

# Usmjeravanje u mrežama s komutacijom paketa temeljeno na modelu najkraćeg puta

---

Miletić, Nikolina

Master's thesis / Diplomski rad

2020

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:119:651536>

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

Download date / Datum preuzimanja: **2024-07-17**



Repository / Repozitorij:

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



**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET PROMETNIH ZNANOSTI**

**Nikolina Miletić**

**USMJERAVANJE U MREŽAMA S KOMUTACIJOM PAKETA**  
**TEMELJENO NA MODELU NAJKRAĆEG PUTA**

**DIPLOMSKI RAD**

---

Zagreb, 2020.

Zagreb, 2. travnja 2020.

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

## DIPLOMSKI ZADATAK br. 5911

Pristupnik: **Nikolina Miletić (0135242751)**  
Studij: **Promet**  
Smjer: **Informacijsko-komunikacijski promet**

Zadatak: **Usmjeravanje u mrežama s komutacijom paketa temeljeno na modelu najkraćeg puta**

### Opis zadatka:

Prikazati proces usmjeravanja u mreži s komutacijom paketa i objasniti ulogu komutacijskih čvorova u tom procesu, s obzirom na njihov položaj u mreži (ulazni, središnji ili izlazni čvorovi).  
Analizirati značajke razvijenih algoritma usmjeravanja koji se temelje na modelu najkraćeg puta.  
U praktičnom dijelu rada napraviti usporednu analizu performansi mreže, sa zadanom topologijom, ovisno o korištenom algoritmu usmjeravanja.

Mentor:

  
\_\_\_\_\_  
prof. dr. sc. Štefica Mrvelj

Predsjednik povjerenstva za  
diplomski ispit:

Sveučilište u Zagrebu  
Fakultet prometnih znanosti

**DIPLOMSKI RAD**

**USMJERAVANJE U MREŽAMA S KOMUTACIJOM PAKETA  
TEMELJENO NA MODELU NAJKRAĆEG PUTA  
SHORTEST PATH ROUTING IN PACKET SWITCH NETWORKS**

Mentor: prof. dr. sc. Štefica Mrvelj

Student: Nikolina Miletić

JMBAG: 0135242751

Zagreb, rujan 2020.

# USMJERAVANJE U MREŽAMA S KOMUTACIJOM PAKETA TEMELJENO NA MODELU NAJKRAĆEG PUTA

## SAŽETAK

S obzirom na to da je način prijenosa podatka u komunikacijskim mrežama paketskim modom obilježje današnjih novih mreža, u radu su analizirane značajke paketskih mreža i razvijenih metoda usmjeravanja u njima. Naglasak je na skupu usmjerivačkih algoritama koji se temelje na modelu najkraćeg puta iz razloga što su ti algoritmi zastupljeniji u primjeni. U radu je prikazan proces usmjeravanja u mreži s komutacijom paketa te je objašnjena uloga čvorova u procesu usmjeravanja s obzirom na njihov položaj u mreži. Nadalje, prikazane su značajke algoritama usmjeravanja koji se temelje na najkraćem putu (Dijkstra, Floyd-Warshall i Bellman-Ford) i načinjena je njihova usporedba. Usmjeravanje temeljeno na modelu najkraćeg puta zapravo određuje minimalni trošak između izvorišta i odredišta, a taj trošak se ogleda u kašnjenju paketa na putu. Korištenjem navedenih algoritama određeni su najkraći putovi za zadanu neopterećenu i opterećenu mrežu, koja se sastoji od 10 čvorova i 13 spojnica. Također su za opterećenu i neopterećenu mrežu određene performanse mreže (kašnjenje na putu i vjerojatnost blokiranja najkraćeg/ prioritetnog puta).

**KLJUČNE RIJEČI:** algoritmi usmjeravanja; najkraći put; usmjeravanje u paketskoj mreži; Dijkstra; Floyd-Warshall; Bellman-Ford, kašnjenje na putu

## SUMMARY

Considering that data transfer mode in packet mode communication networks is a feature of today's new networks, this paper analyzes the characteristics of packet networks and the developed routing methods in them. The emphasis is on a set of routing algorithms based on the shortest path model for the reason that these algorithms are more prevalent in application. Furthermore, the most important features of each shortest path-based routing algorithm are mentioned (Dijkstra, Floyd-Warshall and Bellamn-Ford) and a comparasion of them was made. Shortest path-based routing actually determines the minimum cost between source and destination, and that cost is reflected in the delay of the package on the way. Using the above algorithms, the shortest paths for a given unloaded and loaded network are determined, which

consists of 10 nodes and 13 links. Network performance is also determined for a loaded and unloaded network (delay on the road and probability of blocking the shortest/ priority path).

**KEY WORDS:** routing algorithms; the shortest path; packet routing; Dijkstra; Floyd-Warshall; Bellman-Ford; delay on the road

## SADRŽAJ

1. Uvod .....	1
2. Proces usmjeravanja u mreži s komutacijom paketa .....	3
2.1. Opći model telekomunikacijske mreže .....	3
2.2. Osnovne značajke prijenosa i komutacije paketa .....	5
2.2.1. Usmjeravanje u mreži s konekcijski orijentiranim načinom prijenosa .....	9
2.2.2. Usmjeravanje u mrežama s bezkonekcijski orijentiranim načinom prijenosa .....	10
2.3. Zaglavlje paketa .....	11
3. Uloga čvorova u procesu usmjeravanja s obzirom na njihov položaj u mreži .....	15
3.1. Tablice usmjeravanja .....	17
3.2. Mrežni protokoli .....	18
3.2.1. Usmjereni protokoli i njihove značajke .....	19
3.2.2. Protokoli usmjeravanja i njihove značajke .....	20
3.3. Vrste protokola usmjeravanja .....	21
4. Značajke algoritma usmjeravanja koji se temelje na modelu najkraćeg puta .....	24
4.1. Osnovne značajke <i>Open Shortest Path First</i> protokola .....	24
4.2. Dijkstra algoritam .....	28
4.3. Floyd-Warshallov algoritam .....	31
4.4. Bellman-Ford algoritam .....	33
5. Određivanje performansi mreže za različite algoritme usmjeravanja koji se temelje na najkraćem putu .....	38
5.1. Određivanje performansi mreže .....	40
5.1.1. Računanje najkraćeg puta pomoću Dijkstra algoritma .....	41
5.1.1.1. Kašnjenje na putu u neopterećenoj mreži .....	45
5.1.1.2. Kašnjenje na putu u opterećenoj mreži .....	48
5.1.1.3. Gubici u neopterećenoj i opterećenoj mreži .....	51
5.1.2. Računanje najkraćeg puta pomoću Floyd-Warshallovog algoritma .....	56

5.1.3. Računanje najkraćeg puta pomoću Bellman-Ford algoritma .....	59
6. Usporedba performansi mreže za različite algoritme usmjeravanja koji se temelje na najkraćem putu .....	65
7. Zaključak .....	68
Literatura .....	70
Popis kratica .....	74
Popis slika .....	75
Popis tablica .....	76



## 1. Uvod

Mreže s komutacijom paketa su sustavi koji se dinamički mijenjaju kako bi se prenijeli podatkovni paketi od izvorišta prema odredištu kroz mrežu koja je povezana pomoću komunikacijskih spojnica. Opterećenost komunikacijskih spojnica ovisi o količini paketa koji se trebaju prenijeti do odredišta, pa se stoga oni mogu mijenjati od slabo opterećenih spojnica do preopterećenih spojnica. Ovisno o kojoj opterećenosti se radi, tako će mreža na kraju i pokazati svoje rezultate dostave paketa na odredištu.

Kako bi se osigurao što lakši prijenos paketa kroz mrežu, koriste se razni protokoli usmjeravanja. Najčešće korišteni protokol usmjeravanja je onaj koji se temelji na najkraćem putu (engl. *Open Shortest Path First*). Glavni cilj ovoga protokola je osigurati optimalni i najkraći put kroz mrežu kako bi paketi stigli u što kraćem vremenu do odredišta.

Svaki protokol usmjeravanja ima svoje odgovarajuće algoritme prema kojima radi i koji algoritmu osiguravaju što bolje performanse mreže. Kada je u pitanju odabir odgovarajućeg algoritma kojim se pokušava riješiti problem najkraćeg puta u mreži, prigodan algoritam će se odabrati ovisno o zadanim parametrima mreže i samom vremenu izvođenja algoritma. U ovome radu su opisana tri najpoznatija i najviše korištena algoritma za pronalaženje najkraćih putova: Dijkstrin, Bellman-Fordov te Floyd-Warshallov algoritam. Rad navedenih algoritma će biti prikazan na primjeru manje računalne mreže.

Ovaj rad se sastoji od 7 poglavlja, a to su:

1. Uvod
2. Proces usmjeravanja u mreži s komutacijom paketa
3. Uloga čvorova u procesu usmjeravanja s obzirom na njihov položaj u mreži
4. Značajke algoritama usmjeravanja koji se temelje na modelu najkraćeg puta
5. Određivanje performansi mreže za različite algoritme usmjeravanja koji se temelje na najkraćem putu
6. Usporedba performansi mreže za različite algoritme usmjeravanja koji se temelje na najkraćem putu
7. Zaključak

Drugo poglavlje opisuje koncept rada mreže s gdje se informacije i podaci prenose kroz mrežu, od izvorišta do odredišta, pomoću paketa koji „putuju“ od čvora do čvora. U radu su opisane i glavne značajke prijenosa informacija pomoću komutacije paketa u mrežama.

Uloga čvorova u mreži u procesu prijenosa paketa od izvorišta do odredišta je prikazana u trećem poglavlju. Detaljno su analizirane funkcije svakog čvora, ovisno o njegovoj poziciji u mreži; izvorišni, posredni ili odredišni čvor. Svaki paket je okarakteriziran svojim zaglavljem u kojem su sadržani podaci o odredištu, tj. adresa odredišta. Svaki čvor čita to zaglavlje i usmjerava paket na temelju adrese. Prilikom usmjeravanja važnu ulogu imaju protokoli i algoritmi usmjeravanja, koji su navedeni i objašnjeni.

Četvrto poglavlje prikazuje i objašnjava pregled i ulogu algoritama usmjeravanja u mreži s komutacijom paketa. Podjela algoritama usmjeravanja prema odgovarajućim kriterijima, te najosnovnije odrednice rada svakoga od nabrojanih.

U petom poglavlju opisane su najvažnije mrežne performanse, njihova uloga prilikom biranja odgovarajućeg algoritma usmjeravanja. Nadalje, prikaz mreže koja se sastoji od odgovarajućeg broja čvorova i spojnica na kojoj se provodi sam postupak traženja najkraćeg puta pomoću odgovarajućih algoritama, te je nakon računanja najkraćeg puta prikazano računanje kašnjenja i vjerojatnosti blokiranja za prikazanu mrežu.

U posljednjem poglavlju, prikazana je usporedba navedenih algoritama, tj. njihove prednosti i nedostaci na temelju dobivenih rezultata. Sve karakteristike su navedene, i prema njima se može vidjeti koji algoritam je najbolji a koji najlošiji izbor u ovome slučaju.

## 2. Proces usmjeravanja u mreži s komutacijom paketa

Telekomunikacijska mreža, općenito, predstavlja skup međusobno povezanih tehničkih sustava i pripadajući im spojnice, i osnova su svakog telekomunikacijskog sustava. Prema [1] se navodi da je svrha telekomunikacijske mreže pružanje informacijskih i komunikacijskih usluga krajnjim korisnicima upotrebom jednog ili više oblika informacija, tj. jednog ili više medija: zvuk, govor, podatak ili tekst.

### 2.1. Opći model telekomunikacijske mreže

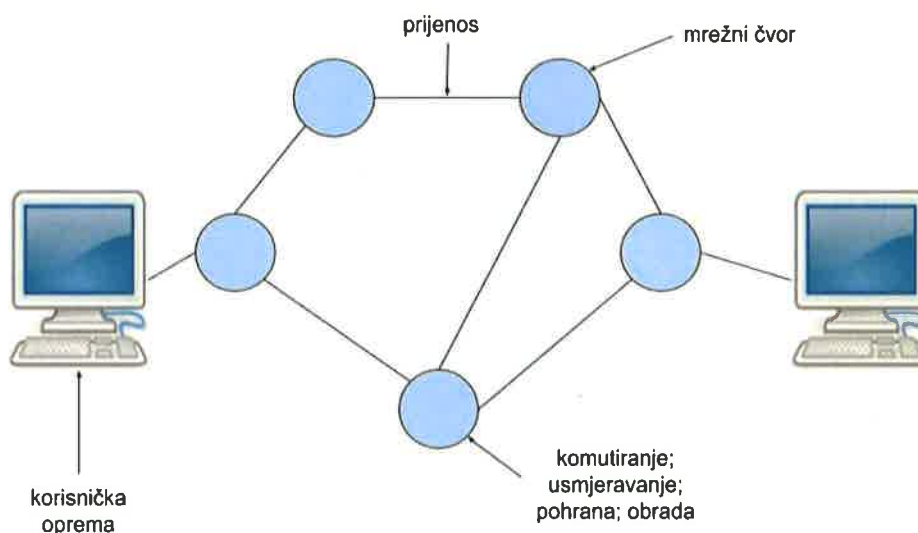
Komunikacijsku mrežu čine sustavi na koje se dalje spajaju računala, i sva druga potrebna oprema i uređaji. Unutar komunikacijske mreže se nalaze čvorovi, čija je glavna zadaća obavljanje funkcije komutiranja (engl. *switching*) kanala od ulaza prema izlazu i usmjeravanja (engl. *routing*) informacija prema drugim čvorovima. Čvorovi su povezani prijenosnim medijima preko kojih se prenose informacije, a prijenos informacija obavljaju prijenosni ili transmisijski sustavi, [2].

Osnovna zadaća svake telekomunikacijske mreže je omogućiti prijenos, tj. protok informacija između čvorova u toj mreži, kako bi one stigle do odgovarajućeg odredišta. Da bi svaka telekomunikacijska mreža mogla pružiti zadovoljavajuću kvalitetu krajnjim korisnicima, potrebno ostvariti i poštovati određene tehničke parametre kvalitete usluge (engl. *Quality of Service, QoS*), kao što su kašnjenje, raspoloživost, pouzdanost i sl. Postoje dva osnovna načina prijenosa informacija u telekomunikacijskoj mreži: kanalom i paketom. U skladu s tim, razlikuju se dva osnovna principa komutacije, [2]:

- komutacija kanala (engl. *circuit switching*)
- komutacija paketa (engl. *packet switching*).

Na slici 1 prikazan je izgled komunikacijske mreže, tj. od kojih se osnovnih dijelova ona sastoji i koja je uloga svakog od njih. Stoga komunikacijska mreža zahtijeva odgovarajuće kontrolne i upravljačke funkcije pomoću kojih se određuje način prijenosa informacija od čvora do čvora, i kako sve to ostvariti s obzirom na cjelokupan promet u mreži i trenutno stanje mreže, npr. opterećenost, smetnje, kvarovi i sl. Pri prijenosu informacije od izvorišta do odredišta obavlja se prijenos informacijskih tokova (proces transmisijske) i usmjeravanje informacijskih tokova prema odredištu (proces komutacije).

Strukturu komunikacijske mreže čine dva osnovna elementa: mrežni čvorovi i poveznice (engl. *links*). Mrežni čvorovi su međusobno povezani u jednu zajedničku mrežu pomoću poveznica koje ih povezuju i preko kojih se omogućuje prijenos informacija kroz mrežu. Čvorovi međusobno razmjenjuju informacije o smjerovima raspoloživima za prijenos informacija kroz mrežu i o raspoloživim prijenosnim kapacitetima na tim smjerovima. Na taj način svaki čvor može učinkovito komutirati primljene informacijske tokove, a ta funkcija koju obavljaju čvorovi se naziva određivanje smjerova prijenosa informacije ili usmjeravanje. Krajnji uređaji se povezuju s čvorovima pomoću prijenosnih poveznica, [3].



**Slika 1.** Prikaz komunikacijske mreže

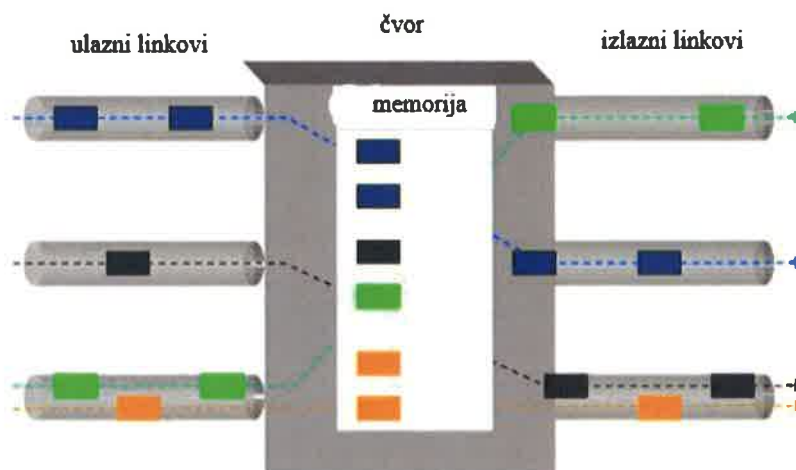
Izvor: [3]

Prema [1], proces komutacije je proces definiran kao uspostavljanje individualne veze, na zahtjev, od željenog ulaza do željenog izlaza dokle god to zahtjeva prijenos informacija. Pored ovih osnovnih zahtjeva komutacijski sustav mora zadovoljiti i zahtjeve u pogledu kvalitete ostvarene veze.

Pod funkcijama procesa komutacije se podrazumijevaju sljedeće dvije, a to su, [4]:

1. funkcija odlučivanja tj. procesiranja korisničkog zahtjeva – postupak tijekom kojega se obrađuje korisnički zahtjev, tj. što, kako i na koji način želi prenijeti putem mreže
2. funkcija prospajanja – postupak kojim se informacija korisnika prenosi s kraja na kraj mreže ili postupak kojim se informacija prenosi od predajnika do prijemnika.

S obzirom na funkcije koje se obavljaju u komutacijskom sustavu razlikuju se direktni komutacijski sustavi (koncentrirano prospajanje i odlučivanje) i indirektni komutacijski sustavi (samo prospajanje).



**Slika 2.** Poopćeni prikaz funkcije čvora

Izvor: [5]

Čvor, u komunikacijskog mreži, prima putem ulaznih spojnica dospjeli promet koji je sadržan u paketima, nakon čega se on privremeno pohranjuje u memoriji toga čvora radi obrade. Nakon obrade zaglavlja paketa, on se dalje šalje na izlazne spojnice čvora i dalje nastavlja svoj put kroz mrežu do idućeg čvora u kojemu će se ponoviti isti postupak.

## 2.2. Osnovne značajke prijenosa i komutacije paketa

U mrežama u kojima se informacije prenose pomoću paketa korisnička informacija se dijeli na blokove, te se svakom bloku dodjeljuju upravljačke informacije (engl. *Control part*). Korisnički dio (engl. *User part*) zajedno s upravljačkim dijelom čini paket koji samostalno putuje kroz mrežu od čvora do čvora. Upravljački dio sadrži skup kontrolnih informacija koje su potrebne usmjerivačima (ruteri, čvorovi) kako bi bili sposobni usmjeriti paket kroz mrežu i isporučiti ga željenom odredištu, [6].

Važnu ulogu u paketskim mrežama ima Internet protokol (IP), čije su osnovne funkcije adresiranje i usmjeravanje paketa kroz mrežu, tj. osigurava prienos jedinica podataka, datagrama<sup>1</sup> između računala i usmjeritelja, kao i između usmjeritelja. IP usmjeravanje podrazumijeva klasični način usmjeravanja u paketskim mrežama pomoću tablica

<sup>1</sup> **Datagram** – je paket promjenjive dužine (do 65535 bajta), koji se sastoji od zaglavlja i podataka.

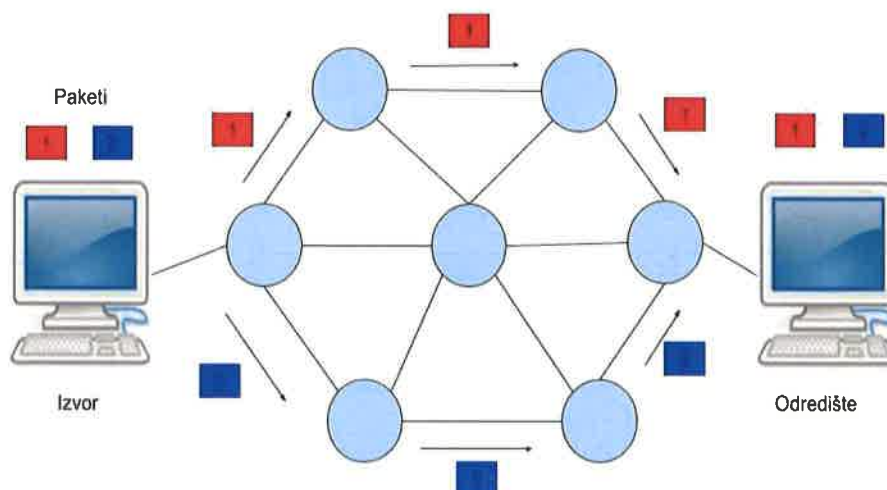
usmjeravanja. IP se isključivo brine o isporuci datagrama, tj. da datagrami uspješno stignu do svoga odredišta.

Svi paketi se šalju kroz mrežu po određenom rasporedu, ali ne nužno i istom rutom, što rezultira različitim vremenima kašnjenja jer različiti putovi imaju različita opterećenja. Radi toga se svi paketi numeriraju kako bi odredišni čvor znao provesti postupak depaketizacije.

Paketi mogu biti fiksne ili varijabilne duljine, što ovisi o tehnologiji koja se koristi za prijenos i komutaciju, kao i o sadržaju koji se prenosi. Ukoliko su paketi varijabilne duljine, definira se maksimalna duljina paketa pri komutaciji. Kod ovakvog načina prijenosa i komutacije informacija cijela pojasna širina je na raspolaganju za slanje svih paketa – ovisno o zahtijevanoj kvaliteti usluge, [6].

Na slici 3 prikazan je način na koji se informacije šalju od izvora do odredišta kroz mrežu komutacijom paketa. Paketi se u mreži usmjeravaju na temelju adresnih informacija u zaglavljju paketa ili oznake *logičkog kanala*. Uloga svakog čvora je da prihvati dolazeće pakete i privremeno ih smješta u svoju ulaznu memoriju gdje će se vršiti obrada toga paketa. U ulaznoj memoriji se provjerava odredišna adresa toga paketa, nakon čega trenutni čvor u kojem je smješten paket, odlučuje o sljedećem čvoru, tj. putu na koji poslati paket. Za vrijeme dok su ti paketi uskladišteni u ulaznoj memoriji, nad njima se vrši kontrola i provjera svih podataka, i u slučaju ukoliko postoje neke pogreške ili nepravilnosti, tada se izvršava korekcija pogrešaka.

Ukoliko postoji neko zagušenje u mreži, tada će paket ostati u ulaznoj memoriji čvora u kojem se trenutno nalazi, dok se zagušenje ne smanji ili u potpunosti nestane. Moguće zagušenje u mreži može nastati uslijed niza parametara, a neki od najčešćih zbog kojih se zagušenja pojavljuju su topologija mreže, strategija usmjeravanja, kapacitet mreže i spojnica, duljina paketa i sl.



