

Analiza i modeliranje metode dodjele kapaciteta u višeuslužnim mrežama WFQ

Radičević, Dora

Master's thesis / Diplomski rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:119:200153>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-11-30**



Repository / Repozitorij:

[Faculty of Transport and Traffic Sciences -
Institutional Repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

Dora Radičević

**ANALIZA I MODELIRANJE METODE DODJELE
KAPACITETA U VIŠEUSLUŽNIM MREŽAMA WFQ**

DIPLOMSKI RAD

Zagreb, 2022.

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI
POVJERENSTVO ZA DIPLOMSKI ISPIT

Zagreb, 23. svibnja 2022.

Zavod: **Zavod za informacijsko komunikacijski promet**
Predmet: **Tehnologija telekomunikacijskog prometa II**

DIPLOMSKI ZADATAK br. 6938

Pristupnik: **Dora Radičević (0135247529)**
Studij: **Promet**
Smjer: **Informacijsko-komunikacijski promet**

Zadatak: **Analiza i modeliranje metode dodjele kapaciteta u višeslužnim mrežama WFQ**

Opis zadatka:

Opisati značajke višeslužnih mreža i ulogu mrežnih čvorišta. Analizirati procese u mrežnim čvorištima vezano uz implementirane sustave obrade i posluživanja paketa te dodjele kapaciteta. Koristeći alate za modeliranje procesa, opisati i analizirati značajke Weighted Fair Queuing metode dodjele kapaciteta u višeslužnim mrežama te ocijeniti njene performanse u različitim prometnim scenarijima.

Mentor:



doc. dr. sc. Marko Matulin

Predsjednik povjerenstva za
diplomski ispit:



prof. dr. sc. Štefica Mrvelj

Sveučilište u Zagrebu
Fakultet prometnih znanosti

DIPLOMSKI RAD

**ANALIZA I MODELIRANJE METODE DODJELE
KAPACITETA U VIŠEUSLUŽNIM MREŽAMA WFQ**

**ANALYSIS AND MODELING OF THE WFQ CAPACITY
ALLOCATION METHOD IN MULTISERVICE NETWORKS**

Mentor:
doc. dr. sc. Marko Matulin

Student: Dora Radičević
JMBAG: 0135247529

Zagreb, rujan 2022.

SAŽETAK

Važan aspekt u višeuslužnim mrežama je osiguranje kvalitete usluge kako bi se osigurale različite razine mrežnih performansi pojedinim tokovima podataka. U području paketno komutiranih telekomunikacijskih mreža, kvaliteta usluge uvelike ovisi o načinu klasifikacije različitih prometnih tokova i metodi dodjele kapaciteta odlaznog linka pojedinim tokovima. Sukladno tome, metode za dodjelu kapaciteta implementirane su u mrežnim čvorovima u cilju minimiziranja nedostatka resursa i osiguravanja pravednosti između tokova podataka koji koriste mrežne resurse. U radu su opisane značajke i funkcionalnosti *Weighted Fair Queueing* (WFQ) metode dodjele kapaciteta koja se temelji na konceptu pravednosti i osigurava da svaki tok podataka dobije određeni udio kapaciteta veze. Pomoću OPNET simulacijskog alata, provedena je analiza performansi WFQ metode u odnosu na druge poznate metode dodjele kapaciteta. Analiza je izvršena na temelju nekoliko parametara kvalitete usluge u cilju ocjenjivanja izvedbe WFQ metode prilikom prijenosa prometa različitih vrsta aplikacija.

KLJUČNE RIJEČI: kvaliteta usluge; metode dodjele kapaciteta; *Weighted Fair Queueing*

SUMMARY

An important aspect in multiservice networks is quality of service assurance so that different levels of network performance to individual data flows can be provided. In the field of packet-switched telecommunications networks, the quality of service largely depends on the method of classification of different traffic flows and the method of allocating outgoing link capacity to individual flows. Accordingly, capacity allocation methods are implemented in network nodes in order to minimize resource shortages and ensure fairness between data flows that use network resources. This study describes the features and functionality of the *Weighted Fair Queueing* (WFQ) capacity allocation method, which is based on the concept of fairness and ensures that each data flow receives a certain share of the connection capacity. Using the OPNET simulation tool, an analysis of the performance of the WFQ method in relation to other known capacity allocation methods was performed. The analysis was performed on the basis of several quality of service parameters in order to evaluate the performance of the WFQ method when transferring traffic of different types of applications.

KEYWORDS: quality of service; capacity allocation methods; *Weighted Fair Queueing*

SADRŽAJ

| | |
|---|----|
| 1. Uvod | 1 |
| 2. Značajke višeuslužnih mreža..... | 3 |
| 2.1. Arhitektura višeuslužnih mreža..... | 4 |
| 2.1.1. OSI referentni model | 5 |
| 2.1.2. TCP/IP protokolarni složaj..... | 7 |
| 2.2. Protokoli u višeuslužnim mrežama | 9 |
| 3. Uloga mrežnog čvora u višeuslužnim mrežama | 12 |
| 3.1. Vrste mrežnih čvorova..... | 12 |
| 3.2. Mrežna topologija | 13 |
| 3.3. Komutacija paketa..... | 15 |
| 4. Kvaliteta usluge u višeuslužnim mrežama | 18 |
| 4.1. Parametri kvalitete usluge | 18 |
| 4.1.1. Širina prijenosnog pojasa..... | 18 |
| 4.1.2. Kašnjenje | 20 |
| 4.1.3. Kolebanje kašnjenja | 21 |
| 4.1.4. Gubitak paketa | 22 |
| 4.2. Zahtjevi aplikacija za kvalitetom usluge..... | 22 |
| 4.2.1. Podatkovne aplikacije | 23 |
| 4.2.2. Glasovne aplikacije | 23 |
| 4.2.3. Video aplikacije | 25 |
| 4.3. Mehanizmi kvalitete usluge | 26 |
| 4.3.1. <i>IntServ</i> mehanizmi | 26 |
| 4.3.2. <i>DiffServ</i> mehanizmi | 28 |
| 5. Opis značajki i modeliranje WFQ metode dodjele kapaciteta primjenom UML dijagrama..... | 32 |
| 5.1. Značajke metoda dodjele kapaciteta | 32 |
| 5.1.1. Disciplina posluživanja FIFO | 33 |
| 5.1.2. Disciplina posluživanja s prioritetom | 34 |
| 5.1.3. Disciplina posluživanja <i>Round Robin</i> | 35 |
| 5.2. Značajke WFQ metode dodjele kapaciteta..... | 36 |
| 5.2.1. Generalizirano dijeljenje procesora | 38 |
| 5.2.2. Vrste WFQ metode dodjele kapaciteta..... | 40 |

| | |
|--|----|
| 5.2.2.1. <i>Class-Based Weighted Fair Queuing</i> | 40 |
| 5.2.2.2. <i>Worst-Case Fair Weighted Fair Queueing</i> | 41 |
| 5.3. Modeliranje WFQ metode dodjele kapaciteta primjenom UML dijagrama | 42 |
| 5.3.1. Dijagram aktivnosti WFQ metode dodjele kapaciteta | 43 |
| 5.3.2. Dijagram međudjelovanja WFQ metode dodjele kapaciteta..... | 45 |
| 6. Analiza performansi WFQ metode dodjele kapaciteta | 47 |
| 6.1. Analiza prijenosa podatkovnog prometa | 50 |
| 6.2. Analiza prijenosa video prometa | 52 |
| 6.3. Analiza prijenosa glasovnog prometa..... | 53 |
| 7. Zaključak | 56 |
| Literatura | 57 |
| Popis kratica | 60 |
| Popis slika | 62 |
| Popis tablica | 63 |
| Popis grafikona..... | 63 |

1. Uvod

Brzi razvoj širokopojasnih mreža rezultirao je stvaranjem niza vremenski osjetljivih multimedijских aplikacija, kao što su VoIP (engl. *Voice over IP*) i video na zahtjev, koje obično imaju različite zahtjeve za kvalitetom usluge (engl. *Quality of Service - QoS*). Sposobnost pružanja QoS podrške za različite vrste aplikacija postala je vrlo važna za dizajn mrežnih čvorova poput preklopnika i usmjerivača. Kako bi pružili različite razine kvalitete usluge pojedinim vrstama aplikacija, mrežni čvorovi koriste discipline posluživanja odnosno metode dodjele kapaciteta odlaznog linka. Metode dodjele kapaciteta predstavljaju važan element u upravljanju zagušenjem i osiguravanju QoS-a definirajući način na koji će pojedini tok prometa biti poslužen. Uslijed mnogih nedostataka najpoznatije metode dodjele kapaciteta FIFO (engl. *First In First Out*), razvijene su druge metode koje bolje mogu odgovoriti na zahtjeve različitih aplikacija.

WFQ (engl. *Weighted Fair Queueing*) je dinamička metoda dodjeljivanja kapaciteta odlaznog linka, koja omogućava pravednu raspodjelu kapaciteta svim prometnim tokovima na temelju težina. Svaki tok povezan je s neovisnim redom čekanja kojem je dodijeljena težina kako bi se osiguralo da promet sa strožim QoS zahtjevima dobije veći udio širine pojasa. Svrha diplomskog rada je opisati način rada i funkcionalnosti WFQ metode za dodjelu kapaciteta primjenom UML (engl. *Unified Modelling Language*) jezika te definirati njezinu ulogu u višeuslužnim mrežama. Cilj diplomskog rada je provesti usporednu analizu performansi WFQ metode s drugim poznatim metodama za dodjelu kapaciteta te na osnovi različitih parametara utvrditi prednosti i nedostatke WFQ metode u danoj situaciji, tj. scenariju primjene. Rad je podijeljen u sedam cjelina:

- 1) Uvod
- 2) Značajke višeuslužnih mreža
- 3) Uloga mrežnog čvora u višeuslužnim mrežama
- 4) Kvaliteta usluge u višeuslužnim mrežama
- 5) Opis značajki i modeliranje WFQ metode dodjele kapaciteta primjenom UML dijagrama
- 6) Analiza performansi WFQ metode dodjele kapaciteta
- 7) Zaključak.

U uvodu je opisana svrha i cilj te koncept diplomskog rada. U drugom poglavlju opisane su najvažnije značajke višeslužnih mreža. Prikazan je primjer arhitekture višeslužne mreže te je opisan koncept OSI (engl. *Open Systems Interconnection*) referentnog modela i TCP/IP (engl. *Transmission Control Protocol/Internet Protocol*) protokolarnog složaja koji služe za razvoj računalnih mreža. Također, definirani su najčešće korišteni protokoli u višeslužnim mrežama.

Trećim poglavljem definirana je uloga mrežnog čvora te su opisane osnovne vrste mrežnih čvorova u višeslužnim mrežama. Prikazane su najčešće korištene mrežne topologije te je definiran i objašnjen koncept komutacije paketa.

Važnost kvalitete usluge u višeslužnim mrežama opisana je četvrtim poglavljem. Definirani su parametri kvalitete usluge te su opisani QoS zahtjevi pojedinih podatkovnih, glasovnih i video aplikacija. Također, mehanizmi kvalitete usluge opisani su na temelju primjene u dvije najpoznatije QoS arhitekture.

U petom poglavlju opisane su značajke metoda dodjele kapaciteta te su na temelju primjera objašnjene funkcionalnosti najpoznatijih metoda poput FIFO, PQ (engl. *Priority Queuing*) i RR (engl. *Round Robin*). Detaljno su opisane značajke i funkcionalnosti WFQ metode dodjele kapaciteta te je pružen uvid u način rada generalizirane politike dijeljenja procesora (engl. *Generalized Processor Sharing - GPS*) na kojoj se ona temelji. Način rada WFQ metode prikazan je primjenom UML jezika, a definirane su i opisane najpoznatije vrste WFQ metode.

U šestom poglavlju provedena je analiza performansi WFQ metode u odnosu na FIFO i PQ metode dodjele kapaciteta. Analiza je izvedena pomoću OPNET simulacijskog alata na način da su provedene mrežne simulacije u tri scenarija. U svakom od scenarija primijenjena je drugačija metoda dodjele kapaciteta, a rezultati simulacija prikazuju kako pojedina metoda utječe na parametre, kao što su poslana i primljena veličina prometnog toka, gubitak paketa, kašnjenje i slično.

Sedmo poglavlje donosi zaključak o performansama WFQ metode na osnovu informacija prikupljenih i obrađenih tijekom istraživanja teme i izrade rada te ističe važnost odabira odgovarajuće metode dodjele kapaciteta kako bi se postigla zadovoljavajuća razina QoS -a.

2. Značajke višeuslužnih mreža

Razvoj nove generacije širokopojasnih telekomunikacijskih mreža odvija se u pravcu višeuslužnih mreža u kojima su integrirane raznovrsne usluge, kao što su:

- prijenos govora
- prijenos podatkovnih aplikacija
- prijenos pokretne slike
- prijenos nepokretne slike
- multimedijske usluge.

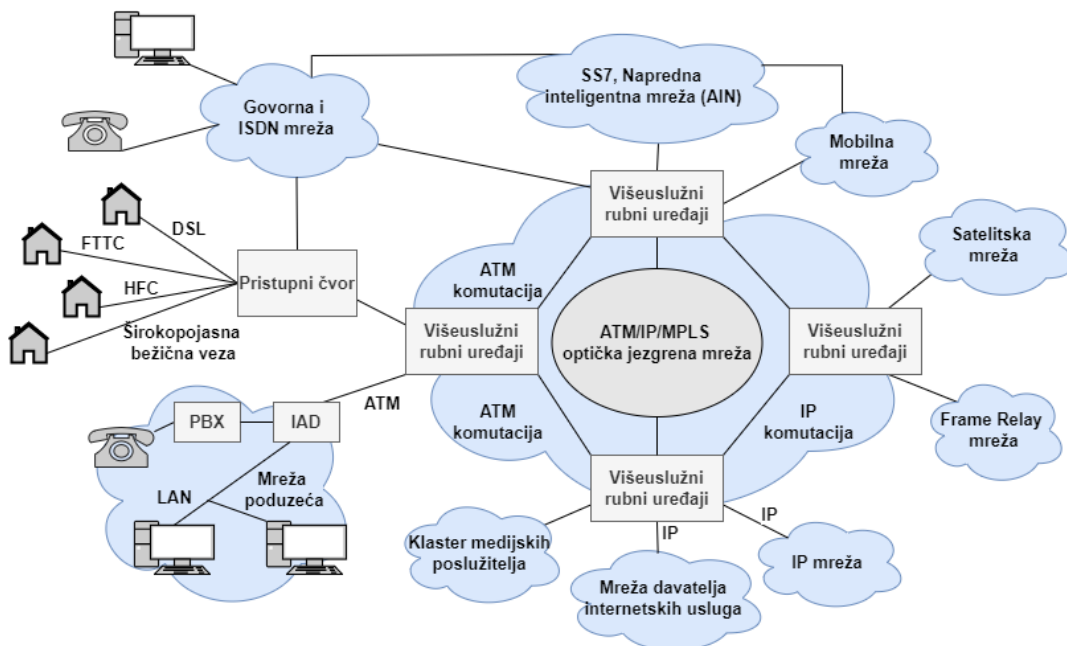
Višeuslužna mreža (engl. *Multi Service Network*) može se definirati kao mreža dizajnirana tako da prenosi promet više od jedne vrste aplikacije preko iste fizičke infrastrukture. Ključni preduvjet za razvoj višeuslužnih mreža je osiguranje različitih razina kvalitete usluge, uz optimalno iskorištenje resursa mreže i stvaranje uvjeta za siguran rad mreže kroz implementaciju odgovarajućih mehanizama upravljanja, kontrole i zaštite. To je u suprotnosti s klasičnim mrežama ili mrežama predviđenim samo za jednu vrstu usluge, kao što je telefonska mreža koja može, općenito govoreći, prenositi samo jednu aplikaciju. Premda internetski promet može biti prenošen telefonskim sustavima, ti sustavi se ne smatraju višeuslužnom mrežom jer nisu dizajnirani s tim ciljem, [1].

Rani primjer višeuslužne mreže je širokopojasna digitalna mreža integriranih usluga (engl. *Broadband Integrated Services Digital Network - B-ISDN*) unutar koje različite usluge imaju različit tretman kako bi se osiguralo da su određena jamstva o kvaliteti ispunjena. Najvažnija karakteristika B-ISDN mreža je koncept podijele prometa u posebne kategorije, koje se nazivaju usluge, prema različitim zahtjevima komunikacije, kao što su vrsta podataka, način prijenosa i brzina prijenosa. Internet se može smatrati još jednim primjerom višeuslužne mreže, iako kvaliteta usluge možda neće zadovoljiti zahtjeve korisnika za neke aplikacije. Privatne mreže koje rade prema istim specifikacijama kao Internet nazivaju se intranet i mogu korisnicima ponuditi bolju kvalitetu usluge, a mrežni operateri mogu ostvariti veću kontrolu nad prometom.

Višeuslužne mreže sljedeće generacije uključuju Ethernet, IP (engl. *Internet Protocol*), VPN¹ (engl. *Virtual Private Network*) i MPLS² (engl. *Multiprotocol Label Switching*) usluge kako bi se prilagodile svim ključnim karakteristikama razvoja i zahtjevima usluga, [2].

2.1. Arhitektura višeuslužnih mreža

Arhitektura višeuslužnih mreža često je kompleksna s obzirom na velik broj usluga koje mreža mora ostvariti, stoga sve komponente moraju biti pažljivo planirane i implementirane. Na slici 1 prikazan je primjer arhitekture širokopojasne višeuslužne mreže koja se sastoji od tri osnovne razine. Prva razina uključuje pristupne uređaje i predstavlja vanjski sloj za pružanje širokopojasne usluge korisniku. Druga razina uključuje rubne uređaje te je povezana s integracijom protokola i podatkovnih usluga. Treća razina, uključuje uređaje u jezgrenom dijelu mreže koji upravljaju prijenosom podataka velikom brzinom kroz okosnicu mreže.



Slika 1. Arhitektura širokopojasne višeuslužne mreže

Izvor: [3]

¹ VPN usluga uspostavlja sigurnu vezu između korisničkog računala i javne mreže.

² MPLS je mrežna tehnologija koja usmjerava promet koristeći najkraći put temeljen na oznakama, a ne na mrežnim adresama.

Pristupni i rubni uređaji dizajnirani su da budu skalabilni, kako u broju portova, tako i u njihovoj sposobnosti pružanja podrške za više usluga, a razvijaju se kako bi uključili inteligentne značajke za upravljanje uslugama. Suprotno tome, jezgri uređaji dizajnirani su da prenose pakete velikim brzinama te im je glavni cilj transportirati promet na pouzdan način, [3].

2.1.1. OSI referentni model

OSI referentni model je apstraktni, slojeviti model koji služi kao preporuka stručnjacima za razvoj računalnih mreža i protokola. Podijeljen je u sedam slojeva, gdje svaki sloj opisuje skup povezanih funkcija koje omogućuju jedan dio računalne komunikacije. OSI referentni model pruža važne smjernice u razvoju mrežnih protokola. Mrežni komunikacijski protokol predstavlja skup određenih pravila (za prikaz podataka, signalizaciju, autorizaciju i otkrivanje pogrešaka) koja su potrebna da bi se podaci mogli prenijeti preko komunikacijskog kanala, [4].

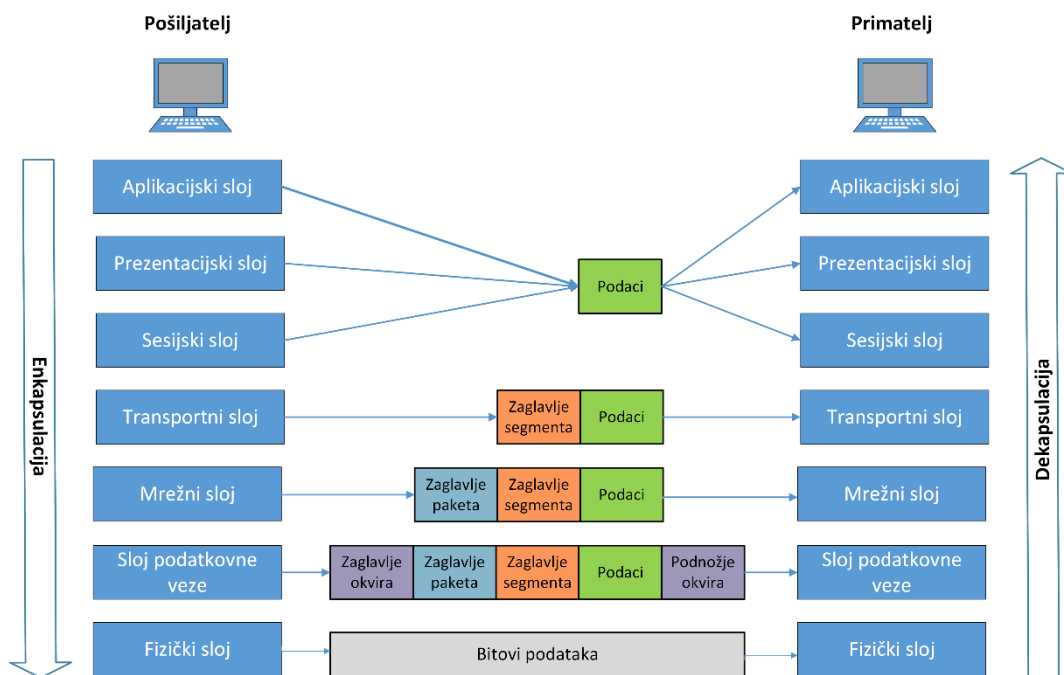
Prema [5], sedam slojeva OSI referentnog modela su:

- 1) Fizički sloj - sastoji se od fizičke opreme uključene u prijenos podataka, kao što su uređaji za multipleksiranje³, mrežna čvorišta, kablovi, repetitori, mrežni adapteri ili modemi. Fizički sloj odgovoran je za stvarnu fizičku vezu između uređaja.
- 2) Sloj podatkovne veze - pruža pouzdanu i učinkovitu komunikaciju između dva ili više uređaja. Glavna funkcija ovog sloja je odgovorna za jedinstvenu identifikaciju svakog uređaja koji se nalazi na lokalnoj mreži.
- 3) Mrežni sloj - omogućava prijenos podataka između računala koja se nalaze u različitim mrežama. Osnovna funkcija mrežnog sloja je usmjeravanje paketa kroz mrežu te odabir najboljeg puta za prijenos paketa. Kako bi se svaki uređaj na mreži jedinstveno identificirao, mrežni sloj definira shemu adresiranja, odnosno postavlja IP adrese pošiljatelja i primatelja u zaglavlje paketa.
- 4) Transportni sloj - odgovoran je za prijenos podataka preko mreže i pruža mehanizme za provjeru grešaka i kontrolu toka podataka.

³ Uređaj za multipleksiranje kombinira više analognih ili digitalnih signala u jedan signal preko zajedničkog medija.

