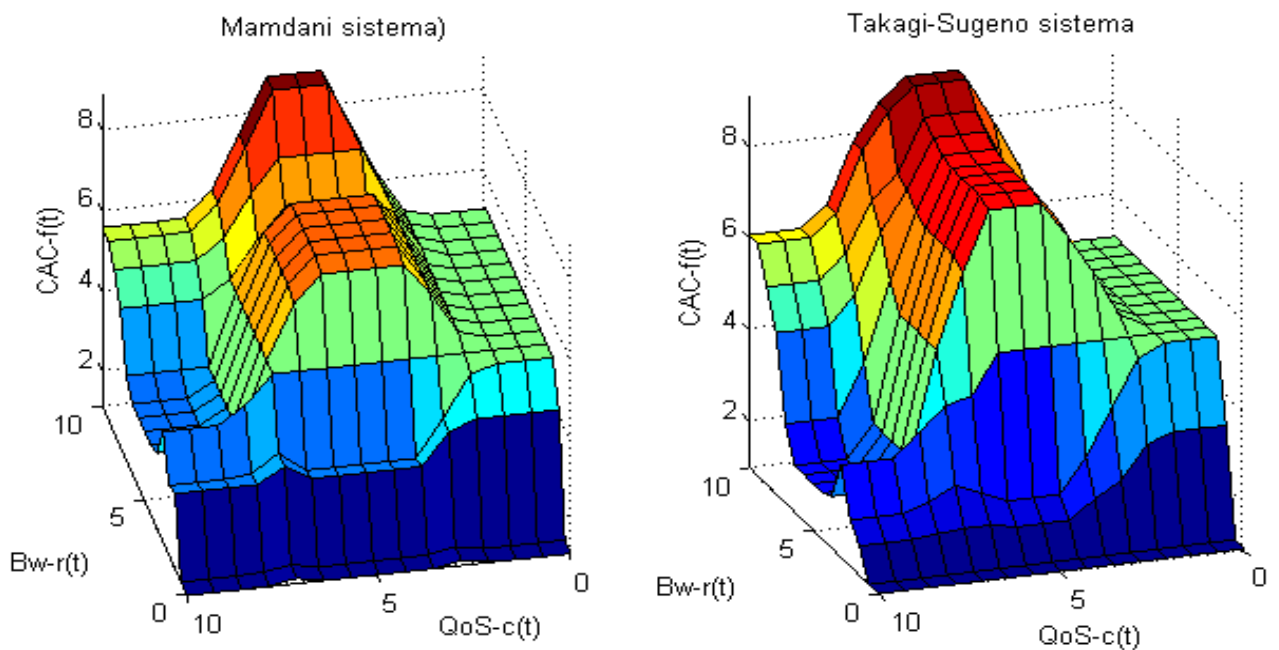
2.24. att. Fazi-CAC realizācijas ātrdarbības salīdzinājums 5000 lēmumiem *Mamdani* un *Takagi-Sugeno* FIS gadījumos

Kā redzams no mērījumu rezultātiem, tad konkrētās fazi-CAC algoritma fazi izvedumu sistēmas skaitļošanas laiks Matlab vidē, *Mamdani* un *Takagi-Sugeno* fazi izvedumu sistēmu gadījumos atšķiras tikai par  $\approx 3\%$ .

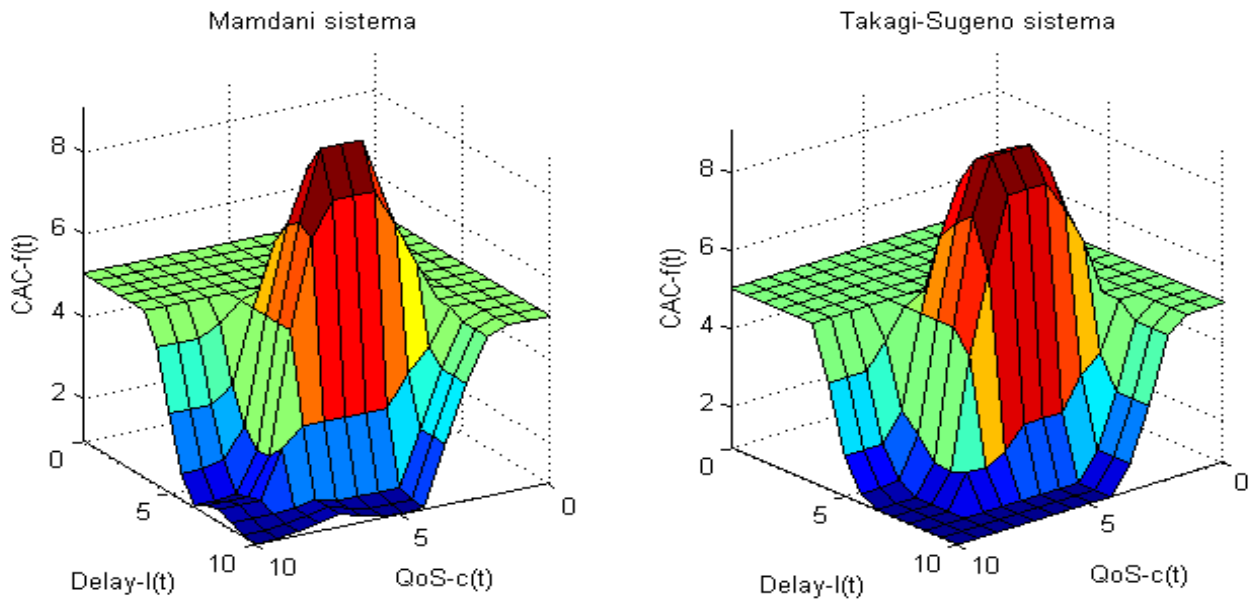
Attēlos 2.25, 2.26 un 2.27 ir doti fazi-CAC fazi izvedumu sistēmas lēmumu virsmu salīdzinoši attēli *Mamdani* un *Takagi-Sugeno* fazi izvedumu sistēmu gadījumā. Lēmumu virsmas ir gandrīz analogiskas, kas nozīmē, ka konkrētajā fazi-CAC realizācijā *Takagi-Sugeno* konstantās izvada parametra mainīgo piederības funkcijas ienesa minimālo ietekmi uz fazi izvedumu sistēmas lēmējloģiku.



2.25. att. Fazi-CAC lēmējmešanās lēmumu virsma  $Delay_l(t_i)$  un  $Bw_r(t_i)$  ieejas mainīgo vērtību gadījumā



2.26. att. Fazi-CAC lēmējmešanās lēmumu virsma  $QoS_c(t_i)$  un  $Bw_r(t_i)$  ieejas mainīgo vērtību gadījumā



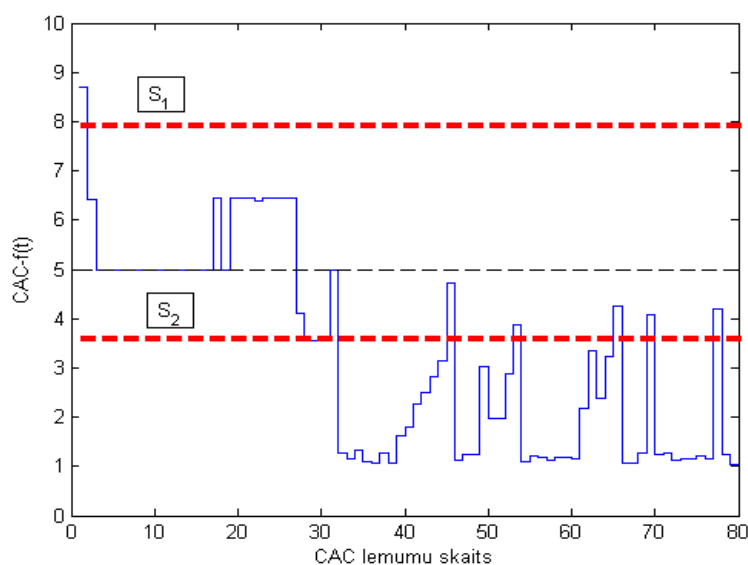
2.27. att. Fazi-CAC lēmējmehānisma lēmumu virsma  $QoS_c(t_i)$  un  $Delay_l(t_i)$  ieejas mainīgo vērtību gadījumā

Analizējot iegūtos rezultātus, tika pieņemts lēmums fazi-CAC lēmējsistēmas uzbūvē pielietot *Mamdani* tipa fazi interfeisu. Tas ir intuitīvi vienkāršāks, prasa apmēram tikpat skaitļošanas laika, un faktiski šajā konkrētajās fazi-CAC realizācijas gadījumā nodrošina analogiskus rezultātus pēc defazifikācijas, pielietojot tādu pašu ieejas parametru vērtību kopu un lingvinistisko mainīgo piederības funkciju deklarācijas. IF-THEN kārtulu zināšanu bāze *Takagi-Sugeno* fazi izvedumu sistēmas gadījumā tika ģenerēta, pielietojot *MATLAB*® tehniskās skaitļošanas vides *Fuzzy Logic Toolbox*<sup>TM</sup> bibliotēkas rīku „mam2fuz()” [36], kas rezultējošās *Takagi-Sugeno* fazi izvedumu sistēmas izvada parametru piederības funkcijas pārveido par konstantēm. Doto konstanšu vērtības tiek noteiktas, izskaitļojot laukuma smaguma centru zem oriģinālās *Mamdani* tipa fazi izvedumu sistēmas izvada parametru piederības funkcijām. IF-THEN kārtulu zināšanu bāzes noteikumi paliek nemainīgi.

### 2.7.10. Fazi-CAC lēmuma pēcapstrāde

Fazi-CAC atbildes  $CAC_f(t_i)$  vērtības pēcapstrāde nodrošina tās pārbaudi attiecībā uz sliekšņa vērtību, kas atšķir LSP pieslēguma atteikuma, atļaujas un atļaujas ar papildu resursu rezervēšanu.

Tā kā  $CAC_f(t_i)$  vērtība pēc defazifikācijas ir robežās no 0 līdz 10, ir jāizvēlas kādas noteiktas sliekšņa vērtības  $s_1$  un  $s_2$ , attiecībā uz kurām var nodrošināt  $CAC_f(t_i)$  pārbaudi un iegūt konkrētus CAC lēmumus (Att. 2.28)

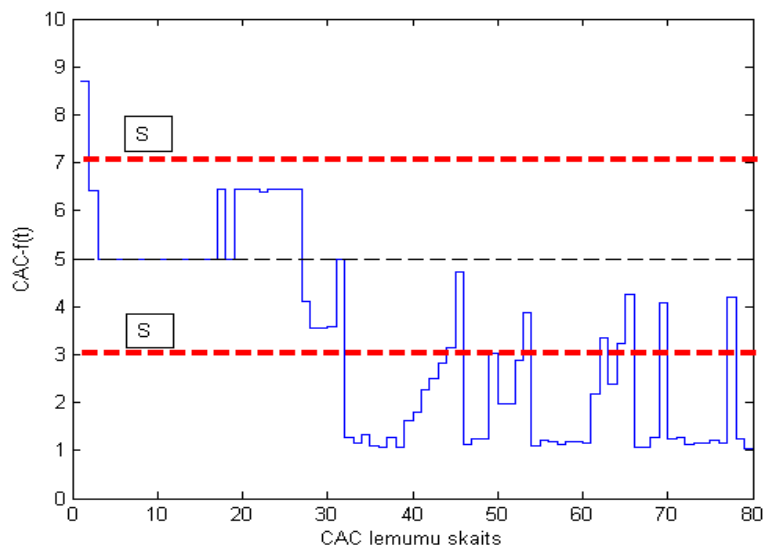


2.28. att. Sliekšņa vērtības  $CAC_f(t_i)$  pēcapstrādei

Fazi izvedumu sistēmas atbildes pēcapstrāde noteiktā veidā ierobežo  $CAC_f(t_i)$  sniegto informāciju par vēlamu CAC atbildi, tomēr tas ir nepieciešams, lai saņemtu precīzi definētus CAC lēmumus konkrēti definētu skaitļu veidā.

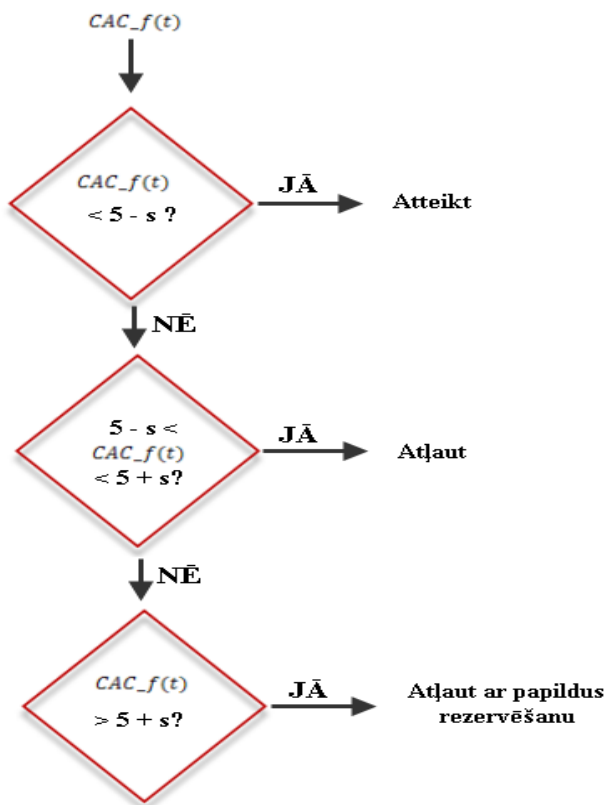
Mainot  $s_1$  un  $s_2$  vērtības, ir iespējams pielāgot fazi-CAC trafika vadības politiku, nodrošinot vēlamu LSP selektivitātes pakāpi. Atstājot pārāk lielu  $CAC_f(t_i)$  vērtību nogriezni CAC lēmumam „atteikt”, tiks nepilnīgi utilizēts links. Turpretim, atstājot pārāk lielu  $CAC_f(t_i)$  vērtību nogriezni CAC atbildei „atļaut”, links resursi tiks utilizēti labāk vienlaicīgi ar QoS parametru nepieļaujamu degradācijas pakāpi. Savukārt, atstājot pārāk lielu  $CAC_f(t_i)$  vērtību nogriezni CAC atbildei „atļaut ar papildu resursu rezervēšanu”, „drošības spilvena” īpatsvars ierobežotos linka resursos rezultātā dos nepamatoti zemu utilizētu linku.

Rezultātā, veicot fazi-CAC algoritma uzbūves izvēli atlasēs veidā, promocijas darba autors nonāca pie secinājuma, ka simetriskā sliekšņa vērtība  $s_1 = s_2 = s = 2$  (Att. 2.29) dod vēlamo fazi-CAC algoritma izvedumu sistēmas darbību.



2.29. att. Sliekšņa vērtības  $CAC_f(t_i)$  pēcapstrādei

Rezultātā tika izvēlēts šādas  $CAC_f(t_i)$  vērtības pēcapstrādes algoritms (Att. 2.30).



2.30. att. Fazi-CAC lēmēj mehānisma atbildes  $CAC_f(t_i)$  pēcapstrādes loģiskā shēma

Dotā fazi-CAC atbildes  $CAC_f(t_i)$  pēcstrāde faktiski dod papildu fazi-CAC algoritma tiešsaites modifikācijas iespējas, nemainot parametru piederības funkciju deklarācijas un IF-THEN kārtulu zināšanu bāzi. Pamainot sliekšņa vērtību, ir iespējams momentāni mainīt fazi izvedumu sistēmas vadības politiku, padarot to vairāk vai mazāk selektīvu attiecībā uz LSP savienojumu atteikumiem vai arī uz LSP savienojumu atļaujām ar papildu resursu rezervēšanu. Šī pieeja tiks sīkāk izdiskutēta nodaļā 4.2.

### 2.7.11. Papildu resursu rezervēšanas koeficienta vērtības izvēle

Šī promocijas darba autora piedāvātā fazi-CAC algoritma pamatā ir 3 CAC lēmumi:

- a.  $CAC_f(t_i) < 3$ , kas nosaka CAC lēmumu „Atteikt”;
- b.  $3 > CAC_f(t_i) < 7$ , kas nosaka CAC lēmumu „Atļaut”;
- c.  $CAC_f(t_i) > 7$ , kas nosaka CAC lēmumu „Atļaut ar papildu resursu rezervēšanu”.

Gadījumā, kad tiek pieņemts lēmums „Atļaut ar papildu resursu rezervēšanu”, LSP plūsmas pieprasījums tiek akceptēts, bet RSVP-TE piezervē vairāk resursu, nekā tiek norādīts aplikācijas pieprasījumā. Šādā veidā augstas QoS prasības kombinācijā ar zemām pakešu aizturēm un lielu  $Bw_r(t_i)$  vērtību rezultātā dod „drošības spilvenu”. Tas palīdz uzturēt pieprasītus QoS parametrus ne tikai no jauna pieslēgtām LSP plūsmām, bet arī visām linkā esošajām konекcijām. Turklāt dotais „drošības spilvens” ir tieši proporcionāls augstas prioritātes aktīvām LSP plūsmām linkā.

Ir svarīgi izvēlēties rezervēšanas koeficientu  $k$ , kas nodrošinātu fazi-CAC vēlamu efektivitāti. Heiristiskās atlases, kļūdu un mēģinājumu metodes rezultātā tika izvēlēta vērtība  $k = 2$ . Tas nozīmē, ka lēmuma „Atļaut ar papildu resursu rezervēšanu” gadījumā RSVP-TE rezervēs 2 reizes vairāk resursu linkā, nekā ir prasīts no aplikācijas puses.

### **3. FAZI-CAC RISINĀJUMA EKSPERIMENTĀLĀ PĀRBAUDE**

#### **Nodaļas kopsavilkums**

Šajā nodaļā tiks sniegts MPLS-TE sakaru sistēmas eksperimentālā tīkla realizācijas detalizēts apraksts, kas ietver sevī gan MPLS-TE tīkla daļu, gan fazi-CAC algoritma vadības bloku, kā arī trafika ģenerējošo un uztverošo pusi. Tiks doti eksperimentu scenāriju apraksti un analizējamo QoS parametru definīcijas.

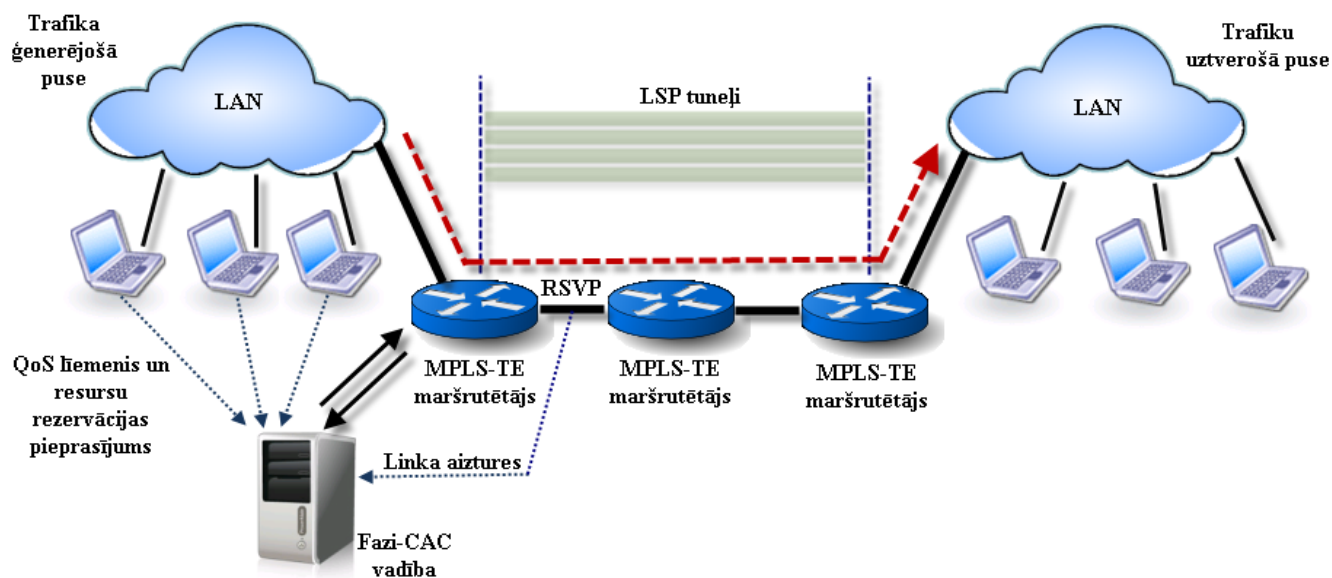
Rezultātā tiks detalizēti izklāstīti eksperimentu rezultāti, salīdzinot šī promocijas darba autora piedāvātā fazi-CAC algoritma darbības efektivitāti ar klasisko sliekšņa CAC algoritmu.

#### **3.1. Praktiskā fazi-CAC algoritma realizācija**

Fazi-CAC algoritma darbības efektivitātes pārbaudei, salīdzinot to ar klasisko sliekšņa CAC algoritmu, tika veikta praktiskā fazi-CAC algoritma darbības realizācija uz reālā maršrutēšanas aprīkojuma. Šī uzdevuma izpildei tika izveidots MPLS-TE sakaru sistēmas eksperimentāls tīkls un nepieciešamais programmatiskais nodrošinājums, kas ļāva praktiski pārliecināties par autora piedāvātā fazi-CAC risinājuma darbības efektivitāti un iegūt nepieciešamos QoS tīkla parametrus raksturojošos datus. Iegūtā informācija ļāva veikt pētāmā eksperimentālā MPLS-TE tīkla elementu veiktspējas analīzi un pārliecināties par piedāvātā algoritma tīkla darbības uzlabojumiem pakešu un plūsmu līmenī.

##### **3.1.1. Praktiskās fazi-CAC realizācijas MPLS-TE eksperimentālā tīkla apraksts**

MPLS-TE sakaru sistēmas eksperimentālā tīkla realizācijai tika izvēlēti RTU, Elektronikas un Telekomunikāciju fakultātes, Telekomunikāciju institūta, maršrutēšanas laboratorijas aprīkojumā 3 esošie CISCO 2800 sērijas maršrutētāji, kas dod iespēju veidot un analizēt nelielu MPLS-TE tīklu. Eksperimentālie dati par šī tīkla robežās iestatīto un atteikto LSP konekcijām, RSVP protokola rezervētiem resursiem un pamata trafika QoS parametru izmaiņām, ejot caur testa tīkla segmentiem, dod iespēju izdarīt svarīgākos secinājumus par fazi-CAC algoritma efektivitāti salīdzinājumā ar klasisko sliekšņa CAC algoritmu.

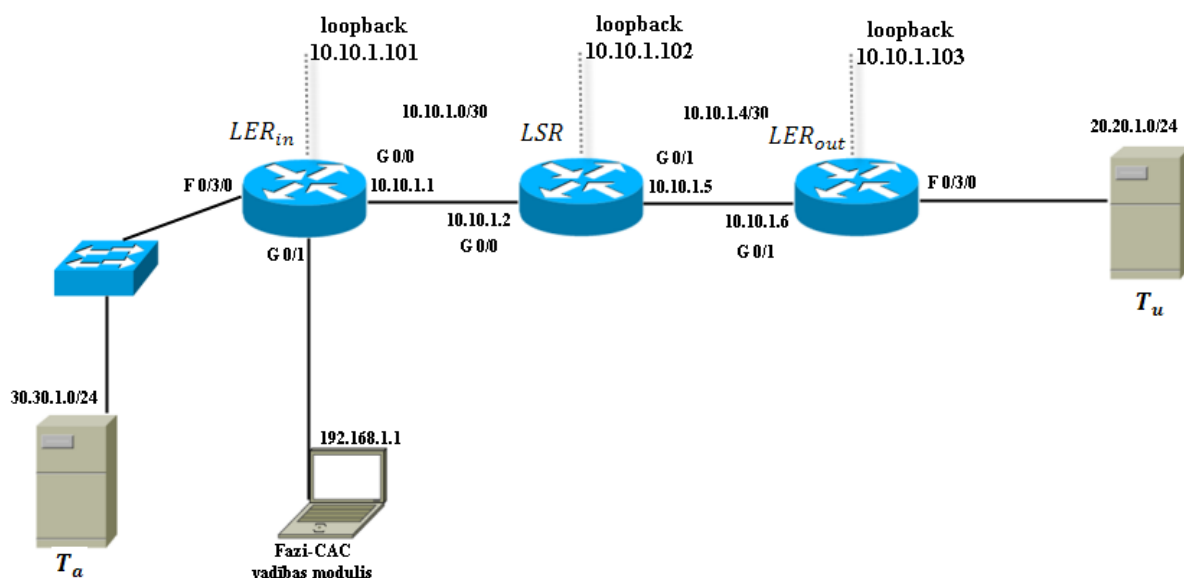


3.1. att. Fazi-CAC algoritma pārbaudes eksperimentālā maketa loģiskā slēguma shēma

Attēlā 3.1 ir parādīta fazi-CAC algoritma pārbaudes MPLS-TE sakaru sistēmas eksperimentālā tīkla shēma. Tīkls sastāv no

- trafika ģenerējošās puses, kas fiziski ir balstīta uz viena Dell PowerEdge SC 1433 servera;
- trafika uztverošās puses, kas fiziski ir balstīta uz viena Dell PowerEdge SC 1433 servera;
- trim CISCO 2800 sērijas maršrutētājiem, kas izpilda  $LER_{in}$ ,  $LSR$  un  $LER_{out}$  funkcijas;
- fazi-CAC vadības bloka, kas fiziski sastāv no Dell Latitude E6500 datora.

Fiziskā fazi-CAC algoritma pārbaudes MPLS-TE sakaru sistēmas eksperimentālā maketa shēma ir attēlota att. 3.2.

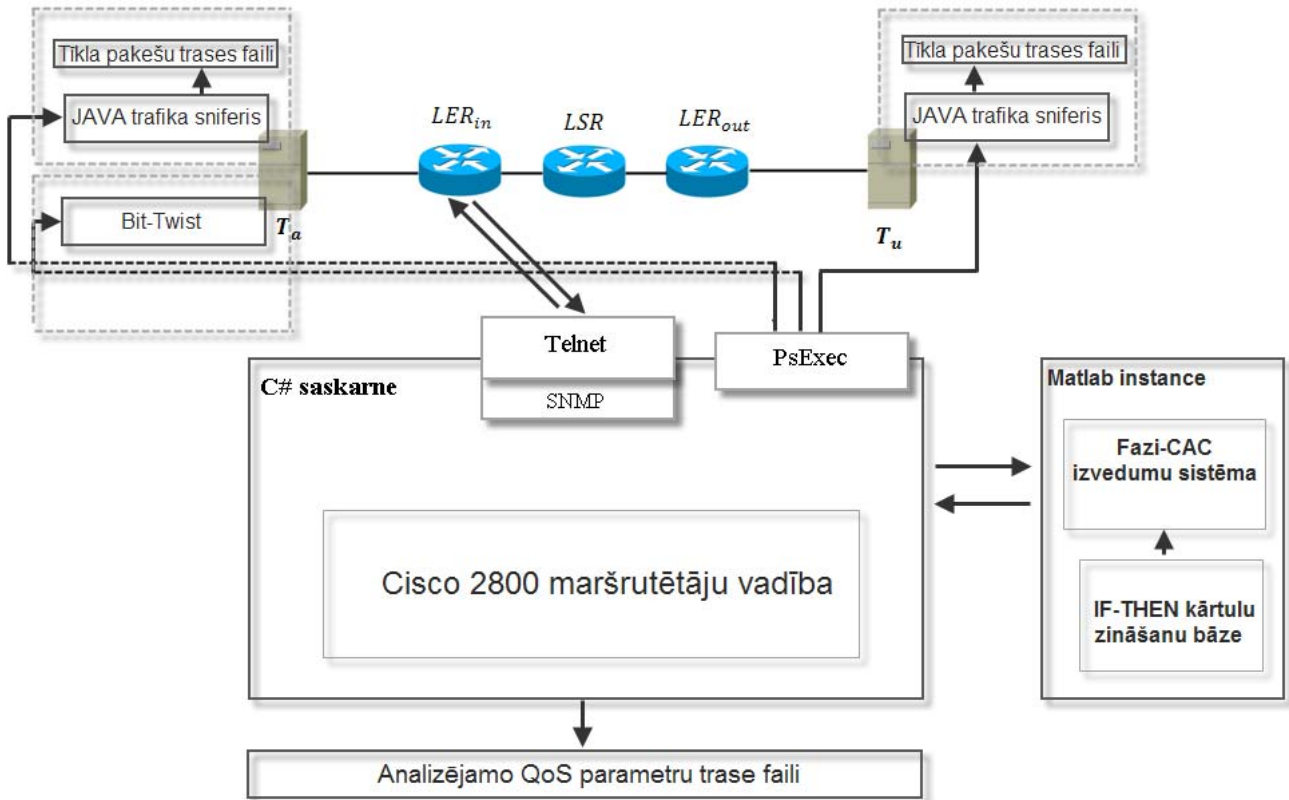


3.2. att. Fazi-CAC algoritma pārbaudes eksperimentālā maketa loģiskā slēguma shēma

No  $LER_{in}$  uz  $LER_{out}$ , pielietojot fazi-CAC algoritmu un klasisko sliekšņa CAC algoritmu, tiek iestatīti LSP tuneļi, kā arī tiek nodrošināta ģenerēto datu plūsmu transportēšana no trafika ģenerējošās puses  $T_a$  uz trafika uztverošu pusi  $T_u$ .

Eksperimentālā tīkla fazi-CAC vadības programmatiskās realizācijas shēma ir parādīta attēlā 3.3. Galvenais fazi-CAC vadības bloks eksperimentālā tīkla uzbūvē ir uz C# bāzes veidota saskarne ar CISCO 2800 sērijas maršrutētāju, kas tiek realizēta ar Telnet protokolu. Saskarnes pirmkods ir pieejams pielikumā Nr. 2. Tas var kalpot par pamatu arī turpmākiem CAC vadības pētījumiem eksperimentālajā MPLS-TE tīklā, nodrošinot vadības saskarni ar MPLS-TE maršrutētāju ar plašām modifikācijas spējām.

CISCO 2800 sērijas maršrutētājs strādāja MPLS-TE režīmā. Maršrutētājs varēja saņemt vadības komandas, piemēram, LSP tuneļa iestatīšanas komandu, un arī atgriezt visu CAC algoritma darbībai un tā efektivitātes pārbaudei nepieciešamo informāciju, piemēram, aktīvo un neaktīvo LSP tuneļu sarakstu, maršrutētāja bufera stāvokļa datus, informāciju par aizturēm linkā, zaudēto pakešu skaitu, informācijas par pakešu aizturu trīci utt.



