

UNIVERZITET SINGIDUNUM
BEOGRAD
DEPARTMAN ZA POSLEDIPLOMSKE STUDIJE I
MEĐUNARODNU SARADNJU

DOKTORSKA DISERTACIJA

JEDNO UNAPREĐENJE OLSR RUTING
PROTOKOLA U MOBILNIM AD HOC MREŽAMA

MENTOR

prof. dr Mladen Veinović

KANDIDAT

Dejan Tepšić, master

BEOGRAD, 2013.

SAŽETAK

Mobilne ad hoc mreže (MANET) su samokonfigurišuće mreže čvorova povezanih bežičnim putem bez bilo kakvog oblika centralizovanog upravljanja. Ove mreže su trenutno jedan od najvažnijih istraživačkih predmeta zbog velikog broja različitih primena (vojna, spasilačka, itd.). U MANET mrežama svaki čvor istovremeno je terminal i ruter. Stoga, svaki čvor mora biti sposoban da prosleđuje pakete do ostalih čvorova. Usled mobilnosti čvorova topologije kod ovih mreža sklone su čestim promenama. Kako bi prevazišli ove probleme posebni ruting protokoli su razvijeni za MANET mreže. Tradicionalni ruting protokoli razvijeni za ožičene mreže ne mogu efikasno raditi u MANET mrežama.

Cilj ovog rada je istraživanje trenutnog stanja postojećih ruting protokola u MANET mrežama i poređenje različitih pristupa u njihovoj izvedbi. Postoje tri glavne kategorije ruting protokola za MANET mreže: reaktivni, proaktivni i hibridni. Proučavanjem prednosti i nedostataka svakog od njih, novi ruting protokol je predložen. Novi protokol nazvan je OLSR-BF. To je proaktivni ruting protokol najvećim delom zasnovan na postojećem OLSR ruting protokolu. Modifikacija unešena u novi ruting protokol zasniva se prvenstveno na upotrebi Belman-Fordovog algoritma prilikom određivanja najkraćeg rastojanja između čvorova unutar jedne MANET mreže.

Kako bi se proverila izvedba i performanse, novi ruting protokol kvalitativno je upoređen sa postojećim AODV, DSDV i OLSR ruting protokolima.

Ključne reči: AODV, bežične mreže, Belman-Ford, DSDV, MANET, mobilne ad hoc mreže, OLSR, ruting protokoli.

ABSTRACT

Mobile ad hoc networks (MANETs) are self-configuring networks of nodes connected via wireless without any form of centralized administration. This kind of networks is currently one of the most important research subjects, due to the huge variety of applications (military, emergency, etc.). In MANETs each node acts both as host and as router, thus it must be capable of forwarding packets to other nodes. Because of the node mobility topologies of these networks change frequently. To solve these problems, special routing protocols for MANETs are needed. Traditional routing protocols used in wired networks cannot work efficiently in MANETs.

The objective of this thesis is to research the current state of the art of existing routing protocols for MANETs, and compare different approaches. There are three main classes of routing protocols for MANETs: reactive, proactive and hybrid. By studying advantages and disadvantages of each one, a new routing protocol is proposed. The new scheme called OLSR-BF is proactive routing protocol mostly based on the existing OLSR routing protocol. Modification entered into a new routing protocol is based primarily on the use of Bellman-Ford algorithm in determining the shortest distance between nodes within a MANET network.

In order to validate the design and performance of the new routing protocol it is qualitatively compared with existing AODV, DSDV and OLSR routing protocols.

Keywords: AODV, Bellman-Ford, DSDV, MANET, mobile ad hoc networks, OLSR, routing protocols, wireless networks.

INDEKS SLIKA

1. Slika 1.1. Bežična mreža sa baznom stanicom (IEEE 802.11 pristupna tačka)
2. Slika 1.2. Ad hoc mreža sastavljena od uređaja povezanih bežičnim putem
3. Slika 1.3. Proces pronalaženja odredišnog čvora u MANET mreži
4. Slika 3.1. Primeri grafova
5. Slika 3.2. Primeri nepovezanih grafova
6. Slika 3.3. Primer grafa sa težinama grana
7. Slika 3.4. Minimalno razapinjujuće stablo
8. Slika 3.5. Prikaz izvođenja Primovog algoritma
9. Slika 3.6. Prikaz izvođenja Kruskalovog algoritma
10. Slika 3.7. Najkraće udaljenosti od čvora A
11. Slika 3.8. Prikaz izvođenja Dijkstrinog algoritma
12. Slika 3.9. Graf za prikaz Flojd-Voršalovog algoritma
13. Slika 3.10. Model mreže komunikacionih čvorova prikazan grafom
14. Slika 3.11. Određivanje maksimalnog protoka kroz mrežu
15. Slika 4.1. Ad hoc mreža sastavljena od uređaja povezanih bežičnim putem
16. Slika 4.2. OSI referentni model i poređenje sa MANET (IEEE 802.11) protokolima
17. Slika 4.3. Logička struktura fizičkog sloja
18. Slika 4.4. Primer problema skrivenog terminala u MANET mreži (čvor C je skriven od čvora A)
19. Slika 4.5. Primer problema izloženog terminala u MANET mreži (čvor C je izložen ka čvoru B)
20. Slika 4.6. Ad hoc mreža sa direkcionim antenama
21. Slika 4.7. Glavni izazovi koji utiču na performanse i dizajn mobilnih ad hoc mreža
22. Slika 5.1. Karakteristike ruting protokola u mobilnim ad hoc mrežama
23. Slika 5.2. Klasifikacija MANET ruting protokola zasnovana na načinu prosleđivanja paketa

- 24. Slika 5.3. **Klasifikacija MANET ruting protokola**
- 25. Slika 5.4. **DSDV proces ažuriranja putanje**
- 26. Slika 5.5. **CGSR metod otkrivanja ruta**
- 27. Slika 5.6. **Format pozdravne poruke kod OLSR ruting protokola**
- 28. Slika 5.7. **Format poruke kontrole topologije kod OLSR ruting protokola**
- 29. Slika 5.8. **Proces otkrivanja i održavanja rute kod AODV ruting protokola**
- 30. Slika 5.9. **Proces otkrivanja rute kod DSR ruting protokola**
- 31. Slika 5.10 **Primer za predstavljanje DYMO ruting protokola**
- 32. Slika 5.11. **Primer ZRP zone izvornog čvora sa faktorom $\rho = 2$**
- 33. Slika 5.12. **Arhitektura ZRP ruting protokola**
- 34. Slika 6.1. **Format OLSR-BF paketa**
- 35. Slika 6.2. **Generisanje i širenje pozdravnih poruka**
- 36. Slika 6.3. **Format pozdravne poruke**
- 37. Slika 6.4. **Brodkastovanje TC poruka**
- 38. Slika 6.5. **Format TC poruke**
- 39. Slika 6.6. **Format HNA poruke**
- 40. Slika 6.7. **Odabir MPR čvorova**
- 41. Slika 6.8. **Odabir MPR čvorova kod OLSR-BF ruting protokola**
- 42. Slika 6.9. **MANET mreža od 7 čvorova**
- 43. Slika 6.10. **Informaciona skladišta kod OLSR-BF ruting protokola**
- 44. Slika 6.11. **Proračun putanje od izvornog čvora S do odredišnog čvora D**
- 45. Slika 7.1. **Proces simulacije projekta kod OPNET simulatora**
- 46. Slika 7.2. **Hijerarhijska organizacija kod OPNET simulatora**
- 47. Slika 7.3. **Opterećenost mreže kontrolnim saobraćajem**
- 48. Slika 7.4. **Primer modela mobilnosti slučajne putanje čvora P**
- 49. Slika 8.1. **Mrežna toplogija sa 20 mobilnih čvorova**
- 50. Slika 8.2. **Propusni opseg [bit/s]**
- 51. Slika 8.3. **Kašnjenje [s]**
- 52. Slika 8.4. **Opterećenost mreže kontrolnim saobraćajem [bit/s]**

- 53. Slika 8.5. Broj poslatih pozdravnih poruka**
- 54. Slika 8.6. Broj proračuna ruting tabele**
- 55. Slika 8.7. Mrežna topologija sa 50 mobilnih čvorova**
- 56. Slika 8.8. Propusni opseg [bit/s]**
- 57. Slika 8.9. Kašnjenje [s]**
- 58. Slika 8.10. Opterećenost mreže kontrolnim saobraćajem [bit/s]**
- 59. Slika 8.11. Broj poslatih pozdravnih poruka**
- 60. Slika 8.12. Broj proračuna ruting tabele**

INDEKS TABELA

1. Tabela 3.1. Tabela incidencije
2. Tabela 3.2. Tabelarni zapis Primovog algoritma
3. Tabela 3.3. Tabelarni zapis Dijkstrinog algoritma
4. Tabela 3.4. Tabela incidencije zadatog grafa
5. Tabela 3.5. Matrica incidencije nakon drugog koraka
6. Tabela 3.6. Matrica incidencije nakon četvrtog koraka
7. Tabela 3.7. Matrica incidencije nakon petog koraka
8. Tabela 3.8. Tabela incidencije
9. Tabela 3.9. Tabela incidencije u prvom koraku
10. Tabela 3.10. Tabela incidencije (2)
11. Tabela 3.11. Tabela incidencije u drugom koraku
12. Tabela 3.12. Tabela incidencije (3)
13. Tabela 3.13. Tabela incidencije u trećem koraku
14. Tabela 3.14. Tabela incidencije (4)
15. Tabela 3.15. Tabela incidencije u četvrtom koraku
16. Tabela 5.1. DSDV ruting tabela čvora A
17. Tabela 5.2. Sumarne karakteristike proaktivnih, reaktivnih i hibridnih ruting protokola
18. Tabela 5.3. Sumarno poređenje karakteristika različitih ruting protokola
19. Tabela 6.1. Odabrani MPR čvorovi i MPR skupovi selektora
20. Tabela 7.1. Parametri simulacije
21. Tabela 7.2. Parametri bežične mreže
22. Tabela 7.3. Parametri AODV ruting protokola
23. Tabela 7.4. Parametri OLSR i OLSR-BF ruting protokola
24. Tabela 7.5. Parametri DSDV ruting protokola
25. Tabela 9.1. Sumarni pregled rezultata eksperimenata

REČNIK POJMOVA

A

ABR (Associatively-Based Routing Protocol) – rutinog protokol zasnovan na povezivanju

ACK (Acknowledgment) – potvrda prijema

ANSN (Advertised Neighbour Sequence Number) – redni broj oglašenog susednog čvora

AODV (Ad hoc On Demand Distance Vector routing) – rutiranje na zahtev zasnovano na vektoru razdaljine

AP (Access Point) – pristupna tačka

AR (Access Router) – pristupni ruter

ASYM_LINK (Asymmetric Link) – asimetrična veza

ATM (Asynchronous Transfer Mode) – asinhroni režim prenosa

B

B.A.T.M.A.N. (Better Approach To Mobile Adhoc Networking) – bolji pristup mobilnom ad hoc umrežavanju

BcastID (Broadcast Identifier) – brodcast identifikator

bps, bit/s (bits per second) – biti po sekundi

BRP (Bordercast Resolution Protocol) – granični protokol za razrešenje

C

CCA (Clear Channel Assessment) – procena slobodnog kanala

CCK (Complementary Code Keying) – komplementarna kodna modulacija

CDMA (Code Division Multiple Access) – jedinstveni kodom višetrukog pristupa

CGSR (Cluster-head Gateway Switch Routing) – promenljivo rutiranje grupisanih zaglavlja i gejtveja

CH (Cluster Head) – glava klastera

CLR (Clear) – brisanje

CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) – višestruki pristup sa osluškivanjem nosioca i izbegavanjem kolizija

CSMA/CD (Carrier Sense Multiple Access/Collision Detection) – višestruki pristup sa osluškivanjem nosioca i otkrivanjem kolizija

CTS (Clear to Send) – dozvola za slanje

D

DAG (Directed Acyclic Graph) – usmereni aciklični graf

DBPSK (Differential Binary Phase Shift Keying) – modulacija diferencijalnog binarnog faznog pomeraja

DCF (Distributed Coordination Function) – distribuirana koordinativna funkcija

DestID (Destination Identifier) – identifikator odredišta

Destination_inter (Intermediate Destination) – središnje odredište

DestSeqNum (Destination Sequence Number) – redni broja odredišta

DKPSK (Differential Quadrature Phase Shift Keyining) – modulaciona šema diferencijalnog kvadraturnog faznog pomeraja

DLL (Data Link Layer) – sloj veze (podataka)

DoS (Denial of Service) – uskraćivanje usluga

DRP (Dynamic Routing Protocol) – dinamički ruting protokol

DSDV (Destination-Sequenced Distance-Vector Routing) – rutiranje destinacionih sekvenci zasnovano na vektoru razdaljine

DSR (Dynamic Source Routing) – dinamičko navođenje tačne putanje

DSRFLOW (Flow State in the Dynamic Source Routing) – stanje toka kod dinamičkog navođenja tačne putanje

DSSS (Direct Sequence Spread Spectrum) – direktno sekvencijalno širenje spektra

DT (Distance Table) – tabela razdaljina

DV (Distance Vector) – vektor razdaljine

DYMO (DYnamic Manet On-demand Routing) – dinamički na zahtev MANET ruting protokol

E

E-mail – elektronska pošta

ELN (Explicit Loss Notifications) – eksplicitno obaveštenje o gubicima

ETX (Expected Transmission Count) – očekivani trošak prenosa

F

FHSS (Frequency Hopping Spread Spectrum) – skokovito frekventno širenje spektra

FP (Forwarding Protocol) – protokol za prosleđivanje

FTP (File Transfer Protocol) – protokol za prenos datoteka

G

GFSK (Gaussian Frequency-Shift Keying) – Gausova digitalna frekvencijska modulacija

GSM (Global System for Mobile Communications) – globalni sistem za mobilne komunikacije

H

HARP (Heading-direction Angle Routing Protocol) – ruting protokol ugaone usmerenosti glave

HNA (Host and Network Association) – poruka o terminalima i mrežnim udruženjima

HR/DSSS (High Rate Direct Sequence Spread Spectrum) – visokobrzinsko direktno sekvencijalno širenje spektra

HSLs (Hazy Sighted Link State) – maglovito pronicljiv ruting protokol zasnovan na stanju veze

HSR (Hierarchical State Routing) – ruting protokol zasnovan na hijerarhiji stanja

Htime (HELLO emission interval) – interval emitovanja pozdravne poruke

HTTP (Hypertext Transfer Protocol) – protokol za prenos hiperteksta

HWMP (Hybrid Wireless Mesh Protocol) – hibridni bežični meš ruting protokol

I

IARP (Intrazone Routing Protocol) – unutarzonski ruting protokol

IEEE (Institute of Electrical and Electronics Engineers) – institut inženjera elektrotehnike i elektronike

IERP (Interzone Routing Protocol) – međuzonski ruting protokol

IETF (Internet Engineering Task Force) – Internet inženjering radna grupa

IP (Internet Protocol) – Internet protokol

IR (Infrared) – infracrveni

IS-IS (Intermediate System-Intermediate System) – protokol za međusistemske veze

ISM (Industrial, Scientific and Medical) – industrijski, naučni i medicinski

ISO (International Standardization Organization) – međunarodna organizacija za standardizaciju

L

LAN (Local Area Network) – lokalna računarska mreža

LCT (Link Cost Table) – tabela težina veza

LLC (Logical Link Control) – upravljanje logičkom vezom

LLC (Least Cluster Change) – algoritam najmanje promenljivog klastera

LS (Link-State) – stanje veze

M

MAC (Media Access Control) – kontrola pristupa medijumu

MANET (Mobile Ad hoc NETwork) – mobilna ad hoc mreža

MIMO (Multiple-Input Multiple-Output) – višestruki-ulaz višestruki-izlaz

MIT (Massachusetts Institute of Technology) – Masačusetski institut za tehnologiju

MITM (Man-In-The-Middle) – čovek u sredini

MPR (Multipoint Relay) – relej više čvorova

MPR_Seq_Num (MPR Sequence Number) – MPR redni broj

MRL (Message Retransmission List) – tabela reemitovanih poruka

MWP (Markovian Waypoint Model) – Markovijanov model mobilnosti putanje

N

NAM (Network Animator) – mrežni animator

NHDP (Neighborhood Discovery Protocol) – protokol za otkrivanje susednih čvorova

Ns-3 (Network Simulator version 3) – mrežni simulator verzije 3

O

OFDM (Orthogonal Frequency Division Multiplexing) – ortogonalno multipleksiranje podelom frekvencije

OLSR (Optimized Link State Routing Protocol) – optimizovano rutiranje zasnovano na stanju veze

OLSR-BF (Optimized Link State Routing Protocol - Bellman-Ford) – optimizovano rutiranje zasnovano na stanju veze - Belman-Ford

OOK (On-Off Keying) – uključi-isključi modulacija

OORP (OrderOne Routing Protocol) – OrderOne rutinog protokol

OPNET (Optimized Network Engineering Tool) – optimizovani mrežni tehnički alat

OSI (Open Systems Interconnection) – povezivanje otvorenih sistema

OSPF (Open Shortest Path First) – otvoreni protokol najkraće putanje (engl.