- 5) Sesijski sloj - koristi se za uspostavljanje, održavanje i sinkronizaciju interakcije između komunikacijskih uređaja. Ovaj sloj osigurava da sesija ostaje otvorena dovoljno dugo za prijenos svih podataka koji se razmjenjuju, a zatim odmah zatvara sesiju kako bi se izbjeglo trošenje resursa.
- 6) Prezentacijski sloj - formatira ili prevodi podatke za aplikacijski sloj na temelju sintakse ili semantike koju aplikacija prihvaća. Ovaj sloj je zadužen za prijevod, enkripciju i kompresiju podataka.
- 7) Aplikacijski sloj - omogućuje korisniku interakciju s aplikacijom ili mrežom kad god korisnik odabere čitanje poruka, prijenos datoteka ili obavljanje drugih aktivnosti povezanih s mrežom.

OSI model opisuje način na koji informacija putuje od aplikacije na jednom računalu do aplikacije koja se izvršava na drugom umreženom računalu. Podaci višeg sloja se enkapsuliraju, odnosno podacima višeg sloja se dodaje zaglavlje sloja kroz koji podaci prolaze što znači da svakim prijelazom u niži sloj, informacija mijenja format. Podaci se prvo enkapsuliraju u transportnom sloju i kao takvi nose naziv segment. Segment se tada prebacuje do mrežnog sloja gdje također zaprima zaglavlje mrežnog sloja. Segment i zaglavlje mrežnog sloja na kraju čine paket. Nakon toga paket putuje do sloja veze podataka gdje dobiva novo zaglavlje što zajedno čini tzv. okvir. Ovo zaglavlje definira način transporta informacije kroz mrežno sučelje do fizičkog sloja mreže. U fizičkom sloju mreže podaci se konvertiraju u oblik podoban za prijenos podataka ovisno o tipu prijenosnog medija. Kada podaci stignu na odredište, postupak je obrnut što znači da se svakim prijelazom u novi sloj, podatku uklanja zaglavlje tog sloja kako bi se vratio u oblik u kojem je podatak bio poslan. Takav postupak naziva se dekapulacija, [6]. Postupak enkapsulacije i dekapulacije podataka prikazan je na slici 2.



Slika 2. Postupak enkapsulacije i dekapulacije podataka

Izvor: [6]

Suvremeni Internet ne temelji se na OSI referentnom modelu, već na jednostavnijem TCP/IP protokolarnom složaju. Međutim, OSI model još uvijek se široko koristi jer služi kao pomoć u vizualizaciji i definiranju rada računalnih mreža te u rješavanju problema s umrežavanjem, [7].

2.1.2. TCP/IP protokolarni složaj

OSI referentni model opisuje funkcije telekomunikacijskog ili mrežnog sustava, dok TCP/IP protokolarni složaj predstavlja skup komunikacijskih protokola koji se koriste za međusobno povezivanje mrežnih uređaja na Internetu. Za razliku od sedam slojeva u OSI modelu, TCP/IP protokolarni složaj sastoji se od četiri funkcionalna sloja koji uključuju specifične protokole. Svrha TCP/IP složaja je omogućiti komunikaciju na velikim udaljenostima, [7]. Prema [8], četiri sloja TCP/IP protokolarnog složaja su:

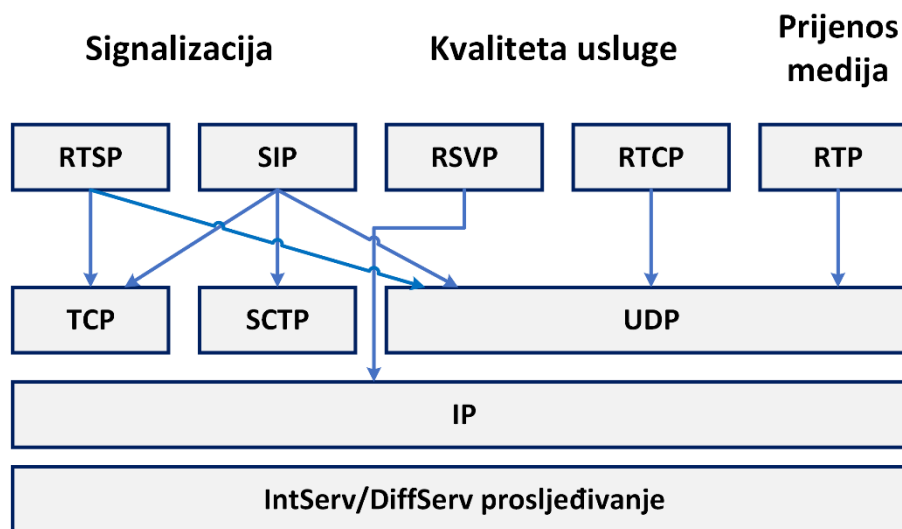
- 1) Sloj pristupa mreži - odgovara kombinaciji podatkovnog i fizičkog sloja OSI modela. Zadužen je za hardversko adresiranje i protokoli prisutni u ovom sloju omogućuju fizički prijenos podataka.

- 2) Internet sloj - odgovara funkcijama mrežnog sloja OSI modela. Definira protokole koji su odgovorni za logički prijenos podataka preko cijele mreže. Najznačajniji protokoli koji se nalaze na ovom sloju su:
- IP - odgovoran je za isporuku paketa od izvornog do odredišnog mrežnog čvora prateći IP adrese u zaglavlju paketa.
 - ICMP (engl. *Internet Control Message Protocol*) - inkapsuliran je unutar IP datagrama i odgovoran je za pružanje informacija o mrežnim problemima.
 - ARP (engl. *Address Resolution Protocol*) - povezuje stalno promjenjivu IP adresu s fiksnom fizičkom adresom računala, također poznatom kao MAC (engl. *Media Access Control*) adresa.
- 3) Prijenosni sloj - odgovoran je za komunikaciju od kraja do kraja i isporuku podataka bez pogrešaka te štiti aplikacije višeg sloja od složenosti podataka. Dva glavna protokola prisutna na ovom sloju su:
- TCP (engl. *Transmission Control Protocol*) - osigurava pouzdanu komunikaciju između krajnjih sustava te obavlja segmentaciju podataka. Također ima značajku potvrde i kontrolira tijek podataka kroz mehanizam kontrole toka.
 - UDP (engl. *User Datagram Protocol*) - smatra se bržim, jednostavnijim i učinkovitijim protokolom u odnosu na TCP, ali ne osigurava pouzdanu komunikaciju i ne garantira kontrolu toka, što rezultira učestalom pojavom mrežnog zagušenja i posljedičnom QoS degradacijom.
- 4) Sloj aplikacije - obavlja funkcije gornja tri sloja OSI modela: aplikacijskog, prezentacijskog i sesijskog. Odgovoran je za komunikaciju između čvorova i kontrolira specifikacije korisničkog sučelja. Najpoznatiji protokoli ovog sloja su:
- HTTP (engl. *Hypertext Transfer Protocol*) - koristi ga *World Wide Web* za upravljanje komunikacijama između internetskog preglednika i poslužitelja.
 - FTP (engl. *File Transfer Protocol*) - koristi se za prijenos datoteka između klijenta i poslužitelja.

Iako je TCP/IP pouzdaniji i jednostavniji pristup u razvoju računalnih mreža, on ovisi o protokolima na temelju kojih je izrađen i stog protokola nije jednostavno zamijeniti. U suprotnosti tome, OSI model je generički standard neovisan o protokolu. Na temelju OSI modela kreirani su protokoli koji odgovaraju potrebama mrežne arhitekture i moguće ih je lako zamijeniti promjenom tehnologije, [8].

2.2. Protokoli u višeuslužnim mrežama

U svrhu ostvarivanja usluga i prijenosa podataka putem višeuslužnih mreža nove generacije, kao što je Internet, koristi se niz mrežnih komunikacijskih protokola. *Internet Engineering Task Force* (IETF) je najpoznatija organizacija za otvorene standarde koja definira operativne Internet protokole, [9]. Na slici 3 prikazan je IETF-ov protokolarni složaj za prijenos multimedijiskog sadržaja.



Slika 3. Protokolarni složaj za prijenos multimedijiskog sadržaja

Izvor: [10]

S obzirom da broj aplikacija koje zahtijevaju uspostavu sesije kontinuirano raste, razvijen je SIP (engl. *Session Initiation Protocol*) protokol s glavnom svrhom uspostavljanja i održavanja sesije između dva ili više sudionika. SIP je tekstualni klijent-poslužitelj protokol, u kojem klijent izdaje zahtjeve, a poslužitelj vraća odgovore. SIP koristi veći dio sintakse i semantike HTTP protokola i svaki SIP zahtjev je pokušaj pozivanja neke metode na poslužitelju, [11].

SIP protokol se može izvoditi na vrhu TCP, UDP ili SCTP (engl. *Stream Control Transmission Protocol*) protokola. SCTP protokol pruža značajku orijentiranu na poruke UDP protokola istovremeno osiguravajući pouzdan prijenos poruka s kontrolom zagušenja kao i TCP protokol. Za razliku od UDP i TCP protokola, SCTP podržava redundantne putove radi povećanja otpornosti i pouzdanosti, [12].

UDP protokol karakteriziraju dva nedostatka. Prvi problem predstavlja nedostatak mehanizama za kontrolu mrežnog zagušenja što je neophodno za prevenciju mrežnog

kolapsa. Drugi problem predstavlja fragmentacija SIP poruka, s obzirom da je UDP nepouzdan protokol i ne osigurava sredstva za oporavak od izgubljenih fragmenata i promjene njihovog redoslijeda. TCP je sigurniji način prijenosa SIP poruka jer postoji kontrola postojeće TCP veze otvorene za uspostavu sesije, ali uvijek je moguće kašnjenje prilikom uspostave veze. Kako bi se izbjeglo vrijeme potrebno za uspostavu i raskidanje TCP veze, UDP protokol je preferirani transportni protokol za prijenos SIP poruka, [10].

SIP ne pruža funkcionalnost potrebnu za slanje multimedijskog sadržaja preko mreže, ili mnoge druge usluge koje se nalaze u komunikacijskim programima. Umjesto toga, to je komponenta koja radi s drugim protokolima, kao što su SDP⁴ (engl. *Session Description Protocol*), RTP (engl. *Real-time Transport Protocol*), MGCP⁵ (engl. *Media Gateway Control Protocol*) i RTSP (engl. *Real-time Streaming Protocol*). Iako se SIP, zajedno s navedenim protokolima, koristi u svrhu omogućavanja potpune usluge korisnicima, njegova osnovna funkcionalnost ne ovisi niti o jednom od navedenih protokola, [13].

Za prijenos vremenski osjetljivih tokova definiran je RTP protokol koji osigurava prijem paketa u ispravnom redoslijedu, nadzor QoS parametara te identifikaciju i specifikaciju različitih tipova korisnih podataka. Iako je proširiv u smislu definiranja profila za formate i tipove korisnih podataka, RTP posjeduje brojna ograničenja. Bez polja za identifikaciju porta, RTP se mora oslanjati na UDP protokol. Za razliku od TCP protokola, RTP ne nastoji osigurati pouzdan prijenos paketa osjetljivih na kašnjenje preko nepouzdanane mreže. Također, ne omogućava rezervaciju resursa, niti garantira QoS za vremenski osjetljive usluge, [10].

Iako djeluje povrh nepouzdanog UDP protokola, RTP pruža određenu pouzdanost u prijenosu podataka. RTP protokol koristi RTCP (engl. *Real-time Transport Control Protocol*) za praćenje isporuke podataka koji se šalju između sudionika. To omogućuje korisničkom agentu koji prima podatke da otkrije postoji li gubitak paketa i omogućuje mu kompenzaciju za sva kašnjenja koja bi mogla nastati dok se podaci prenose kroz mrežu. RTCP omogućuje kvalitetnu povratnu informaciju o usluzi od primatelja do pošiljatelja kako bi mogao prilagoditi svoje karakteristike prijenosa i kako bi bolje

⁴ SDP protokol služi za opisivanje multimedijskih sesija u formatu razumljivom sudionicima u komunikaciji.

⁵ MGCP protokol služi za signalizaciju i kontrolu poziva u VoIP telekomunikacijskim sustavima.

odgovarao prevladavajućim mrežnim uvjetima. Često se koristi zajedno s RTSP protokolom za kontrolu *streaming* medijskih poslužitelja te za uspostavljanje i kontrolu medijskih sesija, [13].

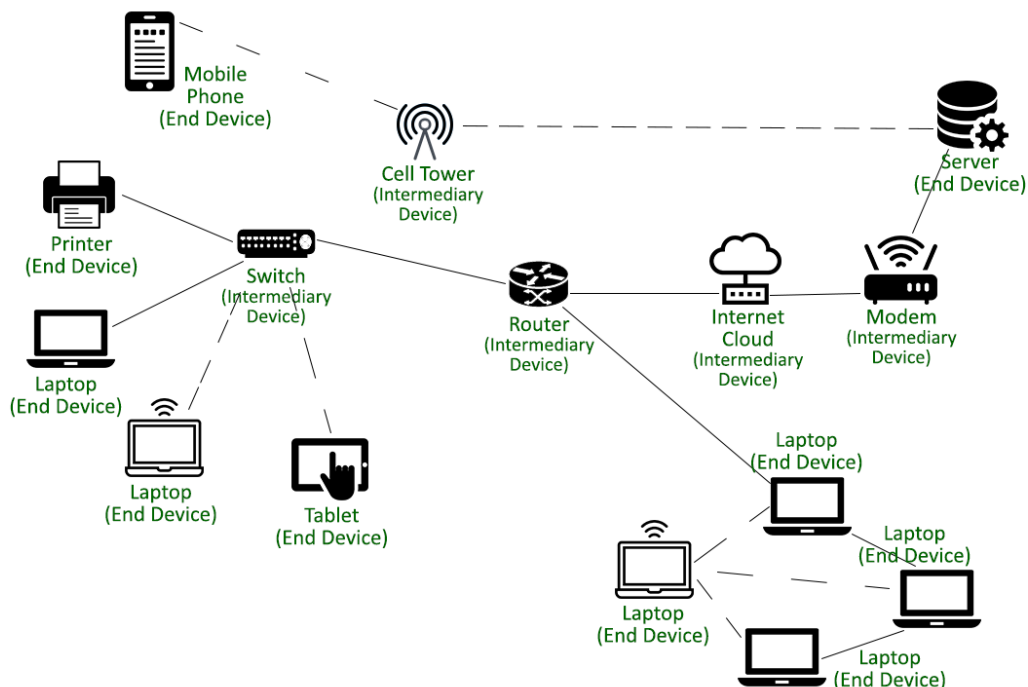
SIP i RTCP ne osiguravaju kvalitetu usluge, stoga se moraju oslanjati na postojeće QoS arhitekture što može prouzročiti problem interoperabilnosti između QoS arhitekture i protokola. Iz tog razloga, posebna pažnja se posvećuje radu na poboljšanju interoperabilnosti, ali i izboru QoS arhitekture. Kako bi se omogućila kvaliteta usluge od kraja do kraja u IP mrežama IETF je definirano dvije osnovne arhitekture: IntServ (engl. *Integrated Services*) i DiffServ (engl. *Differentiated Services*), [10].

3. Uloga mrežnog čvora u višeuslužnim mrežama

Mrežni čvor je fizički uređaj unutar mreže koji može slati, primiti i/ili prosljeđivati informacije. Svaki mrežni čvor djeluje kao spojna točka za prijenos podataka, prepoznavanje procesa i mrežnu distribuciju. Općenito su programirani za identifikaciju, obradu i prijenos podataka s jednog čvora na drugi te mogu obavljati nekoliko funkcija na temelju aplikacije i mreže. Uvelike ovise o navedenoj mreži i sloju protokola za formiranje mrežne veze, a međusobno su povezani poveznicama (engl. *links*). Čvorovi unutar mreže moraju imati neki oblik identifikacije, poput IP adrese ili MAC adrese, kako bi ga drugi mrežni uređaji prepoznali, [14].

3.1. Vrste mrežnih čvorova

Na temelju funkcionalnosti i upotrebe, mrežni čvorovi mogu se klasificirati na krajnje uređaje i posredničke uređaje. Na slici 4 su prikazane različite vrste mrežnih čvorova u komunikacijskoj mreži.



Slika 4. Mrežni čvorovi u komunikacijskoj mreži, [15]

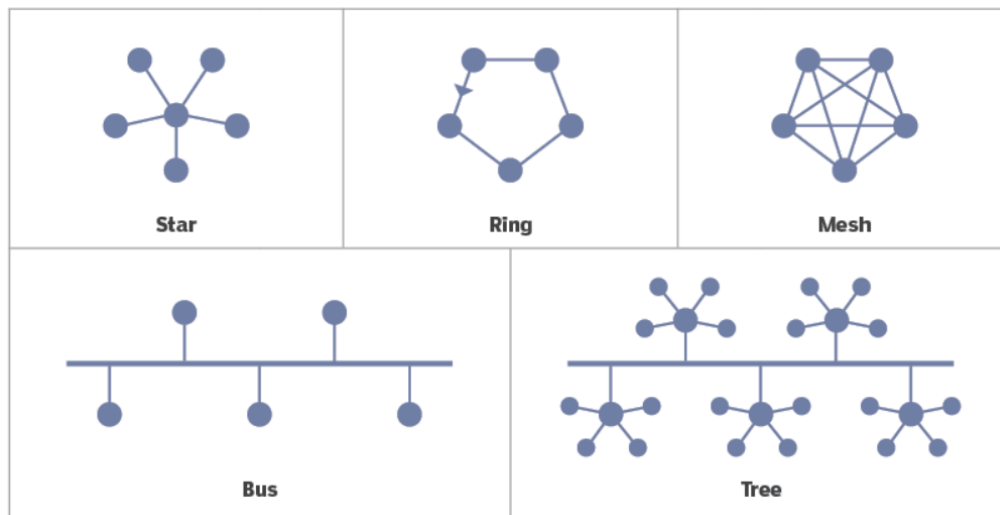
Krajnji uređaji su mrežni čvorovi koji služe kao izvorna ili odredišna točka u komunikaciji te djeluju kao sučelje između krajnjih korisnika i komunikacijske mreže. Mogu biti u funkciji klijenta i poslužitelja, a softver instaliran na uređajima čvora određuje njihovu ulogu u računalnoj mreži. Krajnji uređaj kao klijent obavlja zadatke poput traženja i prikazivanja podataka koji su obično namijenjeni za korištenje krajnjim korisnicima, dok krajnji uređaj kao poslužitelj pruža informacije i usluge poput internetskih stranica ili e-pošte drugim čvorovima na mreži. Primjeri krajnjih uređaja su radne stanice, prijenosna i stolna računala, pisači i skeneri, datotečni i internetski poslužitelji, mobilni telefoni, tableti itd.

Posrednički uređaji su mrežni čvorovi dizajnirani za prosljeđivanje podataka od izvorišta do odredišta u mreži. Djeluju kao povezujući medij za druge čvorove i upravljaju zadacima u pozadini osiguravajući učinkovit protok podataka željenim brzinama putem cijele mreže. Posrednički uređaji za upravljanje podacima koriste različite sustave adresiranja poput IP adresa, MAC adresa i brojevi porta zajedno s informacijama o mrežnim vezama. Kako bi osigurali uspješne prijenose, čvorovi čuvaju podatke o izvornoj adresi, odredišnoj adresi ili različitim putovima koji postoje kroz mrežu. Također, učinkovito otkrivaju kvarove i pogreške te obavještavaju ostale uređaje u mreži kako bi dodatno osigurali toleranciju na greške izvođenjem ispravaka tijekom prijenosa podataka. Primjeri posredničkih uređaja su preklopnici, bežične pristupne točke i drugi uređaji koji se koriste za pristup mreži, internetski poslužitelji, modemi te uređaji koji se koriste za umrežavanje, kao što su usmjerivači, mostovi, repetitori, sigurnosni vatrozidi itd., [15].

3.2. Mrežna topologija

Mrežne topologije opisuju fizički i logički raspored čvorova i veza u mreži te relativnu lokaciju tokova prometa. Administratori mogu koristiti dijagrame topologije mreže kako bi odredili najbolje položaje za svaki čvor i optimalni put za protok prometa. Mrežna topologija igra glavnu ulogu u funkcioniranju mreže. Odabir prave topologije može pomoći u povećanju performansi, budući da pravilno odabrana i održavana topologija mreže povećava energetska učinkovitost i brzine prijenosa podataka. Dobro definirana topologija mreže olakšava mrežnim administratorima lociranje grešaka, rješavanje problema i dodjelu mrežnih resursa.

Fizička topologija mreže predstavlja fizički raspored čvorova i veza. Veze uključuju linije u dijagramima koje povezuju čvorove, kao što su spojni vodovi za Ethernet ili digitalne pretplatničke linije, optička vlakna i slično. Logičke mrežne topologije definiraju kako je mreža postavljena, uključujući koji se čvorovi povezuju i kako, kao i obrazac prijenosa podataka, [16]. Na slici 5 prikazano je nekoliko osnovnih vrsta fizičke mrežne topologije.



Slika 5. Vrste fizičke mrežne topologije, [16]

U topologiji sabirničke mreže (engl. *Bus*) svaki je čvor povezan u seriju duž jednog kabela. Svi podaci u razmjeni se šalju preko centralnog vodiča, a prekid rada jednog čvora dovodi do prestanka u komunikaciji između svih čvorova. Ova topologija se danas uglavnom koristi u kabelskim širokopojasnim distribucijskim mrežama.

U topologiji zvjezdane mreže (engl. *Star*), središnji uređaj povezuje se sa svim drugim čvorovima preko središnjeg čvorišta. Prestanak rada središnjeg čvora rezultira prekidom rada cijele mreže. Prekid rada bilo kojeg drugog čvora na mreži, osim središnjeg, ne utječe na komunikaciju ostalih čvorova u tom mrežnom segmentu. Komutirane lokalne mreže temeljene na Ethernet komutatorima i većina ožičenih kućnih i uredskih mreža imaju topologiju zvijezde.

U topologiji prstenaste mreže (engl. *Ring*), čvorovi su povezani u konfiguraciji zatvorene petlje. Neke prstenaste topologije prenose podatke samo u jednom smjeru, dok druge dopuštaju prijenos podataka u oba smjera. Dvosmjerne prstenaste mreže otpornije su od sabirničkih mreža budući da promet može doći do čvora krećući se u bilo kojem smjeru.

Isprepletana topologija (engl. *Mesh*) povezuje čvorove tako da su dostupni višestruki putovi između barem nekih točaka mreže. Mreža se smatra potpuno isprepletenom ako su svi čvorovi izravno povezani sa svim drugim čvorovima i djelomično isprepletenom ako samo neki čvorovi imaju višestruke veze s drugima. Potpuna isprepletana topologija je preskupa i presložena za primjenu tako da se koristi samo na mjestima gdje je to krajnje nužno i gdje nema veliki broj čvorova koje je potrebno povezati.

Mrežna topologija stabla (engl. *Tree*) sastoji se od jednog korijenskog čvora, a svi ostali čvorovi povezani su u hijerarhiju. Sama topologija povezana je u zvjezdastu konfiguraciju. Mnoge veće mreže Ethernet preklopnika, uključujući mreže podatkovnih centara, konfigurirane su kao stabla.