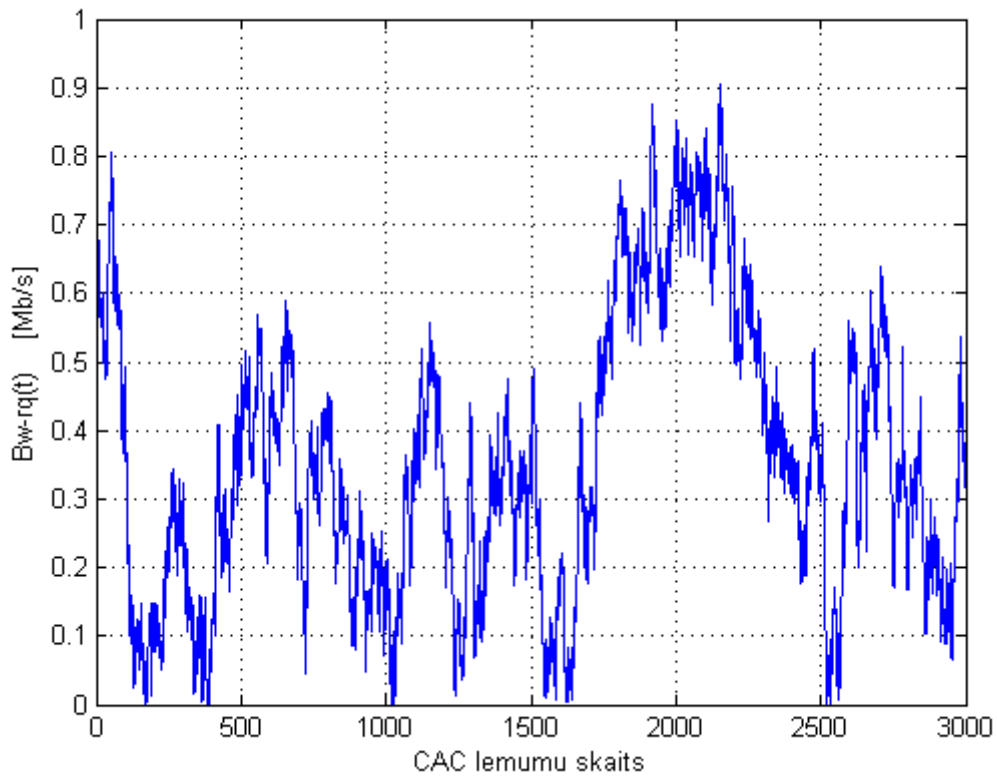
3.3. att. Fazi-CAC algoritma pārbaudes MPLS-TE sakaru sistēmas eksperimentālā tīkla programmatiskās realizācijas elementu slēguma shēma

Pats fazi-CAC algoritms tika realizēts, pielietojot *Fuzzy Logic Toolbox*<sup>TM</sup> bibliotēku, kas ir *MATLAB*® tehniskās skaitļošanas vides rīka sastāvdaļa. Pielietotā FIS fazi interfeisa Matlab kods ir pieejams pielikumā Nr. 2.

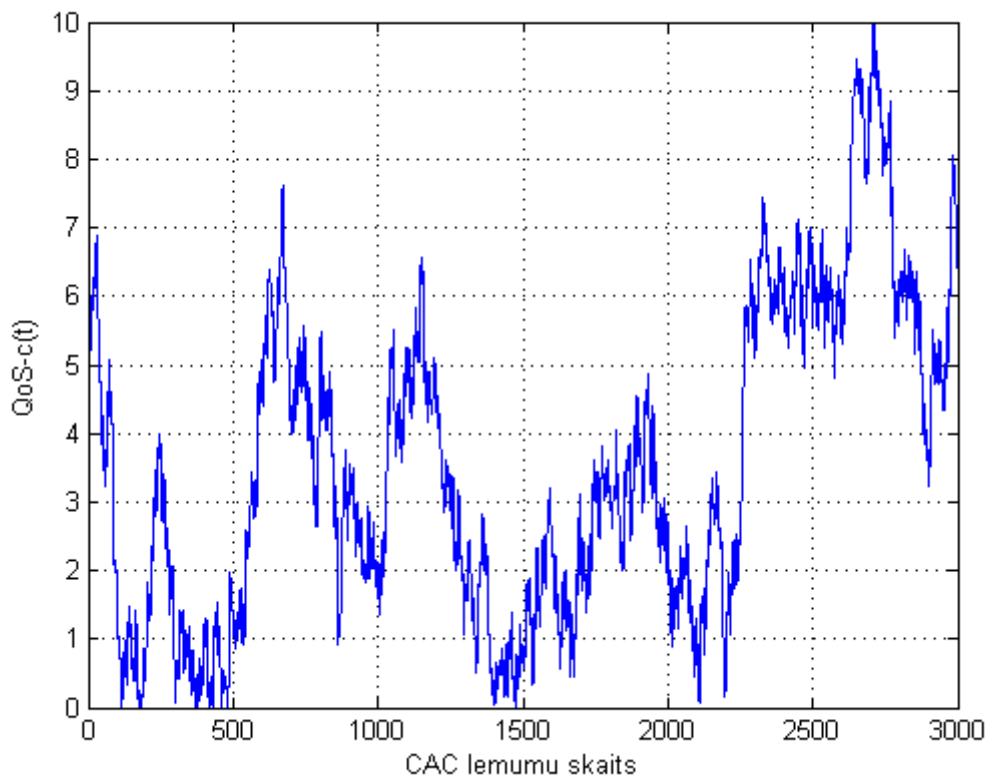
Saskarne ar Matlab bibliotēkām tika nodrošināta, pielietojot *Emanuele Ruffaldi* izstrādāto [89], [88] *Matnet* bibliotēku. Galvenais solis *Matnet* bibliotēkas izmantošanā saiknes nodrošināšanai ar Matlab ir *EngMATAccess* klases objekta izveide, kas atver komunikācijas kanālu ar Matlab instanci, bet ja tāda neeksistē, tad tā tiek izveidota. Faktiski tiek nodrošināts *Platform Invocation Services* (jeb *P/Invoke*) izsaukums, kas garantē piekļuvi pie Matlab bibliotēkām. Šajā realizācijā doto risinājumu var pielīdzināt virtuālam Matlab serverim, kas dod iespēju izsaukt Matlab instanci, nodot tai metodes izsaukumu un saņemt atpakaļ izskaitļotu rezultātu.

Matlab vaicājuma izveidei ir vajadzīga datu kopa, kas sastāv no:

- a. pieprasītās plūsmas lieluma –  $Bw_{rq}(t_i)$  [Mb/s] – fBm ģenerētie dati (att. 3.4)
- b. QoS klases (jeb līmeņa) pieprasījuma –  $QoS_c(t_i)$  – fBm ģenerētie dati (att. 3.5)
- c. aiztures tīklā -  $Delay_l(t_i)$  [ms] – mērījumu dati.



3.4. att. Fazi-CAC algoritma ieejas parametra  $Bw\_rq(t_i)$  fBm ģenerēto datu vektors



3.5. att. Fazi-CAC algoritma ieejas parametra  $QoS\_c(t_i)$  fBm ģenerēto datu vektors

Lai iegūtu salīdzināmus rezultātus, pieprasītās plūsmas lieluma dati tika ģenerēti ar fBm ģeneratoru [8]. Tādi paši dati tika pielietoti fazi-CAC algoritma pārbaudes laikā MPLS-TE sakaru sistēmas simulācijas realizācijā. Tas deva iespēju, pielietojot analogiskas plūsmu pieprasījuma vērtības, efektīvi salīdzināt fazi-CAC ar klasisko sliekšņa CAC, pielietojot vienādas ieejas vērtību kopas. Arī QoS līmeņa ģenerācija notiek, pielietojot to pašu fBm ģeneratoru, un tas tiek ievadīts Matlab vidē realizētā lēmuma pieņemšanas algoritmā kopā ar ģenerēto plūsmas lielumu un nomērīto pakešu aizturi linkā.

Brīdī, kad tiek palaists fazi-CAC vai klasiskais sliekšņa CAC algoritms, tiek aktivētas divu tipu aplikācijas. Pirmā ir trafika plūsmu ģenerācijas programma Bit-Twist [12], kas CAC algoritma pozitīvās atbildes gadījumā palaiž tīklā uz trafika uztverošo pusi pakešu plūsmu ar ātrumu, kāds tika pieprasīts no trafika ģenerējošās puses. Par trafika plūsmas pamatu kalpo FTP sesijas pakešu plūsmas pieprasījuma ierakstīts segments 50MB izmērā, kas tiek nepārtraukti nolasīts un palaists tīklā ar definētu datu caurplūdes ātrumu. Bit-Twist trafika ģenerācijas aplikācijas tiek palaistas, pielietojot PsExec [91] jeb Telnet līdzīgu nelielu utilītu, kas ļauj ērti palaist procesus uz attālinātām sistēmām. PsExec tika pielietots arī, lai vienlaicīgi ieslēgtu otru aplikāciju tipu – uz JAVA tehnoloģijas bāzes izveidotus trafika sniferus, kas ļauj iegūt trafika caurplūdes datus un pakešu atnākšanas laikus analizējamajās trafika plūsmās gan trafika ģenerējošā, gan arī trafika uztverošā pusē.

Rezultātā, fazi-CAC algoritma praktiskās realizācijas eksperimentā tiek savākti šādi dati par MPLS-TE sakaru sistēmas darbību.

1. Trafika ģenerējošās un uztverošās puses nosūtīto un uztverto pakešu izmēri un nosūtīšanas/atnākšanas laiki.
2. Lēmuma pieņemšanas laika moments un tā skaits pēc kārtas.
3. Pieejamā datu caurlaides josla CAC lēmuma pieņemšanas momentā  $Bw_{aw}(t_i)$ .
4. Pieprasītās datu caurlaides joslas lielums CAC lēmuma pieņemšanas momentā  $Bw_{rq}(t_i)$ .
5. Pieprasītais QoS līmenis CAC lēmuma pieņemšanas momentā  $QoS_c(t_i)$ .
6. Maršrutētāja bufera stāvoklis CAC lēmuma pieņemšanas momentā.

7. Vidējās pakešu aiztures starp  $LER_{in}$  un  $LER_{out}$  CAC starp lēmuma pieņemšanas momentiem  $Delay_l(t_i)$
8. Aktīvo LSP tuneļu skaits CAC lēmuma pieņemšanas momentā.
9. Slēgto LSP tuneļu skaits CAC lēmuma pieņemšanas momentā.
10. CAC algoritma atbilde CAC lēmuma pieņemšanas momentā  $CAC_f(t_i)$
11. Pakešu zudumi  $LER_{in}$  maršrutētājā starp CAC lēmuma pieņemšanas momentiem  $Drops_l(t_i)$ .

### 3.1.2. Eksperimenta gaita

Šinī nodaļā promocijas darba autors īsi izklāsta eksperimentu gaitas notikumu secību MPLS-TE sakaru sistēmas eksperimentālā tīkla realizācijā, pielietojot klasisko un fazi-CAC algoritmus. Šajā nodaļā arī tiek pamatota laika nogriežņa izvēle starp CAC lēmumu pieņemšanas momentiem programmatiskajā eksperimenta realizācijā.

Eksperimentu gaita tika sadalīta divās pamata daļās.

1. MPLS-TE tīkla veiktspējas analīze ar fazi-CAC algoritma realizāciju.
2. MPLS-TE tīkla veiktspējas analīze ar klasisko sliekšņa-CAC algoritma realizāciju.

Abos gadījumos tika pielietoti identiski ieejas parametru vektori, kas tika definēti nodaļā 2.2.2 :

- a. pieprasītās plūsmas lielums –  $Bw_rq(t_i)$  [Mb/s] ;
- b. QoS klases (jeb līmeņa) pieprasījums –  $QoS_c(t_i)$  ;
- c. aiztures tīklā –  $Delay_l(t_i)$  [ms] .

Pirmajā gadījumā fazi-CAC algoritms pielietoja visus iepriekš minētos ieejas parametru vektoru mainīgos, savukārt otrajā gadījumā, klasiskā sliekšņa-CAC algoritma darbības laikā, tika pielietoti vienīgi  $Bw_rq(t_i)$  dati.  $Delay_l(t_i)$ , kā arī  $QoS_c(t_i)$  datu vektori netika pielietoti, bet to vērtības tika saglabātas trases failos ar nolūku veikt salīdzinošu algoritmu analīzi.

Gan fazi-CAC, gan klasiskā sliekšņa-CAC algoritma realizācijas eksperimentālās MPLS-TE sakaru sistēmas darbības laikā, tika savākta statistika par tīkla QoS parametriem (skat. nodaļu 3.1.1)

Fazi-CAC lēmuma pieņemšanas momentā notiek vairākas darbības, kas prasa noteiktu skaitļošanas laiku.

- a. Telnet sesijas atvēršana ar nolūku saņemt datus par aktuālo  $LER_{in}$  stāvokli RSVP-TE rezervēto resursu ziņā, kā arī SLA datiem par QoS parametriem starp  $LER_{in}$  un  $LER_{out}$ .
- b. P/Invoke izsaukums, kas nodrošina piekļuvi pie Matlab bibliotēkām.
- c. Faziloģikas CAC atbildes kalkulācija, pielietojot Matlab bibliotēku.
- d. Telnet sesijas atvēršana ar nolūku konfigurēt  $LER_{in}$  un veidot LSP konekciju starp  $LER_{in}$  un  $LER_{out}$ .
- e. PsExec izsaukums un Bit-Twist aplikācijas palaišana uz attālinātās klienta mašīnas ar nolūku veidot reālo trafika plūsmu starp  $LER_{in}$  un  $LER_{out}$  izveidoto LSP konekciju.

Sliekšņa CAC lēmuma pieņemšanas momentā notiek šādas darbības:

- a. *Telnet* sesijas atvēršana ar nolūku saņemt datus par aktuālo  $LER_{in}$  stāvokli RSVP-TE rezervēto resursu ziņā, kā arī SLA datiem par QoS parametriem starp  $LER_{in}$  un  $LER_{out}$ ;
- b. sliekšņa CAC atbildes kalkulācija C# saskarnes modulī.
- c. *Telnet* sesijas atvēršana ar nolūku konfigurēt  $LER_{in}$  un veidot LSP konekciju starp  $LER_{in}$  un  $LER_{out}$ .
- d. *PsExec* izsaukums un *Bit-Twist* aplikācijas palaišana uz attālinātās klienta mašīnas ar nolūku veidot reālo trafika plūsmu starp  $LER_{in}$  un  $LER_{out}$  izveidoto LSP konekciju.

Rezultātā rodas nepieciešamība izvēlēties laika intervālu starp CAC lēmumiem MPLS-TE sakaru sistēmas eksperimentālajā tīklā. Laika intervāls starp CAC lēmumu pieņemšanas momentiem tika izvēlēts empīriski un sastāda 4 sekundes. Tas ļauj nodrošināt eksperimentālā MPLS-TE sakaru sistēmas tīkla CAC vadības programmatiskās realizācijas stabilu darbību, kad visas skaitļošanas darbības un attālinātu aplikāciju izsaukumi notiek

pilnīgā apmērā līdz jaunas LSP plūsmas pieslēguma pieprasījumam un nākamajam CAC lēmuma vaicājumam.

Analizējot fazi-CAC algoritma veikspēju simulācijas laikā, akcents tika likts uz resursu rezervēšanas funkciju, ko veic RSVP-TE protokols. Praktiskajā realizācijā, algoritma darbības efektivitātes kritēriju starpā tiek pārbaudīti ne tikai resursu rezervēšanas dati, bet arī tīkla veikspējas parametri pakešu līmenī, tādi kā pakešu aiztures, pakešu aizturu trīce, pakešu zudumi tīklā, maršrutētāja bufera stāvoklis laikā un reālā linka utilizācija.

Jauno plūsmu pieslēguma pieprasījumi ir 3 reizes biežāki par pieslēgto plūsmu deaktivēšanas gadījumiem, kas paredz, ka links visu laiku atradīsies pārslodzes režīmā. Tieši šādā stāvoklī links ir visvairāk atkarīgs no korektas trafika vadības politikas un tās efektīvas īstenošanas gadījumā ir spējīgs nodrošināt stabilu trafika transportu, minimizējot pakešu zudumus, aiztures un trīci, un nodrošināt selektīvu trafika plūsmu pieslēgšanu.

### 3.1.3. Tīkla QoS parametru mērījumi ar Cisco IOS IP SLA

MPLS-TE sakaru sistēmas eksperimentālā tīkla realizācijā vairums nepieciešamo QoS parametru sakaru sistēmas darba efektivitātes novērtējumam tika iegūti, pielietojot *Cisco IOS IP Service Level Agreements* [27]. Cisco IOS SLA ļāva saņemt uzticamus mērījuma datus par tīkla vājā posma starp  $LER_{in}$  un  $LER_{out}$  QoS parametriem, kas tika izmantoti fazi-CAC algoritma darbības nodrošināšanai un MPLS-TE sakaru sistēmas eksperimentālā tīkla darba monitoringam.

Cisco IOS IP SLA ir pamatdaļa no Cisco IOS programmatiskās bāzes, kas dod iespēju nodrošināt pastāvīgu dažādu servisa kvalitātes kritēriju monitoringu IP līmenī. Cisco IOS IP SLA lieto aktīvo trafika monitoringu – pētāmajā tīkla nogrieznī tiek ģenerēts trafiks, kas tiek tālāk izmantots tīkla QoS parametru analīzei.

Cisco IOS IP SLA var tikt izsaukts, pielietojot Cisco IOS CLI vai arī SNMP, caur RTTMON un SYSLOG MIB.

Attēlā 3.6 var redzēt, ka Cisco IOS IP SLA pielieto divas IP SLA puses:

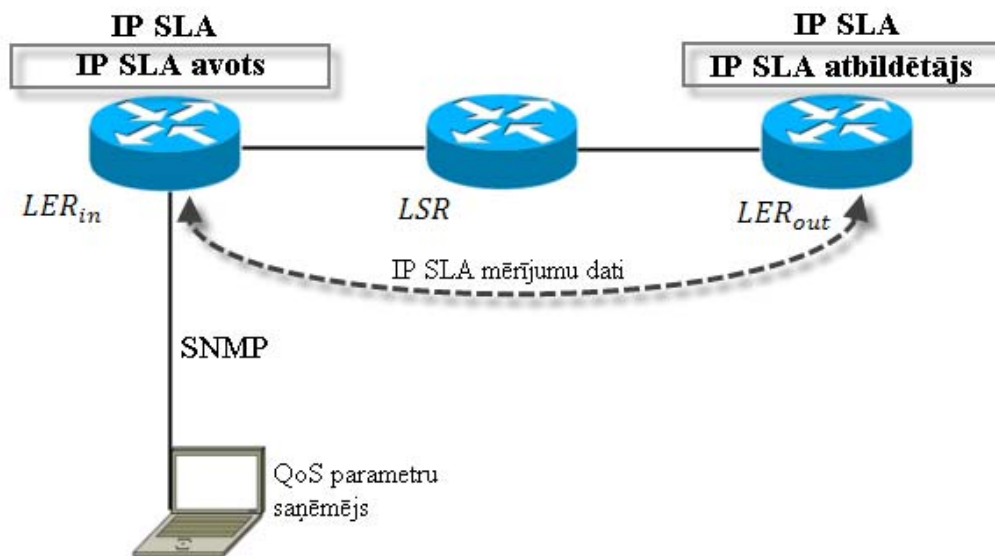
- a. IP SLA avots;
- b. IP SLA atbildētājs [angl. *responder*]

IP SLA atbalstošs maršrutētājs, kas satur IP SLA avota daļu, sūta ģenerētās paketes uztverošajam maršrutētājam, kas satur IP SLA atbildētāja daļu, un atkarībā no IP SLA operācijas tipa atbildētājs atgriezīs laikspiedolu, ko IP SLA avota daļa izmantos attiecīgā QoS parametra kalkulācijai. Mērīšanas operācijām IP SLA pielietoja UDP paketes.

Cisco IOS IP SLA MPLS-TE, MPLS-TE eksperimentālā tīkla ietvaros, bija nokonfigurēts ar nolūku savākt tiešsaistes datus par šādiem QoS parametriem:

- pakešu aiztures starp  $LER_{in}$  un  $LER_{out}$ ;
- pakešu aizturu trīce;
- pakešu zudumi  $LER_{in}$  maršrutētājā.

Informācija par Cisco IOS IP SLA mērījumu metodoloģiju un mērījumu precizitāti ir atrodamā turpmāk dotajos literatūras avotos [29] [26] [108].



3.6. att. Cisco IP SLA mērījumu shēma QoS parametriem uz MPLS-TE sakaru sistēmas eksperimentālā tīkla

### 3.1.4. Analizējamo parametru apskats

Eksperimentālās daļas rezultātu aprakstā tiks operēts ar vairākiem tīkla veikspējas parametriem, kas definē autora piedāvātā algoritma efektivitātes pamatojumu salīdzinājumā ar klasisko sliekšņa-CAC algoritmu. Šajā īsajā sadaļā autors pamatos pētāmo parametru izvēli.

Definējot parametrus, kas būtu nepieciešami analīzei, lai pārbaudītu šī promocijas darba autora piedāvāto fazi-CAC algoritma efektivitāti, ir jāizvērtē pamata vērtības, kas viennozīmīgi parāda tīkla segmentu vai tā atsevišķu elementu darbības efektivitāti.

#### 1. RSVP-TE rezervētie resursi starp $LER_{in}$ un $LER_{out}$

Pirmām kārtām ir svarīgi sekot līdzi RSVP-TE rezervētiem resursiem, kas tiek izdalīti iestatītiem LSP tuneļiem. Šī informācija nenorāda tieši uz trafika parametru izmaiņu vai stabilitāti pakešu līmenī, toties dod lielisku iespēju sekot līdzi linka stāvokļa izmaiņu tendencēm un ļauj pielietot informāciju par rezervētiem linka resursiem vadības uzdevumos, tādās kā šī pētījumā apskatītā CAC kontrole. Tātad kā pirmais analizējamais parametrs tiks apskatīts RSVP-TE rezervēto resursu diskrēto vērtību vektors visā lēmumu pieņemšanas laikā. Jāatzīmē, ka RSVP-TE rezervēto resursu apjoms tiks attēlots bez piesaistes pie reālā laika, bet gan nosakot vērtības diskrētos CAC lēmumu pieņemšanas momentos. Laika vektors šī parametra analīzē nedotu nekādu svarīgu informāciju, jo autora interese šajā jautājumā aprobežojas ar izdalīto resursu sadali analogiskajā lēmumu pieņemšanas skaita ziņā gan klasiskā sliekšņa-CAC gadījumā, gan pielietojot autora piedāvāto fazi-CAC algoritmu. Turklāt promocijas darba autora pielietotais MPLS-TE sakaru sistēmas eksperimentālais tīkls pieņem CAC lēmumu un pieslēdz jaunas LSP plūsmas ar strikti definēto laika intervālu, kura izvēles pamatojums ir dots nodaļā 3.1.2.

#### 1. Pakešu zudumi $LER_{in}$ maršrutētājā

Otrs parametrs, kas dod iespēju kvantitatīvi novērtēt dotā promocijas darbā pētāmo fazi-CAC algoritma efektivitāti, ir pakešu zudumi  $LER_{in}$  maršrutētājā. Abstrahējoties no citiem mazticamiem pakešu zudumu iemesliem linkā, tādiem kā, piemēram, signāla kropļojumi fiziskajā slānī, kā pamata problēma ir jāizvirza bufera pārpildījuma problēma. Autors pielietoja CISCO 2800 sērijas maršrutētājus, kas pēc noklusējuma bija konfigurēti ar 40

pakešu lielu ietiepības izejas buferi. Eksperimentu rezultāti attēlo vidējos pakešu zudumus starp CAC algoritma lēmuma pieņemšanas momentiem, kā arī vidējo pakešu zudumu vērtību visā eksperimenta laikā. Pakešu zudumu dati tika saņemti no  $LER_{in}$  maršrutētāja monitoringa datiem.

## **2. $LER_{in}$ bufera aizpildījums**

Kā nākamo parametru autors izvēlējās maršrutētāja bufera stāvokli. Kā tas jau tika minēts iepriekš, pētījumā pielietotais CISCO 2800 sērijas maršrutētājs pēc noklusējuma konfigurēts ar 40 pakešu izvada buferi un tā momentānās un vidējās vērtības tika analizētas CAC algoritma lēmuma pieņemšanas brīžos. Bufera aizpildījuma dati tika saņemti no  $LER_{in}$  maršrutētāja monitoringa datiem.

## **3. Pakešu aiztures starp $LER_{in}$ un $LER_{out}$**

Pakešu aiztures  $Delay_l(t_i)$  tika izvēlētas kā nākamais analizējamais parametrs. Tika izmantota CISCO IOS IP SLA sniegtā informācija par aizturēm starp maršrutētājiem  $LER_{in}$  un  $LER_{out}$  pētāmajā MPLS-TE sakaru sistēmas eksperimentālā maketa tīklā. Fazi-CAC algoritma darbībai tika pielietoti dati par vidējo pakešu aizturu vērtību starp CAC lēmuma pieņemšanas momentiem. Tā kā LAN daļas trafika ģenerējošā un uztverošā pusē bija aprīkotas ar 100 Mb/s interfeisiem, tad aiztures tur bija niecīgas un netika ņemtas vērā. CISCO IOS IP SLA ziņas par momentānām un vidējām aizturēm starp maršrutētājiem tika analizētas CAC algoritma lēmuma pieņemšanas laikā.

## **4. Pakešu aizturu starp $LER_{in}$ un $LER_{out}$ trīce**

Mērot pakešu aizturu trīci jeb  $Jitter_l(t_i)$ , kuras vērtības tika nolasītas no Cisco IOS IP SLA, faktiski tika pielietoti dati par vidējo trīces vērtību starp CAC lēmuma pieņemšanas momentiem. Trīces analīze paralēli notika, arī balstoties uz  $T_u$  un  $T_a$  pusēs savāktiem datiem par nosūtītajām un pienākošajām paketēm, jo trafika analizatori, kas atradās trafika ģenerējošajā un uztverošajā pusē deva iespēju fiksēt katras nosūtītās un saņemtās paketes

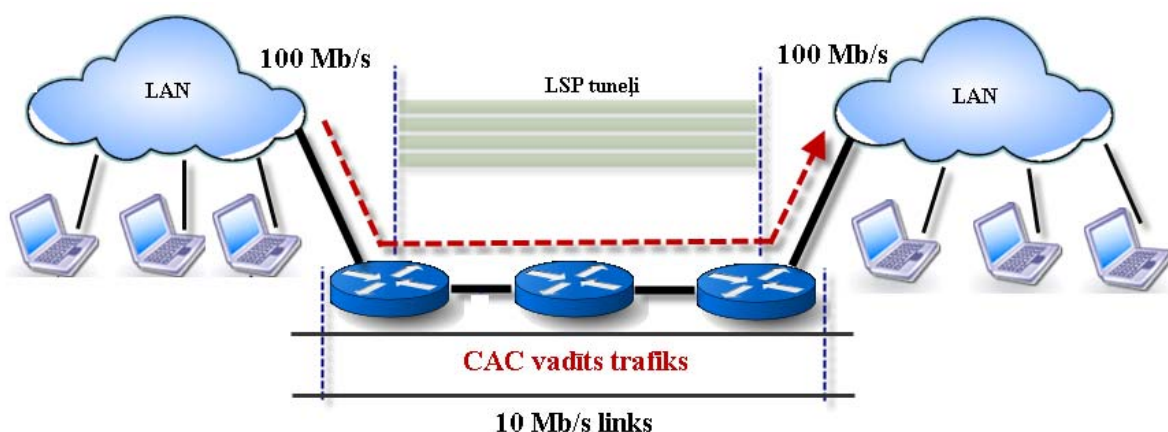
nosūtīšanas un pienākšanas laikus, sekojot līdzī trafika caurplūdei tīklā un pakešu starpatnākšanas laiku izmaiņām  $T_u$  un  $T_a$  datu saskarnēs.

## 5. Dati par pieslēgtām un atteiktām LSP plūsmām

Ekspierimenta laikā tika savākti dati par pieslēgtām un atteiktām LSP plūsmām CAC lēmumu pieņemšanas momentos, kā arī par plūsmām, kas tika pieslēgtas ar papildu resursu rezervēšanu. Katras pieslēgtās vai atteiktās LSP plūsmas gadījumā dati par fazi-CAC fazi izvedumu sistēmas ieejas un izejas parametru vērtībām tika saglabāti trases failā turpmākai analīzei.

### 3.1.5. Ekspierimentu scenāriju apraksts

Fazi-CAC algoritma praktiskās realizācijas pārbaude tika veikta, definējot divus pamata scenārijus. Pirmais jeb optimistiskais scenārijs izslēdza uzliesmojošā fona trafika klātbūtni un paredzēja tikai un vienīgi pieprasīto plūsmu pieslēgšanu pie sākotnēji tukša 10Mb/s kanāla, kas tika papildīts, balstoties tikai un vienīgi uz fazi-CAC vai arī klasiskā sliekšņa CAC algoritma sniegtajiem vērtējumiem par plūsmu pieslēgšanu vai noraidīšanu (sk. att. 3.7).

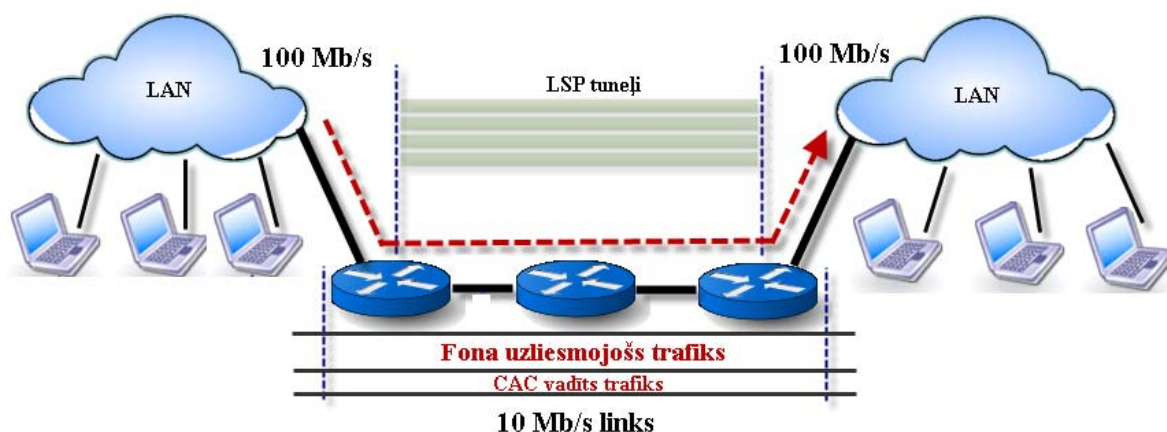


3.7. att. Ekspierimenta optimistiskā scenārija linka papildīšanas politikas shēma

Otrais jeb pesimistiskais scenārijs paredzēja fazi-CAC un klasiskā CAC algoritmu darbības efektivitātes analīzi, pielietojot 10Mb/s linku, kas tika piepildīts vidēji ar 6 Mb/s uzliesmojošo datu plūsmu. Trafika plūsmas parametri:

- vidējais datu pārraides ātrums – 6Mb/s;
- maksimālais uzliesmojuma apjoms – +/- 3Mb/s ;
- uzliesmojumu ilgums  $\approx 40\%$  no pārraides laika.

Šajā gadījumā uz CAC algoritma darbības pamata tika vadīta paliekošās 4 Mb/s caurplūdes joslas utilizācija, kas rezultātā deva iespēju izvērtēt fazi-CAC un klasiskā CAC algoritmu darbību pie kritiskiem apstākļiem, kad pakešu zudumi maršrutētājā ir ievērojami un maršrutētāja buferis lielāko eksperimenta daļu ir daļēji vai arī pilnīgi aizņemts (sk. att. 3.8).