P

PDA (Personal Digital Assistant) – lični digitalni pomoćnik

PHY (Physical Layer) – fizički sloj

PLCP (Physical Layer Convergence Protocol) – protokol konvergencije fizičkog sloja

PMD (Physical Medium Dependent) – zavistan od fizičkog medijuma

PPM (Pulse Position Modulation) – impulsno-poziciona modulacija

PRNET (Packet Radio Networks) – paketske radio mreže

PSN (Packet Sequence Number) – redni broj paketa

Q

QRY (Query) – upit

R

R_dist (Route Distance) – ruting rastojanje

RERR (Route Error) – greška na ruti

RF (Radio Frequency) – radio-frekvencija

RREP (Route Response) – odgovor za rutom

RREQ (Route Request) – zahtev za rutom

RT (Routing Table) – ruting tabela

RTS (Request to Send) – zahtev za slanje

RWP (Random Waypoint Mobility Model) – model mobilnosti slučajne putanje

RWPB (Random Waypoint on the Border) – slučajna putanja na granici

Rxgroup (Receiver Group) – grupa prijemnika

S

SACK (Selective Acknowledgment Options) – selektivno prihvatanje opcija

SrcID (Source Identifier) – identifikator izvora

SrcSeqNum (Source Sequence Number) – redni broj izvora

SSA (Signal Stability-Based Adaptive Routing Protocol) – adaptivni ruting protokol zasnovan na stabilnosti signala

SST (Signal Stability Table) – tabela stabilnosti signala

SYM_LINK (Symetric Link) – simetrična veza

T

TC (Topology Control) – kontrola topologije

TCP (Transport Control Protocol) – protokol za upravljanje prenosom

TDMA (Time Division Multiple Access) – višestruki pristup podelom vremena

TORA (Temporally Ordered Routing Algorithm) – privremeno uređen ruting algoritam

TTL (Time-to-Live) – vreme života

U

UDP (User Datagram Protocol) – protokol za korisničke datagrame

UPD (Update) – ispravka

V

VoIP (Voice over IP) – protokol za prenos glasa preko Interneta

Vtime (Validity Time) – vreme validnosti

W

W-CDMA (Wideband Code Division Multiple Access) – širokopojasni kodirani višestruki pristup

WLAN (Wireless Local Area Network) – bežična lokalna računarska mreža

WRP (Wireless Ruting Protokol) – bežični ruting protokol

Z

ZRP (Zone Routing Protocol) – zonski ruting protokol

SADRŽAJ

1. UVOD.....	18
1.1. Kontekst istraživanja.....	20
1.2. Motivacija.....	20
1.3. Ciljevi istraživanja.....	23
1.4. Metodologija istraživanja.....	24
1.5. Struktura teze.....	24
2. PREGLED POSTOJEĆIH RADOVA.....	25
Prvi deo – Teorija grafova	
3. TEORIJA GRAFOVA.....	29
3.1. Osnovni pojmovi u teoriji grafova.....	29
3.2. Razapinjuća stabla.....	31
3.2.1. Primov algoritam.....	31
3.2.2. Kruskalov algoritam.....	34
3.3. Najkraće udaljenosti u grafu.....	36
3.3.1. Dijkstrin algoritam.....	37
3.3.2. Belman-Fordov algoritam.....	40
3.3.3. Floyd-Voršalov algoritam.....	41
3.4. Maksimalni protok u mreži.....	44
3.4.1. Ford-Fulkersonov algoritam.....	46
Drugi deo – Ruting protokoli u mobilnim ad hoc mrežama	
4. MOBILNE AD HOC MREŽE.....	51
4.1. MANET i OSI referentni model.....	52
4.1.1. Fizički sloj.....	53
4.1.2. Sloj veze (podataka).....	55
4.1.3. Mrežni sloj.....	57
4.1.4. Transportni sloj.....	57
4.2. Izazovi u ad hoc bežičnim mrežama.....	58
4.3. Primena ad hoc bežičnih mreža.....	61
5. RUTING PROTOKOLI U MOBILNIM AD HOC MREŽAMA.....	62
5.1. Karakteristike ruting protokola.....	62
5.2. Klasifikacija postojećih ruting protokola.....	64
5.3. Unikast ruting protokoli.....	66
5.3.1. Proaktivni ruting protokoli.....	66
5.3.1.1. DSDV.....	66
5.3.1.2. WRP.....	68
5.3.1.3. CGSR.....	69
5.3.1.4. Babel.....	70
5.3.1.5. B.A.T.M.A.N.	71
5.3.1.6. HSR.....	72

5.3.1.7. OLSR.....	72
5.3.2. Reaktivni ruting protokoli.....	75
5.3.2.1. AODV.....	76
5.3.2.2. HARP.....	79
5.3.2.3. DSR.....	80
5.3.2.4. DSRFLOW.....	82
5.3.2.5. ABR.....	82
5.3.2.6. SSA.....	83
5.3.2.7. DYMO.....	84
5.3.2.8. TORA.....	86
5.3.3. Hibridni ruting protokoli.....	87
5.3.3.1. ZRP.....	88
5.3.3.2. HSLS.....	90
5.3.3.3. HWMP.....	92
5.3.3.4. OORP.....	92
5.4. Poređenje unicast ruting protokola.....	94
Treći deo – Sopstveni ruting protokol u mobilnim ad hoc mrežama	
6. SOPSTVENI RUTING PROTOKOL.....	98
6.1. Osnovne funkcionalnosti OLSR-BF ruting protokola.....	100
6.2. OLSR-BF format paketa.....	100
6.3. OLSR-BF poruke.....	102
6.3.1. Pozdravne poruke.....	102
6.3.2. Poruke kontrole topologije.....	103
6.3.3. Poruke o terminalima i mrežnim udruženjima.....	104
6.4. MPR čvorovi.....	105
6.5. Informaciona skladišta.....	108
6.6. Proračun ruting tabele.....	109
7. SIMULACIJA.....	111
7.1. Metodologija istraživanja.....	111
7.1.1. Pretpostavke.....	111
7.1.2. Opravdanost metodologije istraživanja.....	111
7.2. OPNET Modeler.....	114
7.2.1. Arhitektura.....	114
7.2.2. Hijerarhijsko modelovanje.....	115
7.3. Metrike.....	116
7.3.1. Propusni opseg (propusnost).....	117
7.3.2. Kašnjenje.....	117
7.3.3. Opterećenost mreže kontrolnim saobraćajem.....	118
7.4. Simulaciono okruženje.....	120
7.4.1. Model mobilnosti čvorova.....	122
8. REZULTATI SIMULACIJA.....	124
8.1. Mreža od 20 mobilnih čvorova.....	124
8.1.1. Propusni opseg.....	125
8.1.2. Kašnjenje.....	125
8.1.3. Opterećenost mreže kontrolnim saobraćajem.....	126
8.1.4. Broj poslatih pozdravnih poruka.....	126
8.1.5. Broj proračuna ruting tabele.....	127

8.2. Mreža od 50 mobilnih čvorova.....	128
8.2.1. Propusni opseg.....	128
8.2.2. Kašnjenje.....	129
8.2.3. Opterećenost mreže kontrolnim saobraćajem.....	129
8.2.4. Broj poslatih pozdravnih poruka.....	130
8.2.5. Broj proračuna ruting tabele.....	131
9. ANALIZA REZULTATA EKSPERIMENATA.....	132
9.1. Analiza posmatranih metrika na performanse ruting protokola.....	132
9.1.1. Propusni opseg.....	132
9.1.2. Kašnjenje.....	133
9.1.3. Opterećenost mreže kontrolnim saobraćajem.....	133
9.1.4. Iskorišćenost resursa.....	134
9.1.5. Skalabilnost.....	135
9.2. Analiza posmatranih faktora na performanse ruting protokola.....	135
9.2.1. Veličina mreže.....	135
9.2.2. Mobilnost čvorova.....	135
9.3. Komparativna analiza ruting protokola.....	136
9.4. Sumarna analiza rezultata eksperimenata.....	137
10. ZAKLJUČAK.....	140
10.1. Budući rad.....	142
11. LITERATURA.....	145
PRILOG.....	153
Izvorni kod Belman-Fordovog algoritma u C programskom jeziku.....	153

1. UVOD

Sa unapređenjem bežičnih komunikacionih tehnologija, smanjenjem veličine mikročipova i povećanjem procesorskih performansi komunikacioni uređaji se sve više koriste u svakodnevnom životu i poslovnom okruženju (npr. laptop računari, PDA uređaji i pametni mobilni telefoni opremljeni sa bežičnim mrežnim adapterom) [1]. Dok infrastrukturni bežični sistemi predstavljaju tradicionalni model za ostvarivanje mobilne bežične komunikacije, u ovom radu razmatraju se bežične mreže koji se ne oslanjaju na fiksnu infrastrukturu. Takve mreže nazivaju se mobilne ad hoc mreže (MANET).

Mobilna ad hoc mreža (engl. *Mobile Ad hoc Network*, *MANET*) je samoorganizujući skup međusobno nezavisnih mobilnih čvorova koji razmenjuju informacije bez potrebe za uspostavljanjem fiksne infrastrukture i centralizovane administracije. Svaki čvor održava ažurnim informacije o mreži i ruting protokolima na osnovu kojih obavlja usmeravanje saobraćaja. MANET mreže su stekle ogromnu popularnost zbog svoje lakoće za postavljanjem i njihove potencijalne primene u brojnim oblastima. MANET mreže mogu se koristiti u situacijama gde nema dovoljno vremena ili resursa za konfigurisanje standardne infrastrukturne bežične ili ožičene mreže. Mobilne ad hoc mreže su atraktivne za taktičke komunikacije u vojnim i policijskim snagama u kojima su jedinice nasumice raspoređene i mobilne, pri čemu se centralna jedinica ne može koristiti za sinhronizaciju, kao i u civilnim upotrebama poput kongresnih centara, konferencija i elektronskih učionica.

U MANET mrežama komunikacija se odvija bez potrebe za uređenom infrastruktorom, tj. baznim stanicama i pristupnim tačkama [2]. MANET mreže su stekle značajnu pažnju u poslednjih nekoliko godina zbog njihove potencijalne primene u mnogim oblastima. Takve mreže mogu biti korisne u scenarijima spašavanja gde nema dovoljno vremena ili resursa za konfigurisanje ožičene mreže. Ad hoc mreže se koriste u vojnim operacijama u kojima se jedinice kreću po terenu na slučajan način, pri čemu se centralna kontrolna jedinica ne može koristiti za sinhronizaciju [3]. Bežična ad hoc mreža je skup mobilnih čvorova koji dinamički formiraju privremenu mrežu bez upotrebe bilo koje postojeće mrežne infrastrukture. U MANET mrežama čvorovi nisu samo pošiljaoci i primaoci poruka na kojima se nalaze podaci i aplikacije, već svaki čvor istovremeno radi i kao ruter pri čemu učestvuje u prosleđivanju podataka ka njihovoj krajnjoj destinaciji.

Ukratko, MANET se definiše kao mrežni interfejs za bežičnu konekciju [4]. Veza se uspostavlja u trajanju od jedne sesije koja počinje kada se čvor pridružuje mreži, i završava se kada čvor napušta mrežu, pri čemu se ne zahteva kontrolna jedinica za organizovanje ovog procesa. Kao alternativa, čvorovi otkrivaju druge čvorove unutar svog transmisionog opsega i formiraju mrežu. Veze je moguće ostvariti preko više čvorova i više različitih puteva, pri čemu se formira ad hoc mreža sa više preskoka (engl. *multihop*). Zadatak ruting protokola je da obezbedi i održava veze aktivnim, čak i ako se čvorovi kreću izvan granica mrežnog područja [5]. Mobilne ad hoc mreže su samoorganizujuće, brzo razmeštajuće, i ne zahtevaju fiksnu infrastrukturu. Bežični čvorovi međusobno sarađuju radi uspostavljanja dinamičkih komunikacija uz korišćenje ograničene mrežne administracije. To je razlog zašto ad hoc ruting protokoli funkcionišu na distribuiran način. Čvorovi u ad hoc mrežama mogu biti mobilni ili stacionarni. Međusobno se razlikuju u smislu njihovih mogućnosti i upotrebe. Cilj

ad hoc mrežne arhitekture je postizanje veće fleksibilnosti, mobilnosti i lakoće upravljanja u odnosu na ožičene ili bežične mreže sa uređenom infrastrukturom [6].

Struktura MANET mreža menja se dinamički zbog mobilnosti čvorova [1]. Čvorovi u ovim mrežama dele isti slučajni bežični pristupni kanal. Oni međusobno sarađuju kako bi ostvarili prosleđivanje sa preskocima između čvorova koji inače ne bi mogli direktno komunicirati. Svaki čvor funkcioniše kao terminal (engl. *host*), ali i kao ruter koji održava rute i prosleđuje podatke ka ostalim čvorovima u mreži koji ne mogu ostvariti direktnu bežičnu komunikaciju. Rutiranje u mobilnim ad hoc mrežama suočava se sa ekstremnim izazovima poput mobilnosti čvorova, potencijalno veoma velikog broja čvorova i ograničenih komunikacionih resursa (npr. veličine propusnog opsega i energije za napajanje čvorova). Ruting protokoli u mobilnim ad hoc bežičnim mrežama moraju se brzo prilagoditi čestim i nepredvidljivim promenama topologije, i moraju biti efikasni u komunikaciji i obradi resursa.

Komunikacioni sistemi za vojne, policijske i hitne operacije razvijaju se iz sistema jednog preskoka (engl. *singlehop*) u kojima se prenosi samo glas, ka sofisticiranijim širokopojasnim sistemima višestrukih preskoka (engl. *multihop*) koji podržavaju raznolikost usluga, uključujući prenos glasa, podataka i video komunikacije. Poboljšanje funkcionalnosti i povećanje kapaciteta omogućuje efikasniju razmenu važnih informacija, i omogućuje brži odgovor u vanrednim i hitnim operacijama [7].

Nerealno je očekivati da mobilna ad hoc mreža bude u potpunosti povezana, pri čemu bi svaki čvor mogao direktno komunicirati sa svakim drugim čvorom u mreži. Tipično čvorovi moraju koristiti tehniku više preskoka (engl. *multihop*) za prenos, a paket može preći više čvorova pre nego što stigne na odredište.

Tehnologija koju nude ad hoc mreže je zamišljena da bude jedna od ključnih komponenti u realizaciji takvih komunikacionih mreža. Ad hoc mreža može se koristiti za proširenje postojeće ožičene komunikacione infrastrukture, ili može funkcionisati kao nezavisna mreža bez uređene infrastrukture. Ključna karakteristika mobilnih ad hoc mreža je sposobnost mobilnih uređaja da komuniciraju i da izgrađuju mrežu bez potrebe za bilo kakvom infrastrukturom. Na taj način celokupna komunikacija se u potpunosti zasniva na bežičnim vezama, što olakšava brzu implementaciju mreže, na znatno isplativiji način. Pored toga, mreža se automatski i brzo prilagođava dinamici u hitnim i spasilačkim operacijama. Čvorovima je dozvoljeno da budu mobilni, da se pridruže i isključe sa mreže u bilo kom trenutku.

Iako su mobilne ad hoc mreže obećavajuća tehnologija za vojne i civilne komunikacije, postoje brojni izazovi koje treba rešiti. Ovi izazovi se odnose na faktore kao što su efikasno rutiranje paketa, nedostatak centralizovanog upravljanja, ograničen bežični radio kapacitet, smetnje usled zračenja drugih radio sistema, kao i problem sigurnosti [7]. Zbog činjenice da je propusni opseg MANET čvorova znatno manji u odnosu na ožičene mreže, problem skalabilnost bežičnih ruting protokola uglavnom se bavi prekomernom količinom kontrolnih ruting poruka izazvanih povećanjem broja čvorova u mreži i njihove mobilnosti. Veličina ruting tabele je takođe veliki problem MANET mreža, jer velike ruting tabele podrazumevaju razmenu velike količine kontrolnih paketa, a samim time i dodatno trošenje resursa na bežičnom linku.

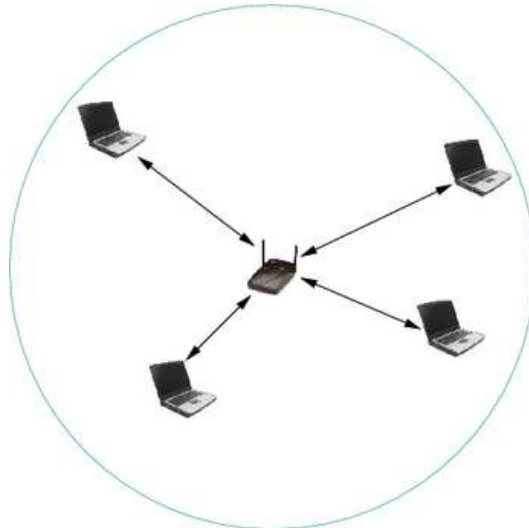
1.1. Kontekst istraživanja

Glavni razlozi za dizajn i razvijanje mobilnih ad hoc mreža je da se odgovori na vojne potrebe za bojno polje [8]. Situacije na bojnom polju zahtevaju od vojnika da se kreću sa jedne na drugu poziciju bez ikakvih ograničenja od strane ožičene komunikacije, i da pritom međusobno komuniciraju jedni sa drugima bez zavisnosti od fiksne infrastrukture. Pošto je gotovo nemoguće obezbediti fiksnu mrežnu infrastrukturu na pojedinim teritorijama, kao što su pustinje, pravovremeno raspoređivanje mobilnih čvorova koji komuniciraju putem bežične mreže postaje kritično. Drugi odlučujući motiv je razmatranje fizike elektromagnetne propagacije. Činjenica je da su frekvencije mnogo veće od 100 MHz ograničene njihovom propagacionom udaljenošću. Za ostvarivanje komunikacije između jednog mobilnog čvora sa drugim mobilnim čvorovima izvan svog radijusa zračenja, neophodni su ruting protokoli sa više preskoka (engl. *multihop*). To znači da se poruke prenose od izvornog ka odredišnom čvoru preko središnjih čvorova.

U poslednjih nekoliko godina došlo je do ekspanzije u istraživanju mobilnih ad hoc mreža (MANET). MANET mreže nemaju uređenu infrastrukturu, a karakteriše ih mobilnost čvorova koji se stalno kreću. Čvorovi mogu direktno komunicirati jedni sa drugima, pod uslovom da uđu u međusobno preklapajući komunikacioni domet. Čvorovi mogu prosledivati pakete, tj. služiti kao releji. Stoga, poruka se prenosi kroz ad hoc mrežu s jednog čvora na drugi, sve dok ne dostigne svoje odredište. S obzirom da se čvorovi kreću to postaje izazovan zadatak, jer je topologija mreže podložna stalnim promenama. Kako pronaći odredište, kako pronaći put do odredišta, i kako obezbediti robustnu komunikaciju usled stalnih promena topologije glavni su izazovi u mobilnim ad hoc mrežama. Rutiranje u mobilnim ad hoc mrežama tema je stalnog proučavanja.

1.2. Motivacija

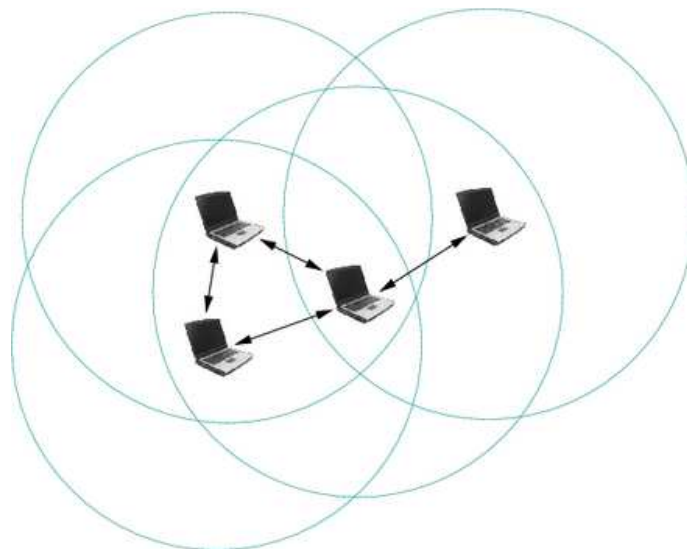
U današnje vreme bežične komunikacije se uveliko koriste. Koriste se uređaji koji podržavaju različite protokole (GSM, W-CDMA, IEEE 802.11, itd.) u cilju ostvarivanja komunikacije sa drugim uređajima i pristupa Internetu. Ovi uređaji mogu biti laptop računari, mobilni telefoni, PDA uređaji i sl. Uređaji mogu pristupiti bežičnoj mreži u okviru opsega pokrivenosti bazne stanice, pri čemu se podaci prenose do i od bazne stanice (slika 1.1). Komunikacija između izvornog i odredišnog klijenta se ostvaruje putem bazne stanice.