**Slika 3.** Prijenos informacije kroz mrežu komutacijom paketa

Nakon toga se paketima analiziraju adresni i upravljački podaci kako bi ih se moglo svrstati na odgovarajuće redove radi daljnjeg prijenosa kroz mrežu do sljedećeg čvora. Pri tome se koristi postupak usmjeravanja. Kako bi se spriječilo da preveliko vrijeme isporuke paketa degradira kvalitetu usluge, u čvorovima paketske mreže se može implementirati da paketi budu posluživani po prioritetu, [7].

Ukoliko nema slobodnog puta na koji se paket može usmjeriti, tada se on zadržava u čvoru u kojem se trenutno nalazi dok se ne oslobodi neki put za daljnje usmjeravanje. Broj i veličina paketa se mogu prilagoditi količini informacije i na taj način omogućiti bolju iskorištenost komunikacijskih kapaciteta. U razdobljima kada jedan izvor ne odašilje pakete, oni se mogu prenositi iz nekih drugih izvora, što ukazuje na bolju učinkovitost ovakvog rada mreže, jer je u svakom trenutku mreža u stalnom radu, [2].

Ovakav način rada i slanja paketa od izvorišta do odredišta mreže posjeduje određene nedostatke:

- nepouzdana, bezkonekcijska isporuka (engl. *connectionless*) – ne podržava QoS
- analiza zaglavlja paketa koja se izvodi pri svakom skoku kroz mrežu
- zagušenje spojnica povećava gubitak paketa, duža kašnjenja i smanjenu propusnost
- ne koristi se dodatna metrika prilikom računanja puta, nego se dizajnira najkraći put.

Nedostaci IP usmjeravanja, posebice za sve zahtjevnije aplikacije, razlogom su razvoja novijih rješenja kao što je MPLS tehnologija koja bi trebala riješiti nedostatke i ponuditi unaprijeđeni način usmjeravanja. MPLS (engl. *Multiprotocol Label Switching*) ili višeprotokolarno komutiranje labela je tehnologija koja omogućava da se iz jednog zapisa u tablici usmjeravanja može odrediti nekoliko sljedećih točaka na samom putu do odredišta, tj. osigurava tradicionalni model prosljeđivanja paketa kroz mrežu, ali na mnogo efikasniji i brži način, nego što je to bio slučaj sa IP. MPLS koristi stoga postupak komutacije oznaka, tj. informacije koje se nalaze u zaglavlju paketa se analiziraju samo na jednom, ulaznom, rubnom usmjerivaču (engl. *Label Edge Router – LER*). Na taj bi se način uštedjelo na vremenu za procesuiranje zaglavlja paketa u svakom čvoru u mreži. Usmjeravanje u MPLS mrežama se dijeli na: usmjeravanje s upravljanjem iz izvorišnog čvora ili usmjeravanja s upravljanjem iz izvorišnog čvora uz prenošenje. Ovakav način predstavlja efikasnije i brže usmjeravanje u mrežama, [8].

**Tablica 1.** Usporedba najvažnijih parametara za IP i MPLS usmjeravanje

	Parametar	IP	MPLS
1	Trošak	Mali	Veći
2	Kašnjenje	Veće	Manje
3	Kompleksnost	Manja u odnosu na ATM	Manja u odnosu na ATM
4	Kvaliteta usluge	Manja razina	Veća razina
5	Broj <i>bytova</i> pri transferu	100-1500 <i>bytes</i>	Varijabilna dužina paketa

Izvor: [9]

Tablicom 1 prikazane su neke razlike između rada IP i MPLS prilikom usmjeravanja paketa u mreži. Iz tablice je vidljivo kako je MPLS tehnologija omogućila bolje performanse mreže uz mnoga poboljšanja koja su nastala kao rezultat negativnih stavki koje je imala IP tehnologija.

Prema [7], najvažnije funkcije koje obavljaju paketni komutacijski čvorovi, a sam postupak će biti detaljnije pojašnjen u poglavlju 3, su:

- pristupna koncentracija



- kontrola prometnog toka/zagušenja
- kontrola i korekcija pogrešaka
- tranzitno rutiranje paketa kroz mrežu prema odredištu
- klasifikacija prometnog toka i raspoređivanje kapaciteta odlaznih spojnica.

Postoje dva načina (modela) prijenosa paketa prema usmjeravanju paketa kroz mrežu, a to su:

- konekcijski orijentiran način prijenosa (engl. *Connection-oriented network*)
- bezkonekcijski orijentiran način prijenosa (engl. *Connectionless network*).

Konekcijski orijentiran i bezkonekcijski orijentiran način usmjeravanja su dvije različite komunikacijske tehnike u mrežama podataka koje imaju neke slične i zajedničke karakteristike, ali i karakteristike prema kojima se međusobno razlikuju, a detaljnije će biti obrađene u nastavku.

### 2.2.1. Usmjeravanje u mreži s konekcijski orijentiranim načinom prijenosa

Prema [10], konekcijski orijentiran način prijenosa paketa kroz mrežu radi na način da se uspostavlja virtualni kanal između proizvoljnih točaka u mreži. Ovaj način prijenosa paketa se odvija u tri faze: faza postavljanja (uspostava virtualnog kanala), prijenosa podataka i uništavanja (raskidanje virtualnog kanala).



**Slika 4.** Struktura paketa virtualnog kanala

Izvor: [11]

Kod načina prijenosa, koji se naziva virtualni kanal, paketi mrežom putuju po unaprijed određenoj i dogovorenoj stazi (virtualnim kanalom) prilikom kojeg izvor započinje proces slanja zahtjeva za slanje paketa kroz mrežu. Nakon što dobije odgovor na zahtjev, tada se započinje ili otkazuje slanje paketa, ovisno o odgovoru čvorova u mreži. Ukoliko se radi o pozitivnom odgovoru, tada će se razmjenjivati kontrolne informacije o tome kako samu vezu u

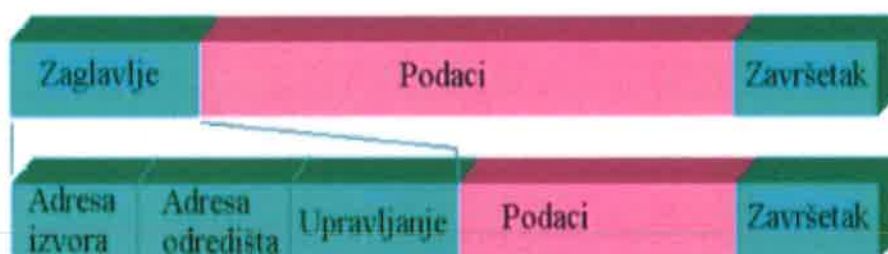
mreži treba uspostaviti i održavati. Na odredište paketi pristižu u polaznom redoslijedu, te ih je na taj način jednostavnije posložiti u cjelovitu poruku, [11].

Kvaliteta usluge (QoS), u mrežama u kojima se koristi konekcijski način prijenosa, orijentirana je na sami prijenos paketa kroz mrežu, tj. je li prijenos bez grešaka i može li se dodijeliti neka razina prioriteta određenim paketima s ciljem da će se paketi s većim prioritetom prenijeti prije onih koji imaju manji prioritet. Stoga će kašnjenja u ovakvom načinu prijenosa biti manja.

Paketi putuju mrežom koristeći isti put koji se uspostavlja na zahtjev korisnika (SVC) ili na dulje vrijeme (PVC). Kod SVC-a (engl. *Switched Virtual Circuit Services*) prijenos se odvija u tri faze: uspostavljanje, transmisija i raskidanje, dok kod PVC-a nema prve faze, tj. faze uspostavljanja, jer je konekcija uvijek raspoloživa korisniku, [7].

### 2.2.2. Usmjeravanje u mrežama s bezkonekcijski orijentiranim načinom prijenosa

Kod bezkonekcijski orijentiranog načina prijenosa, prema [7], sadržaj paketa se obično dijeli (fragmentiraju) na manje dijelove, tj. segmente. Svaki taj segment je određen adresom i numeriran je, kako bi se mogli razlikovati jedan od drugoga, i tako se mogu prenositi različitim putovima do odredišta neovisno jedan o drugome, bez ikakvog prethodnog obavještanja. Paketi se šalju mrežom, pri čemu svaki usmjerivač na putu prima taj paket i prosljeđuje ga najboljem sljedećem usmjerivaču, i tako sve dok paket ne stigne na odredište.



**Slika 5.** Struktura datagrama

Izvor: [11]

Paketi s podacima u mrežama koje koriste ovakav način prijenosa se jednostavno prosljeđuju do sljedećeg čvora bez ikakve prethodno uspostavljene veze. S ovakvim načinom prijenosa čvorovi su ti koji određuju sljedeći put za prosljeđivanje paketa.

Za usmjeravanje datagrama služe adrese i upravljački podaci, a na samom odredištu paketi se poslože u cjelovitu poruku. Ovakav način prijenosa i slanja paketa ne jamči siguran i pouzdan prijenos.

### **2.3. Zaglavlje paketa**

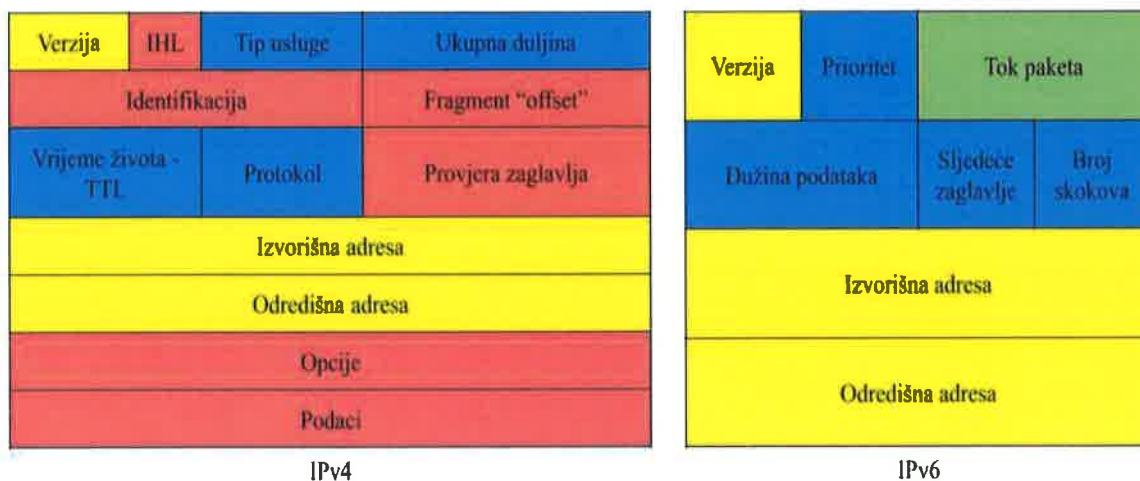
Kada se podaci prenose s jednog uređaja na drugi u računalnoj mreži, tada se ti podaci raščlanjuju na manje jedinice koji se zovu paketi. Ukoliko veličina podataka koja se želi smjestiti u jedan paket je veća od njegove maksimalne veličine, tada IP (engl. *Internet Protocol*) provodi fragmentiranje tih podataka. Svaki paket je određen svojim zaglavljem u kojem se nalaze osnovne informacije koje su važne prilikom prijenosa podataka kroz mrežu, kako bi oni uspješno stigli do odredišta. Sama analiza IP zaglavlja u svakom čvoru mrežnog sloja doprinosi kašnjenju i degradaciji same performanse mreže, čime je takav način usmjeravanja zamijenjen MPLS načinom usmjeravanja, kao što je to već prethodno prikazanom tablicom 1.

Danas egzistiraju dvije verzije zaglavlja paketa, verzija 4 (IPv4) i verzija 6 (IPv6). Verzija 4 je, s razvojem novih tehnologija, aplikacija, usluga i veličinom podataka, postala ograničavajuća zbog niza faktora, a najvažniji faktor je veličina adresnog prostora. Zbog toga se razvila unaprjeđenija i bolja verzija.

Osnovne informacije koje svako zaglavlje paketa sadrži su, [4]:

- IP adrese izvorišnih i odredišnih čvorova
- brojevi paketa koji pomažu pri ispravnom redoslijedu primanja paketa kad oni stignu na odredište
- ostale korisne tehničke informacije.

Osim gore navedenih osnovnih informacija koje zaglavlje paketa sadržava, također ovisno o kojoj se verziji radi, zaglavlje može sadržavati još niz dodatnih informacija. IPv4 zaglavlje se koristilo dugi niz godina, a danas se puno rjeđe koristi. Glavni nedostatak ove verzije je nedovoljno prostora za pohranu adresa. Zbog tog razloga se razvila verzija 6, koja omogućava niz prednosti, a glavna je veći adresni prostor. Prema slici 6 vidljive su razlike između ove dvije verzije zaglavlja IP paketa.



**Slika 6.** Usporedba zaglavlja IPv4 i IPv6

Izvor: [10]

Prva, i najvažnija, razlika između ove dvije verzije zaglavlja IP paketa je njihova veličina. Kod verzije 4, adrese su 32-bitni binarni brojevi, a kod verzije 6, radi se o puno većem adresnom prostoru, 128-bitni binarni brojevi. Zaglavlje verzije 6 je puno jednostavnije, tj. sastoji se od manje odjeljaka. Polja koja su ostala i ista su kod oba zaglavlja označena žutom bojom. Plava polja su zadržana, ali su kod verzije 6 promijenili poziciju i naziv. Zeleno polje kod verzije 6 uvedeno kao novo polje, kako bi se odredio slijed paketa određene usluge, dok kod verzije 4 ta funkcionalnost ne postoji, ali zbog toga zahtijeva puno veći broj provjera mrežnih priključaka i adresa.

Uvođenjem IPv6 protokola, mrežama je omogućen niz prednosti i jednostavnosti prilikom usmjeravanja paketa, a neke od najvažnijih su, [13]:

- novi i jednostavniji format zaglavlja
- veličina adresnog prostora (128 bita)
- poboljšana podrška za kvalitetu usluge (QoS – *Quality of Service*)
- proširivost
- ugrađeni sigurnosni mehanizmi
- pojednostavljeno usmjeravanje.

Kod IPv4 zaglavlja polje opcija je smješteno u osnovno IPv4 zaglavlje koje je varijabilne duljine, dok je kod IPv6 zaglavlja iz osnovnog zaglavlja su maknute sve opcije, a dodana su dodatna zaglavlja koja predstavljaju puno naprednije funkcije i nalaze se nakon osnovnog

zaglavlja. Na taj se način smanjilo vrijeme obrade u usmjernicima, jer se tako ne troši procesorsko vrijeme na obradu polja koja se odnose na njih.

Kod IPv6, zaglavlje počinje poljem *Verzija* i ono predstavlja verziju IP protokola, i za IPv6 u to polje mora biti upisan broj 6. Nakon toga slijedi oznaka klase prometa i oznaka toka kojem paket pripada. U to se polje upisuje nula, a u slučaju da je potrebna veća nosivost paketa, postoji proširenje unutar ovog zaglavlja.

Polje *Sljedeće zaglavlje* određuje protokol transportnog sloja, a dvije najčešće verzije protokola koja se koriste su TCP i UDP. Polje *Broj skokova* označava maksimalni broj skokova koje paket može napraviti u mreži. I na samom kraju zaglavlja su polja *Izvorišne* i *Odredišne adrese* koja su duga 128 bit-a svaka, [10].

Prilikom slanja paketa kroz mrežu, računala šalju paket na prvi slobodni i najbliži ruter. Taj paket je okarakteriziran svojim zaglavljem, koje pomaže usmjerivačima u mreži da ga prosljeđuju do odredišta. Kad središnji usmjerivači prime pakete, gledaju njegovo IP zaglavlje. Najvažnije polje njegovog zaglavlja je njegova odredišna IP adresa. Na slici 7 je prikazan primjer kako izgleda dio zaglavlja IP paketa, u kojem je sadržana njegova adresa (izvorišna i odredišna).

IP header	
Field	Content
Source IP Address	216.3.192.1
Destination IP Address	91.198.174.192
Version	4
Time to Live	64
... plus 10 more fields!	

**Slika 7.** Izgled zaglavlja IP paketa

Izvor: [4]

Središnji (jezgreni) usmjerivači prosljeđuju pakete među sobom na temelju tablice usmjeravanja, u kojoj se nalaze svi potrebni podaci da bi se paket mogao dalje poslati kroz mrežu. Kod IPv4, tablica nema redak za svaku moguću adresu, nego ima  $2^{32}$  mogućih kombinacija, tj. adresa koje može pohraniti. Umjesto toga, tablica ima retke za prefikse IP adrese. IP adrese su hijerarhijske, stoga, kad dvije IP adrese započinju sa istim prefiksom, to

znači da su u istoj mreži. Tablice usmjeravanja usmjerivača iskorištavaju tu činjenicu tako da mogu pohraniti daleko manje podataka.

Nakon što usmjerivač pronađe odgovarajući redak u tablici za određenu IP adresu, tada on taj paket šalje na taj put. I svaki idući središnji usmjerivač će ponoviti isti postupak sve dok taj paket ne dođe do odredišta, koje može biti osobno računalo ili poslužitelj.

<b>IP address prefix</b>	<b>path</b>
91.112	#1
91.198	#2
192.92	#3
...	

**Slika 8.** Prefiks IP adrese

Izvor: [4]

### 3. Uloga čvorova u procesu usmjeravanja s obzirom na njihov položaj u mreži

Telekomunikacijska mreža je skup dviju ili više međusobno povezanih mreža koje dijele neke zajedničke resurse (programe, podatke, sklopovlje i sl.). Mreže se smatraju povezanim ako mogu međusobno komunicirati, tj. razmjenjivati informacije. Taj proces joj omogućavaju mrežni čvorovi, koji su raspoređeni u mreži, a većinom imaju funkcije prospajanja ili usmjeravanja paketa u mreži.

Čvorovi predstavljaju entitete između kojih se nalaze veze ili *linkovi*. Stupanj čvora označava koliko se veza nalazi između čvorova. U telekomunikacijskim mrežama postoje tri osnovne grupe čvorova s obzirom na njihov položaj u mreži, a to su:

- izvorišni (početni) čvor: ima ulogu primanja i adresiranja paketa koristeći adresu koja će omogućiti slanje paketa kroz mrežu i nakon toga prosljeđuje taj paket slobodnom čvoru koji se nalazi u samoj mreži
- središnji čvorovi: imaju manje zadataka za odraditi u odnosu na početne i završne. Njihova najvažnija uloga je da primaju pristigle pakete i na osnovu adrese i prosljeđuju dalje do sljedećeg čvora u mreži, i tako sve dok ne dođu do odredišnog čvora
- odredišni (završni) čvor: prima paket i šalje obavijest o primanju.

Rubni ili *edge* čvorovi se nalaze u mrežnoj okosnici, a zadaća im je da prihvaćaju ulazni promet u mrežu. Rubni čvorovi se često još nazivaju i *host*-čvorovima (domaćinima), jer na njima rade raznovrsni serveri na koje se spajaju korisnici te tako postaju dio mreže. Služe da osiguravaju sigurnost mreže od mogućih napada i zaštite jezgru mreže osiguravanjem IP prometa od drugih rubnih usmjerivača kao i osnovnih usmjerivača. Oni također omogućuju povezivanje između glavne mreže i vanjske mreže, tj. za međusobno povezivanje mreža. Obično se koriste u mrežama širokog područja (WAN) ili na Internetu.

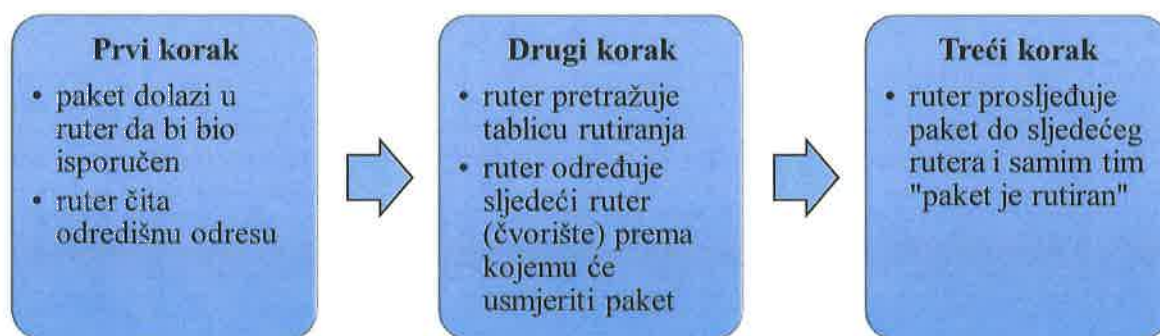
Jezgreni ili *core* čvorovi se nalaze u samom središtu mreže, tj. u unutrašnjosti, i upravljaju protokom paketa unutar mreže sa ostalim jezgrenim usmjerivačima. Uloga unutrašnjih čvorova može biti dodijeljena različitim mrežnim uređajima kao što su preklopnici (engl. *switches*), usmjerivači (engl. *routers*), mostovi (engl. *bridges*) i vrata (engl. *gateways*). Svaki od njih ima svoj položaj i posebnu ulogu u mreži.

Cilj usmjeravanja prometa u mreži je osigurati i omogućiti da tok podataka, koji putuje od izvorišta do odredišta, ima na raspolaganju potrebne kapacitete za prijenos i pri tome optimalno iskoristiti mrežu i osigurati zadovoljavajuću kvalitetu usluge. Tu funkciju ostvaruju usmjerivači, koji za svoje usmjeravanje koriste odgovarajuće usmjerivačke protokole, [12].

Ruter ili usmjerivač (engl. *router*) je tip prospojnika između vanjske mreže (javne mreže ili WAN) i LAN. Njegova funkcija je slična onoj koju obavlja komutacijski čvor, a neke od značajnijih funkcija je npr. „filtriranje prometa“, što zapravo znači da propušta pakete s važnijim podacima, dok pri tome blokira neke manje važnije ili velike datoteke.

Usmjerivači su aktivni mrežni uređaji koji rade na mrežnoj razini OSI referentnog modela, i koriste odredišnu IP adresu u podatkovnom paketu kako bi odredio gdje proslijediti paket.

Prema [7], uloge središnjih rutera u paketskoj komutacijskoj mreži su podijeljene u tri koraka, a prikazana su na slici 9.



**Slika 9.** Uloge središnjih usmjernika pri prosljeđivanju paketa

Izvor: [7]

Prema slici 9, kada paket dođe kod bilo kojeg središnjeg rutera, tada taj isti ruter ispituje odredišnu IP adresu zaglavlja paketa i uspoređuje ga s tablicom usmjeravanja kako bi utvrdio najbolji idući skok paketa.

Važnu ulogu prilikom usmjeravanja paketa između čvorova imaju usmjereni protokoli i protokoli usmjeravanja, koji se koriste kako bi se odabrao najbolji mogući put za usmjeravanje paketa do sljedećeg čvora. Odluke o usmjeravanju ne moraju biti unaprijed određene i nepromjenjive, a bilo da je riječ o usmjerenim protokolima ili protokolima usmjeravanja, oba se ona temelje i rade pomoću tablica usmjeravanja, [15].