3.8. att. Eksperimenta pesimistiskā scenārija linka piepildīšanas politikas shēma

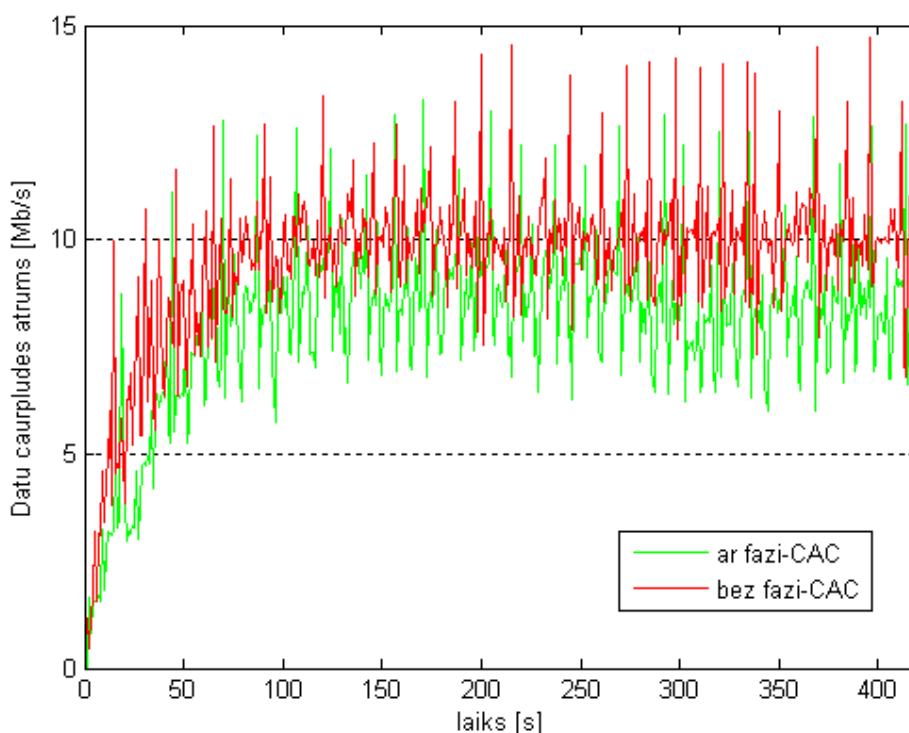
### 3.1.6. Eksperimenta rezultātu attēlojums

Katra eksperimenta laikā tika analizēti 3000 lēmumi. Ņemot vērā viena lēmuma pieņemšanas intervāla laiku, kas sastādīja 4 sekundes, rezultātā viena eksperimenta laiks bija 3 stundas un 20 minūtes. 3000 lēmumu rezultātus nevar attēlot grafiski, jo tie satur pārāk daudz vērtību. Empīriskā ceļā tika noteikta 80 lēmumu robeža, kad tīkla darbība jau ir nostabilizējusies, spriežot pēc QoS parametru rezultātu vizuāliem novērojumiem, ka arī rezultātu grafiskais attēlojums ir skaidrs un ar pietiekamu elementu skaitu, lai vērotu kopējās procesa tendences. Kopējie rezultāti 3000 lēmumiem ir sniegti vidējo vērtību tabulu veidā.

### 3.2. Eksperimenta rezultāti – optimistiskais scenārijs

Kā jau tika minēts iepriekš, optimistiskais scenārijs paredz pilnīgi tukšu 10 Mb/s linku sākuma stāvoklī, kas tiek piepildīts ar fazi-CAC un klasiskā sliekšņa-CAC algoritmu vadītām plūsmām, balstoties uz ieejas parametru datu kopu. Jauno plūsmu pieslēguma pieprasījumi ir 3 reizes biežāki par pieslēgto plūsmu deaktivēšanas gadījumiem, kas paredz, ka links visu laiku atradīsies pārslodzes režīmā.

Attēlā 3.9 ir parādīti trafika caurplūdes dati uz ģenerējošās puses  $T_a$  interfeisa. Attēls 3.10 parāda trafika uztverošās  $T_u$  puses interfeisa trafika caurplūdes ainu. Ģenerēto plūsmu uzliesmojošā daba ir labi redzama 100 Mb/s linkā, kas tiek savienots ar pirmo maršrutētāju, kurš tiek vadīts ar fazi-CAC un klasisko sliekšņa-CAC algoritmu.



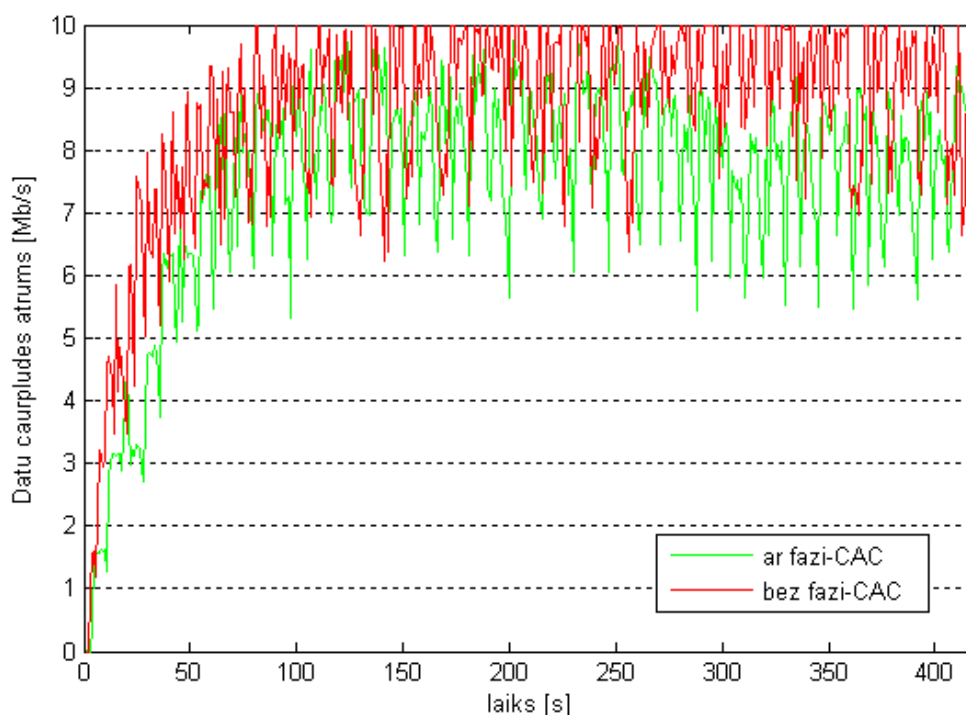
3.9. att. Trafika caurplūde uz trafika ģenerējošās puses  $T_a$  interfeisa

Uztverošās puses  $T_u$  interfeisa trafika caurplūdes aina (att. 3.10) parāda nolīdzinātu trafiku, kas izgāja cauri tīkla šaurajam posmam  $LER_{in} - LSR - LER_{out}$  ar definēto maksimālo caurplūdes ātrumu 10Mb/s.

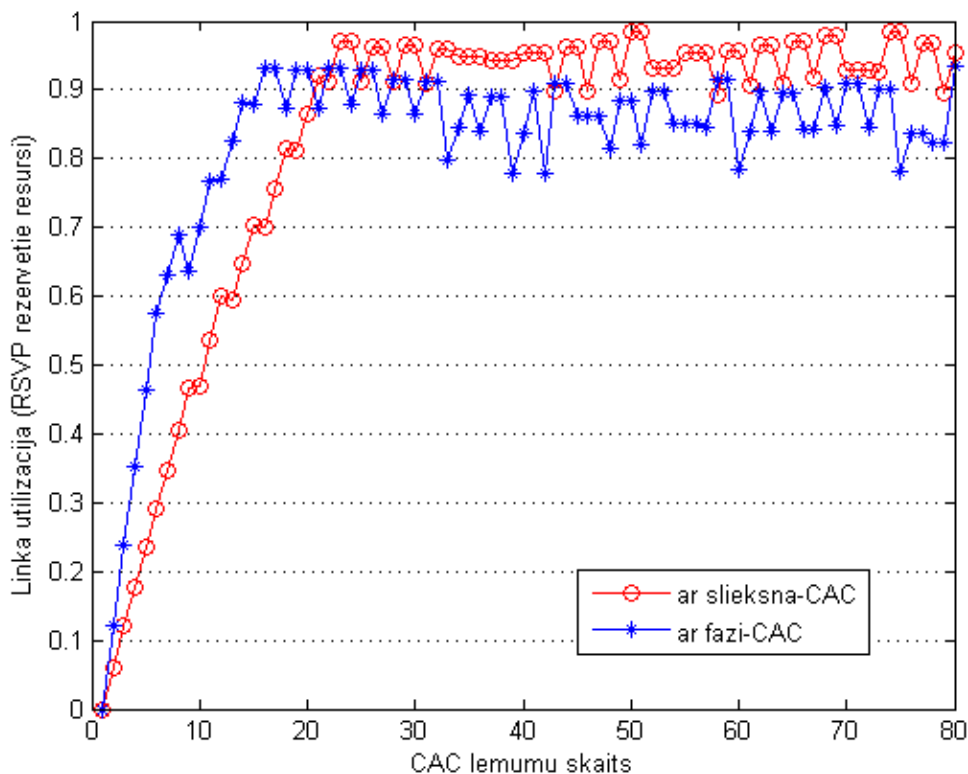
No attēla 3.11, kurā redzams RSVP-TE rezervēto resursu daudzums linkā, varam secināt, ka klasiskā sliekšņa-CAC gadījumā links tiek utilizēts pilnīgāk, ja to salīdzina ar fazi-CAC algoritma darbību. Šie dati faktiski neattēlo reālo linka utilizācijas pakāpi, bet gan rezervēto resursu daudzumu attiecībā pret kopējiem pieejamiem resursiem jeb  $util(t_i)$ . Šie dati tiek izmantoti, nodrošinot trafika vadību – piemēram, šajā gadījumā, RSVP-TE rezervētie resursi tiek izskaitļoti, sasummējot visu LSP plūsmu rezervēto resursu daudzumus, kas tiek nolasīti no  $LER_{in}$ .

Fazi-CAC ir selektīvs un nepieslēdz jaunas plūsmas tikai tāpēc, ka linkā ir brīvie resursi. Balstoties uz pielietoto IF-THEN kārtulu zināšanu bāzi, lēmumi tiek pieņemti, izvērtējot visus zināšanu bāzē definētos nosacījumus un ieejas parametru vērtības.

No attēliem 3.10 un 3.11 varam secināt, ka  $util(t_i)$ , kas tiek izrēķināta, balstoties uz RSVP-TE rezervētiem resursiem, uzrāda lielāku vidēju vērtību nekā  $util_r(t_i)$  jeb reāli nomērītā linka utilizācija. Tas ir izskaidrojams ar to, ka šī promocijas darba autora piedāvātais fazi-CAC algoritms pielieto arī CAC lēmumu „Atļaut ar papildu resursu rezervēšanu”. Rezultātā pie noteiktiem linka QoS parametriem prioritārās LSP plūsmas tiek pieslēgtas ar papildu resursu rezervēšanu. Šāda trafika vadības politika veido t.s. „drošības spilvenu”, kas ir tieši proporcionāls prioritāro LSP plūsmu skaitam linkā un nodrošina QoS parametru saglabāšanu pieļaujamās robežas.



3.10. att. Trafika caurplūde uz trafika uztverošās puses  $T_u$  interfeisa

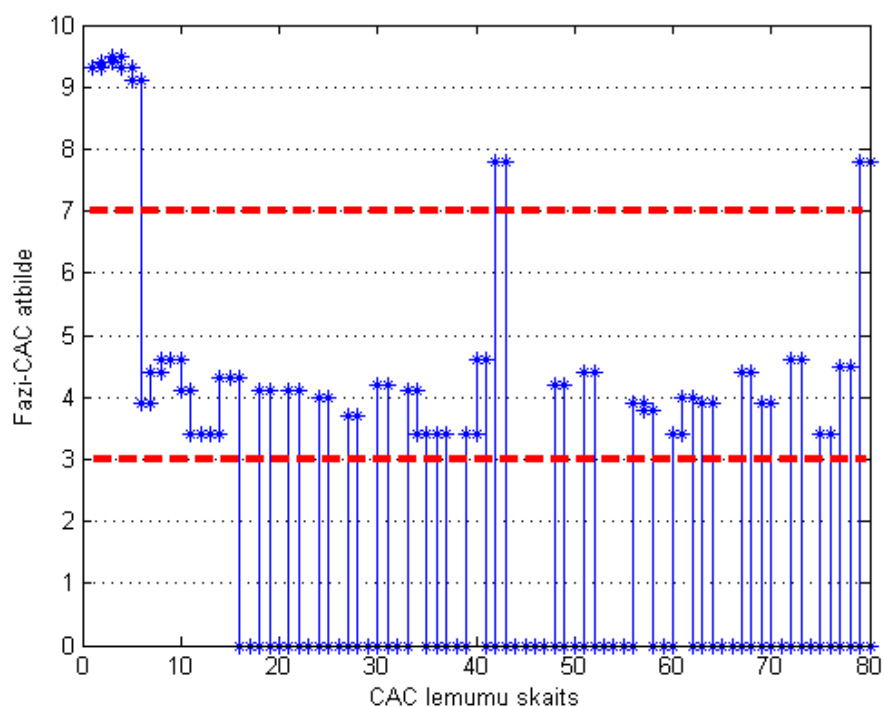


3.11. att. Linka utilizācija / LSP tuneļu RSVP-TE rezervēti resursi

Attēlā 3.12 parādīta fazi-CAC algoritma atbilde  $CAC_f(t_i)$ , kā arī pēcapstrādes sliekšņa vērtības. Attēlā ir parādīti CAC lēmumi, kad LSP plūsmas tiek pieslēgtas, tiek pieslēgtas ar papildu resursu rezervēšanu vai arī plūsmu pieslēgumi tiek atteikti. Konkrēto CAC lēmumu nosaka sliekšņa vērtības, kas ir attēlotas ar sarkano svītraino līniju. Rezultātā  $CAC_f(t_i)$  vērtību apgabals tiek sadalīts 3 segmentos:

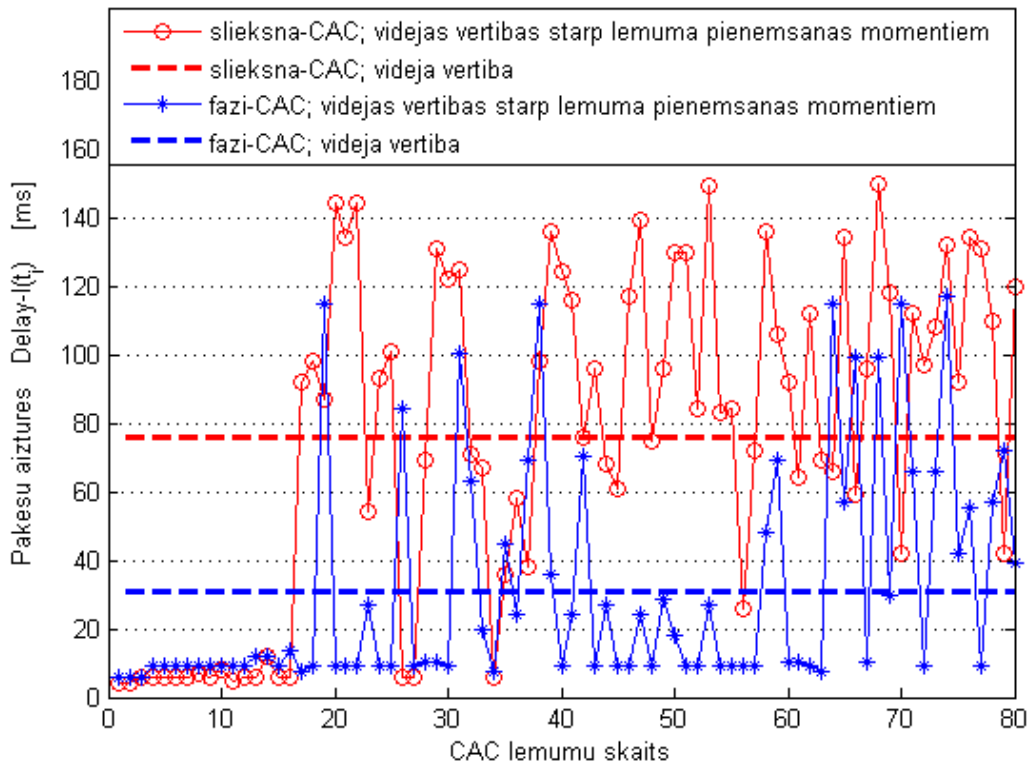
- $CAC_f(t_i) < 3$ , kas nosaka CAC lēmumu „Atteikt”;
- $3 > CAC_f(t_i) < 7$ , kas nosaka CAC lēmumu „Atļaut”;
- $CAC_f(t_i) > 7$ , kas nosaka CAC lēmumu „Atļaut ar papildu resursu rezervēšanu”

$CAC_f(t_i)$  vērtības, kas CAC lēmuma pieņemšanas momentā atradīsies vienā no šiem intervāliem, dos attiecīgi CAC lēmumu.

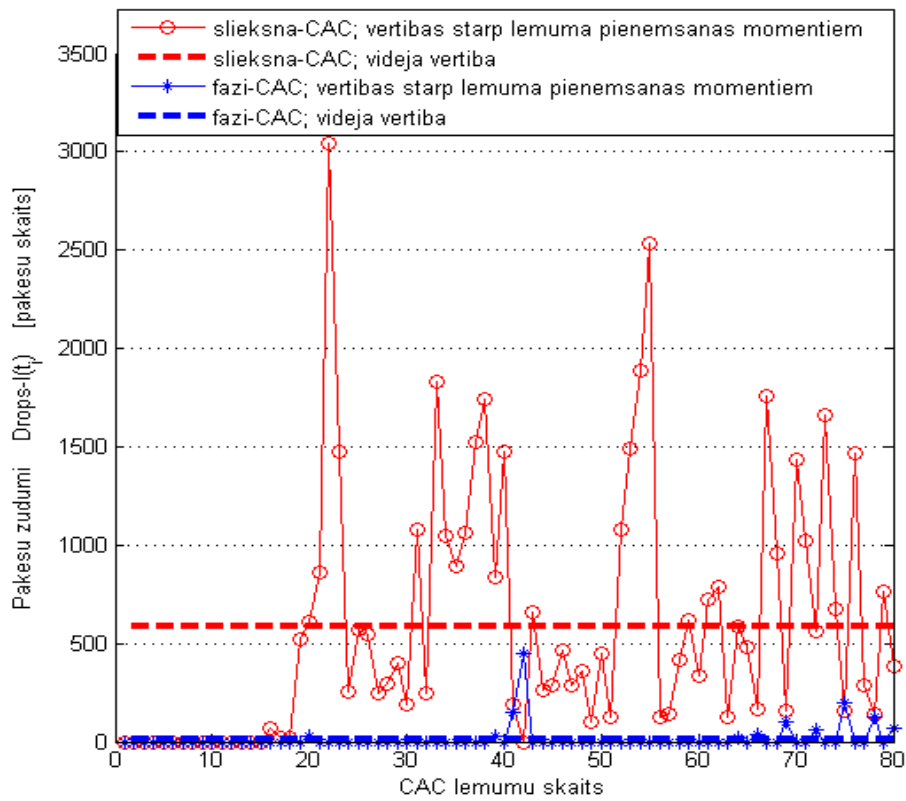


3.12. att. Fazi-CAC atbilde un CAC lēmuma pēcapstrādes sliekšņu vērtības

Tālāk dotais attēls 3.13. parāda pakešu aizturu vērtības eksperimentālajā maketa tīklā starp  $LER_{in}$  un  $LER_{out}$ . Grafikā redzamās vērtības attēlo vidējās pakešu aizturu vērtības starp CAC lēmumu pieņemšanas momentiem, kā arī vidējās pakešu aizturu vērtības 3000 CAC lēmumiem. Grafikā ir redzams, ka vidējās pakešu aizturu vērtības 77ms klasiskā sliekšņa-CAC gadījumā vairāk nekā 2 reizes pārsniedz pakešu aizturu vērtības 33ms, kas tika iegūtas, pielietojot fazi-CAC algoritmu.



3.13. att. Pakešu aiztures tīklā – vidējās vērtības starp lēmumu pieņemšanas momentiem un 3000 lēmumu vidējās vērtība

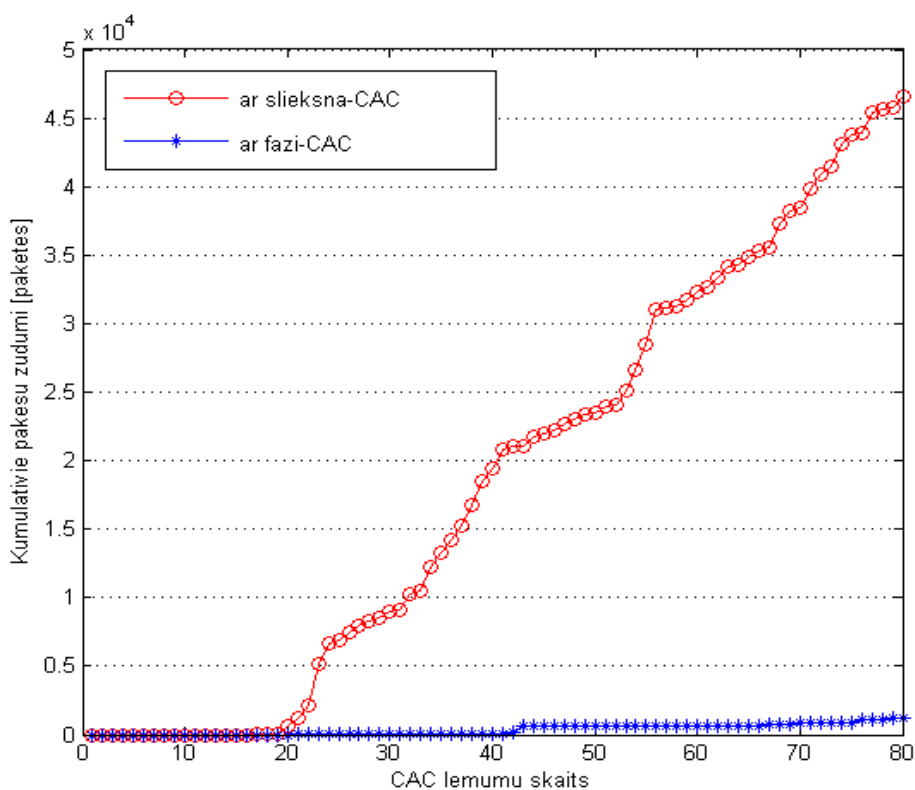


3.14. att. Pakešu zudumu momentānās un vidējā vērtība

Ir jāatzīmē, ka trafika uzliesmojumu gadījumā arī fazi-CAC vadītā MPLS-TE tīklā pakešu aiztures īslaicīgi sniedzās tuvu sliekšņa-CAC vadītā tīkla iegūtām vērtībām. Tomēr atšķirībā no sliekšņa-CAC uz fazioloģiku balstīts CAC risinājums lēmumu pieņemšanas momentos, kad tika detektētas lielas pakešu aiztures, liedza LSP plūsmu pieslēgumus. Tas rezultātā deva  $util_r(t_i)$  samazinājumu un pēc kārtējās aktīvās LSP plūsmas atslēgšanas – arī pakešu aizturu samazinājumu.

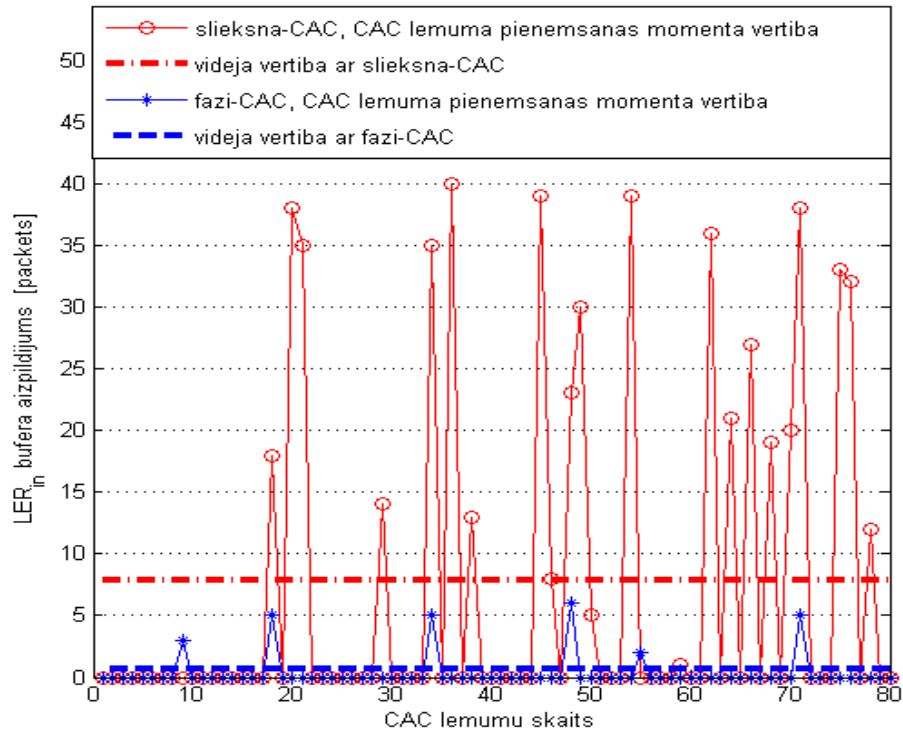
Attēlā 3.14 ir parādīti pakešu zudumi starp  $LER_{in}$  un  $LER_{out}$ . Grafikā ir attēlotas vidējās pakešu zudumu vērtības starp CAC lēmumu pieņemšanas momentiem, kā arī vidējās pakešu zudumu vērtības 3000 CAC lēmumiem.

Pakešu zudumu kumulatīvā sadalījuma grafiks ir parādīts attēlā 3.15. Grafikā ir redzama robeža, pie kuras pētāmais MPLS-TE tīkls tiek pārslogots un sāk parādīties pakešu zudumi. Klasiskā sliekšņa-CAC gadījumā jaunās LSP plūsmas tiek pieslēgtas, balstoties tikai uz brīvo resursu pieejamību tīklā, tikmēr fazi-CAC selektīvā LSP pieslēguma politika dod daudz labāku rezultātu un faktiski nodrošina proaktīvu LSP plūsmu vadību, nepieļaujot lielus pakešu zudumus pētāmajā tīklā.

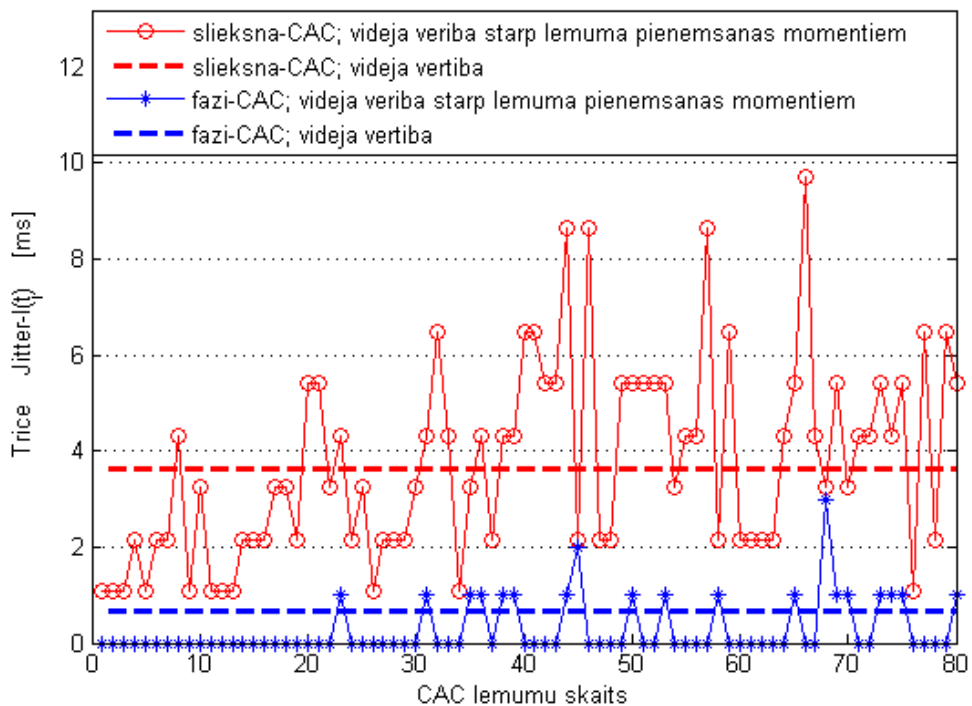


3.15. att. Pakešu zudumu kumulatīvā sadalījuma statistika

Attēlā 3.16 ir parādīti rezultāti par  $LER_{in}$  bufera stāvokli eksperimenta laikā. No attēla varam secināt, ka klasiskā sliekšņa-CAC algoritma gadījumā buferis tiek bieži pārpildīts, radot pakešu zudumus, tikmēr fazi-CAC gadījumā bufera aizpildījums ir minimāls.



3.16. att.  $LER_{in}$  maršrutētāja bufera aizpildījuma momentānās un vidējā vērtība. Bufera izmērs - 40 paketes.



3.17. att. Trīces vērtības CAC lēmuma pieņemšanas brīdī

Attēlā 3.17 ir parādīti rezultāti par pakešu aizturu trīci. Kā varam redzēt no grafika, klasiskā sliekšņa-CAC gadījumā vidējās pakešu trīces vērtības ir vairāk nekā 5 reizes lielākas, kā arī trīces uzliesmojumi sniedzas pie 10 ms robežas.

Tālāk ir doti salīdzinoši rezultāti par fazi-CAC un klasiskā sliekšņa-CAC darbību raksturojošiem QoS parametriem 3000 lēmumiem. Kā var redzēt no eksperimenta rezultātu vidējām vērtībām, tad vidējam utilizācijas līmenim stabilizējoties 0.84 līmenī, QoS degradācijas ierobežojumos definētā 33 ms pakešu aizturu robeža jau tika sasniegta, pielietojot fazi-CAC algoritmu. Pakešu zudumu un pakešu aizturu trīces vērtības joprojām bija zemākas par QoS degradācijas ierobežojumos definētām vērtībām, un, veidojot fazi-CAC lēmējsistēmas struktūru, kas ļautu pietuvoties QoS degradācijas ierobežojumiem šiem parametriem, pakešu aiztures vienmēr kļuva lielākas par degradācijas ierobežojumu 33 ms.

3.1. tabula

Vidējās QoS parametru vērtības 3000 lēmumiem

Eksperimenta rezultāti 3000 lēmumiem – optimistiskais scenārijs – vidējās parametru vērtības		
Parametrs	Sliekšņa-CAC	Fazi-CAC
Linka vidējā utilizācija	0.94	0.84
Pakešu zudumi	3.10 %	0.10 %
Trīce	3.3 ms	0.6 ms
Bufera aizpildījums	8 paketes	1 pakete
Pakešu aiztures	77 ms	33 ms

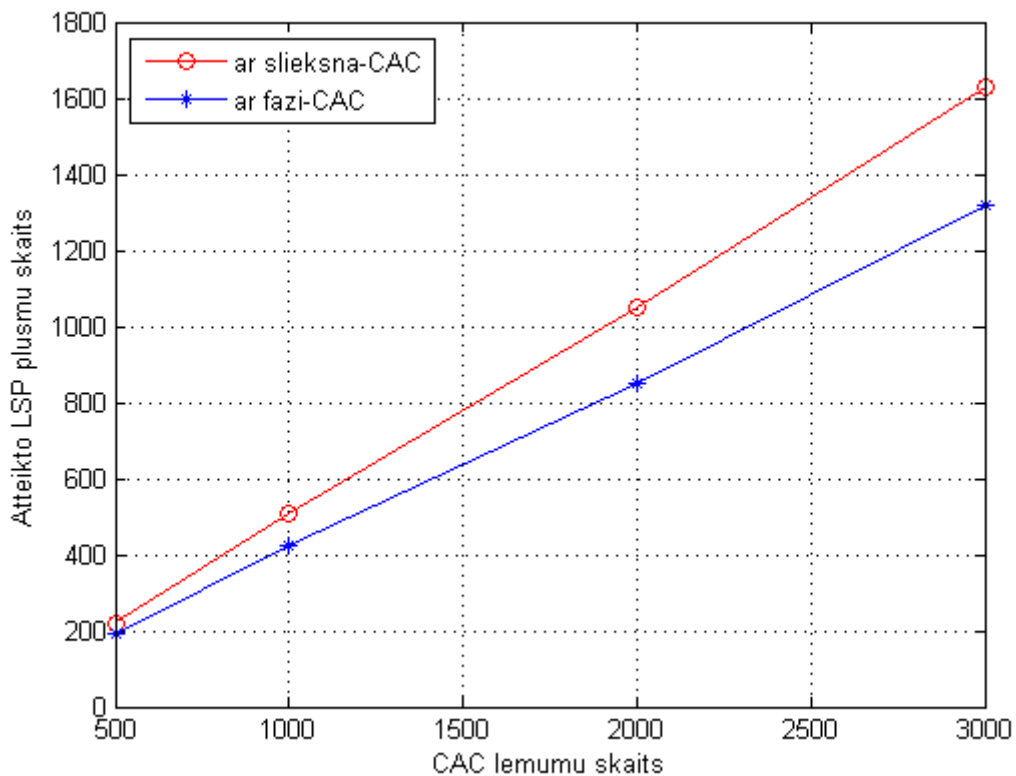
Tālāk ir doti rezultāti par pieslēgtām un atteiktām LSP plūsmām CAC lēmumu pieņemšanas momentos, kā arī par plūsmām, kas tika pieslēgtas ar papildu resursu rezervēšanu. Tabulā 3.2 un attēlā 3.18 ir parādīti rezultāti par atteikto LSP plūsmu skaitu. Šī promocijas darba autora piedāvātā fazi-CAC algoritma selektīvā darbība pie dotā optimistiskā scenārija nodrošina  $\approx 14\%$  LSP plūsmu atteikumu samazinājumu, salīdzinot ar sliekšņa-CAC. Tas tiek panākts, pieslēdzot vairāk jaunas plūsmas ar mazākām caurlaides joslas prasībām, kuras raksturo lielākās  $Bw_r(t_i)$  vērtības, un rezultātā lielāka varbūtība saņemt pozitīvu CAC lēmumu. Sliekšņa-CAC pieslēdz jebkuru LSP plūsmu, ja  $Bw_r(t_i) \geq 1$ , kas noved pie tā, ka

tiek pieslēgtas plūsmas ar lielām caurlaides joslas prasībām, momentāni utilizējot linka brīvos resursus. Rezultātā līdz brīdim, kad kāda no lielām plūsmām linkā tiks atslēgta, tiek atteikti pieslēgumi lielam skaitam turpmāko LSP plūsmu pieslēgumu mēģinājumu.