SLIKA 1.1. BEŽIČNA MREŽA SA BAZNOM STANICOM (IEEE 802.11 PRISTUPNA TAČKA)

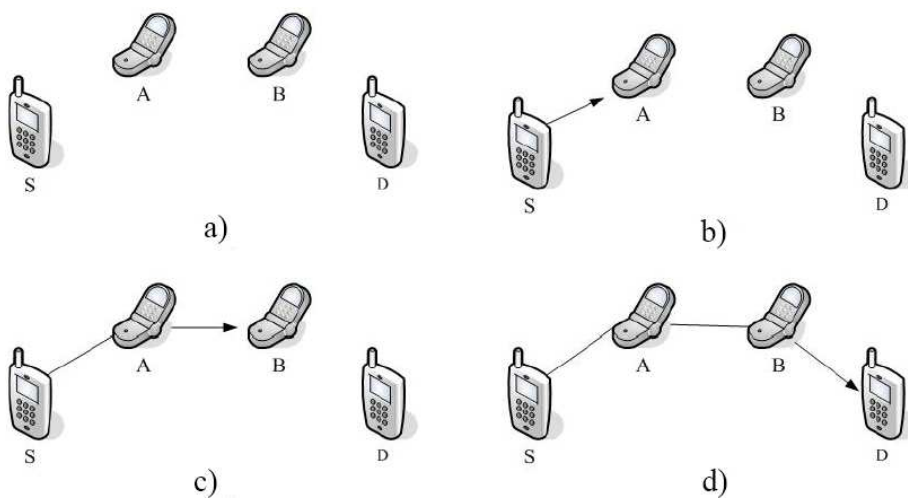
Kod ovog modela mreže, pokrivenost i opterećenje bazne stanice može ograničiti komunikaciju između klijenata. Komunikacija ne može biti uspostavljena kada je bazna stanica zauzeta ili kada korisnik napusti zonu pokrivenosti. Kada bazna stanica primi više zahteva nego što može da obradi postaje nedostupna, odnosno ne može obezbediti dovoljno resursa, što dovodi do toga da se komunikacija ne može uspostaviti kroz ovu baznu stanicu. Pri tome, ako je potrebna primopredaja komunikacije iz određene bazne stanice u drugu baznu stanicu koja je zagušena, komunikacija će biti blokirana, a paketi podataka će biti izgubljeni.

Jedno od mogućih rešenja je da se smanji opterećenost baznih stanica, odnosno može se ostvariti komunikacija bez upotrebe baznih stanica. Mobilne ad hoc mreže omogućuju klijentima da međusobno komuniciraju bez infrastrukturne bazne stanice (slika 1.2). S obzirom da fiksna bazna stanica ne postoji, osnovni problem koji se nameće u mobilnim ad hoc mrežama je da se pronađe odredišni čvor, i da se uspostavi rutinog putanja do tog čvora.



SLIKA 1.2. AD HOC MREŽA SASTAVLJENA OD UREĐAJA POVEZANIH BEŽIČNIM PUTEM

U MANET mrežama komunikacija se ostvaruje upotrebom drugih mobilnih uređaja koji kao središnji čvorovi prosleđuju pakete dok se ne dosegne krajnja destinacija [9]. Klijenti mogu poslati svoje pakete podataka tek nakon što se sve komunikacione veze između središnjih čvorova uspostave. Ovaj model komunikacije zahteva tehniku višestrukih preskoka (engl. *multihop*). Kako bi saznao gde se nalazi odredišni čvor, izvorni čvor emituje zahtev za rutom (engl. *Route Request, RREQ*) ka susednim čvorovima. Susedni (središnji) čvorovi reemituju RREQ poruku, jedan po jedan, sve dok se svi čvorovi ne pretraže ili dok se ne pronađe odredišni čvor. Slika 1.3 prikazuje proces pronalaženja odredišnog čvora. Pretpostavimo da je čvor *S* izvorni čvor, i da je čvor *D* odredišni čvor. U početku (slika 1.3.a) ne postoji putanja uspostavljena između izvornog i odredišnog čvora. Zatim, izvorni čvor *S* emituje RREQ poruku (slika 1.3.b). Na slici 1.3.c čvor *A* prima RREQ poruku i reemituje je ka susednom čvoru *B*. Konačno, čvor *B* reemituje ovu poruku ka odredišnom čvoru *D*. Potom se uspostavlja komunikaciona putanja (slika 1.3.d). Stoga, uočljivo je da se komunikacija između izvornog i odredišnog čvora može uspostaviti bez uređene mrežne infrastrukture.



SLIKA 1.3. PROCES PRONALAŽENJA ODREDIŠNOG ČVORA U MANET MREŽI

Međutim, raznoliki mobilni uređaji ne funkcionišu na istovetan način. Resursi kojima raspolažu različiti uređaji poput laptopova, mobilnih telefona i PDA uređaja razlikuju se od jednog do drugog uređaja. Na primer, status svakog uređaja, kao što su napajanje, propusni opseg, procesorska snaga i mobilnost, mogu biti različiti. Štaviše, zbog mobilnosti različitih mobilnih uređaja, topologija bežičnih mreža dinamički se menja. Drugim rečima, brojni faktori mogu uticati na stabilnost ruting putanje. Pored toga, resursi svakog čvora su ograničeni. Tokom uspostavljanja ruta i prosleđivanja paketa čvor troši svoje resurse: propusni opseg, procesorsku snagu i napajanje. Dakle, drugi problem u MANET mrežama je izbor ruting putanja, održavanje njihove stabilnosti i dobro upravljanje ograničenim resursima.

Ograničeni propusni opseg, ograničeno baterijsko napajanje i računarska efikasnost su faktori koji utiču na stabilnost jedne ruting putanje [10, 11]. Štaviše, svaki od ovih faktora može individualno uticati na stanje stabilnosti nekog od središnjih čvora. Ako središnji čvor ne može obezbediti dovoljno resursa da zadovolji zahtevane resurse sa izvornog čvora, ovaj čvor postaje neupotrebljiv i sve ruting putanje kroz ovaj čvor su blokirane. Dakle, važno je uzeti u obzir trenutno stanje resursa u čvoru tokom procesa otkrivanja i odabira ruting putanja. Pošto postoje ograničeni resursi u svakom čvoru, samim time postoji i veći rizik da ruting putanja bude blokirana. Stoga, komunikacioni putevi trebaju biti distribuirani. Ako čvor opslužuje

manji broj ruting putanja, resursi ovog čvora možda će biti zadovoljavajući. Čak i ako taj čvor napusti komunikacioni opseg pokrivanja drugih čvorova ili se ugasi, broj ruting putanja koje zavise od ovog čvora biće manji.

S obzirom da su bežične mreže dinamične, pored problema održavanja resursa u svakom mobilnom uređaju javlja se i problem razdaljine između dva čvora. Ako je rastojanje duže od komunikacionog opsega pokrivenosti ova dva čvora, komunikacija između, i kroz ova dva čvora biće prekinuta. Osim toga, sve ruting putanje koje prolaze preko ova dva mobilna čvora biće odbačene. Kada se ruting putanja između izvornog i odredišnog čvora raskine, izvorni čvor počinje tražiti drugu putanju do odredišnog čvora. Ako ne može pronaći putanju do odredišnog čvora komunikacija između korisnika na izvornom i odredišnom čvoru biće raskinuta. Većina reaktivnih ruting protokola među svim dostupnim ruting putanjama izabire najkraću. To znači da ovi ruting algoritmi uvek odabiru ruting putanju sa manjim brojem središnjih čvorova, i generalno najkraću ruting putanju. Kod ruting algoritama koji ruting putanju pronalaze na osnovu najkraćeg puta, isti središnji čvorovi mogu biti izabrani veći broj puta i samim time pretrpani saobraćajem. To znači da ako postoje brojne ruting putanje uspostavljene kroz iste središnje čvorove, ovi čvorovi lako mogu postati usko grlo (engl. *bottleneck*) sa preopterećenim saobraćajem [10].

Ključni motiv ovog rada je da istraži postojeće i dopuni jedan od postojećih ruting protokola u MANET mrežama. Istraživanje se zasniva na simulaciji i komparativnom poređenju sa drugim ruting protokolima koje će bolje predstaviti eventualne nedostatke u predloženom ruting protokolu i otkriti njihove uzroke. Dubinska analiza postojećih ruting protokola je preduslov za njihovo unapređenje i uvođenje novih pristupa.

1.3. Ciljevi istraživanja

Pronalaženje ruting putanja i njihovo održavanje veliki su problem MANET mreža.

Naučni cilj istraživanja ovog rada je modifikacija postojećeg proaktivnog OLSR ruting protokola kako bi se na osnovu njega predložio novi ruting protokol koji će iskoristiti efikasnost Belman-Fordovog algoritma iz teorije grafova za unapređenje postojećeg OLSR protokola.

Poseban cilj istraživanja je sprovođenje sveobuhvatne analize i klasifikovanje postojećih ruting protokola u MANET mrežama.

Za sprovođenje ovog istraživanja izvedeni su eksperimenti u OPNET Modeler i ns-3 mrežnim simulatorima. Predloženi ruting protokol upoređen je sa postojećim AODV, DSDV i OLSR ruting protokolima. Podaci dobijeni u tim eksperimentima kvantifikuju i porede odnos u performansama, kao što su propusni opseg, kašnjenje i količina generisanog kontrolnog saobraćaja.

1.4. Metodologija istraživanja

Metodologija istraživanja u ovom radu obuhvata složen i organizovan postupak polazeći od logičkih načela i principa po utvrđenim fazama.

Osnovu i suštinu istraživanja čini sistematičan pristup definisanju i pokušaju rešavanja problema. Primenjena je kombinovana teorijsko-eksperimentalna metoda rada, tj. osnovne naučne metode primerene ovom nivou i tipu naučnog istraživanja:

- Prikupljanje i proučavanje postojeće literature, modela i analiza.
- Metode komparativne analize prikupljenog i proučenog materijala.
- Metode analize i sinteze, zasnovane na stečenom znanju i iskustvima.
- Primena eksperimentalnih i kvantitativnih metoda.

Ovaj izbor istraživačkih metoda upotrebljen je kako bi se istraživanje i tok istraživačkog procesa u svim fazama, tj. identifikaciji i definisanju problema, planiranju dizajna istraživanja, sakupljanju, obradi i analizi podataka, kao i formulaciji zaključaka korektno sproveo u skladu sa osnovnim principima naučno istraživačkog rada. Pristup istraživanju je integrativan, sintetički u tom smislu što se ni jednom metodološkom postupku ne daje isključiva prednost, već su svi podjednako zastupljeni.

1.5. Struktura teze

Rad je podeljen u nekoliko funkcionalnih celina. Nakon uvodnog poglavlja, u drugom poglavlju dat je pregled postojećih radova u ovoj oblasti.

Prvi deo rada, ujedno i treće poglavlje, odnosi se na teoriju grafova.

Drugi, teoretski deo rada daje sveobuhvatni pregled postojećih ruting protokola u mobilnim ad hoc mrežama. Sastoji se iz dva poglavlja. U četvrtom poglavlju detaljnije su opisane MANET mreže, a u petom ujedno i najvećem poglavlju klasifikovani su postojeći unikast ruting protokoli.

Treći deo rada odnosi se na praktičnu implementaciju sopstvenog ruting protokola u mobilnim ad hoc mrežama. U šestom poglavlju opisan je novopredloženi OLSR-BF proaktivni ruting protokol koji se u najvećoj meri zasniva na upotrebi postojećeg OLSR ruting protokola. Novost se ogleda u primeni Belman-Fordovog algoritma iz teorije grafova. U sedmom poglavlju opisan je metod simulacije korišćen za kvalitativno poređenje performansi OLSR-BF ruting protokola. Rezultati simulacija prikazani su u osmom poglavlju, dok je analiza rezultata eksperimenata data u devetom poglavlju.

U desetom poglavlju izneseni su zaključci i ograničenja ovog eksperimentalnog istraživanja. Pregled korišćene literature dat je u poslednjem, jedanaestom poglavlju.

U prilogu rada dat je kompletan izvorni kod Belman-Fordovog algoritma u C programskom jeziku korišćen u ovom istraživanju.

2. PREGLED POSTOJEĆIH RADOVA

Motivacija iza ovog istraživanja je potreba za uvek boljim performansama i optimizacijom ruting protokola u mobilnim ad hoc mrežama. Postojeći radovi u ovoj oblasti delimično su to i učinili.

Nasuprot generalnom mišljenju, studija [12] pokazuje da reaktivni ruting protokoli ne ostvaruju nizak nivo generisanih kontrolnih poruka u apsolutno svim modelima MANET mreža. Naime, količina generisanog kontrolnog saobraćaja kod reaktivnih ruting protokola osetljiva je na opterećenost saobraćaja u pogledu broja saobraćajnih tokova, kao i na mobilnost čvorova u smislu stope promena veza. Dakle, reaktivni ruting protokoli mogu biti pogodni za MANET mreže sa malim brojem saobraćajnih tokova i malom stopom promena veza.

Autor je pokazao da je moguće održati celokupnu mrežnu topologiju MANET mreža sa relativno niskom količinom generisanih kontrolnih poruka. U radu su predstavljene sve navedene metodologije i odgovarajuće aplikacije razvijene za MANET ruting protokole.

U istraživanju [13] predstavljeni su unicast, multicast i broadcast ruting protokoli namenjeni za primenu u MANET mrežama. Komparativnom analizom različitih ruting protokola autor je došao do zaključka da različiti ruting protokoli imaju različite prednosti i nedostatke. Stoga, niti jedan ruting protokol ne može biti adekvatan za primenu u svim mogućim scenarijima i saobraćajnim obrascima MANET mreža. Uopšteno govoreći, reaktivni unicast ruting protokoli nadmašuju proaktivne protokole u pogledu niže količine generisanog kontrolnog saobraćaja unutar mreže. Hibridni unicast ruting protokoli teoretski su napredniji od čisto proaktivnih i reaktivnih ruting protokola.

Trenutno zastupljeni ruting protokoli nemaju ugrađene bezbednosne mehanizme, pa su stoga ranjivi na napade lažiranja IP adresa (engl. *IP address spoofing*) i brojne druge napade. Predloženi sigurnosni ruting protokoli pokušavaju da spreče takve napade na račun veće potrošnje procesorskih resursa i energije, kao i neminovnim uvođenjem znatno dužeg kašnjenja.

U istraživanju [14] predstavljen je novi ruting protokol SEHARP za ad hoc bežične mreže. SEHARP je proširenje postojećeg HARP ruting protokola prvenstveno u vidu integracije bezbednosnih mehanizama. Novi ruting protokol zasnovan je na upravljanju ključevima i sigurnim putanjama između čvorova kako bi se zaštitili prenošeni podaci.

Autor je predstavio novi pristup bezbednosti u neprijateljskom okruženju zasnovan na interakciji između čvorova i na digitalnim sertifikatima, na osnovu kojih svaki čvor treba da bude u stanju da definiše sopstvene bezbednosne odluke u saradnji sa drugim susednim čvorovima.

Izvedba SEHARP protokola proverena je korišćenjem ns-2 mrežnog simulatora korišćenjem metrika kao što su procenat uspešno prenešenih paketa podataka, prosečno kašnjenje u njihovom prenosu i količina generisanih kontrolnih poruka. Rezultati eksperimenata pokazuju

da SEHARP ruting protokol obezbeđuje visok nivo bezbednosti, dostupnosti, skalabilnosti i efikasnosti za primenu u ad hoc bežičnim mrežama.

U [15] predstavljena je modifikovana verzija tradicionalnog ožičenog OSPF ruting protokola koja obuhvata podršku za rutiranje na višestrukim interfejsima, OSPF-MCDS. Novopredloženi ruting protokol implementiran je u okviru ns-2 mrežnog simulatora. Simulacioni model razvijen je kako bi se uporedile performanse OSPF-MCDS ruting protokola sa postojećim MANET protokolima. Rezultati simulacija pokazuju da nijedan ruting protokol ne postiže optimalne performanse za sve slučajeve mobilnosti.

U radu [16] predstavljeni su i upoređeni AODV, DSR i OLSR ruting protokoli. AODV je reaktivni protokol koji započinje pretragu do odredišnog čvora kad god je potrebno poslati bilo kakvu informaciju. Takođe, zahteva periodično oglašavanje ruta i otkrivanje susednih čvorova. DSR je takođe reaktivni protokol, ali za razliku od AODV protokola ne zahteva periodično oglašavanje putanja i otkrivanje suseda. OLSR je proaktivni ruting protokol, a samim time svaki čvor periodično emituje svoju ruting tabelu kako bi se osiguralo da svaki čvor ima istovetan globalni pogled na mrežnu topologiju.

Autor je istražio učinak navedenih ruting protokola u različitim scenarijima sa različitim saobraćajnim opterećenjima, uključujući kašnjenje u prenosu paketa, propusni opseg, kao i količinu generisanog kontrolnog saobraćaja. Rezultati simulacija pokazuju da u slučaju prenosa TCP paketa podataka, OLSR ostvaruje najbolje rezultate u pogledu kašnjenja i vremena odziva, ali uz veliku količinu generisanog kontrolnog ruting saobraćaja. U slučaju prenosa UDP paketa podataka, AODV ostvaruje najvišu vrednost propusnog opsega i pritom ima nisku stopu generisanog kontrolnog saobraćaja.

U radu [17] simulacija različitih MANET ruting protokola izvedena je u QualNet mrežnom simulatoru. Pokazano je da AODV ruting protokol ostvaruje najbolje rezultate u mrežama sa niskom gustom čvorova, kao i da u mrežama visoke gustine čvorova dominiraju OLSR i DSR ruting protokoli. Rezultati izvedenih eksperimenata pokazuju da je DSR optimalan ruting protokol za prenos paketa podataka u scenarijima gde su isporuka i propusni opseg kritični faktori. Takođe, navedeno je da je OLSR najbolji ruting protokol za prenos glasa i video paketa, usled niske stope kašnjenja.

U [18] pokazano je da mobilna ad hoc mrežna arhitektura omogućuje jeftino i jednostavno ostvarivanje mrežnih usluga gotovo bilo gde i bilo kada. Pored pristupa Internetu, od strane krajnjih korisnika najviše se traže i upotrebljavaju multimedijalne aplikacije. Međutim, one nameću stroga ograničenja kvaliteta servisa u vidu propusnog opsega, kašnjenja i gubitka paketa.

Autori su predložili proširenje postojećeg OLSR ruting protokola u vidu modula za ostvarivanje kvaliteta servisa. Njegov učinak i performanse prilikom prenošenja govornih komunikacija analizirali su pomoću OPNET mrežnog simulatora. Rezultati izvedenih eksperimenata pokazuju ostvarene performanse koje krajnji korisnici uključeni u govornu komunikaciju mogu očekivati usled prisustva prenosa drugih tipova saobraćaja, među ostalim korisnicima u istoj mreži.

U [19] ocenjeni su i upoređeni različiti MANET ruting protokoli. Analiziran je uticaj kvaliteta servisa na faktore kao što su propusni opseg i kašnjenje kod AODV i TORA ruting protokola. U bežičnim ad hoc mrežama čvorovi moraju biti u stanju da se povežu jedni sa

drugima kako bi uspostavi vezu od izvora do odredišta. Rezultati eksperimenata pokazuju da AODV nadmašuje TORA ruting protokol u veličini propusnog opsega, ali i da AODV unosi veće kašnjenje. Pokazano je da je AODV bolji za primenu u mrežama sa većim brojem čvorova, dok je TORA primenljiviji za okruženje sa manjim brojem čvorova.

U radu [20] analizirana je veličina propusnog opsega u mobilnim ad hoc mrežama. Performanse MANET mreža u velikoj meri zavise od broja korisnika i opterećenosti saobraćajem. Kada je broj korisnika ili opterećenost mreže visoka povećava se broj sudara, što rezultuje u snižavanju ukupnog protoka. Autori su sprovedli eksperimentalna istraživanja u okviru OPNET Modeler mrežnog simulatora, kako bi uporedili veličinu propusnog opsega mobilnih ad hoc mreža u scenarijima sa različitim brojem korisnika. Analizom dobijenih rezultata, zaključeno je da sa porastom broja korisnika iznad određene granice propusni opseg opada.