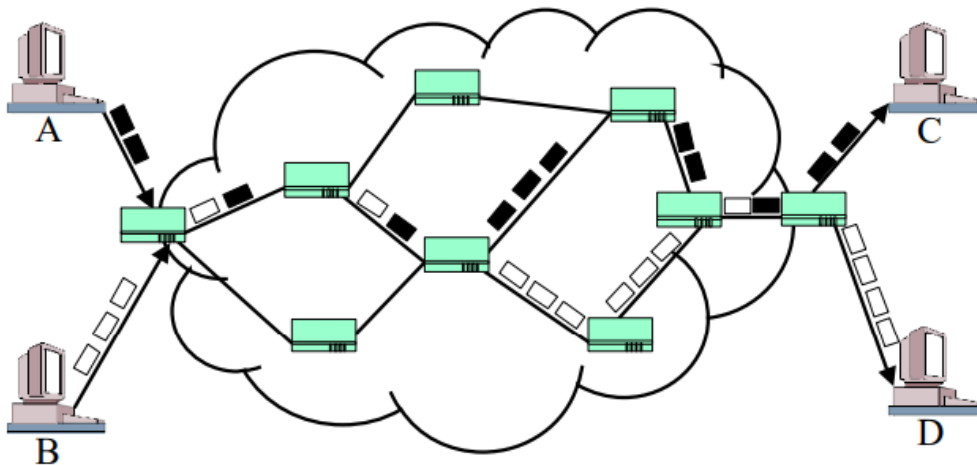
Logička mrežna topologija odnosi se na odnos između čvorova i logičkih veza, odnosno definiranje načina prijenosa podataka. Kao primjer, topologije logičke sabirnice i logičkog prstena mogu se koristiti za definiranje tokova prijenosa podataka. Topologija logičke sabirnice sadrži čvorove koji emitiraju podatke cijeloj mreži. Ostali čvorovi na mreži provjeravaju jesu li podaci namijenjeni njima. Topologija logičkog prstena dopušta samo jednom čvoru prijenos podataka u isto vrijeme, [16].

3.3. Komutacija paketa

U mrežnim su čvorovima implementirane funkcije komutiranja i usmjeravanja informacijskih tokova. Komutiranje je moguće definirati kao operaciju pomoću koje čvor informaciju, koju primi od svojih ulaznih priključaka, šalje na jedan ili više točno određenih izlaznih priključaka. Mehanizam komutiranja u čvoru ovisi o tome primjenjuje li se u mreži načelo komutacije kanala ili komutacije paketa. Mrežni uređaj koji obavlja funkciju komutiranja naziva se komutator ili komutacijski čvor. Čvorovi međusobno razmjenjuju informacije o smjerovima raspoloživima za prijenos informacija kroz mrežu i o raspoloživim prijenosnim kapacitetima. Tako svaki čvor može učinkovito komutirati primljene informacijske tokove, a ta se funkcija čvora naziva određivanje smjerova prijenosa informacije ili, skraćeno, usmjeravanje, [17].

Komutacija paketa prenosi podatke putem mreže razlažući ih na blokove ili pakete. Svaki put kada jedan uređaj pošalje datoteku drugom, on rastavlja datoteku u pakete kako bi mogao odrediti najučinkovitiji put za slanje podataka preko mreže u danom

trenutku. Mrežni uređaji tada mogu usmjeriti pakete do odredišta gdje ih prijemni uređaj ponovno sastavlja za upotrebu. Kod komutacije paketa ne postoji rezervacija resursa, ali niti jamstvo pravovremenosti dostave podataka, [18]. Primjer komutacije paketa prikazan je na slici 6. Računalo A šalje podatke računalu C, a računalo B šalje podatke računalu D. Podaci s računala A prikazani su crnim paketima, a podaci s računala B prikazani su bijelim paketima. U paketno komutiranoj mreži paketi iz različitih veza dijele kapacitet mreže. Međutim, ako stopa dolazaka premašuje kapacitet prosljeđivanja na usmjerivaču, stvorit će se red i može doći do zagušenja i odbacivanja paketa.



Slika 6. Komutacija paketa, [19]

Prema [19], postoje dvije glavne vrste komutacije paketa:

- 1) Komutacija paketa bez povezivanja (engl. *Connectionless Packet Switching*).
- 2) Komutacija paketa orijentirana na vezu (engl. *Connection-Oriented Packet Switching*).

Komutacija paketa bez povezivanja uključuje više paketa, od kojih se svaki pojedinačno usmjerava. To znači da svaki paket sadrži potpune informacije o usmjeravanju, ali i su mogući različiti putovi prijenosa i isporuke paketa različitim redoslijedom, ovisno o trenutnom promjenjivom opterećenju mrežnih čvorova. Ova vrsta komutacije paketa ponekad se naziva i komutacija datagrama. Svaki paket u svom odjeljku zaglavlja sadrži IP adresu izvora, IP adresu odredišta, ukupan broj

paketa sesije i redni broj za ponovno sastavljanje. Nakon što paketi stignu do svog odredišta različitim rutama, prijemni uređaj ih sastavlja u izvornu poruku.

U komutaciji paketa orijentiranoj na vezu prije početka prijenosa, između pošiljatelja i primatelja, pomoću signalnog protokola uspostavlja se logička staza ili virtualna veza. Svi paketi koji pripadaju određenom toku prenose se uzastopno unaprijed definiranom rutom. ID svake virtualne veze osiguravaju mrežni čvorovi za njihovu jedinstvenu identifikaciju.

Komutacija kanala razlikuje se od komutacije paketa jer stvara fizički kanal između izvora i odredišta i zahtijeva njegovu rezervaciju prije početka prijenosa podataka. Nakon uspostavljanja kanala, cijela poruka se prenosi od izvora do odredišta bez dijeljenja u pakete. Budući da u mreži postoji ograničeni kapacitet, mreža s komutacijom kanala može podržati samo fiksni broj veza istovremeno ovisno o kapacitetu mreže i količini kapaciteta rezerviranoj za svaku vezu. Ako je napravljena rezervacija i veza je uspostavljena, ali se podaci ne prenose, nijedna druga veza ne može koristiti neiskorišteni kapacitet koji je rezerviran, [18]. Komutacija kanala koristi se u telefonskim sustavima za prijenos podataka u stvarnom vremenu, dok su mreže s komutacijom paketa učinkovitije za podatke koji mogu tolerirati neka kašnjenja u prijenosu, kao što su podaci o lokaciji i poruke e-pošte, [19].

4. Kvaliteta usluge u višeuslužnim mrežama

Kvaliteta usluge može se definirati kao mjera sposobnosti mreže i računalnih sustava da pruže različite razine usluga odabranim aplikacijama i povezanim mrežnim tokovima. Tradicionalno, višeuslužne mreže temeljene na IP protokolu nudile su samo *Best Effort* (BE) uslugu isporuke prometa gdje se sav mrežni promet tretira jednako. Kao što ime sugerira, prosljeđivanje paketa se izvodi uz maksimalne napore, ali bez jamstva osnovnih parametara QoS-a. Time se isključuje sposobnost IP mreže da prenosi promet koji ima određene minimalne mrežne resurse i zahtjeve za jamstvom kvalitete usluge.

Za podršku glasovnom, video i podatkovnom prometu s različitim zahtjevima usluge, sustavi u jezgri IP mreže trebaju razlikovati i posluživati različite vrste prometa na temelju njihovih potreba. Stoga je podrška različitih razina QoS-a preduvjet za primjenu IP tehnologije u višeuslužnim mrežama. QoS pruža garanciju isporuke usluge od kraja do kraja i kontrolu mjera performansi mreže. Također osigurava učinkovitije korištenje mrežnih resursa, koegzistenciju aplikacija osjetljivih na performanse mreže te pruža podršku postojećim i nadolazećim zahtjevima multimedijских aplikacija, [20], [21].

4.1. Parametri kvalitete usluge

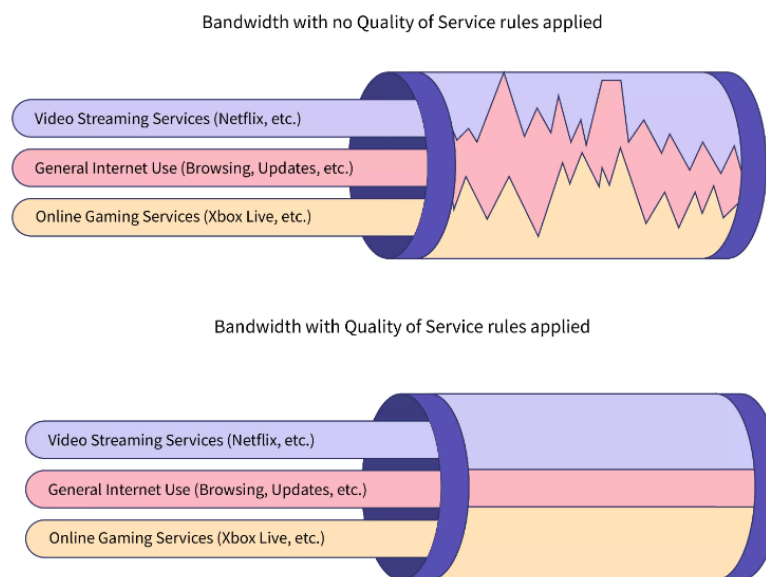
Performanse mreže mogu se detaljnije opisati pomoću nekoliko parametara kvalitete usluge, kao što su kašnjenje, kolebanje kašnjenja, širina pojasa i intenzitet gubitka paketa. Navedeni parametri koriste se za karakterizaciju performansi veze unutar mreže. Mreže koje podržavaju QoS nastoje jamčiti kvalitetu usluge ne prekoračujući ograničenja navedenih QoS parametara, [11].

4.1.1. Širina prijenosnog pojasa

Širina prijenosnog pojasa je kapacitet mrežnog kanala za prijenos maksimalne količine podataka kroz kanal u određenom vremenu i obično se mjeri bitovima po sekundi (b/s). Kao pojam, širina prijenosnog pojasa često se poistovjećuje s propusnošću. Međutim, glavna razlika između propusnosti i širine pojasa je ta što se širina pojasa odnosi na najveću moguću brzinu kojom uređaj može prenijeti podatke,

dok se propusnost odnosi na stvarnu brzinu kojom uređaj prenosi podatke u određenom trenutku. Stoga je širina pojasa uvijek veća ili jednaka propusnosti, [22].

Neodgovarajuća širina pojasa dovodi do povećanja kašnjenja paketa što posljedično rezultira gubitkom paketa jer se međuspremnici (engl. *buffer*) usmjerivača pretrpavaju i dolazi do odbacivanja paketa. QoS optimizira mrežu upravljanjem njezinom širinom pojasa i postavljanjem prioriteta za one aplikacije koje zahtijevaju više resursa u usporedbi s drugim aplikacijama. Ako dođe do zagušenja prometa tijekom razdoblja intenzivne mrežne komunikacije, značajke QoS osiguravaju da određeni tokovi podataka dobiju prednost kako bi se osigurao dosljedan protok podataka. Na primjer, mreže koje prenose audio ili video u stvarnom vremenu zahtijevaju visoku razinu QoS-a kako bi se osigurala isporuka bez pogrešaka i gubitaka.



Slika 7. Usporedba iskoristivosti širine pojasa prije i nakon implementacije QoS-a, [23]

Na slici 7 je prikazana usporedba iskoristivosti dostupne širine pojasa u slučaju kada je implementirana podrška za kvalitetom usluge i kad nije. QoS dodjeljuje određenu količinu kapaciteta prijenosa različitim redovima čekanja za različite vrste prometa ovisno o zahtjevima aplikacija. Na taj način izbjegava se narušavanje kvalitete ostalih aplikacija koje se isporučuju putem istog medija, [23].

4.1.2. Kašnjenje

Mrežno kašnjenje općenito se definira u smislu jednosmjernog kašnjenja za neprilagodljive (neelastične) vremenski kritične aplikacije kao što su VoIP i video, te u smislu povratnog kašnjenja (engl. *Round Trip Time* - RTT) za prilagodljive (elastične) aplikacije, poput onih koje koriste TCP protokol. Jednosmjerno kašnjenje je vrijeme potrebno za prijenos IP paketa s definirane ulazne točke mreže do definirane izlazne točke mreže. RTT je vrijeme potrebno za prijenosa IP paketa s izvorišta do odredišta i naknadnog primitka odgovarajućeg paketa odgovora s tog odredišta, isključujući kašnjenja obrade krajnjeg sustava. Bilo da se radi o jednosmjernom ili povratnom kašnjenju, kašnjenja inducirana u mreži sastoje se od sljedeće četiri komponente [24]:

- Propagacija - propagacijsko kašnjenje je vrijeme potrebno za prijenos jednog bita s izlaznog porta na usmjerivaču do ulaznog porta drugog usmjerivača. Ograničeno je brzinom svjetlosti u prijenosnom mediju, stoga ovisi o udaljenosti veze i o korištenom fizičkom mediju.
- Procesiranje – odnosi se na vrijeme potrebno za kodiranje/kompresiju, dekodiranje/dekompresiju i stvaranje paketa podataka (paketiranje).
- Serijalizacija - označava vrijeme potrebno da se paket usmjeri na odlazni link te ovisi o brzini veze i veličini paketa. Kašnjenje uslijed serijalizacije proporcionalno je veličini paketa i obrnuto proporcionalno brzini odlaznog linka.
- Kašnjenje zbog čekanja na posluživanje - predstavlja vremensku razliku između postavljanja paketa u red čekanja i početka posluživanja u čvoru. To je funkcija korištenog algoritma raspoređivanja i iskorištenosti reda čekanja. Ova vrsta kašnjenja kontrolira se upravljanjem prometnim opterećenjem i primjenom odgovarajućih disciplina posluživanja i raspoređivanja.

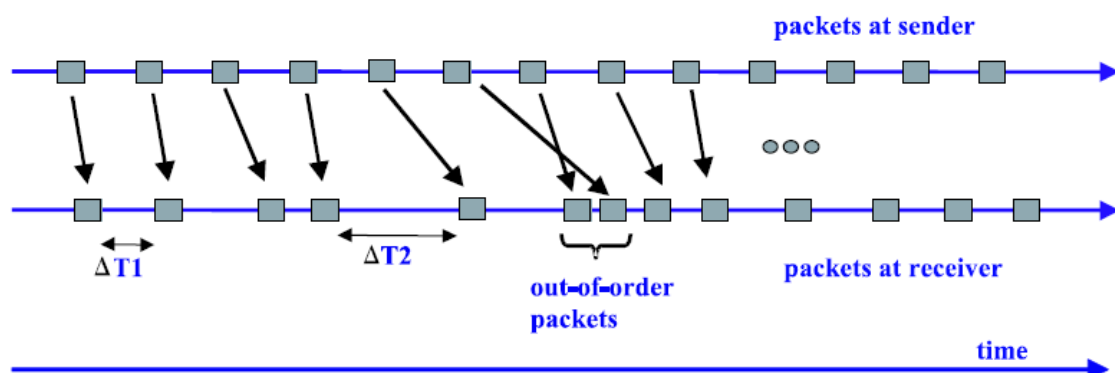
Paketi u mreži nemaju isto kašnjenje, već ono može varirati ovisno o promjenjivim mrežnim uvjetima. Ako mreža nije zagušena, neće se stvarati redovi čekanja na usmjerivačima. Stoga fiksne komponente kašnjenja (propagacija, procesiranje i serijalizacija) predstavljaju ukupno kašnjenje paketa i to je minimalno kašnjenje koje mreža može ponuditi, [24].

4.1.3. Kolebanje kašnjenja

Ako je mreža zagušena, kašnjenja nastala zbog čekanja počet će utjecati na kašnjenja s kraja na kraj i pridonijet će kolebanju kašnjenja (engl. *jitter*) između paketa na istom *linku*. *Jitter* uzrokuju varijacije u međudolaznim vremenima paketa iste sesije. Kontrola kolebanja kašnjenja važna je za aplikacije čija ispravna obrada ovisi o isporuci paketa u zajamčenim vremenskim intervalima. Na primjer, aplikacije VoIP i FoIP (engl. *Fax over IP*) ne ponašaju se adekvatno ako paketi ne stignu zadanom brzinom na odredište. Prema [25], čimbenici koji utječu na varijacije kašnjenja su:

- Promjenjivo vrijeme obrade na posredničkim uređajima
- Promjenjiva kašnjenja javnih komunikacijskih usluga
- Zagušenje
- Ostali čimbenici povezani s radom mreže.

S obzirom na varijabilnost obrade i kašnjenja, evidentno je da je vrijeme obrade različito za svaki dolazni paket zbog nepredvidivog ponašanja raspodjele resursa u čvoru paketne mreže. Također, važno je spomenuti da je zagušenje kritičan problem za kolebanje kašnjenja. Jednom kada mreža ili njezini čvorovi dođu u stanje zagušenja, kašnjenja se povećavaju, pridonoseći povećanju *jitter*-a. Slika 8 prikazuje efekt kolebanja kašnjenja u mreži. Važno je primijetiti da kolebanje kašnjenja ne samo da uzrokuje varijabilnost u isporuci paketa, već također može uzrokovati da paketi budu primljeni nepravilno.



Slika 8. Efekt kolebanja kašnjenja, [25]

Da bi se prikrile varijacije u kašnjenju paketa i spriječile neželjene pojave, dimenzioniraju se tzv. *jitter* međuspremnici koji čuvaju dolazne pakete za obradu na takav način da će primatelj uvijek imati dolazni paket spreman za obradu. Time se

nastoje izgladiti varijacije kašnjenja koje su prisutne u paketnoj mreži i izvršiti ponovni poredak paketa koji su primljeni nepravilnim redoslijedom. *Jitter* međuspremnicima imaju važnu ulogu kod aplikacija za reprodukciju koje šalju kontinuirani tok informacija, uključujući aplikacije kao što su interaktivni glasovni pozivi i videokonferencije, [25].

4.1.4. Gubitak paketa

Intenzitet gubitka paketa označava broj paketa koji ne stignu do odredišta u odnosu na sve poslana paketa. Gubitak paketa uglavnom ima tri uzroka [25]:

- Ispuštanje paketa zbog zagušenja
- Greške u prijenosu npr. zbog loše kvalitete veze
- Odluke o obradi usmjeravanja.

Gubitak paketa općenito se događa na točkama zagušenja kada broj dolaznih paketa daleko premašuje ograničenje veličine reda čekanja u čvoru što rezultira odbacivanjem paketa. Pogreška u prijenosu mnogo je rjeđi uzrok gubitka paketa, budući da su stope pogrešaka znatno smanjene upotrebom optičkih vlakana i drugih komunikacijskih tehnika. Gubitak paketa zbog odluka o obradi usmjeravanja odgovara situaciji u kojoj je paket neispravno preusmjeren i zbog toga je ispušten negdje u mreži.

Iz perspektive krajnjeg korisnika, intenzitet gubitaka paketa određuje kvalitetu prezentacije korisničkih aplikacija, na primjer, kvalitetu zvuka, slike i videa. Intenzitet gubitka paketa razlikuje se ovisno o protokolu. Ako je paket odbačen ili nije potvrđen, TCP protokol je dizajniran za njegovo ponovno slanje. Međutim, UDP nema mogućnost ponovnog slanja i stoga ne rješava problem izgubljenih paketa. Primjerice, kada se paketi odbace tijekom komunikacije u stvarnom vremenu, kao što je audio ili video sesija dolazi do varijacije kašnjenja i praznine u govoru, [25].

4.2. Zahtjevi aplikacija za kvalitetom usluge

Mnoge mrežne aplikacije dobro rade s *Best Effort* uslugama, dok druge imaju stroge zahtjeve za kvalitetom usluge i rade samo kada je ona zajamčena. Kada je riječ o interaktivnim aplikacijama u stvarnom vremenu, nužna je implementacija QoS-a kako bi se osigurala dovoljna propusnost i omogućila kontrola kašnjenja, kolebanja kašnjenja i gubitka paketa. Postoje brojne aplikacije koje se isporučuju putem višeslužnih mreža i svaka od njih ima individualne QoS zahtjeve, [24]. U nastavku

rada opisani su zahtjevi za kvalitetom usluge pojedinih podatkovnih, glasovnih i video aplikacija.

4.2.1. Podatkovne aplikacije

Podatkovne aplikacije poput prijenosa datoteka i e-pošte primjeri su elastičnih aplikacija koje se mogu prilagoditi promjeni QoS parametara. Elastične aplikacije nazivaju se i *Best Effort* aplikacije. One ne zahtijevaju zajamčenu propusnost i imaju visok stupanj tolerancije na kašnjenje i kolebanje kašnjenja. U slučaju niske propusnosti, prijenos datoteke ili e-pošte uzrokuje kašnjenje od nekoliko milisekundi ili sekundi. Ukoliko dođe do većeg kašnjenja paketa, te ako paket zbog toga bude odbačen, zahvaljujući TCP protokolu paket će biti ponovno poslan što neće znatno utjecati na kvalitetu usluge. Međutim, ukoliko je intenzitet gubitka paketa veći od 10% kašnjenje se značajno povećava što negativno utječe na korisničko iskustvo. Većina podatkovnih aplikacija zahtjeva visoku pouzdanost isporuke paketa, osobito ako je riječ o aplikacijama Internet bankarstava. Stoga je vrlo bitno da se paketi isporučuju ispravno i bez gubitaka, [11], [24].

4.2.2. Glasovne aplikacije

Glasovne aplikacije odvijaju se u stvarnom vremenu i po prirodi su uglavnom neelastične jer zahtijevaju stroga jamstva QoS-a. Međutim pojedine aplikacije mogu imati sposobnost prilagodbe određenim promjenama parametara QoS-a. Na primjer, VoIP se može prilagoditi manjoj propusnosti korištenjem učinkovitijih kodeka. Kodek je softverski ili hardverski uređaj koji pretvara analognu informaciju u digitalni tok za prijenos i ponovno pretvara u analognu informaciju na odredištu, [11].

Kodeci su dizajnirani da komprimiraju informacije u cilju minimiziranja zahtijevanog kapaciteta i prostora za memoriranje. To neizbježno vodi do određenog smanjenja kvalitete prenesene informacije, stoga dizajneri nastoje upotrijebiti minimalnu širinu pojasa i/ili memorijskih kapaciteta istovremeno osiguravajući prihvatljivu kvalitetu prenesenih informacije, [20]. U tablici 1 prikazani su zahtjevi za širinom prijenosnog pojasa različitih kodeka. Ovi podaci ne uključuju opterećenje drugog sloja i ne uzimaju u obzir moguće sheme kompresije. Iz tablice je vidljivo da VoIP kodek G.711 sa zadanom brzinom paketiranja od 50 paketa po sekundi, svakih 20 ms generira novi VoIP paket veličine 160 bajtova i zahtjeva širinu pojasa od 80 kb/s. U odnosu na druge

kodeke navedene u tablici kodek G.711 zahtjeva veću širinu pojasa, ali osigurava i veću kvalitetu zvuka, [26].