3.2. tabula

Atteikto LSP plūsmu skaits ar sliekšņa-CAC un fazi-CAC

Optimistiskais scenārijs – atteikto LSP plūsmu skaits		
Lēmumu skaits	Sliiekšņa-CAC	Fazi-CAC
500	220	190
1000	492	425
2000	970	790
3000	1630	1320



3.18. att. Atteikto LSP plūsmu skaits sliekšņa-CAC un fazi-CAC gadījumā

Tabulā 3.3 ir redzami LSP plūsmu pieslēgumu rezultāti, kas parāda sliekšņa-CAC pieslēgtas LSP konekcijas ar neatbilstošiem QoS kritērijiem. Par neatbilstošiem kritērijiem tiek uzskatīti QoS parametri, kas neatbilst šī promocijas darba autora piedāvātā fazi-CAC fazi

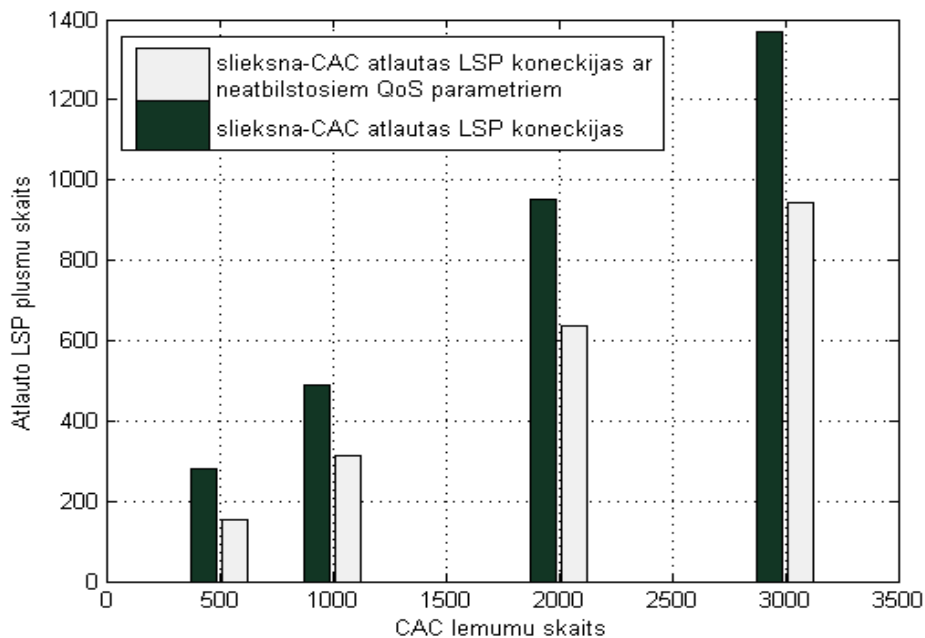
izvedumu sistēmā definētai trafika vadības politikai. Tālāk doti rezultāti attēlo šādus LSP plūsmu pieslēgumus, kas tiku atteikti gadījumā, kad ir pielietots fazi-CAC algoritms.

3.3. tabula

Sliekšņa-CAC pieslēgtas LSP plūsmas ar neatbilstošiem QoS kritērijiem

Optimistiskais scenārijs – sliekšņa-CAC pieslēgtas LSP plūsmas ar neatbilstošiem QoS kritērijiem	
Lēmumu skaits	Sliekšņa-CAC
500	155
1000	315
2000	637
3000	945

Salīdzinot iepriekš doto sliekšņa-CAC pieslēgtu LSP plūsmu skaitu ar visām sliekšņa-CAC pieslēgtām LSP plūsmām (att. 3.19), varam redzēt, ka optimistiskā scenārija gadījumā klasiskais sliekšņa-CAC algoritms  $\approx 60\%$  gadījumos pieslēdz LSP plūsmas, kuras tiek atteiktas fazi-CAC gadījumā.



3.19. att. Sliekšņa-CAC visu atļauto konekciju salīdzinājums ar sliekšņa-CAC atļautām konekcijām ar neatbilstošiem QoS parametriem

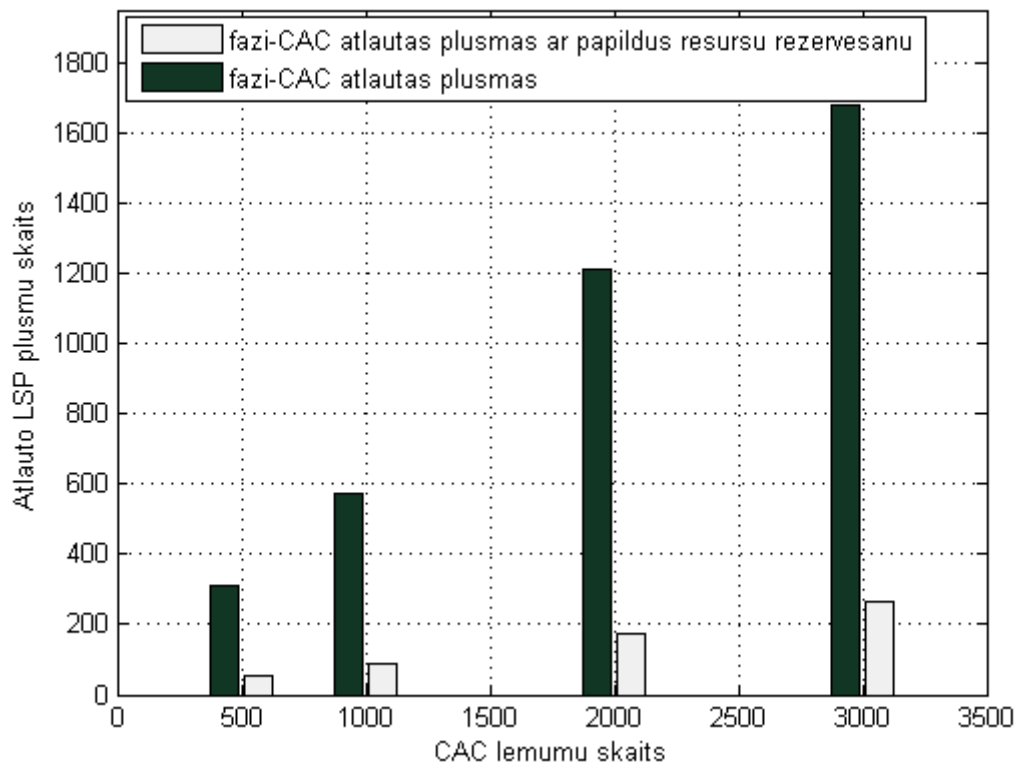
Tabulā 3.4 ir parādītas fazi-CAC pieslēgtas LSP plūsmas ar papildu resursu rezervēšanu. Šādi CAC lēmumi fazi-CAC gadījumā tiek pieņemti, ja  $CAC_f(t_i) > 7$ .

3.4. tabula

Fazi-CAC pieslēgtas LSP plūsmas ar papildus resursu rezervēšanu.

Eksperimenta rezultāti 3000 lēmumiem – optimistiskais scenārijs – fazi-CAC pieslēgtas LSP plūsmas ar papildus resursu rezervēšanu	
Lēmumu skaits	Fazi-CAC
500	57
1000	88
2000	172
3000	265

Salīdzinot visas fazi-CAC pieslēgtās LSP plūsmas ar papildu resursu rezervēšanu pieslēgtām plūsmām, varam redzēt (att. 3.20), ka optimistiskā scenārija gadījumā  $\approx 15\%$  no visām pieslēgtām LSP tiek atļautas ar papildu resursu rezervēšanu.



3.20. att. Fazi-CAC atļautās LSP plūsmas un LSP plūsmas ar papildu resursu rezervēšanu

Tāpat augstas QoS prasības kombinācijā ar zemām pakešu aizturēm un lielu  $Bw_r(t_i)$  vērtību rezultātā dod „drošības spilvenu” ne tikai no jauna pieslēgtām LSP plūsmām, bet arī visām linkā esošajām konekcijām. Turklāt dotais „drošības spilvens” ir tieši proporcionāls augstas prioritātes aktīvām LSP plūsmām linkā.

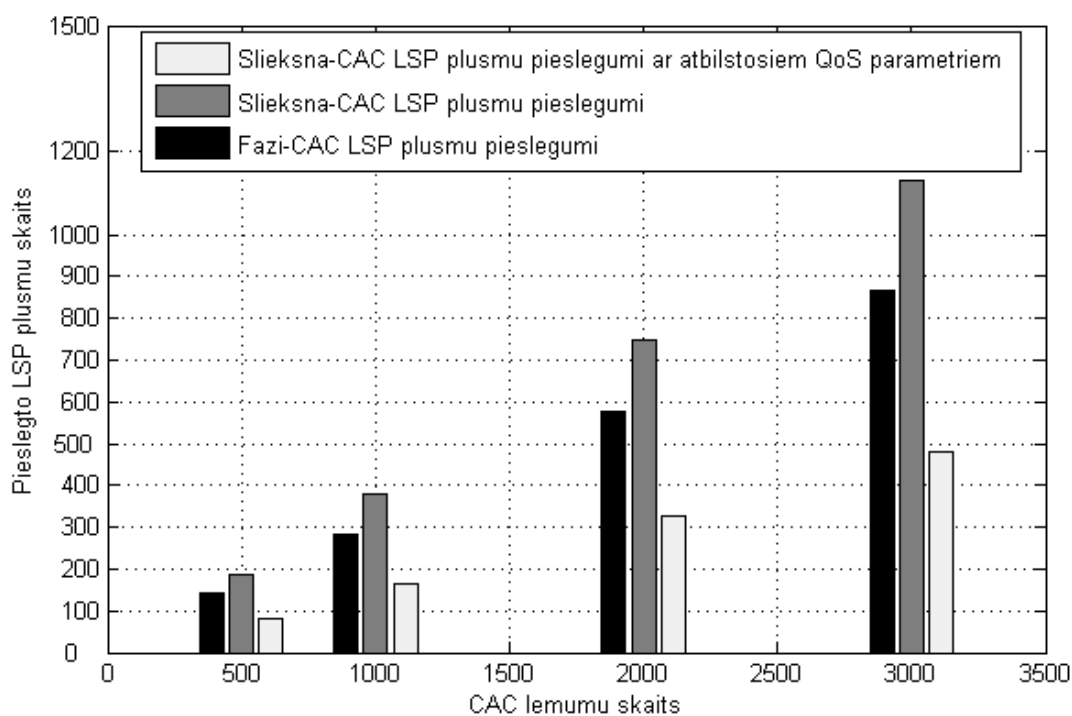
Tālāk dotajā tabulā 3.5 ir parādīti eksperimenta rezultāti, kas attēlo LSP plūsmu pieslēgumus ar  $QoS_c(t_i) > 5$ . Tāpat tiek attēlotas pieslēgtās plūsmas ar  $QoS_c(t_i)$  vērtībām pēc fazifikācijas, kas nosaka augsta līmeņa QoS prasības no aplikācijas puses. Tabulā ir attēloti sliekšņa-CAC plūsmu pieslēgumi, fazi-CAC plūsmu pieslēgumi, kā arī sliekšņa-CAC pieslēgumi. Tiek attēloti arī sliekšņa-CAC LSP plūsmu pieslēgumi augstā QoS līmeņa plūsmām ar neatbilstošiem QoS parametriem tīklā. Par neatbilstošiem kritērijiem tiek uzskatīti QoS parametri, kas neatbilst šī promocijas darba autora piedāvātā fazi-CAC fazi izvedumu sistēmā definētai trafika vadības politikai. Tālāk dotie rezultāti attēlo šādus LSP plūsmu pieslēgumus, kas tiktu atteikti gadījumā, kad ir pielietots fazi-CAC algoritms.

3.5. tabula

Augsta QoS līmeņa LSP plūsmu pieslēgumi

Optimistiskais scenārijs – augsta QoS līmeņa plūsmu pieslēgumi			
Lēmumu skaits	Sliekšņa-CAC	Fazi-CAC	Sliekšņa-CAC ar atbilstošiem QoS parametriem
500	187	142	82
1000	378	284	165
2000	748	575	327
3000	1129	867	482

Attēla 3.21 ir parādīti iepriekš dotie eksperimenta rezultāti. No grafika ir skaidri redzams, ka, pielietojot sliekšņa-CAC, tika pieslēgtas par  $\approx 30\%$  vairāk LSP plūsmu ar augstām QoS prasībām. Tas ir izskaidrojams ar to, ka sliekšņa-CAC nepieslēdz LSP plūsmas selektīvi, izvērtējot aplikācijas QoS prasības un tīkla iespējas pieņemt šādu plūsmu, bet gan balstoties vienīgi uz informāciju par pieejamiem linka resursiem. Rezultātā, sliekšņa-CAC pieslēgtās augsta QoS līmeņa plūsmas tikai  $\approx 43\%$  gadījumos atbilda aplikāciju QoS prasībām. Jaunas LSP plūsmas, kas tika pieslēgtas ar neatbilstošiem QoS parametriem, nespētu nodrošināt aplikācijas prasības un tiktu atslēgtas, tāpat tās nevaram uzskatīt par veiksmīgi pieslēgtām.



3.21. att. LSP plūsmu pieslēgumi fazi-CAC un sliksna-CAC gadījumā ar neatbilstošiem QoS parametriem

Rezultātā, salīdzinot augstā QoS līmeņa fazi-CAC pieslēgtu plūsmu skaitu ar sliksna-CAC pieslēgtu plūsmu skaitu, kuru QoS parametri atbilda aplikāciju prasībām, varam secināt, ka fazi-CAC pieslēdz vidēji par  $\approx 76\%$  vairāk prioritāro LSP plūsmu. Turklāt katrā CAC lēmuma pieņemšanas momentā tiek ņemtas vērā aplikācijas QoS prasības.

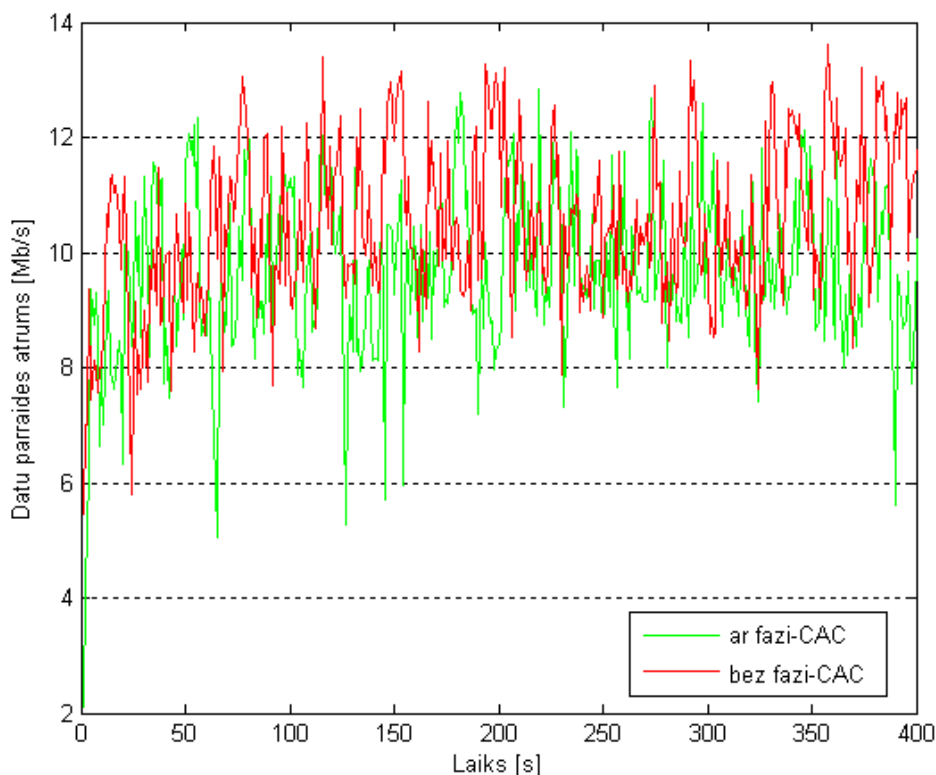
### 3.3. Pesimistiskais scenārijs – rezultāti

Pesimistiskā scenārija pamatā ir izteikti uzliesmojošs fona trafiks ar ātrumu  $\approx 6$  Mb/s, kas neļauj precīzi „dozēt” trafika plūsmas linkā un konsekventi saglabāt QoS parametrus vēlamā līmenī. Tomēr šādā stāvoklī var ļoti labi novērtēt piedāvātā CAC algoritma ietekmi uz sakaru sistēmas darbības robustumu jeb sistēmas spēju saglabāt vai atjaunot pareizu funkcionēšanu pēc dažādu iekšēja vai ārēja rakstura kļūdainu situāciju izveidošanās [2]. Uzliesmojošs fona trafiks labi emulē situāciju, kad linka esošās plūsmas izrāda negaidītu uzvedību un īslaicīgi vai arī ilglaicīgi iziet ārpus rezervēto resursu rāmjiem, kas tika definēti CAC lēmuma laikā un piešķirti ar RSVP-TE protokola palīdzību. Strauji uzliesmojumi nedod iespēju viennozīmīgi balstīt resursu pārvaldības lēmumus, tādus kā CAC, uz RSVP-TE sniegtajiem rezervēšanas datiem. Tas savukārt

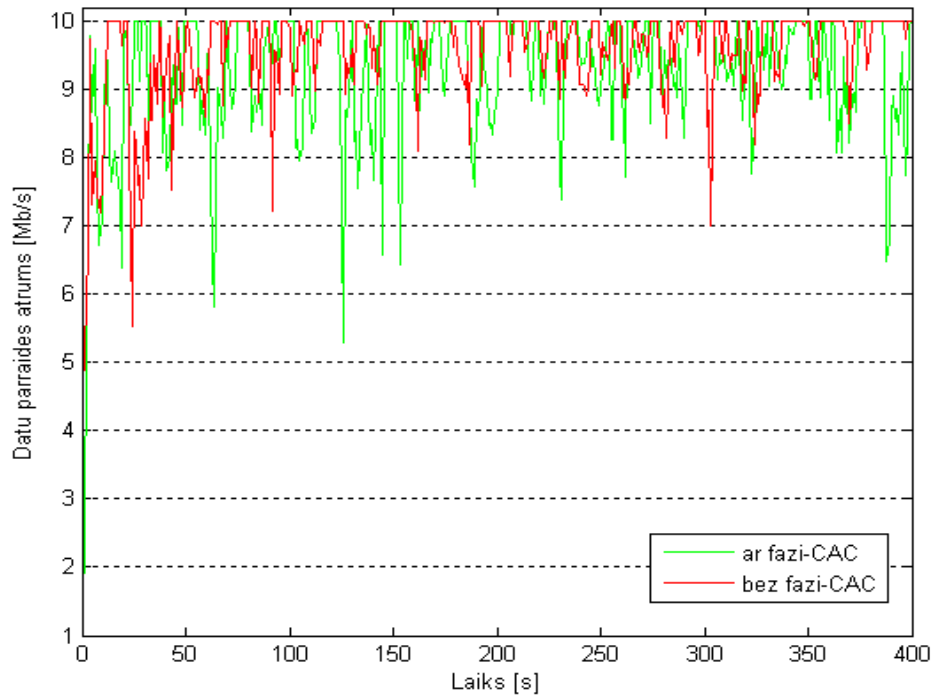
dod lielu priekšroku algoritmiskajiem risinājumiem, kas ir izteikti selektīvi un ar zināmu „piesardzības devu”. Šādu uzvedību piedāvā promocijas darba autora veidotais fazi-CAC algoritms.

Līdzīgi kā optimistiskā scenārijā, attēlā 3.22 ir parādīti trafika caurplūdes dati uz ģenerējošajās puses  $T_a$  interfeisa. Attēlā 3.23 parādīta uztverošās puses  $T_u$  interfeisa trafika caurplūdes aina ar nolīdzinātu trafiku, kas izgāja caur tīkla šaurajam posmam  $LER_{in} - LSR - LER_{out}$  ar definēto maksimālo caurplūdes ātrumu 10Mb/s.

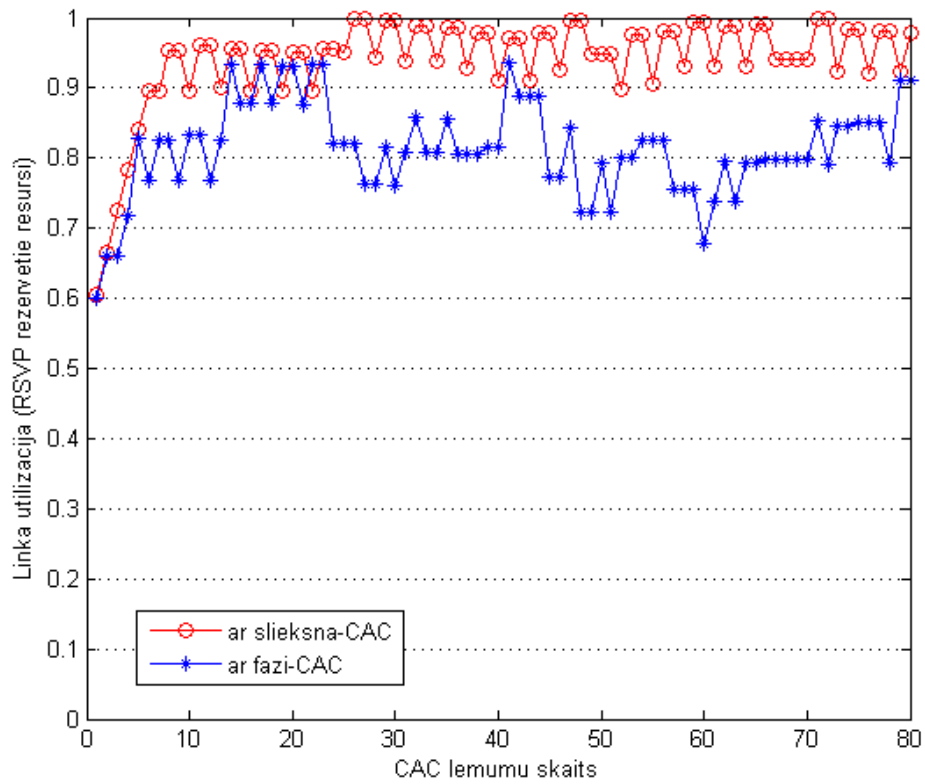
Attēlā 3.24 ir parādīta linka utilizācija, kas izteikta ar RSVP-TE rezervētiem resursiem. Kā jau tika minēts iepriekšējā nodaļā, šie dati faktiski neattēlo reālo linka utilizācijas pakāpi, bet gan rezervēto resursu daudzumu attiecībā pret kopējiem pieejamiem resursiem jeb  $util(t_i)$ .



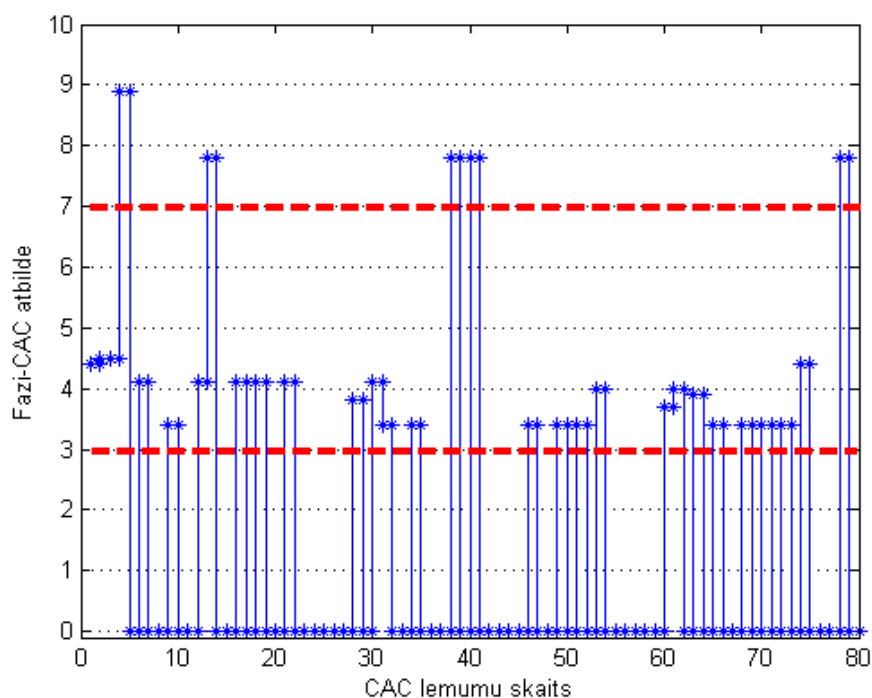
3.22. att. Trafika caurplūde tīkla ģenerējošajā galā [Mb/s]



3.23. att. Trafika caurplūde tīkla uztverošajā galā [Mb/s]



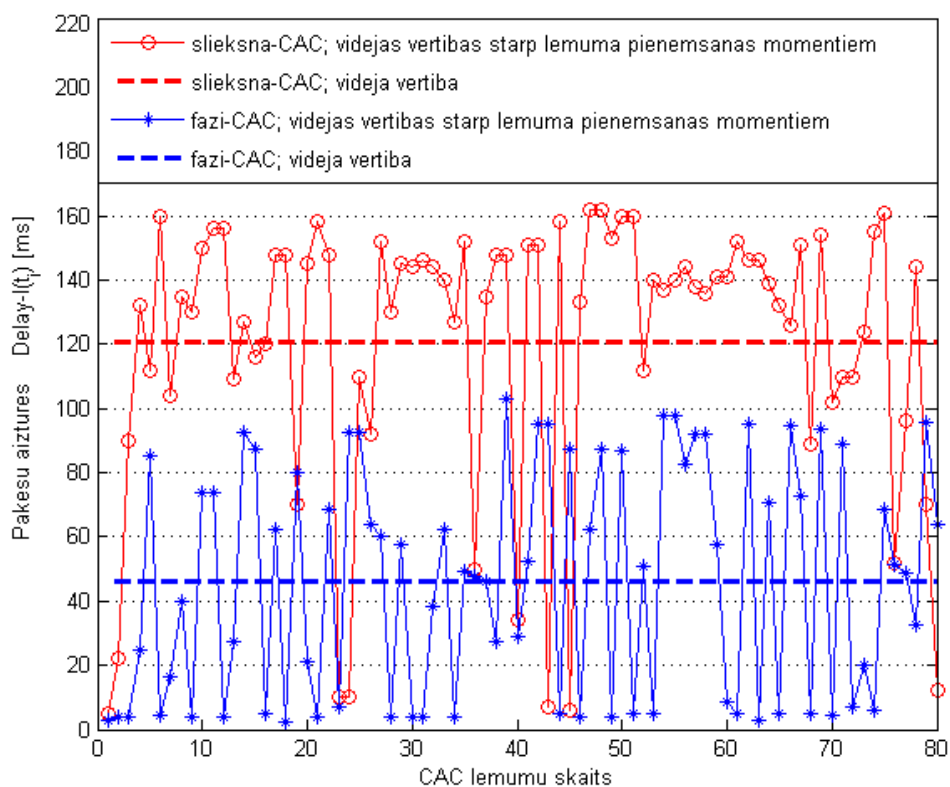
3.24. att. Linka utilizācija / RSVP rezervētie resursi



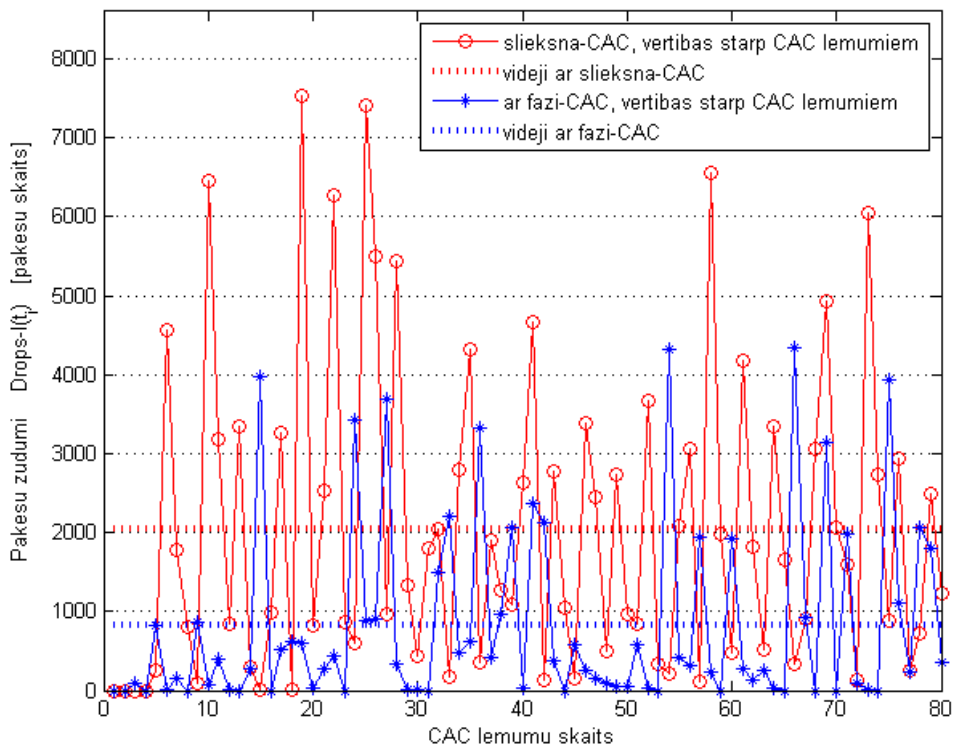
3.25. att. Fazi-CAC atbilde un CAC lēmuma pēcapstrādes sliekšņu vērtības

Attēlā 3.25 parādīta fazi-CAC algoritma atbilde  $CAC_f(t_i)$ , kā arī pēcapstrādes sliekšņa vērtības. Attēlā ir parādīti CAC lēmumi, kad LSP plūsmas tiek pieslēgtas, pieslēgtas ar papildu resursu rezervēšanu vai arī plūsmu pieslēgumi tiek atteikti. Salīdzinājumā ar optimistisko scenāriju no attēla varam secināt, ka ir ievērojami audzis „atteikto” LSP plūsmu skaits un mazinājies „atļauto” un „atļauto ar papildu resursu rezervēšanu” LSP plūsmu skaits.