U [21] analizirane su performanse AODV, DSR i OLSR ruting protokola korišćenjem ns-2 mrežnog simulatora. Prilikom izvođenja simulacija beleženi su rezultati procenta uspešno isporučenih paketa, količine generisanog kontrolnog saobraćaja, propusnog opsega i kašnjenja. Na osnovu ovih rezultata autori su zaključili da veličina vrednosti propusnog opsega kod sva tri ruting protokola opada sa povećanjem brzine mobilnosti čvorova. Među njima, DSR ostvaruje najvišu vrednost propusnog opsega. Nasuprot tome, prosečna vrednost kašnjenja kod DSR i OLSR ruting protokola raste sa povećanjem brzine čvorova. Tokom izvođenja eksperimenata AODV ruting protokol je ostvario najkonstantnije rezultate. Takođe, pokazano je da OLSR ruting protokol nije pogodan za primenu u visoko mobilnim mrežama.

PRVI DEO

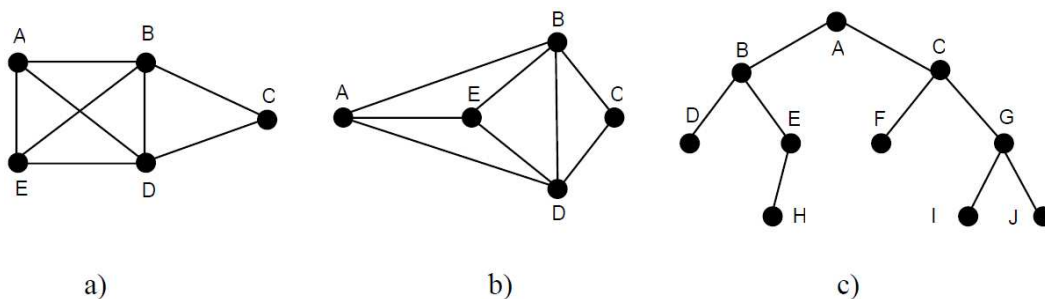
TEORIJA GRAFOVA

3. TEORIJA GRAFOVA

Teorija grafova jedna je od grana matematike koja nalazi veliku primenu u oblasti računarskih mreža, prvenstveno u područjima ruting algoritama, traženja puteva kroz mrežu, kao i opisivanja mrežne topologije [22].

3.1. Osnovni pojmovi u teoriji grafova

Graf je matematički model čiji su primeri navedeni na slici 3.1.



SLIKA 3.1. PRIMERI GRAFOVA

Kao što je sa slike 3.1 vidljivo graf se sastoji od čvorova (označeni tačkama A, B, C, itd.) i grana koje te čvorove povezuju. Grafovi su od posebnog značaja, jer je njihovom upotrebom moguće modelovanje složene računarske mreže i njene topologije na veoma jednostavan način.

Formalna definicija grafa

Definicija: Jednostavan graf G sastoji se od nepraznog konačnog skupa $V(G)$ čiji elementi se zovu čvorovi i konačnog skupa $E(G)$ različitih parova elemenata $V(G)$ koji se zovu grane.

Uobičajeno je u definicijama i teoremama u teoriji grafova čvorove označavati malim slovima u i v , a grane malim slovima e i f . Pri tome $e=\{u,v\}$ označava granu koja spaja vrhove u i v , što se može kraće zapisati kao $e=uv$. Često se grani u grafu pridružuje još jedan broj koji predstavlja njenu težinu. Ovaj dodatni podatak se najčešće koristi pri modelovanju problema mreže grafom. U ovom radu posmatra se samo jedan graf odjednom, pa nema potrebe posebno naglašavati u oznaci kojem grafu pojednini skup čvorova ili grana pripada. Zato će u nastavku graf biti označavan oznakom $G(V,E)$. Radi jednostavnosti za oznaku čvorova koriste se velika slova A, B, C, itd. Skupovi V i E za primer grafa sa slike 3.1.a su:

$$V = \{A, B, C, D, E\}$$

$$E = \{AB, AD, AE, BC, BD, BE, CD, DE\}$$

Kako bi se bolje razumeli algoritmi koji će biti predstavljeni potrebno je definisati još nekoliko pojmova vezanih uz grafove, a to su prvenstveno susednost čvorova, susednost grana, put u grafu i povezanost grafa [23, 24].

Definicija: Čvorovi u i v su susedni ako postoji grana $e=uv$ u tom grafu koja ih spaja. Grane e i f su susedne ako postoji čvor u u tom grafu koji je njima zajednički.

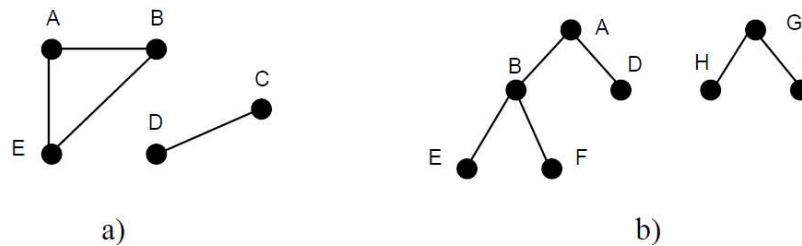
Na primeru sa slike 3.1.a uočava se da su čvorovi A i B susedni, isto tako čvorovi B i C su susedni, itd. Susedne grane na tom istom grafu su npr. AB i AE , jer im je čvor A zajednički.

Definicija: Put u grafu G je konačan niz grana $v_1v_2, v_2v_3, \dots, v_{n-1}v_n$ u kojem su svake dve grane susedne i svi čvorovi su različiti, osim eventualno početnog i krajnjeg. Put se može označiti i kao $v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow \dots \rightarrow v_n$.

Definicija: Graf G je povezan onda i samo onda ako postoji put između svaka dva čvora.

Put u primeru sa slike 3.1.a je npr. $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$.

Svi grafovi na slici 3.1 su povezani, jer postoji put od svakog čvora do svih ostalih čvorova u tim grafovima. Primeri nepovezanih grafova dati su na slici 3.2.

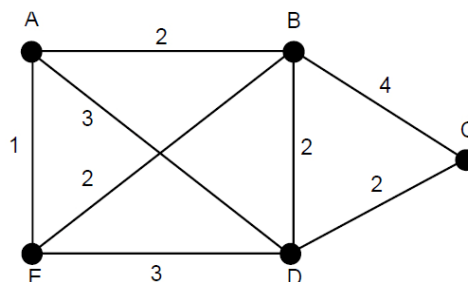


SLIKA 3.2. PRIMERI NEPOVEZANIH GRAFOVA

Kao što je sa slike 3.2.a vidljivo ne postoji niz grana koji bi činio put od vrha D do vrha A .

Najjednostavniji način prikazivanja grafa je tabelom incidencije. To je tabela u kojoj redovi i kolone predstavljaju čvorove grafa, a polje u tabeli na ukrštanju reda i i kolone j predstavlja težinu čvora koji spaja vrhove i i j . Težina grane najčešće se koristi kao mera udaljenosti dva čvora. Za čvorove koji nisu spojeni granom podrazumeva se beskonačna udaljenost ∞ , ali se taj zapis u tabeli incidencije ostavlja praznim.

Budući da na niti jednom grafu do sada nije bila označena udaljenost između čvorova, na slici 3.3 prikazan je graf sa slike 3.1.a na kojem su te udaljenosti označene, kako bi za taj isti graf bilo moguće prikazati tabelu incidencije.



SLIKA 3.3. PRIMER GRAFA SA TEŽINAMA GRANA

Za graf sa slike 3.3 odgovarajuća tabela incidencije data je u tabeli 3.1.

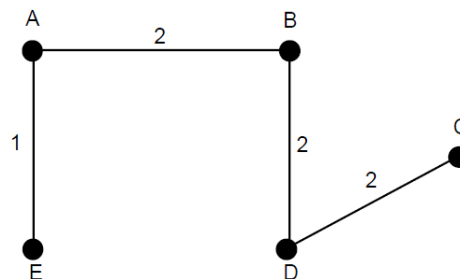
TABELA 3.1. TABELA INCIDENCIJE

	A	B	C	D	E
A	0	2		3	1
B	2	0	4	2	2
C		4	0	2	
D	3	2	2	0	3
E	1	2		3	0

Uočava se da je tabela incidencije za ovakvu vrstu grafa simetrična u odnosu na glavnu dijagonalu. Takođe, primećuje se da se na glavnoj dijagonali pojavljuju samo nule, što znači da je u ovakvom prikazu udaljenost čvora od samog sebe jednaka nuli, što ne mora biti slučaj u svakom modelu mreže koja se predstavlja grafom.

3.2. Razapinjuća stabla

Za predstavljanje razapinjućeg stabla biće korišćen primer sa slike 3.1.a. Graf na toj slici poseduje cilkuse, tj. puteve u grafu koji počinju i završavaju istim čvorom. Ukoliko bi se uklonio dovoljan broj grana iz grafa dobio bi se graf koji nema cirkuluse, pa se takav graf naziva stablo. Primer stabla može se videti na slici 3.1.c. Stablo koje se dobije uklaňanjem određenog broja grana iz grafa, a da pritim dobijeni graf ostane povezan, naziva se razapinjuće stablo.



SLIKA 3.4. MINIMALNO RAZAPINJUJUĆE STABLO

Pronalaženje razapinjućeg stabla jednostavan je postupak, ali u praksi je poželjno da takvo razapinjuće stablo ima i neka dodatna svojstva, kao što su minimalna ili maksimalna suma težina grana. Graf na slici 3.4. predstavlja minimalno razapinjuće stablo dobijeno iz grafa na slici 3.3. Za dobijanje takvih razapinjućih stabala postoje dva poznata algoritma koja su opisana u nastavku [25, 26].

3.2.1. Primov algoritam

Dobijanje razapinjućeg stabla iz zadatog grafa Primovim algoritmom može se prikazati sledećim pseudokodom:

Stablo = $\emptyset$

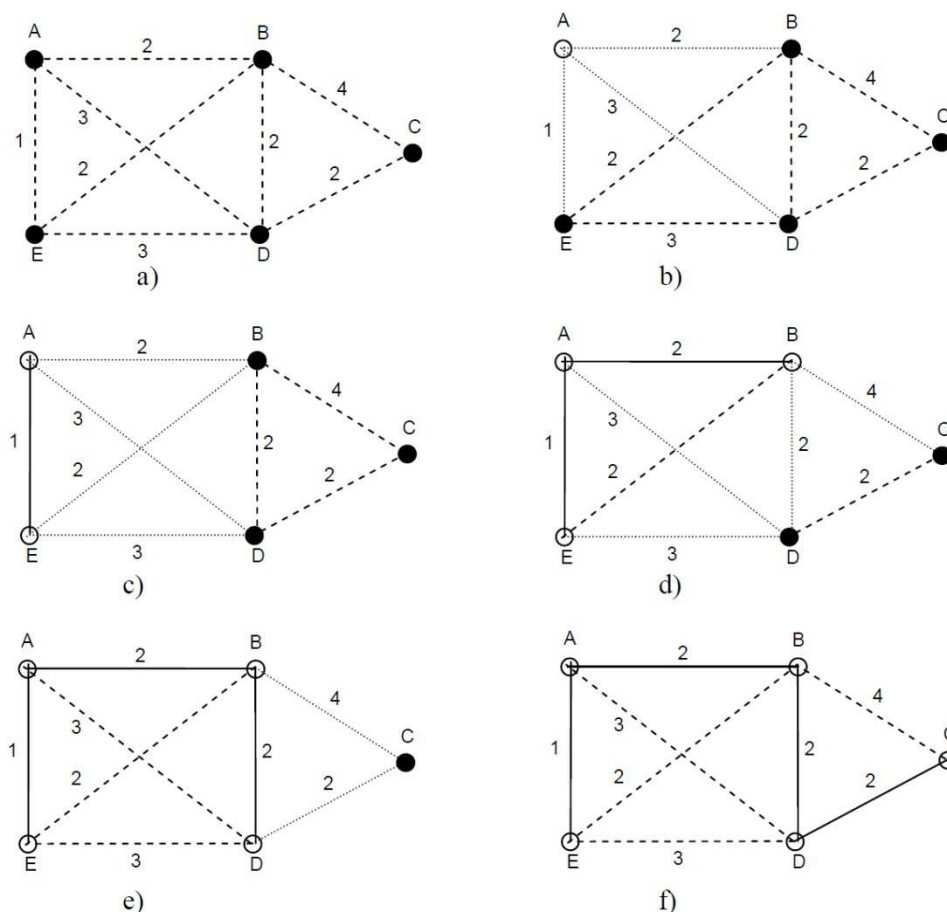
Izaberi proizvoljni čvor iz $V(\text{Graf})$ i stavi ga u $V(\text{Stablo})$.

Dok je $\text{Broj_Čvorova}(\text{Stablo}) < \text{Broj_Čvorova}(\text{Graf})$ ponavlja

Izaberi čvor koji nije u $V(\text{Stablo})$, a susjed je nekom čvoru iz $V(\text{Stablo})$ i pri tome je težina grane koja ih spaja minimalna. Stavi taj čvor zajedno sa njemu pripadajućom granom u Stablo .

Ovim algoritmom pokušava se iz zadatog grafa Graf izgraditi razapinjuće stablo Stablo . U početku je Stablo prazno, pa se počinje graditi dodavanjem proizvoljnog čvora iz skupa čvorova početnog grafa. Postupak se nastavlja dodavanjem grane u Stablo koja ima osobinu da povezuje jedan čvor koji se već nalazi u stablu Stablo i jedan koji se se u njemu ne nalazi, pazeći pri tome da je težina te grane minimalna. Na kraju algoritma graf Stablo predstavlja traženo minimalno razapinjuće stablo.

U nastavku se opisuje postupak izvođenja Primovog algoritma na grafu sa slike 3.3 u svim njegovim koracima. Inicijalno, grane početnog grafa se označavaju isprekidanim linijama (slika 3.5.a), a grane dodate u razapinjuće stablo se označavaju punim linijama. Pri tome, grane koje se posmatraju u datom koraku označavaju se tačkastim linijama. Izvođenje algoritma započinje od čvora A.



SLIKA 3.5. PRIKAZ IZVOĐENJA PRIMOVOG ALGORITMA

Kako bi se lakše sproveo Primov algoritam koristi se tabelarni zapis u svakom njegovom koraku. Za graf se prvo pravi tabela incidencije. Potom se u prvoj koloni tabele incidencije zaokružuje oznaku čvora od koga započinje algoritam i precrtavaju se svi brojevi u koloni u kojoj se taj čvor nalazi, kao što je prikazano u tabeli incidencije 3.2.a. Prvi korak algoritma zapisuje se u tabelu za traženje razapinjućeg stabla, kao što je prikazano u tabeli 3.2.b. U

prvoj koloni zapisuje se naziv čvora koji se nalazi u razapinjućem stablu, u drugoj koloni zapisuje se naziv čvora koji nije u razapinjućem stablu, a susedan je čvoru iz prve kolone. Treća kolona sadrži težinu grane koja ta dva čvora spaja. Kako bi se uštedelo u pisanju za svaki čvor iz razapinjućeg stabla u tabeli se zapisuju samo najbliži susedni čvorovi koji nisu u razapinjućem stablu. To se lako vidi iz reda tabele incidencije pridruženom tom čvoru. Ukoliko postoji više čvorova sa tim svojstvom, svi se upisuju. Nakon popisivanja svih čvorova odabire se čvor koji nije u stablu i čija je pripadajuća grana najmanje težine, i dodaje se u stablo. Naziv tog čvora zapisuje se u četvrtu kolonu tabele, u petu kolonu zapisuje se težina pripadajuće grane, dok se u šestu kolonu upisuje naziv pripadajuće grane. Nakon toga precrtavaju se svi brojevi u koloni tabele incidencije koji pripadaju datom čvoru, kako više ne bi bili korišćeni u proračunu, i zaokružuje se naziv grane u njemu pripadajućem redu matrice incidencije, kao što je prikazano u tabeli incidencije 3.2.c. U sledećim tabelama (tabele 3.2.d-h) dati su svi koraci Primovog algoritma za prethodni primer.

TABELA 3.2. TABELARNI ZAPIS PRIMOVOG ALGORITMA

	A	B	C	D	E
(A)	0	2		3	1
B	2	0	4	2	2
C		4	0	2	
D	3	2	2	0	3
E	1	2		3	0

a)

1	A	E	1	E	1	AE
---	---	---	---	---	---	----

b)

	A	B	C	D	E
(A)	0	2		3	1
B	2	0	4	2	2
C		4	0	2	
D	3	2	2	0	3
(E)	1	2		3	0

c)

1	A	E	1	E	1	AE
2	A	B	2	B	2	AB
	E	B	2			

d)

	A	B	C	D	E
(A)	0	2		3	1
(B)	2	0	4	2	2
C		4	0	2	
D	3	2	2	0	3
(E)	1	2		3	0

e)

1	A	E	1	E	1	AE
2	A	B	2	B	2	AB
	E	B	2			
3	A	D	3	D	2	BD
	B	D	2			
	E	D	3			

f)

	A	B	C	D	E
(A)	0	2		3	1
(B)	2	0	4	2	2
C		4	0	2	
(D)	3	2	2	0	3
(E)	1	2		3	0

g)

1	A	E	1	E	1	AE
2	A	B	2	B	2	AB
	E	B	2			
3	A	D	3	D	2	BD
	B	D	2			
	E	D	3			
4	B	C	4	C	2	DC
	D	C	2			

h)

Kao što je iz tabele 3.2 vidljivo, dodata je još jedna kolona na početku tabele kako bi se videlo koji deo tabele pripada kojem koraku algoritma.

U prvom koraku (slika 3.5.b) posmatraju se grane AB , AD i AE . U tabelarnom zapisu zapisuje se samo najkraća grana koja povezuje čvor A sa nekim drugim čvorom, tj. grana AE i to je prikazano u tabeli 3.2.b. Grana AE najkraća je od svih grana zapisanih u tabeli, pa se stoga dodaje u razapinjujuće stablo. U tabeli incidencije zaokružuje se oznaka čvora E i precrtavaju se svi brojevi u koloni matrice incidencije pridruženi tom čvoru, kao što je prikazano u tabeli 3.2.c. U drugom koraku (slika 3.5.c) posmatraju se grane AB , AD , BE i DE . U tabelarnom zapisu zapisuju se samo grane AB i BE budući da su one najkraće, a potom se grana AB odabire kao najkraća, što je prikazano u tabeli 3.2.d. U tabeli incidencije zaokružuje se oznaka čvora B i precrtavaju se svi brojevi u koloni matrice incidencije pridruženi tom čvoru, kao što je prikazano u tabeli 3.2.e. U trećem koraku posmatraju se grane AD , BD , BC i DE , a u tabelu se zapisuju grane AD , BD i ED . Uočava se da se grana BE više ne uzima u obzir, jer bi njeno dodavanje stvorilo ciklus. Najkraća grana BD se dodaje u razapinjujuće stablo, kao što je prikazano u tabeli 3.2.f. U tabeli incidencije zaokružuje se oznaka čvora D i precrtavaju se svi brojevi u koloni matrice incidencije pridruženi tom čvoru, kao što je prikazano u tabeli 3.2.g. U poslednjem koraku između grana BC i CD se odabire grana CD kao najkraća, što je vidljivo u tabeli 3.2.h. Grane AD i ED se ne razmatraju, jer bi njihovo dodavanje napravilo ciklus. Minimalno razapinjujuće stablo je otkriveno i Primov algoritam se završava.