### 3.1. Tablice usmjeravanja

Prema [15], tablice usmjeravanja predstavljaju baze podataka smještene na usmjerivačima unutar kojih su pohranjeni podaci o topologiji mreže, a koriste se prilikom prosljeđivanja podatkovnih paketa kroz mrežu.

Glavna zamisao i svrha postojanja tablica usmjeravanja je odabir najefikasnijeg puta po kojem će paketi putovati. Tablice usmjeravanja se koriste prilikom prosljeđivanja podatkovnih paketa tako što se adresa odredišta povezuje s mrežnim rutama koje vode do odredišta. Izgradnja i održavanje ovih tablica je osnovni zadatak protokola usmjeravanja.

Slika 10 prikazuje izgled tablice usmjeravanja, koja se sastoji od linija i stupaca. Svaki stupac definira posebni kriterij prema kojemu usmjerivač može izabrati najbolju rutu za usmjeravanje paketa.



```
Active Routes:
Network Destination    Netmask          Gateway          Interface        Metric
0.0.0.0                0.0.0.0          192.168.0.254   192.168.0.1      25
127.0.0.0              255.0.0.0        127.0.0.1       127.0.0.1        1
169.254.0.0            255.255.0.0      192.168.0.1     192.168.0.1      20
192.168.0.0            255.255.255.0    192.168.0.1     192.168.0.1      25
192.168.0.1            255.255.255.255  127.0.0.1       127.0.0.1        25
192.168.0.255         255.255.255.255  192.168.0.1     192.168.0.1      25
192.168.1.0            255.255.255.0    192.168.0.253   192.168.0.1      1
224.0.0.0              240.0.0.0        192.168.0.1     192.168.0.1      25
255.255.255.255       255.255.255.255  192.168.0.1     192.168.0.1      1
Default Gateway:      192.168.0.254
```

Slika 10. Izgled tablice usmjeravanja

Izvor: [15]

Prema [15], svaki stupac u tablici usmjeravanja ima drugu funkciju i značenje, a to su:

- *Network destination*. Mrežna sučelja, lokalne i udaljene podmreže koje usmjerivač zna su prikazani u ovom stupcu.
- *Netmask* (mrežna maska). Podaci maske podmreže usmjerivači koriste da bi odredili postoji li veza između paketa koji se trenutno obrađuje i bilo kojeg drugog predmeta u tablici usmjeravanja. Ako postoji veza, tada stupac *Interface* određuje na koji izlaz proslijediti paket.
- *Default gateway* (dodijeljeni pristupnik). Ukoliko ne postoji veza, koja se određivala u prethodnom koraku, tada se paket šalje na izlaz.
- *Metric*. Relativna vrijednost koja govori koliki je trošak rute.

Tablice usmjeravanja su pohranjene u memoriji usmjerivača, a zadatak izgradnje i održavanja tablica usmjeravanja imaju protokoli usmjeravanja. Osnovni problem u izgradnji tablica usmjeravanja je potreba za pohranjivanjem ruta prema velikom broju mrežnih odredišta unutar ograničenog memorijskog prostora. Rješenje je u pretpostavci da su slične adrese smještene blizu, tako da se jednim zapisom u tablici usmjeravanja omogućuje usmjeravanje prema više odredišta. Kako bi se smanjilo vrijeme koje svaki usmjerivač u mreži mora izdvojiti na čitanje tablice usmjeravanja, a na taj način i poboljšale same performanse samog usmjeravanja, koristi se tehnika koja je poznata kao višeprotokolarno komutiranje labela (engl. *Multiprotocol Label Switching* – MPLS).

Kod usmjeravanja je također vrlo važno izbjeći pojavu beskonačnih petlji, koje nastaju kada skupina usmjerivača neprekidno prosljeđuje isti paket međusobno ne prosljeđujući ga uopće prema njegovom odredištu. Na taj se način usporava rad mreže, zauzima prijenosni medij, dolazi do nepotrebnih gubitaka ostalih paketa.

Tablice usmjeravanja se dijele, također, i prema rutama usmjeravanja. S obzirom na to, imamo tri tipa, a to su: direktne rute, statičke rute i dinamičke rute.

Statičke rute u tablici usmjeravanja sadržavaju mrežnu adresu, masku podmreže i adresu sljedećeg usmjerivača u skoku. Statičke tablice namijenjene su za manje računalne mreže u kojima ne dolazi do nekih većih promjena u prometu. Dok dinamičke rute, tj. dinamičke tablice se koriste za one mreže koje imaju veći opseg i sklone su nekim većim promjenama u strukturi i rastu prometa. Zbog toga je potrebno dinamički prilagođavati stanje u tablicama sa stvarnim stanjem mreže, [14].

### **3.2. Mrežni protokoli**

Protokol je skup pravila koja određuju kako dva uređaja u mreži međusobno komuniciraju i definiraju format podatkovnih paketa koji se šalju komunikacijskim linijama.

Mrežni protokoli predstavljaju skup pravila po kojima komuniciraju računala u mreži. Oni definiraju red po kojemu se svi procesi u mreži odvijaju. Naziv „mrežni protokoli“ opisuje skup različitih protokola koji obavljaju različite funkcije, i na taj način povezuju različite aspekte komunikacijskog procesa.

### 3.2.1. Usmjereni protokoli i njihove značajke

Usmjerivački protokoli omogućuju mreži dinamičko prilagođavanje uvjetima. Usmjereni protokoli su oni koje je moguće usmjeravati i koji se koriste za prijenos različitih informacija računalnom mrežom.

Prema [12], postoje dvije osnovne vrste usmjerivačkih protokola, a koriste se za dinamičke rute. Dije se prema načinu računanja optimalnog puta, a to su:

- Protokoli koji se temelje na vektoru udaljenosti (engl. *Distance Vector*) određuju najbolji put na osnovu informacije koliko je udaljeno odredište paketa. Udaljenost može biti broj usmjernika do odredišta, a može biti i kombinacija nekih vrijednosti koje će definirati tu udaljenost. Susjednim usmjernicima se šalje cijela tablica usmjeravanja. Najpoznatiji protokol vektora udaljenosti je RIP (*Routing Information Protocol*) protokol, a koristi Bellman-Fordov algoritam.
- Protokoli koji se temelje na stanju veze (engl. *Link State*) rade na način da svaki usmjernik zna topologiju mreže i ne šalje se cijela tablica usmjeravanja, nego se svim usmjernicima u mreži šalje samo informacija o stanju veze u obliku malih LSA (*Link State Advertisement*) paketa. Na osnovu dobivenih informacija usmjerenici ponovo izračunaju puteve. Najpoznatiji predstavnik ove skupine protokola je OSPF (*Open Shortest Path First*) protokol, a koristi Dijkstrin algoritam.

Usmjerivački protokoli se također dijele na unutarnje i vanjske. Unutarnji se koriste za usmjeravanje paketa unutar mreže, a vanjski se koriste za usmjeravanje paketa između mreža.

Neki od najpoznatijih usmjerenih protokola, prema [14], su:

- IP (engl. *Internet Protocol*) – najpoznatiji usmjereni protokol pomoću kojeg se odvija podatkovna komunikacija između izvorišnog i odredišnog računala
- OSI mrežni protokoli
- XNS (engl. *Xerox Network System*)
- Apple Talk
- DECnet (engl. *Digital Equipment Corporation*).

### 3.2.2. Protokoli usmjeravanja i njihove značajke

Protokoli usmjeravanja omogućuju usmjerivačima da dinamički dijele informacije o udaljenim mrežama i automatski dodaju informacije u svoje tablice usmjeravanja. Oni služe za određivanje optimalnog puta do neke mreže koji se dodaje u tablicu. Protokoli usmjeravanja su u osnovi skup procesa, poruka i algoritama koji se koriste u razmjeni informacija među usmjerivačima i u popunjavanju tablice usmjeravanja.

Glavne značajke protokola usmjeravanja su:

- otkrivanje udaljenih mreža
- održavanje informacija
- određivanje najboljeg puta do odredišta
- mogućnost pronalaska alternativnih puteva ukoliko trenutni nije više dostupan ili nije više najbolji izbor za usmjeravanje.

Prema [14], neki od najpoznatijih protokola usmjeravanja su:

- OSPF (engl. *Open Shortest Path First*)
- RIP (engl. *Routing Information Protocol*)
- IS-IS (engl. *Intermediate System to Intermediate System*)
- BGP (engl. *Border Gateway Protocol*)
- IGRP (engl. *Interior Gateway Routing Protocol*)
- EIGRP (engl. *Enhanced IGRP*).

Svaki od ovih nabrojanih protokola se upotrebljavaju u mreži prema njihovim karakteristikama, tj. ovisno o načinu na koji rade. Svaki protokol ima neke specifične i zasebne karakteristike prema kojemu radi, tj. usmjerava promet u mreži, a također imaju i poželjne karakteristike poput točnosti, stabilnosti, efikasnosti i robusnosti.

U ovom radu biti će detaljnije obrađen samo OSPF protokol, gdje će biti navedene njegove osnovne značajke, način rada, prednosti, nedostaci, problemi i algoritmi koji se koriste za rješavanje tih problema.

Određivanje najboljeg puta do odredišta podrazumijeva odabir puta na temelju niza metrika, tj. karakteristika. Ovisno o primjeni za koju se svaki od ovih protokola usmjeravanja razvijen i namijenjen, karakteristike prema kojima se oni razlikuju su sljedeće, [14]:

- Optimalnost je sposobnost usmjerivača da odabere najbolju rutu u određenom trenutku, a najbolja ruta ovisi o korištenoj metrici.
- Jednostavnost je važna karakteristika protokola usmjeravanja jer oni moraju svoju zadaću, tj. funkciju obavljati na što efikasniji i jednostavniji način, i pri tome zauzimajući što manje resursa usmjerivača.
- Robusnost i stabilnost su sposobnost da protokoli usmjeravanja trebaju ispravno funkcionirati u neobičnim i nepredviđenim okolnostima, i pri tome moraju ostati stabilni.
- Brza konvergencija je postupak u kojem se svi usmjerivači slože oko najbolje rute, u suprotnom se može pojaviti spora konvergencija koja uzrokuje stvaranje beskonačnih petlji i česte ispade mreže.
- Prilagodljivost je sposobnost da se protokoli usmjeravanja trebaju što brže i pravilnije prilagođavati različitim pojavama i događajima u mreži.

### 3.3. Vrste protokola usmjeravanja

Svi protokoli usmjeravanja rade na poseban način, i svi imaju svoje prednosti i nedostatke prilikom usmjeravanja paketa kroz mrežu. Prilikom samog usmjeravanja, kod svih protokola, se pojavljuju s vremena na vrijeme neke poteškoće ili neispravnosti, zbog čega dolazi do usporenog rada mreže, gubitka ostalih paketa i sl. U takvim trenucima se koriste algoritmi usmjeravanja, koji zapravo služe za rješavanje problema koja se povezuju za pojedine protokole. Svaki protokol ima svoj algoritam pomoću kojega rješava probleme ili poteškoće u radu.

Prema [16], algoritam usmjeravanja je skup postupaka koji se koriste prilikom usmjeravanja internetskog prometa kroz mrežu. Algoritmi usmjeravanja se koriste kako bi se matematički odredio najbolji put do odredišta. Različiti algoritmi usmjeravanja koriste različite metode za određivanje najboljeg puta.

Prema tipu usmjeravanja, algoritmi se dijele na:

- statičke i dinamičke
- algoritme s jednom ili više ruta
- jednorazinske i hijerarhijske
- izvorišno usmjeravanje i usmjeravanje usmjerivačima
- unutar domene i među domenama

- *link state i distance vector.*

Statički i dinamički algoritmi usmjeravanja su već prethodno spomenuti. Ukratko, statički su namijenjeni za manje mreže koje imaju jednostavnije topologije s predvidljivim prometom, i tablice usmjeravanja ispunjava i ažurira mrežni administrator<sup>2</sup>. A dinamički algoritmi su namijenjeni mrežama koje su izložene čestim i nepredvidivim promjenama, i tim promjenama se moraju prilagoditi svi usmjerivači u mreži.

Algoritmi usmjeravanja s jednom rutom rade po principu da uz pomoć tablice usmjeravanja pronađu jednu optimalnu rutu do odredišta kojom će se podaci slati. Kod ovog načina usmjeravanja se ne iskorištavaju svi mrežni resursi, i vrlo je neučinkovit ukoliko se pojave neke neočekivane promjene ili zagušenja. Dok algoritmi usmjeravanja s više ruta pronalaze više optimalnih ruta za slanje podataka kroz mrežu, i na taj se način iskorištava više resursa mreže.

Kod jednorazinskog algoritma usmjeravanja svi usmjerivači su jednake važnosti i dio su iste mreže. Svi usmjerivači sadrže informacije o svakom usmjerivaču iz te iste mreže, a u slučaju nekih promjena, potrebno je izmijeniti tablice usmjeravanja svakog čvora u mreži. Hijerarhijsko usmjeravanje radi na način da je uvedena organizacija u kojoj su usmjerivači grupirani po zasebnim organizacijskim grupacijama.

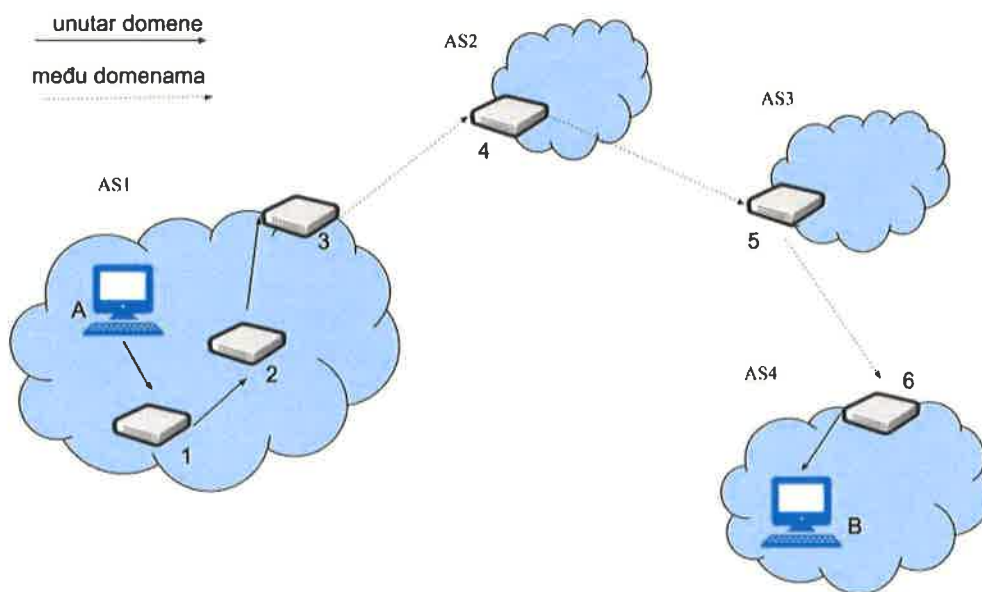
Izvorišno usmjeravanje je usmjeravanje u kojem se put kroz mrežu kojom će se poslati podaci određuje na izvorišnom čvoru. Pri tome tada izvorišni usmjerivač mora poznavati topologiju cijele mreže kako bi mogao odrediti optimalni put po kojem poslati podatke. Kod slučaja usmjeravanja na svim usmjerivačima, svi usmjerivači u mreži sami odlučuju na koji put prosljediti pakete, tj. odrediti sljedeći skok paketa.

Algoritmi usmjeravanja unutar domene i između domena su dio autonomnih sustava. Algoritmi unutar domene provode usmjeravanje unutar jedne domene između mrežnih elemenata koji su dio toga jednog autonomnog sustava. Najpoznatiji protokoli koji spadaju u ovu skupinu su *Open Shortest Path First Protocol* i *Routing Information Protocol*. Algoritmi usmjeravanja među domenama razmjenjuju promet između više domena, gdje ta svaka domena sama odlučuje kojim putem će usmjeriti promet prema sljedećoj domeni, [17].

---

<sup>2</sup> **Mrežni administrator** – osoba koja je zadužena za nadzor i održavanje informacijsko-komunikacijske infrastrukture neke organizacije ili tvrtke.

Na slici 11 je prikazan način na koji rade ova dva algoritma, tj. na koji način usmjeravaju promet između domena ili unutar pojedine domene.



**Slika 11.** Usmjeravanje unutar i među domenama

Izvor: [17]

*Link state* algoritmi šalju podatke koji su potrebni za usmjeravanje svim čvorovima u mreži. Rade na način da šalju male ažurirajuće poruke na mnogo adresa. Imaju brzu konvergenciju i zbog toga su otporniji na pojave petlji, koriste se uglavnom unutar podmreža. *Distance vector* algoritmi šalju podatke o usmjeravanju tako da svaki usmjerivač cijelu svoju tablicu usmjeravanja dijeli sa susjednim usmjerivačima. *Distance vector* protokoli najčešće usmjeravaju promet među mrežama i preko Interneta, [14].

Protokoli usmjeravanja se imaju svrhu korištenja u mrežama za lakši način usmjeravanje informacija u mreži između izvorišta i odredišta, od jednog čvora do ostalih čvorova u mreži. Osim te funkcije, važno pitanje koje se još povezuje za protokole usmjeravanja se naziva informacijskim *push* („guranjem“) i informacijskim *pull* („povlačenjem“). Pod *push* se zapravo smatra i odnosi na pojedini čvor koji „gura“ informacije prema drugim čvorovima (većinom se to izvodi periodično), dok se pod *pull* odnosi na čvor koji kontaktira/zahtijeva druge čvorove za dobivanje informacija koje su mu potrebe za računanje ruta i slične mrežne kontrole. Neki protokoli usmjeravanja koriste *pull* način, a jedni *push* način, a također neki koriste kombinaciju *pull-push* rada.

## 4. Značajke algoritma usmjeravanja koji se temelje na modelu najkraćeg puta

Algoritmi za usmjeravanje su dio softvera mrežnog sloja i odgovorni su za donošenje odluke po kojoj će se paketi usmjeravati do odredišta. Kada paket putuje mrežom, postoji veći broj putova kojima se taj paket može usmjeriti. Paketi se usmjeravaju pomoću protokola usmjeravanja, a prilikom toga usmjeravanja kroz mrežu, paketi često nailaze na razne poteškoće i probleme. U tu svrhu se koriste algoritmi, koji omogućavaju rješavanje tih problema.

Postoji veći broj protokola usmjeravanja, kao što su već prethodno u radu bili spomenuti, i svaki od tih protokola ima svoje posebne algoritme. Svaki algoritam ima svoje drugačije karakteristike i značajke rada, a u ovome radu je naglasak na *Open Shortest Path First* usmjerivačkom protokolu, pa će se samo obraditi oni algoritmi koji su povezani sa ovim protokolom.

### 4.1. Osnovne značajke *Open Shortest Path First* protokola

OSPF (*Open Shortest Path First*) je usmjerivački hijerarhijski protokol stanja veze kojeg je ustanovio IETF<sup>3</sup> (engl. *Internet Engineering Task Force*). Svrstava se u IGP (engl. *Internet Engineering Task Force*) vrstu protokola, tj. protokola usmjeravanja koji se koriste unutar autonomnih sustava. To je najčešći IGP protokol kod velikih računalnih mreža. Koristi Dijkstra SPF algoritam za pronalaženje najkraćeg puta.

OSPF je protokol koji zahtjeva slanje obavijesti o stanju veze ostalim usmjernicima unutar istog hijerarhijskog prostora. Računanje najkraćeg puta na grafu koji zapravo predstavlja mrežu sadržava: uređaj (vrh), veza (rub), težina ruba (mjerni podaci).

Prema konačnom čvoru i karakteristikama puta, problem najkraćeg puta može se podijeliti u nekoliko skupina: najkraći put između dva čvora, najkraći put među svim čvorovima,  $K$  najkraćih putova, najkraći put u stvarnom vremenu, i najkraći put u mreži. Većinom se računaju najkraći putovi druge skupine, tj. među svim čvorovima u mreži, [18].

Svojstva OSPF protokola:

- sve staze (spojnice) do odredišta čine stablo

---

<sup>3</sup> IETF (engl. *Internet Engineering Task Force*) – organizacija koje specificira službene standarde TCP/IP mreža.



- jednostavan i jedinstven algoritam.

Metrika OSPF-a se računa prema sljedećoj formuli, [18]:

$$C = \frac{10^8}{\text{Kapacitet veze} \left(\frac{\text{bit}}{\text{s}}\right)} \quad (1)$$

Prema izrazu (1) vidljivo je da je cijena puta ( $C$ ) obrnuto proporcionalna kapacitetu neke veze. Iz toga je npr. vidljivo da veza sa 100 Mb/s ima veću cijenu nego veza od 1 Gb/s, a paket će biti usmjeren na put sa manjom cijenom.

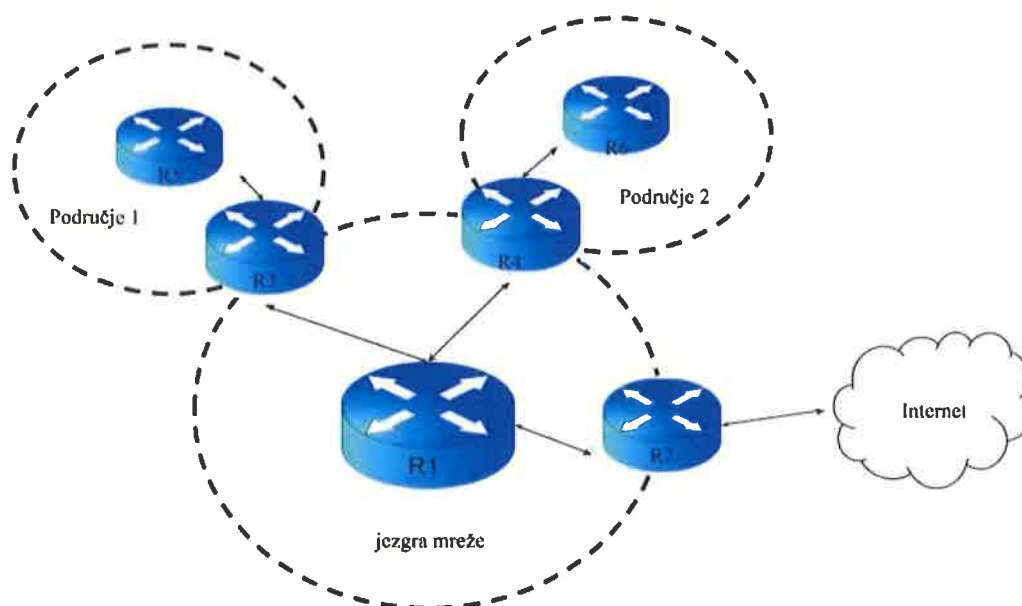
OSPF je protokol koji dobro funkcionira u mrežama srednjeg i velikog kapaciteta, minimalno opterećuje mrežu, omogućava praktički neograničen rast mreže.

Kao što je prethodno već spomenuto, OSPF protokol je hijerarhijski protokol, tj. područja su raspoređena u hijerarhijske razine, i na taj način se postiže dodatna razina sigurnosti usmjeravanja. Poruke koje razmjenjuju usmjerivači unutar mreže, moraju biti u takvom obliku da ih mogu razumjeti samo oni usmjerivači kojima su te poruke namijenjene.

Kod svih računalnih mreža bitno je da se ona podijeli na neke manje logičke cjeline, radi bolje razumijevanja.