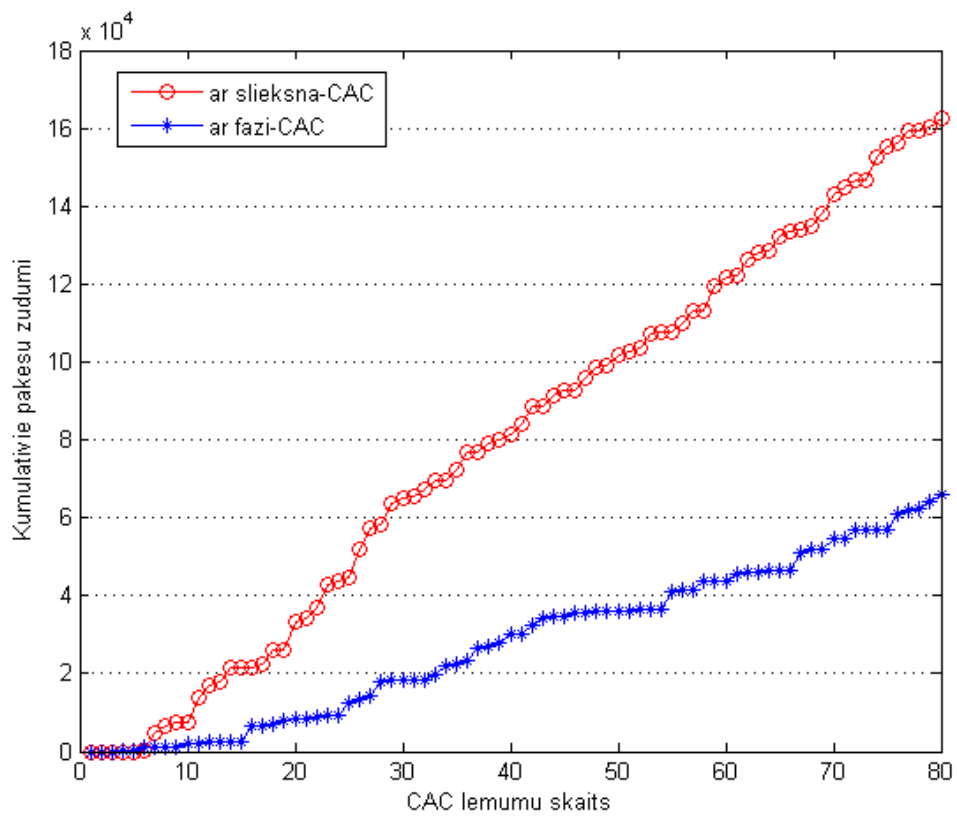
Attēlā 3.26 parādītas pakešu aizturu vērtības eksperimentālajā maketa tīklā starp  $LER_{in}$  un  $LER_{out}$ . Grafikā redzamās vērtības attēlo vidējās pakešu aizturu vērtības starp CAC lēmumu pieņemšanas momentiem, kā arī vidējās pakešu aizturu vērtības 3000 CAC lēmumiem. Salīdzinot ar optimistisko scenāriju, pakešu aizturu vidējās vērtības, fazi-CAC gadījumā ir augušas par  $\approx 40\%$  un sastādīja 46 ms. Klasiskā sliekšņa-CAC gadījumā pakešu aizturu vidējās vērtības ir augušas par  $\approx 57\%$  un sastādīja 121 ms.



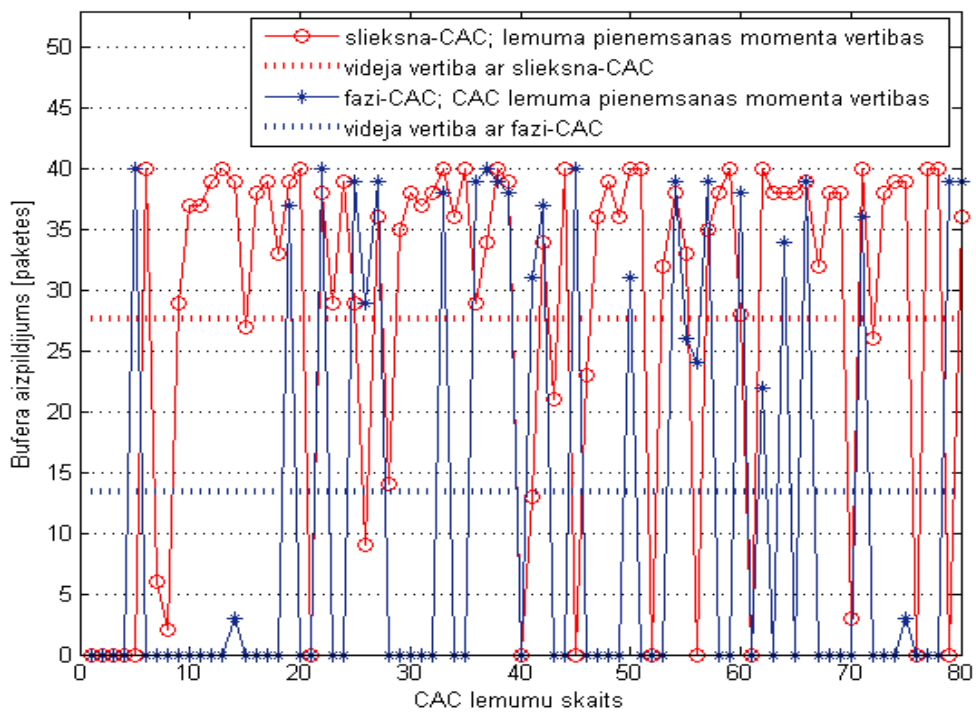
3.26. att. Momentānās un vidējās pakešu aiztures ar un bez fazi-CAC algoritma pielietojanas



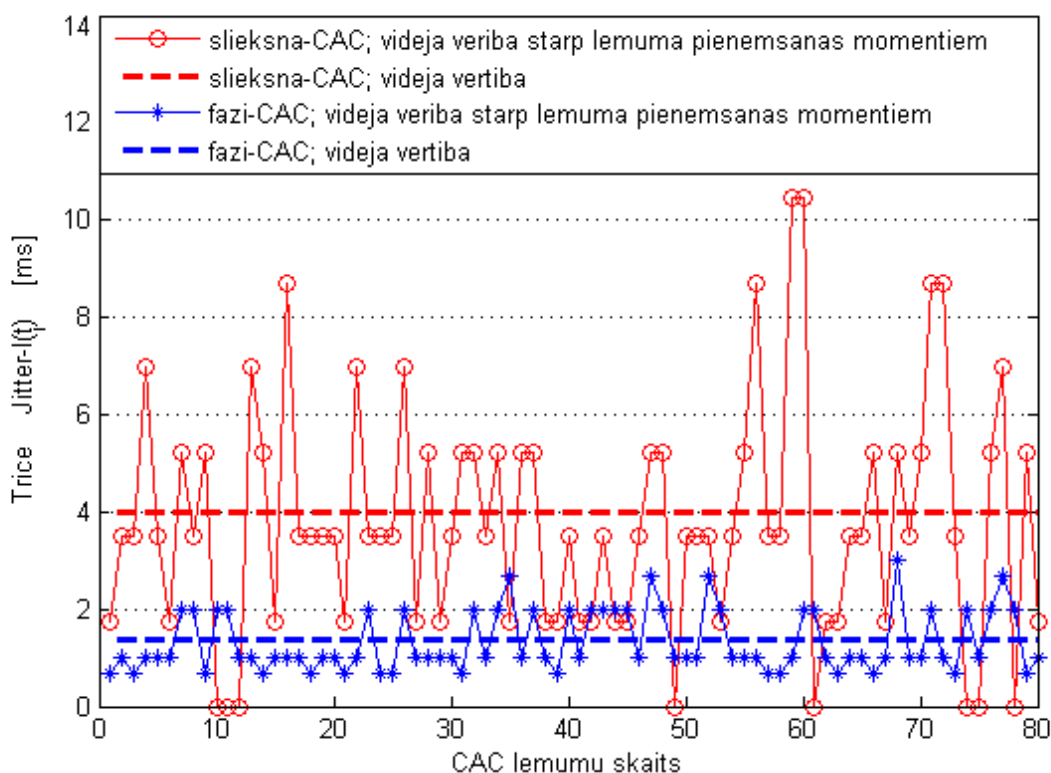
3.27. att. Pakešu zudumi starp CAC lēmuma pieņemšanas laikiem



3.28. att. Pakēšu zudumu kumulatīvs grafiks



3.29. att. Momentānās un vidējās vērtības maršrutētāja bufera stāvoklis. Bufēra izmērs - 40 paketes



3.30. att. Momentānās trīces vērtības CAC lēmuma pieņemšanas brīdī un vidējās vērtības

Attēlā 3.29 ir doti rezultāti par  $LER_{in}$  bufera stāvokli eksperimenta laikā pesimistiskā scenārija gadījumā. No grafika varam secināt, ka gan klasiskā sliekšņa-CAC algoritma gadījumā, gan fazi-CAC gadījumā buferis tiek bieži pārpildīts, radot pakešu zudumus. Salīdzinot ar optimistiskā scenārija eksperimenta rezultātiem, varam redzēt, ka sliekšņa-CAC gadījumā vidējais bufera aizpildījums pieauga  $\approx 3,5$  reizes un sastādīja 28 paketes. Tikmēr fazi-CAC gadījumā vidējais bufera aizpildījums pieauga 14 reizes un sastādīja 14 paketes.

Tālāk ir doti salīdzinoši eksperimentālie dati par fazi-CAC un klasiskā sliekšņa-CAC darbību raksturojošiem QoS parametriem 3000 lēmumiem pesimistiskā scenārija gadījumā (tabula 3.6). Kā var redzēt no eksperimenta rezultātu vidējām vērtībām, tad gan sliekšņa-CAC, gan fazi-CAC algoritma gadījumā MPLS-TE sakaru sistēmas QoS parametri, salīdzinot ar optimistiskā scenārija eksperimenta rezultātiem, izteikti degradēja. Šī eksperimenta mērķis bija redzēt, cik stipri atšķirsies fazi-CAC un sliekšņa-CAC QoS parametru degradācijas līmeņi pie kritiskiem tīkla noslogotības režīmiem.

Linka utilizācijas vērtības samazinājās no 0,84 līdz 0,82 fazi-CAC gadījuma un pieauga no 0,94 līdz 0,97 sliekšņa-CAC gadījumā. Šādu RSVP-TE rezervēto resursu linka utilizācijas  $util(t_i)$  samazinājumu fazi-CAC gadījumā var izskaidrot ar to, ka uzliesmojošā fona trafika gadījumā tā uzlēcienu laikos pieaug QoS parametru degradācija. Palielinās arī pakešu aiztures

$Delay_l(t_i)$ , un fazi-CAC arī pie tām pašām  $Bw_r(t_i)$  vērtībām pieņem vairāk negatīvo CAC lēmumu. Tādā veidā fazi-CAC padara trafika vadības politiku vairāk selektīvu attiecībā uz jauno LSP plūsmu pieslēgumiem uzliesmojošā trafika gadījumā.

Savukārt sliekšņa-CAC vadītā tīkla utilizācijas pieaugumu var izskaidrot ar to, ka fona trafika uzliesmojumi rāda lokālus caurplūdes pīķus un kritumus. Tas savukārt noved pie tā, ka, pieņemot CAC lēmumus gadījumā, kad  $Bw_r(t_i)$  vērtība ir pietiekami augsta, dod pozitīvu CAC lēmumu, kaut arī vidēji  $Bw_r(t_i)$  ir ar zemāko vērtību.

3.6. tabula

Vidējās QoS parametru vērtības 3000 lēmumiem

Eksperimenta rezultāti 3000 lēmumiem – pesimistiskais scenārijs – vidējās parametru vērtības		
Parametrs	Sliekšņa-CAC	Fazi-CAC
Linka vidējā utilizācija	0,97	0,82
Pakešu zudumi	11,80 %	3,90 %
Trīce	4,0 ms	1,3 ms
Bufera aizpildījums	28 paketes	14 paketes
Pakešu aiztures	121 ms	46 ms

Analizējot datus par pieslēgtām un atteiktām LSP plūsmām CAC lēmumu pieņemšanas momentos, kā arī par plūsmām, kas tika pieslēgtas ar papildu resursu rezervēšanu, ir iegūti šādi rezultāti.

Tabulā 3.7 un attēlā 3.31 ir parādīti rezultāti par atteikto LSP plūsmu skaitu. Šī promocijas darba autora piedāvātā fazi-CAC algoritma selektīvā darbība pie dotā pesimistiskā scenārija nodrošina  $\approx 20\%$  LSP plūsmu atteikumu samazinājumu, salīdzinot ar sliekšņa-CAC.

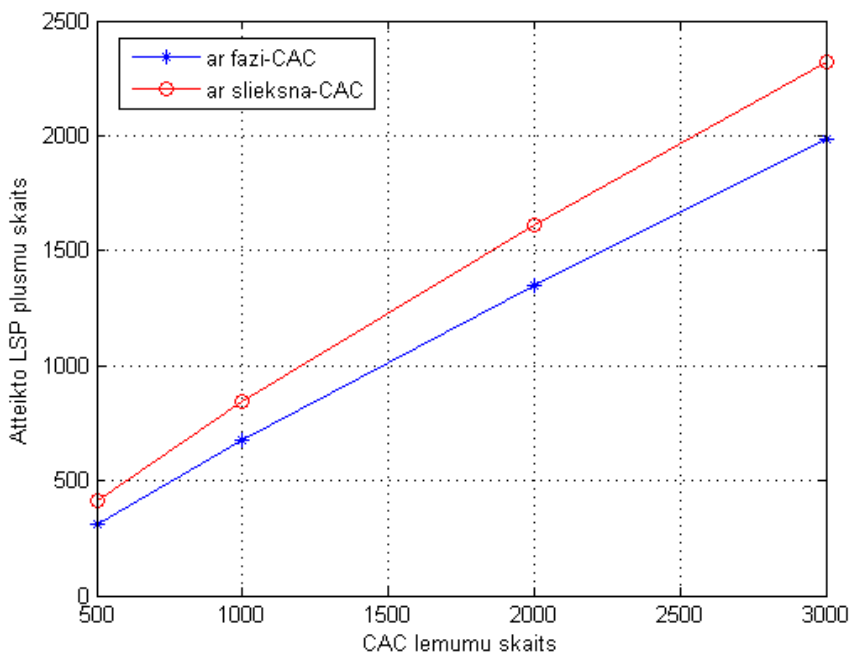
3.7. tabula

Atteikto LSP plūsmu skaits ar sliekšņa-CAC un fazi-CAC

Eksperimenta rezultāti 3000 lēmumiem – pesimistiskais scenārijs – atteikto LSP plūsmu skaits		
Lēmumu skaits	Sliekšņa-CAC	Fazi-CAC
<b>500</b>	410	312
<b>1000</b>	845	679
<b>2000</b>	1610	1348
<b>3000</b>	2321	1987

Tāpat kā optimistiskā scenārija gadījumā, tas tiek panākts, pieslēdzot vairāk jaunas plūsmas ar mazākām caurlaides joslas prasībām, kuras lēmuma pieņemšanas momentā

raksturo lielākās  $Bw_r(t_i)$  vērtības. Rezultātā šādām LSP plūsmām ir lielāka varbūtība saņemt pozitīvu CAC lēmumu.



3.31. att. Atteikto LSP plūsmu skaits sliekšņa-CAC un fazi-CAC algoritmu gadījumā

Tabulā 3.8 ir redzami LSP plūsmu pieslēgumu rezultāti, kas parāda sliekšņa-CAC pieslēgtās LSP konekcijas ar neatbilstošiem QoS kritērijiem. Par neatbilstošiem kritērijiem tiek uzskatīti QoS parametri, kas neatbilst šī promocijas darba autora piedāvātā fazi-CAC fazi izvedumu sistēmā definētai trafika vadības politikai. Tālāk dotie rezultāti attēlo šādus LSP plūsmu pieslēgumus, kas tiktu atteikti gadījumā, kad ir pielietots fazi-CAC algoritms.

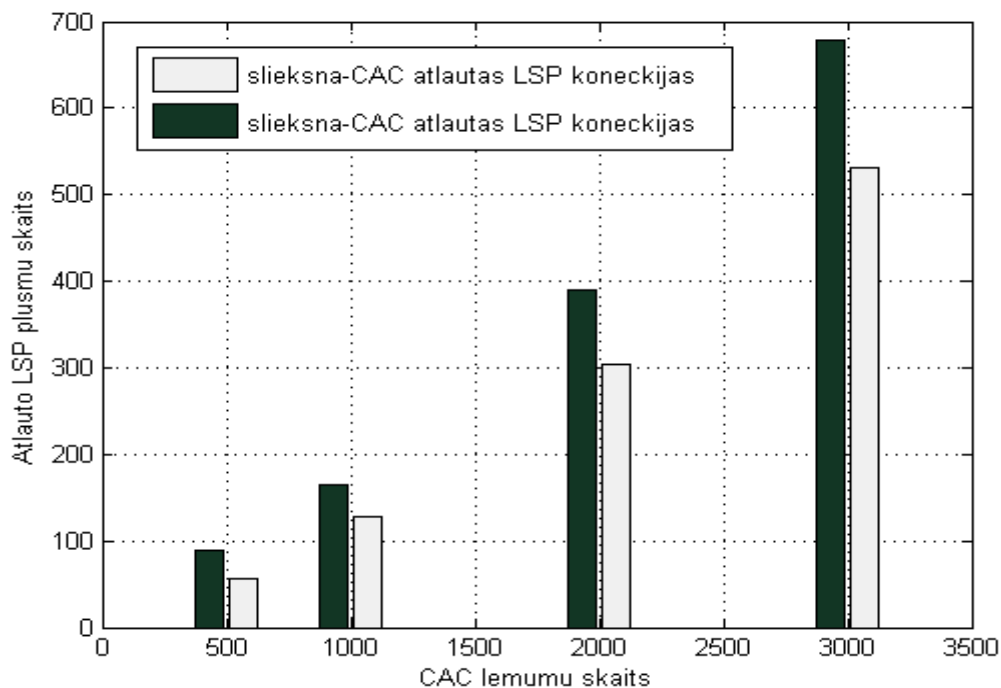
3.8. tabula

Sliekšņa-CAC pieslēgtas LSP konekcijas ar neatbilstošiem QoS kritērijiem

Eksperimenta rezultāti 3000 lēmumiem – pesimistiskais scenārijs – sliekšņa-CAC pieslēgtas LSP konekcijas ar neatbilstošiem QoS kritērijiem	
Lēmumu skaits	Sliekšņa-CAC
<b>500</b>	70
<b>1000</b>	122
<b>2000</b>	306
<b>3000</b>	538

Salīdzinot iepriekš dotu sliekšņa-CAC pieslēgtu LSP plūsmu skaitu ar neatbilstošiem QoS parametriem (att. 3.32) ar visām sliekšņa-CAC pieslēgtām LSP plūsmām, varam redzēt,

ka pesimistiskā scenārija gadījumā klasiskais sliekšņa-CAC algoritms  $\approx 80\%$  gadījumos pieslēdz LSP plūsmas, kuras tiek atteiktas fazi-CAC gadījumā.



3.32. att. Sliekšņa-CAC visu atļauto konekciju salīdzinājums ar sliekšņa-CAC atļautām konekcijām ar neatbilstošiem QoS parametriem

Tabulā 3.9 ir parādītas fazi-CAC pieslēgtās LSP plūsmas ar papildu resursu rezervēšanu. Šādi CAC lēmumi fazi-CAC gadījumā tiek pieņemti, ja  $CAC_f(t_i) > 7$ .

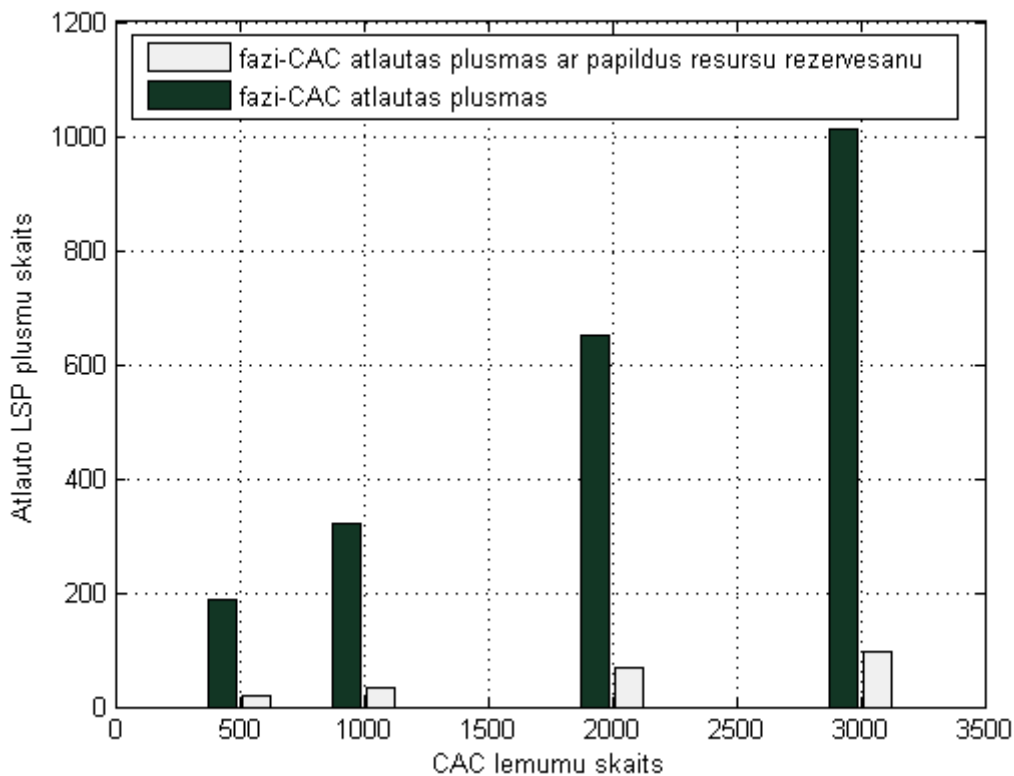
3.9. tabula

Fazi-CAC pieslēgtas LSP konekcijas ar papildu resursu rezervēšanu

Eksperimenta rezultāti 3000 lēmumiem – pesimistiskais scenārijs – fazi-CAC pieslēgtas LSP konekcijas ar papildus resursu rezervēšanu	
Lēmumu skaits	fazi-CAC
<b>500</b>	19
<b>1000</b>	32
<b>2000</b>	67
<b>3000</b>	98

Salīdzinot visas fazi-CAC pieslēgtās LSP plūsmas ar papildu resursu rezervēšanu pieslēgtām plūsmām, varam redzēt (att. 3.33), ka pesimistiskā scenārija gadījumā  $\approx 10\%$  no

visām pieslēgtajām LSP tiek atļautas ar papildu resursu rezervēšanu. Optimistiskā scenārija gadījumā  $\approx 15\%$  no visām pieslēgtajām LSP tika atļautas ar papildus resursu rezervēšanu. Tas ir izskaidrojams ar to, ka pesimistiskā scenārija linka QoS parametru vērtības neļāva selektīvajam fazi-CAC tikpat bieži virzīties uz atbildēm, kad  $CAC_f(t_i) > 7$ . Rezultātā mazāk LSP plūsmu ar papildus resursu rezervēšanu tika pieslēgtas linkā.



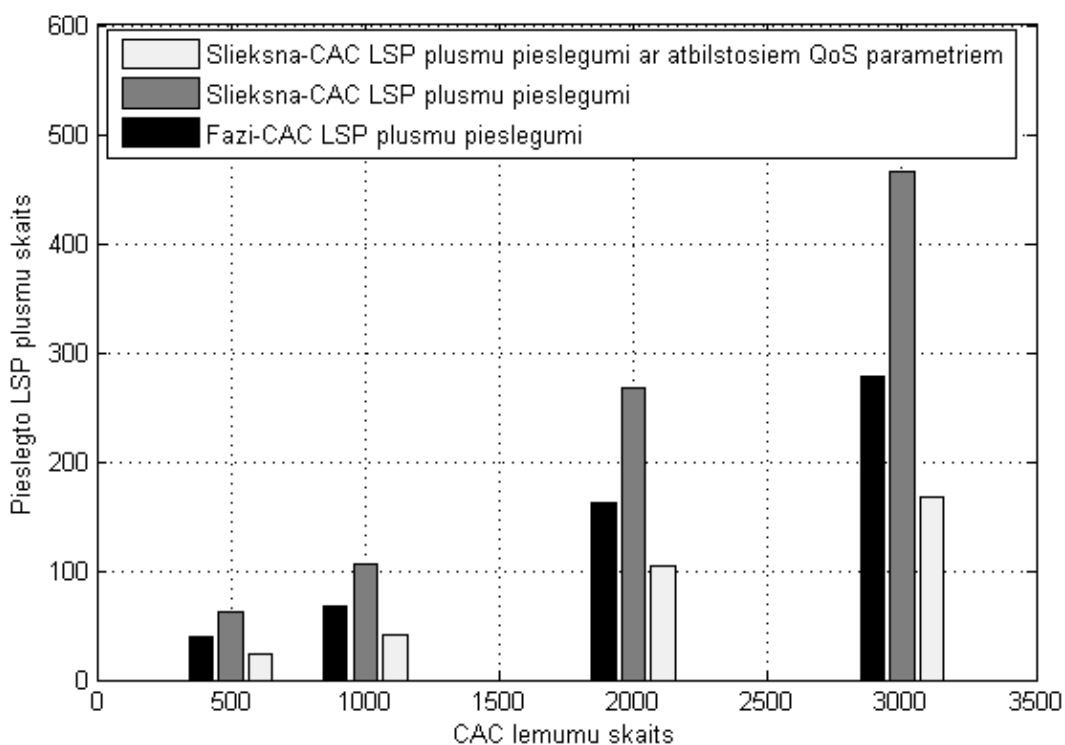
3.33. att. Fazi-CAC atļautās LSP plūsmas un LSP plūsmas ar papildu resursu rezervēšanu

Tabulā 3.10 ir parādīti eksperimenta rezultāti, kas attēlo LSP plūsmu pieslēgumus ar  $QoS_c(t_i) > 5$ . Tātad tiek attēlotas pieslēgtās plūsmas ar  $QoS_c(t_i)$  vērtībām pēc fazifikācijas, kas nosaka augsta līmeņa QoS prasības no aplikācijas puses. Tabulā ir attēloti sliekšņa-CAC plūsmu pieslēgumi, fazi-CAC plūsmu pieslēgumi, kā arī sliekšņa-CAC pieslēgumi. Tiek attēloti arī sliekšņa-CAC LSP plūsmu pieslēgumi augstā QoS līmeņa plūsmām ar neatbilstošiem QoS parametriem tīklā. Par neatbilstošiem kritērijiem, tiek uzskatīti QoS parametri, kas neatbilst šī promocijas darba autora piedāvātā fazi-CAC fazi izvedumu sistēmā definētai trafika vadības politikai. Tātad, tālāk dotie rezultāti attēlo šādus LSP plūsmu pieslēgumus, kas tiktu atteikti gadījumā, kad ir pielietots fazi-CAC algoritms.

## Augsta QoS līmeņa plūsmu pieslēgumi

Pesimistiskais scenārijs – augsta QoS līmeņa plūsmu pieslēgumi			
Lēmumu skaits	Sliekšņa-CAC	Fazi-CAC	Sliekšņa-CAC ar atbilstošiem QoS parametriem
500	63	39	24
1000	106	67	41
2000	268	163	104
3000	465	278	167

Attēlā 3.34 ir atspoguļoti iepriekš dotie eksperimenta rezultāti. No grafika ir redzams, ka pesimistiska scenārija gadījumā, pielietojot sliekšņa-CAC, tika pieslēgtas par  $\approx 40\%$  vairāk LSP plūsmu ar augstām QoS prasībām.



3.34. att. LSP plūsmu pieslēgumi fazi-CAC un sliekšņa-CAC gadījumā ar neatbilstošiem QoS parametriem

Tas ir izskaidrojams ar to, ka sliekšņa-CAC nepieslēdz LSP plūsmas selektīvi, izvērtējot aplikācijas QoS prasības un tīkla iespējas pieņemt šādu plūsmu, bet gan balstoties vienīgi uz informāciju par pieejamiem linka resursiem. Rezultātā sliekšņa-CAC pieslēgtās augsta QoS līmeņa plūsmas tikai  $\approx 38\%$  gadījumu atbilda aplikāciju QoS prasībām. Jaunas LSP plūsmas, kas tika pieslēgtas ar neatbilstošiem QoS parametriem, nespētu nodrošināt aplikācijas prasības un tiktu atslēgtas, tātad tās nevaram uzskatīt par veiksmīgi pieslēgtām. Rezultātā, salīdzinot augstā QoS līmeņa fazi-CAC pieslēgtu plūsmu skaitu ar sliekšņa-CAC pieslēgtu plūsmu skaitu, kuru QoS parametri atbilda aplikāciju prasībām, varam secināt, ka fazi-CAC pieslēdz vidēji par  $\approx 60\%$  vairāk prioritāro LSP plūsmu.

## 4. FAZI-CAC ADAPTĀCIJAS MEHĀNISMI

### Nodaļas kopsavilkums

Šajā nodaļā promocijas darba autors īsi apskata piedāvātā fazi-CAC algoritma iespējamās adaptācijas mehānismus, kas nodrošinātu efektīvas algoritma pielāgošanas spējas ātri mainīgajai ārējai videi. Tiek apskatīta iespēja tiešsaistē pielāgot fazi-CAC darbības politiku, modificējot lingvinistisko mainīgo piederības funkciju deklarācijas un fazi-CAC atbildes lēmuma pieņemšanas sliekšņa vērtības, analizējot izmaiņu ietekmi uz MPLS-TE sakaru sistēmas darbību. Pēdējais risinājums, kaut arī tiešā veidā neietekmē fazi-lēmējsistēmas darbību, tomēr nodrošina ļoti ātru un efektīvu fazi-CAC tiešsaistes modifikāciju.

Rezultātā tiks sniegti eksperimentu rezultāti par iespējamo risinājumu ietekmi uz fazi izvedumu sistēmas darbības efektivitāti, nodrošinot fazi-CAC vadību MPLS-TE tīklā.

Tā kā Fazi-CAC lēmējsistēma nodrošina lēmuma pieņemšanu apstākļos, kad tīkla parametru vērtības un trafika kopējais raksturs ir lielā mērā neparedzams, ir jāpēta fazi-CAC algoritma adaptācijas mehānismi, kas nodrošinātu efektīvas algoritma pielāgošanas spējas ātri mainīgajai ārējai videi.

Faktiski pastāv divi pamata veidi, kā izmainīt fazi-lēmējsistēmas uzvedību. Pirmkārt, ir iespējams modificēt IF-THEN kārtulu zināšanu bāzi, kas apraksta sistēmas vēlamā uzvedību. Otrkārt, var modificēt lingvinistisko mainīgo piederības funkciju deklarācijas. Pēdējā gadījumā var izcelt vai arī pretēji – samazināt kāda ieejas mainīgā nozīmi lēmuma pieņemšanā. Tas savukārt nozīmē, ka, atstājot IF-THEN kārtulu zināšanu bāzi neskartu, tiek mainītas tikai noteiktu lingvinistisko mainīgo definīcijas, šādā veidā izmainot lēmuma pieņemšanas tendences vēlamajā virzienā.

Šīs disertācijas autors piedāvā apskatīt vēl vienu paņēmieni – lēmuma pieņemšanas sliekšņa izmaiņas, kas, kaut arī tiešā veidā neietekmē fazi-lēmējsistēmas darbību, tomēr var tikt pielietots, lai nodrošinātu ļoti ātru un efektīvu fazi-CAC interfeisa tiešsaistes modifikāciju.