3.2.2. Kruskalov algoritam

Dobijanje razapinjujućeg stabla iz zadatog grafa Kruskalovim algoritmom može se prikazati sledećim pseudokodom:

Stablo = $\emptyset$

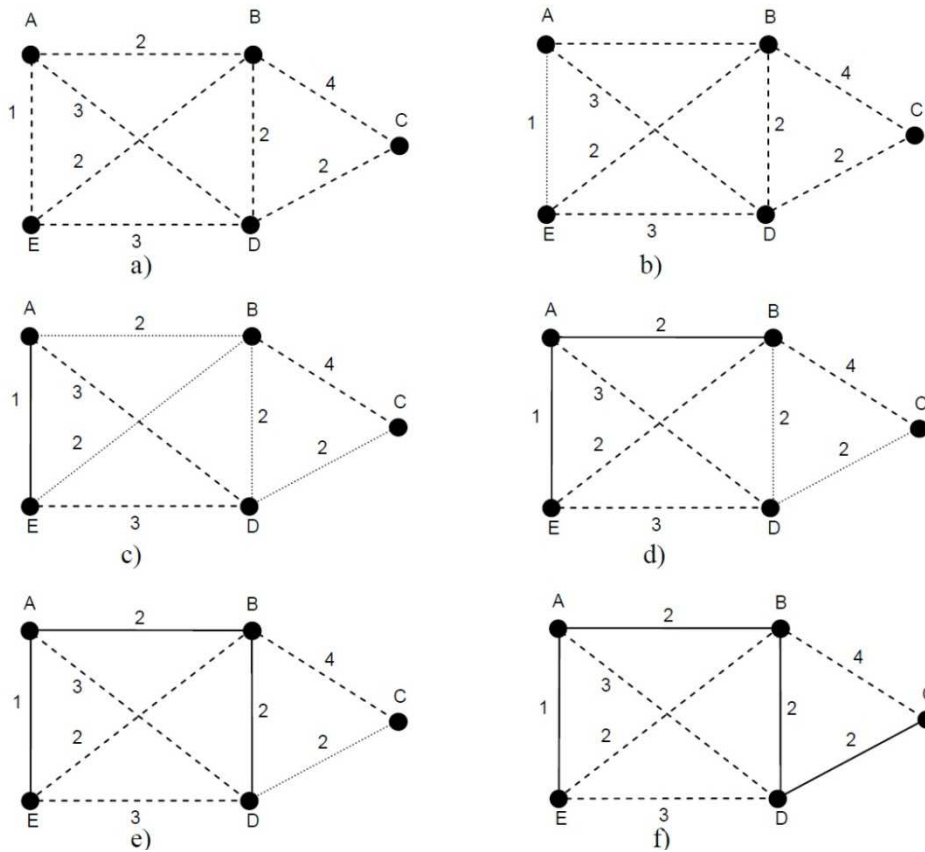
Grane grafa poređaj u padajući niz.

Dok postoji grana čije dodavanje u *Stablo* ne stvara ciklus ponavljaj

 Uzmi najmanju granu koja ne stvara ciklus sa granama iz *Stabla* i dodaj je u *Stablo*.

Kao i kod Primovog algoritma započinje se sa praznim razapinjujućim stablom *Stablo*, i postepeno se dodaju grane iz početnog grafa *Graf* u razapinjujuće stablo. U svakom koraku dodaje se po jedna grana u *Stablo* i to tako da dužina grane koja se dodaje bude najmanja i da pritom ne stvara ciklus sa ostalim granama u *Stablu*. Algoritam se zaustavlja kad se u *Stablo* više ne može dodati grana koja ne bi stvorila ciklus. Na kraju algoritma *Stablo* predstavlja razapinjujuće stablo.

U nastavku je prikazano izvođenje Kruskalovog algoritma u svim njegovim koracima na prethodnom primeru sa slike 3.3.



SLIKA 3.6. PRIKAZ IZVOĐENJA KRUSKALOVOG ALGORITMA

Za razliku od Primovog algoritma, razapinjuće stablo ne mora biti povezano tokom izvođenja Kruskalovog algoritma.

U prvom koraku, prikazanom na slici 3.6.b, u obzir dolazi samo grana AE budući da je njena težina najmanja. U drugom koraku (slika 3.6.c) u obzir dolaze grane AB , BD , BE i CD , pa se među njima odabire proizvoljna grana, npr. grana AB . U trećem koraku (slika 3.6.d) preostaju samo grane BD i CD , te se uzima proizvoljna grana BD . Uočava se da se grana BE ne može staviti u stablo, jer bi se time stvorio ciklus. U četvrtom koraku (slika 3.6.e) preostaje samo grana CD , te se njenim stavljanjem u stablo završava gradnja istog.

Isti postupak može se sprovesti i bez crtanja grafa, jednostavnim poznavanjem samo grana grafa i njihovih težina. Primena ovog postupka na istovetnom primeru sa slike 3.3 započinje ređanjem svih grana u rastući niz po njihovim težinama:

$AE-1, AB-2, BD-2, BE-2, CD-2, AD-3, DE-3, BC-4$

Kako bi se uvidelo da li dodavanje neke grane u stablu čini ciklus pamte se samo skupovi čvorova koji predstavljaju stablo. U početku postoji pet jednočlanih skupova od kojih svaki sadrži jedan čvor. Zbog jednostavnosti zapisa takvi jednočlani skupovi se ne zapisuju u postupku. U prvom koraku u graf se dodaje grana sa najmanjom težinom i potom se izbacuje iz prethodnog niza grana, što se može zapisati na sledeći način:

$\{A, E\} \quad \{AE\} \quad AB-2, BD-2, BE-2, CD-2, AD-3, DE-3, BC-4$

U prvoj koloni nalazi se skup čvorova koji predstavljaju stablo, u drugoj koloni nalaze se grane koje čine stablo, a u trećoj koloni nalaze se grane koje se još uvek mogu ubaciti u stablo. U sledećem koraku ubacuje se druga grana u stablo, grana AB . Budući da se oznake čvora B i čvora A ne nalaze u istom skupu grana u prvoj koloni, ubacivanjem te grane neće se stvoriti ciklus. Zapis ovog koraka izgleda:

$$\{A, B, E\} \quad \{AB, AE\} \quad BD-2, BE-2, CD-2, AD-3, DE-3, BC-4$$

U sledećem koraku ubacuje se grana BD , jer ista ne stvara ciklus sa već ubačenim granama. Zapis ovog koraka je:

$$\{A, B, D, E\} \quad \{AB, AE, BD\} \quad BE-2, CD-2, AD-3, DE-3, BC-4$$

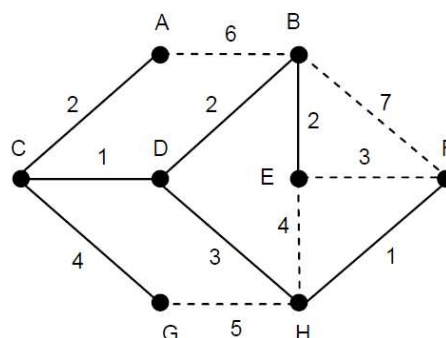
U narednom koraku ne može se ubaciti grana BE , jer se oznake čvorova B i E nalaze u istom skupu čvorova u prvoj koloni. To znači da u stablu već postoji put koji povezuje čvorove B i E , te bi dodavanje još jedne grane koja ih povezuje stvorilo ciklus. Sledeća grana koja dolazi u obzir je grana CD , jer njeno ubacivanje ne stvara ciklus u stablu. Zapis nakon tog koraka je:

$$\{A, B, C, D, E\} \quad \{AB, AE, BD, CD\} \quad AD-3, DE-3, BC-4$$

Budući da je u stablo ubačen maksimalan broj grana Kruskalov algoritam se završava. Važno je uočiti da se u svakom koraku grana koja se ubacuje može birati proizvoljno. Od drugog koraka nadalje bilo je moguće ubaciti u stablo bilo koju granu dužine 2, i time se ne bi narušila minimalnost rešenja, jer se od jednog grafa može napraviti više različitih minimalnih razapinjućih stabala, tj. rešenje nije jedinstveno.

3.3. Najkraće udaljenosti u grafu

Udaljenost u grafu definiše se kao suma težina grana koje sačinjavaju put od početnog do krajnjeg čvora. Budući da u grafu može postojati više različitih puteva između dva čvora njihove dužine ne moraju nužno biti iste. U ovom poglavlju proučavaju se putevi u grafu koji predstavljaju najkraće udaljenosti između dva čvora [24].



SLIKA 3.7. NAIKRAĆE UDALJENOSTI OD ČVORA A

Na slici 3.7 prikazane su minimalne udaljenosti od čvora A do svih ostalih čvorova unutar grafa. Konstruisanje takvog minimalnog puta metodom pokušaja i pogrešaka za manje grafove je jednostavno, ali problemi nastaju kada se broj čvorova grafa počinje povećavati na

više desetina. Zbog toga se koriste posebni algoritmi za pronalaženje najkraćeg puta od jednog čvora do svih ostalih.

3.3.1. Dijkstrin algoritam

Dijkstrin algoritam za pronalaženje najkraćih puteva je u osnovi proširenje Primovog algoritma prikazanog u poglavlju 3.2.1. Ideja je u tome da se gradi stablo koje se sastoji od grana koje čine minimalne puteve od početnog čvora do svih ostalih čvorova. Kao i kod Primovog algoritma u svakom koraku Dijkstrinog algoritma odabire se grana koja spaja čvor koji nije u stablu, ali se pri tome za svaki čvor računa udaljenost od početnog čvora. Od svih grana koje se u jednom koraku mogu izabrati uzima se ona za koju je pripadajući čvor, koji nije u stablu, najmanje udaljen od početnog čvora.

Postupak izvođenja Dijkstrinog algoritma može se predstaviti sledećim pseudokodom:

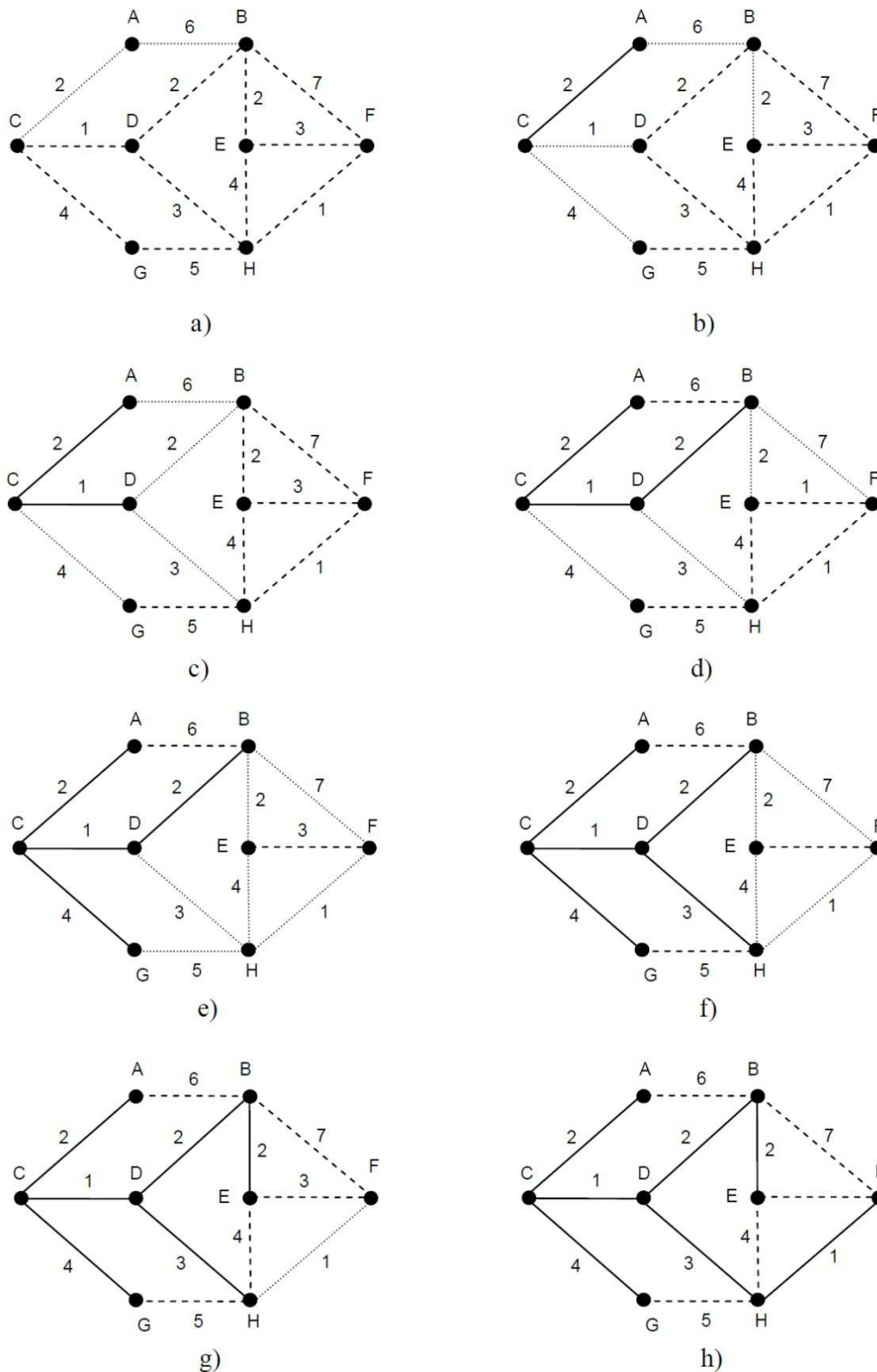
Stablo = $\emptyset$

Stavi u stablo početni čvor i označi ga sa udaljenošću 0.

Dok je Broj_Čvorova(Stablo) < Broj_Čvorova(Graf) ponavljaj

Za sve čvorove koji su susedni nekom čvoru iz stabla izračunaj najmanju udaljenost od početnog čvora kao udaljenost čvora u stablu kojem je taj čvor susedan + težina grane koja ih spaja. Od svih tih čvorova izaberi onaj koji ima najmanju vrednost udaljenosti i stavi ga u stablo zajedno sa pripadajućom granom.

Sledeći primer prikazuje kako su izgrađeni putevi minimalne udaljenosti od čvora A u grafu prikazanom na prethodnoj slici 3.7.



SLIKA 3.8. PRIKAZ IZVOĐENJA DIJKSTRINOG ALGORITMA

U prvom koraku (slika 3.8.a) bira se između grana AB i AC. Izabire se grana AC, jer je udaljenost do čvora C (2) manja nego do čvora B (6). U drugom koraku (slika 3.8.b) bira se između grana AB, CD i CG, te se odabire grana CD, jer je čvor D najbliži (3). U trećem koraku odabire se grana DB, jer je čvor B najmanje udaljen (5). U četvrtom koraku uzima se grana CG, jer je udaljenost do čvora G najmanja (6), a u petom koraku grana DH, jer je čvor H najbliži (6). U poslednja dva koraka odabiru se grane BE, jer je udaljenost do čvora E (7), i grana HF, jer je udaljenost do čvora F (7).

Kako bi se Dijkstrin algoritam lakše sproveo koristi se tabelarni zapis (tabela 3.3) slično kao kod Primovog algoritma. Jedina razlika je ta što se u trećoj koloni upisuje udaljenost od početnog čvora.

TABELA 3.3. TABELARNI ZAPIS DIJKSTRINOGR ALGORITMA

	A	B	C	D	E	F	G	H
A	0	6	2					
B	6	0		2	2	7		
C	2		0	1			4	
D		2	1	0				3
E		2			0	3		4
F		7			3	0		1
G			4				0	5
H				3	4	1	5	0

1	A	C	2	C	2	AC
---	---	---	---	---	---	----

	A	B	C	D	E	F	G	H
A	0	6	2					
B	6	0		2	2	7		
C	2		0	1			4	
D		2	1	0				3
E		2			0	3		4
F		7			3	0		1
G			4				0	5
H				3	4	1	5	0

2	A	B	6	D	3	CD
	C	D	3			

	A	B	C	D	E	F	G	H
A	0	6	2					
B	6	0		2	2	7		
C	2		0	1			4	
D		2	1	0				3
E		2			0	3		4
F		7			3	0		1
G			4				0	5
H				3	4	1	5	0

3	A	B	6	B	5	BD
	C	G	6			
	D	B	5			

	A	B	C	D	E	F	G	H
A	0	6	2					
B	6	0		2	2	7		
C	2		0	1			4	
D		2	1	0				3
E		2			0	3		4
F		7			3	0		1
G			4				0	5
H				3	4	1	5	0

4	B	E	7	G	6	CG
	C	G	6			
	D	H	6			

	A	B	C	D	E	F	G	H
(A)	0	6	2					
(B)	6	0		2	2	7		
(C)	2		0	1			4	
(D)		2	1	0				3
E		2			0	3		4
F		7			3	0		1
(G)			4				0	5
H				3	4	1	5	0

5	B	E	7	H	6	DH
	D	H	6			
	G	H	11			

	A	B	C	D	E	F	G	H
(A)	0	6	2					
(B)	6	0		2	2	7		
(C)	2		0	1			4	
(D)		2	1	0				3
E		2			0	3		4
F		7			3	0		1
(G)			4				0	5
(H)				3	4	1	5	0

6	B	E	7	E	7	BE
	H	F	7			

	A	B	C	D	E	F	G	H
(A)	0	6	2					
(B)	6	0		2	2	7		
(C)	2		0	1			4	
(D)		2	1	0				3
(E)		2			0	3		4
F		7			3	0		1
(G)			4				0	5
(H)				3	4	1	5	0

7	B	F	12	F	7	HF
	E	F	10			
	H	F	7			

Ovim algoritmom veoma jednostavno se dobijaju najmanje udaljenosti od jednog čvora do svih ostalih čvorova u grafu. Međutim, u slučaju kada u grafu postoje grane sa negativnim težinama, Dijkstrin algoritam može se zameniti sa Belman-Fordovim algoritmom.

Problem najkraćeg puta spada u P klasu¹ složenosti. Efikasnost algoritma Dijkstra je $O(n^2)$, gde n predstavlja broj čvorova grafa.

3.3.2. Belman-Fordov algoritam

Belman-Fordov algoritam traži najkraći put za jedan početni čvor u težinskom usmerenom grafu. Za razliku od Dijkstrinog algoritma, kod Belman-Fordovog algoritma graf sme imati negativne težine grana, a sam algoritam ima mogućnost detekcije negativnih ciklusa u grafu. Ovaj algoritam je u osnovi vrlo sličan Dijkstrinom algoritmu, ali umesto odabira čvora sa najmanjom udaljenošću za sledeći čvor, algoritam pristupa svim granama. Ta operacija se

¹ problemi odluke koje je moguće rešiti determinističkim mehanizmom u polinomijalnom vremenu (tj. složenost najgoreg slučaja je ograničena polinomijalnom funkcijom).

izvodi $n-1$ puta, gde je n broj čvorova u grafu. Ta ponavljanja omogućuju precizno propagiranje udaljenosti kroz graf, prvenstveno zbog prirode najkraće putanje koja svaki čvor može posetiti samo jednom.