Tablica 1. Karakteristike različitih kodeka

| Kodek | Interval paketiranja | Veličina paketa (Bajt) | Brzina paketiranja | Širina pojasa |
|---------------|----------------------|------------------------|--------------------|---------------|
| G.711 | 20 ms | 160 | 50 (pak/s) | 80 kb/s |
| G.711 | 30 ms | 240 | 33 | 74 kb/s |
| G.729A | 20 ms | 20 | 50 | 24 kb/s |
| G.729A | 30 ms | 30 | 33 | 19 kb/s |

Izvor: [26]

Različiti kodeci korisnika utječu na broj generiranih paketa za vrijeme jednog razgovora kao i na duljinu paketa. Veći paketi mogu smanjiti opterećenje u mreži. U tom slučaju prikuplja se nekoliko audio uzoraka i spaja u jedan paket, ali dulji intervali pakiranja povećavaju kašnjenje. Štoviše, povećana je i osjetljivost na gubitak paketa, jer jedan paket uključuje prilično dugačak niz uzastopnih audio uzoraka i njegov gubitak rezultira prazninom u razgovoru. Gubitak paketa ne donosi samo probleme u komunikaciji, nego može uzrokovati i prekid poziva. Kako bi se osigurala kvaliteta glasovne usluge intenzitet gubitka paketa ne smije biti veći od 1%, [11].

Za razliku od podatkovnih aplikacija koje često imaju obilježje usnopljenosti⁶, prijenos glasa zahtjeva konstantnu količinu kapaciteta linka. Također glasovne aplikacije imaju malu toleranciju na kašnjenje i varijacije kašnjenja. Razgovori postaju teško razumljivi kada je vrijeme kašnjenja veće od 100 ms, a kada je kašnjenje signala veće od 200 ms, kvaliteta usluge je neprihvatljiva. Preporučeno maksimalno dopušteno kašnjenje za VoIP iznosi 150 ms u jednom smjeru, [20]. Visoka razina varijacije kašnjenja može dovesti do pojave jeke, izobličivosti zvuka i prekida poziva. Kako bi se to izbjeglo varijacija kašnjenja ne bi smijala iznositi više od 30 ms, [26].

⁶ Usnopljenost je obilježje komunikacije koje se odnosi na prijenos podataka s prekidima u nizu. Nastaje prijenosom velike količine podataka u kratkom vremenu što rezultira dosegom praga raspoložive širine pojasa.

4.2.3. Video aplikacije

Glasovne i video aplikacije imaju mnogo toga zajedničkog. Ako su interaktivne, obje vrste aplikacija osjetljive su na kašnjenje, varijacije kašnjenja i gubitak paketa. Međutim, potrebna propusnost za prijenos video sadržaja mnogo je veća, što uvelike ovisi o razini kvalitete koju korisnik želi postići ili koju podržava video oprema. Zahtjevi za propusnost ovise o nekoliko parametara sustava, kao što su dubina boje, razlučivost zaslona, broj sličica u sekundi i prihvatljiva degradacija kvalitete uzrokovana kompresijom, [11].

Videokonferencija uključuje prijenos audio i video zapisa između dva ili više korisnika. Zahtijevaju prilično visoku propusnost za dobru kvalitetu. Treba napomenuti da je potrebna propusnost u oba smjera prijenosa podataka, a svaki dodatni korisnik u videokonferenciji zahtijeva veću propusnost. Kvaliteta videokonferencije uvelike ovisi o korištenoj internetskoj vezi. Kućne internetske veze velike brzine imaju brzine silaznog linka u rasponu od 256 kb/s do 25 Mb/s i brzine uzlaznog linka u rasponu od 128 kb/s do 3 Mb/s. Uobičajeno se videokonferencije na kućnim internetskim vezama provode u srednje do niskoj kvaliteti zbog ograničenja brzine uzlazne veze. Također zahtijeva se nisko kašnjenje i mali intenzitet gubitka paketa. Kašnjenje veće od približno 100 ms bit će ometajuće, uzrokujući da više sudionika slučajno govori u isto vrijeme. Izgubljeni paket se može zanemariti, što proizvodi blagu pogrešku u audio ili video prijenosu. No, ova će pogreška biti samo trenutna (samo nekoliko sličica video zapisa ili nekoliko milisekundi zvuka), pa je korisnik možda neće ni primijetiti. Stoga je poželjno zanemariti male količine gubitka paketa u interesu održavanja dobrih performansi kašnjenja i varijacije kašnjenja. Zbog toga se videokonferencija naziva uslugom koja je tolerantna na gubitke i osjetljiva na kašnjenje.

Video na zahtjev (engl. *Video on Demand* – VoD) odnosi se na niz aplikacija pomoću kojih korisnici mogu zatražiti pristup video poslužiteljima koji pružaju usluge nepokretne i pokretne slike. Važna značajka VoD aplikacije jest zahtijevanje kapaciteta za dvosmjerni prijenos, s obzirom na to da korisnicima mora biti omogućeno da odabiru željene sadržaje. Vrijeme odziva u videu na zahtjev odnosi se na vrijeme proteklo od generiranja zahtjeva do početka reprodukcije videozapisa. Iako se preferira kratko vrijeme odziva, veća važnost pridodaje se smanjenju kašnjenja tijekom reprodukcije.

Zahtjev za propusnost ovisi broju korisnika kojem video poslužitelj treba pružiti simultanu podršku, [27].

U slučaju da se radi o prijenosu u stvarnom vremenu, video se snima, zatim komprimira i odmah šalje na prijenos. Većina, ali ne sve, sheme video komprimiranja proizvode promet koji zahtijeva varijabilnu brzinu prijenosa, budući da se komprimiranjem postiže prenošenje samo onih uzoraka slike koji se mijenjaju. Veličina zahtijevanog kapaciteta ovisit će o sadržajima koje korisnici gledaju u pojedinom trenutku, [20].

4.3. Mehanizmi kvalitete usluge

Best Effort usluga ne pruža rezervaciju resursa ili bilo koji drugi mehanizam koji se odnosi na pružanje neke vrste posebnog tretmana u mreži. Iz tog razloga, ne može udovoljiti zahtjevima aplikacija koje se odvijaju u stvarnom vremenu. Također, ne bi se trebala koristiti kada mrežni resursi nisu dovoljni za ispunjavanje zahtjeva aplikacije u smislu glavnih QoS pokazatelja. U tim slučajevima kvaliteta krajnjeg korisničkog iskustva mogla bi biti vrlo loša ako ne postoji nikakav drugi mehanizam za upravljanje nepravednošću. Kako bi zadovoljio potrebe pružanja određene razine resursa za aplikacije, IETF je razvio dva mehanizma za osiguravanje dopuštenih ili ugovorenih vrijednosti parametara kvalitete usluge. Jedan se temelji na modelu integriranih usluga (*IntServ*), a drugi na modelu diferenciranih usluga (*DiffServ*), [28].

4.3.1. *IntServ* mehanizmi

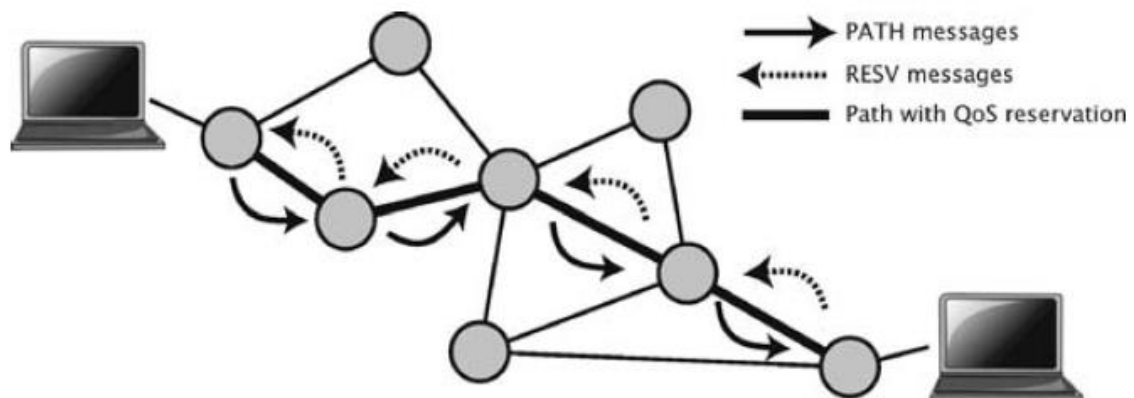
IntServ arhitektura zasniva se na tzv. pristupu po toku i osigurava QoS rješenje s kraja na kraj koristeći signalizaciju, održavanje stanja, te kontrolu pristupa za svaki mrežni element. Integrirane usluge su izvedene u mrežnim čvorovima, a promatraju se pojedinačno za svaki tok. To se ostvaruje na način da se uspostavlja put između krajnjih točaka u mreži uz rezervaciju resursa. Količina rezerviranih resursa ovisi o vrsti prometnog toka, te se uvijek rezervira ona količina resursa koja je potrebna za traženi QoS. Prema [20], u *IntServ* arhitekturi moguće je osigurati dvije vrste usluga:

- Usluga kontroliranog opterećenja (engl. *Controlled load service*) - Kod usluge kontroliranog opterećenja tok podataka ima kvalitetu usluge kakvu bi imao pri

neopterećenoj mreži ili pri malom opterećenju mrežnog elementa u BE modelu. Namijenjena je za aplikacije osjetljive na preopterećenje.

- Garantirana usluga (engl. *Guaranteed service*) - Kod garantirane usluge, postoji čvrsta gornja granica kašnjenja s kraja na kraj, pri čemu su uzeta u obzir kašnjenja svih mrežnih elemenata na putu te propusnost u skladu sa zahtjevima. Ta usluga namijenjena je za vremenski osjetljive aplikacije.

Kako bi se signalizirao i rezervirao željeni QoS za svaki tok u mreži, *IntServ* model koristi protokol za rezervaciju resursa (engl. *Resource Reservation Protocol* - RSVP). Dvije najvažnije signalne poruke RSVP protokola su PATH koja zahtjeva rezervaciju resursa i RESV koja uspostavlja rezervaciju. Pošiljalatelj toka šalje PATH poruke zajedno s uobičajenim paketima. To uzrokuje da svaki usmjerivač na putu pamti svoj prethodni skok, što omogućuje svim RESV porukama koje putuju u suprotnom smjeru da slijede isti put. RESV poruke nose stvarni zahtjev za rezervaciju toka. Podaci o rezervacijama se čuvaju kao meko stanje (engl. *Soft state*). Meko stanje podrazumijeva dinamičko održavanje rezervacijskog stanja pri čemu se rezervacijsko stanje periodički osvježava rezervacijskim porukama, a na isti se način mogu zatražiti i promjene rezervacije. Međutim, ako usmjerivač ne primi nikakvu poruku određeno vrijeme, rezervacija će isteći. Na taj način se sprječava čuvanja nepotrebnih podataka u tablicama usmjeravanja kada krajnji sustavi ne otkažu svoje rezervacije, [11]. Na slici 9 prikazan je postupak uspostave rezervacije RSVP porukama.



Slika 9. Postupak uspostave rezervacije u *IntServ* arhitekturi, [11]

Za realizaciju modela integriranih usluga, usmjerivači moraju biti u stanju pružiti odgovarajući QoS za svaki tok. Funkcija usmjerivača koja pruža različite kvalitete usluge naziva se kontrola prometa, a sastoji se od sljedećih komponenti [29]:

- Klasifikator paketa - identificira i mapira svaki dolazni paket u određenu klasu prometa. Svi paketi koji pripadaju istoj klasi prometa dobivaju isti tretman u mreži. Odabir klase može se izvoditi na osnovu sadržaja, zaglavlja paketa ili dodanim klasifikacijskim brojem koji se dodaje u svaki paket.
- Raspoređivač paketa - upravlja prosljeđivanjem različitih tokova paketa na temelju njihove klase, koristeći upravljanje redom čekanja i različite algoritme za raspoređivanje. Također može kontrolirati ili oblikovati promet kako bi bio u skladu s određenom razinom usluge.
- Kontrola rezervacije resursa - kontrolom upravlja RSVP protokol, a koristi kontrolu politike i kontrolu pristupa. Kada poruka PATH stigne do usmjerivača, kontrola pristupa provjerava jesu li potrebni resursi dostupni, a kontrola politike provjerava ima li korisnik administrativne ovlasti na zahtjev za određenim QoS-om. Ako bilo koja od tih dviju kontrola nije zadovoljena, pošiljatelju se vraća poruka o grešci. Inače, izvršava se rezervacija resursa i postavljaju se parametri u klasifikatoru i raspoređivaču paketa kako bi se osigurao traženi QoS za pakete u skladu s odgovarajućim tokom.

IntServ model pokazao se održivim u mrežama male i srednje veličine, ali u mrežama velikih razmjera pokazao je ozbiljne probleme s skalabilnošću. S obzirom na to da informacije za svaku rezervaciju moraju biti održavane na svakom usmjerivaču duž puta, skalabilnost s velikim brojem tokova kroz mrežu postaje problem. Također, održavanje mekog stanja, kombinirano s kontrolom pristupa na svakom skoku i povećanjem memorijskih zahtjeva pridonosi kompleksnosti svakog mrežnog čvora duž puta. Čak i ako *IntServ* model ima jasne nedostatke, njegove značajke omogućuju da svaka aplikacija kojoj je prihvaćen zahtjev za rezervaciju ima čvrstu garanciju kvalitete usluge, [29].

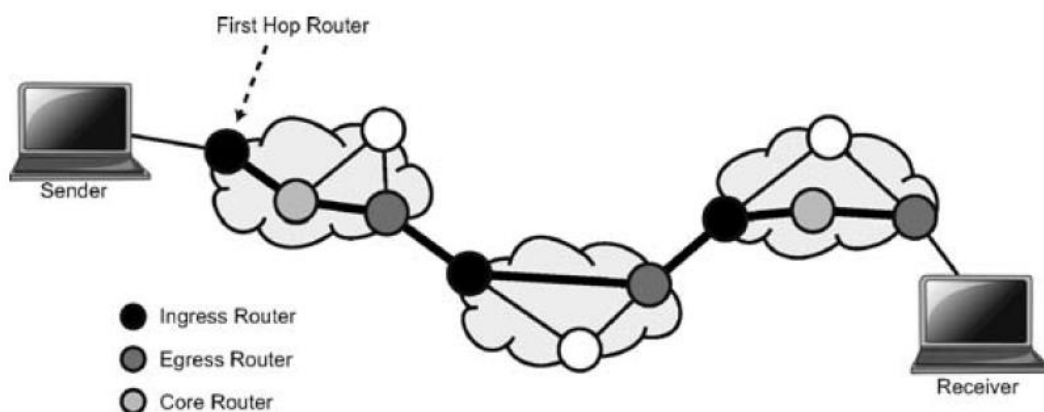
4.3.2. DiffServ mehanizmi

DiffServ mehanizmi imaju za cilj pružiti podršku za QoS u mrežama velikih razmjera, izbjegavajući pritom probleme skalabilnosti koji se pojavljuju u konceptu integriranih usluga. U *DiffServ* arhitekturi mrežni elementi omogućavaju posluživanje

više klasa prometa s različitim QoS zahtjevima. Također, eliminirana je signalizacija za QoS, a broj zahtijevanih stanja koja treba održavati na svakom mrežnom elementu je drastično reduciran, [20].

Diferencirane usluge izgrađene su na temelju koncepta mrežnih domena. Obično mrežne domene odgovaraju određenoj mreži ISP-a (engl. *Internet Service Provider*). Mrežni operateri međusobno pregovaraju o ugovorima o razini usluge koji sadrže specifikacije razine usluge i definiraju QoS parametre, usluge šifriranja, ograničenja usmjerenja itd. *DiffServ* arhitektura prikazana je na slici 10. Unutar mreže može se razlikovati nekoliko vrsta mrežnih čvorova [11]:

- Granični čvorovi - nalaze se na granici prema drugim domenama i obavljaju složene zadatke poput klasifikacije prometa i prilagođavanja uvjetima. Njihova odgovornost je nadzirati dolazni i odlazni promet kontrolirajući njegovu usklađenost s postojećim specifikacijama razine usluge. Ovisno o smjeru prometa, nazivaju se ulazni i izlazni čvorovi mreže.
- Jezgreni čvorovi - nalaze se unutar domene. Oni prosljeđuju pakete prema specifičnom tretmanu naznačenom u zaglavlju paketa.
- *First-hop* čvorovi - predstavljaju poseban slučaj ulaznih čvorova. Izravno su povezani s krajnjim korisnikom i označavaju korisničke pakete kodom diferencirane usluge (engl. *Differentiated Services Code Point* - DSCP) prema dogovorenim pravilima.



Slika 10. DiffServ arhitektura, [11]

Kada paket stigne u ulazni čvor domene obavlja se klasifikacija paketa na način da se paketu dodjeli kod diferencirane usluge. DSCP pohranjen je u prvih šest bitova ToS (engl. *Type of Service*) polja u slučaju IPv4⁷ paketa ili *Traffic Class Octet* polja u slučaju IPv6⁸ paketa. DSCP specificira tretman prosljeđivanja koji paket mora primiti unutar DiffServ domene svakog operatera. Specifični tretmani prosljeđivanja koji se formalno nazivaju PHB (engl. *Per-Hop Behavior*) će biti primijenjeni na svakom mrežnom elementu, omogućavajući tako da paket ima odgovarajuću granicu QoS parametara. Arhitektura DiffServ definira dvije osnovne vrste PHB-a:

- PHB s ubrzanim prosljeđivanjem (engl. *Expedited Forwarding*)
- PHB sa sigurnim prosljeđivanjem (engl. *Assured Forwarding*).

PHB s ubrzanim prosljeđivanjem koristi se da osigura posluživanje apsolutno garantirane kvalitete od jednog do drugog kraja mreže s malim gubicima, kašnjenjem, *jitter*-om i garantiranim propusnim opsegom. PHB sa sigurnim prosljeđivanjem pruža relativne garancije QoS-a, zasnovane na statističkim preduvjetima. Predviđeno je postojanje više klasa, unutar kojih se određuju prioriteta, s ciljem da se spriječi dugotrajno zagušenje mreže.

Osim klasifikacije, čvorovi *DiffServ* arhitekture imaju implementirane funkcije raspoređivanja i upravljanja redovima čekanja. Granični čvorovi izvode i prilagođavane prometa uvjetima upotrebljavajući sljedeće elemente [20]:

- Mjerač prometa - mjeri klasificirani promet i uspoređuje ga s prometnim profilima⁹, odnosno određuje jesu li paketi unutar ili izvan zadanog prometnog profila. Dobivene rezultate prosljeđuje do markera i funkcije za oblikovanje.
- Marker - postavlja odgovarajuću DSCP vrijednost za paket. On uzima u obzir stanje mjerača. Ako je paket izvan profila, on će primiti veću vrijednost prioriteta pri ispuštanju unutar iste klase usluge. U slučaju da je paket unutar profila, on će primiti manju vrijednost prioriteta ispuštanja i bit će zadnji koji će biti ispušten unutar te klase usluge u slučaju zagušenja.

⁷ IPv4 je četvrta verzija IP protokola koja koristi 32-bitnu shemu adresiranja za identifikaciju uređaja u mreži.

⁸ IPv6 je šesta verzija IP protokola koja koristi 128-bitnu shemu adresiranja za identifikaciju uređaja u mreži.

⁹ Prometni profil odnosi se na mjere definirane ugovorom o kvaliteti usluge.

- Funkcija oblikovanja - odgađa određeni promet tako što sprema neke pakete u međuspremnik kako bi bili u skladu s prometnim profilom. Ako ne postoji slobodan prostor u memoriji za čekanje, oni mogu biti odbačeni.
- Funkcija ispuštanja paketa - odbacuje sav promet koji nije u skladu s profilom prometa. Ova radnja se često naziva i prometna policija (engl. *Police traffic*).

Najveći nedostatak *DiffServ* modela je nedovoljno precizno definiranje zahtjeva za kvalitetom usluge i odsustvo rezervacije resursa, zbog čega se ne može u potpunosti garantirati traženi QoS s kraja na kraj. Stoga se ova arhitektura kombinira s MPLS tehnologijom koja omogućuje funkcionalnosti kao što su rezervacija resursa, tolerancija na pogreške i optimizacija iskorištenosti resursa, [11], [20].

5. Opis značajki i modeliranje WFQ metode dodjele kapaciteta primjenom UML dijagrama

U cilju upravljanja zagušenjem i osiguravanja zahtijevane kvalitete usluge svaki usmjerivač mora imati implementiranu određenu metodu posluživanja paketa koja specificira servisnu politiku reda čekanja unutar čvora. U praksi, koriste se razni algoritmi raspoređivanja koji određuju redosljed kojim se paketi odabiru iz reda čekanja i prijenose preko komunikacijskog kanala. Glavni problem proizlazi iz nemogućnosti dodjele kapaciteta određenom toku podataka u svakom trenutku. GPS predstavlja idealnu politiku koja pretpostavlja posluživanje svih prometnih tokova u beskonačno malim količinama kako bi se postiglo da su, u bilo kojem trenutku, svi tokovi prometa pošteno posluženi. Ovim poglavljem detaljno su opisane funkcionalnosti i način rada WFQ metode posluživanja koja planira redosljed stvarnog prijenosa paketa primjenom GPS politike, [24].

5.1. Značajke metoda dodjele kapaciteta

U višeuslužnim IP mrežama paketi se prosljeđuju od izvora do odredišta preko posredničkih uređaja, koja se nazivaju usmjerivači. Svaki paket sadrži adresu izvora i odredišta kako bi usmjerivači znali kamo poslati paket i kako bi odredište znalo tko je poslao paket. Kada paketi stignu na usmjerivač brže nego što se mogu obraditi i proslijediti, usmjerivač sprema dolazne pakete u red čekanja. U redu čekanja paketi se nalaze u memoriji i čekaju na posluživanje, odnosno na usmjeravanje kako bi mogli nastaviti svoj put od izvora do odredišta. Zagušenje mreže nastaje kada je broj paketa u redu čekanja velik dulje vrijeme. Što je red čekanja duži, to paketi na kraju reda moraju duže čekati prije prijenosa, čime se povećava njihovo kašnjenje. Zagušenje mreže može uzrokovati potpuno popunjavanje reda čekanja. Kada se to dogodi, dolazni paketi se odbacuju i nikada ne stignu do odredišta, [19].

Kako bi se osigurala kvaliteta usluge u višeuslužnim mrežama, potrebno je upravljanje redovima čekanja. Upravljanje redovima čekanja nastoji ublažiti zagušenje mreže pa ako je potrebno i odbacivanjem paketa. Ako se u prijenosu pojave paketi koji su uslijed kašnjenja postali neupotrebljivi, tada takvi paketi bivaju odbačeni kako bi oslobodili kapaciteti jer će ih primatelj podataka ionako odbaciti. Upravljanje redovima

čekanja koristi tri osnovne strategije odbacivanja paketa u slučaju popunjenosti međuspremnik:

- *Tail drop* - odbacuje zadnji pristigli paket
- *Front drop* - odbacuje prvi paket u redu čekanja
- *Random drop* - odbacuje nasumično odabrani paket unutar reda čekanja.