Prema [18], kod OSPF postoje tri osnovna područja koja su kategorizirana prema funkcijama koje imaju usmjerivači, a to su:

- Jezgra mreže koja povezuje sva područja u jednu cjelinu, direktnim vezama ili vezama preko drugih usmjerivača.
- *Stub* područje koje sadržava usmjerivače koji su zaduženi za primanje informacija o topologiji mreže samo onih autonomnih sustava u kojima se oni nalaze.
- Tranzitno područje koje može se sastojati od dva ili više granična usmjerivača koji imaju zadatak propuštanja prometa podataka iz jednog područja u drugo.



**Slika 12.** OSPF područja mreže

Izvor: [18]

Granični usmjerenici (engl. *Area Border Routers*), na slici su to R2, R3 i R4 usmjerenici, održavaju topološku bazu za svako područje. Topološka baza sadrži skup LSA-ova (engl. *Localized Service Area*) svih usmjernika u istom području. Ako su usmjerenici unutar istog područja onda imaju jednake topološke baze. Razdvajanje područja stvara dva različita tipa OSPF usmjeravanja, ovisno o tome jesu li izvorište i odredište u istim ili različitim područjima.

Jezgra mreže ili područje okosnice (engl. *Backbone Area*) služi za distribuiranje usmjerivačkih informacija između područja, tj. sav promet koji povezuje neka druga područja prolazi preko njega. Svaki usmjerenik unutar ovoga područja mora poznavati topologiju cijele mreže.

Ukoliko se pojavi slučaj postojanja većeg broja usmjerenika u nekom području, tada se mora pronaći način kako optimalno razmijeniti podatke između njih. Taj problem se rješava pomoću postavljanja jednog glavnog usmjerenika (engl. *Designated Router*) i njegovog pomoćnog glavnog usmjerenika (engl. *Backup Designated Router*) za svako OSPF područje. svaki usmjernik na tom području uspostavlja vezu samo prema DR-u i BDR-u, a oni preplavljaju (engl. *flooding*) mrežu podacima i šalju informacije svim ostalim usmjernicima, [18].

8	16	32 bit
VERZIJA	TIP PAKETA	DUŽINA PAKETA
ID USMJERNIKA		
ID PODRUČJA		
PROVJERA		TIP AUTENTIKACIJE
AUTENTIKACIJA (64 bita)		

**Slika 13.** Zaglavlje OSPF okvira

Izvor: [18]

Prema [18], format OSPF paketa prikazan je na slici 13. Sukladno slici, OSPF paket se sastoji od devet dijelova, tj. devet polja. Svako polje ima svoje posebno značenje i funkciju.

Značenja polja su:

- *Version number* – verzija OSPF paketa koji se koristi (verzija 1, 2 ili 3)
- *Packet type* – tip OSPF paketa. Postoji 5 tipova paketa, a to su:
  - a) „Hello“ paketi – utvrđuju susjedne veze
  - b) Paketi za opis baze – opisuju sadržaj topološke baze podataka
  - c) Paketi za zahtjev stanja veze – zahtjeva dijelove topološke baze podataka od susjednih rutera
  - d) Paketi za osvježavanje stanja veze – odgovara na paket *link state* zahtjeva
  - e) Paketi za potvrdu stanja veze – potvrđuje *link state* dopunjene pakete
- *Packet length* – duljina paketa u oktetima
- *Router ID* – identifikator usmjerenika koji je izvor paketa
- *Area ID* – identifikator područja kojemu paket pripada
- *Checksum* – kontrolni broj zaglavlja, ne uključuje 64-bitno autentikacijsko polje
- *AuType* – autentikacijska shema koja se koristi
- *Authentication* – 64-bitno autentikacijsko polje.

OSPF protokol usmjeravanja ne prenosi podatke koristeći transportne protokole, nego on samostalno oblikuje IP pakete koristeći sljedeće poruke, [19]:

- *Hello* paketi - uspostavljaju i održavaju veze sa susjednim OSPF usmjerivačima
- *Database Description* - sadržava listu zapisa koji se provjeravaju s glavnom bazom

- *Link State Request* – prijemni usmjerivači koriste ovaj paket za izvršavanje upita nad bazom
- *Link State Update* – koristi se kao odgovor na LSR i ažuriranje novim informacijama
- *Link State Acknowledgement* – potvrda da je LSU paket primljen.

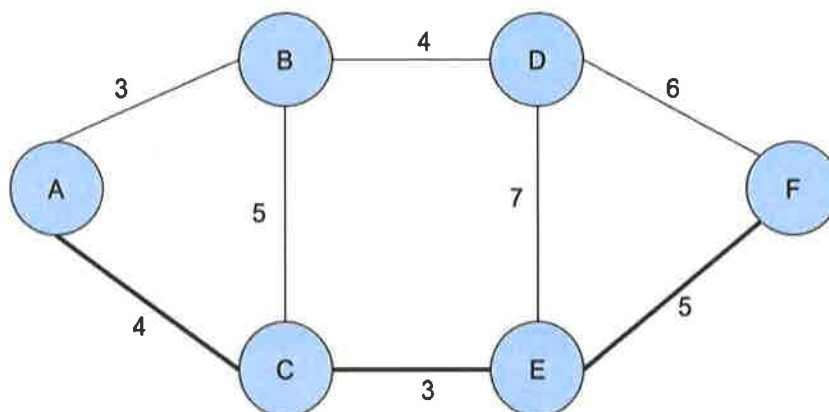
## 4.2. Dijkstra algoritam

Dijkstra algoritam je dobio ime prema znanstveniku Edgeru W. Dijkstri, koji ga je i dizajnirao i po tome je algoritam dobio ime. Ovaj algoritam se najčešće upotrebljava kod OSPF protokola, u svrhu pronalaženja najkraćeg puta od izvorišta do odredišta u nekoj mreži.

Sljedeće točke trebaju se dobro shvatiti ukoliko se koristi Dijkstra algoritam za pronalaženje najkraćeg puta, [20]:

- Dijkstra algoritam radi samo na povezanim grafovima
- Dijkstra algoritam radi samo na onim grafovima koji nemaju negativne čvorove
- stvarni Dijkstra algoritam ne daje najkraće staze
- pruža samo vrijednost ili cijenu najkraćih staza
- unošenjem manjih izmjena u stvarni algoritam mogu se dobiti najkraći putovi
- Dijkstra algoritam radi za usmjerene kao i za neusmjerene grafove.

Računanje najkraćeg puta pomoću Dijkstra algoritma nije teško za neke jednostavnije grafove, tj. mreže, no ukoliko se radi o složenijim mrežama koje se sastoje i većeg broja čvorova u mreži, tada zapravo ovaj algoritam i nije preporučljivo koristiti, posebice ukoliko se najkraći put traži „ručno“. Na slici 14 je primjer jednostavnijeg grafa sa 6 čvorova, uključujući i izvorišni i odredišni, i na njemu je objašnjen princip računanja najkraćeg i najboljeg puta do odredišta.

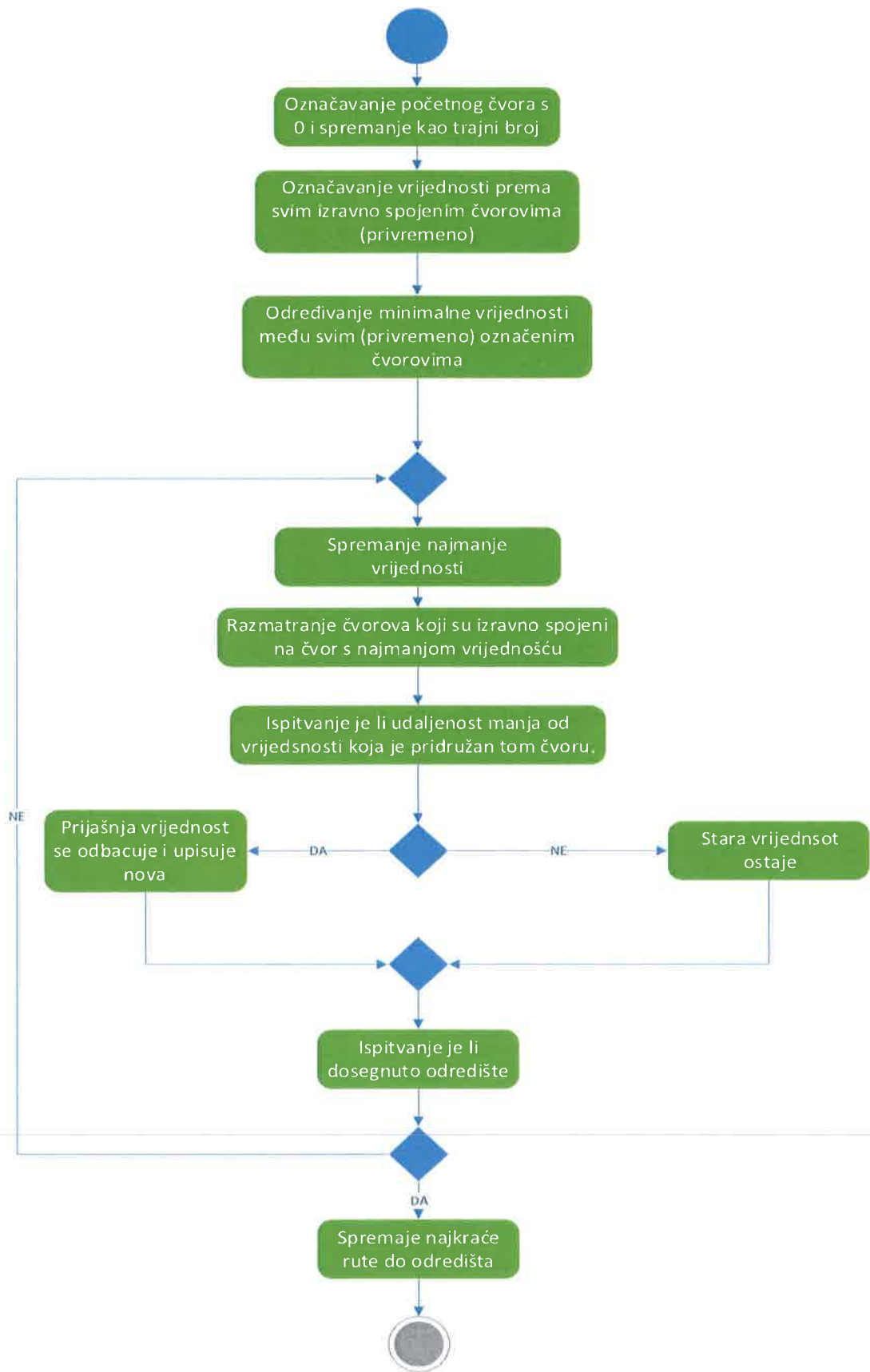


**Slika 14.** Računanje najkraćeg puta korištenjem Dijkstrinog algoritma

Prema slici 14, graf se sastoji od 6 čvorova, i svaki čvor je povezan sa susjednim, i ta veza je okarakterizirana vrijednošću koja zapravo predstavlja udaljenost do idućeg čvora. Prilikom računanja najkraćeg puta, algoritam će svakom čvoru dodijeliti njegovu težinu zbrajanjem vrijednosti odabranih spojnice. Prema ovom primjeru, izvorišni čvor je čvor A, a odredišni je čvor F. Izvorišnom čvoru je dodijeljena vrijednost 0, pošto je on početni čvor. Nakon što krene usmjeravanje, algoritam odvija iteracije prilikom kojih se u svakom sljedećem čvoru zbraja vrijednost koja je bila u prethodnom čvoru sa vrijednosti koja se nalazi između ta dva čvora. Sa primjera sa slike, prva iteracija bi bila u čvoru C, kojem bi se dala vrijednost 4, u čvoru B vrijednost 3. I na svakom sljedećem čvoru bi se ove vrijednosti zbrajale sa vrijednostima koje se nalaze na spojnim vezama između ta dva čvora. Prema ovom primjeru i temelju računanja najkraćeg puta pomoću Dijkstra algoritma, najkraći put je između A-C-E-F čvorova.

Nedostatak Dijkstra algoritma je što on pogrešno rukuje s negativnim vrijednostima udaljenosti čvorišta.

Na slici 15 prikazan je dijagram aktivnosti Dijkstrinog algoritma usmjeravanja, u kojem je pojašnjen proces traženja najkraćeg puta u mreži.



Slika 15. Dijagram aktivnosti Dijkstrinog algoritma usmjeravanja

Izvor: [21]

### 4.3. Floyd-Warshallov algoritam

Prema [22], Floyd-Warshallov algoritam rješava problem najkraćeg puta svih parova čvorova na usmjerenom grafu  $G=(V, E)$ . Grane grafa mogu poprimiti negativne vrijednosti, no vrijedi pretpostavka kao i kod Bellman-Fordovog algoritma, da na grafu ne postoje negativni ciklusi. Ovaj algoritam koristi dinamički pristup programiranju u svrhu pronalaženja najkraće rute do odredišta.

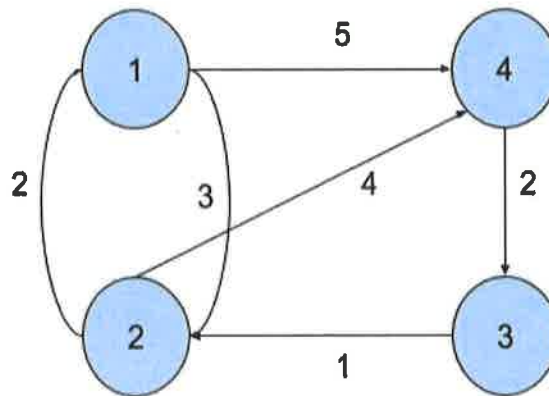
Floyd-Warshallov algoritam se može koristiti za rješavanje sljedećih problema, [22]:

- najkraći put u usmjerenim grafovima (Floydov algoritam)
- testiranje je li neizravni graf bipartitan
- brzo računanje *Pathfinder* mreža
- najširi putevi / putevi maksimalne propusnosti.

Floyd-Warshall algoritam najbolje odgovara i najviše se koristi kod gustih grafova, zato što njegova složenost ovisi samo o broju vrhova u danom grafu. Glavna prednost Floyd-Warshallovog algoritma je njegova jednostavnost.

Ovaj algoritam radi po principu da pronalazi najkraći put među svim parovima čvorova usmjerenog težinskog grafa. Rezultat Floyd-Warshallovog algoritma je matrica u kojoj se upisuje vrijednost udaljenosti između tih čvorova dimenzija  $m \times m$ , gdje  $m$  predstavlja broj čvorova. Ovaj algoritam ima vremensku kompleksnost  $O$  vrijednosti  $V^3$ . Dijagonala matrice uvijek mora biti popunjena sa vrijednosti nula, a ako među čvorovima ne postoji nikakva izravna veza tada se u matricu upisuje pozitivna beskonačnost.

Na slici 16 nalazi se primjer grafa kojim je objašnjen način traženja najkraćeg puta pomoću Floyd-Warshallovog algoritma.



**Slika 16.** Graf nad kojim je izvršen Floyd-Warshall algoritam

Da bi se mogao izračunati najkraći put od jednog čvora do drugoga, potrebno je kreirati već spomenutu matricu. U matrici će vrijednost  $m$  biti jednaka 4, pošto se graf sastoji od 4 čvora (vrha). Redak i stupac matrice su označeni kao  $i$  i  $j$ , koji su vrhovi grafa. Prema tablici 2 vidljivo je da su vrijednosti na njenoj dijagonali jednake nuli. To proizlazi iz zaključka da je najkraća udaljenost od izvorišnog do odredišnog čvora, ukoliko se radi o istom čvoru, jednaka nuli. Nakon toga se napravi krug po grafu i upišu se odgovarajuće vrijednosti u tablicu koje se računaju prema formuli (2).

**Tablica 2.** Finalna tablica Floyd-Warshalla algoritma

	1	2	3	4
1	0	3	7	5
2	2	0	6	4
3	3	1	0	5
4	5	3	2	0

Plavi stupac tablice predstavlja izvorišne čvorove, dok plavi redak predstavlja odredišne čvorove prema kojima se računa udaljenost.



Svakom sljedećom iteracijom po grafu prolazi se kroz svaki čvor te se mijenjaju vrijednosti u matrici na temelju sljedeće formule, [22]:

$$dist [i][j] > dist [i][k] + dist [k][j] \quad (2)$$

$i, j$  – čvorovi čija se udaljenost prikazuje

$i$  – stupac u matrici (izvorišni čvor)

$j$  – redak u matrici (odredišni čvor)

$k$  - oznaka definiranog tranzitnog čvora

Posljednja iteracija grafa sa slike 16 prikazana je u tablici 2, a prikazuje najkraće vrijednosti puteva čvorova.

Ovaj algoritam ima vremensku kompleksnost  $O$  vrijednosti  $V^3$ . Rezultantna matrica prikazuje vrijednosti najkraćih puteva  $V^3$  međurješenja. To označava maksimalno vrijeme koje bi ovim algoritmom bilo potrebno da se izračunaju vrijednosti najkraćih puteva u matrici. No, ukoliko ažuriranje vrijednosti najkraćih putova nije potrebno u svakom međukoraku, to će smanjiti vrijeme potrebno za izvršavanje algoritma, [22].

#### 4.4. Bellman-Ford algoritam

Bellman-Ford algoritam po svojoj izvedbi i načinu rada je jako sličan Dijkstrinom algoritmu. Razlikuje se od Dijkstrinog algoritma po tome što služi za rješavanje problema najkraćeg puta jednog izvora u slučaju kada težinske vrijednosti grana mogu poprimiti negativne vrijednosti udaljenosti. U tome slučaju, algoritam će to prijaviti i nakon toga će nastaviti sa svojim iteracijama na taj način da okonča pretraživanje na toj ruti gdje se pojavila negativna vrijednost, [23].

Pretpostavke:

- pozitivno-negativne težinske vrijednosti grana
- nema negativnih ciklusa.

Cilj:

- pronaći najkraće staze (spojnice, putove) od izvora do središnjih čvorova
- pronaći najkraće staze od središnjih čvorova do odredišta.

Bellman-Ford algoritam vraća *boolean* vrijednost koja zapravo služi kao pokazatelj u slučaju postoji li ili ne u grafu negativni ciklus koji je dostupan izvorišnom čvoru, i ukoliko postoji, tada u tom slučaju algoritam ne može riješiti problem najkraćeg puta. Ukoliko ne postoji negativni ciklus, tada algoritam normalno nastavlja sa svojim izvođenjem i traženjem najkraćeg puta i računanja njegove težinske vrijednosti. Bellman-Ford algoritam pronalazi najkraće putove od početne točke s ograničenjem na jednu spojnicu, a nakon toga se svakom sljedećom iteracijom broj povećava za vrijednost 1, dok se ne dođe do određene čvoru. Popunjavanje tablice ovisi o broju spojnica. Proces izmjenjivanja informacija između usmjereniča (čvorova) se izvršava i ukoliko nema nikakvih promjena u susjedstvu.

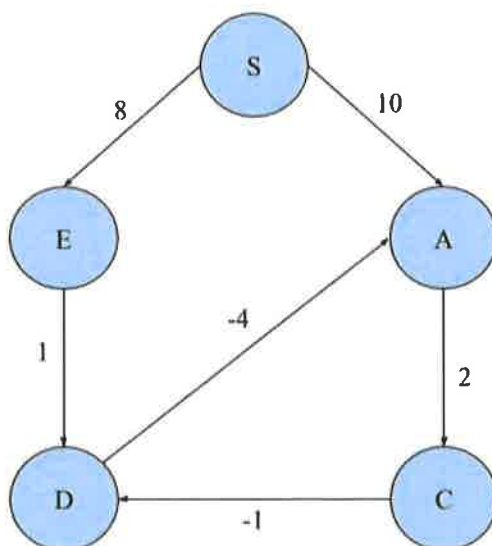
Vremenska kompleksnost Bellman-Fordovog algoritma se može izraziti kao  $V \times E$ , gdje je  $V$  broj svih čvorova u grafu, a  $E$  broj svih grana u grafu, [21].

Formula pomoću koje se računa broj svake sljedeće iteracije u grafu je:

$$D_i^{(h+1)} = \min \left[ D_i^{(h)}, \min (D_j^{(h)} + d_{ji}) \right] \quad (3)$$

Prema [22], izvođenje Bellman-Fordovog algoritma se dijeli u tri koraka:

1. Svaki usmjerenik izračunava udaljenost između sebe i svih ostalih čvorova unutar autonomnog sustava i te informacije pohranjuje u svoju tablicu.
2. Svaki usmjerenik šalje svoju tablicu svim svojim susjednim usmjerenicima.
3. Kada usmjerenik primi tablice udaljenosti od svojih susjednih usmjerenika, tada on izračunava najkraće putove do svih ostalih susjednih usmjerenika i ažurira svoju vlastitu tablicu kako bi zabilježio promjene, i nakon toga je šalje svojim susjednim.



**Slika 17.** Primjer usmjerenog grafa s negativnim vrijednostima

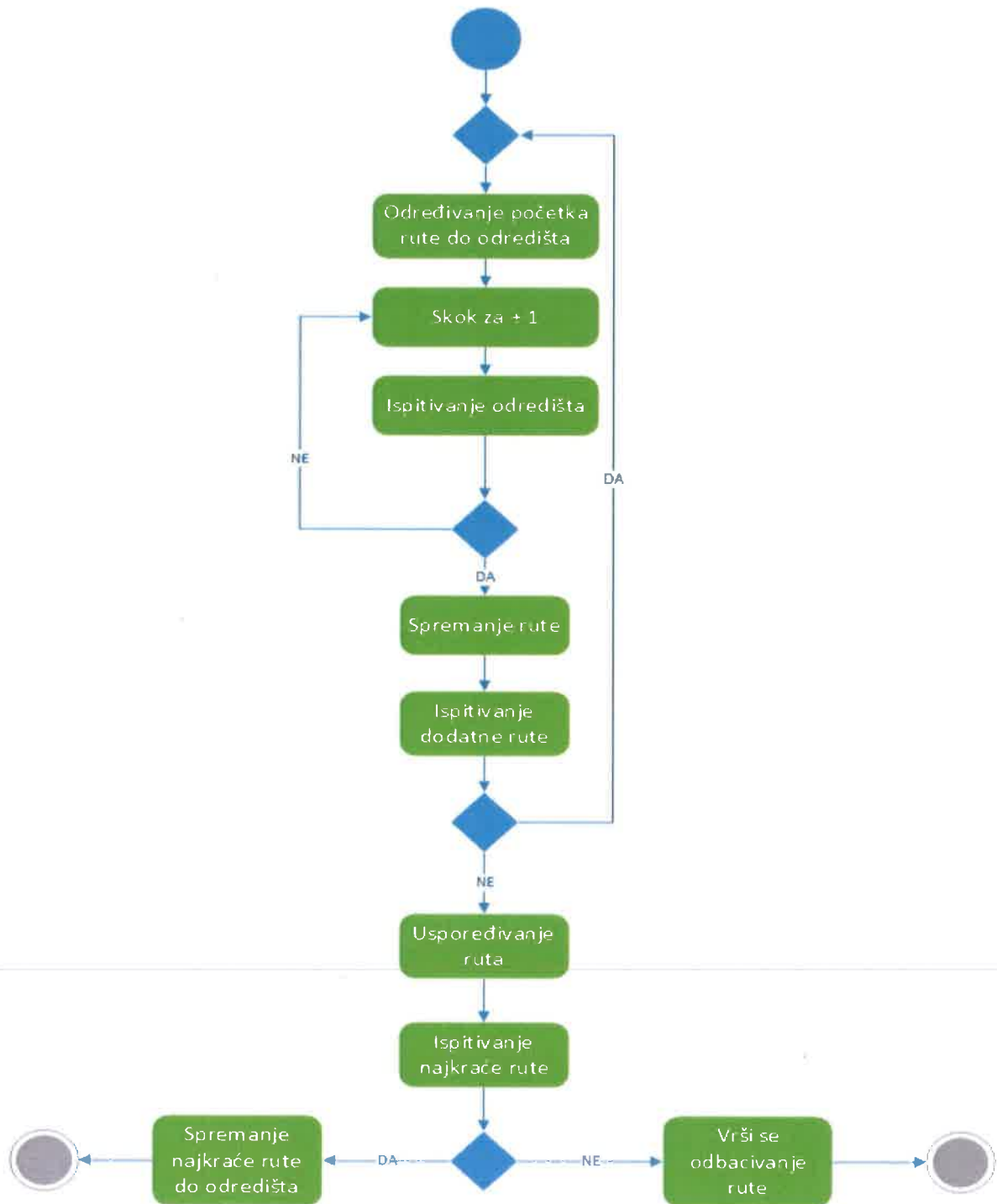
Slikom 17 je prikazan primjer usmjerenog grafa s negativnim vrijednostima nad kojim je primijenjen Bellman-Fordov algoritam. Za primjer koji je dan na slici 17, prema odrednicama Bellman-Fordovog algoritma, algoritam će proći kroz najviše 4 iteracije, jer sadrži 5 čvorova. Izvorišni čvor je čvor S. Prvi korak kod ovoga algoritma je isti kao kod Dijkstra algoritma, tj. početne vrijednosti se trebaju postaviti nula, a vrijednost do svih odredišnih čvorova se postavlja u beskonačnost. U svakoj sljedećoj iteraciji potrebno je proučiti sve grane grafa, te vidjeti postoji li novi najkraći put do čvora koji je sa trenutnim posjećenim čvorom povezan izlaznom granom iz trenutnog čvora, i nakon toga ažurirati tablicu prema novom najkraćem putu.