IF-THEN kārtulas nodrošina fazi-lēmējsistēmas uzvedības aprakstošu zināšanu bāzi, kas savā būtībā diezgan viennozīmīgi nosaka fazi algoritma lēmuma pieņemšanas stratēģiju. Šis fazi kontroles mehānisma pamatelements iepriekšējos promocijas darba autora fazi-CAC algoritma pētījumos tika izvēlēts empīriski, pamatojoties uz autora izpratni par vēlamā

algoritma lēmumu pieņemšanas loģiku un izvirzītiem mērķiem fazi-CAC FIS darbībai (sk. nodaļu 2.7.1). Šāda pieeja ir pieļaujama un dod iespēju apkopot zināšanas, kas ir izteiktas IF-THEN kārtulu veidā.

Rodas jautājums par to, kādā veidā nodrošināt fazi sistēmas iespējamo ieejas un vēlamu izejas parametru savietojumu, automātiski ģenerējot vienu no iespējamajām fazi lēmējsistēmas realizācijām, ar definētām piederības funkciju deklarācijām un konkrētu IF-THEN kārtulu zināšanu bāzi.

Fazi izvedumu sistēmas zināšanu bāzes izvēles jautājums būtībā ir ļoti plašs un komplikēts, un tam vien var veltīt veselu promocijas darbu. Tāpēc šī promocijas darba ietvaros darba autors, nevēlēdamies veikt paviršus un virspusējus pētījumus šajā jomā, neapskata dziļāk šo jautājumu.

Vienīgi gribētos pieminēt dažus tehnoloģiskus risinājumus, kas tiek pielietoti fazi izvedumu sistēmas IF-THEN kārtulu zināšanu bāzes izveidē, tādus kā neironu tīkli [41] [71], ģenētiski algoritmi [144] [78] un datizrace [49] [112] [72] [22]. Šīs pieejas nodrošina iespēju veidot FIS zināšanu bāzes IF-THEN kārtulu veidā:

- balstoties uz pašapmācošiem algoritmiem, analizējot sistēmas reakciju tiešsaistē un pielāgojot FIS darbību izvirzītiem mērķiem;
- analizējot statistiskus datus par sistēmas darbību, šādā veidā izvēloties uz noteiktiem kritērijiem balstītus sistēmas stāvokļus, definējot FIS zināšanu bāzi;
- pielietojot ģenētisko pieeju, kad noteiktiem kritērijiem definētās derīguma funkcijas (angl. *Fitness function*) vērtības virza ģenētisko modifikāciju procesu, nonākot pie vēlamā rezultāta – FIS.

Šī promocijas darba autora pielietotais empīriskais FIS zināšanu bāzes izvēles veids ir ātrs un vienkāršs. Šāds zināšanu bāzes izveides paņēmiens var tikt piemērots, kad zināšanu bāze ir neliela un ieejas/izejas parametru skaits ir neliels.

#### **4.1. Piederības funkciju deklarāciju modifikācijas**

Ir vērts pievērsties ieejas parametru lingvinistisko mainīgo piederības funkciju deklarāciju izmaiņu ietekmei uz fazi-CAC algoritma darbības efektivitāti. Tieši piederības funkciju deklarāciju izmaiņas var sniegt ievērojamas fazi-CAC algoritma darbības rakstura izmaiņas, saglabājot neskartu zināšanu bāzi, kā arī nodrošināt plašu modifikāciju spektru.

Atgriezīsimies pie nodaļā 3.1.5 aprakstītā pesimistiskā scenārija – fazi-CAC algoritms strādā apstākļos, kad jaunu LSP konekciju intensitāte ir daudzkārt lielāka nekā linka kapacitātes spējas un fona trafiks ir ar stipri izteiktu uzliesmojošo dabu, tādā veidā stipri ietekmējot visus QoS parametrus. Šāda tīkla uzvedība prasa mainīt trafika vadības politikas iestatījumus, lai mazinātu QoS parametru degradāciju un uzturētu esošās linka plūsmas, saglabājot tiem nepieciešamās QoS parametru vērtības. Ņemsim par piemēru šī promocijas darba autora piedāvātā fazi-CAC algoritma fazi izvedumu sistēmu un modificēsim tās ieejas parametru piederības funkciju deklarācijas ar nolūku padarīt fazi-CAC lēmumu politiku vairāk selektīvu un nepieļaut papildu LSP tuneļu izveidi datu plūsmu transportam. Šādā veidā varam nodrošināt datu sakaru linkam spēju uzturēt nepieciešamos QoS parametrus definētās degradācijas robežās.

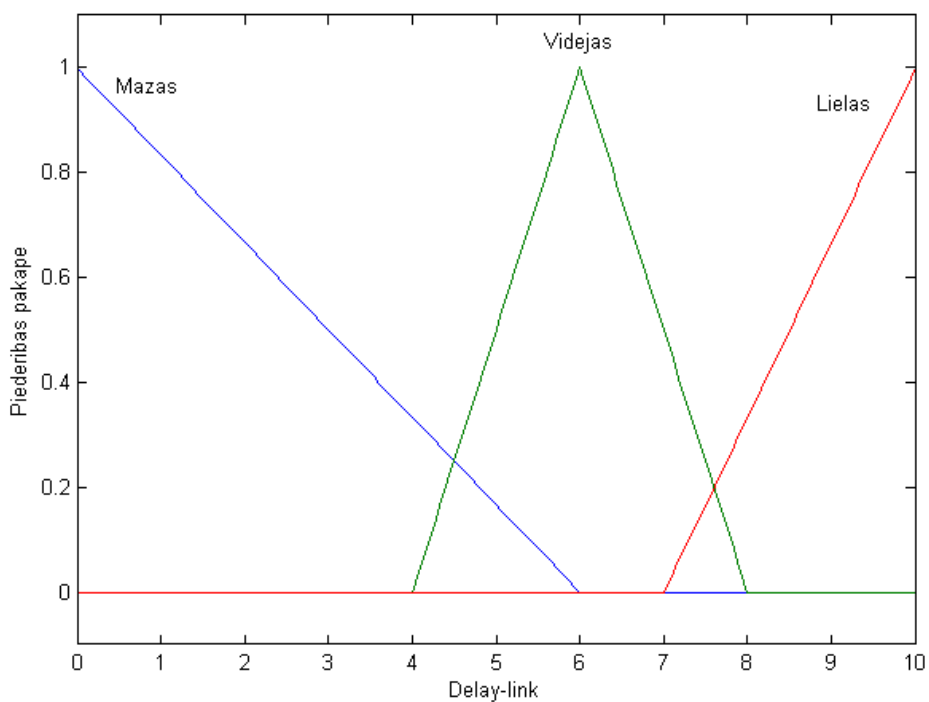
Tātad šī eksperimenta mērķis ir mainīt ieejas parametru lingvinistisko mainīgo piederības funkciju deklarācijas pētamajā fazi-CAC sistēmā ar nolūku modificēt kopējo fazi-CAC algoritma lēmuma pieņemšanas raksturu. Tas dos iespēju palielināt fazi-CAC selektivitāti attiecībā uz jauno LSP tuneļu izveidi datu plūsmu transportam caur datu linku, nemainot IF-THEN kārtulu zināšanu bāzi.

Mainot ieejas parametru lingvinistisko mainīgo piederības funkciju deklarācijas, ir iespējams uzsvērt vai arī pretēji – mazināt kāda no lingvinistiskā mainīgā ietekmi uz IF-THEN zināšanu bāzē definēto kārtulu. Šādā veidā varam ietekmēt izvada mainīgo sniegto vērtību pēc defazifikācijas.

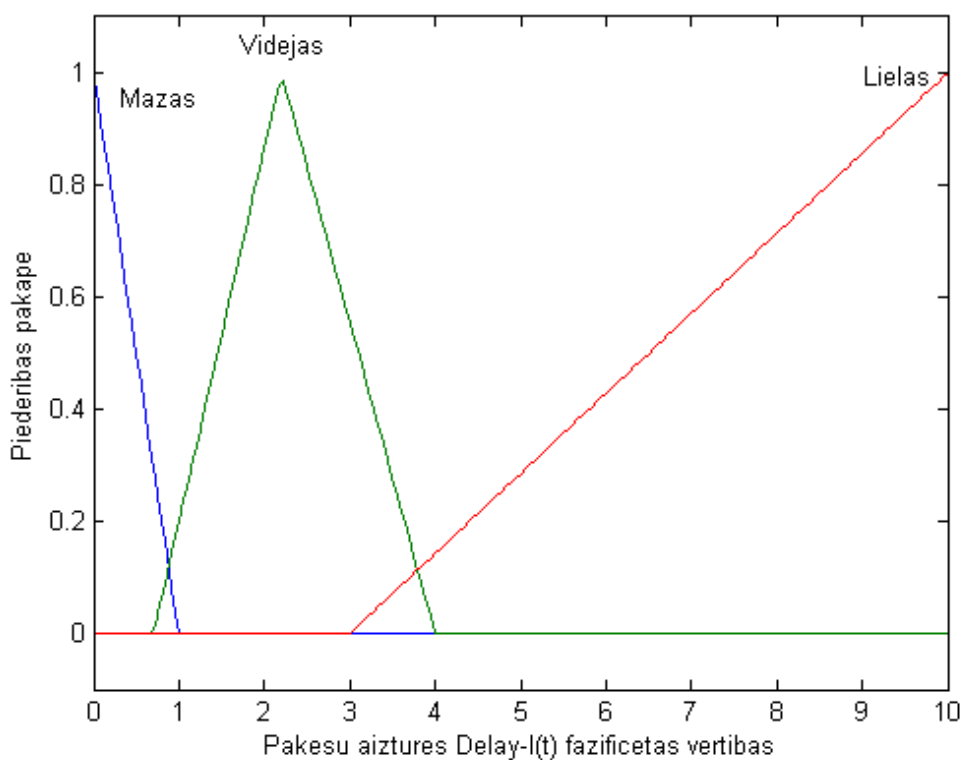
Šī promocijas darba autora pielietotā fazi-CAC algoritma fazi izvedumu sistēma pielieto trīs ieejas parametrus:

- a. pieprasītās plūsmas lieluma –  $Bw_{rq}(t_i)$  [Mb/s];
- b. aizturēm tīklā –  $Delay_l(t_i)$  [ms];
- c. QoS klases (jeb līmeņa) pieprasījuma –  $QoS_c(t_i)$ .

Tātad, lai padarītu fazi-CAC vadības politiku vairāk selektīvu, mēs varam modificēt kādu no iepriekš dotās fazi izvedumu sistēmas ieejas parametru lingvinistisko mainīgo piederības funkciju deklarācijām. Šajā gadījumā, lai padarītu fazi-CAC algoritma darbību vairāk selektīvu un mazinātu QoS parametru degradāciju, tiks izanalizēta  $Delay_l(t_i)$  mainīgā ietekme uz lēmuma pieņemšanas loģiku, mainot tā deklarāciju šādā veidā (sk. att. 4.1 un 4.2).



4.1. att.  $Delay\_l(t_i)$  ieejas parametra lingvistisko mainīgo piederības funkciju deklarācijas pirms modifikācijas

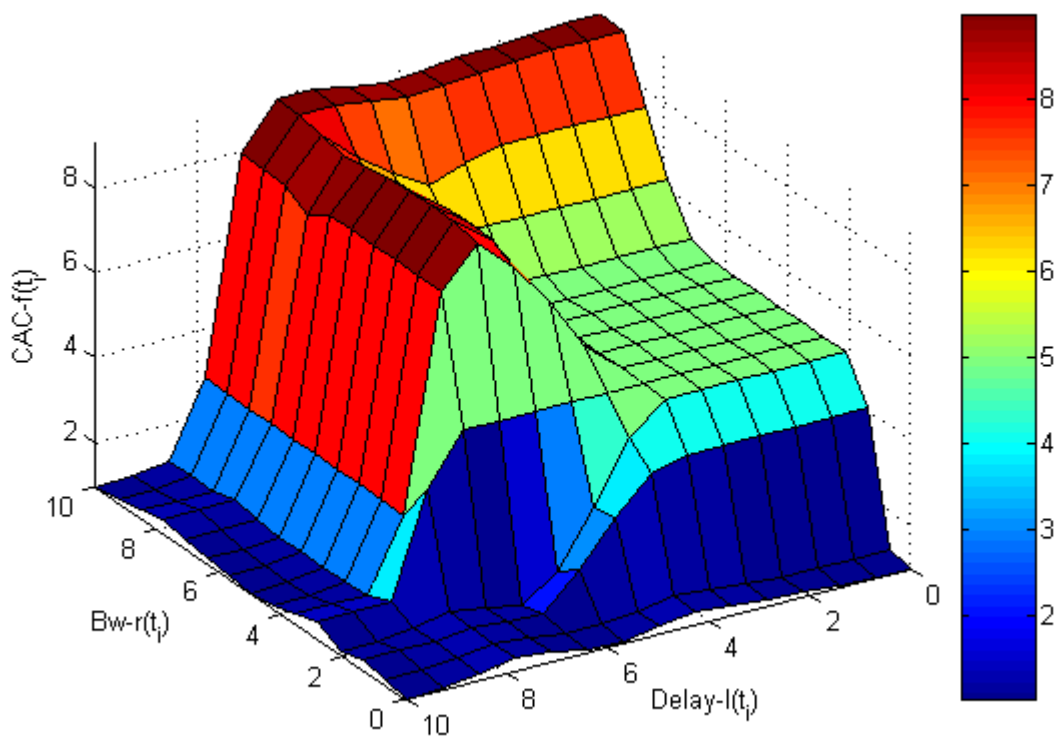


4.2. att.  $Delay\_l(t_i)$  ieejas parametra lingvistisko mainīgo piederības funkciju deklarācijas pēc modifikācijas

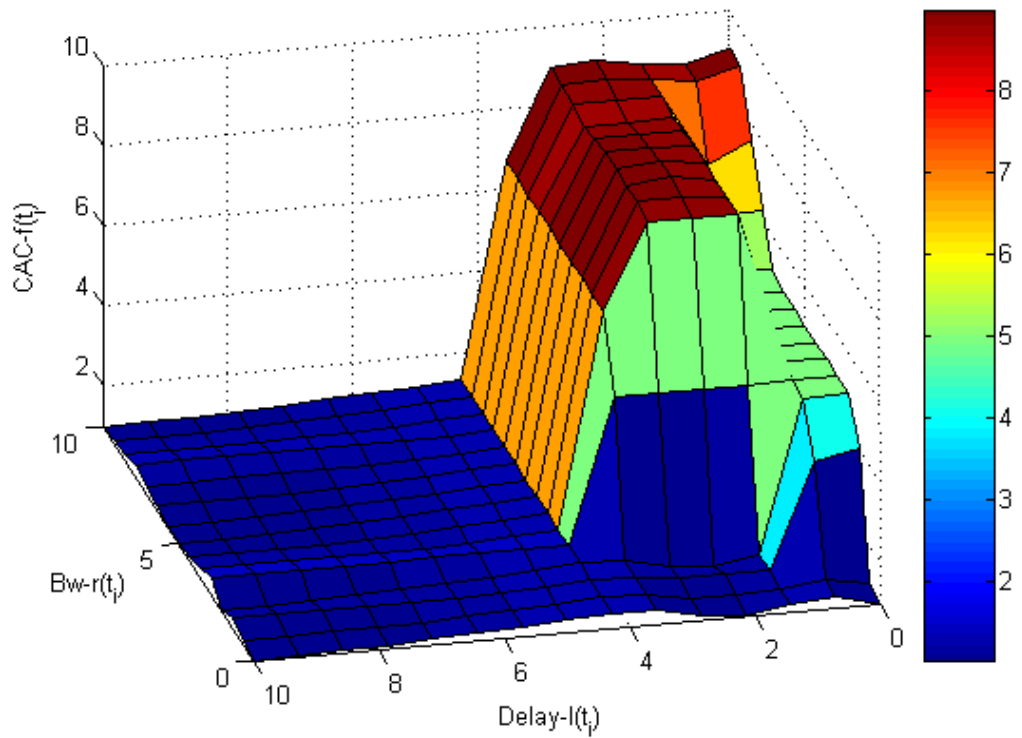
Rezultātā lielākā daļa fazificēto  $Delay\_l(t_i)$  ieejas mainīgā vērtību fazi izvedumu laikā tiks klasificētas kā „lielas”. Tas savukārt nozīmē, ka tās IF-THEN kārtulas, kas definē konkrētās fazi algoritma atbildes, gadījumā, kad  $Delay\_l(t_i)$  ir „lielas”, kļūs noteicošās lēmuma pieņemšanas brīdī.

Mainīgais  $Delay\_l(t_i)$  tika izvēlēts, jo tam ir definēta liela ietekme konkrētās fazi sistēmas IF-THEN kārtulu zināšanu bāzē un tas ļoti labi attēlo veikto izmaiņu ietekmi uz kopējo fazi-CAC fazi izvedumu sistēmas darbības efektivitāti.

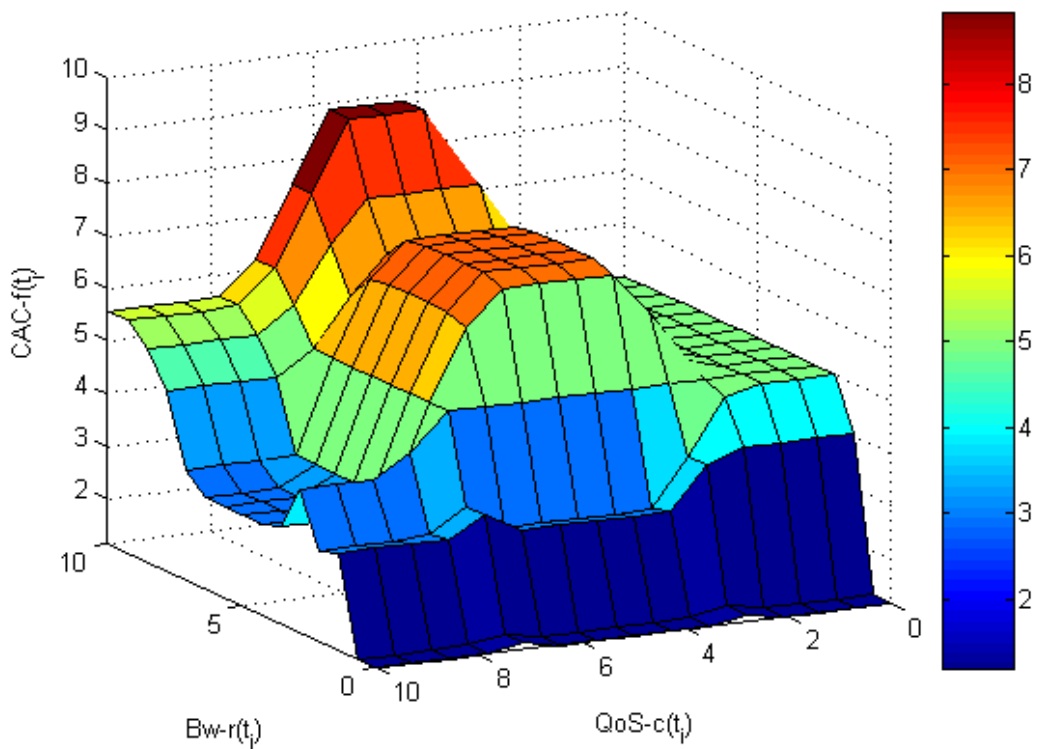
Tālāk ir doti modificētās fazi izvedumu sistēmas lēmumu virsmu attēli (att. 4.3, 4.4, 4.5, 4.6, 4.7 un 4.8), kas parāda lēmumu virsmas pirms un pēc lingvinistisko mainīgo piederības funkciju modifikācijas.



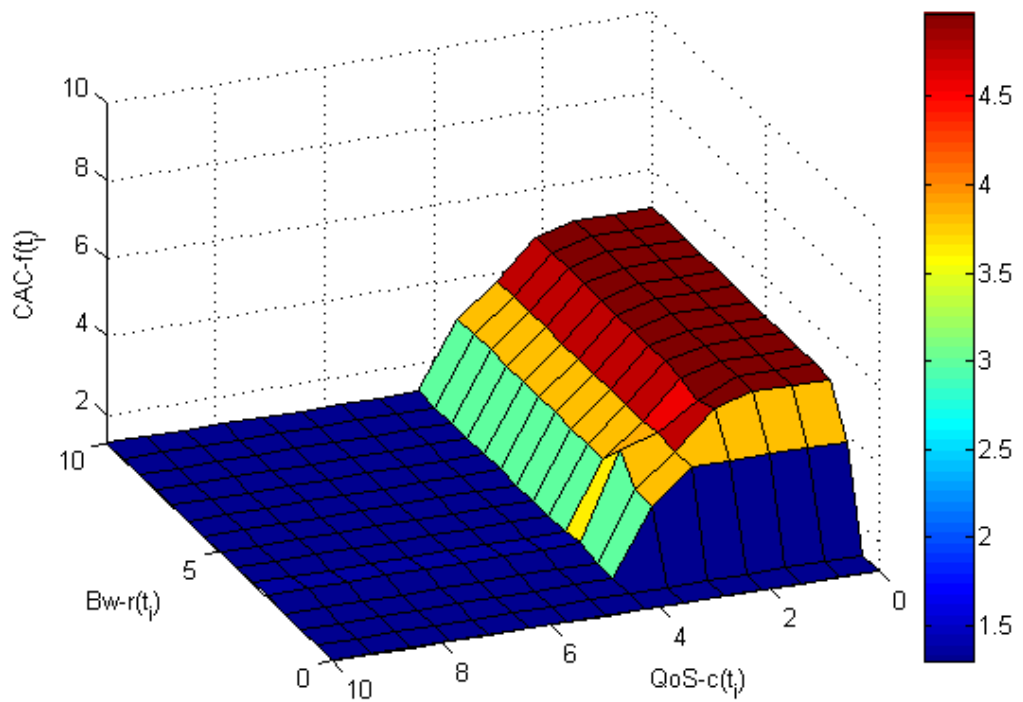
4.3. att.  $Delay\_l(t_i)$  un  $Bw\_r(t_i)$  ieejas mainīgo fazi-CAC lēmuma pieņemšanas virsma pirms  $Delay\_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas



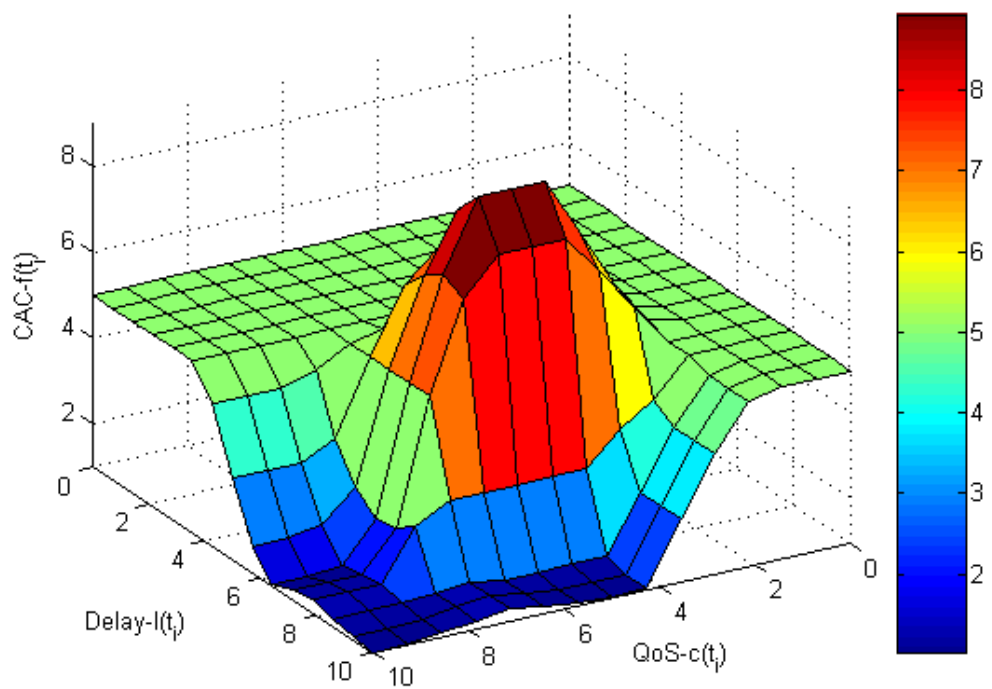
4.4. att.  $Delay\_l(t_i)$  un  $Bw\_r(t_i)$  ieejas mainīgo fazi-CAC lēmuma pieņemšanas virsma pēc  $Delay\_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas



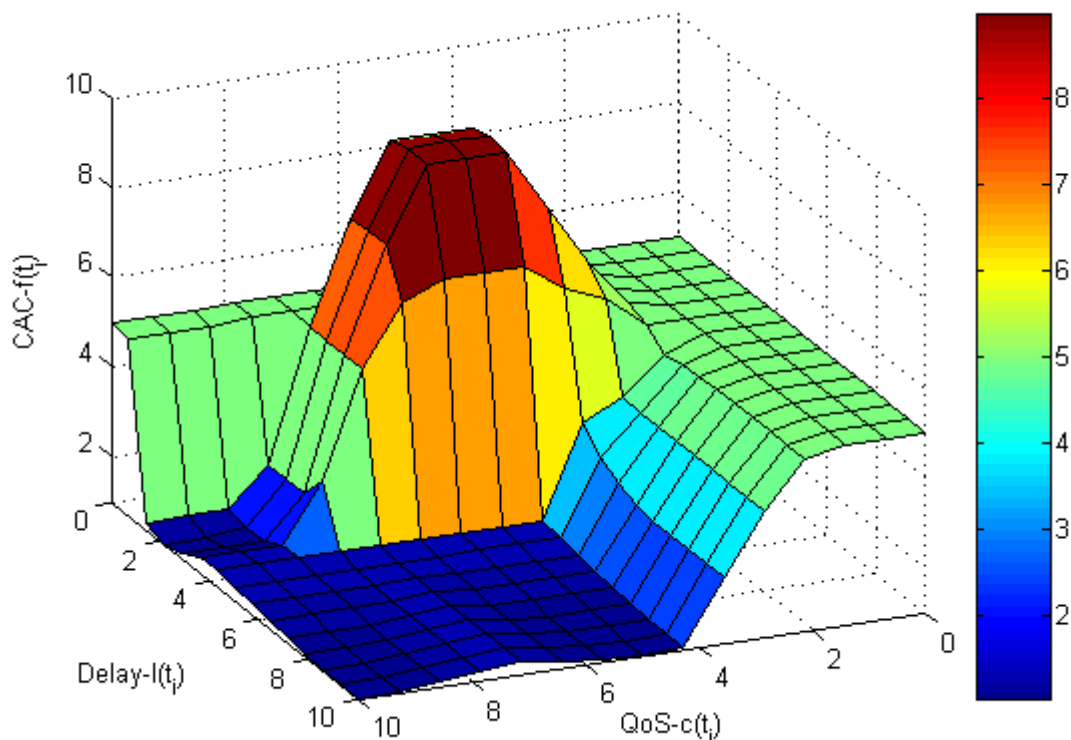
4.5. att.  $QoS\_c(t_i)$  un  $Bw\_r(t_i)$  ieejas mainīgo fazi-CAC lēmuma pieņemšanas virsma pirms  $Delay\_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas



4.6. att.  $QoS_c(t_i)$  un  $Bw_r(t_i)$  ieejas mainīgo fazi-CAC lēmuma pieņemšanas virsma pēc  $Delay_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas



4.7. att.  $QoS_c(t_i)$  un  $Delay_l(t_i)$  ieejas mainīgo fazi-CAC lēmuma pieņemšanas virsma pirms  $Delay_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas

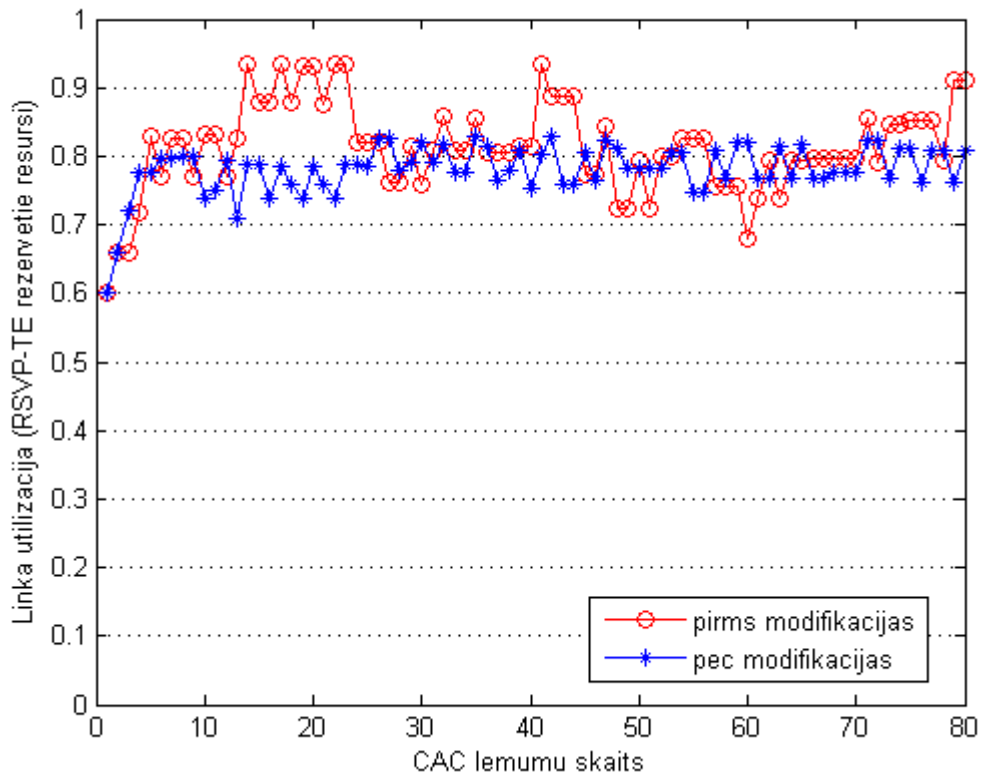


4.8. att.  $QoS\_c(t_i)$  un  $Delay\_l(t_i)$  ieejas mainīgo fazi-CAC lēmuma pieņemšanas virsma pēc  $Delay\_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas

Kā redzams no iepriekš dotiem lēmumu virsmu attēliem,  $Delay\_l(t_i)$  modifikācija padarīja fazi-CAC fazi izvedumu sistēmu daudz selektīvāku attiecībā uz jaunu LSP plūsmu pieslēgumu mēģinājumiem. Arī gadījumā, kad pakešu zudumi ir salīdzinoši mazi, IZ-THEN kārtulu zināšanu bāzes nosacījumi, kas definē lēmumus lielu aizturu gadījumā, sāk spēlēt noteicošu lomu. Rezultātā tiek noraidīts lielāks skaits jaunu LSP plūsmu pieslēgumu mēģinājumu.

Uzliesmojošā trafika gadījumā, tūlītēji mazinot RSVP-TE rezervēto resursu daudzumu linkā, ir iespējams nodrošināt QoS parametru degradācijas mazināšanu. Rezultātā fazi-CAC kontrole pie noteiktiem apstākļiem var nodrošināt linka utilizācijas „apmaiņu” pret QoS parametru saglabāšanu pieļaujamās robežās, nemainot IF-THEN kārtulu zināšanu bāzi.

Attēlā 4.9 ir attēlota linka utilizācija, pielietojot fazi-CAC pesimistiska scenārija gadījumā – pirms un pēc  $Delay\_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas. Kā var redzēt no attēla, selektivitātes paaugstināšana mazināja RSVP-TE rezervēto resursu daudzumu.



4.9. att. Linka utilizācija / LSP tuneļu RSVP-TE rezervēti resursi

$Delay_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijas tika veiktas ar nolūku reducēt QoS parametru degradāciju līdz nodaļā 2.7.2 definētiem ierobežojumiem. Pesimistiska scenārija gadījumā, modificējot lingvinistisko mainīgo piederības funkcijas, dominējošā Pareto fronte (sk. nodaļu 2.7.1) bija saistīta ar pakešu zudumu pieaugumu, jo īslaicīgu uzliesmojumu dēļ  $LER_{in}$  ieejas buferis tika momentāni pārpildīts, tādā veidā novedot pie īslaicīgiem pakešu zudumu uzliesmojumiem. Vidējie eksperimenta rezultāti 3000 lēmumiem ir parādīti tabulā 4.1. Varam redzēt, ka pie utilizācijas līmeņa 0,79 pakešu zudumu vērtība jau sasniedz 0,15%, kas ir definēta QoS parametru degradācijas robeža.