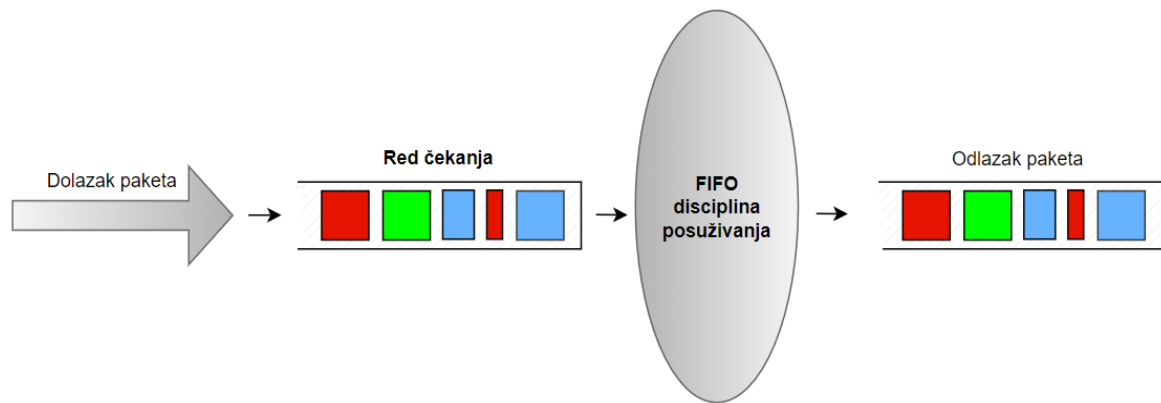
Također postoje dinamičke strategije za odbacivanje paketa prije negoli dođe do zagušenja:

- *Random Early Detection* (RED) - otkriva početno zagušenje izračunavanjem prosječne veličine reda čekanja. Ako prosječna veličina reda čekanja prijeđe fiksni prag, RED odbacuje ili označava svaki pristigli paket s određenom vjerojatnošću odbacivanja.
- *Weighted Random Early Detection* (WRED) - selektivno odbacuje pakete na temelju njihovog prioriteta slijedeći pravilo da je manja vjerojatnost da će paketi s višim prioritetom biti odbačeni nego paketi s nižim prioritetom.
- *Flow-based Weighted Random Early Detection* - predstavlja posebnu značajku WRED-a, koja omogućava veću pravednost za sve tokove ovisno o tome kako se paketi ispuštaju, [29].

Upravljanje čekanjem u redu osigurava održavanje i poštivanje prioriteta raspoređivanja raspoloživih kapaciteta koristeći algoritam raspoređivanja ili planiranja koji određuje koji paket će biti slijedeći poslan na posluživanje. Algoritmi raspoređivanja posluživanja koriste se za dodjeljivanja kapaciteta prijenosnog linka u cilju minimiziranja nedostatka resursa i osiguravanja pravednosti između tokova podataka koji koriste mrežne resurse. Implementacija algoritama raspoređivanja ovisi o primijenjenoj metodi dodjele kapaciteta odnosno disciplini posluživanja, [20]. Postoji mnogo disciplina posluživanja od kojih su najpoznatije: FIFO, PQ, RR, FQ (engl. *Fair Queuing*) i WFQ.

5.1.1. Disciplina posluživanja FIFO

FIFO je najjednostavnija disciplina posluživanja, a često se naziva i FCFS (eng. *First Come First Served*). Kao što je prikazano na slici 11, paketi se poslužuju prema redosljedu dolaska neovisno o njihovoj veličini, vrsti, sadržaju ili bilo kojim drugim karakteristikama paketa.



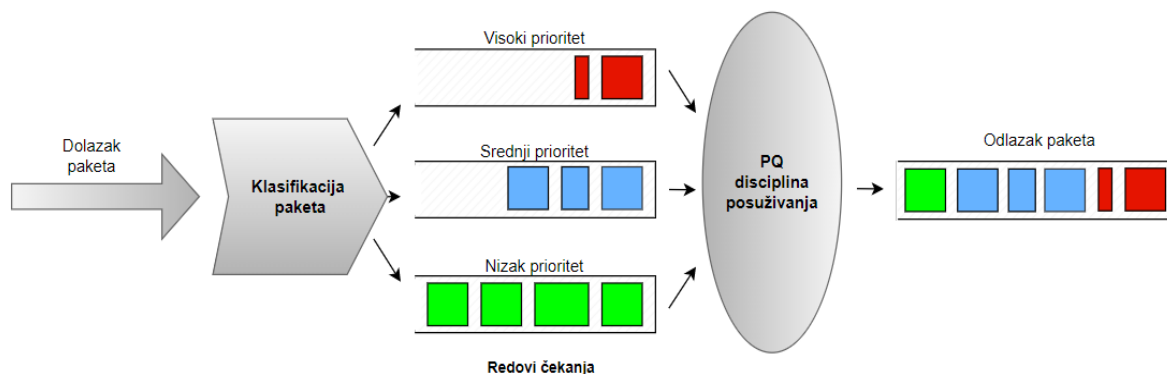
Slika 11. Disciplina posluživanja FIFO

Izvor: [25]

FIFO ne utjelovljuje koncept prioriteta ili klasifikacije prometa te postoji samo jedan red čekanja, stoga se svi paketi tretiraju jednako. S obzirom na IP usmjeravanje, glavna prednost FIFO metode je jednostavnost i učinkovitost u slučaju malog kašnjenja i minimalnog zagušenja. Tijekom zagušenja, FIFO stvara potencijalni problem za provedbu QoS-a. Budući da ne razlikuje duljine paketa, mali paketi mogu doživjeti velika kašnjenja dok se dugi paketi prenose. Pod pretpostavkom da kratke pakete generira aplikacija koja zahtijeva realno vrijeme i QoS, ova karakteristika pridonosi povećanoj sljepoći usmjerivača za kritične aplikacije, [25].

5.1.2. Disciplina posluživanja s prioritetom

U klasičnom posluživanju s prioritetom (engl. *Priority Queuing* - PQ), paketi se prvo klasificiraju od strane sustava, a zatim se smještaju u redove čekanja s različitim klasama prioriteta, kao što je prikazano na slici 12. Klasa prioriteta paketa može ovisiti o eksplicitnoj oznaci u zaglavlju paketa (npr. vrijednost ToS polja u IPv4 paketu), njegovoj izvornoj ili odredišnoj IP adresi i sl. U PQ disciplini prvo se poslužuje red s najvišom klasom prioriteta, a zatim se poslužuju ostali redovi čekanja. Izbor između paketa u istoj klasi prioriteta obično se izvodi na FIFO način.



Slika 12. Disciplina posluživanja s prioritetom

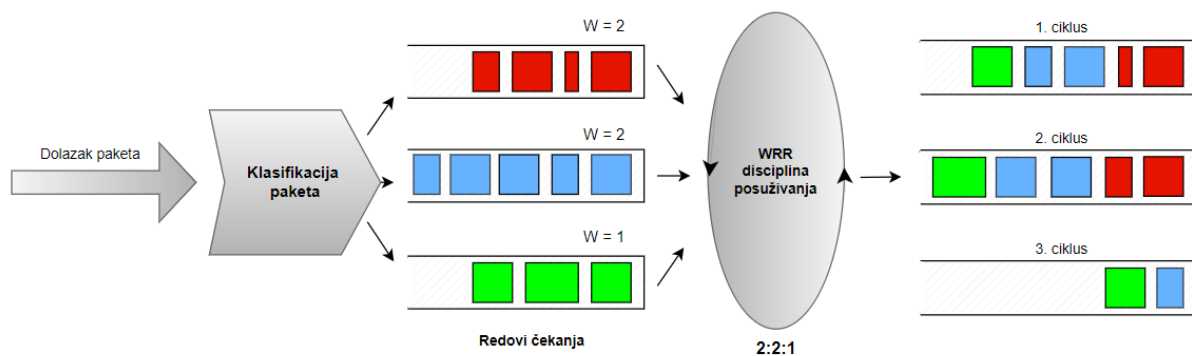
Izvor: [30]

Jedan od najvećih problema pri korištenju PQ-a je velika količina prometa visokog prioriteta. Ako obujam prometa višeg prioriteta postane pretjeran, promet nižeg prioriteta može se odbaciti jer se prostor međuspremnik koji je dodijeljen redovima čekanja niskog prioriteta počne prelijevati. To bi moglo dovesti do potpunog gubitka mrežnih resursa za promet nižeg prioriteta. Kako bi se spriječilo da se to dogodi, koriste se politika pristupa i kontrola prometa za ograničavanje količine prometa visokog prioriteta u redovima čekanja. Posluživanje s prioritetom može se odvijati bez prekida (engl. *non-preemptive*) gdje se posluživanje paketa ne prekida nakon što je započelo ili s prekidom (engl. *preemptive*) gdje se trenutno posluživanje paketa prekida kako bi se poslužio pristigli paket višeg prioriteta, [30].

5.1.3. Disciplina posluživanja *Round Robin*

Round Robin (RR) je jedna od najjednostavnijih metoda dodjele kapaciteta, odnosno algoritama raspoređivanja posluživanja. U RR metodi paketi se klasificiraju i stavljaju u redove čekanja. Klasifikacija se može izvršiti prema vrsti paketa, vrsti aplikacije ili bilo kojoj drugoj karakteristici paketa. U klasičnoj RR metodi paketi se poslužuju po jedan iz svakog reda čekanja u kružnom procesu. Nijedan red čekanja nema prioritet nad ostalima, a ako su veličine paketa iz svih redova čekanja (otprilike) iste, širina pojasa se efektivno dijeli između redova. Međutim, ako red čekanja ima pakete veće veličine od ostalih redova čekanja, taj red čekanja zauzima više pojase širine od ostalih redova. Iako RR rješava problem nedostatka mrežnih resursa za pojedine redove čekanja, on ne uzima u obzir veličinu paketa i stoga se smatra nepravednim, [31].

Weighted Round Robin (WRR) jedna je od mogućih varijacija RR discipline. WRR dodjeljuje težinu svakom redu čekanja, a na temelju te težine svaki red efektivno dobiva dio kapaciteta izlaznog linka. Ako je bilo koji red čekanja neaktivan, WRR poslužuje sljedeći red čekanja i stoga se neiskorištena širina pojasa za neaktivne redove redistribuira između aktivnih redova čekanja proporcionalno njihovim relativnim težinama. Na slici 13 prikazan je način rada WRR metode gdje je dodjela težina redovima čekanja definirana u smislu paketa, [32].



Slika 13. Disciplina posluživanja WRR

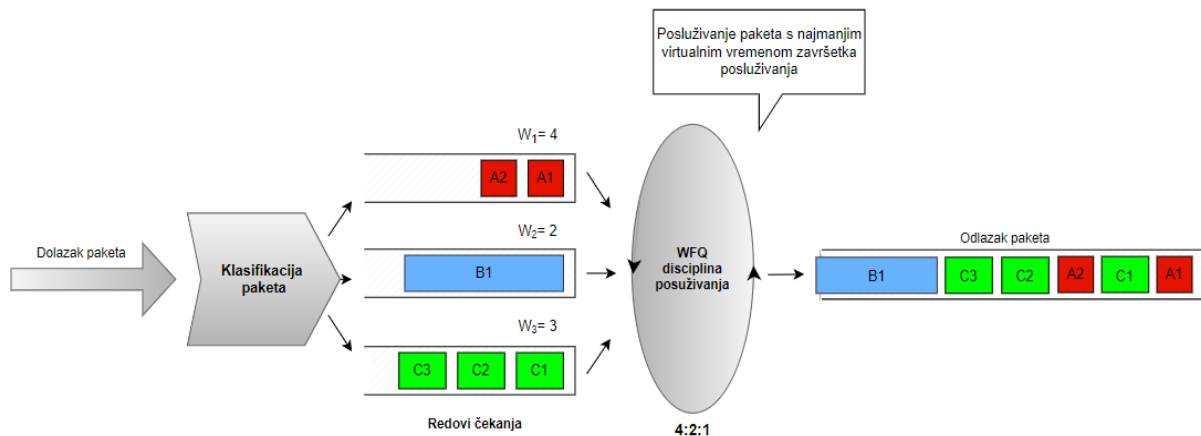
Izvor: [32]

Dolazni paketi klasificirani su u tri klase prometa i smješteni u redove čekanja s omjerom težina 2:2:1. U prvom ciklusu WRR poslužuje dva paketa iz prvog reda čekanja, zatim poslužuje dva paketa iz dugog reda čekanja te na kraju poslužuje jedan paket iz trećeg reda čekanja. S obzirom da su svi redovi čekanja aktivni nakon prvog ciklusa, u drugom ciklusu postupak se ponavlja. U trećem ciklusu prvi red čekanja je neaktivan pa se poslužuju paketi drugog i trećeg reda sukladno njihovim težinama.

5.2. Značajke WFQ metode dodjele kapaciteta

Ponderirano pravedno raspoređivanje kapaciteta linka ili WFQ je dinamička metoda raspoređivanja paketa odnosno dodjeljivanja kapaciteta odlaznog linka, koja omogućava pravednu raspodjelu kapaciteta svim prometnim tokovima. WFQ predstavlja implementaciju generalizirane politike dijeljenja procesora i prirodno proširenje FQ metode. Za razliku od FQ metode koja svakom redu čekanja dodjeljuje jednak udio kapaciteta, WFQ dijeli dostupni kapacitet između redova na temelju njihovih težina. Svakom redu čekanja dodjeljuje se težina koja određuje postotak širine pojasa koji će biti dodijeljen tom redu. Stoga redovi s većom težinom imaju prednost i bit će posluženi ranije. Kada dođe do zagušenja svaki je red zaštićen i tretiran

pravedno u skladu s njegovom težinom. Na slici 14 prikazan je princip rada WFQ metode, [33].



Slika 14. Disciplina posluživanja WFQ

Izvor: [33]

WFQ je metoda posluživanja temeljena na protoku. Pristigli paketi se klasificiraju u tokove, a svaki tok se dodjeljuje FIFO redu čekanja. Tokovi se identificiraju na temelju polja u zaglavlju paketa, kao što su izvorna i odredišna IP adresa, broj protokola, vrsta usluge (ToS), broj izvornog i odredišnog TCP/UDP porta. Na temelju navedenih polja generira se $hash^{10}$ vrijednosti. Budući da paketi istog protoka prometa završavaju s istom $hash$ vrijednošću, dodjeljuju se istom redu čekanja.

U svakom redu čekanja paketima je dodijeljeno virtualno teoretsko vrijeme završetka posluživanja koje WFQ određuje pomoću GPS politike. Sustav GPS fluidnog modela simulira se paralelno sa stvarnim sustavom baziranim na paketima kako bi se izračunalo vrijeme završetka posluživanja za svaki pristigli paket. WFQ poslužuje pakete prema njihovom vremenu završetka odnosno od dostupnih paketa odabire onaj s najmanjom vrijednošću vremena završetka GPS-a. Vrijeme završetka posluživanja dobiveno na temelju GPS politike ne ovisi o kasnijim dolascima paketa. Što znači da paketi koji kasnije stižu u sustav mogu imati kraća virtualna vremena završetka, ali posluživanje paketa u tom trenutku neće biti prekinuto, [33].

WFQ predstavlja idealno rješenje u situacijama kada je poželjno osigurati konstantno vrijeme odgovora za zahtjevnije aplikacije bez dodavanja prekomjerne širine pojasa. Prema [25], najvažniji ciljevi WFQ metode su:

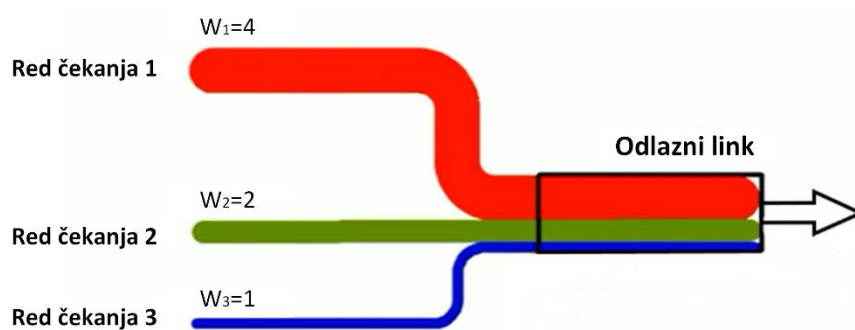
¹⁰ Hash je jedinstvena vrijednost nastala šifriranjem informacija iz zaglavlja paketa.

- Omogućiti poštnu raspodjelu širine pojasa aktivnim redovima
- Omogućiti veću širinu pojasa redovima višeg prioriteta
- Ukloniti nedostatke koje nameću FIFO i PQ metode koje mogu uzrokovati velika kašnjenja, varijacije u kašnjenja i tzv. „izgladnjivanje“ paketa redova čekanja nižeg prioriteta.

Jedna od prednosti WFQ metode je da može podržavati promjenjive veličine paketa pa postoji efektivna pravednost za pakete male veličine u usporedbi s paketima velike veličine. Najveći nedostatak ove metode je to što je iznimno složen jer se raspoređivanje izvodi na temelju izračuna koji ovisi o bitovima koji pripadaju svakom paketu u svakom redu čekanja, [25].

5.2.1. Generalizirano dijeljenje procesora

Generalizirano dijeljenje procesora predstavlja idealizirani algoritam raspoređivanja koji postiže savršenu pravednost između različitih klasa prometa. GPS pretpostavlja da je promet fluidan, tj. beskonačno djeljiv, tako da kad god neka vrsta aplikacije ima pakete u redu čekanja bit će joj dodijeljen zahtijevani dio kapaciteta odlaznog linka. Na slici 15 je vidljiva ilustracija GPS modela gdje su paketi prikazani kao tekućina, a odlazni link kao cijev koja ima određeni ukupni kapacitet. Paketi iz svakog reda čekanja se šalju u cijev istovremeno zauzimajući dio kapaciteta koji im je dodijeljen prema težinama.



Slika 15. Ilustracija GPS modela

Izvor: [34]

GPS model je idealiziran jer pretpostavlja da se redovi mogu posluživati u beskrajno malim količinama. S obzirom da je jedan paket najmanja jedinica koja se može poslužiti iz reda čekanja i zahtjeva određeno vrijeme posluživanja, GPS

predstavlja samo teorijsku politiku na temelju koje razvijeno je nekoliko algoritama za raspoređivanje. WFQ metoda je najpoznatija implementacija GPS-a, a poznata je i pod nazivom *Packet-by-packet Generalized Processor Sharing* (PGPS) jer predstavlja paketnu verziju GPS-a, [34].

U WFQ metodi se za svaki dolazni paket izračunava završni broj (engl. *finish number*). Teoretsko značenje završnog broja je vrijeme kada bi posljednji bit paketa trebao biti poslan ako bi se koristio GPS model. U praksi, završni broj samo je servisna oznaka i ne označava stvarno vrijeme u kojem je paket poslužen. WFQ poslužuje pakete na temelju završnog broja na način da prvo poslužuje pakete s najmanjim završnim brojem odnosno pakete s najmanjim virtualnim vremenom završetka posluživanja.

Za izračunavanje završnog broja svakog paketa, WFQ metoda održava trenutni okrugli broj (engl. *round number*). Okrugli broj odgovara broju krugova koje je završio *Round Robin* algoritam, uz pretpostavku da algoritam poslužuje jedan bit iz svakog aktivnog reda čekanja. Okrugli broj odražava količinu opsluženog prometa za aktivne redove čekanja. Završni broj paketa koji stiže na neaktivnu vezu je zbroj trenutnog okruglog broja i veličine paketa u bitovima. Ako paket stiže na aktivnu vezu, završni broj je zbroj završnog broja prethodnog paketa u redu čekanja i veličine paketa u bitovima. Kada se težine dodjeljuju redovima čekanja, izračun završnog broja uzima u obzir ponderirane veličine paketa. Prema [25] završni broj paketa računa se pomoću formule (1):

$$F(i, k, t) = \max\{F(i, k - 1, t), R(t)\} + P(i, k, t) / w(i) \quad (1)$$

gdje je:

- $F(i, k, t)$ - završni broj k -tog paketa u redu čekanja i s vremenom dolaska t
- $P(i, k, t)$ - duljina k -tog paketa u redu čekanja i s vremenom dolaska t
- $R(t)$ - okrugli broj u trenutku t
- $w(i)$ - težina u redu čekanja i .

Okrugli broj se ažurira pri dolasku ili odlasku svakog paketa. Pri odlasku paketa, red čekanja se smatra neaktivnim kada je najveći završni broj u redu čekanja manji od okruglog broja. Po dolasku paketa, okrugli broj se ponovno izračunava i računa se vrijeme završetka. Što je težina reda veća, to je manje vrijeme završetka za danu

veličinu paketa, što u praksi znači da je više kapaciteta dodijeljeno tom redu čekanja, [25].

5.2.2. Vrste WFQ metode dodjele kapaciteta

Iako se WFQ smatra učinkovitom i pravednom metodom dodjele kapaciteta, sadrži određena ograničenja i nedostatke. Naime, WFQ ne omogućuje konfiguraciju klasifikacije i podržan je samo na sporim vezama (2,048 Mb/s i manje). U slučaju postojanja više istodobnih tokova prometa, zbog svoje pravednosti, ne može pružiti zahtijevanu kvalitetu usluge za vremenski osjetljive aplikacije. U svrhu postizanja bolje učinkovitosti i rješavanja navedenih ograničenja razvijeno je nekoliko vrsta odnosno oblika WFQ metode. Dvije najpoznatije vrste WFQ metode su *Class-Based Weighted Fair Queuing* (CBWFQ) i *Worst-Case Fair Weighted Fair Queueing* (WF2Q), [35].

5.2.2.1. Class-Based Weighted Fair Queuing

CBWFQ pruža bolju fleksibilnost u dodjeljivanju minimalnog iznosa propusnosti na temelju klasa koje je definirao administrator. Umjesto pružanja reda čekanja za svaki tok podataka, koriste se klase koje definira mrežni administrator. Zakazivanje i jamstvo propusnosti ponuđeno svakom redu unutar CBWFQ sustava temelji se na težini koja mu je dodijeljena. Težina se izračunava na temelju propusnosti koju administrator dodjeli određenoj klasi. Ako tok prometa odgovara klasi koju je definirao administrator, odmah se stavlja u takvu klasu, gdje već ima rezervirani kapacitet linka. Ako tok prometa ne odgovara niti jednoj klasi koju je definirao administrator, može koristiti samo preostalu propusnost veze, koja nije rezervirana ni za jednu drugu klasu. Svaka klasa dobiva korisnički definiranu minimalnu garanciju propusnosti, ali može koristiti veću propusnost ako je dostupna.