**Tablica 3.** Prva iteracija Bellman-Fordovog algoritma za navedeni graf

Korak	S	A	C	D	E
1	0	$\infty$	$\infty$	$\infty$	$\infty$
2	0	10	$\infty$	$\infty$	8
3	0	10	12	$\infty$	8
4	0	10	12	$\infty$	8
5	0	10	12	$\infty$	8
6	0	10	12	9	8
<b>Neposjećeni čvorovi: /</b>					

U tablici 3 prikazana je samo prva iteracija Bellman-Fordovog algoritma koji je primijenjen na grafu sa slike 17.

Iteraciju za navedeni primjer je potrebno ponoviti  $V - 1$  puta, gdje je  $V$  broj čvorova, što znači da bi trebalo još napraviti 3 iteracije, jer je jedna već napravljena i prikazana tablicom 3.



**Slika 18.** Dijagram aktivnosti Bellman-Ford algoritma

Izvor: [21]

Na slici 18 je prikazan dijagram aktivnosti Bellman-Fordovog algoritma koji pokazuje način na koji radi ovaj algoritam, tj. postupak pri kojem traži najkraći put do odredišta. Nedostatak ovoga algoritma je što se ažuriranja ne događaju trenutno jer se šire od točke do točke kroz mrežu, pa se zbog toga u tablicama usmjeravanja mogu pojaviti krive informacije.

## 5. Određivanje performansi mreže za različite algoritme usmjeravanja koji se temelje na najkraćem putu

Performanse mreže su definirane pomoću ukupne kvalitete usluge (engl. *Quality of Service* – *QoS*) koju mreža pruža. To uključuje brojne parametre i metrike koje se moraju zajednički analizirati kako bi se procijenila određena mreža. Mjerenje performansi mreže stoga se definira kao ukupnost svih procesa i alata koji se mogu koristiti da bi se kvantitativno i kvalitativno ocijenile performanse mreže i pružili svi potrebni podaci koji bi mogli biti od značaja za rješavanje bilo kakvih problema u mreži. Kvaliteta usluge (*QoS*) kvantificira se skupom parametara koji određuju performanse prometa u mreži, [24].

Prema preporuci ITU u E.800, performanse mreže se definiraju kao „sposobnost mreže ili njenog dijela da osigura funkcije povezane s komuniciranjem među korisnicima“. Iskazuju se pomoću parametara koji ovise o sloju na kojem se nalaze.

Alati pomoću kojih se mogu mjeriti performanse mreže se kategoriziraju u dvije skupine:

- pasivne
- aktivne.

Alati koji se koriste za mjerenje pasivne mreže nadgledaju (ili mjere) postojeće aplikacije na mreži radi prikupljanja podataka o mjernim podacima. Kod ove skupine alata minimaliziraju se svi mogući poremećaji u radu mreže jer sam alat ne uvodi nikakav dodatni promet, i pored toga može se dobiti realna procjena korisničkog iskustva.

Alati koji se koriste za mjerenje aktivne mreže i njenih performansi generiraju podatke koji se mogu prilagoditi osnovnim performansama koristeći unaprijed postavljene rute. Ovakav način rada zahtijeva dodatnu razinu podatkovnog prometa, tako da se mora na odgovarajući način unaprijed zakazati da bi se smanjio utjecaj na postojeći mrežni promet, [24].

Analiza performansi mreže predstavlja analizu svih prikupljenih podataka u cilju procjene odstupanja postojećih od željenih performansi određene mreže. Da bi se analiza mogla uopće i napraviti, prvo je potrebno odrediti i izmjeriti parametre mreže. Većina parametara mreže, tj. performansi mreže su usredotočeni na brzinu podataka i kvalitetu podataka.

Prema [25], osnovni parametri *QoS*-a u paketno orijentiranim mrežama su:

- propusnost

- gubitak paketa
- kašnjenje
- kolebanje kašnjenja.

Propusnost (*throughput*), ili kako se još naziva propusna sposobnost ili efektivni kapacitet, je parametar koji izražava efektivnu brzinu prijenosa podataka izraženu brojem prenesenih bita u sekundi. Ta je veličina manja od kapaciteta kanala izraženog brojem bita u sekundi. Propusnost također označava jednu od veličina koje se koriste za izračun kapaciteta mreže. Propusnost predstavlja mjeru količine podataka koji mogu biti poslani u određenom vremenskom okviru, tj. mjeri brzinu kojom poruke uspješno stižu na odredište.

Gubitak paketa (*packet loss*) nastaje onda kada dođe do prepunjivanja spremnika u čvorovima paketne mreže (ruterima), kao posljedica čekanja paketa u redovima za usmjeravanje, tj. rutiranje. Za neke aplikacije, ukoliko paket kasni prekomjerno, to je isto kao da je izgubljen. Gubitak paketa može biti uzrokovan zastojem rada mreže, performansama usmjerivača, softverskim problemima i sl.

Kašnjenje (*latency, delay*) označuje vrijeme koje je potrebno da se paket prenese od izvorišta do odredišta. Na kašnjenje utječu brojne komponente koje mogu biti fiksne ili varijabilne. Pojava povećanog kašnjenja zbog velikog prometa u mreži zove se zagušenje (*congestion*). Komponente koje se odnose na kašnjenja u mreži teško se mogu predvidjeti jer ovise o trenutačnom opterećenju čvorova kao i o performansama mrežnih elemenata, [25].

Brojni čimbenici utječu na veličinu kašnjenja, a neki od njih su, [26]:

- kašnjenje zbog prolaska (*propagation delay*). Vrijeme potrebno signalu da prođe kroz medij.
- kašnjenje zbog prospajanja (*switching delay*). Vrijeme potrebno da paketna sklopka prihvati cijeli paket te izabere sljedeći skok.
- kašnjenje zbog čekanja na pristup (*access delay*). Vrijeme koje računalo u mreži mora čekati da bi dobilo pristup do zajedničkog medija.
- kašnjenje zbog čekanja u redu (*queuing delay*). Vrijeme koje paket provede čekajući u memoriji paketne sklopke.

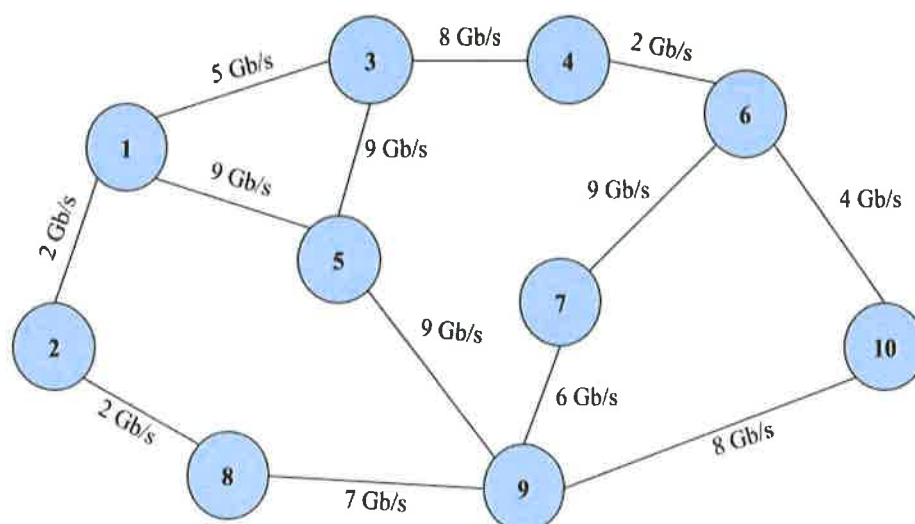
Kolebanje kašnjenja (*jitter*) definira se kao razlika u kašnjenju između susjednih paketa iste sesije. Daje broj koji kaže koliko kašnjenje može biti veće ili manje od svoje prosječne

vrijednosti. Mjeri se u vremenskim jedinicama, npr. u milisekundama. Mjere *jitter*-a ovise o frekvenciji kojom se paketi šalju i fokusiraju isključivo na kratkoročne efekte.

## 5.1. Određivanje performansi mreže

Računanje performansi mreže vrlo je bitan koncept pomoću kojega se mogu utvrditi neke pozitivne i negativne strane pojedine računalne mreže. Tako će na primjer, onaj put koji je okarakteriziran sa većim opterećenjima biti najlošiji izbor za usmjeravanje prometa, a onaj koji ima manja opterećenja će biti najbolji izbor.

Na slici 19 je prikazana je topologija mreže na kojoj će biti provedena analiza. Na svakoj spojnici koja povezuje dva čvora naznačene su vrijednosti koje označavaju kapacitet spojnice. Zadatak je, za zadanu mrežu, odrediti kašnjenja na najkraćim putovima u mreži i vjerojatnost blokiranja na istim tim putovima.



**Slika 19.** Topologija zadane mreže s kapacitetima spojnica

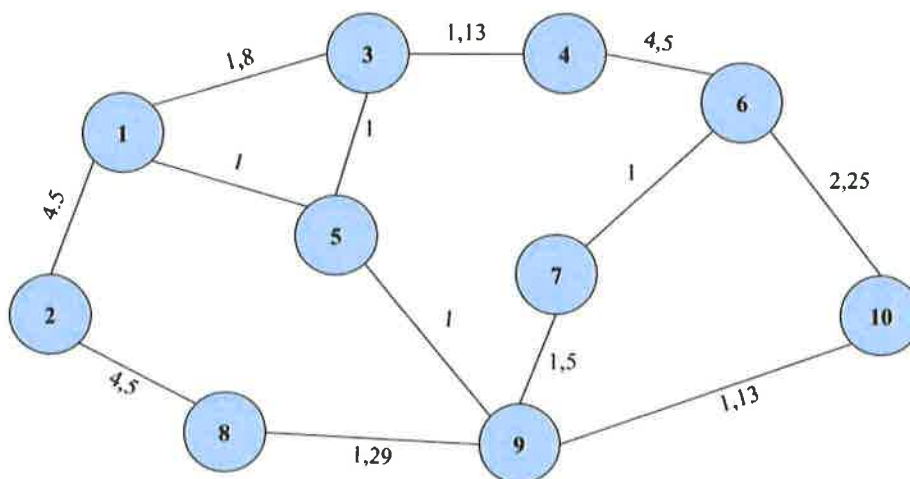
Kako bi se odredile tražene performanse, potrebno je najprije naći najkraće putove u zadanoj mreži. Za pronalazak najkraćih putova koriste se tri algoritma opisana u poglavlju 4.

Za pronalazak najkraćih putova potrebno je poznavati „troškove“ grana/spojnica. S obzirom na to da veći kapacitet znači i kraće vrijeme prijenosa na spojnici, odnosno veći broj prenesenih paketa u jedinici vremena, što je suprotno od veće udaljenosti, trošak puta je izračunat prema sljedećem izrazu:



$$\text{Trošak puta}_i = \frac{\max_{i=1,n} C_i}{C_i} \quad (4)$$

Na slici 19 vidljivo je da maksimalna vrijednost kapaciteta svih grana iznosi 9 Gb/s. Korištenjem izraza (4) izračunate su sve vrijednosti troška na spojnica u zadanoj mreži (slika 20).



**Slika 20.** Prikaz mreže sa troškovima spojnica

One spojnice koje imaju manji trošak puta i povezuju više čvorova su najvjerojatnije i spojnice koje će biti najbolji izbor za usmjeravanje prometa između čvorova.

Nakon ovoga koraka može se pristupiti pronalaženju najkraćih putova u mreži između svih parova čvorova. Za pronalazak najkraćih putova korištena su tri algoritma, a postupka i rezultati su prikazani u nastavku ovog poglavlja.

### 5.1.1. Računanje najkraćeg puta pomoću Dijkstra algoritma

Kao što je već prethodno spomenuto i obrađeno, u poglavlju 4.2., Dijkstrin algoritam služi za pronalaženje najkraćeg i optimalnog puta u mreži između čvorova na način da zbraja njihove vrijednosti koje se nalaze na spojnica koje spajaju čvorove.

Postupak određivanja najkraćeg puta prikazan je tablicom 4. Postupak se sastoji u označavanju čvorova, a ta oznaka predstavlja minimalnu udaljenost od polaznog čvora. Uvijek se razmatra u koji čvor se sve može dospjeti ali samo iz označenih čvorova. Kako je vidljivo iz tablice 4, iz čvora 1, koji u ovom slučaju predstavlja polazište, se može dospjeti u čvorove 2, 3 i 5, no „najkraći put“ je do čvora 5 (trošak puta je najmanji). To znači da čvor 5 prelazi u skup

označenih čvorova i dalje se promatra do kojih se sve čvorova može dospjeti i s kojim minimalnim troškom, što je prikazano u drugom retku tablice. U tom koraku najkraći put (ili najmanji troška) je dolazak do čvora 3, što prevodi čvor 3 u skup označenih čvorova. Postupak se nastavlja dok svi čvorovi ne budu označeni. Oznake troška u tablici obojene crvenom bojom označavaju najkraći put od polaznog čvora (u ovom slučaju je to čvor 1) do čvora označenog u odgovarajućem stupcu tablice.

**Tablica 4.** Redoslijed označavanja čvorova u postupku pronalaženja najkraćeg puta za čvor 1

	označeni čvorovi	2	3	4	5	6	7	8	9	10
1	1	4,5	1,8	∞	1	∞	∞	∞	∞	∞
2	15	4,5	1,8	∞		∞	∞	∞	2	∞
3	153	4,5		2,93		∞	∞	∞	2	∞
4	1539	4,5		2,93		∞	3,5	3,29		3,13
5	15394	4,5				7,43	3,5	3,29		3,13
6	1539410	4,5				5,38	3,5	3,29		
7	15394108	4,5				5,38	3,5			
8	153941087	4,5				4,5				
9	1539410876	4,5								
10	15394108762									

Iz tako popunjene tablice se mogu se iščitati sve rute za usmjeravanje paketa kojima je čvor 1 polazište. Na isti način pristupa se pronalaženju udaljenosti od svih čvorova ka svim ostalim čvorovima. Pronalazak najkraćih putova iz čvora 2 ka svim ostalim čvorovima i rezultati su prikazani u tablici 6.

**Tablica 5.** Popis najkraćih putova za sve parove čvorova u zadanoj neopterećenoj mreži

Najkraći putovi iz čvora 1	Najkraći putovi iz čvora 2	Najkraći putovi iz čvora 3	Najkraći putovi iz čvora 4	Najkraći putovi iz čvora 5	Najkraći putovi iz čvora 6	Najkraći putovi iz čvora 7	Najkraći putovi iz čvora 8	Najkraći putovi iz čvora 9	Najkraći putovi iz čvora 10
1-5	2-1	3-1	4-3-1	5-1	6-7-9-5-1	7-9-5-1	8-9-5-1	9-5-1	10-9-5-1
1-3	2-8	3-1-2	4-3-1-2	5-1-2	6-7-9-8-2	7-9-8-2	8-2	9-8-2	10-9-8-2
1-5-9	2-1-5	3-4	4-3	5-3	6-9-5-3-4	7-9-5-3	8-9-5-3	9-5-3	10-9-5-3
1-3-4	2-8-9	3-5	4-3-5	5-3-4	6-4	7-9-5-3-4	8-9-5-3-2	9-5-3-4	10-9-5-3-4
1-5-9-10	2-1-3	3-5-9	4-3-5-9	5-9	6-7-9-5	7-9-5	8-9-5	9-5	10-9-5
1-5-9-8	2-8-9-10	3-5-9-10	4-3-5-9-10	5-9-10	6-7	7-6	8-9-7-6	9-7-6	10-6
1-5-9-7	2-8-9-7	3-5-9-8	4-6	5-9-8	6-10	7-9	8-9-7	9-7	10-9-7
1-5-9-7-6	2-1-3-4	3-5-9-7	4-3-5-9-8	5-9-7	6-7-9	7-9-10	8-9	9-8	10-9-8
1-2	2-8-9-7-6	3-5-9-7-6	4-3-5-9-7	5-9-7-6	6-7-9-8	7-9-8	8-9-10	9-10	10-9

Postupak određivanja najkraćih putova iz ostalih čvorova nije prikazan u ovom radu, ali udaljenosti za sve čvorove su određeni i prikazani u tablici 5.

Iz tablice 5 može se vidjeti da će čvor 1 neovisno o odredištu usmjeravati pakete u čvor 5, osim u slučaju ako je odredište čvor 2 s kojim je čvor 1 u direktnoj vezi. Nadalje se može vidjeti da je u većinu ruta uključen dio puta 1-5-9.

**Tablica 6.** Redoslijed označavanja čvorova u postupku pronalaženja najkraćeg puta za čvor 2

	označeni čvorovi	1	2	4	5	6	7	8	9	10
1	2	4,5	∞	∞	∞	∞	∞	4,5	∞	∞
2	21		6,3	∞	5,5	∞	∞	4,5	∞	∞
3	218		6,3	∞	5,5	∞	∞		5,79	∞
4	2185		6,3	∞		∞	∞		5,79	∞
5	21859		6,3	∞		∞	7,29			6,92
6	218593			7,43		∞	7,29			6,92
7	21859310			7,43		9,17	7,29			
8	218593107			7,43		8,29				
9	2185931074					8,29				
10	21859310746									

**Tablica 7.** Redoslijed označavanja čvorova u postupku pronalaženja najkraćeg puta

	Izvorišni čvor	Redoslijed označavanja čvorova
1	1	1-5-3-9-4-10-8-7-6-2
2	2	2-1-8-5-9-3-10-7-4-6
3	3	3-5-4-1-9-10-7-6-2
4	4	4-3-5-1-9-10-6-8-7-2
5	5	5-1-3-9-4-10-8-7-6-2
6	6	6-7-10-9-5-8-1-3-4-2
7	7	7-6-9-5-10-8-1-3-4-2
8	8	8-9-5-10-7-6-1-3-4-2
9	9	9-5-10-8-7-1-3-6-4-2
10	10	10-9-5-6-8-1-3-7-4-2

Kao što je vidljivo u tablici 7, udaljenost među pojedinim čvorovima ne ovise o tome da li se pojedini čvor nalazi odmah pored drugog čvora, tj. da li mu je prvi susjedni ili ne, nego ovise o trošku puta na toj spojnici koja ih povezuje. Na taj način su se dobili rezultati koji su prikazani u tablici 7, i vidljivo je prema rezultatima da svaki čvor ima drugačiju udaljenost prema krajnjem čvoru, jer je svaki čvor drugačije povezan sa ostalima, i na taj način će i trošak puta prilikom zbrajanja biti drugačiji, i na taj način i krajnji rezultati.

Nakon što se dobiju rezultati najkraćih putova među svim čvorovima, potrebno je odrediti kašnjenje na putu. U prvom slučaju određivanja kašnjenja na putu, pretpostaviti će se da je mreža neopterećena.

#### **5.1.1.1. Kašnjenje na putu u neopterećenoj mreži**

Pod neopterećenom mrežom u ovom slučaju se smatra da su sve spojnice malo opterećene i to opterećenje je isto za sve spojnice ( $\rho = 0,1$ ). Iako su veličine paketa u mreži različite, u ovom poglavlju određivanje ukupnog kašnjenja na putu je provedeno uz pretpostavku da je

prosječna veličina paketa  $\bar{p} = 1500 \text{ Byte}$ . Pored navedenog, pretpostavka je da prosječno vrijeme obrade paketa u pojedinom čvoru  $T_{obr}$  iznosi će  $1 \mu s$ .

Na osnovu gore navedenih vrijednosti, kašnjenje puta će se računati prema modelu  $M/M/1$ , prema sljedećoj formuli, [27]:

$$T_{q_{j-k}} = \sum_{i=1}^{\text{broj čvorova na putu}} T_{obr_i} + \sum_{i=1}^{\text{broj čvorova na putu}} T_{w_i} + \sum_{i=1}^{\text{broj spojnica uključenih u put}} T_{s_i} \quad (5)$$

Pri čemu se vrijeme čekanja i prosječno vrijeme prijensa paketa na spojnici računa prema sljedećim formulama, [27]:

$$T_w = \frac{\rho \times p}{(1 - \rho) \times C} \quad (6)$$

$$T_s = \frac{p}{C} \quad (7)$$

Uvedene oznake imaju sljedeće značenje:

$T_{obr_i}$  = vrijeme obrade  $k$  u čvoru  $i$

$T_{w_i}$  = prosječno vrijeme čekanja paketa na posluživanje u čvoru  $i$

$T_{s_i}$  = prosječno vrijeme prijensa paketa na spojnici  $i$

Prema tablici 5, prva ruta između čvorova 1 i 5 i s obzirom na to da su ta dva čvora direktno povezana, komponente koje čine kašnjenje na putu između ta dva čvora su:

$$\begin{aligned} T_{q_{1-5}} &= T_{obr1} + T_{w1} + T_{s_{1-5}} + T_{obr5} = 10^{-6} + \frac{0.1 \times \frac{12000}{9 \times 10^9}}{1 - 0.1} + \frac{12000}{9 \times 10^9} + 10^{-6} \\ &= 3,481 \times 10^{-6} s \end{aligned}$$

Na isti način određuje se i kašnjenje između čvorova 1 i 3:

$$\begin{aligned} T_{q_{1-3}} &= T_{obr1} + T_{w1} + T_{s_{1-3}} + T_{obr3} = 10^{-6} + \frac{0.1 \times \frac{12000}{5 \times 10^9}}{1 - 0.1} + \frac{12000}{5 \times 10^9} + 10^{-6} \\ &= 4.667 \times 10^{-6} s \end{aligned}$$

Kašnjenje pakete na putu koji se sastoji od više spojnica i čvorova, kao što je put između čvorova 1 i 10 (ruta: **1-5-9-10**):

$$T_{q1-5-9-10} = T_{obr1} + T_{w1} + T_{s1-5} + T_{obr5} + T_{w5} + T_{s5-9} + T_{obr9} + T_{w9} + T_{s9-10} + T_{obr10} \\ = 0,000010629 \text{ s}$$

Vrijednosti kašnjenja paketa na svim najkraćim putovima iz čvora 1 prikazane su u tablici 8.