4.1. tabula

Eksperimenta rezultāti 3000 lēmumiem vidējās parametru vērtības

Eksperimenta rezultāti 3000 lēmumiem vidējās parametru vērtības		
Parametrs	Fazi-CAC pirms modifikācijas	Fazi-CAC pēc modifikācijas
Linka vidējā utilizācija (RSVP-TE rezervētie resursi)	0,82	0,79
Pakešu zudumi	3,90 %	0,15 %
Trīce	1,3 ms	0,6 ms
Bufera aizpildījums	14 packet	5 packet
Pakešu aiztures	46 ms	29 ms

Analizējot atteikto plūsmu skaitu (tabula 4.2), varam redzēt, ka fazi-CAC risinājums pēc  $Delay_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijām dod rezultātā  $\approx 17\%$  atteikto plūsmu skaita pieaugumu.

4.2. tabula

Atteikto LSP plūsmu skaits

Atteikto LSP plūsmu skaits		
Lēmumu skaits	Fazi-CAC pirms modifikācijas	Fazi-CAC pēc modifikācijas
500	312	378
1000	679	769
2000	1348	1583
3000	1987	2343

4.3. tabula

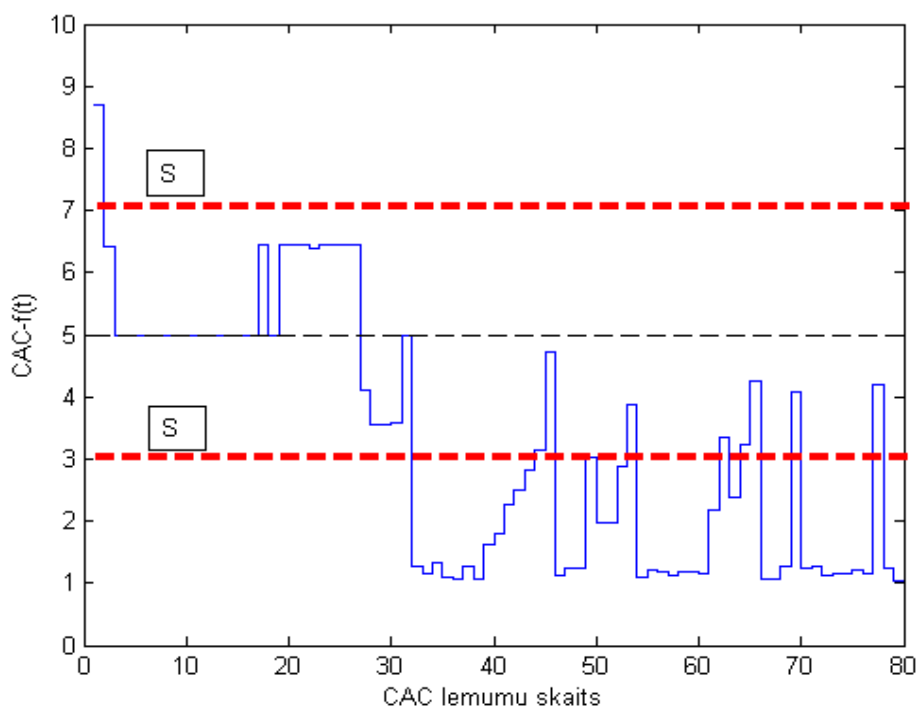
Fazi-CAC pieslēgtas LSP plūsmas ar papildu resursu rezervēšanu

fazi-CAC pieslēgtas LSP plūsmas ar papildu resursu rezervēšanu		
Lēmumu skaits	Fazi-CAC pirms modifikācijas	Fazi-CAC pēc modifikācijas
500	19	10
1000	32	17
2000	67	41
3000	98	54

Tabulā 4.3 ir parādīti rezultāti fazi-CAC darbībai, pieslēdzot LSP plūsmas ar papildu resursu rezervēšanu. Fazi-CAC risinājums pēc  $Delay_l(t_i)$  lingvinistisko mainīgo piederības funkciju modifikācijām, dod rezultātā  $\approx 55\%$  LSP plūsmu pieslēgumu samazinājumu, kuras tiek pieslēgtas ar papildu resursu rezervēšanu. Fazi-CAC palielināta LSP plūsmu pieslēgumu selektivitāte veicina lielāku LSP plūsmu atteikumu skaitu un QoS prasīgo LSP plūsmu pieslēgumu samazinājumu. Tādā veidā tiek nodrošināta QoS parametru saglabāšana pieļaujamās robežās.

## 4.2. Lēmuma pieņemšanas sliekšņa modifikācijas

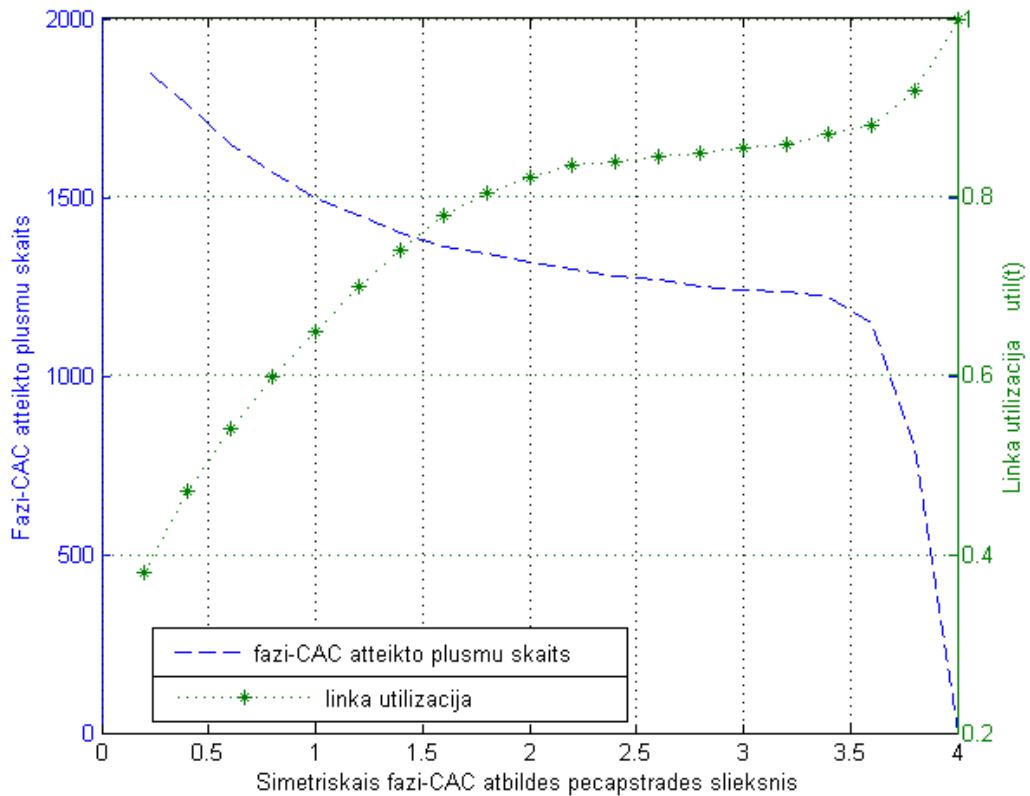
Dotā pieeja var tikt pieskaitīta pie fazi-izvedumu sistēmas atbildes pēcapstrādes metodes un tiešā mērā neietekmē fazi interfeisa sistēmas struktūru. Kaut gan šāds fazi-CAC faziloģiskas izvedumu sistēmas atbildes pēcapstrādes paņēmieni autora atrastajā literatūrā nekur nav pieminēti, tas, pēc autora domām, dod iespēju ātri un efektīvi tiešsaistes režīmā nodrošināt fazi-CAC algoritma adaptāciju pie mainīgiem datu pārraides apstākļiem. Šādā veidā, nemainot fazi-CAC fazi izvedumu sistēmas darbību, var momentāni veikt kopējo trafika vadības politikas maiņu.



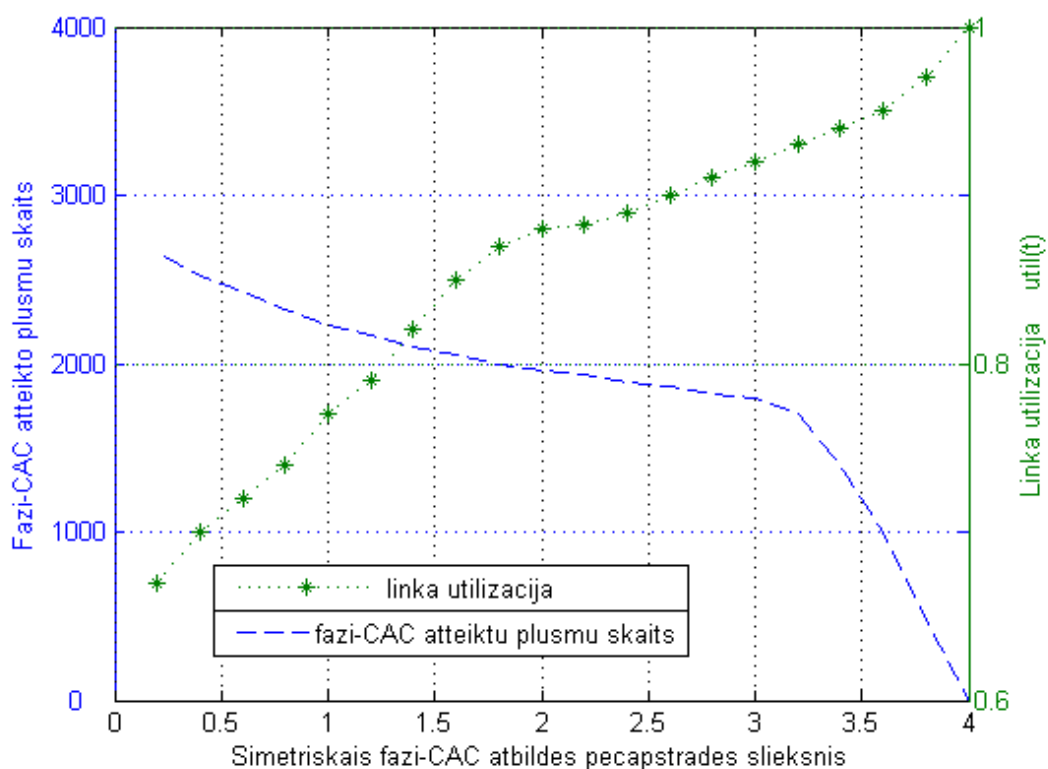
4.10. att. Sliekšņa  $s$  vērtības un  $CAC_f(t_i)$  vērtības lēmumu pieņemšanas momentos

Kā jau tika minēts nodaļā 2, kas apskatīja kopējo autora piedāvāto fazi-CAC risinājuma darbību, tad fazi-CAC algoritma atbildes  $CAC_f(t_i)$  ir definētas vērtību apgabalā no 0 līdz 10. Fazi-CAC atbildes  $CAC_f(t_i)$  pēcapstrāde nodrošina tās pārbaudi attiecībā uz kādu noteiktu sliekšņa vērtību, kas atšķir LSP lēmumus par pieslēguma atteikumu, atļauju un atļauju ar papildu resursu rezervēšanu.

Attēlā 4.11 ir parādīti simulācijas rezultāti 3000 fazi-CAC lēmumu gadījumā, mainot simetrisko sliekšņa  $s$  (Att. 4.10.) vērtību nodaļā 3.1.5 aprakstītā optimistiskā scenārija gadījumā. Savukārt attēlā 4.12 ir parādīti simulācijas rezultāti 3000 fazi-CAC lēmumu gadījumā pesimistiska scenārija gadījumā, kas ir aprakstīts nodaļā 3.1.5.



4.11. att. Sliekšņa  $s$  vērtības, linka utilizācija un fazi-CAC atteikto plūsmu skaits



4.12. att. Sliekšņa  $s$  vērtības, linka utilizācija un fazi-CAC atteikto plūsmu skaits

Attēlotie rezultāti parāda vidējās linka utilizācijas 3000 CAC lēmumu gadījumā un atteikto LSP plūsmu skaita atkarību no izvēlētās simetriskā sliekšņa vērtības.

No rezultātiem ir redzams, ka, palielinot sliekšņa vērtību, tiek paplašināta „Atļaujas” zona, daļēji ignorējot fazi izvedumu sistēmā definētās zināšanu bāzes definīcijas un piederības funkciju iestatījumus. Vienlaicīgi tiek sašaurināta „Atteikumu” un „Atļaujas ar papildu resursu rezervēšanu” zona. Šādā veidā ir iespējams momentāni mainīt fazi-CAC algoritma trafika vadības politikas virzienu un, piemēram, nodrošināt īslaicīgu selektivitātes pastiprināšanu attiecībā uz jaunu LSP plūsmu atteikumiem. Tomēr, atstājot pārāk lielu  $CAC_f(t_i)$  vērtību nogriezni CAC lēmumam „atteikt” tiks nepilnīgi utilizēts links. Atstājot pārāk lielu  $CAC_f(t_i)$  vērtību nogriezni CAC atbildei „atļaut”, links resursi tiks utilizēti pilnīgāk vienlaicīgi ar QoS parametru nepieļaujamu degradācijas pakāpi. Savukārt, atstājot pārāk lielu  $CAC_f(t_i)$  vērtību nogriezni CAC atbildei „atļaut ar papildu resursu rezervēšanu”, „drošības spilvena” īpatsvars ierobežotos linka resursos rezultātā dos nepamatoti zemu linka utilizācijas pakāpi.

Attēlā 4.10 ir redzams, ka selektīvākas fazi-CAC fazi izvedumu sistēmas gadījumā fazi-CAC atbildes  $CAC_f(t_i)$  vērtības vairāk sliecas uz atteikumiem un dod vērtības, kas arī tuvojas pie  $CAC_f(t_i)$  robežvērtības „1”

Ir jāatzīmē, ka katras konkrētās fazi izvedumu sistēmas, kā arī tīkla konfigurācijas un tā noslodzes gadījumā fazi-CAC atbildes pēcapstrādes sliekšņa izmaiņas būs savā ziņā unikālas. Iespējams, ka fazi-CAC vadības sistēmu pamatā ir vērts veidot kaut ko līdzīgu kodeku pakām, kas pašlaik tiek pielietotas personālo datoru multimediju aplikāciju darbības nodrošināšanai. Attēlos 4.11 un 4.12 attēloti rezultāti kalpotu par pamatu vadības „kartei”, kas spētu nodrošināt sistēmas momentāno pielāgošanu pie ārējiem apstākļiem, tādā veidā nepieļaujot QoS parametru nekontrolējamo degradāciju.

Tomēr ir jāuzsver, ka dotās pēcapstrādes metodes pielietojums var kalpot vienīgi kā īslaicīgās iedarbības instruments, kas ir ātrs un neprasa fazi-CAC fazi izvedumu sistēmas tiešās modifikācijas. Tas var tikt pielietots īslaicīgu trafika plūsmu uzliesmojumu gadījumā, momentāni mazinot jauno LSP plūsmu pieslēgumus ar nolūku saglabāt QoS parametrus pieļaujamās robežās. Lai precīzi koriģētu fazi-CAC algoritmu ilglaicīgai darbībai, ir jādefinē trafika vadības politikas konkrēto fazi izvedumu sistēmu veidā, kas var tikt glabātas un modificētas maršrutētājos un pielietotas atkarībā no ārējo apstākļu prasībām.

Varam arī iztēloties situāciju, kad trafika vadības paradigma balstās uz multiaģentu sistēmu ar „menedžera aģentu” un kādu noteiktu skaitu „vadības aģentu” un „zondēšanas aģentu”. „Vadības” un „zondēšanas” aģenti atrodas maršrutētājos un nodrošina LSP plūsmu CAC kontroli un tīkla veiktspējas parametru mērījumu. Kādu noteiktu tīkla segmentu „novēro” „menedžera aģents”, saņemot no „zondēšanas aģentiem” informāciju par maršrutētāju vai konkrētu tīkla linku raksturojošiem parametriem. Šī informācija tiek analizēta un tiek pieņemts lēmums „vadības aģentam” vai arī vairākiem aģentiem nosūtīt jaunu vadības politiku fazi izvedumu sistēmas veidā. Faktiski šādā gadījumā tiek nosūtīti fazi izvedumu sistēmas galvenie raksturojošie parametri, tādi kā piederības funkciju deklarācijas, zināšanu bāzes izmaiņas utt.

Šāda tipa risinājumus un pētījumus šī darba autors definē kā turpmāko pētījumu iespējamo novirzienu.

## 5. SECINĀJUMI

Promocijas darbā tika izstrādāta uz faziloģiku balstīta CAC risinājuma izveides metode, kas dod iespēju veidot fazi-CAC vadību MPLS-TE tīklos ar RSVP-TE protokolu. Autora piedāvātais fazi-CAC risinājums ir lietojams arī GMPLS tīklu vadības plaknē. Fazi-CAC svarīgākā sastāvdaļa ir uz faziloģiku balstīta izvedumu sistēma un zināšanu bāze IF-THEN kārtulu veidā. Atšķirībā no klasiskā sliekšņa-CAC, kas pielieto 2 CAC lēmumu: „atļaut” un „atteikt”, promocijas darba autora piedāvātais CAC risinājums pielieto 3 CAC lēmumus:

- a.  $CAC_f(t_i) > s_1$ , kas nosaka CAC lēmumu „Atteikt”;
- b.  $s_1 > CAC_f(t_i) > s_2$ , kas nosaka CAC lēmumu „Atļaut”;
- c.  $CAC_f(t_i) > s_2$ , kas nosaka CAC lēmumu „Atļaut ar papildu resursu rezervēšanu”.

Promocijas darba autora piedāvātais fazi-CAC risinājums tiek paplašināts ar CAC lēmumu „Atļaut ar papildus resursu rezervēšanu”. Rezultātā pie noteiktiem linka QoS parametriem un aplikācijas QoS prasībām prioritārās LSP plūsmas tiek pieslēgtas ar papildu resursu rezervēšanu. Šādu lēmumu pamatā ir fazi-CAC selektīvā darbība un spēja vienlaicīgi analizēt vairākus QoS parametrus, pieņemot CAC lēmumu. Rezultātā augstas QoS prasības kombinācijā ar zemām pakešu aizturēm un lielu  $Bw_r(t_i)$  vērtību nodrošina t.s. „drošības spilvenu” ne tikai no jauna pieslēgtām LSP plūsmām, bet arī visām linkā esošajām konekcijām. Ir svarīgi uzsvērt, ka šādas trafika vadības politikas veidotais „drošības spilvens” ir tieši proporcionāls prioritāro LSP plūsmu skaitam linkā un ļauj nodrošināt QoS parametru saglabāšanu pieļaujamās robežās. Rezultātā  $util(t_i)$ , kas tiek izrēķināta balstoties uz RSVP-TE rezervētiem resursiem, var uzrādīt lielāku vidējo vērtību nekā  $util_r(t_i)$  jeb reāli nomērītā linka utilizācija.

Klasiskais sliekšņa-CAC algoritms uzrāda lielāku vidējo utilizācijas līmeni salīdzinājumā ar šī promocijas darba autora piedāvāto fazi-CAC risinājumu. Fazi-CAC selektīvā darbība, „upurējot” daļu no linka utilizācijas, ļauj nodrošināt stabilus QoS parametrus, kā arī selektīvus jaunu LSP plūsmu pieslēgumus.

Veidojot fazi-CAC FIS, kas balstās uz šī promocijas darba autora piedāvātiem ieejas un izejas parametriem, ir jāatrod

- fazi-CAC ieejas un izejas parametru lingvinistisko mainīgo skaits;
- piederības funkciju definīcijas;

- IF-THEN kārtulu zināšanu bāze;
- fazi izvedumu sistēmas tips;
- ieejas parametru fazifikācijas līknes;
- izejas parametra defazifikācijas metode;
- fazi-CAC atbildes pēcapstrādes metode.

Šī uzdevuma risināšanai labi noder heuristikā atlasas pētījumu metode, kas dod iespēju atrast vēlamo FIS struktūru, nepielietojot komplicētas FIS definīcijas metodes.

Ir svarīgs arī skaitļošanas laiks – tāda uzdevuma kā CAC vadība efektīva nodrošināšana prasa ātrus skaitļošanas risinājumus. Lai dotais fazi-CAC vadības risinājums būtu spējīgs vadīt komerciālus produktus, ir jānodrošina programmas koda optimizācija. Jāpāriet arī uz vienotu un ātrdarbīgu programmēšanas valodu tādu kā C.

Eksperimentālā fazi-CAC realizācija parāda, ka

- linka utilizācija (RSVP-TE rezervētie resursi), pielietojot fazi-CAC,
  - o samazinās par  $\approx 10\%$  optimistiska scenārija gadījumā,
  - o samazinās par  $\approx 15$  pesimistiska scenārija gadījumā;
- pakešu aiztures starp *LERin* un *LERout*, pielietojot fazi-CAC,
  - o samazinās  $\approx 2,3$  reizes optimistiska scenārija gadījumā,
  - o samazinās  $\approx 2,6$  reizes pesimistiska scenārija gadījumā;
- pakešu kumulatīvie zudumi starp *LERin* un *LERout*, pielietojot fazi-CAC,
  - o samazinās  $\approx 30$  reizes optimistiska scenārija gadījumā,
  - o samazinās  $\approx 3$  reizes pesimistiska scenārija gadījumā;
- *LERin* bufera aizpildījums pielietojot fazi-CAC,
  - o samazinās  $\approx 8$  reizes optimistiska scenārija gadījumā,
  - o samazinās  $\approx 2$  reizes pesimistiska scenārija gadījumā;
- pakešu aizturu trīce starp *LERin* un *LERout* pielietojot fazi-CAC,
  - o samazinās  $\approx 5$  reizes optimistiska scenārija gadījumā,
  - o samazinās  $\approx 3$  reizes pesimistiska scenārija gadījumā.;

- atteikto LSP plūsmu skaits pielietojot fazi-CAC,
  - o Samazinās par  $\approx 15\%$  optimistiska scenārija gadījumā,
  - o Samazinās par  $\approx 18\%$  pesimistiska scenārija gadījumā;
  
- sliekšņa-CAC,
  - o  $\approx 60\%$  gadījumos pieslēdz LSP plūsmas, kuras tiek atteiktas fazi-CAC gadījumā (optimistiskais scenārijs),
  - o  $\approx 80\%$  gadījumos pieslēdz LSP plūsmas, kuras tiek atteiktas fazi-CAC gadījumā (pesimistiskais scenārijs);
  
- pielietojot fazi-CAC,
  - o tiek pieslēgts vidēji par  $\approx 76\%$  vairāk prioritāro LSP plūsmu optimistiska scenārija gadījumā,
  - o tiek pieslēgts vidēji par  $\approx 60\%$  vairāk prioritāro LSP plūsmu pesimistiska scenārija gadījumā;
  
- pielietojot fazi-CAC,
  - o ar papildus resursu rezervēšanu tie pieslēgtas  $\approx 15\%$  no visām atļautām plūsmām - optimistiska scenārija gadījumā,
  - o ar papildu resursu rezervēšanu tie pieslēgtas  $\approx 10\%$  no visām atļautām plūsmām – pesimistiska scenārija gadījumā. s

Savukārt fazi-CAC adaptācijas mehānismi dod iespēju ātri un efektīvi pielāgoties mainīgai apkārtējai videi, reaģējot uz tās izmaiņās, piemēram, uz uzliesmojošā fona trafika klātbūtni. Mainot fazi-CAC atbildes lēmuma pieņemšanas sliekšni vai modificējot ieejas parametru lingvinistisko mainīgo piederības funkcijas, ir iespējams palielināt vai arī mazināt fazi-CAC FIS selektivitāti attiecībā uz pieslēgtām LSP plūsmām, tādā veidā nodrošinot QoS parametru saglabāšanu pieļaujamās robežās.

Analizējot visus šajā promocijas darbā iegūtos eksperimentu rezultātus, varam secināt, ka, izstrādājot promocijas darbu, ir iegūti šādi galvenie rezultāti.

1. Tika konstatēts, ka, ja RSVP-TE protokola CAC lēmuma pieņemšanas politika MPLS-TE tīklos balstās vienīgi uz linka pieejamo caurlaides joslu, tad šāda pieeja nenodrošina efektīvu trafika vadību MPLS-TE tīklos. Darbā ir atrasts CAC vadības alternatīvs veids, kas balstās uz faziloģisko lēmēj mehānismu un CAC lēmuma pieņemšanas laikā nodrošina vienlaicīgu vairāku QoS parametru analīzi. Tas nodrošina selektīvu LSP tuneļu iestatīšanu attiecībā uz aplikācijas puses QoS prasībām un tīkla QoS raksturojošiem parametriem. Vienlaicīgi tiek nodrošināta QoS parametru saglabāšana pieļaujamās robežās.
2. Uz faziloģiku balstīts CAC risinājums nodrošina mazāku skaitu LSP plūsmu pieslēgumu atteikumu, kā arī lielāku skaitu augsta QoS līmeņa plūsmu pieslēgumu, vienlaicīgi saglabājot QoS parametrus pieļaujamās robežās.
3. Fazi-CAC risinājums ar trim CAC lēmumiem, kas ietver sevī arī LSP plūsmas pieslēgumu ar papildu resursu rezervēšanu, nodrošina proaktīvu trafika vadības politiku, veidojot t.s. „drošības spilvenu” LSP plūsmām, kura „izmērs” ir tieši proporcionāls pieslēgtām augstās prioritātes LSP plūsmām un ļauj saglabāt QoS parametrus pieļaujamās robežās.
4. Fazi-CAC algoritma darbības novērtējums MPLS-TE eksperimentālā tīklā, vadot RSVP-TE protokola resursu rezervēšanu LSP tuneļu iestatīšanas brīžos, parāda, ka arī stipri pārslogotā tīklā uzliesmojošā fona trafika gadījumā fazi-CAC nodrošina selektīvo LSP plūsmu kontroli un mazākus QoS parametru degradācijas lielumus.
5. Tika konstatēts, ka ir iespējams veikt ātru fazi-lēmēj mehānisma atbildes pēcapstrādi, definējot konkrētas fazi-CAC atbildes sliekšņa vērtības, momentāni mainot fazi-CAC algoritma politiku, nemainot ieejas lingvinistisko mainīgo piederības funkcijas un/vai IF-THEN kārtulu zināšanu bāzi.
6. Analizējot iespējamus fazi-CAC lēmēj mehānisma tiešsaites adaptācijas mehānismus, ir konstatēts, ka fazi-CAC algoritma SOM, MOM un LOM izvada piederības funkciju defazifikācijas metodes samazina informācijas daudzumu par IF-THEN kārtulu definētiem lēmumiem un neļauj pielietot fazi-CAC atbildes efektīvu pēcapstrādi.

## Darba praktiskā vērtība

- Ir izveidots fazi-CAC FIS modelis, kā arī ir veikta izveidotā modeļa verifikācija un validācija.
- Ir izveidota praktiskā fazi-CAC realizācija RSVP-TE protokola vadībai MPLS-TE tīklā, kas ir eksperimentāls produkts.
- Eksperimentālais MPLS-TE sakaru sistēmas tīkls ar RSVP-TE protokola faziloģikas balstītu CAC kontroli dod tehnoloģisku pamatu plaša loka pētījumiem par fazi-CAC pielietojumu, saglabājot vēlamos QoS parametrus.
- Uz eksperimentālā MPLS-TE sakaru tīkla bāzes, pielietojot vairāku maršrutētāju fazi-CAC vadību ar attiecīgu starpprogrammatūru (piemēram, JADE), var realizēt vairāku fazi-CAC aģentu sadarbību multiaģentu slēgumā.
- Eksperimentālais tīkls un tam piesaistītais programmatiskais nodrošinājums tiek pielietots bakalaura un maģistra darbu izstrādei.
- Ir izveidota metodika MPLS-TE maršrutētāja darbības korekcijai, nodrošinot RSVP-TE protokola vadību ar fazi-CAC fazi izvedumu sistēmu (FIS).

Rezumējot šo promocijas darbu, gribu uzsvērt, ka iegūtie rezultāti ir daudzsološi. Tie parāda, ka fazi-CAC risinājums RSVP-TE protokola vadībā MPLS-TE tīklos spēj nodrošināt dinamisku un proaktīvu LSP iestatīšanas vadību, saglabājot QoS parametrus definētajās robežās. Autora piedāvātais fazi-CAC vadības risinājums ir praktiski realizēts uz MPLS-TE maršrutēšanas iekārtas, radot eksperimentālo produktu.

Šī promocijas darba autors izvirza šādas nākotnes pētījumu tēmas:

- fazi-CAC multiaģentu slēgumā;
- fazi-CAC adaptācijas mehānismi;
- fazi-CAC FIS struktūras automātiskā ģenerācija;
- fazi-CAC FIS IF-THEN kārtulu zināšanu bāzes automātiskā izveide.