CBWFQ može koristiti FIFO ili WFQ algoritam raspoređivanja unutar zadanog reda čekanja. U slučaju korištenja WFQ algoritma, kada CBWFQ odluči uzeti paket iz reda čekanja, uzima paket s najmanjim završnim brojem. CBWFQ može stvoriti do 64 reda čekanja, jedan za svaku korisnički definiranu klasu. Ako red čekanja dosegne maksimalno ograničenje broja paketa, dolazi do odbacivanja zadnjeg paketa u redu (engl. *tail drop*). CBWFQ podržava i dinamički metodu ispuštanja paketa WRED koja odbacuje pakete prije nego što se red čekanja zaista ispuni, s namjerom da neke TCP veze reagiraju na izgubljeni paket i uspore slanje paketa. Uz spomenuta 64 reda, uvijek

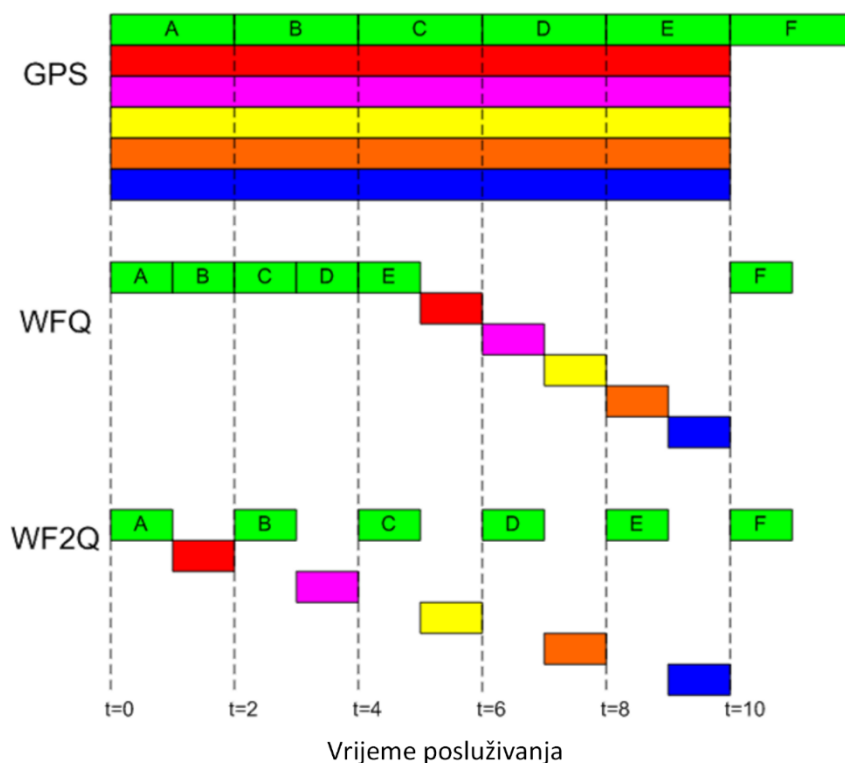
je prisutan red koji se naziva *class-default*. Paketi koji ne odgovaraju niti jednoj od definiranih klasa dodjeljuju se ovom redu čekanja, [35].

CBWFQ metoda je vrlo korisna za izbjegavanje situacija u kojima bi više tokova niskog prioriteta moglo preplaviti tok podataka visokog prioriteta. Primjerice, za prijenos videa strujanjem, gdje se često zahtjeva visoka propusnost, WFQ može osigurati dovoljnu količinu širine pojasa, ali samo u samo u slučajevima kada je prisutan mali broj tokova podataka. U slučajevima kada je prisutno više tokova u isto vrijeme, video sesija će imati manju propusnost, budući da mehanizam WFQ radi na principu pravednosti. Suprotno tome, CBWFQ će uvijek dodijeliti propusnost koju je administrator definirao za tu klasu prometa, [36].

5.2.2.2. Worst-Case Fair Weighted Fair Queueing

Jedan od problema WFQ metode je što redovi s mnogo kratkih paketa dobivaju više prilika za prijenos u odnosu na redove koji sadrže velike pakete. Kako bi se riješio ovaj problem, definirana je modifikacija WFQ-a pod nazivom *Worst-Case Fair Weighted Fair Queueing* (WF2Q). WF2Q se postiže manjom preinakom WFQ algoritma. WF2Q kao i WFQ koristi koncept virtualnog vremena. Virtualno vrijeme završetka je vrijeme kada bi GPS završio posluživanje paketa. Umjesto traženja najmanjeg virtualnog vremena završetka od svih paketa u redu čekanja, WF2Q traži paket s najmanjim virtualnim vremenom završetka, čije se virtualno vrijeme početka posluživanja već dogodilo. WF2Q poslužuje paket koji ima najmanje virtualno vrijeme završetka između paketa u redu čekanja koji su započeli uslugu pod GPS metodom.

WFQ metoda ponekad ne uspijeva aproksimirati GPS. Na slici 16 prikazana je usporedba redosljeda posluživanja u slučaju korištenja WFQ i WF2Q metode. Svi paketi su iste veličine, a brzina prijenosa podataka je jedan paket po jedinici vremena. Postoji šest tokova pri čemu tok F1 ima dodijeljenu težinu od 50% ukupnog odlaznog linka, a tokovi F2-F6 imaju dodijeljenu težinu od 10%. Šest paketa stiže na F1 u isto vrijeme kada pojedinačni paketi stižu na tokove F2-F6. Vidljivo je da WFQ metoda rezultira velikim vremenskim razmakom između posluživanja paketa E i paketa F, iako im je dodijeljena veća težina, [37].



Slika 16. Usporedba WFQ i WF2Q metoda dodjele kapaciteta, [37]

WF2Q metoda je fleksibilna i učinkovita, no nije jednostavna za implementaciju. WF2Q+ predstavlja poboljšanu verziju WF2Q metode koja implementira novu funkciju virtualnog vremena koja rezultira manjom složenošću i većom preciznošću. Kada paket dođe na početak reda čekanja, dodjeljuje mu se virtualno vrijeme početka i virtualno vrijeme završetka. Samo paketi koji se nalaze na početku redova čekanja s virtualnim početnim vremenom manjim ili jednakim trenutnom virtualnom vremenu prihvatljivi su za posluživanje. Među prihvatljivim paketima, za posluživanje se odabire onaj s najmanjim virtualnim vremenom završetka, [38].

5.3. Modeliranje WFQ metode dodjele kapaciteta primjenom UML dijagrama

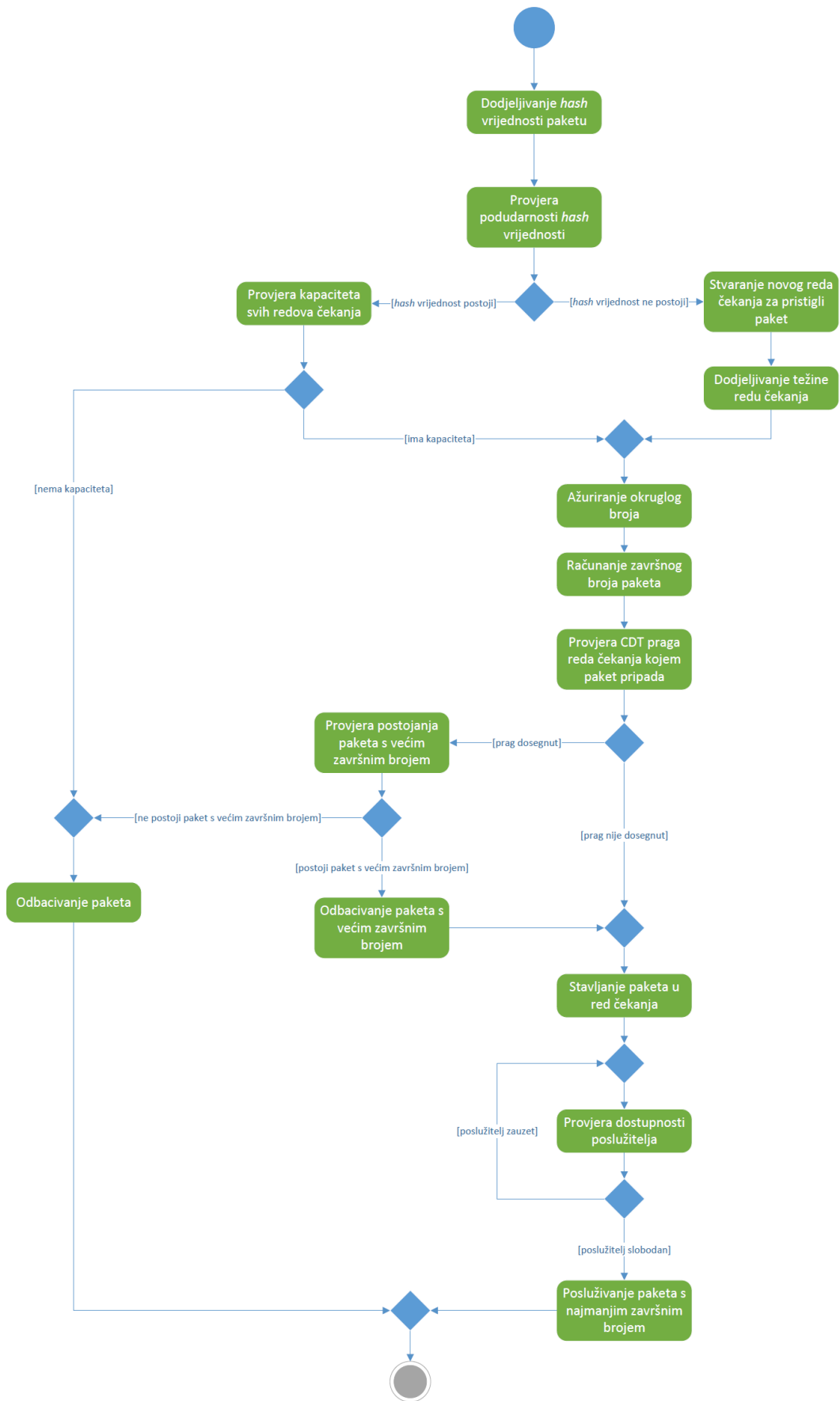
U svrhu boljeg razumijevanja i vizualizacije stvarnog ili zamišljenog sustava koriste se različiti modeli. U nastavku rada, funkcionalnosti WFQ metode dodjele kapaciteta prikazane su primjenom UML jezika koji se koristi za razvoj i modeliranje u području softverskog inženjerstva. Glavni cilj UML-a je definirati standardni način za vizualizaciju dizajna sustava te smanjiti rizik krivog tumačenja njegovog ponašanja. UML jezik služi kao sredstvo za formalni opis strukture i ponašanja sustava, a implementira se u obliku UML dijagrama, [39]. Definirano je nekoliko vrsta UML

dijagrama, a u ovom radu naglasak je na dijagramu aktivnosti i dijagramu međudjelovanja.

5.3.1. Dijagram aktivnosti WFQ metode dodjele kapaciteta

Dijagram aktivnosti opisuje ponašanje sustava prikazujući redoslijed izvođenja aktivnosti pridruženih promatranom objektu. Najčešće se koristi za modeliranje radnog tijeka i opisa rada algoritma, [39]. Na slici 17 prikazan je dijagram aktivnosti koji opisuje sustav posluživanja paketa unutar mrežnog čvora primjenom WFQ metode dodjele kapaciteta. Važno je napomenuti da je dijagram izrađen uz pretpostavku da je zadovoljen uvjet maksimalnog broja redova čekanja, odnosno da je dopušteno stvaranje novih redova. S obzirom da se broj postojećih redova u WFQ sustavu temelji na broju aktivnih tokova, WFQ dinamički stvara i briše redove čekanja.

Dolaznom paketu dodjeljuje se *hash* vrijednost temeljena na njegovim poljima zaglavija. Ako je paket prvi iz novog toka, njegova *hash* vrijednost ne postoji u sustavu, stoga se stvara novi red čekanja za taj tok i dodjeljuje mu se težina. Ako *hash* vrijednost paketa već postoji u sustavu, podrazumijeva se da postoji red čekanja kojem paket pripada. Prije negoli se paket stavi u red čekanja, donosi se odluka o ispuštanju paketa. WFQ koristi modificiranu politiku ispuštanja repa (engl. *modified tail-drop policy*) koja je jasno prikazana dijagramom.



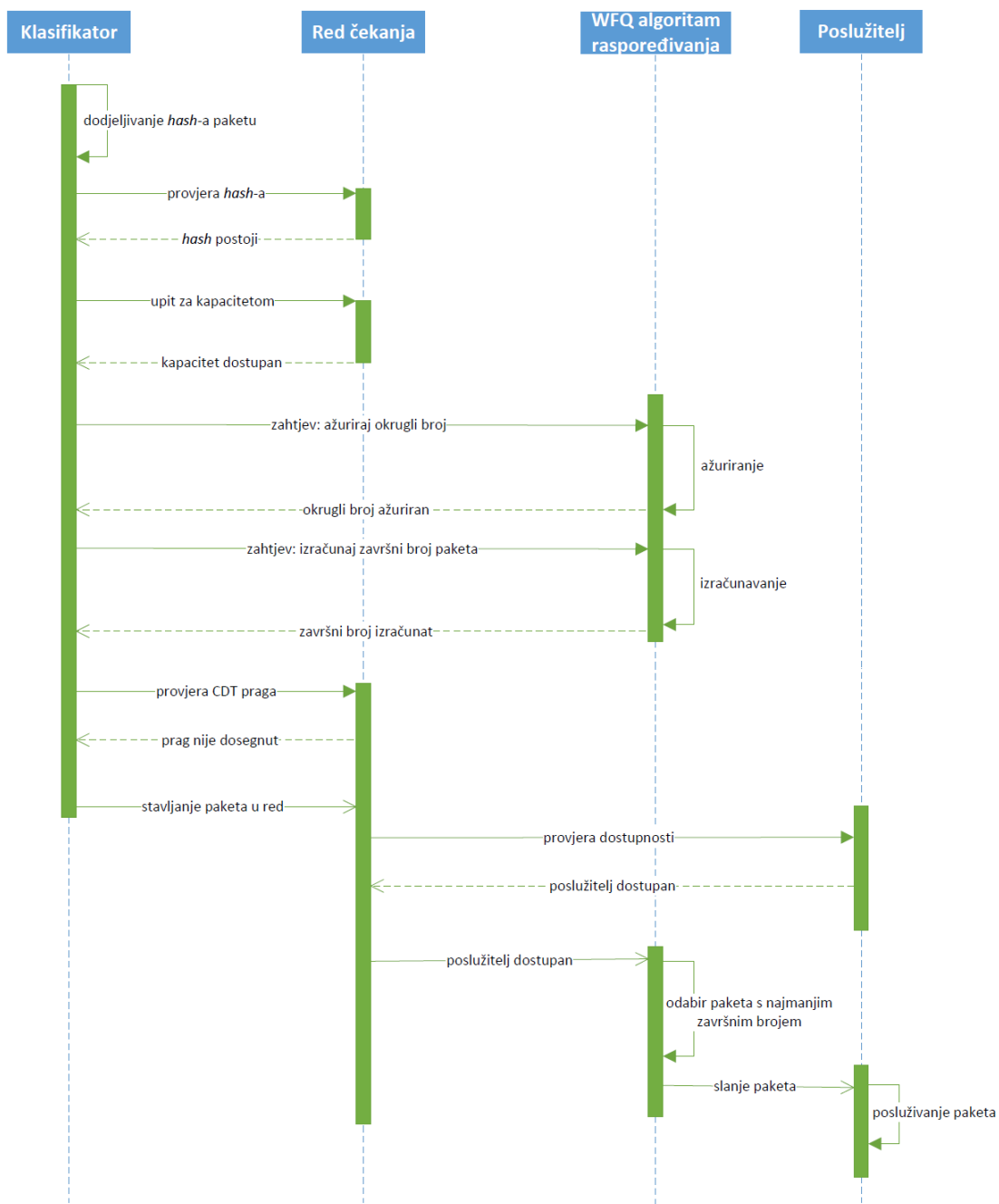
Slika 17. Dijagram aktivnosti WFQ metode dodjele kapaciteta

Odluka o tome hoće li paket biti odbačen temelji se na nekoliko čimbenika, uključujući i završni broj paketa jer je on dio modificirane politike ispuštanja repa. Prva odluka temelji se na kapacitetu koji se odnosi na svu memoriju koju zauzimaju paketi prisutni u WFQ sustavu. Ako paket stigne dok je kapacitet svih redova popunjen, paket se odbacuje. To se naziva WFQ agresivno odbacivanje. U slučaju da je paket dodijeljen praznom redu, on se ne odbacuje.

Sljedeća odluka se temelji na CDT pragu (engl. *Congestive Discard Threshold*) koji ograničava broj paketa u svakom pojedinačnom redu čekanja. Ako je dosegnut CDT u redu čekanja u koji treba staviti pristigli paket, WFQ razmatra odbacivanje dolaznog paketa. Međutim, ako već postoji paket s većim završnim brojem u bilo kojem redu čekanja, WFQ umjesto dolaznog paketa odbacuje paket s većim završnim brojem. Pod određenim okolnostima veličina reda čekanja određenog toka može prekoračiti ograničenje CDT-a, ali na račun drugih paketa u različitim redovima čekanja. Na taj način WFQ pokušava prigušiti najagresivnije tokove koji imaju najveće završne brojeve. Ukoliko dolazni paket nije odbačen, stavlja se u red čekanja. Kada poslužitelj postane dostupan, poslužuje se paket s najmanjem završnim brojem, [33].

5.3.2. Dijagram međudjelovanja WFQ metode dodjele kapaciteta

Dijagram međudjelovanja prikazuje ponašanje sustava naglašavajući njegovu dinamiku. Koristi se za prikaz vremenskog slijeda poruka između objekata u interakciji, [39]. Slika 18 prikazuje dijagram međudjelovanja koji opisuje posluživanje paketa primjenom WFQ metode. Dijagram se odnosi na situaciju u kojoj dolazni paket pripada toku prometa koji se već nalazi u sustavu te je uspješno smješten u red čekanja.



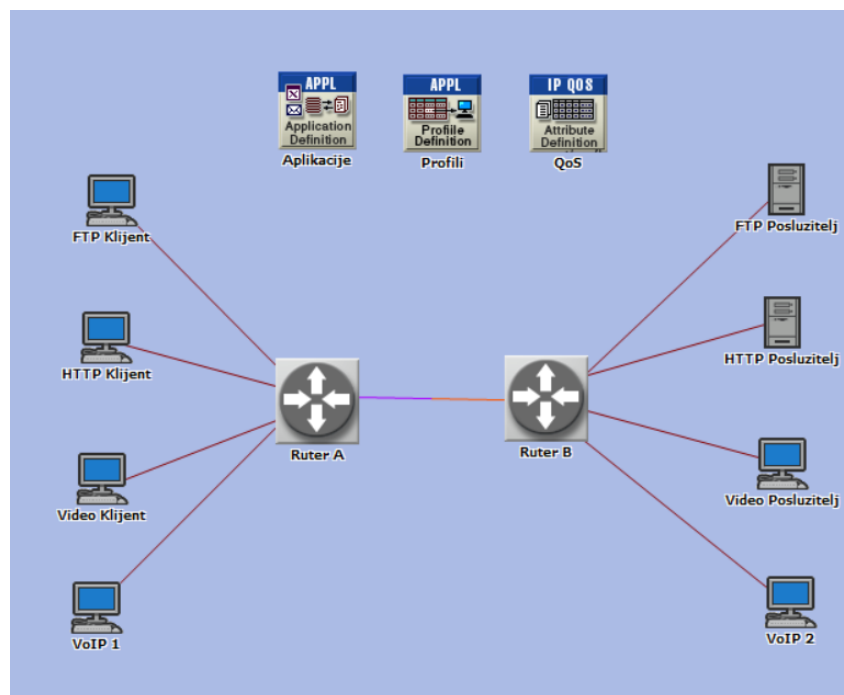
Slika 18. Dijagram međudjelovanja WFQ metode dodjele kapaciteta

Interakcija se odvija između objekata koji predstavljaju komponente unutar WFQ sustava. Radi lakšeg razumijevanja provjera uvjeta modificirane politike ispuštanja repa dodijeljena je klasifikatoru. Osnovna svrha klasifikatora je identifikacija paketa i stavljanje paketa u odgovarajući red čekanja. Kako bi to ostvario klasifikator provjerava jesu li zadovoljeni svi potrebni uvjeti modificirane politike ispuštanja repa.

6. Analiza performansi WFQ metode dodjele kapaciteta

U nastavku rada provedena je analiza performansi WFQ metode u odnosu na druge poznate metode dodjele kapaciteta korištenjem OPNET simulacijskog alata. OPNET je objektno orijentirani simulacijski alat za mrežno modeliranje i QoS analizu simulacijom mrežne komunikacije, mrežnih uređaja i protokola. OPNET predstavlja učinkovit alat za ocjenu performansi različitih metoda dodjele kapaciteta uz visok stupanj fleksibilnosti, [40]. U svrhu analize provedene su mrežne simulacije primjenom različitih metoda dodjela kapaciteta kako bi se odredile veličine parametara kvalitete usluge prilikom prijenosa različitih vrsta aplikacija. Provedena analiza ukazuje na potencijalne prednosti i nedostatke WFQ metode u odnosu na FIFO i PQ metode posluživanja.

Analiza je temeljena na jednostavnoj Ethernet mrežnoj topologiji prikazanoj na slici 19. Mrežna topologija omogućuje prijenos HTTP, FTP, video i glasovnog prometa, a sastoji se od četiri radne stanice (klijenta) koje generiraju promet prema pripadajućim poslužiteljima preko dva usmjerivača. Poslužitelji i klijenti žično su povezani s usmjerivačima putem dvosmjernog linka koji omogućava brzinu prijenosa podataka do 10 Mb/s. Usmjerivači su međusobno povezani dvosmjernim PPP_DS1 linkom koji podržava brzinu prijenosa do 1,544 Mb/s.



Slika 19. Mrežna topologija definirana u svrhu analize

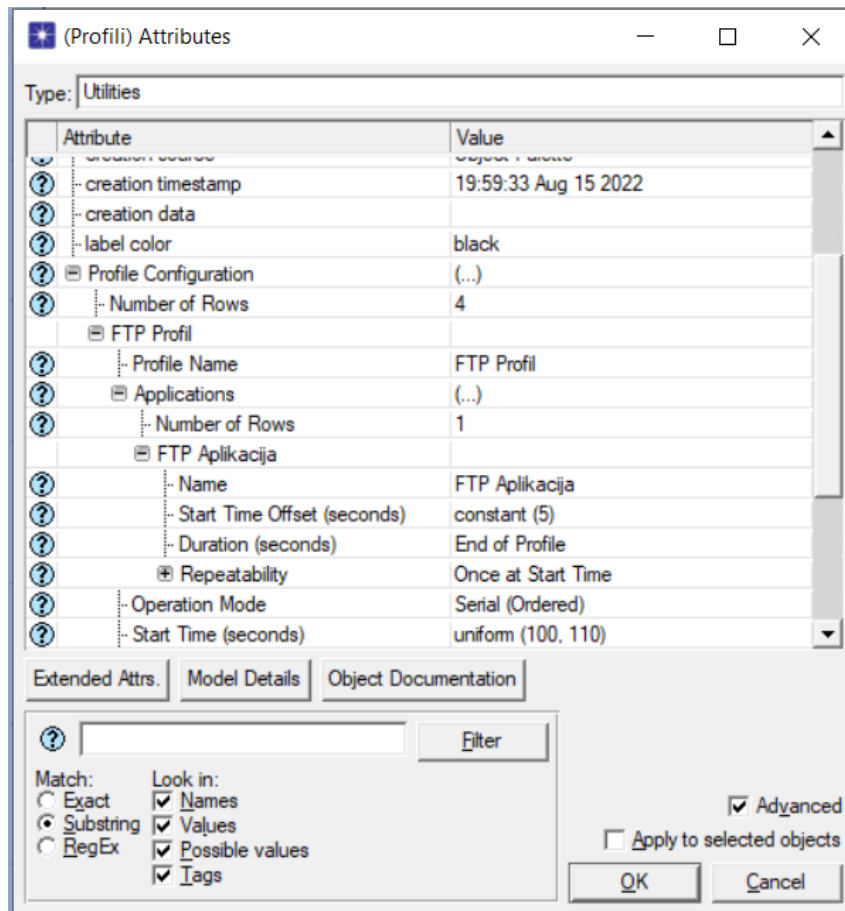
OPNET pruža nekoliko komponenata koje olakšavaju mrežnu konfiguraciju. U svrhu analize korištene su komponente konfiguracije aplikacija, profila i QoS atributa. Komponenta konfiguracije aplikacija koristi se za definiranje i konfiguriranje svih aplikacija u mreži prema zahtjevima korisnika. Na temelju mrežne topologije definirane su i konfigurirane četiri vrste aplikacija kao što je prikazano tablicom 2.