**Tablica 8.** Kašnjenje paketa na svim najkraćim putovima iz čvora 1 za neopterećenu mrežu

Ruta	Kašnjenje na najkraćem putu [s]
1-5	3,48E-06
1-3	4,67E-06
1-5-9	6,96E-06
1-5-9-10	1,06E-05
1-3-4	8,33E-06
1-5-9-8	1,09E-05
1-5-9-7	1,12E-05
1-5-9-7-6	1,47E-05
1-2	7,00E-06

Iz rezultata prikazanih u tablici 8 može se vidjeti da se paketi najmanje zadržavaju na putu od čvora 1 do čvora 5 (ruta: 1-5). Razlog tomu je što prolaze samo kroz dva čvora i jednom spojnicom koja ima vrlo velik kapacitet. Najviše paketi kasne na putu od čvora 1 do čvora 6 (ruta: **1-5-9-7-6**). Razlog tomu se što na tom putu paketi prolaze kroz najveći broj čvorova. Nadalje, može se zaključiti da paketi na rutama sa samo jednom spojnicom između izvorišta i odredišta kasne do 7  $\mu\text{s}$ , s dvije spojnice do 6,9  $\mu\text{s}$ , s tri spojnice od 10 do 11  $\mu\text{s}$  i s četiri spojnice 14,665  $\mu\text{s}$ . može se također vidjeti da paketi manje kasne na ruti koja se sastoji do tri čvora i

dvije spojnice (ruta: 1-5-9), nego na ruti koja se sastoji samo od dva čvora i jedne spojnice (ruta: 1-2). Razlog tomu su kapaciteti spojnica.

### 5.1.1.2. Kašnjenje na putu u opterećenoj mreži

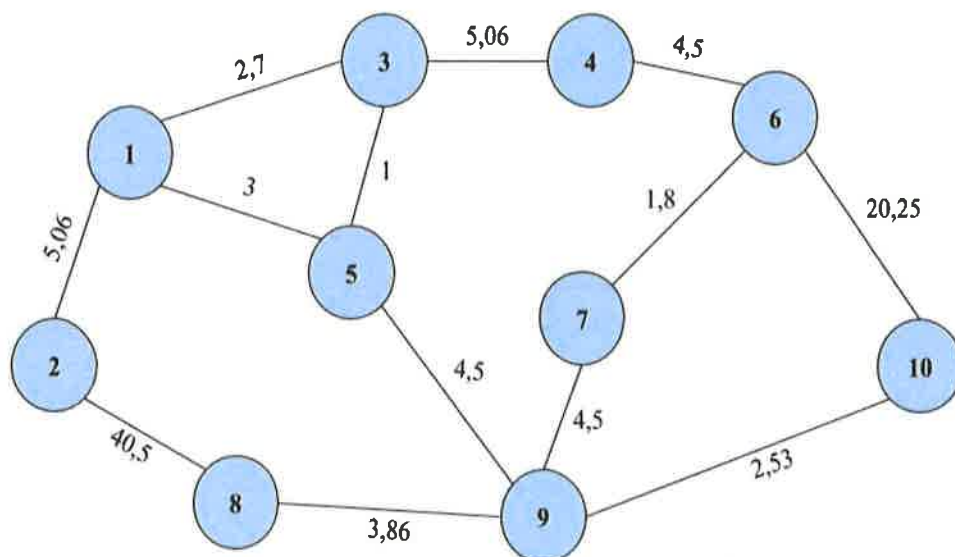
Pod opterećenom mrežom u ovom primjeru smatra se da spojnice mogu imati i znatno veće opterećenje od 0,1. Za postavljanje opterećenja ( $\rho$ ) na pojedinoj spojnici zadane mreže (slika 19) korištena je funkcija u RANDOM u *excel*-u (tablica 9). Nakon određivanja opterećenja pojedine spojnice potrebno je odrediti preostali slobodni kapacitet (tablica 9).

**Tablica 9.** Opterećenja spojnica u mreži i troškovi spojnica

Spojnica	Kapacitet	Preostali slobodni kapacitet	Postotak opterećenja pomnožen sa polaznim kapacitetom	Postotak zauzeća spojnice	Trošak
1-3	5	3	2	40%	2,70
1-5	9	2,7	6,3	70%	3,00
1-2	2	1,6	0,4	20%	5,06
3-4	8	1,6	6,4	80%	5,06
3-5	9	8,1	0,9	10%	1,00
4-6	2	1,8	0,2	10%	4,50
6-10	4	0,4	3,6	90%	20,25
6-7	9	4,5	4,5	50%	1,80
9-10	8	3,2	4,8	60%	2,53
9-8	7	2,1	4,9	70%	3,86
8-2	2	0,2	1,8	90%	40,50
9-5	9	1,8	7,2	80%	4,50
9-7	6	1,8	4,2	70%	4,50

Uzimajući u obzir vrijednosti preostalog slobodnog kapaciteta na spojnica određeni su novi troškovi spojnica za zadanu mrežu (tablica 9) i prikazani na slici 21. S obzirom na to da se stanje u mreži promijenilo potrebno je odrediti nove najkraće putove u mreži za usmjeravanje paketa od izvorišta do odredišta.





**Slika 21.** Topologija zadane mreže s troškovima spojnica za opterećenu mrežu

Postupak određivanja najkraćeg puta je identičan postupku u neopterećenoj mreži kako je opisano u poglavlju 5.1.1. Rezultati za najkraće putove u opterećenoj mreži iz svih čvorova ka svim čvorovima prikazani su u tablici 10.

**Tablica 10.** Popis najkraćih putova za sve parove čvorova u zadanoj opterećenoj mreži

Najkraći putovi iz čvora 1	Najkraći putovi iz čvora 2	Najkraći putovi iz čvora 3	Najkraći putovi iz čvora 4	Najkraći putovi iz čvora 5	Najkraći putovi iz čvora 6	Najkraći putovi iz čvora 7	Najkraći putovi iz čvora 8	Najkraći putovi iz čvora 9	Najkraći putovi iz čvora 10
1-2	2-1	3-1	4-3-1	5-1	6-4-3-1	7-9-5-1	8-9-5-1	9-5-1	10-9-5-1
1-3	2-1-3	3-1-2	4-3-1-2	5-1-2	6-4-3-1-2	7-9-5-2-1	8-9-5-1-2	9-5-1-2	10-9-5-1-2
1-3-4	2-1-3-4	3-4	4-3	5-3	6-4-3	7-9-5-3	8-9-5-3	9-5-3	10-9-5-3
1-5	2-1-5	3-5	4-3-5	5-3-4	6-4	7-6-4	8-9-5-3-4	9-5-3-4	10-9-5-3-4
1-3-4-6	2-1-3-4-6	3-4-6	4-6	5-3-4-6	6-4-3-5	7-9-5	8-9-5	9-5	10-9-5
1-5-9-7	2-1-5-9-7	3-5-9-7	4-6-7	5-9-7	6-7	7-6	8-9-7-6	9-7-6	10-9-7-6
1-5-9-8	2-1-5-9-8	3-5-9-8	4-3-5-9-8	5-9-8	6-7-9-8	7-9-8	8-9-7	9-7	10-9-7
1-5-9	2-1-5-9	3-5-9	4-3-5-9	5-9	6-7-9	7-9	8-9	9-8	10-9-8
1-5-9-10	2-1-5-9-10	3-5-9-10	4-3-5-9-10	5-9-10	6-7-9-10	7-9-10	8-9-10	9-10	10-9

U tablici 10 označene su žutom bojom rute koje su ostale iste i nakon opterećenja mreže, a rute koje nisu označene žutom bojom su one koje su promijenjene nakon opterećenja mreže. Kako je vidljivo iz tablice, za čvor 1, 3, 4, 5 i 9 promijenjena je samo jedna ruta, dok je

opterećenje mreže utjecalo na promjenu čak pet ruta iz čvorova 2 i 6. Kod ostalih čvorova nakon opterećenja mreže zadanim prometom promijenile su se po dvije rute.

Kašnjenja na putu od izvorišta do odredišta izračunata su i za opterećenu mrežu primjenom izraza 5-7. Kašnjenje na putovima iz čvora 1 se povećava zbog povećanja opterećenja na spojnica koje je navedeno u tablici 8. Izračun kašnjenja prikazan u nastavku je samo za parove čvorova za koje je kašnjenje određeno i kod neopterećene mreže. Sve vrijednosti kašnjenja paketa na putu od čvora 1 do ostalih čvorova prikazane su u tablici 11, za rute prikazane u tablici 10.

$$T_{q1-2} = T_{obr1} + T_{w1} + T_{s1-2} + T_{obr2} = 10^{-6} + \frac{0.2 \times \frac{12000}{2 \times 10^9}}{1 - 0.2} + \frac{12000}{2 \times 10^9} + 10^{-6}$$

$$= 9,5 \times 10^{-6} s$$

$$T_{q1-3} = T_{obr1} + T_{w1} + T_{s1-3} + T_{obr3} = 10^{-6} + \frac{0.4 \times \frac{12000}{5 \times 10^9}}{1 - 0.4} + \frac{12000}{5 \times 10^9} + 10^{-6}$$

$$= 6 \times 10^{-6} s$$

Kašnjenje pakete na putu koji se sastoji od više spojnica i čvorova, kao što je put između čvorova 1 i 10 (ruta: **1-5-9-10**):

$$T_{q1-10} = T_{obr1} + T_{w1} + T_{s1-5} + T_{obr5} + T_{w5} + T_{s5-9} + T_{obr9} + T_{w9} + T_{s9-10} + T_{obr10}$$

$$= 10^{-6} + \frac{0,7 \times \frac{12000}{9 \times 10^9}}{1 - 0,7} + \frac{12000}{9 \times 10^9} + 10^{-6} + \frac{0,8 \times \frac{12000}{9 \times 10^9}}{1 - 0,8} + \frac{12000}{9 \times 10^9}$$

$$+ 10^{-6} + \frac{0,6 \times \frac{12000}{8 \times 10^9}}{1 - 0,6} + \frac{12000}{8 \times 10^9} + 10^{-6} = 1,781667 \times 10^{-5} s$$

**Tablica 11.** Kašnjenje paketa na najkraćim putovima iz čvora 1 za opterećenu mrežu

Spojnicica	Kašnjenje na najkraćem putu [s]
1-2	9,50E-06
1-3	6,00E-06
1-3-4	1,45E-05
1-5	6,40E-06
1-3-4-6	2,21E-05
1-5-9-7	2,07E-05
1-5-9-8	1,98E-05
1-5-9	1,31E-05
1-5-9-10	1,78E-05

Iz rezultata prikazanih u tablici 11 može se vidjeti da se paketi najmanje zadržavaju na putu od čvora 1 do čvora 3 (ruta: 1-3). Razlog tomu je što prolaze samo kroz dva čvora i jednom spojnicom koja ima velik kapacitet i malo je opterećenje navedene spojnice (40%). U neopterećenoj mreži najmanje kašnjenje je na ruti 1-5, ali zbog velikog opterećenja na toj spojnici (70%), kašnjenje na toj ruti se povećalo u odnosu na rutu 1-3. Najviše paketi kasne na putu od čvora 1 do čvora 6 (ruta: **1-3-4-6**), što je slučaj i kod neopterećene mreže, iako se rute razlikuju.

### 5.1.1.3. Gubici u neopterećenoj i opterećenoj mreži

Vjerojatnost blokiranja na putu za prethodno definirane najkraće putove opterećene i neopterećene mreže (poglavlja 5.1.1. i 5.1.1.2.) određuje se korištenjem izraza, [28]:

$$p_{b_{j-k}} = 1 - \prod_i (1 - p_{b_i}) \quad (8)$$

Pretpostavke koje se koriste u određivanju vjerojatnosti u ovom diplomskom radu su sljedeće:

1. za neopterećenu mrežu smatrat će se da je vjerojatnost blokiranja na svim spojnica jednaka i ona iznosi 0,001
2. za opterećenu mrežu smatrat će se da je vjerojatnost blokiranja na spojnica:
  - 0,1 za one spojnice koje su opterećene 80% i više
  - 0,05 za one spojnice koje su opterećene od 50% do 80%
  - 0,01 za one spojnice koje su opterećene manje od 50%.

Vjerojatnosti da paket ne bude usmjeren najkraćim putem prikazane su u tablici 13.

Primjer računanja vjerojatnosti blokiranja za spojnicu **1-2** je:

$$p_{b1-2} = 1 - [(1 - p_{b1-2})] = 1 - [(1 - 0,001)] = 0,001$$

Vjerojatnost blokiranja na putu koji se sastoji od više spojnica i čvorova, kao što je put između čvorova 1 i 6 (ruta: **1-3-4-6**):

$$\begin{aligned} p_{b1-6} &= 1 - [(1 - p_{b1-3}) \times (1 - p_{b3-4}) \times (1 - p_{b4-6})] \\ &= 1 - [(1 - 0,001) \times (1 - 0,001) \times (1 - 0,001)] = 0,002997 \end{aligned}$$

**Tablica 12.** Vjerojatnosti blokiranja na najkraćim putovima u neopterećenoj mreži

Najkraći putovi iz čvora 1	Rezultat	Najkraći putovi iz čvora 2	Rezultat	Najkraći putovi iz čvora 3	Rezultat	Najkraći putovi iz čvora 4	Rezultat	Najkraći putovi iz čvora 5	Rezultat
1-2	0,001	2-1	0,001	3-1	0,001	4-3-1	0,001999	5-1	0,001
1-3	0,001	2-1-3	0,001999	3-1-2	0,001999	4-3-1-2	0,002997	5-1-2	0,001999
1-3-4	0,001999	2-1-3-4	0,002997	3-4	0,001	4-3	0,001	5-3	0,001
1-5	0,001	2-1-5	0,001999	3-5	0,001	4-3-5	0,001999	5-3-4	0,001999
1-3-4-6	0,002997	2-1-3-4-6	0,003994	3-4-6	0,001999	4-6	0,001	5-3-4-6	0,002997
1-5-9-7	0,002997	2-1-5-9-7	0,003994	3-5-9-7	0,002997	4-6-7	0,001999	5-9-7	0,001999
1-5-9-8	0,002997	2-1-5-9-8	0,003994	3-5-9-8	0,002997	4-3-5-9-8	0,003994	5-9-8	0,001999
1-5-9	0,001999	2-1-5-9	0,002997	3-5-9	0,001999	4-3-5-9	0,002997	5-9	0,001
1-5-9-10	0,002997	2-1-5-9-10	0,003994	3-5-9-10	0,002997	4-3-5-9-10	0,003994	5-9-10	0,001999
Najkraći putovi iz čvora 6	Rezultat	Najkraći putovi iz čvora 7	Rezultat	Najkraći putovi iz čvora 8	Rezultat	Najkraći putovi iz čvora 9	Rezultat	Najkraći putovi iz čvora 10	Rezultat
6-7-9-5-1	0,002997	7-9-5-1	0,002997	8-9-5-1	0,002997	9-5-1	0,001999	10-9-5-1	0,002997
6-7-9-8-2	0,003994	7-9-8-2	0,003994	8-2	0,003994	9-8-2	0,002997	10-9-8-2	0,003994
6-9-5-3-4	0,001999	7-9-5-3	0,002997	8-9-5-3	0,002997	9-5-3	0,001999	10-9-5-3	0,002997
6-4	0,001	7-9-5-3-4	0,001999	8-9-5-3-2	0,003994	9-5-3-4	0,002997	10-9-5-3-4	0,003994
6-7-9-5	0,002997	7-9-5	0,001999	8-9-5	0,001999	9-5	0,001	10-9-5	0,001999
6-7	0,001	7-6	0,001	8-9-7-6	0,002997	9-7-6	0,001999	10-6	0,002997
6-10	0,002997	7-9	0,001999	8-9-7	0,001999	9-7	0,001	10-9-7	0,001999
6-7-9	0,001999	7-9-10	0,001	8-9	0,001	9-8	0,001	10-9-8	0,001999
6-7-9-8	0,002997	7-9-8	0,001999	8-9-10	0,001999	9-10	0,001	10-9	0,001

Iz rezultata prikazanih u tablici 12 vidi se da vjerojatnost blokiranja na putu za neopterećenu mrežu raste s porastom broja spojnica kroz koje paket prolazi na putu od izvorišta do odredišta. Najmanje su vjerojatnosti blokiranja na onim rutama koje se sastoje samo od jedne spojnice, a najveće vrijednosti su na rutama koje se sastoje od 4 spojnice na putu. Dobivene vrijednosti su rezultati izraza 8.

Za opterećenu mrežu, kako je prethodno navedeno, vjerojatnost blokiranja nije ista za sve spojnice u mreži, nego ovisi o postotku opterećenja pojedine spojnice. U tablici 13 su prikazana opterećenja za svaku spojnicu, i vjerojatnost blokiranja na osnovu opterećenja.

**Tablica 13.** Vjerojatnost blokiranja za opterećenu mrežu

	Spojnice	Opterećenje %	Izračun dostupnosti spojnice $1 - p_{b_i}$	Vjerojatnost dostupnosti spojnice
1	1-3	40	1-0,01	0,99
2	1-5	70	1-0,05	0,95
3	1-2	20	1-0,01	0,99
4	3-4	80	1-0,1	0,9
5	3-5	10	1-0,01	0,99
6	4-6	10	1-0,01	0,99
7	6-10	90	1-0,1	0,9
8	6-7	50	1-0,05	0,95
9	9-10	60	1-0,05	0,95
10	8-9	70	1-0,05	0,95
11	8-2	90	1-0,1	0,9
12	9-5	80	1-0,1	0,9
13	9-7	70	1-0,05	0,95

Na osnovu rezultata koji su prikazani u posljednje stupcu tablice 13, računaju se vjerojatnosti blokiranja na isti način kao i za neopterećenu mrežu (izraz 8).

Primjer računanja vjerojatnosti blokiranja za spojnicu 1-2 slijedi:

$$p_{b_{1-2}} = 1 - [(1 - p_{b_{1-2}})] = 1 - [(1 - 0,01)] = 0,01$$

Primjer računanja vjerojatnosti blokiranja za spojnicu 1-3-4 slijedi:

$$p_{b_{1-4}} = 1 - [(1 - p_{b_{1-3}}) \times (1 - p_{b_{3-4}})] = 1 - [(1 - 0,01) \times (1 - 0,1)] = 0,109$$

Rezultati vjerojatnosti blokiranja na najkraćim putevima od izvorišta do odredišta (prema tablici 10) prikazani su u tablici 14.

**Tablica 14.** Vjerojatnost blokiranja na najkraćim putovima u opterećenoj mreži

Najkraći putovi iz čvora 1	Rezultat	Najkraći putovi iz čvora 2	Rezultat	Najkraći putovi iz čvora 3	Rezultat	Najkraći putovi iz čvora 4	Rezultat	Najkraći putovi iz čvora 5	Rezultat
1-2	0,01	2-1	0,01	3-1	0,01	4-3-1	0,109	5-1	0,05
1-3	0,01	2-1-3	0,0199	3-1-2	0,0199	4-3-1-2	0,11791	5-1-2	0,0595
1-3-4	0,109	2-1-3-4	0,11791	3-4	0,1	4-3	0,1	5-3	0,01
1-5	0,05	2-1-5	0,0595	3-5	0,01	4-3-5	0,109	5-3-4	0,109
1-3-4-6	0,11791	2-1-3-4-6	0,126731	3-4-6	0,109	4-6	0,01	5-3-4-6	0,11791
1-5-9-7	0,18775	2-1-5-9-7	0,195873	3-5-9-7	0,15355	4-6-7	0,0595	5-9-7	0,145
1-5-9-8	0,18775	2-1-5-9-8	0,195873	3-5-9-8	0,15355	4-3-5-9-8	0,238195	5-9-8	0,145
1-5-9	0,145	2-1-5-9	0,15355	3-5-9	0,109	4-3-5-9	0,1981	5-9	0,1
1-5-9-10	0,18775	2-1-5-9-10	0,195873	3-5-9-10	0,15355	4-3-5-9-10	0,238195	5-9-10	0,145
Najkraći putovi iz čvora 6	Rezultat	Najkraći putovi iz čvora 7	Rezultat	Najkraći putovi iz čvora 8	Rezultat	Najkraći putovi iz čvora 9	Rezultat	Najkraći putovi iz čvora 10	Rezultat
6-4-3-1	0,11791	7-9-5-1	0,18775	8-9-5-1	0,18775	9-5-1	0,145	10-9-5-1	0,18775
6-4-3-1-2	0,126731	7-9-5-1-2	0,195873	8-9-5-1-2	0,195873	9-5-1-2	0,15355	10-9-5-1-2	0,195873
6-4-3	0,109	7-9-5-3	0,15355	8-9-5-3	0,15355	9-5-3	0,109	10-9-5-3	0,15355
6-4	0,01	7-6-4	0,0595	8-9-5-3-4	0,238195	9-5-3-4	0,1981	10-9-5-3-4	0,238195
6-4-3-5	0,11791	7-9-5	0,145	8-9-5	0,145	9-5	0,1	10-9-5	0,145
6-7	0,05	7-6	0,05	8-9-7-6	0,142625	9-7-6	0,0975	10-9-7-6	0,142625
6-7-9-8	0,142625	7-9-8	0,0975	8-9-7	0,0975	9-7	0,05	10-9-7	0,0975
6-7-9	0,0975	7-9	0,05	8-9	0,05	9-8	0,05	10-9-8	0,0975
6-7-9-10	0,142625	7-9-10	0,0975	8-9-10	0,0975	9-10	0,05	10-9	0,05

Upoređujući rezultate u prikazane u tablicama 14 i 12 može se zaključiti da vjerojatnost blokiranja u tablici 14 ne ovise samo o broju spojnica kroz koje paket prolazi na putu od izvorišta do odredišta. Primjerice vjerojatnost blokiranja, odnosno vjerojatnost da paket ne bude usmjeren najkraćim putem, je veća na ruti 3-4 nego na ruti 4-6-7 koja se sastoji od dvije spojnice.

### 5.1.2. Računanje najkraćeg puta pomoću Floyd-Warshallovog algoritma

Kao što je već prethodno spomenuto i obrađeno u poglavlju 4.3., Floyd-Warshallov algoritam rješava problem najkraćeg puta svih parova čvorova na usmjerenom grafu  $G=(V, E)$ . Algoritam radi po principu da pronalazi najkraći put među svim parovima čvorova usmjerenog težinskog grafa, a rezultat svih računanja je matrica u kojoj se upisuje vrijednost udaljenosti između svih čvorova u mreži.