## LITERATŪRA

- 1 Abn, G and Woojik, C. Design and Implementation of MPLS Network Simulator Supporting LDP and CR-LDP // *8th IEEE International Conference on Networks ICON2000*, 2000, lpp. 441 - 446.
- 2 Akadēmiskā terminu datubāze. LZA terminoloģijas komisija. Internets. - <http://termini.lza.lv/term.php?term=robustums&list=robustums&lang=LV>.
- 3 Alandjani, G. and Johnson, E.E. Fuzzy routing in ad hoc networks // *Conference Proceedings of the 2003 IEEE International Performance, Computing, and Communications Conference*, 2003 , lpp. 525 - 530.
- 4 Ali, Z., Sheikh, W., Chong, E.K.P., and Ghafoor, A. A Scalable Call Admission Control Algorithm // *IEEE/ACM Transactions on Networking*, Sēj. 16,Nr. 2, 2008 , lpp. 424 - 434.
- 5 Antonopoulos, A. and Verikoukis, C. Traffic-Aware Connection Admission Control Scheme for Broadband Mobile Systems // *IEEE Communications Letters*, Sēj. 14,Nr. 8, 2010 , lpp. 719 - 721.
- 6 Avresky, D, Shurbanov, V, Natchev, N, Zuccarino, F, and Mehra, P. Optimal Utilization of Equivalent Paths in Computer Networks with Static Routing // *IEEE International Symposium on Network Computing and Applications* , 2001, lpp. 133 - 144.
- 7 Bae, Young, Badr, Youakim, and Abraham, Ajith. A Takagi-Sugeno Fuzzy Model of a Rudimentary Angle Controller for Artillery Fire // *Proceedings of the UKSim 2009: 11th International Conference on Computer Modelling and Simulation*, 2009, lpp. 59 – 64.
- 8 Bardet, G., Lang, G., Oppenheim, A., Philippe, S., Stoev, M.S., and Taqqu. Generators of long-range dependence processes: a survey // *Theory and applications of long-range dependence*, Birkhauser, 2003, lpp. 579 - 623.
- 9 Begen, Ali C. Enhancing the multimedia experience in emerging networks // *Ph.D. Dissertation, Georgia Institute of Technology, Atlanta, GA* , 2006.
- 10 Begen, Ali C. and Altunbasak, Yucel. Estimating packet arrival times in bursty video applications // *IEEE International Conference on Multimedia and Expo (ICME)* , 2005, lpp. 767 - 770.
- 11 Beran, J., Sherman, R., Taqqu, M.S., and Willinger, W. Long-range dependence in variable-bit-rate video traffic // *IEEE Transactions on Communications*, Sēj. 43,Nr. 234, 1995 , lpp. 1566 - 1579.
- 12 Bit-Twist - libpcap-based Ethernet packet generator. Internets. - <http://bittwist.sourceforge.net/>.
- 13 *Bit-Twist tīkla pakešu ģenerātors. Internets. - http://bittwist.sourceforge.net/*.
- 14 Bobrovs, V. and Ivanovs, Ģ. Investigation of Minimal Channel Spacing in HDWDM Systems // *Electronics and Electrical Engineering, Lithuania, Kaunas* , 2009, lpp. 53 - 56.
- 15 Bobrovs, V. and Ivanovs, Ģ. Investigation of Mixed Data Rate and Format Transmission in WDM Networks // *Electronics and Electrical Engineering, Lietuva, Kaunas*, 2008, lpp. 63 - 66.
- 16 Bobrovs, V., Jeļinskis, J., Ivanovs, Ģ., and Lauks, G. Research of Traffic Management in FTTx Optical Communication Systems // *Latvian Journal of Physics and Technical Sciences*, Sēj. 2, 2009, lpp. 41- 55.
- 17 Bobrovs, V., Ozoliņš, O., Ivanovs, Ģ., and Poriņš, J. Realization of HDWDM Transmission System // *International Journal of Physical Sciences* , Sēj. 5,Nr. 5, 2010,

- lpp. 452 - 458.
- 18 Cao, Jin, Cleveland, William S., Lin, Dong, and Sun, Don X. Internet Traffic Tends Toward Poisson and Independent as the Load Increases // *Nonlinear Estimation and Classification*, Springer, New York, 2002. lpp: 1 – 18.
  - 19 Cao, Jin, Cleveland, William S., Lin, Dong, and Sun, Don X. On the Nonstationarity of Internet Traffic // *Proceedings ACM SIGMETRICS '01* , 2001, lpp. 102 - 112.
  - 20 Cappe, O., Moulines, E., Pesquet, J.-C., Petropulu, A.P., and Yang, Xueshi. Long-range dependence and heavy-tail modeling for teletraffic data // *IEEE Signal Processing Magazine*, Sēj. 19,Nr. 3, 2002 , lpp. 14 - 27.
  - 21 Cedric, F. Lam. The Outlook of Broadband Optical Access Networks // *Wireless Optical Communication Conference WOCC2008, Taiwan, Pieejams tiešsaitē: <http://www.wocc.org/wocc2008/PDF/O22.pdf>*, 2008.
  - 22 Chen, Peng, Hongye, Su, Lichao, Guo, and Yu, Qu. Mining fuzzy association rules in data streams // *2nd International Conference on Computer Engineering and Technology (ICCET)*, Sēj. 4, 2010, lpp. 144-153.
  - 23 Chen, Kuan, Huang, Polly, Wang, Guo S, Huang, Chun Y., and Lei, Chin L. On the Sensitivity of Online Game Playing Time to Network QoS // *Proceedings of IEEE INFOCOM'06, Barcelona, Spain* , 2006 , lpp. 1 - 12.
  - 24 Cheong, France, Lai, Richard, and Yong Soo, Kim. The Design of a Fuzzy Logic Controller of an ATM Switch and its Simulation // *Journal of Information Science and Engineering*, Sēj. 23, 2007, lpp. 741 - 755.
  - 25 Chrysostomou, C., Pitsillides, A., Rossides, L., and Sekercioglu, A. Fuzzy logic controlled RED: congestion control in TCP/IP differentiated services networks // In *Soft Computing*. Springer-Verlag, 2003, lpp. 79 – 92.
  - 26 Cisco IOS IP Service Level Agreements User Guide // *Cisco Systems, Products White Paper*, 2005.
  - 27 Cisco IOS IP SLAs Configuration Guide, Release 12.4, Last Updated: December 8, 2005, lpp. 1 - 12.
  - 28 Clegg, Richard. A Practical Guide to Measuring the Hurst Parameter // *International Journal of Simulation: Systems, Science & Technology*, Sēj. 7,Nr. 2, 2006, lpp. 3 - 14.
  - 29 Configuring Cisco IP SLAs UDP Jitter Operation // *Cisco Systems, Inc. Products White Paper* , August, 2006.
  - 30 Davis, T. *The Laboratory for International Fuzzy Engineering Research*. Report Number A083292, Asian Office of Aerospace Research and Development APO AP 96337-0007, 1994.
  - 31 Denise, M., Bevilacqua, Masi, and Fischer, J. Martin. Modeling Internet Service Provider Backbone Networks // *The Telecommunications Review NP2007-026 Noblis*, Sēj. 18, 2007, lpp. 91 - 103.
  - 32 Ding, Liya. *A New Paradigm of Knowledge Engineering by Soft Computing*. World Scientific Publishing Company, lpp. 374, 2001.
  - 33 Dini, Paolo and Cusani, Roberto. A Fuzzy Logic Approach to Solve Call Admission Control Issues in CDMA Systems // *EUSFLAT - LFA*, 2005, lpp. 797 - 802.
  - 34 European Society for Fuzzy Logic and Technology // *WWW mājaslapa, Internets*. - <http://www.eusflat.org/index.htm>.
  - 35 Farooq, A and Gad, M. Applications of Fuzzy Logic in Engineering Problems // *Proc of Annual Conference of IEEE Industrial Electronics Society*, 2001.

- 36 Fuzzy Logic Toolbox for use with Matlab // *User`s Guide, The MathWorks, 244 lpp.*
- 37 Girard, Andre. Broadband and FTTx - What has been done? Where is it going? // *IEEE LEOS NEWSLETTER - Special Feature on FTTx*, Aprīlis 2008, lpp. 4 - 12.
- 38 Haider, Aun, Potter, Richard, and Nakao, Akihiro. Challenges in Resource Allocation in Network Virtualization // *20th ITC Specialist Seminar, Hoi An, Vietnam, 2009*, lpp. 11 - 19.
- 39 Harhira, H.A. Pierre, S. A Mathematical Model for the Admission Control Problem in MPLS Networks with End-to-End delay guarantees // *ICCCN 2007. Proceedings of 16th International Conference on Computer Communications and Networks, 2007*, lpp. 1193 - 1197.
- 40 Hartmann, Dennis. Cisco IP Phone Audio Codecs, November 2009, Cisco, Unified Communications, Global Knowledge, Pieejams tiešsaistē:  
<http://globalknowledgeblog.com/technology/unified-communications/cisco-ip-phone-audio-codecs/>.
- 41 Hayashi, Y. Imura, A. Fuzzy neural expert system with automated extraction of fuzzy If-Then rules from a trained neural network // *First International Symposium on Uncertainty Modeling and Analysis, USA, 1990*, lpp. 489 - 494.
- 42 Hongliang, Li, Guizhong, Liu, Yonhli, Li, and Zhongwei, Zhang. An effective burstiness estimation model for VBR video stream // *International Conference on Multimedia and Expo, Sēj. 2, 2003*, lpp. 57 - 60.
- 43 Hwang, Jin-Ho, Kim, Nak-Po, Ji, Yung-Ha, and Ahn, Tae-Hyo. IMS Centralized Network Architecture towards Convergence Services // *Fifth International Conference on Systems and Networks Communications (ICSNC), 2010*, lpp. 283 - 288.
- 44 Hyperconnectivity and the Approaching Zettabyte Era // *Part of the Cisco® Visual Networking Index (VNI), Internets. - http://www.cisco.com*, June 2 2010.
- 45 Ilyas, M.U. and Radha, H. Long Range Dependence of IEEE 802.15.4 Wireless Channels // *ICC '08. IEEE International Conference on Communications, Beijing, 2008*, lpp. 4261 - 4265.
- 46 INTERNAP SERVICE LEVEL AGREEMENT. Internets. - <http://www.internap.com/best-customer-support/sla/>.
- 47 INTERNAP SERVICE LEVEL AGREEMENT. Pieejams tiešsaistē:  
<http://www.internap.com/best-customer-support/sla/>.
- 48 Internet 2010 in numbers // *Royal Pingdom 2010 year internet statistics. Internets. - http://royal.pingdom.com/2011/01/12/internet-2010-in-numbers/*.
- 49 Ishibuchi, H. and Yamamoto, T. Heuristic extraction of fuzzy classification rules using data mining techniques: an empirical study on benchmark data sets // *IEEE International Conference on Fuzzy Systems, Sēj. 1, 2004*, lpp. 161.
- 50 Ivanovs, Ģ., Ļašuks, I., and Ščemeļevs, A. A Hybrid TDM/WDM-PON System with Fwm-Generated Source of Multiwavelength Optical Signals // *Latvian Journal of Physics and Technical Sciences, Sēj. 5, 2010*, lpp. 3 - 14.
- 51 Jassbi, J.J., Serra, P.J.A., Ribeiro, R.A., and Donati, A. A Comparison of Mandani and Sugeno Inference Systems for a Space Fault Detection Application // *World Automation Congress, WAC '06, Budapest, 2006*, lpp. 1 - 8.
- 52 Jean, J and Diab, B. Defuzzification Methods and New Techniques for Fuzzy Controllers // *Iranian Journal of electrical and computer engineering, Sēj. 3,Nr. 2, 2004*, lpp. 161 - 174.

- 53 Jun, Ye and Xuemin, Shen, Jon, W. Mark. Call Admission Control in Wideband CDMA Cellular Networks by Using Fuzzy Logic // *EEE Transactions on Mobile Computing*, Sēj. 4, Nr. 2, 2005, lpp. 129 – 141.
- 54 Kazi, Khurram. *Optical networking standards: a comprehensive guide for professionals*. Springer , 862 lpp., USA, 2006.
- 55 Kim, Wonho, Sharma, Puneet, Lee, Jeongkeun, Banerjee, Sujata, Tourrilhes, Jean, Lee, Sung-Ju, and Yalagandula, Praveen. Automated and scalable QoS control for network convergence // *Proceedings of the 2010 internet network management conference on Research on enterprise networking* , 2010, lpp. 1 - 6.
- 56 knol.google.com. *Social Media Taken Hostage By Michael Jackson*, Pieejams tiešsaitē: <http://knol.google.com/k/social-media-taken-hostage-by-michael-jackson#>.
- 57 Koucheryavy, Y., Giambene, G., Staehle, D., Barcelo-Arroyo, F., Braun, T., and Siris, V. *Traffic and QoS Management in Wireless Multimedia Networks, COST 290 Final Report, Series: Lecture Notes in Electrical Engineering, Sēj. 31*. Springer, 307. lpp, 2009.
- 58 Kulikovs, M. Petersons, E. Sharkovsky, S. Adaptive traffic measurement for MBAC system // *19th Annual Wireless and Optical Communications Conference (WOCC), Shanghai* , 2010 , lpp. 1 - 5.
- 59 Kuļikovs, M. and Pētersons, E. Pašlīdzīgo ieejas traffika statistisko parametru novērtējums mērījumos bāzētā piekļuves kontrolē // *RTU zinātniskie raksti, Telekomunikācijas un elektronika*, Sēj. 7, Nr. 8, 2008, lpp. 37 - 42.
- 60 Kuļikovs, M. and Pētersons, E. Real-Time Traffic Analyzer for Measurement-Based Admission Control // *Proceeding of The Fifth Advanced International Conference on Telecommunications AICT2009, Venice/Mestre, Italy*, 2009, lpp. 72 - 75.
- 61 Kuļikovs, M., Pētersons, E., and Šarkovskis, S. Integral measurement process of incoming traffic for Measurement-Based Admission Control // *Proceedings, 2010 IEEE Region 8 International Conference on Computational Technologies in Electrical and Electronics Engineering, Krievija, Irkutsk, Listvjanka*, Sēj. 1, 2010, lpp. 183 - 186.
- 62 Lachlan, L. H. and Su-Hsien, Kuan. Improving the robustness of fuzzy logic ATM ABR rate control for large round trip times // *International Conference of Intelligent Signal Processing and Communications, Melbourne, Australia*, 1998, lpp. 10-14.
- 63 Leekwijck, Van. Defuzzification: criteria and classification // *Journal of Fuzzy Sets and Systems*, Sēj. 108, 1999, lpp. 159 - 178.
- 64 Leland, Will, Taqqu, Murad, Willinger, Walter, and Wilson, Daniel. On the Self-Similar Nature of Ethernet Traffic // *IEEE/ACM Transactions on Networking*, Sēj. 2, Nr. 1, 1994, lpp. 1 - 15.
- 65 Lightner, Constance. A Fuzzy Logic Approach to Buffer Management in ATM Networks // *Fuzzy Optimization and Decision Making, Kluwer Academic Publishers, USA*, Sēj. 5, Nr. 1, 2006, lpp. 49 - 69.
- 66 Mamdani, E. H. and Assilian, S. An experiment in linguistic synthesis with a fuzzy logic controller // *International Journal of Man-Machine Studies* , Sēj. 7, Nr. 1, 1975, lpp. 1 - 13.
- 67 McEntegart, Jane. Google Mistakes Michael Jackson's Death for Attack, Pieejams tiešsaitē: <http://www.tomsguide.com/us/Google-Michael-Jackson-Death-Attack,news-4140.html>, 26. jūnijs, 2009.
- 68 Michael Jackson's Death Causes Internet to Slow // *Discovery Magazine blog, Internets*. - <http://blogs.discovermagazine.com/discoblog/2009/06/26/michael-jacksons-death->

causes-internet-to-slow/.

- 69 Michael Jackson's death challenges Google, *IT news Africa, Internets*. - <http://www.itnewsafrika.com/?p=2832>.
- 70 Mirabedini, S.J., Teshnehlab, M., and Rahmani, A.M. FLAR: An Adaptive Fuzzy Routing Algorithm for Communications Networks Using Mobile Ants // *Convergence Information Technology*, ISBN: 0-7695-3038-9, 2007, lpp. 1308 – 1315.
- 71 Mitra, S. and Hayashi, Y. Neuro-fuzzy rule generation: survey in soft computing framework // *IEEE Transactions on Neural Networks*, Sēj. 11, Nr. 3, 2000, lpp. 748 - 768.
- 72 Mohamadlou, H., Ghodsi, R., Razmi, J., and Keramati, A. A method for mining association rules in quantitative and fuzzy data // *International Conference on Computers & Industrial Engineering CIE 2009*, 2009, lpp. 453.
- 73 Msadaa, I.C., Câmara, D., and Filali, F. Scheduling and CAC in IEEE 802.16 Fixed BWNs: A Comprehensive Survey and Taxonomy // *IEEE Communications Surveys & Tutorials*, Sēj. 12, Nr. 4, 2010, lpp. 459 - 487.
- 74 Natsheh, E., Khatun, S., and Jantan, A.B. Adaptive Fuzzy Route Lifetime for Wireless Ad-hoc Networks // *The International Arab Journal of Information Technology*, Sēj. 3, Nr. 4, 2006, lpp. 283 - 290.
- 75 Navara, Mirko and Petrik, Milan. Fuzzy Control – Expectations, Current State, and Perspectives // *Computational Intelligence, Theory and Applications International Conference 9th Fuzzy Days in Dortmund, Germany*, 2006, lpp. 667 – 676.
- 76 Nejad, Hamed Vahdat, Yaghmaee, Mohammad Hossien, and Tabatabaee, Hamid. Fuzzy TCP: Optimizing TCP Congestion Control // *APCC '06. Asia-Pacific Conference on Communications*, 2006, lpp. 1 - 5.
- 77 NNT COMMUNICATIONS SERVICE LEVEL AGREEMENT. Internets. - <http://www.eu.ntt.com/en/products/global-network/transit/sla-of-global-ip-network.html>.
- 78 Nojima, Y. Ishibuchi, H. Interactive genetic fuzzy rule selection through evolutionary multiobjective optimization with user preference // *IEEE symposium on computational intelligence in multi-criteria decision-making MCMD '09*, 2009, lpp. 136 - 141.
- 79 NOKIA SIEMENS NETWORKS CORPORATION. A vision of tomorrow's connected world // *A White Paper on converged networks and flexible business models, Internets*. - [http://www.nokiasiemensnetworks.com/NR/ronlyres/12101186-5339-42F2-99BE-CF9510C700B4/0/WP\\_Vision\\_of\\_Tomorrows\\_Connected\\_World\\_v1.pdf](http://www.nokiasiemensnetworks.com/NR/ronlyres/12101186-5339-42F2-99BE-CF9510C700B4/0/WP_Vision_of_Tomorrows_Connected_World_v1.pdf), 2008.
- 80 Petersons, E. and Jerjomins, R. Self-similar traffic in wireless networks // *Queues. Flows, systems, networks - Mathematical methods of optimisation of telecommunication networks, Belarus, Minsk*, 2005, lpp. 49 - 55.
- 81 Pitsillides, A., Sekercioglu, Y. A., and Ramamurthy, G. Effective control of traffic flow in ATM networks using fuzzy explicit rate marking (FERM) // *IEEE Journal Selected Areas. Communications*, Sēj. 15, Nr. 2, 1997, lpp. 209 – 225.
- 82 QUEST SERVICE LEVEL AGREEMENT. Internets. - [http://www.qwest.com/legal/docs/Qwest\\_iQ\\_Networking\\_SLA\\_\\_09\\_14\\_10\\_.pdf](http://www.qwest.com/legal/docs/Qwest_iQ_Networking_SLA__09_14_10_.pdf).
- 83 QUEST SERVICE LEVEL AGREEMENT. Pieejams tiešsaitē: [http://www.qwest.com/legal/docs/Qwest\\_iQ\\_Networking\\_SLA\\_\\_09\\_14\\_10\\_.pdf](http://www.qwest.com/legal/docs/Qwest_iQ_Networking_SLA__09_14_10_.pdf).
- 84 Raja, S. V. Kasmir and Raj, P. Herbert. Integrated Subset Split for Balancing Network Utilization and Quality of Routing // *World Academy of Science, Engineering and Technology*, Nr. 26, 2007.

- 85 Rea, S. and Pesch, D. Multi-metric routing decisions for ad hoc networks using fuzzy logic // *Wireless Communication Systems*, ISBN: 0-7803-8472-5, 2004, lpp. 403 – 407.
- 86 Reed, I.S., Lee, P.C., and Troung, T.K. Spectral Presentation of Fractional Brownian Motion in N Dimentions and its Properties // *IEEE Transactions on Information Theory*, Sēj. 41 ,Nr. 5, 2002, lpp. 1439 – 1451.
- 87 Resende, R.A., Rossi, S.M., Yamakami, A., Bonani, L.H., and Moschim, E. Traffic engineering with MPLS using fuzzy logic for application in IP networks // *FUZZ - The 12th IEEE International Conference on Fuzzy Systems*, Sēj. 2, 2003, lpp. 1146 - 1151.
- 88 Ruffaldi, Emanuele. „1.2.3 ways of integrating MATLAB with the.NET”, Internets. - <http://www.codeproject.com/KB/dotnet/matlabeng.aspx>.
- 89 Ruffaldi, Emanuele. Matlab Interface Library EngMATLib 2002, Internets. - <http://www.sssup.it/~pit/vai>  
[http://intd.com.br/rodrsilv/PUC/computa%e7%e3o%20evolucion%e1ria/CE-Exemplo\\_tsp/Exemplo\\_tsp\\_062/EngMATLib/](http://intd.com.br/rodrsilv/PUC/computa%e7%e3o%20evolucion%e1ria/CE-Exemplo_tsp/Exemplo_tsp_062/EngMATLib/).
- 90 Runtong, Zhang and Xiaomin, Zhu. Fuzzy Routing in QoS Networks // In *Fuzzy Systems and Knowledge Discovery, Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Sēj. 3614, 2005, lpp. 880 - 890.
- 91 Russinovich, Mark. PsExec v1.98, Internets. - <http://technet.microsoft.com/en-us/sysinternals/bb897553>.
- 92 Shapira, Joseph. Dynamic Enhancement and Optimal Utilization of CDMA Networks // In *Emerging Location Aware Broadband Wireless Ad Hoc Networks*. Springer US, 2005, lpp: 93 - 108.
- 93 Sheng, Ma and Chuanyi, Ji. Modeling Heterogeneous Network Traffic in Wavelet Domain // *IEEE/ACM Transactions on networking*, Sēj. 9,Nr. 5, 2001, lpp. 634 – 649.
- 94 Shounak, Roychowdhury. An inquiry into the theory of defuzzification, Granular computing: an emerging paradigm // *Physica-Verlag GmbH, Vacija*, 2001, lpp. 143 – 162.
- 95 Silvestre-Blanes, J., Almeida, L., Marau, R., and Pedreiras, P. Online QoS Management for Multimedia Real-Time Transmission in Industrial Networks // *IEEE Transactions on Industrial Electronics* , Sēj. 58,Nr. 3, 2011 , lpp. 1061 - 1071.
- 96 Siriwong, K., Lipsky, L., and Ammar, R. Study of Bursty Internet Traffic // *NCA 2007. Sixth IEEE International Symposium on Network Computing and Applications*, Cambridge, MA , 2007, lpp. 53 - 60.
- 97 SPRINT NETWORKS SERVICE LEVEL AGREEMENT. Internets. - [https://www.sprint.net/sla\\_performance.php?network=sl](https://www.sprint.net/sla_performance.php?network=sl).
- 98 SPRINT NETWORKS SERVICE LEVEL AGREEMENT. Pieejams tiešsaitē: [https://www.sprint.net/sla\\_performance.php?network=sl](https://www.sprint.net/sla_performance.php?network=sl).
- 99 Stanojevic, Rade and Shorten, Robert. How expensive is link utilization? // *Proceedings of the 1st EuroFGI international conference on Network control and optimization*, Avignon, France, 2007, lpp. 54 - 64.
- 100 Stevens-Navarro, E., Mohsenian-Rad, A.H., and Wong, V. Connection Admission Control for Multiservice Integrated Cellular/WLAN System // *IEEE Transactions on Vehicular Technology*, Sēj. 57,Nr. 6, 2008 , lpp. 3789 - 3800.
- 101 Stoev, S., Michailidis, G., and Vaughan, J. On Global Modeling of Backbone Network Traffic // *IEEE Proceedings on INFOCOM2010*, 2010, lpp. 1 – 5.
- 102 Sujit, Pant and Keith, Holbert. *Fuzzy Logic in Decision Making and Signal Processing -*

- The teaching online module of Fulton School of Engineering.* Pieejams tiešsaitē:  
<http://enpub.fulton.asu.edu/powerzone/fuzzylogic/index.htm>.
- 103 Takagi, T. and Sugeno, M. "Fuzzy identification of systems and its applications to modeling and control", // *IEEE Transactions on Systems, Man and Cybernetics*, Sēj. 15, 1985, lpp. 116 - 132.
  - 104 Taute, Rika. Modeling of Internet Traffic Data // *SATNAC2006, Western Cape, South Africa*, 2006, lpp. 1 - 10.
  - 105 *Tcpdump pakešu analizātors uz "libpcap" bibliotēkas bāzes. Internets.* - <http://www.tcpdump.org/>.
  - 106 Terrasa, S, Sáez, S, Vila, J, and E, Hernandez-Orallo. Comparing the utilization bounds of IntServ and DiffServ // *2nd International Conference on Performance Modelling and Evaluation of Heterogeneous Networks*, 2004, lpp. 1 - 10.
  - 107 Todinca, Doru, Holban, Stefan, Perry, Philip, and Murphy, John. Fuzzy Logic Based Admission Control for GPRS/EGPRS Networks // *Transactions on Automatic Control and Computer Science*, Sēj. 49, Nr. 63, 2004, lpp. 10 - 15.
  - 108 Tychon, Emmanuel. Advanced Performance Measurement with Cisco IOS IP SLA // *Cisco Systems internal documentation*, October 2006.
  - 109 Vázquez, Enrique, Álvarez-Campana, Manuel, and B. García, Ana. Network Convergence over MPLS // *High Speed Networks and Multimedia Communications, Lecture Notes in Computer Science*, Sēj. 3079/2004, 2004, lpp. 290 - 300.
  - 110 VERIO SERVICE LEVEL AGREEMENT. Internets. - <http://www.verio.com/about/sla/global-ip/>.
  - 111 VERIO SERVICE LEVEL AGREEMENT. Pieejams tiešsaitē:  
<http://www.verio.com/about/sla/global-ip/>.
  - 112 Wai-Ho Au Chan, K.C.C. FARM: a data mining system for discovering fuzzy association rules // *IEEE International Fuzzy Systems Conference FUZZ-IEEE '99*, Sēj. 3, 1999, lpp. 1217.
  - 113 Willinger, Walter and Taqqu, Murad, Erramilli, Ashok. A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks // *Stochastic Networks: Theory and Applications, Royal Statistical Society Lecture Notes Series*, Sēj. 4, 1996, Oxford University Press.
  - 114 Wu, Jin and Djemame, Karim. FAOM: A Novel Active Queue Management with Fuzzy Logic for TCP-Based Interactive Communications // *Lecture Notes in Computer Science - Networking and Mobile Computing*, Sēj. 3619, 2005, lpp. 1198 - 1207.
  - 115 Xue, Yuan, Li, Baochun, and Nahrstedt, Klara. Optimal Resource Allocation in Wireless Ad Hoc Networks: A Price-based Approach // *IEEE Transactions on Mobile Computing*, Sēj. 5, 2004, lpp. 347 - 364.
  - 116 Yeganeh, Hassan, Shakiba, Maryam, and Samie, Mehdi. Optimal Resource Allocation in Next Generation Network Services using Engineering Optimization with Linear Constraint Particle Swarm // *IJCSNS International Journal of Computer Science and Network Security*, Sēj. 8, Nr. 11, 2008, lpp. 324 - 338.
  - 117 *YouTube press statistics. Internets.* - [http://www.youtube.com/t/press\\_statistics](http://www.youtube.com/t/press_statistics).
  - 118 Yuqiang, Li and Qianxing, Xiong. Dijkstra Algorithm Implementing Services Composing Way Selection // *2nd International Workshop on Intelligent Systems and Applications (ISA)*, 2010, lpp. 1 - 4.
  - 119 Zadeh, L.A. A fuzzy-algorithmic approach to the definition of complex or imprecise

- concepts // *International Journal of Man-Machine Studies*, Sēj. 8, 1976, lpp. 246 - 291.
- 120 Zadeh, L.A. A fuzzy-set theoretic interpretation of linguistic hedges // *Journal of Cybernetics*, Sēj. 2, 1972, lpp. 4 - 34.
- 121 Zadeh, L.A. A new direction in AI. Towards a computational theory of perceptions // *AI Magazine*, Sēj. 22,Nr. 1, 2001, lpp. 73 - 84.
- 122 Zadeh, L.A. Calculus of fuzzy restrictions // In Zadeh L.A., Fu K.S., Tanaka K. Shimura M., ed., *Fuzzy Sets and Their Applications to Cognitive and Decision Processes*. Academic Press, New York, 1975, lpp. 1 - 39.
- 123 Zadeh, L.A. From computing with numbers to computing with words – from manipulation of measurements to manipulation of perceptions // *IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications*, Sēj. 45,Nr. 1, 1999, lpp. 105 - 119.
- 124 Zadeh, L.A. Fuzzy logic = computing with words // *IEEE Trans. on Fuzzy Systems*, Sēj. 4, 1996, lpp. 103 - 111.
- 125 Zadeh, L.A. Fuzzy logic and its application to approximate reasoning // *Information Proc.*, Sēj. 74, 1974, lpp. 591 - 594.
- 126 Zadeh, L.A. Fuzzy logic and the calculi of fuzzy rules and fuzzy graphs: a precis // *Multiple Valued Logic*, Sēj. 1, 1996, lpp. 1 - 38.
- 127 Zadeh, L.A. Fuzzy logic, neural networks and soft computing // *Commun. Of the ACM*, Sēj. 37, 1994, lpp. 77 - 84.
- 128 Zadeh, L.A. Fuzzy sets and information granularity // In Gupta M., Ragade R., Yager R., ed., *Advances in Fuzzy Set Theory and Applications*. North Holland, Amsterdam, 1979, lpp. 3-18.
- 129 Zadeh, L. A. Fuzzy Sets. // *Information and Control*, Sēj. 8, 1965, lpp. 338-353.
- 130 Zadeh, L.A. New Frontiers in Fuzzy Logic and Soft Computing // *Proc. Of Fourth Conference, Neural Networks and Their Application, Zakopane, Poland* , 1999, lpp. 1 - 4.
- 131 Zadeh, L.A. On the analysis of large scale systems // In H., Gottinger, ed., *Systems Approaches and Environment Problems*. Vandenhoeck and Ruprecht, 1974, lpp. 23-37.
- 132 Zadeh, L.A. Outline of a computational approach to meaning and knowledge representation based on a concept of a generalized assignment statement // In Thoma M., Wyner A., ed., *Proceedings of the International Seminar on Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, Heidelberg, 1986, lpp. 198 - 211.
- 133 Zadeh, L.A. Outline of a new approach to the analysis of complex systems and decision processes // *IEEE Trans. On Systems, Man, and Cybernetics*, Sēj. SMC-3,Nr. 1, 1973, lpp. 28 - 44.
- 134 Zadeh, L.A. Outline of computational theory of perceptions based on computing with words // In Sinha N.K., Gupta M.M., ed., *Soft Computing and Intelligent Systems: Theory and Applications*. Academic Press, San Diego, New York, Tokio , 2000, lpp. 3 - 22.
- 135 Zadeh, L.A. Similarity relations and fuzzy orderings // *Information Science*, Sēj. 3, 1971, lpp. 177 - 200.
- 136 Zadeh, L.A. Test-score semantics for natural language and meaning representation via PRUF // In B., Rieger, ed., *Empirical Semantics*. Germany, 1981, lpp. 281 - 349.
- 137 Zadeh, L.A. The concept of a linguistic variable and its application to approximate reasoning // *Information Science*, Sēj. 8,Nr. I, 1975, lpp. 199-249.
- 138 Zadeh, L.A. The concept of a linguistic variable and its application to approximate

- reasoning // *Information Science*, Sēj. 8,Nr. II, 1975, lpp. 301 - 357.
- 139 Zadeh, L.A. The concept of a linguistic variable and its application to approximate reasoning // *Information Science*, Sēj. 9,Nr. III, 1975, lpp. 43-80.
- 140 Zadeh, L.A. The role of fuzzy logic in the management of uncertainty in expert systems // *Fuzzy sets and systems*, Sēj. 11, 1983, lpp. 199 - 227.
- 141 Zadeh, L.A. Towards a Theory of Fuzzy Systems // In N, Kalman R.E. and DeClaris, ed., *Aspects of Network and System Theory*. Rinehart and Winston, New York, 1971.
- 142 Zadeh, L.A. Towards the theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic // *Fuzzy Sets and Systems*, Sēj. 90, 1997, lpp. 1 - 4.
- 143 Zhang, S., Yu, F.R., and Leung, V.C.M. Joint connection admission control and routing in IEEE 802.16-based mesh networks // *IEEE Transactions on Wireless Communications*, Sēj. 9,Nr. 4, 2010 , lpp. 1370 - 1379.
- 144 Zhang, Huai-xiang, Zhang, Bo, and Wang, Feng. Automatic Fuzzy Rules Generation Using Fuzzy Genetic Algorithm // *Sixth International Conference on Fuzzy Systems and Knowledge Discovery, China*, 2009, lpp. 107 - 112.
- 145 Zhao, Ning, Vinokurov, Anton, and Mason, Lorne. Traffic Modeling for Agile All-Photonic Network Dimensioning // *ICOIN'09 Proceedings of the 23rd international conference on Information Networking*, 2009, lpp. 1 – 3.
- 146 Zhu, Rongbo and Liu, Xiaozhu. Effective CAC Scheme in IEEE 802.11e Wireless LANs // *Propagation and EMC Technologies for Wireless Communications, International Symposium on Microwave Antenna, Hangzhou* , 2007 , lpp. 164 - 167.
- 147 Гольдштейн, А.Б. and Гольдштейн, Б. С. Технология и протоколы MPLS // BHV-Санкт-Петербург, 2005, 304 lpp.
- 148 Райзберг, Б.А., Лозовский, Л.Ш., and Стародубцева, Е.Б. Современный экономический словарь // 2-е изд., испр. М.: ИИФРА-М, ISBN 5-86225-758-6, 1999, lpp. 479.

## **PIELIKUMI**