Tablica 2. Konfiguracija aplikacija

| Aplikacija | Opis usluge | Vrsta usluge (ToS) |
|-------------------|--|---------------------------|
| FTP | Prijenos datoteke veličine 1 MB | Best Effort (0) |
| HTTP | Internetsko pretraživanje | Excellent Effort (3) |
| Video | Prijenos videa niske rezolucije | Streaming Multimedia (4) |
| VoIP | Prijenos govora - PCM kvaliteta govora | Interactive Voice (6) |

FTP aplikacija je konfigurirana da podržava *Best Effort* uslugu, a HTTP nastoji pružiti *Excellent Effort* uslugu za postizanje očekivane razine kvalitete usluge. Također, paketi FTP i HTTP aplikacije imaju međudolazna vremena od deset sekundi. Video aplikacija konfigurirana je da podrži prijenos videa niske rezolucije s međudolaznim vremenom zahtjeva od 10 okvira/s i veličinom okvira od 128X120 piksela. VoIP aplikacija podržava interaktivne glasovne usluge pružajući PCM (engl. *Pulse Code Modulation*) kvalitetu govora.

Komponenta konfiguracije profila koristi se za stvaranje korisničkih profila, a svaki profil može sadržavati jednu ili više aplikacija. Za potrebe simulacije kreirano je četiri profila kako bi se odredilo vrijeme početka i završetka prijensa za svaku aplikaciju. Prema zadanim postavkama aplikacije započinju slati zahtjeve nakon 100 sekundi trajanja simulacije. Također je definiran vremenski pomak prijensa prve instance svake aplikacije u trajanju od pet sekundi. Primjer konfiguracije profila za FTP aplikaciju prikazan je na slici 20.

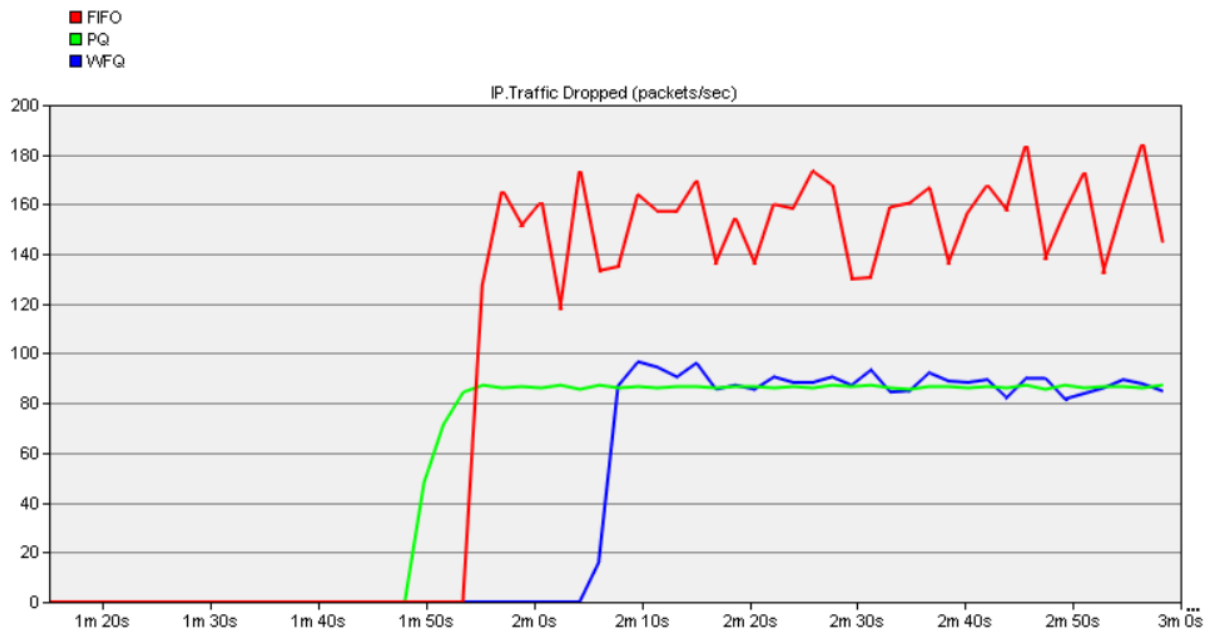


Slika 20. Konfiguracija profila za FTP aplikaciju

Komponenta konfiguracije QoS atributa omogućava definiranje i konfiguraciju različitih metoda dodjele kapaciteta. U svrhu analize koristile su se zadane postavke OPNET alata za implementaciju WFQ, FIFO i PQ metoda posluživanja. Kod primjene WFQ i PQ metode klasifikacija se izvodi na temelju ToS polja zaglavlja paketa.

Na temelju definirane mrežne topologije i prethodno navedenih ulaznih vrijednosti provedene su mrežne simulacije u tri scenarija. U svakom od scenarija primijenjena je drugačija metoda dodjele kapaciteta. Rezultati simulacija dani su u nastavku rada te prikazuju kako pojedina metoda utječe na pojedine parametre kvalitete usluge prilikom prijenosa podatkovnog, video i glasovnog prometa. Trajanje svake simulacije iznosi tri minute.

Graf 1 prikazuje gubitak IP paketa za FIFO, PQ i WFQ metode posluživanja. Broj odbačenih IP paketa po sekundi najveći je uz primjenu FIFO metode, dok PQ i WFQ metode imaju manje stope gubitka paketa na svim sučeljima tijekom simulacije.

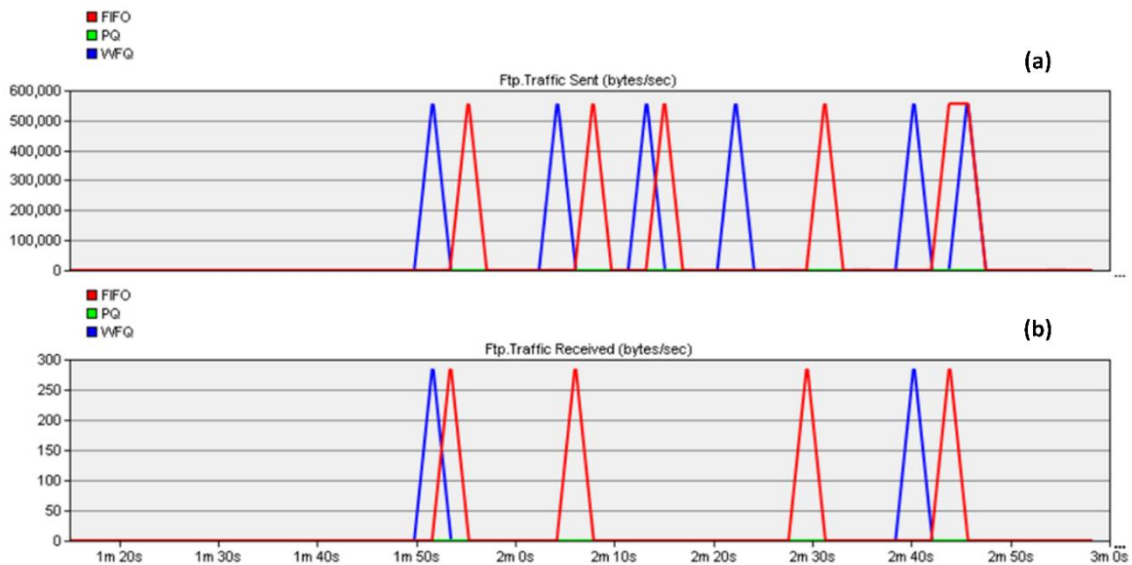


Graf 1. Gubitak IP paketa za FIFO, PQ i WFQ metode posluživanja

Budući da FIFO metoda ima jedan red čekanja i odbacuje pakete bez obzira kojem toku pripadaju, očekuje se da broj odbačenih paketa raste kako se red čekanja puni. Suprotno tome, PQ i WFQ metoda implementiraju višestruke redove čekanja i pružaju uslugu koja ovisi o ToS vrijednosti pridruženoj paketima. FIFO metoda započinje odbacivati pakete nakon 110 sekundi trajanja simulacije i doseže stopu gubitka od 180 paketa po sekundi. PQ metoda ima manju stopu gubitka paketa u odnosu na WFQ, međutim treba uočiti kako je značajno ranije započela odbacivati pakete.

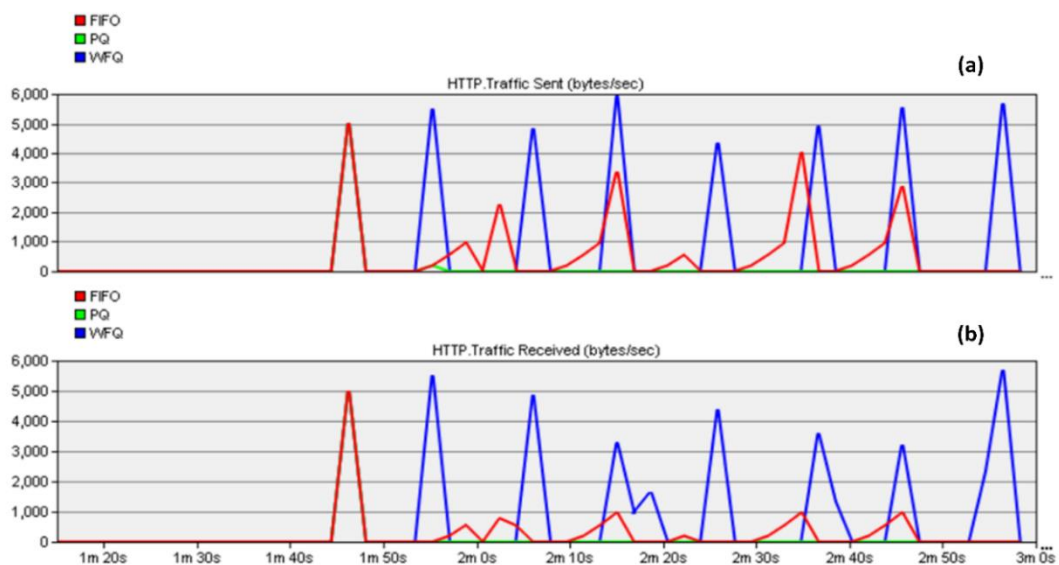
6.1. Analiza prijenosa podatkovnog prometa

Analiza prijenosa podatkovnog prometa temelji se na veličini poslanog i primljenog prometnog toka FTP i HTTP aplikacija. Veličina poslanog prometnog toka odnosi se na prosječni broj bajtova po sekundi koji je dostavljen transportnim slojevima od strane svih FTP i HTTP aplikacija u mreži. Veličina primljenog prometnog toka je prosječni broj bajtova po sekundi koje transportni slojevi prosljeđuju svim FTP i HTTP aplikacijama u mreži.



Graf 3. Veličina poslanog (a) i primljenog (b) prometnog toka FTP aplikacije za FIFO, PQ i WFQ metode posluživanja

Graf 2 prikazuje usporedbu veličine poslanog i primljenog prometnog toka FTP aplikacije za FIFO, PQ i WFQ metode posluživanja. U sva tri scenarija veličina primljenog prometnog toka značajno je manja od one koja je poslana. S obzirom da tok prometa FTP aplikacije predstavlja *Best Effort* uslugu, u slučaju korištenja PQ metode, označen je s najmanjim prioritetom. To je uzrokovalo „izgladnjivanje“ FTP prometa pa je veličina prometnog toka jednaka nuli. Kod WFQ metode, FTP prometnom toku dodijeljena je najmanja težina što je rezultiralo kašnjenjem i odbacivanjem paketa. Najbolje rezultate pokazuje FIFO metoda jer poslužuje pakete neovisno o prioritetu.

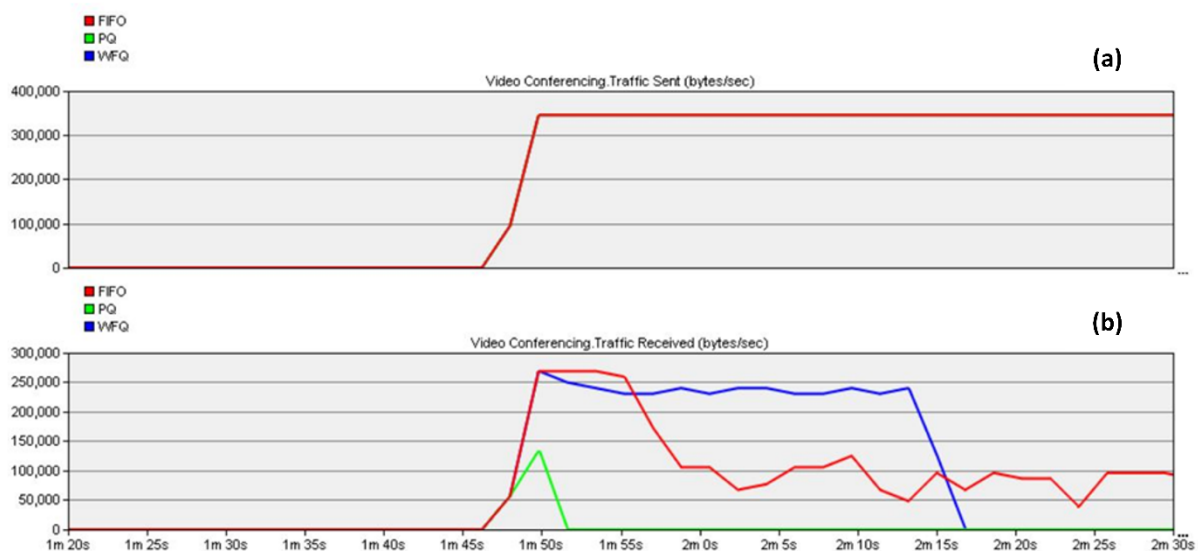


Graf 2. Veličina poslanog (a) i primljenog (b) prometnog toka HTTP aplikacije za FIFO, PQ i WFQ metode posluživanja

Grafom 3 prikazana je usporedba veličine poslanog i primljenog prometnog toka HTTP aplikacije za FIFO, PQ i WFQ metode posluživanja. Najbolji rezultati postignuti su primjenom WFQ metode uz relativno mala kašnjenja. PQ metoda ostvaruje najlošije rezultate jer prednost u posluživanju imaju prometni tokovi video i VoIP aplikacija koji su označeni većim prioritetom.

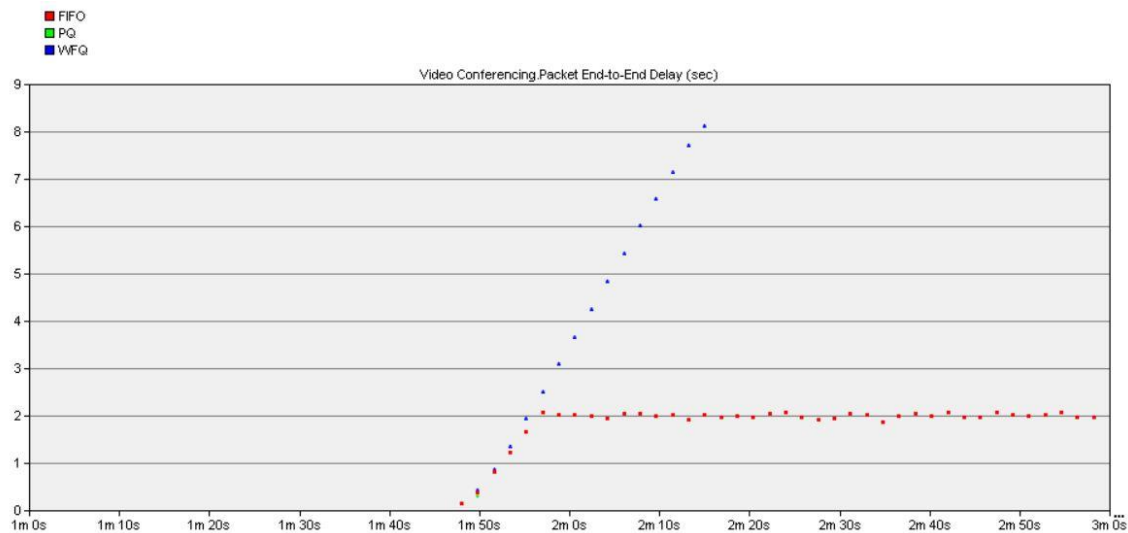
6.2. Analiza prijenosa video prometa

Analiza prijenosa video prometa provedena je na temelju kašnjenja s kraja na kraj te veličine poslanog i primljenog prometnog toka video aplikacije. Kašnjenje s kraja na kraj odnosi se na vrijeme potrebno za prijenos paketa video aplikacije sa izvorišnog na odredišni čvor.



Graf 4. Veličina poslanog (a) i primljenog (b) prometnog toka video aplikacije za FIFO, PQ i WFQ metode posluživanja

Grafom 4 prikazana je usporedba veličine poslanog i primljenog prometnog toka video aplikacije za FIFO, PQ i WFQ metode posluživanja. Veličina poslanog prometnog toka video aplikacije jednaka je za sve tri metode posluživanja te doseže do 350 kB/s. Najmanja veličina primljenog prometnog toka ostvarena je kod PQ metode gdje je uočen značajan gubitak paketa video aplikacije. Najveća veličina primljenog prometnog toka u sekundi ostvarena je primjenom FIFO i WFQ metode. FIFO metoda započinje odbacivati pakete nakon 115 sekundi trajanja simulacije nakon čega veličina prometnog toka varira između 40 kB/s i 130 kB/s. Kod WFQ metode veličina prometnog toka varira oko 230 kB/s te nakon 130 sekundi trajanja simulacije započinje naglo odbacivanje paketa.



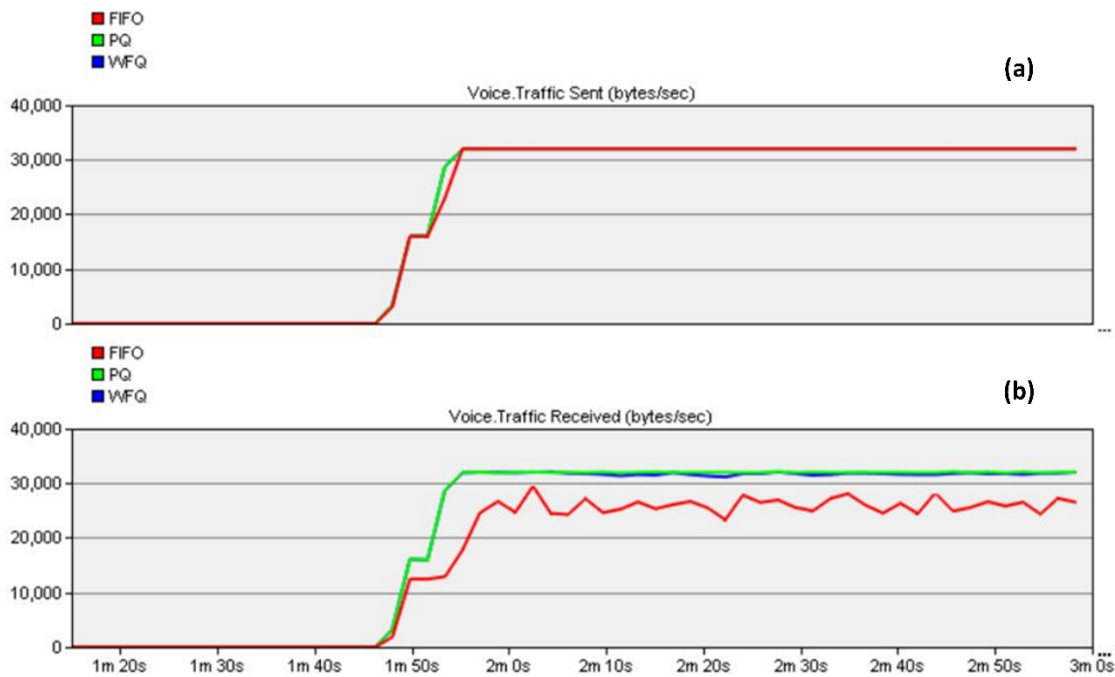
Graf 5. Kašnjenje video prometa za FIFO, PQ i WFQ metode posluživanja

Kašnjenje s kraja na kraj zabilježeno tijekom prijenosa video prometa za FIFO, PQ i WFQ metode posluživanja vidljivo je grafom 5. PQ metoda nema zabilježeno kašnjenje, a razlog tome je velika količina odbačenih paketa. WFQ metoda ima najveće kašnjenje u prijenosu prometa video aplikacije. U 135 sekundi trajanja simulacije, WFQ metoda ima kašnjenje veće od osam sekundi što ne udovoljava zahtjevima kvalitete usluge.

6.3. Analiza prijenosa glasovnog prometa

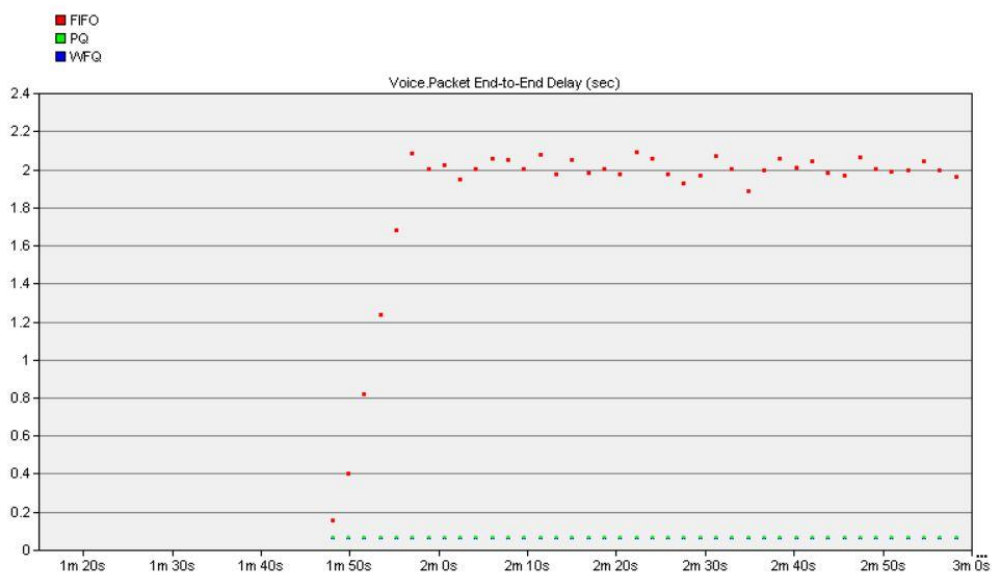
Analiza glasovnog prometa generiranog od strane VoIP aplikacije provedena je na temelju kašnjenja s kraja na kraj te veličine poslanog i primljenog prometnog toka. Također, provedena je analiza varijacije kašnjenja glasovnog prometa koja se odnosi na vremensku razliku u kašnjenju između paketa VoIP aplikacije.