Algoritam Belman-Ford je implementacija nad grafom koja na osnovu liste čvorova i grana izmeni čvorove tako da njihove udaljenosti i atributi koji predstavljaju njihovog prethodnika sadrže najkraći put. Pseudokod Belman-Fordovog algoritma dat je u nastavku:

Procedura *BelmanFord*(lista čvorova, lista grana, izvorni čvor)

Korak 1 – Inicijalizacija grafa

Za svaki čvor v među čvorovima:

Ako v je izvor onda $v.rastojanje := 0$

Inače $v.rastojanje := \infty$

$v.prethodnik := null$

Korak 2 – Upotreba grana za najkraći put (uv je grana od čvora u do čvora v)

Za i od 1 do broj(čvorova)-1:

Za svaku granu uv među granama:

$u := uv.izvor$

$v := uv.odredište$

Ako $u.rastojanje + uv.težina < v.rastojanje$:

$v.rastojanje := u.rastojanje + uv.težina$

$v.prethodnik := u$

Korak 3 – Provera negativnih ciklusa

Za svaku granu uv među granama:

$u := uv.izvor$

$v := uv.odredište$

Ako $u.rastojanje + uv.težina < v.rastojanje$:

Greška „Graf sadrži negativni ciklus.“

Složenost algoritma Belman-Ford iznosi $O(n*m)$, gde n označava broj čvorova grafa, a m označava broj grana grafa. Algoritam ima lošije performanse od algoritma Dijkstra, ali ponekad je nemoguće modelovati neki problem bez korišćenja negativnih težina grana.

Algoritmi Dijkstra i Belman-Ford svrstavaju se u grupu algoritama koji traže najkraći put iz jednog početnog čvora. Pomoću ovih algoritama veoma jednostavno se dobijaju najmanje udaljenosti od jednog čvora do svih ostalih čvorova u grafu. Međutim, kada su potrebne vrednosti najmanjih udaljenosti između svaka dva čvora u grafu, tada bi jedan od ovih algoritama trebalo ponoviti onoliko puta koliko ima čvorova u grafu. Nije teško primetiti da je to veoma zahtevan posao, te se za te potrebe koristi poseban algoritam iz teorije grafova. Reč je o Flojd-Voršalovom algoritmu pomoću kojeg se veoma jednostavno mogu pronaći vrednosti najmanjih udaljenosti između svih parova čvorova u grafu.

3.3.3. Flojd-Voršalov algoritam

Flojd-Voršalovim algoritmom pronalaze se najmanje udaljenosti između svih parova čvorova u grafu. Ideja algoritma je ispitivanje svih mogućih puteva u grafu, ali se pri takvom

ispitivanju koristi činjenica da ovakav problem ima optimalnu podstrukturu, te da se do ukupnog minimuma može doći spajanjem minimuma problema manjeg reda. Ovaj algoritam je tipičan primer dinamičkog programiranja, a dokaz da je rezultat izvođenja algoritma ispravan je veoma složen. Floyd-Voršalov algoritam može se prikazati pseudokodom, pri čemu je matrica incidencije označena sa $G[i,j]$:

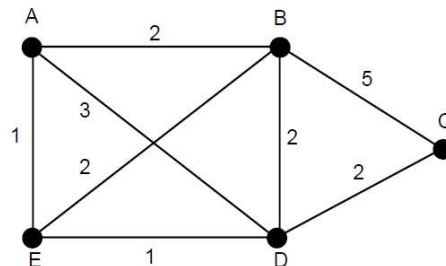
```

Za čvor_kroz = 1 do broj_čvorova
  Za čvor_od = 1 do broj_čvorova
    Ako  $G[\text{čvor\_od}, \text{čvor\_kroz}] \neq \infty$ 
      Za čvor_do = 1 do broj_čvorova
        Ako  $G[\text{čvor\_kroz}, \text{čvor\_do}] \neq \infty$ 
          Ako  $G[\text{čvor\_od}, \text{čvor\_do}] = \infty$  ili
             $G[\text{čvor\_od}, \text{čvor\_do}] > G[\text{čvor\_od}, \text{čvor\_kroz}] + G[\text{čvor\_kroz}, \text{čvor\_do}]$ 
               $G[\text{čvor\_od}, \text{čvor\_do}] = G[\text{čvor\_od}, \text{čvor\_kroz}] + G[\text{čvor\_kroz}, \text{čvor\_do}]$ 

```

Ovako napisano algoritam izgleda veoma složeno, ali u suštini je veoma jednostavan. Za svaki čvor čvor_kroz iz grafa pokušava se poboljšati put između neka druga dva čvora čvor_od i čvor_do , pri čemu se proverava da li se udaljenost smanjuje ako put između ta dva čvora prođe kroz čvor_kroz . Algoritam tokom izvođenja menja tabelu incidencije, tako da se na kraju izvođenja algoritma u tabeli incidencije na poziciji i,j nalazi najmanja udaljenost između čvorova i i j .

Floyd-Voršalov algoritam će biti prikazan na manjem primeru budući da broj koraka raste sa brojem čvorova grafa. Graf za prikazivanje izvođenja algoritma prikazan je na slici 3.9, a njemu odgovarajuća tabela incidencije data je u tabeli 3.4.



SLIKA 3.9. GRAF ZA PRIKAZ FLOYD-VORŠALOVOG ALGORITMA

TABELA 3.4. TABELA INCIDENCIJE ZADATOG GRAFA

	A	B	C	D	E
A	0	2		3	1
B	2	0	5	2	2
C		5	0	2	
D	3	2	2	0	1
E	1	2		1	0

U prvom koraku uzima se čvor A za čvor kroz koji će se pokušati poboljšati put između ostalih čvorova, te se redom zapisuju putevi koji se žele poboljšati.

$B \rightarrow C$ – Ne može se poboljšati, jer trenutno ne postoji put od A do C

$B \rightarrow D$ – Ne poboljšava se, jer je $BA + AD = 5$, a već postojeća vrednost je 2

$B \rightarrow E$ – Ne može se poboljšati već postojeći put

$C \rightarrow D$ – Ne može se poboljšati, jer trenutno ne postoji put od C do A
 $C \rightarrow E$ – Ne može se poboljšati, jer trenutno ne postoji put od C do A
 $D \rightarrow E$ – Ne može se poboljšati već postojeći put

Uočljivo je da prvi korak uopšte nije promenio matricu incidencije, te da se sa čvorom A ne mogu poboljšati već postojeći putevi. Prelazi se na drugi korak sa čvorom B .

$A \rightarrow C$ – Budući da put ne postoji, biće stvoren i označen dužinom 7
 $A \rightarrow D$ – Ne može se poboljšati već postojeći put
 $A \rightarrow E$ – Ne može se poboljšati već postojeći put
 $C \rightarrow D$ – Ne može se poboljšati već postojeći put
 $C \rightarrow E$ – Budući da put ne postoji, biće stvoren i označen dužinom 7
 $D \rightarrow E$ – Ne može se poboljšati već postojeći put

TABELA 3.5. MATRICA INCIDENCIJE NAKON DRUGOG KORAKA

	A	B	C	D	E
A	0	2	7	3	1
B	2	0	5	2	2
C	7	5	0	2	
D	3	2	2	0	1
E	1	2		1	0

U trećem koraku koristi se čvor C .

$A \rightarrow B$ – Ne može se poboljšati već postojeći put
 $A \rightarrow D$ – Ne može se poboljšati već postojeći put
 $A \rightarrow E$ – Ne može se poboljšati već postojeći put
 $B \rightarrow D$ – Ne može se poboljšati već postojeći put
 $B \rightarrow E$ – Ne može se poboljšati već postojeći put
 $D \rightarrow E$ – Ne može se poboljšati već postojeći put

Treći korak takođe nije ništa promenio, a algoritam se nastavlja sa čvorom D .

$A \rightarrow B$ – Ne može se poboljšati već postojeći put
 $A \rightarrow C$ – Udaljenost $AD + DC = 5$ što je manje od 7, te se menja tabela incidencije
 $A \rightarrow E$ – Ne može se poboljšati već postojeći put
 $B \rightarrow C$ – Udaljenost $BD + DC = 4$ što je manje od 5, te se menja tabela incidencije
 $B \rightarrow E$ – Ne može se poboljšati već postojeći put
 $C \rightarrow E$ – Budući da put ne postoji, biće stvoren i označen dužinom 3

TABELA 3.6. MATRICA INCIDENCIJE NAKON ČETVRTOG KORAKA

	A	B	C	D	E
A	0	2	5	3	1
B	2	0	4	2	2
C	5	4	0	2	3
D	3	2	2	0	1
E	1	2	3	1	0

U poslednjem petom koraku koristi se čvor E .

$A \rightarrow B$ – Ne može se poboljšati već postojeći put
 $A \rightarrow C$ – Udaljenost $AE + EC = 4$ što je manje od 5, te se menja tabela incidencije
 $A \rightarrow D$ – Udaljenost $AE + ED = 2$ što je manje od 3, te se menja tabela incidencije
 $B \rightarrow C$ – Ne može se poboljšati već postojeći put
 $B \rightarrow D$ – Ne može se poboljšati već postojeći put
 $C \rightarrow D$ – Ne može se poboljšati već postojeći put

TABELA 3.7. MATRICA INCIDENCIJE NAKON PETOG KORAKA

	A	B	C	D	E
A	0	2	4	2	1
B	2	0	4	2	2
C	4	4	0	2	3
D	2	2	2	0	1
E	1	2	3	1	0

Kao što je iz matrice incidencije vidljivo dobijene su vrednosti minimalnih udaljenosti za sve parove čvorova u grafu, međutim ne zna se kako se te minimalne udaljenosti ostvaruju. Praćenjem svih koraka izvođenja Floyd-Voršalovog algoritma moguće je rekonstruisati puteve tako da se isprate promene težine u matrici incidencije u svakom koraku.

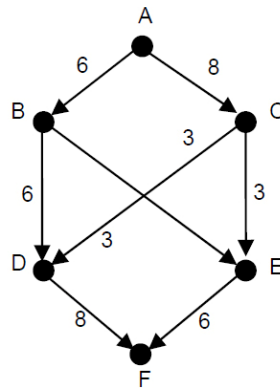
- 2: $A \rightarrow C \equiv A \rightarrow B \rightarrow C$
 $C \rightarrow E \equiv C \rightarrow B \rightarrow E$
 4: $A \rightarrow C \equiv A \rightarrow D \rightarrow C$
 $B \rightarrow C \equiv B \rightarrow D \rightarrow C$
 $C \rightarrow E \equiv C \rightarrow D \rightarrow E$
 5: $A \rightarrow C \equiv A \rightarrow E \rightarrow C \equiv A \rightarrow E \rightarrow D \rightarrow C$
 $A \rightarrow D \equiv A \rightarrow E \rightarrow D$

Ostali putevi koji nisu navedeni ostvareni su direktnom vezom, tj. samo jednom granom.

Efikasnost Floyd-Voršalovog algoritma iznosi $O(n^3)$, gde je n broj čvorova u grafu.

3.4. Maksimalni protok u mreži

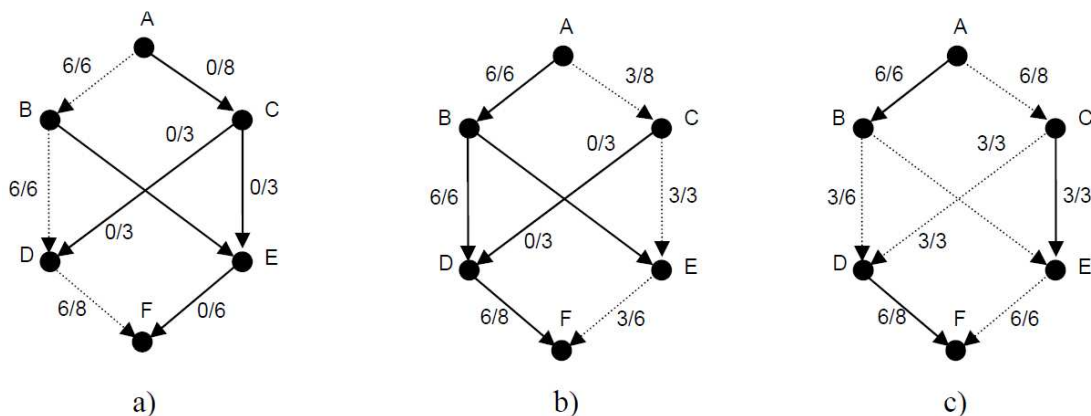
Kao što je već rečeno, grafovima je moguće veoma jednostavno modelovati probleme iz svakodnevnog života. Jedna od uobičajenih primena grafova je opisivanje mreže komunikacionih čvorova, cevovoda, saobraćajnica i sličnih protočnih struktura [23, 24]. Primer mreže komunikacionih čvorova sa unidirekcionim usmerenim vezama opisan grafom dat je na slici 3.10.



SLIKA 3.10. MODEL MREŽE KOMUNIKACIONIH ČVOROVA PRIKAZAN GRAFOM

Kao što je sa slike 3.10 vidljivo grane koje prikazuju unidirekzione usmerene veze označene su strelicama kako bi se naglasilo da paketi kroz tu granu mogu teći samo u jednom smeru. Takav graf naziva se usmerenim težinskim grafom, jer su mu grane usmerene i svaka od njih ima odgovarajuću težinu, odnosno u ovom modelu to se naziva kapacitetom (propusnim opsegom). Čvor iz koga sve strelice izlaze naziva se izvor, u primeru sa slike 3.10 to je čvor A, a onaj u koji su usmerene sve strelice naziva se ponor, u primeru sa slike 3.10 to je čvor F. Problem koji se prirodno nameće je koliki kapacitet svake grane mora biti iskorišćen kako bi ukupni protok od izvora do ponora bio maksimalan. Rešenje takvog problema za veće mreže nije trivijalno, te se za te potrebe koriste razni algoritmi kako bi se takav problem efikasno rešio.

Za prikladno opisivanje problema, težina svake grane proširuje se na dva broja od kojih drugi broj govori koliki je kapacitet grane, tj. koliki je maksimalni protok kroz tu granu. Prvi broj označava koliki deo kapaciteta grane se koristi, te se to može nazvati protokom kroz tu granu. Na slici 3.11 u svega nekoliko koraka prikazano je maksimizovanje protoka kroz zadatu mrežu.



SLIKA 3.11. ODREĐIVANJE MAKSIMALNOG PROTOKA KROZ MREŽU

Kao što je vidljivo sa slike 3.11 povećavanje protoka sprovodi se postupno sve dok se protok više ne može povećati. U prvom koraku (slika 3.11.a) povećava se protok kroz grane AB, BD i DF sa 0 na 6. U drugom koraku (slika 3.11.b) povećava se protok kroz grane AC, CE i EF sa 0 na 3. U trećem koraku (slika 3.11.c) povećanje protoka je malo komplikovanije i sastoji se u tome da se protok kroz granu BD smanji za 3, te da se taj deo protoka preusmeri na grane BE i EF. Budući da sada u čvoru D postoji manjak, taj manjak se nadoknađuje dodatnim protokom kroz grane AC i CD. Kao što je iz ovog kratkog primera vidljivo, rešenje ovakvog

problema nije trivijalno. Algoritam koji efikasno rešava ovaj problem je Ford-Fulkersonov algoritam [25, 26].

3.4.1. Ford-Fulkersonov algoritam

Kao što je već rečeno Ford-Fulkersonovim algoritmom pronalazi se maksimalni protok kroz zadatu mrežu. Algoritam se sprovodi u nizu koraka, tako što se u svakom koraku povećava ukupni protok u mreži sve dok se ne dostigne maksimum.

Ideja algoritma je pronaći put u grafu koji povećava trenutni protok, i tada promeniti protok po granama tog puta čime se povećava ukupni protok. Kad više nije moguće pronaći put koji bi poboljšao protok algoritam se završava. Ford-Fulkersonov algoritam može se prikazati sledećim pseudokodom:

Protok kroz sve grane stavi na 0.

Dok možeš pronaći put koji poboljšava protok

Promeni protok kroz grane pronađenog puta.

U samom algoritmu nije specificirano kako pronaći put koji poboljšava protok, a to znači da bilo koji postupak pronalaženja tog puta daje ispravno rešenje. Međutim, broj koraka u kojem se postiže rešenje u velikoj meri zavisi od načina pronalaženja takvog puta. Idealno bi bilo da se u svakom koraku pronađe put koji maksimalno povećava protok, a to se može učiniti algoritmom koji je veoma sličan Dijkstrinom algoritmu. No pre svega trebalo bi detaljnije objasniti kako se označava protok kroz grane. Kao što je iz samog problema vidljivo koriste se usmerene grane kako bi se opisalo da paketi mogu prolaziti samo u jednom smeru. Zbog potreba algoritma uvode se i grane u suprotnom smeru koje imaju iste vrednosti protoka i kapaciteta kao i grane u pravom smeru, ali sa suprotnim predznakom.

Mogući iznos za koji je moguće povećati protok kroz neku granu u pravom smeru je jednak razlici kapaciteta i trenutnog protoka, tj. moguće je kroz tu granu progurati još paketa sve dok se ne postigne pun iznos kapaciteta. Ono što je možda zbunjujuća činjenica je mogućnost povećanja protoka kroz granu u suprotnom smeru za apsolutnu vrednost njegovog protoka, čime se smanjuje protok paketa u pravom smeru. Zašto bi to trebalo povećati ukupni protok uviđa se iz primera ostvarivanja maksimalnog protoka na mreži sa slike 3.10.

Ideja algoritma koji se primenjuje je pronalaženje vrednosti maksimalnih protoka od izvornog čvora do svakog drugog čvora u grafu. Razlika u odnosu na Dijkstrin algoritam uočava se u trećoj koloni tabele incidencije koja se gradi. U tu kolonu upisuje se manji od brojeva koji predstavlja protok do čvora koji se nalazi u stablu i mogućeg protoka kroz granu koja vodi od čvora u stablu do čvora koji nije u stablu. U nastavku je prikazano izvođenje Ford-Fulkersonovog algoritma na zadatoj mreži sa slike 3.10.

Na početku algoritma pravi se tabela incidencije (tabela 3.8) koja predstavlja protok kroz mrežu. Svaka oznaka predstavlja protok/kapacitet grane.

TABELA 3.8. TABELA INCIDENCIJE

	A	B	C	D	E	F
A		0/6	0/8			
B	0/-6			0/6	0/3	
C	0/-8			0/3	0/3	
D		0/-6	0/-3			0/8
E		0/-3	0/-3			0/6
F				0/-8	0/-6	

Na osnovu te tabele kreira se tabela incidencije (tabela 3.9) sa kojom se pronalazi put koji povećava protok. Vrednosti u tabeli prikazuju koliko je moguće povećati protok kroz odgovarajuću granu. Grane kod kojih je iznos za koji je moguće povećati protok 0 ne uzimaju u obzir u algoritmu. U svakom koraku Ford-Fulkersonovog algoritma umesto najmanje vrednosti, kao što je to bio slučaj u Dijkstrinom algoritmu, u pripadajućem redu posmatranog čvora uzima se najveća vrednost, jer se traži maksimalno povećavanje protoka. U susednoj tabeli sa desne strane (tabela 3.9) prikazani su koraci algoritma.

TABELA 3.9. TABELA INCIDENCIJE U PRVOM KORAKU

	A	B	C	D	E	F
A		6	8			
B				6	3	
C				3	3	
D						8
E						6
F						

1	A	C	8	C	8	AC
2	A	B	6	B	8	AB
	C	D	3			
	C	E	3			
3	B	D	6	D	6	BD
	C	D	3			
4	B	E	3	F	6	DF
	C	E	3			
	D	F	6			

Postupak se zaustavlja kada se dostigne ponor, tj. čvor *F*. Iz tabele 3.9 vidljivo je da maksimalno povećanje protoka iznosi 6 i to kroz put $A \rightarrow B \rightarrow D \rightarrow F$. Shodno tome menjaju se protoci kroz te grane, i te vrednosti se unose u početnu tabelu incidencije za protoke (tabela 3.10).