Za računanje ovoga algoritma korišten je primjer mreže sa slike 20, prema kojoj će se računati udaljenost čvorova jednih od drugih.

Prvotno je potrebno kreirati matricu dimenzije  $m \times m$ , gdje  $m$  označava broj čvorova. U ovome slučaju će tada matrica biti veličine  $10 \times 10$ , jer se mreža sastoji od 10 čvorova. Dijagonala matrice uvijek mora imati vrijednosti jednake nuli, a ako ne postoji izravna veza između dva čvora, tada se upisuje vrijednost beskonačno.

U tablici 15 prikazane su vrijednosti udaljenosti svih čvorova od ostalih. Prvi stupac označava izvorišne čvorove prema kojima se gledala udaljenost, a prvi redak označava odredišne čvorove.

**Tablica 15.** Početna udaljenost između svih čvorova za neopterećenu mrežu

	1	2	3	4	5	6	7	8	9	10
1	0	4,5	1,8	$\infty$	1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
2	4,5	0	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	4,5	$\infty$	$\infty$
3	1,8	$\infty$	0	1,13	1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
4	$\infty$	$\infty$	1,13	0	$\infty$	4,5	$\infty$	$\infty$	$\infty$	$\infty$
5	1	$\infty$	1	$\infty$	0	$\infty$	$\infty$	$\infty$	1	$\infty$
6	$\infty$	$\infty$	$\infty$	4,5	$\infty$	0	1	$\infty$	$\infty$	2,25
7	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	1	0	$\infty$	1,5	$\infty$
8	$\infty$	4,5	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	0	1,29	$\infty$
9	$\infty$	$\infty$	$\infty$	$\infty$	1	$\infty$	1,5	1,29	0	1,13
10	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	2,25	$\infty$	$\infty$	1,13	0

Nakon početne udaljenosti, u nastavku se računa udaljenost svih čvorova od ostalih prema izrazu (2), Vrijednost  $k$  označava broj iteracije, a u ovom slučaju će biti 10 iteracija, jer se mreža sastoji od 10 čvorova.

Tablicom 16 je prikazana udaljenost čvorova nakon prve iteracije, i crveno označena polja su one udaljenosti koje su se promijenile, tj. manje su nego u prethodnoj tablici (početna udaljenost).



Primjerice, polje 2-1, koje ima početnu udaljenost 4,5, ostaje nepromijenjena prema izrazu (2), u kojemu se gleda udaljenost polja 2-1. Vrijednost  $i$  je 2, a vrijednost  $j$  je 1, i tada bi odnos bio:  $dist(2,1) > dist(2,1) + dist(1,1)$ . Prema tablici 15, vrijednosti u tim poljima su 4,5 i 0, čime je zbroj tih vrijednosti jednak početnoj, stoga se u tome polju ništa ne mijenja. Ukoliko je ta vrijednost manja od postojeće tada bi se nova zamijenila sa starom, no u ovome slučaju se radi o istom iznosu, pa se ne moraju vrijednosti mijenjati.

No, primjerice polje 3-2 je zamijenjeno sa novom vrijednošću. Stara udaljenost između čvorova 3-2 je beskonačno, a iz usporedbe prema izrazu (2) zbrajamo vrijednosti polja 3-1 i 1-2, a one bi bile 1,8 i 4,5, čiji je rezultat 6,3. Taj rezultat je manji od beskonačno stoga se on zamjenjuje. Na isti princip se provjerava i popunjava ostatak tablice.

**Tablica 16.** Udaljenost čvorova nakon prve iteracije ( $k=1$ )

	1	2	3	4	5	6	7	8	9	10
1	0	4,5	1,8	$\infty$	1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
2	4,5	0	6,3	$\infty$	5,5	$\infty$	$\infty$	4,5	$\infty$	$\infty$
3	1,8	6,3	0	1,13	1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
4	$\infty$	$\infty$	1,13	0	$\infty$	4,5	$\infty$	$\infty$	$\infty$	$\infty$
5	1	5,5	1	$\infty$	0	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
6	$\infty$	$\infty$	$\infty$	4,5	$\infty$	0	1	$\infty$	$\infty$	2,25
7	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	1	0	$\infty$	1,5	$\infty$
8	$\infty$	4,5	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	0	1,29	$\infty$
9	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	1,5	1,29	0	1,13
10	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	2,25	$\infty$	$\infty$	1,13	0

Svakom sljedećom tablicom se mijenjaju vrijednosti u određenim poljima ovisno o tome jesu li trenutne vrijednosti manje ili veće od novih. Ukoliko su nove vrijednosti manje tada se one stavljaju umjesto starih vrijednosti i nove predstavljaju najkraće udaljenosti između pojedinih čvorova.

U tablici 17 prikazana je posljednja iteracija Floyd-Warshallvog algoritma, prema kojoj se vidi da su određene udaljenosti između čvorova promijenile svoju vrijednost u odnosu na početnu tablicu 15.

**Tablica 17.** Udaljenost između čvorova nakon posljednje iteracije

	1	2	3	4	5	6	7	8	9	10
1	0	4,5	1,8	2,93	1	4,5	2,5	3,29	2	3,13
2	4,5	0	6,3	7,43	5,5	8,29	7,29	4,5	5,79	6,92
3	1,8	6,3	0	1,13	1	4,5	3,5	3,29	2	3,13
4	2,93	7,43	1,13	0	2,13	4,5	4,63	4,42	3,13	4,26
5	1	5,5	1	2,13	0	3,5	2,5	2,29	1	2,13
6	4,5	8,29	4,5	4,5	3,5	0	1	3,79	2,5	2,25
7	2,5	7,29	3,5	4,63	2,5	1	0	2,79	1,5	2,63
8	3,29	4,5	3,29	4,42	2,29	3,79	2,79	0	1,29	2,42
9	2	5,79	2	3,13	1	2,5	1,5	1,29	0	1,13
10	3,13	6,92	3,13	4,26	2,13	2,25	2,63	2,42	1,13	0

Tablica 17 prikazuje najkraće udaljenosti od svih čvorova prema svim čvorovima. Uspoređujući te vrijednosti sa vrijednostima najkraćih udaljenosti kod Dijkstra algoritma, vidljivo je da se te vrijednosti podudaraju.

S obzirom na to da su rezultati najkraćih puteva, a time i ruta isti kao i kod Dijkstra algoritma, vrijednosti dobivene u poglavlju (5.1.1.) za kašnjenje i vjerojatnost blokiranja ostaju nepromijenjene.

Nakon primjene Floyd-Warshallovog algoritma za traženje najkraćeg puta u neopterećenoj mreži, u nastavku je prikazan isti postupak računanja najkraćeg puta, ali za opterećenu mrežu sa vrijednostima koje su prikazane slikom 21.

**Tablica 18.** Početna udaljenost između svih čvorova za opterećenu mrežu

	1	2	3	4	5	6	7	8	9	10
1	0	5,06	2,7	$\infty$	3	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
2	5,06	0	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	40,5	$\infty$	$\infty$
3	2,7	$\infty$	0	5,06	1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
4	$\infty$	$\infty$	5,06	0	$\infty$	4,5	$\infty$	$\infty$	$\infty$	$\infty$
5	3	$\infty$	1	$\infty$	0	$\infty$	$\infty$	$\infty$	4,5	$\infty$
6	$\infty$	$\infty$	$\infty$	4,5	$\infty$	0	1,8	$\infty$	$\infty$	20,25
7	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	1,8	0	$\infty$	4,5	$\infty$
8	$\infty$	40,5	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	0	3,86	$\infty$
9	$\infty$	$\infty$	$\infty$	$\infty$	4,5	$\infty$	4,5	3,86	0	2,53
10	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	20,25	$\infty$	$\infty$	2,53	0

Tablicom 19 je prikazana udaljenost čvorova nakon prve iteracije, i crveno označena polja su one udaljenosti koje su se promijenile, tj. manje su nego u prethodnoj tablici (tablica 18).

**Tablica 19.** Udaljenost čvorova nakon prve iteracije

	1	2	3	4	5	6	7	8	9	10
1	0	5,06	2,7	∞	3	∞	∞	∞	∞	∞
2	5,06	0	7,76	∞	8,06	∞	∞	40,5	∞	∞
3	2,7	7,76	0	5,06	1	∞	∞	∞	∞	∞
4	∞	∞	5,06	0	∞	4,5	∞	∞	∞	∞
5	3	8,06	1	∞	0	∞	∞	∞	4,5	∞
6	∞	∞	∞	4,5	∞	0	1,8	∞	∞	20,25
7	∞	∞	∞	∞	∞	1,8	0	∞	4,5	∞
8	∞	40,5	∞	∞	∞	∞	∞	0	3,86	∞
9	∞	∞	∞	∞	4,5	∞	4,5	3,86	0	2,53
10	∞	∞	∞	∞	∞	20,25	∞	∞	2,53	0

**Tablica 20.** Udaljenost između čvorova nakon posljednje iteracije u opterećenoj mreži

	1	2	3	4	5	6	7	8	9	10
1	0	5,06	2,7	7,76	3	12,26	12	11,36	7,5	10,03
2	5,06	0	7,76	12,82	8,06	17,32	17,06	16,42	12,56	15,09
3	2,7	7,76	0	5,06	1	9,56	10	9,36	5,5	8,03
4	7,76	12,82	5,06	0	6,06	4,5	6,3	14,42	10,56	13,09
5	3	8,06	1	6,06	0	10,56	9	8,36	4,5	7,03
6	12,26	17,32	9,56	4,5	10,56	0	1,8	10,16	6,3	8,83
7	12	17,06	10	6,3	9	1,8	0	8,36	4,5	7,03
8	11,36	16,42	9,36	14,42	8,36	10,16	8,36	0	3,86	6,39
9	7,5	12,56	5,5	10,56	4,5	6,3	4,5	3,86	0	2,53
10	10,03	15,09	8,03	13,09	7,03	8,83	7,03	6,39	2,53	0

Podaci (najkraće udaljenosti dobiveni pomoću Floyd-Warshallovog algoritma) koji su prikazani tablicom 20, a odnose se na opterećenu mrežu odgovaraju onim udaljenostima koji su se dobili kod Dijkstra algoritma iz čega proizlazi i da su rute kojim će se usmjeravti paketi identične.

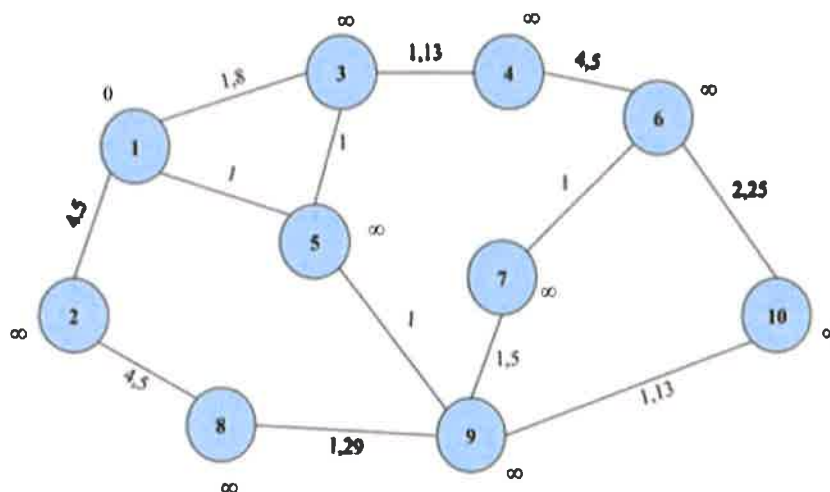
S obzirom na to da su rezultati najkraćih puteva, a time i ruta isti kao i kod Dijkstra algoritma, vrijednosti dobivene u poglavlju (5.1.1.) za kašnjenje i vjerojatnost blokiranja ostaju nepromijenjene u odnosu na rezultate prikazane tablicama 11 i 14.

### 5.1.3. Računanje najkraćeg puta pomoću Bellman-Ford algoritma

Bellman-Ford algoritam pronalazi najkraće putove od početne točke s ograničenjem na jednu spojnicu, a nakon toga se svakom sljedećom iteracijom broj povećava za vrijednost 1, dok se ne dođe do odredišnog čvora. Broj iteracija koje se odvijaju u pojedinom grafu ovisi o broju čvorova u samom grafu, pa tako na primjeru sa slike 21, sa koje se gledaju vrijednosti

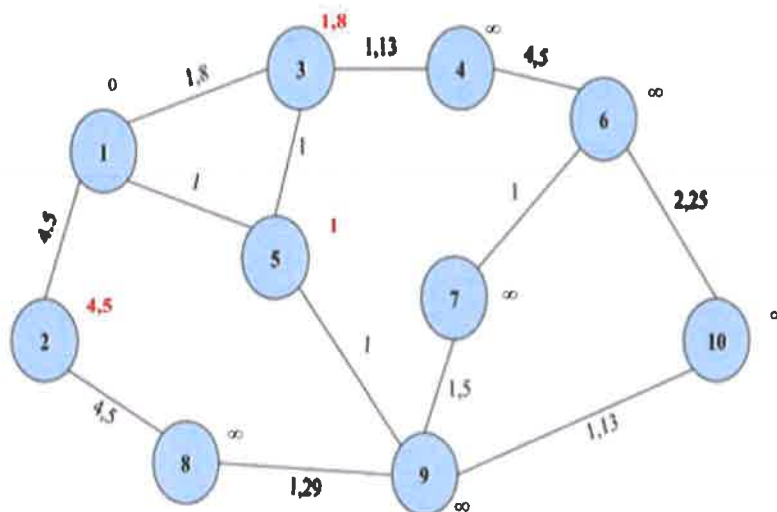
prilikom računanja najkraćeg puta, se vidi da se sastoji od 10 čvorova, pa je tako najveći broj iteracija 9.

Prvi korak kod Bellman-Ford algoritma, prikazan slikom 22., je uzeti jedan čvor da bude početni i njegovu vrijednost postaviti na 0, a vrijednosti ostalih čvorova u prvom koraku postaviti u beskonačnost.



Slika 22. Prvi korak Bellman-Ford algoritma

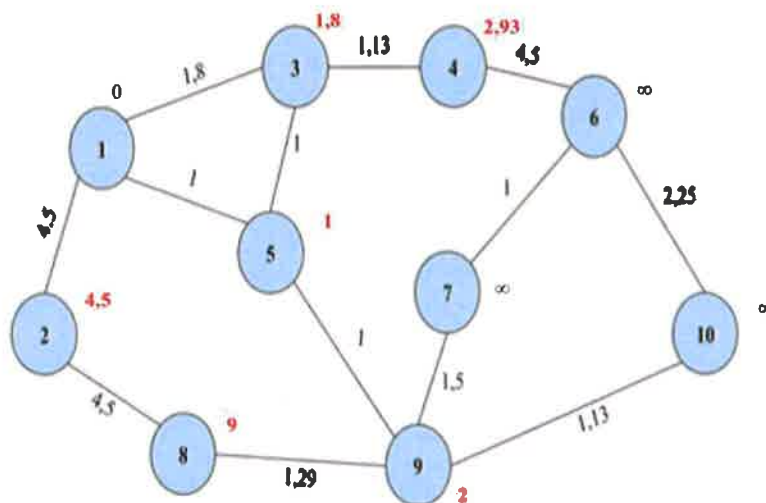
Sljedeći korak je označiti udaljenosti od čvora 1 do njegovih susjednih čvorova s kojima su povezani spojnicom, a u ovom slučaju bi to bili čvorovi 2, 3 i 5, a za ostale čvorove ostaje beskonačna udaljenost, pošto nisu izravno povezani sa čvorom 1. Taj korak je prikazan slikom 23.



Slika 23. Prvi korak traženja najkraće udaljenosti

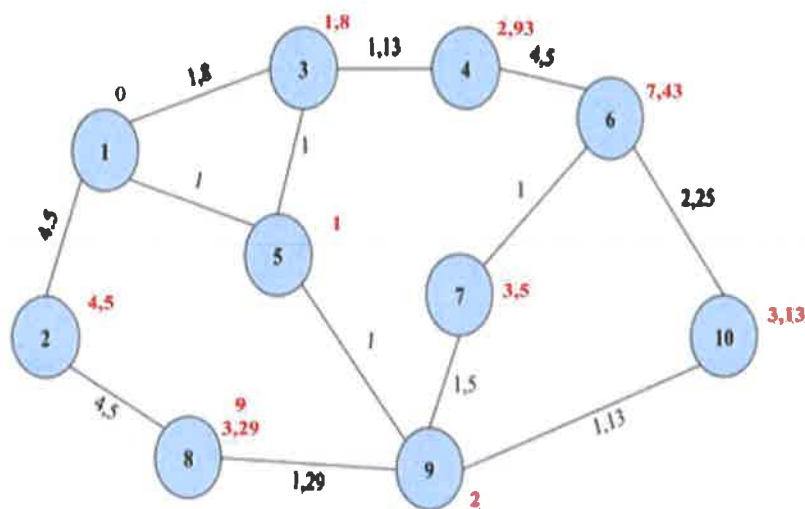
Prema slici 23, novi čvorovi preko kojih se može tražiti udaljenosti do sljedećih susjednih su 2, 3 i 5, i njihovi susjedni do kojih se traži sljedeća najkraća udaljenost je 4, 8 i 9.

Nove udaljenosti i promjene su prikazane slikom 24.



Slika 24. Nove udaljenosti do označenih čvorova

Novi označeni čvorovi su 4, 8 i 9, a njihove nove udaljenosti su označene crvenom bojom na slici 24. U sljedećem koraku se mogu označiti i naći udaljenosti do preostalih čvorova, a to su 6, 7 i 10, i istovremno se izvršava nova provjera udaljenosti do označenih čvorova, s obzirom da se do određenih čvorova može doći pomoću novooznačenih, a te vrijednosti su prikazane na slici 25.



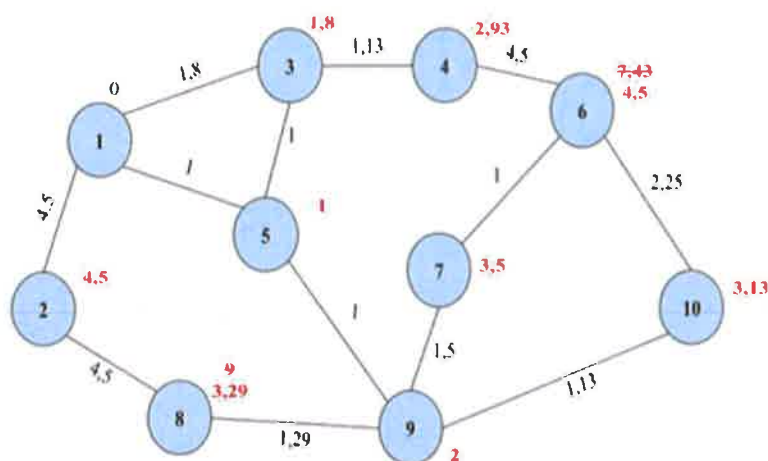
Slika 25. Nove udaljenosti prema svim čvorovima

Na slici 25 prikazane su nove vrijednosti udaljenosti od čvora 1 prema novooznačenim čvorovima 6, 7 i 10, ali također se pri tome provjerava postoji li neka nova kraća udaljenost do označenih čvorova preko novooznačenih. Za zadanu mrežu to znači da se provjerava postoji li neka kraća udaljenost od postojećih udaljenosti do čvorova 2, 3, 4, 5, 8 i 9.

Prema slici 25, vidljivo da je se udaljenost smanjila samo na čvoru 8, i ta najkraća udaljenost nema utjecaja na ostale čvorove, što znači da ostale udaljenosti ostaju iste, jer ne postoji kraća udaljenost za njih od postojećih. Do čvora 8 nova najkraća udaljenost ide preko čvorova 3 i 9, i nova udaljenost je prikazana na slici.

Posljednji korak provjere najkraćih udaljenosti pomoću Bellman-Ford algoritma je provjera udaljenosti do svih označenih čvorova, tj. postoji li neka nova najkraća udaljenost do njih. Ako se iz čvora 8 ide do čvorova 3 i 5, to je veća udaljenost od postojećih, tako da postojeće ostaju nepromjenjene. U čvor 4 se može doći preko čvorova 3 i 6, ali put iz čvora 3 je i dalje kraći pa ta udaljenosti također ostaje nepromjenjena. U čvor 6 se može doći preko čvora 4, 7 i 10, i najkraća udaljenost je preko čvora 7, pa se udaljenost na čvoru 6 izmjeni na novu najkraću. U čvor 10 se može doći preko čvorova 6 i 9, a kraći put je preko čvora 9, tako da i ta udaljenost ostaje nepromjenjena.

Sve izmjene i konačni rezultati najkraćih udaljenosti neopterećene mreže su prikazani na slici 26.

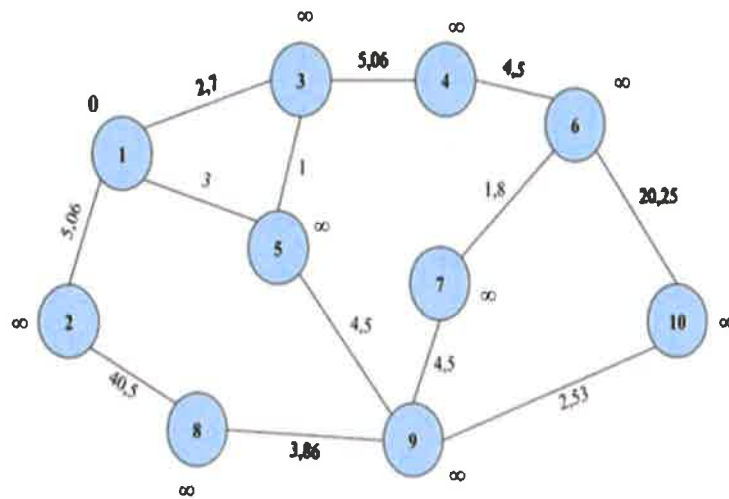


**Slika 26.** Najkraće udaljenosti neopterećene mreže

Prema izračunima i rezultatima, vidljivo je da konačne najkraće udaljenosti neopterećene mreže do svih čvorova Bellman-Ford algoritma odgovaraju rezultatima i najkraćim udaljenostima koje su se dobile pomoću Dijkstra algoritma. Najčešće korištena poveznica do

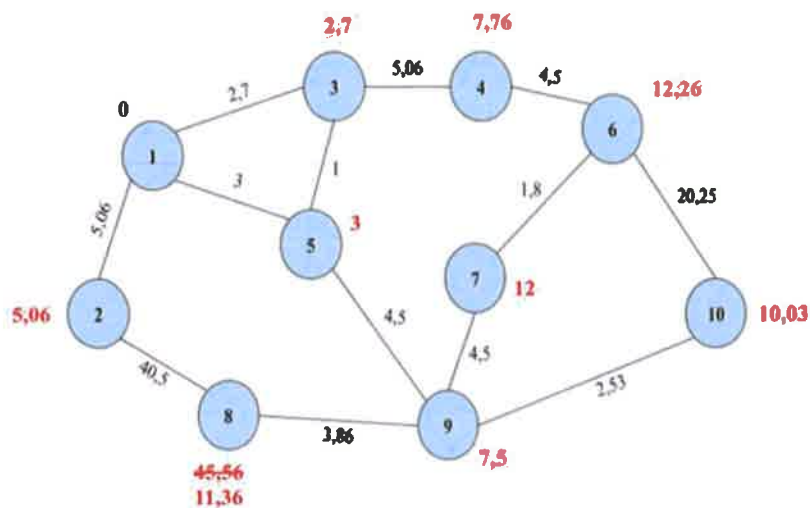
čvorova, koja je rezultat najkraćih udaljenosti, je 1-5-9, čime odgovara rezultatima prethodno dva obrađena algoritma, stoga se daljnji tijek računanja kašnjenja i vjerojatnosti blokiranja ne mora ponovno provoditi i računati. Prikazani rezultati su udaljenosti za neopterećenu mrežu, stoga je isti postupak potrebno provesti i za opterećenu mrežu.