Graf 6 prikazuje usporedbu veličine poslanog i primljenog prometnog toka VoIP aplikacije za FIFO, PQ i WFQ metode posluživanja. Maksimalna veličina poslanog prometnog toka u jedinici vremena jednaka je za sve tri discipline posluživanja i iznosi 35 kB/s.



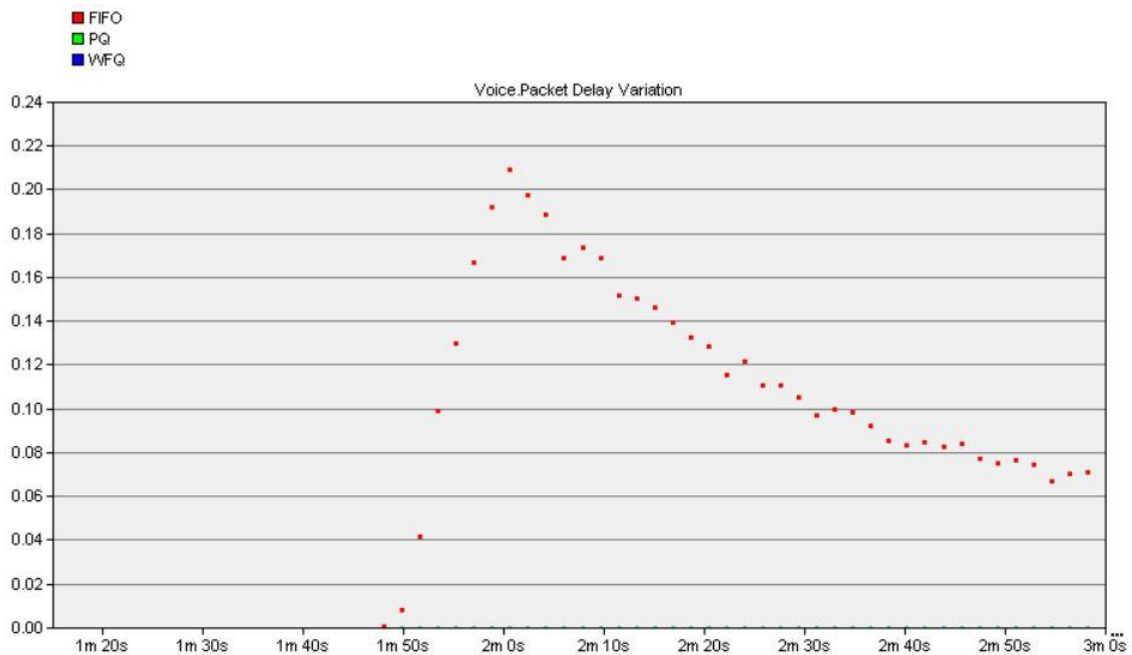
Graf 6. Veličina poslanog (a) i primljenog (b) prometnog toka VoIP aplikacije za FIFO, PQ i WFQ metode posluživanja

Rezultati pokazuju da je najveća veličina primljenog prometnog toka ostvarena primjenom PQ metode. Slične rezultate pokazuje i WFQ metoda uz mala odstupanja. Treba uzeti u obzir da PQ označava VoIP aplikaciju najvećim prioritetom kako bi uvijek imala prednost u posluživanju, dok WFQ pravedno poslužuje sve tokove prometa u mreži u skladu s dodijeljenim težinama. Nezavidne rezultate pokazuje FIFO metoda kod koje su uočeni značajni gubici glasovnog prometa, što rezultira nepovoljnim utjecajem na ukupnu kvalitetu glasovnog signala.



Graf 7. Kašnjenje glasovnog prometa za FIFO, PQ i WFQ metode posluživanja

Grafom 7 prikazano je kašnjenje s kraja na kraj zabilježeno tijekom prijenosa glasovnog prometa za FIFO, PQ i WFQ metode posluživanja. Primjenom PQ i WFQ metoda postignuto je kašnjenje manje od 100 ms čime je zadovoljen uvjet maksimalnog dopuštenog kašnjenja za pružanje zadovoljavajuće kvalitete usluge. FIFO ostvaruje kašnjenje veće od dvije sekunde što rezultira neprihvatljivom kvalitetom usluge VoIP aplikacije.



Graf 8. Varijacije kašnjenja glasovnog prometa za FIFO, PQ i WFQ metode posluživanja

Varijacije kašnjenja zabilježene tijekom prijenosa glasovnog prometa za FIFO, PQ i WFQ metode posluživanja prikazane su na grafu 8. Vidljivo je da WFQ i PQ metode imaju zanemarivo male varijacije u kašnjenju koje su približno jednake nuli. Kod primjene FIFO metode zabilježene su varijacije kašnjenja od 210 ms što daleko premašuje preporučeno ograničenje za postizanje zadovoljavajuće kvalitete usluge.

7. Zaključak

Prometne karakteristike vremenski osjetljivih aplikacija zahtijevaju određenu kvalitetu usluge u smislu zahtjeva za širinom pojasa i kašnjenjem. Najveći problemi u višeuslužnoj mreži povezani su s dodjelom mrežnih resursa, poput međuspremnik usmjerivača i širine pojasa, različitim vrstama aplikacija. Kako bi se maksimizirale mrežne performanse, ograničena količina resursa mora se učinkovito dijeliti između velikog broja različitih tokova prometa što uvelike ovisi o implementiranoj metodi dodjele kapaciteta.

Analizom rezultata simulacija provedenih u radu vidljivo je da WFQ metoda dodjele kapaciteta pokazuje relativno dobre performanse u većini scenarija primjene. S obzirom da FIFO metoda ne koristi višestruke redove čekanja i tretira sav mrežni promet jednako, ne može udovoljiti strogim zahtjevima za kvalitetom usluge glasovnih i video aplikacija. Suprotno tome, PQ metoda dodjeljuje najveći prioritet prometu VoIP aplikacije što rezultira nedostatkom mrežnih resursa za podatkovni i video promet. Za razliku od FIFO i PQ metoda, WFQ pravedno dodjeljuje udio širine pojasa svim tokovima prometa na temelju njihovih težina. Na taj način osigurava da promet većeg prioriteta dobiva više kapaciteta i istovremeno sprječava „izgladnjivanje“ prometa nižeg prioriteta.

WFQ učinkovito nadilazi nedostatke FIFO i PQ metode garantirajući propusnost svim vrstama prometa u mreži. Međutim, postoje situacije kada ne može osigurati visoku kvalitetu usluge za aplikacije u stvarnom vremenu koje zahtijevaju značajne mrežne resurse, poput videokonferencije, osobito kada je opterećenje mreže veliko. Također, u slučaju velikog broja istodobnih prometnih tokova, zbog svoje pravednosti, WFQ dodjeljuje određenu širinu pojasa svakom toku u mreži, što može uzrokovati nedostatak resursa za glasovne i video aplikacije. U svrhu uklanjanja nedostataka, razvijeno je nekoliko modificiranih vrsta WFQ metode koje osiguravaju manje stope gubitka, veću propusnost te u konačnici bolje mrežne performanse.

Literatura

- [1] Mrvelj Š. *Višeuslužne telekomunikacijske mreže i usluge u tehnološkom okruženju*. [Prezentacija] Tehnologija telekomunikacijskog prometa 2. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2020.
- [2] OpenLearn. *Protocols in multi-service networks*. Preuzeto s: <https://www.open.edu/openlearn/science-maths-technology/computing-ict/protocols-multi-service-networks/content-section-1.2> [Pristupljeno: 15. srpnja 2022.]
- [3] Goleniewski L. *Telecommunications Essentials: The Complete Global Source for Communications Fundamentals, Data Networking and the Internet, and Next-Generation Networks*. London: Pearson Education, Inc.; 2002. Preuzeto s: <https://flylib.com/books/en/2.567.1.57/1/> [Pristupljeno: 15. srpnja 2022.]
- [4] Carnet. *Računalne mreže - OSI referentni model*. Preuzeto s: <https://sysportal.carnet.hr/node/352> [Pristupljeno: 15. srpnja 2022.]
- [5] Medium. *The OSI Model*. Preuzeto s: <https://medium.com/@derya.cortuk/the-osi-model-3ace8c01f9b7> [Pristupljeno: 18. srpnja 2022.]
- [6] Mrvelj Š. *DiffServ i analiza posluživanja u čvoru paketne mreže*. [Prezentacija] Tehnologija telekomunikacijskog prometa 2. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2020.
- [7] TechTarget. *OSI model (Open Systems Interconnection)*. Preuzeto s: <https://www.techtarget.com/searchnetworking/definition/OSI> [Pristupljeno: 18. srpnja 2022.]
- [8] GeeksforGeeks. *TCP/IP Model*. Preuzeto s: <https://www.geeksforgeeks.org/tcp-ip-model/?ref=lbpl> [Pristupljeno: 18. srpnja 2022.]
- [9] TechTarget. *Internet Engineering Task Force (IETF)*. Preuzeto s: <https://www.techtarget.com/whatis/definition/IETF-Internet-Engineering-Task-Force> [Pristupljeno: 18. srpnja 2022.]
- [10] Mrvelj Š. *SIP, RTP, RTCP protokoli*. [Prezentacija] Tehnologija telekomunikacijskog prometa 2. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2020.
- [11] Braun T, Diaz M, Enríquez-Gabeiras J, Staub T. *End-to-End Quality of Service Over Heterogeneous Networks*. Berlin: Springer-Verlag Berlin Heidelberg; 2008. Preuzeto s: https://books.google.hr/books?id=uvEHSYLWS7gC&printsec=copyright&redirect_esc=y#v=onepage&q&f=false [Pristupljeno: 19. srpnja 2022.]
- [12] DBpedia. *Stream Control Transmission Protocol*. Preuzeto s: https://dbpedia.org/page/Stream_Control_Transmission_Protocol [Pristupljeno: 19. srpnja 2022.]
- [13] Porter T, Gough M. *How to Cheat at VoIP Security*. Syngress Publishing, Inc.; 2007. Preuzeto s: <https://www.sciencedirect.com/topics/computer-science/time-streaming-protocol> [Pristupljeno: 19. srpnja 2022.]
- [14] Solarwinds. *What Is a Network Node*. Preuzeto s: <https://www.solarwinds.com/resources/it-glossary/network-node> [Pristupljeno: 25. srpnja 2022.]

- [15] GeeksforGeeks. *Types of Node Devices in a Computer Network: End devices and Intermediary Devices*. Preuzeto s: <https://www.geeksforgeeks.org/types-of-node-devices-in-a-computer-network-end-devices-and-intermediary-devices/> [Pristupljeno: 25. srpnja 2022.]
- [16] TechTarget. *Network topology*. Preuzeto s: <https://www.techtarget.com/searchnetworking/definition/OSI> [Pristupljeno: 25. srpnja 2022.]
- [17] Enciklopedija. *Komunikacijska mreža*. Preuzeto s: <https://www.enciklopedija.hr/natuknica.aspx?id=42221> [Pristupljeno: 26. srpnja 2022.]
- [18] Avinetworks. *Packet Switching*. Preuzeto s: <https://avinetworks.com/glossary/packet-switching/> [Pristupljeno: 26. srpnja 2022.]
- [19] Weigle M. *Investigating the Use of Synchronized Clocks in TCP Congestion Control*. Disertacija. University of North Carolina; 2003.
- [20] Mrvelj Š. *Promet u internet mreži*. [Prezentacija] Tehnologija telekomunikacijskog prometa 2. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2009.
- [21] Vegesna S. *IP Quality of Service*. Cisco Press;2001. Preuzeto s: <https://www.oreilly.com/library/view/ip-quality-of/1578701163/> [Pristupljeno: 26. srpnja 2022.]
- [22] WebLogographic. *Razlika između propusnosti i širine pojasa*. Preuzeto s: <https://hr.weblogographic.com/difference-between-throughput> [Pristupljeno: 27. srpnja 2022.]
- [23] Scaler. *Quality of Service in Computer Networks (QoS)*. Preuzeto s: <https://www.scaler.com/topics/qos/> [Pristupljeno: 27. srpnja 2022.]
- [24] Evans JW, Filis C. *Deploying IP and MPLS QoS for Multiservice Networks: Theory and Practice*. Amsterdam: Elsevier; 2010. Preuzeto s: <http://webspn.hit.bme.hu/~telek/oktatas/konyv2.pdf> [Pristupljeno: 27. srpnja 2022.]
- [25] Martins JSB. *Managing IP Networks: Challenges and Opportunities*. U: Aidarous S, Plevyak T. (ur.) *Quality of Service in IP Networks*. Hoboken, USA: John Wiley & Sons, Inc.; 2005. pp. 57 – 142.
- [26] CiscoPress. *Quality of Service Design Overview*. Preuzeto s: <https://www.ciscopress.com/articles/article.asp?p=357102> [Pristupljeno: 27. srpnja 2022.]
- [27] Chen Y, Farley T, Ye N. *QoS Requirements of Network Applications on the Internet*. Researchgate. 2004; 55-76. Preuzeto s: https://www.researchgate.net/publication/229008214_QoS_Requirements_of_Network_Applications_on_the_Internet [Pristupljeno: 30. srpnja 2022.]
- [28] The Cisco Learning Network. *QoS architecture models: IntServ vs DiffServ*. Preuzeto s: <https://learningnetwork.cisco.com/s/question/0D53i00000KsqtXCAR/qos-architecture-models-intserv-vs-diffserv> [Pristupljeno: 30. srpnja 2022.]
- [29] Parziale L, Liu W, Matthews C, Rosselot N, Davis C, Forrester J, Britt DT. *TCP/IP Tutorial and Technical Overview*. IBM Redbook; 2006. Preuzeto s: <https://www.redbooks.ibm.com/redbooks/pdfs/gg243376.pdf> [Pristupljeno: 30. srpnja 2022.]

- [30] Gili ME, Samani A. *The Effect of Queuing Mechanisms First in First out (FIFO), Priority Queuing (PQ) and Weighted Fair Queuing (WFQ) on Network's Routers and Applications*. Scientific Research. 2016;8(5): 77-84. Preuzeto s: <https://www.scirp.org/journal/paperinformation.aspx?paperid=66994> [Pristupljeno: 30. srpnja 2022.]
- [31] WhatWhenHow. *First-In-First-Out, Priority Queuing, Round-Robin, and Weighted Round-Robin Queuing (Congestion Management and Queuing)*. Preuzeto s: <http://what-when-how.com/ccnp-ont-exam-certification-guide/first-in-first-out-priority-queuing-round-robin-and-weighted-round-robin-queuing-congestion-management-and-queuing/> [Pristupljeno: 30. srpnja 2022.]
- [32] WhatWhenHow. *Queuing and Scheduling (QOS-Enabled Networks) Part 2*. <http://what-when-how.com/qos-enabled-networks/queuing-and-scheduling-qos-enabled-networks-part-2/> [Pristupljeno: 5. kolovoza 2022.]
- [33] WhatWhenHow. *Weighted Fair Queuing (Congestion Management and Queuing)*. Preuzeto s: <http://what-when-how.com/ccnp-ont-exam-certification-guide/weighted-fair-queuing-congestion-management-and-queuing/> [Pristupljeno: 5. kolovoza 2022.]
- [34] Intronet works. *Queuing and Scheduling*. Preuzeto s: <http://intronetworks.cs.luc.edu/1/html/queuing.html#gps> [Pristupljeno: 5. kolovoza 2022.]
- [35] WhatWhenHow. *Class-Based Weighted Fair Queuing (Congestion Management and Queuing)*. Preuzeto s: <http://what-when-how.com/ccnp-ont-exam-certification-guide/class-based-weighted-fair-queuing-congestion-management-and-queuing/> [Pristupljeno: 7. kolovoza 2022.]
- [36] Klampfer S, Chowdhury A, Mohorko J, Cucej Z. *Influences of Classical and Hybrid Queuing Mechanisms on VoIP's QoS Properties*. ResearchGate. 2011. Preuzeto s: https://www.researchgate.net/publication/221910737_Influences_of_Classical_and_Hybrid_Queueing_Mechanisms_on_VoIP's_QoS_Properties [Pristupljeno: 7. kolovoza 2022.]
- [37] IPsit. *Worst Case Fair Fair Weighted Queue*. Preuzeto s: <https://ipsit.bu.edu/sc546/sc441Spring2003/wfq/wf2q.htm> [Pristupljeno: 10. kolovoza 2022.]
- [38] Osman EB, Omer M, Ahmed M, Ali M. *A Comparative Analysis of Different Queuing Techniques in Wireless LANs*. Sudan University of Science and Technology. 2016. Preuzeto s: <http://repository.sustech.edu/bitstream/handle/123456789/15238/Acompartive%20analysis%20of%20different%20queueing%20techniques%20in%20WLAN.pdf?sequence=1> [Pristupljeno: 10. kolovoza 2022.]
- [39] Visual Paradigm. *What is Unified Modeling Language (UML)*. Preuzeto s: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/> [Pristupljeno: 15. kolovoza 2022.]
- [40] Opnet projects. *Opnet network simulator*. Preuzeto s: <https://opnetprojects.com/opnet-network-simulator/> [Pristupljeno: 15. kolovoza 2022.]

Popis kratica

| | |
|----------|--|
| ARP | Address Resolution Protocol |
| BE | Best Effort |
| B-ISDN | Broadband Integrated Services Digital Network |
| CBWFQ | Class-Based Weighted Fair Queuing |
| CDT | Congestive Discard Threshold |
| DiffServ | Differentiated Services |
| DSCP | Differentiated Services Code Point |
| FIFO | First-In-First-Out |
| FoIP | Fax over IP |
| FQ | Fair Queuing |
| FTP | File Transfer Protocol |
| GPS | Generalized Processor Sharing |
| HTTP | HyperText Transfer Protocol |
| ICMP | Internet Control Message Protocol |
| IETF | Internet Engineering Task Force |
| IP | Internet Protocol |
| IntServ | Integrated Services |
| ISP | Internet Service Provider |
| MAC | Media Access Control |
| MGCP | Media Gateway Control Protocol |
| MPLS | Multiprotocol Label Switching |
| OSI | Open Systems Interconnection |
| PCM | Pulse Code Modulation |
| PGPS | Packet-by-packet Generalized Processor Sharing |
| PHB | Per-Hop Behavior |
| PQ | Priority Queuing |
| QoS | Quality of Service |
| RED | Random Early Detection |

| | |
|--------|---|
| RR | Round Robin |
| RSVP | Resource Reservation Protocol |
| RTCP | Real-time Transport Control Protocol |
| RTP | Real-time Transport Protocol |
| RTSP | Real-time Streaming Protocol |
| RTT | Round Trip Time |
| SCTP | Stream Control Transmission Protocol |
| SDP | Session Description Protocol |
| TCP | Transmission Control Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| ToS | Type of Service |
| UDP | User Datagram Protocol |
| UML | Unified Modelling Language |
| VoD | Video on Demand |
| VoIP | Voice over IP |
| VPN | Virtual Private Network |
| WF2Q | Worst-Case Fair Weighted Fair Queueing |
| WFQ | Weighted Fair Queuing |
| WRED | Weighted Random Early Detection |
| WRR | Weighted Round Robin |

Popis slika

| | |
|--|----|
| Slika 1. Arhitektura širokopojasne višeuslužne mreže | 4 |
| Slika 2. Postupak enkapsulacije i dekapulacije podataka | 7 |
| Slika 3. Protokolarni složaj za prijenos multimedijskog sadržaja | 9 |
| Slika 4. Mrežni čvorovi u komunikacijskoj mreži | 12 |
| Slika 5. Vrste fizičke mrežne topologije | 14 |
| Slika 6. Komutacija paketa | 16 |
| Slika 7. Usporedba iskoristivosti širine pojasa prije i nakon implementacije QoS-a.. | 19 |
| Slika 8. Efekt kolebanja kašnjenja | 21 |
| Slika 9. Postupak uspostave rezervacije u IntServ arhitekturi | 27 |
| Slika 10. DiffServ arhitektura | 29 |
| Slika 11. Disciplina posluživanja FIFO..... | 34 |
| Slika 12. Disciplina posluživanja s prioritetom | 35 |
| Slika 13. Disciplina posluživanja WRR | 36 |
| Slika 14. Disciplina posluživanja WFQ | 37 |
| Slika 15. Ilustracija GPS modela | 38 |
| Slika 16. Usporedba WFQ i WF2Q metoda dodjele kapaciteta | 42 |
| Slika 17. Dijagram aktivnosti WFQ metode dodjele kapaciteta..... | 44 |
| Slika 18. Dijagram međudjelovanja WFQ metode dodjele kapaciteta | 46 |
| Slika 19. Mrežna topologija definirana u svrhu analize | 47 |
| Slika 20. Konfiguracija profila za FTP aplikaciju | 49 |

Popis tablica

| | |
|---|----|
| Tablica 1. Karakteristike različitih kodeka | 24 |
| Tablica 2. Konfiguracija aplikacija..... | 48 |

Popis grafikona

| | |
|--|----|
| Graf 1. Gubitak IP paketa za FIFO, PQ i WFQ metode posluživanja | 50 |
| Graf 3. Veličina poslanog (a) i primljenog (b) prometnog toka HTTP aplikacije za FIFO, PQ i WFQ metode posluživanja | 51 |
| Graf 2. Veličina poslanog (a) i primljenog (b) prometnog toka FTP aplikacije za FIFO, PQ i WFQ metode posluživanja..... | 51 |
| Graf 4. Veličina poslanog (a) i primljenog (b) prometnog toka video aplikacije za FIFO, PQ i WFQ metode posluživanja | 52 |
| Graf 5. Kašnjenje video prometa za FIFO, PQ i WFQ metode posluživanja | 53 |
| Graf 6. Veličina poslanog (a) i primljenog (b) prometnog toka VoIP aplikacije za FIFO, PQ i WFQ metode posluživanja | 54 |
| Graf 7. Kašnjenje glasovnog prometa za FIFO, PQ i WFQ metode posluživanja | 54 |
| Graf 8. Varijacije kašnjenja glasovnog prometa za FIFO, PQ i WFQ metode posluživanja..... | 55 |

Sveučilište u Zagrebu
Fakultet prometnih znanosti
Vukelićeva 4, 10000 Zagreb


IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOSTI

Izjavljujem i svojim potpisom potvrđujem da je _____ diplomski rad _____
(vrsta rada)
isključivo rezultat mogega vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu, a što pokazuju upotrijebljene bilješke i bibliografija. Izjavljujem da nijedan dio rada nije napisan na nedopušten način, odnosno da je prepisan iz necitiranog rada te da nijedan dio rada ne krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu diplomskog rada pod naslovom _____ Analiza i modeliranje metode dodjele kapaciteta u višeuslužnim mrežama WFQ _____ , u Nacionalni repozitorij završnih i diplomskih radova ZIR.

Student/ica:

U Zagrebu, _____ 2.9.2022. _____

Dora Radičević, 
(ime i prezime, potpis)