TABELA 3.10. TABELA INCIDENCIJE (2)

	A	B	C	D	E	F
A		6/6	0/8			
B	-6/-6			6/6	0/3	
C	0/-8			0/3	0/3	
D		-6/-6	0/-3			6/8
E		0/-3	0/-3			0/6
F				-6/-8	0/-6	

Protok koji je ovom tabelom definisan prikazan je na slici 3.11.a. Iz ove tabele kreira se nova tabela incidencije (tabela 3.11) uz pomoć koje se traže dalja povećanja protoka.

TABELA 3.11. TABELA INCIDENCIJE U DRUGOM KORAKU

	A	B	C	D	E	F
A			8			
B	6				3	
C				3	3	
D		6				2
E						6
F				6		

1	A	C	8	C	8	AC
2	C	D	3	D	3	CD
	C	E	3			
3	C	E	3	E	3	CE
	D	B	3			
4	D	B	3	B	3	DB
	E	F	3			
5	E	F	3	F	3	EF

U ovom koraku protok se može povećati za 3 i to kroz put $A \rightarrow C \rightarrow E \rightarrow F$. Promenjena tabela incidencije za protoke je prikazana u tabeli 3.12.

TABELA 3.12. TABELA INCIDENCIJE (3)

	A	B	C	D	E	F
A		6/6	3/8			
B	-6/-6			6/6	0/3	
C	-3/-8			0/3	0/3	
D		-6/-6	0/-3			6/8
E		0/-3	-3/-3			3/6
F				-6/-8	-3/-6	

Tabela 3.12 prikazuje stanje na slici 3.11.b. Iz ove tabele kreira se sledeća tabela incidencije za povećanje protoka (tabela 3.13).

TABELA 3.13. TABELA INCIDENCIJE U TREĆEM KORAKU

	A	B	C	D	E	F
A			5			
B	6				3	
C	3			3		
D		6				2
E			3			3
F				6	3	

1	A	C	5	C	5	AC
2	C	D	3	D	3	CD
3	D	B	3	B	3	DB
4	B	E	3	E	3	BE
	D	F	2			
5	D	F	2	F	3	EF
	E	F	3			

U ovom koraku može se povećati protok za 3 i to kroz put $A \rightarrow C \rightarrow B \rightarrow E \rightarrow F$. Kao što je vidljivo sa slike 3.11.c granom BD se prolazi u suprotnom smeru, što zapravo znači da se time smanjuje protok kroz tu granu. Ono što nije odmah očigledno je da se takvim smanjivanjem postiže povećanje ukupnog protoka. Sa slike 3.11.c uočljivo je da se taj deo protoka kroz granu BD nadoknađuje protokom iz grane CD , a deo koji više ne prolazi kroz granu BD sada prolazi granom BE i time povećava ukupni protok.

TABELA 3.14. TABELA INCIDENCIJE (4)

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>
<i>A</i>		6/6	6/8			
<i>B</i>	-6/-6			3/6	3/3	
<i>C</i>	-6/-8			3/3	3/3	
<i>D</i>		-3/-6	-3/-3			6/8
<i>E</i>		-3/-3	-3/-3			6/6
<i>F</i>				-6/-8	-6/-6	

Tabela 3.14 prikazuje stanje na slici 3.11.c.

U narednom koraku pokušava se dalje povećanje protoka, na osnovu čega se kreira tabela 3.15.

TABELA 3.15. TABELA INCIDENCIJE U ČETVRTOM KORAKU

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>
<i>A</i>			3			
<i>B</i>	6			3		
<i>C</i>	6			3		
<i>D</i>		3	3			2
<i>E</i>		3	3			
<i>F</i>				6	6	

1	<i>A</i>	<i>C</i>	2	<i>C</i>	2	<i>AC</i>
---	----------	----------	---	----------	---	-----------

Kao što je iz postupka u tabeli 3.15 vidljivo, nije moguće pronaći put kojim bi se povećao protok kroz mrežu, pa se Ford-Fulkersonov algoritam tu završava.

DRUGI DEO

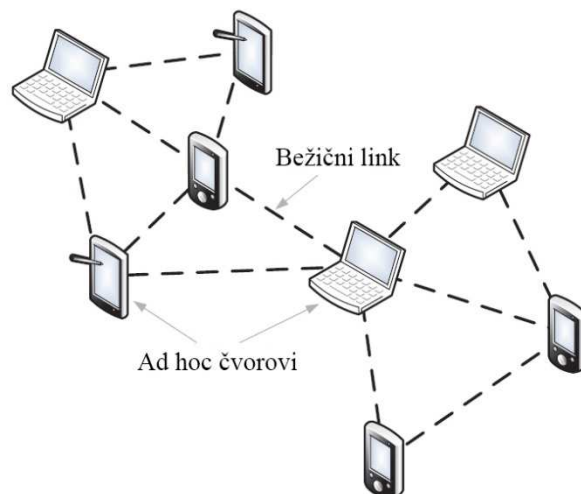
***RUTING PROTOKOLI U
MOBILNIM AD HOC MREŽAMA***

4. MOBILNE AD HOC MREŽE

Ad hoc bežične mreže predmet su značajnog istraživanja. Mobilne ad hoc mreže (MANET) su naziv međunarodne radne grupe u okviru Internet inženjering radne grupe (engl. *Internet Engineering Task Force, IETF*) koja za cilj ima uvođenje MANET rutinog specifikacija u Internet standarde [27].

Istorija mobilnih ad hoc mreža (MANET) može se pratiti unazad do ranih 1970-tih godina kada je DARPA razvila paketske radio mreže (engl. *Packet Radio Networks, PRNET*) [28]. Ovo je na kraju evoluiralo u SURAN (engl. *Survivable Adaptive Radio Networks*) program u ranim 1980-tim [29]. Cilj ovih programa bilo je paketsko umreženje različitih terminala (engl. *hosts*), koje se može koristiti za vojne operacije u mobilnim i neprijateljskim sredinama. Ova vrsta mreža je kreirana sa ciljem da se izbegne potreba za centralnom bazom za komunikaciju. Od ovih mreža je bilo očekivano da prosleđuju informacije na brz i stabilan način, kao i da pokriju najveći deo mogućeg opsega bez potrebe za prethodnom infrastrukturom.

Mobilna ad hoc mreža (MANET) je dinamična mreža koja omogućuje bežično umreženje u pokretu bez potrebe za prethodno izgrađenom mrežnom infrastrukturom, kao što je prikazano na slici 4.1. Sastoji se od pokretnih čvorova, pri čemu čvor može biti čovek koji nosi ručni računar opremljenim sa odgovarajućim uređajem za bežičnu komunikaciju, laptop računar, robot bez posade, tj. svako ko je opremljen sa odgovarajućom opremom za bežičnu komunikaciju [30]. MANET mreže funkcionišu bez centralizovane administracije, a čvorovi međusobno sarađuju kako bi obezbedili povezivanje.



SLIKA 4.1. AD HOC MREŽA SASTAVLJENA OD UREĐAJA POVEZANIH BEŽIČNIM PUTEM

Ad hoc bežična mreža je mreža sa bežičnim uređajima koja se formira i funkcioniše bez centralnog upravljanja i uspostavljene infrastrukture. Dakle, ad hoc bežični uređaji moraju biti u stanju da detektuju prisustvo drugih uređaja u cilju formiranja takve mreže u letu. U ad hoc bežičnim mrežama bežični uređaji mogu komunicirati sa drugim uređajima u okviru, ili van okvira njihovog radio spektra u samoorganizujućem obliku. Da bi podržali takve

komunikacije, bežični uređaji u ad hoc bežičnim mrežama funkcionišu istovremeno i kao terminali i kao ruteri, što znači da oni mogu prosleđivati pakete podataka od izvora do odredišta.

Komunikacija se obavlja preko bežičnih veza pomoću bežičnih omnidirekcionih radio interfejsa. MANET mreže su veoma dinamične u smislu da se mreža formira na spontan i privremen način. Čvorovi se mogu nasumično pridružiti i napustiti mrežu, i mogu biti pokretni. Zbog dinamične prirode mreže ne postoji centralizovano upravljanje, već svi čvorovi samostalno i podjednako sarađuju na distribuirani način kako bi formirali mrežu sa preskocima (engl. *multihop*). To podrazumeva da čvor ima ulogu terminala u komunikaciji s kraja na kraj (engl. *end-to-end*), i istovremeno ulogu rutera za prosleđivanje paketa podataka do drugih terminala koji nisu u opsegu direktnog prenosa svojih izvornih čvorova. MANET mreža može raditi kao samostalna mreža, ili se može integrisati sa spoljnim mrežama kao što je globalna Internet mreža preko gejtveja (engl. *gateway*).

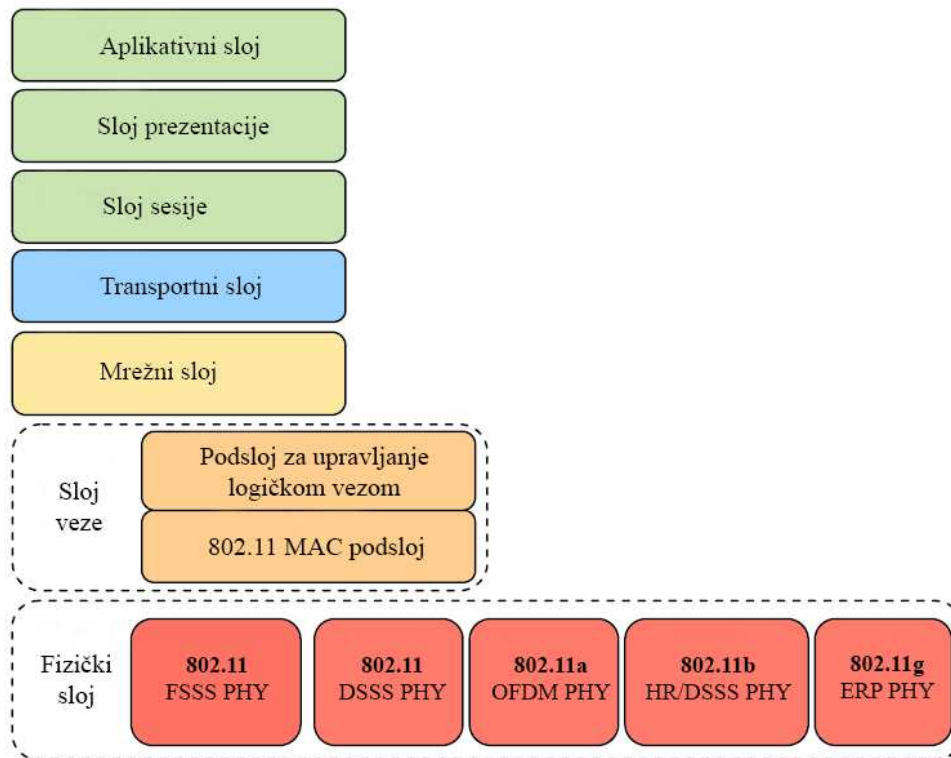
Čvorovi u takvoj mreži mogu biti vrlo pokretni što uzrokuje vrlo brzu promenu položaja čvorova, a time i stvaranje i raskidanje veza među njima. S obzirom da su čvorovi neprestano u pokretu, topologija mreže je promenljiva. MANET mreža je sposobna da izlaskom nekog čvora ili više njih iz mreže ne izgubi poslate pakete, već da sama pronađe novi put do odredišnog čvora.

Svrha MANET mreža je da osiguraju mrežnu konektivnost koja se može odmah postaviti u proizvoljnom komunikacionom okruženju, kao i da se brzo prilagodi na topološke promene u mreži.

Primena ad hoc bežičnih mreža se može naći u mnogim scenarijima. Pošto im ne treba uspostavljena infrastruktura mogu biti brzo raspoređene. Upotreba ad hoc bežičnih mreža se može naći na prostorima gde se standardna mrežna infrastruktura teško može razviti, kao što su vojna borbeno polje, scenariji katastrofa i spasilačkih operacija. S obzirom da se bežične tehnologije neprestano razvijaju, primena ad hoc bežičnih mreža postaje moguća i u civilnom okruženju komunikacije sa ličnim uređajima, kao što su PDA uređaji, mobilni telefoni i prenosni računari.

4.1. MANET i OSI referentni model

Međunarodna organizacija za standardizaciju (engl. *International Standardization Organization, ISO*) je u ranim 1980-tim predložila povezivanje otvorenih sistema (engl. *Open Systems Interconnection, OSI*) kao referentni model [31] koji je pre svega dizajniran da omogući kompatibilnost i komunikaciju između različitih proizvođača kompjuterske opreme. Slojevita OSI arhitektura predstavlja okvir za izgradnju modularnih sistema (slika 4.2). Ona deli mrežne funkcionalnosti koje su uključene u pripremanje prenosa podataka s kraja na kraj (engl. *end-to-end*) u hijerarhijskim nivoima koji sadrže podzadatke (podfunkcije). OSI model definiše sedam slojeva u hijerarhiji koja ide od fizičkog do aplikativnog nivoa. Danas, OSI model je i dalje referentni model koji se često koristi da opiše i navede različite nivoe mrežnih protokola i njihove međusobne interakcije. Komunikacioni mehanizmi koji se odnose na MANET mrežnu arhitekturu su uglavnom povezani sa protokolima koji rade na slojevima 1 do 3 OSI referentnog modela. Viši slojevi su aktivni samo na izvornom i odredišnom čvoru.



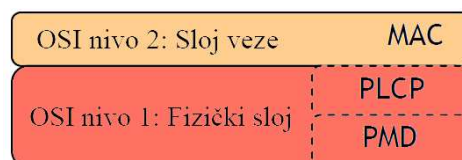
SLIKA 4.2. OSI REFERENTNI MODEL I POREĐENJE SA MANET (IEEE 802.11) PROTOKOLIMA

4.1.1. Fizički sloj

Fizički sloj (engl. *Physical, PHY*) IEEE 802.11 standarda [32, 33] služi kao interfejs između MAC podsloja i bežičnog medijuma gde se prenose okviri. Ona obezbeđuje mehanizme za detekciju bežičnog kanala i ukazuje na MAC podsloj kada je signal detektovan ili kada je kanal u stanju mirovanja. Ovaj mehanizam je poznat kao procena slobodnog kanala (engl. *Clear Channel Assessment, CCA*). Kao što je prikazano na slici 4.3, fizički sloj je podeljen na dva podsloja:

- protokol konvergencije fizičkog sloja (engl. *Physical Layer Convergence Protocol, PLCP*) podsloj, i
- zavistan od fizičkog medijuma (engl. *Physical Medium Dependent, PMD*) podsloj.

PLCP apstrahuje funkcionalnosti poput statusa kanala koji fizički sloj mora da ponudi MAC podsloju, dok PMD upravlja kodiranjem, dekodiranjem i modulacijom signala.



SLIKA 4.3. LOGIČKA STRUKTURA FIZIČKOG SLOJA

Tri PHY standarda u okviru IEEE 802.11 protokola prvobitno su definisani 1997. godine [34]. Prva dva, skokovito frekventno širenje spektra (engl. *Frequency Hopping Spread Spectrum, FHSS*) i direktno sekvencijalno širenje spektra (engl. *Direct Sequence Spread*

Spectrum, DSSS), koriste radio frekventni (RF) pojas elektromagnetnog spektra, a treći standard koristi infracrveni opseg (IR).

FHSS standard koristi skup uskih kanala za prenos podataka. Sistem „skače“ preko svih kanala u unapred određenom redosledu. Na primer, 2,4 GHz frekvencijski opseg je podeljen na 70 kanala od po 1 MHz svaki. Svakih 20 do 400 milisekundi prenos „skače“ na novi kanal prateći unapred određeni ciklični obrazac. Sistem radi na 1 Mbit/s korišćenjem 2 nivoa Gausove digitalne frekvencijske modulacije (engl. *Gaussian Frequency-Shift Keying*, 2GFSK) i 2 Mbit/s koristeći 4 nivoa GFSK (4GFSK) [34, 35].

Kod DSSS standarda tok podataka se širi preko većeg frekventnog opsega primenom sekvence podintervala [34, 36]. DSSS radi u 2,4 GHz radio frekventnom opsegu, pri brzinama prenosa podataka od 1 Mbit/s koristeći modulacionu šemu diferencijalnog binarnog faznog pomeraja (engl. *Differential Binary Phase Shift Keying*, DBPSK) i 2 Mbit/s pomoću modulacione šeme diferencijalnog kvadrarnog faznog pomeraja (engl. *Differential Quadrature Phase Shift Keying*, DQPSK).

Treća specifikacija IEEE 802.11 standarda fizičkog sloja zasniva se na infracrvenom (IR) opsegu koji podržava stope prenosa podataka do 4 Mbit/s. Međutim, modulacione tehnike za RF veze, kao one koje se koriste u DSSS, FHSS i DBPSK standardima, izuzetno je teško iskoristiti u bežičnim infracrvenim vezama zbog teškoća prikupljanja signala u jednom elektromagnetskom režimu. Kao posledica toga, IR sistemi koriste modulaciju intenziteta sa direktnom detekcijom, kao što su impulsno-poziciona modulacija (engl. *Pulse Position Modulation*, PPM) i uključi-isključi modulacija (engl. *On-Off Keying*, OOK) [37]. Uopšteno, radio frekventni opseg je preferiraniji u odnosu na infracrveni zbog svoje fleksibilnosti, podrške za mobilnost i sposobnosti da prodre kroz zidove i neprozirne objekte.

Najnovija dostignuća u tehnologiji čipova, RF kodiranja signala i modulacionih tehnika IEEE 802.11 operativnih uređaja stvorilo je dodatne fizičke slojeve: 802.11a PHY, 802.11b PHY i 802.11g PHY [34].

802.11a PHY standard koristi modulacionu šemu ortogonalnog multipleksiranja podelom frekvencije (engl. *Orthogonal Frequency Division Multiplexing*, OFDM) [38] kako bi podržao brzine prenosa podataka do 54 Mbit/s u opsegu od 5 GHz. Koristeći OFDM, širokopojasna modulacija je podeljena na mnoge podnosiocice od kojih svaki ima uzak opseg u odnosu na koherentni opseg tipičnog okruženja u zatvorenom prostoru (engl. *indoor*). Međutim, ovim se gubi kompatibilnost sa originalnim IEEE 802.11 standardom.

802.11b PHY standard je ekstenzija DSSS standarda koji podržava brzine prenosa podataka od 1 Mbit/s, 2 Mbit/s, 5,4 Mbit/s i 11 Mbit/s pomoću poboljšanog algoritma sekvenci podintervala poznatog kao komplementarna kodna modulacija (engl. *Complementary Code Keying*, CCK) [39] za modulaciju signala.

802.11g PHY standard omogućuje brzine prenosa podataka uporedive sa 802.11a standardom, i zadržava kompatibilnost sa DSSS i 802.11b standardima dok još rade u ISM opsegu (tj. 2,4 GHz). 802.11g fizički sloj koristi OFDM [38] modulacionu šemu koju koristi i 802.11a kako bi ostvario višu stopu prenosa podataka.