Za opterećenu mrežu postupak određivanja najkraće udaljenosti je isti kao i kod neopterećene mreže, svi koraci su isti, stoga se za opterećenu mrežu neće prikazivati svi koraci nego samo početni i završni korak.



Slika 27. Početne udaljenosti opterećene mreže

Nakon što se u početnom koraku prvo postave beskonačne udaljenosti prema svakom čvoru, nakon toga slijedi obilježavanje udaljenosti do susjednih povezanih čvorova, i nakon toga redom označavanje svih ostalih dok se ne dođe do posljednje označenog čvora.



Slika 28. Najkraće udaljenosti opterećene mreže

Posljednji korak provjere najkraćih udaljenosti pomoću Bellman-Ford algoritma je provjera udaljenosti do svih označenih čvorova, tj. dali postoji neka nova najkraća udaljenost do njih. Prema slici 28., vidljivo je da se jedina promjena dogodila za čvor 8. Nova najkraća udaljenost za čvor 8 se dobila preko čvorova 5 i 9. Ako se iz čvora 8 ide do čvorova 3 i 5, to je veća udaljenost od postojećih, tako da postojeće ostaju nepromjenjene. U čvor 4 se može doći preko čvorova 3 i 6, ali put iz čvora 3 je i dalje kraći pa ta udaljenosti također ostaje nepromjenjena. U čvor 6 se može doći preko čvora 4, 7 i 10, i najkraća udaljenost je preko čvora 4, pa je i ta udaljenost nepromjenjiva. U čvor 10 se može doći preko čvorova 6 i 9, a kraći put je preko čvora 9, tako da i ta udaljenost ostaje nepromjenjena.

Pošto i rezultati i udaljenosti, dobiveni pomoću Bellman-Ford algoritma, opterećene mreže odgovaraju rezultatima prethodno dva obrađena algoritma, također se kašnjenje i vjerojatnost blokiranja ne mora ponovno računati, pošto su isti rezultati.



## 6. Usporedba performansi mreže za različite algoritme usmjeravanja koji se temelje na najkraćem putu

Dijkstra i Bellman-Ford su algoritmi koji određuju najkraće putove za mreže koje se sastoje od više čvorova, povezane su spojnica koje imaju određenu vrijednost i pomoću tih opterećenja na tim spojnica se računa najkraći put. Oba ova algoritma određuju najkraću udaljenost svakog čvora od vrha jednog izvora. Oba spomenuta algoritma koriste metode izračuna relaksacije, što zapravo znači da koriste postupak pronalaženja najkraćeg puta modificiranjem vrijednosti  $D(i)$  tijekom postupka prelaska sa jednog čvora na druge putem spojnica kojima su povezani. Koriste se za pronalazak optimalnog rješenja najkraćeg puta za posmatranu mrežu, ali Bellman-Ford ima velike redundancije i nižu učinkovitost. Osim spomenute razlike i negativnosti za Bellman-Ford algoritam, postoje još neke ključne razlike među ova dva algoritma, [29].

Neke od najvažnijih razlika su prikazane tablicom 21.

**Tablica 21.** Komparacija numeričkih i eksperimentalnih svojstava

	Dijkstra algoritam	Bellman-Ford algoritam	Floyd-Warshall algoritam
prostorna zahtjevanost	$O(M)$	$O(M)$	$O(N^2)$
vremenska zahtjevnost	$O(N^2)$	$O(MN)$	$O(N^3)$
troškovi na grana su negativni	NE	DA	DA
<b>M – broj bridova/spojnica</b> <b>N – broj čvorova</b>			

Izvor: [29]

Još jedna važna karakteristika prema kojoj se razlikuju Dijkstra i Floyd-Warshallov algoritam je njihov rad prema distribuiranim sustavima. Za razliku od Dijkstra algoritma, Floyd-Warshallov algoritam se može implementirati u distribuirani sustav, što ga čini pogodnim za podatkovne strukture.

Dijkstra algoritam se može koristiti samo za slučaje pronalazjenja najkraćeg puta s jednim izvorom, dok je Floyd-Warshallov algoritam pogodan za pronalazjenje najkraćeg puta između bilo koje dvije točke. Upravo je zbog toga prikladan za pronalazjenje najkraćeg puta među svim čvorovima ili u malom opsegu podataka, [30].

Kao što je prethodno spomenuto i objašnjeno u poglavlju 5, Dijkstra, Floyd-Warshallov i Bellman-Ford su algoritmi koji su okarakterizirani različitim svojstvima prema kojima se oni međusobno razlikuju, no u konačnici bi svi trebali imati iste rezultate. Sva tri algoritma na različit način traže najkraće putove u mreži, stoga neki od njih su zahtjevniji i obuhvaćaju više koraka da se dobije konačni rezultat, dok neki od njih su vrlo jednostavni i lako se dođe do krajnjeg rezultata.

U poglavlju 5 fokus je bio, prema primjeru sa slike 20, pronaći najkraći put i na tome putu odrediti kašnjenje i vjerojatnost blokiranja za sva tri navedena algoritma.

Sam postupak određivanja najkraćeg puta, od navedena tri algoritma, najzahtjevniji je za Dijkstra i Floyd-Warshallov algoritam. Oba spomenuta algoritma, u procesu pronalaska najkraćeg puta, zahtjevaju kreiranje 10 tablica (pošto se mreža sa slike 20 sastoji od toliko čvorova), i svaka tablica se treba popuniti prema pravilima karakterističnim za pojedini algoritam. Svaka tablica mijenja vrijednosti od pojedinih čvorova, i na samom kraju u posljednjoj tablici su konačne vrijednosti. Iz posljednje tablice se isčitaju vrijednosti koje su zapravo vrijednosti najkraćeg puta u mreži.

Bellman-Ford algoritam je dosta jednostavniji po tom pitanju. Na primjeru mreže za slike 20, ovaj algoritam je već nakon 4 iteracije došao do krajnjeg rezultata, tj. najkraćeg puta, u odnosu na ostala dva algoritma, sa kojima se mora popuniti svih 10 tablica da se dobije isti rezultat.

Kako bi se provjerila točnost i vjerodostojnog najkraćeg puta za sva tri algoritma, usporedi se krajnji rezultat, tj. dali sva tri algoritma daju isti rezultat (najkraći put).

Rezultat kod spomenutih algoritama bi bila različita, da se promatrana mreža na primjer sastoji od negativnih spojnica, što u ovome slučaju nije riječ. Sve spojnice imaju pozitivne vrijednosti. Prema poglavlju 5, kašnjenje i vjerojatnost blokiranja se računala samo za čvor 1, tj. najkraće putove za taj čvor, a isti bi postupak bio i za ostale čvorove, stoga se samo obradio jedan čvor samo radi prikaza postupka.

Da su rješenja najkraćih putova za čvor 1 za sva tri algoritma bila različita, tada bi se i kašnjenje i vjerojatnost blokiranja moralo računati posebno za sva tri, no kako su rješenja bila ista, tada ta rješenja vrijede za sve obrađene algoritme.

Zaključno, Dijkstra algoritam primjenjuje rijetki graf (graf u kojem je broj spojnica mnogo manji od maksimalnog). U stvarnoj primjeni algoritam je uvijek optimiziran, poput „optimizacije hrpe (hrpa je najjednostavnija struktura podataka)“. Vremenska zahtjevnost algoritma tada se smanjuje na  $n * \log n$ . Bellman-Ford algoritam je neučinkovit, ali ga je jednostavno implementirati. Kada je problemom koji treba riješiti mreža s velikim brojem čvorova i bridova/spojnica, Floyd-Warshallov algoritam je najsporiji i troši suviše prostora, [29].

Zaključno sa svim navedenim prednostima i manama sva tri algoritma, najbolji izbor za računanje najkraćeg puta ne ovisi samo o njihovim svojstvima, nego i o samoj mreži (broj čvorova i spojnica) i njenim svojstvima. Pri samom izboru se trebaju sagledati sva svojstva svih algoritama, i donijeti odluka o tome koji bi algoritam bio najbolji izbor.

## 7. Zaključak

Paketske mreže, tj. mreže koje koriste paketski način prijenosa podataka kroz mrežu, su najzastupljeniji način prijenosa podataka od izvorišta do odredišta. Zamjenjuje prethodni način prijenosa podataka kroz mrežu, a to je komutacija kanala, jer je paketski način puno efikasniji i brži. Većina telekomunikacijskih sustava danas implementira i ulaže u razvijanje i poboljšanje ovakvog načina prijenosa podataka, kako bi se omogućilo što bolji i pouzdaniji rad mreže, i na samom kraju zadovoljstvo korisnika.

Internet danas omogućuje ljudima diljem svijeta brz pristup i slanje velike količine podataka. Shodno tome, u paketskim mrežama se koriste različite metode i alati za što pouzdaniji i brži prijenos paketa (informacija, podataka) od izvora do odredišta kako bi se moglo upravljati i manipulirati tom velikom količinom podataka. Najzastupljeniji način je korištenje odgovarajućih algoritama koji se temelje na usmjeravanju paketa kroz mrežu najkraćim putem.

Glavna odrednica prilikom odabira odgovarajućeg algoritma je određivanje kompleksnosti mreže i na osnovu njenih odrednica se odabire algoritam koji najbolje odgovara njenim zahtjevima i svojstvima. Analizom mreže se odabire algoritam koji će dati pouzdani prijenos putem najkraćeg puta.

Temeljna svrha ovoga rada je prikazati pregled najosnovnijih algoritama usmjeravanja koji se temelje na najkraćem putu, prikazati osnovni rad i karakteristike svih algoritama, njihove prednosti i mane, te objasniti sam rad i funkcioniranje algoritama na primjeru manje mreže koja se sastoji od određenog broja čvorova i spojnica.

Nadalje, na temelju te mreže određeni su najkraćeg putevi pomoću tri osnovna algoritma (Dijkstra, Floyd-Warshall i Bellman-Ford algoritam), i na osnovu tih rezultata omogućeno je računanje kašnjenja i vjerojatnosti blokiranja za neopterećenu i opterećenu mrežu za sva tri algoritma. Rezultati kašnjenja i vjerojatnosti blokiranja je različito za opterećenu i neopterećenu mrežu, kao što se također moglo vidjeti da ono i ne ovisi o broju čvorova i spojnica kroz koje se prolazi, nego ovisi o tome koliko je opterećenje pojedine spojnice. Zato je kašnjenje i vjerojatnost blokiranja manje na nekim spojnica koje su na putovima sadržavale više čvorova od onih koje su sadržavale manji broj čvorova. Na posljetku su uspoređeni dobiveni rezultati i sva tri algoritma na temelju kojih se mogu vidjeti pozitivne i negativne strane sva tri algoritma.

Nakog provedenog istraživanja i računanja se dolazi do zaključka da sa rastom mreže, tj. čvorova i spojnica raste i vrijeme same izvedbe algoritma. Danas se zbog toga koriste puno napredniji algoritmi za pronalaženje najkraćih puteva u Internetu, jer se mreže na internetu sastoje od par stotina milijuna čvorova, pa se zbog toga i trebaju konstantno unaprjeđivati algoritmi.

## Literatura

- [1] Mrvelj Š. *Predmet proučavanja i temeljni pojmovi*. [Prezentacija], Tehnologija telekomunikacijskog prometa I. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2019.
- [2] Lovrek, I. *Telekomunikacijska tehnologija i specifičnosti telekomunikacijskog tržišta*. Zagreb: Element; 2007. p. 1-36.
- [3] Leksikografski zavod Miroslav Krleža. *Komunikacijska mreža*. Dostupno na: <https://www.enciklopedija.hr/natuknica.aspx?id=42221> [Pristupljeno: 4. Travanj 2020].
- [4] Fox P. *Internet routing protocol*. [Prezentacija], Khan Academy. [Pristupljeno: 9. Svibanj 2020].
- [5] Ross K., Kurose J. *Computer Networking: A Top Down Approach*. Brooklyn: Pearson. 3th/5th edition; 2009.
- [6] Visković I. *Temeljne strukture komutacijskih sustava – tipovi komutacijskih sustava*. [Prezentacija] Komutacija i upravljanje u telekomunikacijskoj mreži. Sveučilišni studijski centar za stručne studije, Split. Travanj 2011.
- [7] Mrvelj Š. *Komutacijski i transmisijski sustavi podatkovne mreže*. [Prezentacija] Tehnologija telekomunikacijskog prometa I. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2019.
- [8] Mrvelj Š. *Višeprotokolno komutiranje labela*. [Prezentacija] Tehnologija telekomunikacijskog prometa II. Fakultet prometnih znanosti Sveučilišta u Zagrebu, 2020.
- [9] Community.fs.com. *IP vs MPLS: What's the difference?* Dostupno na: <https://community.fs.com/blog/vpn-vs-mpls-difference.html> [Pristupljeno: 7. Lipanj 2020.]

- [10] Uttamkumar. *IPv6 and IPv4 Headers*. Dostupno na: <http://www.startnetworks.info/2011/08/ipv6-and-ipv4-headers.html> [Pristupljeno: 23. Travanj 2020]
- [11] Mexico Documents. *Komutacija paketa*. [Prezentacija] Dostupno na: <https://vdocuments.mx/komutacija-paketa.html> [Pristupljeno: 9. Travanj 2020].
- [12] Mujarić E. *Usmjeravanje i usmjerivački protokoli*. Dostupno na: <https://sysportal.carnet.hr/node/650> [Pristupljeno: 9. Travanj 2020].
- [13] Popovski H, Rako I. *IPv6 – protokol nove generacije*. Dostupno na: <https://www.srce.unizg.hr/search/node/IPv6%20%E2%80%93%20protokol%20nove%20generacije>. [Pristupljeno: 9. Svibanj 2020].
- [14] CARNet, CERT, LS&S. *Hrvatska akademska i istraživačka mreža: Osnovne mrežnog usmjeravanja*. 2007. Dostupno na: <https://www.cert.hr/wp-content/uploads/2019/04/CCERT-PUBDOC-2007-02-183.pdf> [Pristupljeno: 10. Travanj 2020].
- [15] IT Geared. *What is a Network Router?*. Dostupno na: <http://www.itgeared.com/articles/1352-network-router-overview/> [Pristupljeno: 10. Travanj 2020].
- [16] Computer Hope. *Routing algorithm*. Dostupno na: <https://www.computerhope.com/jargon/r/routing-algorithm.htm> [Pristupljeno: 5. Travanj 2020].
- [17] Zhang M. *On the State of the Inter-domain and Intra-domain Routing Security*. Dostupno na: [https://pdfs.semanticscholar.org/ba8f/07383aba2922f01d9d93588701fd60fd2a91.pdf?\\_ga=2.11641716.1501553573.1586771093-828639845.1586771093](https://pdfs.semanticscholar.org/ba8f/07383aba2922f01d9d93588701fd60fd2a91.pdf?_ga=2.11641716.1501553573.1586771093-828639845.1586771093) [Pristupljeno: 5. Travanj 2020].
- [18] Mujarić E. *OSPF protokol*. Dostupno na: <https://sysportal.carnet.hr/node/652> [Pristupljeno: 14. Travanj 2020].

- [19] Balchunas A. *Open Shortest Path First*. Dostupno na: <http://www.routeralley.com/guides/ospf.pdf> [Pristupljeno: 14. Travanj 2020].
- [20] Singhal A. *Dijkstra Algorithm | Example | Time Complexity*. Dostupno na: <https://www.gatevidyalay.com/dijkstras-algorithm-shortest-path-algorithm/> [Pristupljeno: 17. Travanj 2020].
- [21] Havaš L, Keček D, Knez K. *Usporedba i primjena „Distance Vector“ i „Link State“ mrežnih protokola*. 2013, Tehnički glasnik 7, 108-115. Dostupno na: <https://hrcak.srce.hr/105589> [Pristupljeno: 17. Travanj 2020].
- [22] Programiz.com. *Floyd-Warshall Algorithm*. Dostupno na: <https://www.programiz.com/dsa/floyd-warshall-algorithm> [Pristupljeno: 17. Travanj 2020].
- [23] Programin.com. *Bellman Ford's Algorithm*. Dostupno na: <https://www.programiz.com/dsa/bellman-ford-algorithm> [Pristupljeno: 17. Travanj 2020].
- [24] ViaViSolutions.com. *How to Measure Network Performance*. Dostupno na: <https://www.viavisolutions.com/en-us/how-measure-network-performance> [Pristupljeno: 22. Travanj 2020].
- [25] Mrvelj Š. *Ciljevi razine usluge QoS/GoS/NP*. [Prezentacija] Tehnologija telekomunikacijskog prometa I. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2018.
- [26] Manger R. *Mjerenje performansi mreže*. [Prezentacija] PMF – Matematički odsjek, Sveučilište u Zagrebu. 23. Travanj 2020.
- [27] Matulin M. *Sustavi posluživanja s čekanjem*. [Prezentacija] Tehnologija telekomunikacijskog prometa I. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2018.
- [28] Matulin M. *Planiranje mobilne ćelijske mreže*. [Prezentacija] Tehnologija telekomunikacijskog prometa I. Fakultet prometnih znanosti Sveučilišta u Zagrebu. 2018.



## Popis slika

<b>Slika 1.</b> Prikaz komunikacijske mreže .....	4
<b>Slika 2.</b> Poopćeni prikaz funkcije čvora .....	5
<b>Slika 3.</b> Prijenos informacije kroz mrežu komutacijom paketa .....	7
<b>Slika 4.</b> Struktura paketa virtualnog kanala .....	9
<b>Slika 5.</b> Struktura datagrama .....	10
<b>Slika 6.</b> Usporedba zaglavlja IPv4 i IPv6 .....	12
<b>Slika 7.</b> Izgled zaglavlja IP paketa .....	13
<b>Slika 8.</b> Prefiks IP adrese .....	14
<b>Slika 9.</b> Uloge središnjih usmjernika pri prosljeđivanju paketa .....	16
<b>Slika 10.</b> Izgled tablice usmjeravanja .....	17
<b>Slika 11.</b> Usmjeravanje unutar i među domenama .....	23
<b>Slika 12.</b> OSPF područja mreže .....	26
<b>Slika 13.</b> Zaglavlje OSPF okvira .....	27
<b>Slika 14.</b> Računanje najkraćeg puta korištenjem Dijkstrinog algoritma .....	29
<b>Slika 15.</b> Dijagram aktivnosti Dijkstrinog algoritma usmjeravanja .....	30
<b>Slika 16.</b> Graf nad kojim je izvršen Floyd-Warshallow algoritam .....	32
<b>Slika 17.</b> Primjer usmjerenog grafa s negativnim vrijednostima .....	35
<b>Slika 18.</b> Dijagram aktivnosti Bellman-Ford algoritma .....	36
<b>Slika 19.</b> Topologija zadane mreže s kapacitetima spojnica .....	40
<b>Slika 20.</b> Prikaz mreže sa troškovima spojnica .....	41
<b>Slika 21.</b> Topologija zadane mreže s troškovima spojnica za opterećenu mrežu .....	49
<b>Slika 22.</b> Prvi korak Bellman-Ford algoritma .....	60
<b>Slika 23.</b> Prvi korak traženja najkraće udaljenosti .....	60
<b>Slika 24.</b> Nove udaljenosti do označenih čvorova .....	61
<b>Slika 25.</b> Nove udaljenosti prema svim čvorovima .....	61
<b>Slika 26.</b> Najkraće udaljenosti neopterećene mreže .....	62
<b>Slika 27.</b> Početne udaljenosti opterećene mreže .....	63
<b>Slika 28.</b> Najkraće udaljenosti opterećene mreže .....	63

## Popis tablica

<b>Tablica 1.</b> Usporedba najvažnijih parametara za IP i MPLS usmjeravanje .....	8
<b>Tablica 2.</b> Finalna tablica Floyd-Warshalla algoritma .....	32
<b>Tablica 3.</b> Prva iteracija Bellman-Fordovog algoritma za navedeni graf .....	35
<b>Tablica 4.</b> Redoslijed označavanja čvorova u postupku pronalaženja najkraćeg puta za čvor 1 .....	42
<b>Tablica 5.</b> Popis najkraćih putova za sve parove čvorova u zadanoj neopterećenoj mreži .....	43
<b>Tablica 6.</b> Redoslijed označavanja čvorova u postupku pronalaženja najkraćeg puta za čvor 2 .....	44
<b>Tablica 7.</b> Redoslijed označavanja čvorova u postupku pronalaženja najkraćeg puta .....	45
<b>Tablica 8.</b> Kašnjenje paketa na svim najkraćim putovima iz čvora 1 za neopterećenu mrežu .....	47
<b>Tablica 9.</b> Opterećenja spojnica u mreži i troškovi spojnica .....	48
<b>Tablica 10.</b> Popis najkraćih putova za sve parove čvorova u zadanoj opterećenoj mreži .....	49
<b>Tablica 11.</b> Kašnjenje paketa na najkraćim putovima iz čvora 1 za opterećenu mrežu .....	51
<b>Tablica 12.</b> Vjerojatnosti blokiranja na najkraćim putovima u neopterećenoj mreži .....	53
<b>Tablica 13.</b> Vjerojatnost blokiranja za opterećenu mrežu .....	54
<b>Tablica 14.</b> Vjerojatnost blokiranja na najkraćim putovima u opterećenoj mreži .....	55
<b>Tablica 15.</b> Početna udaljenost između svih čvorova za neopterećenu mrežu .....	56
<b>Tablica 16.</b> Udaljenost čvorova nakon prve iteracije ( $k=1$ ) .....	57
<b>Tablica 17.</b> Udaljenost između čvorova nakon posljednje iteracije .....	58
<b>Tablica 18.</b> Početna udaljenost između svih čvorova za opterećenu mrežu .....	58
<b>Tablica 19.</b> Udaljenost čvorova nakon prve iteracije .....	59
<b>Tablica 20.</b> Udaljenost između čvorova nakon posljednje iteracije u opterećenoj mreži .....	59
<b>Tablica 21.</b> Komparacija numeričkih i eksperimentalnih svojstava .....	65



Sveučilište u Zagrebu  
Fakultet prometnih znanosti  
10000 Zagreb  
Vukelićeva 4

## IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj \_\_\_\_\_ diplomski rad  
isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na  
objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz  
necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako 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 **Usmjeravanje u mrežama s komutacijom paketa temeljeno na**  
**modelu najkraćeg puta**

na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom  
repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

U Zagrebu, \_\_\_\_\_ 16/09/2020 \_\_\_\_\_

Student/ica:

*Nikolina Pušić*  
\_\_\_\_\_  
(potpis)