802.11n PHY [40] je najnoviji u nizu standarda koji je kreirao IEEE komitet sa ciljem da bude robustniji, sigurniji i da obezbedi veće brzine prenosa podataka u bežičnim

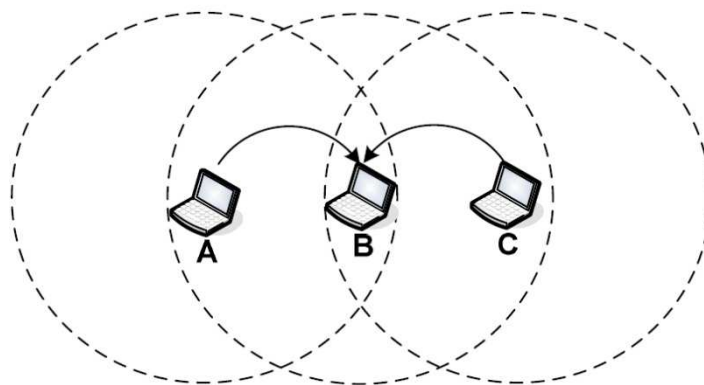
komunikacijama. Dizajniran je da dostigne vrednost propusnog opsega od 100 Mbit/s nakon što se oduzmu svi protokolarni troškovi. 802.11n standard izgrađen je na prethodnim 802.11 standardima, posebno na 802.11a, ugrađujući antene sa višestrukim-ulazima višestrukim-izlazima (engl. *Multiple-Input Multiple-Output, MIMO*). Pre 802.11n standarda 802.11 uređaji su imali jednu ili dve antene u raznovrsnim konfiguracijama, ali jedan od uslova je da samo „najbolja“ antena bude izabrana. Međutim, kod MIMO tehnologije svaki RF lanac je sposoban za istovremeni prijem ili prenos na više od jedne antene. Istovremeni prijem i obrada lanca RF signala na različitim antenama bežičnog čvora rešava problem slabljenja zbog različitih putanja, i može poboljšati kvalitet primljenih signala.

4.1.2. Sloj veze (podataka)

Sloj veze (engl. *Data Link Layer, DLL*) obavlja nekoliko važnih funkcija kao što su kontrola greške, kontrola protoka, adresiranje, uokviravanje, i sredstvo komunikacije kontrole pristupa [31]. Sloj veze se sastoji od dva podsloja (slika 4.2):

1. podsloj za upravljanje logičkom vezom (engl. *Logical Link Control, LLC*), koji je odgovoran za kontrolu greške i kontrolu protoka, i
2. podsloj za upravljanje pristupom medijumima (engl. *Medium Access Control, MAC*), koji brine o adresiranju, uokviravanju i upravljanju pristupa medijumima. Pošto čvorovi u MANET mrežama dele isti komunikacioni kanal, sudari mogu nastati ako više od jednog čvora vrši emitovanje u isto vreme. Kao posledica toga, podsloj za upravljanje pristupom medijumima ima zadatak da efikasno kontroliše pristup deljenom kanalu između čvorova u MANET mrežama.

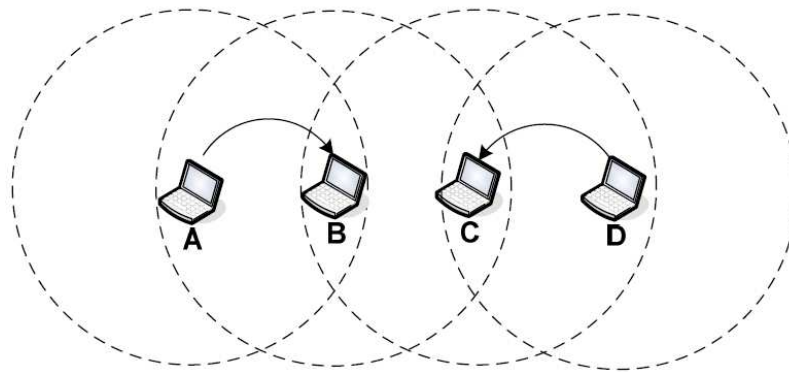
Glavni izazov za MAC podsloj je problem skrivenog terminala. U slučaju skrivenog terminala, sudar paketa se dešava na željenom prijemu ako postoji prenos sa skrivenog terminala. Kao što je prikazano na slici 4.4, kada čvor A šalje okvir za čvor B, čvor C (skriveni terminal) nije svestan prenosa zbog svoje udaljenosti od čvora A. Ako čvor C istovremeno prenosi okvir do čvora B, dešava se sudar na čvoru B.



SLIKA 4.4. PRIMER PROBLEMA SKRIVENOG TERMINALA U MANET MREŽI
(ČVOR C JE SKRIVEN OD ČVORA A)

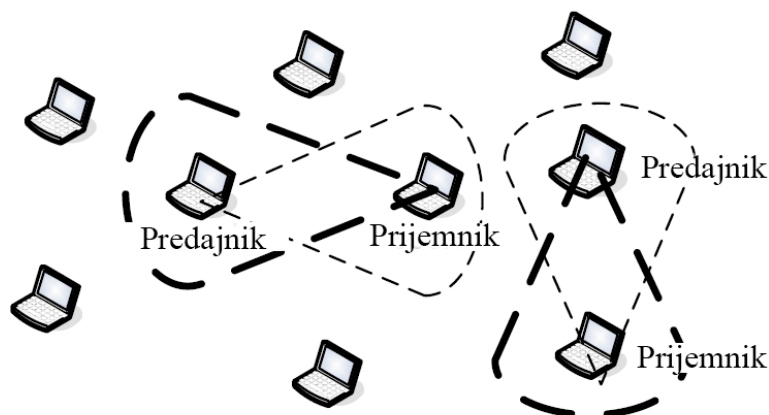
Mnogi MAC protokoli [21, 32] su predloženi kako bi se ublažio negativan efekat problema skrivenog terminala kroz izbegavanje sudara. Većina šema za izbegavanje sudara, kao što je višestruki pristup sa osluškivanjem nosioca i izbegavanjem kolizija (engl. *Carrier Sense Multiple Access with Collision Avoidance, CSMA/CA*) [32] primenjen u distribuiranoj koordinativnoj funkciji (engl. *Distributed Coordination Function, DCF*) koja je sastavna

komponenta MAC podsloja u IEEE 802.11 standardu, su inicirane od strane pošiljaoca, uključujući razmenu rezervacije kanala za kontrolne okvire između čvorova koji komuniciraju pre prenosa podataka. U tom slučaju, svi susedni čvorovi datog komunikacionog čvora treba da budu obavešteni da će kanal biti zauzet tokom određenog vremenskog perioda. Kao što je prikazano na slici 4.5, čvor A koji želi da prenese okvir podataka ka čvoru B, prvo brodkastuje zahtev za slanje (engl. *request to send*, RTS) koji sadrži dužinu paketa podataka i adresu čvora B. Po prijemu RTS okvira, čvor B odgovara emitovanjem dozvole za slanje (engl. *clear to send*, CTS) koja sadrži dužinu paketa podataka i adresu čvora A. Svaki drugi čvor koji čuje bilo koji od ova dva kontrolna okvira čuti tokom celokupnog perioda prenosa. Ovaj tihi period je poznat kao virtuelni osećaj nosioca (engl. *virtual carrier sense*).



SLIKA 4.5. PRIMER PROBLEMA IZLOŽENOG TERMINALA U MANET MREŽI
(ČVOR C JE IZLOŽEN KA ČVORU B)

Prekomerno saznanje nekog od RTS ili CTS okvira od susednih čvorova može sprečiti jedan čvor u prenošenju podataka na druge čvorove izvan komunikacionog dometa. Na primeru sa slike 4.5, komunikacija između čvorova A i B će sprečiti čvor D od pokretanja komunikacije sa čvorom C. Ovaj problem je poznat kao problem izloženog terminala (engl. *exposed terminal problem*). Ovaj problem može potencijalno dovesti do neefikasnog iskorišćenja komunikacionog kanala. Jedno od predloženih rešenja za ublažavanje problema izloženog terminala je upotreba pametnih antena ili usmerenih (direkcionih) antena [41], gde su CTS, RTS i okviri podataka usmereni ka namenjenim čvorovima (slika 4.6).



SLIKA 4.6. AD HOC MREŽA SA DIREKCIONIM ANTENAMA

4.1.3. Mrežni sloj

Mrežni sloj (engl. *Network layer*) obezbeđuje uslugu prenosa s kraja na kraj (engl. *end-to-end*). To uključuje razmenu ruting informacija, pronalaženje putanje do odredišta, popravku neispravne veze i obezbeđivanje efikasnog korišćenja raspoloživog komunikacionog propusnog opsega [31]. Jedna od najvažnijih osobina MANET mreža je mobilnost njihovih čvorova. Međutim, pokretljivost čvorova rezultuje u čestim prekidima ruta, sudarima paketa, prolaznim petljama, zastarelim ruting informacijama i teškoćama u rezervisanju resursa. Kao posledica toga, dobar ruting protokol bi trebalo da bude u stanju da reši navedene probleme uz male potrebe za resursima.

Zbog ograničenog protoka i napajanja u MANET mrežama, upotreba ruting protokola sa niskim opterećenjem komunikacije je od ključnog značaja za ukupne performanse sistema. Razmenu kontrolnih paketa za pronalaženje novih ruta i održavanje postojećih treba svesti na minimum. Kontrolni paketi troše ograničeni propusni opseg i mogu izazvati sudare sa paketima podataka, posebno kada se podešava mreža u smislu broja čvorova [43]. Dakle, efikasan ruting protokol koji može da se nosi sa velikom gustinom mreže, i da pritom koristiti mali broj kontrolnih paketa je veoma poželjan.

4.1.4. Transportni sloj

Glavni ciljevi protokola transportnog sloja (engl. *Transport layer*) obuhvataju postavljanje i održavanje konekcije s kraja na kraj (engl. *end-to-end*), pouzdanu isporuku paketa podataka, kontrolu toka i kontrolu zagušenja [31]. Dva najvažnija protokola u transportnom sloju su:

1. protokol za upravljanje prenosom (engl. *Transmission Control Protocol, TCP*), i
2. protokol za korisničke datagrame (engl. *User Datagram Protocol, UDP*).

TCP [43] pruža pouzdanu isporuku toka bajtova, što ga čini pogodnim za aplikacije kao što su prenos fajlova i elektronska pošta. Protokol je optimizovan za obezbeđivanje pouzdanosti isporuke, pre nego li za pravovremenost isporuke. Kao posledica toga, TCP ponekad može izazvati značajna kašnjenja dok čeka izgubljene pakete u određenom redosledu (obično se nazivaju segmenti) ili reemitovanju izgubljenih segmenata. Stoga, TCP nije naročito pogodan za aplikacije u realnom vremenu, kao što je govor (engl. *Voice over IP, VoIP*).

UDP [44] omogućava komunikacionim čvorovima razmenu kratkih poruka, takođe poznate kao datagrame. UDP protokol ne garantuje pouzdanost isporuke i naručivanje datagrama na način na koji to TCP radi. Datagrami mogu stići na destinaciju van uređenog redosleda, mogu biti duplirani, ili čak nestati. Izbegavanje provere da li je svaki paket stigao čini UDP bržim i efikasnijim, makar za aplikacije koje ne zahtevaju zagarantovanu isporuku, kao što su radio-difuzne, video i govor.

U početku, kada se TCP konekcija pokreće između izvora i odredišta, TCP ulazi u spori algoritam (engl. *slow-start*) [43]. U ovoj fazi, prozor zagušenja (odnosno broj segmenata prenosivih po potvrdi prijema) se povećava za svaku potvrdu prijema (engl. *acknowledgment, ACK*). Veličina prozora se povećava sa brojem potvrđenih segmenata. Ovo ponašanje efektivno udvostručuje veličinu prozora sa svakim novim krugom povratnog vremena (engl. *round trip time*). Dakle, postoji eksponencijalni rast prozora zagušenja. To se dešava sve dok neki ACK nije primljen za neke segmente ili dok se ne dosegne predodređena granična

vrednost. Kada se dostigne prag, veličina prozora se povećava za po jedan stepen za svaki krug povratnog vremena. Ova faza je poznata kao faza izbegavanja zagušenja, pri čemu je progresija veličine prozora linearna. Povećanje se nastavlja sve dok se ne dogodi gubitak. Prilikom otkrivanja gubitka, izvorni čvor detektuje zagušenje i pobuđuje algoritam kontrole zagušenja smanjenjem veličine prozora. Korišćenjem mehanizma kontrole zagušenja, TCP se pokazao efikasnim u ožičenim mrežama.

U ožičenim mrežama stopa gubitka paketa je veoma mala, a samim time pouzdanost je visoka. Kada se u ožičenoj mreži otkrije gubitak paketa to je uglavnom zbog zagušenja mreže, i kao posledica toga TCP protokol smanjuje stopu emitovanja podataka. S druge strane, u bežičnim mrežama glavni problem nije zagušenje mreže, već gubitak podataka zbog visoke stope grešaka u prenosu paketa. Kao posledica toga dešava se da TCP protokol smanjuje stopu slanja paketa kada to zapravo nije potrebno, pogoršavajući performanse MANET mreža.

U bežičnim mrežama, kao što je MANET, TCP je suočen sa degradacijom zbog svoje nesposobnosti da razlikuje gubitak paketa zbog zagušenja od gubitka usled čestih prekida ruti, prisustva zastarelih ruting informacija, visoke stope greške kanala i čestog particionisanja mreže. U [45] sprovedena je procena učinka TCP protokola usled korišćenja različitih ruting protokola u MANET mrežama. Detalji predloženih izmena TCP protokola u MANET mrežama prikazani su u [46]. Za bolje funkcionisanje TCP protokola postoje brojna poboljšanja, na primer Novi Reno (engl. *New Reno*) [47], selektivno prihvatanje opcija (engl. *Selective Acknowledgment Options*, *SACK*) [48], i eksplicitno obaveštenje o gubicima (engl. *Explicit Loss Notifications*, *ELN*) [49] koji se mogu koristiti u bežičnim mrežama za poboljšanje učinka TCP protokola.

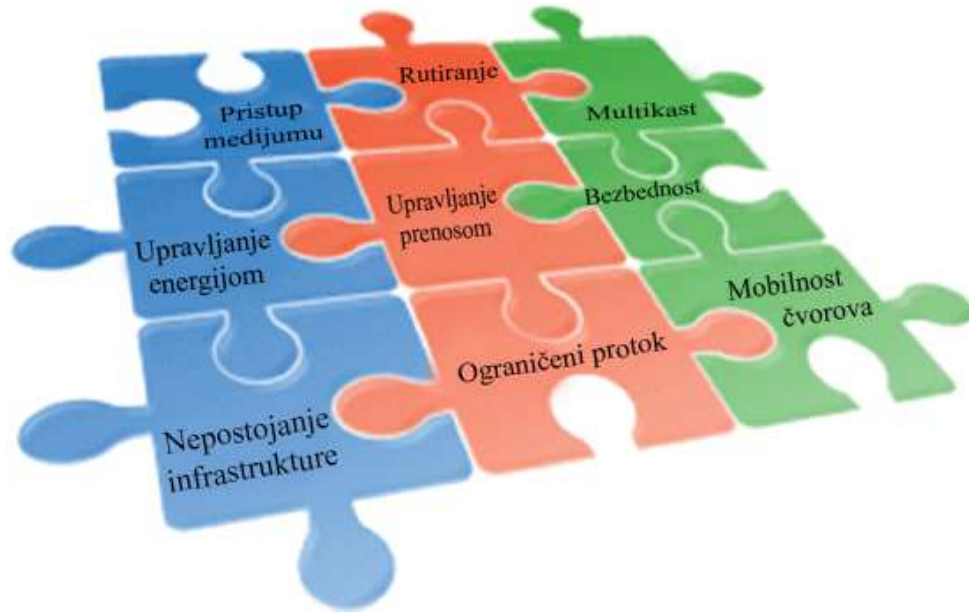
U skorijim istraživanjima [50, 51] unakrsni međuslojni dizajn se razmatra za upotrebu u MANET mrežama. Unakrsni međuslojni dizajn omogućuje protokolima koji pripadaju različitim slojevima da sarađuju u deljenju informacija o mrežnom statusu, a da pritom i dalje održavaju razdvajanje slojeva. Upotreba unakrsnog međuslojnog dizajna optimizuje upravljanje energijom. Smatra se da će ova nova arhitektura poboljšati performanse MANET mreža. Metoda unakrsnog međuslojnog dizajna ne koristi se u ovom radu, međutim može biti iskorišćena u njegovom kasnijem proširenju.

Tri gornja sloja (sesija, prezentacija i aplikacija) nisu razmotreni u ovom radu, jer se ovo istraživanje fokusira na protokolima koji rade u prva četiri sloja OSI referentnog modela. Zainteresovani čitalac može se informisati o ovim slojevima u [52].

4.2. Izazovi u ad hoc bežičnim mrežama

U poređenju sa drugim bežičnim mrežama, mobilne ad hoc mreže imaju specifične karakteristike koje stvaraju dodatne izazove. Glavni izazovi u ad hoc bežičnim mrežama su njihov dizajn i funkcionisanje proistekli usled nedostatka centralizovanog entiteta i infrastrukturnih elemenata, kao što su bazne stanice, komunikacioni tornjevi i pristupne tačke. Mobilnost čvorova i činjenica da se celokupna komunikacija obavlja putem bežičnog medijuma dodatno otežavaju njihovu izvedbu. Ove jedinstvene karakteristike unose

netrivijalne izazove u ad hoc bežične mreže [53, 54], mređu kojima su navedeni samo najvažniji (slika 4.7).



SLIKA 4.7. GLAVNI IZAZOVI KOJI UTIČU NA PERFORMANSE I DIZAJN MOBILNIH AD HOC MREŽA

Pristup medijumu

Distribuirani pristup zajedničkom kanalu prilikom prenosa paketa glavna je odgovornost podsloja za upravljanje pristupa medijumu (MAC). Glavno pitanje koje treba razmotriti u projektovanju MAC protokola za ad hoc bežične mreže je pokretljivost čvorova [55].

Rutiranje

Primarna odgovornost rutirg protokola je razmena informacija o trasama (rutama) i odabir najboljeg puta do odredišta na osnovu kriterijuma kao što su broj preskoka, minimalan utrošak energije i vek trajanja bežične veze. Jedan od ključnih problema mobilnosti je činjenica da se veze kreiraju i raskidaju na neodređen način. Usled nepostojanja fiksnih rutera i stabilnih veza, tradicionalni rutirg protokoli koji se koriste u ožičenim mrežama nisu u stanju da održe korak sa takvim čestim promenama topologije.

Multikast

Multikast ima vitalnu ulogu u primeni ad hoc bežičnih mreža u vojnim komunikacijama. Kod uređenih mreža ruteri su statični, tako da se većina multikast protokola oslanja na njih. Nakon što se multikast stablo formira, njeni čvorovi se neće pomeriti. Međutim, to nije slučaj kod ad hoc bežičnih mreža [56].

Upravljanje energijom

Upravljanje energijom se definiše kao proces upravljanja izvora i potrošača energije u čvorovima ili unutar mreže u celini, kako bi se poboljšao životni vek mreže. Većina postojećih mrežnih protokola zahteva prisustvo statičkih terminala i rutera koji imaju stalan izvor napajanja, tako da se potrošnja energije ne razmatra kao poseban problem. Čvorovi u ad hoc bežičnim mrežama istovremeno funkcionišu kao terminali i ruteri, dok za napajanje koriste baterije ili neki drugi vid ograničenih energetske izvora sa ograničenim životnim vekom. Dakle, za čvorove u takvim mrežama važan kriterijum sistemskog dizajna jeste

optimizacija u smislu efikasne potrošnje energije. Ograničenost resursa, kao što su baterijsko napajanje i količina memorije za skladištenje mogu ograničiti sposobnost ruting protokola.

Upravljanje prenosom

Glavni ciljevi protokola transportnog sloja obuhvataju postavljanje i održavanje konekcije s kraja na kraj (engl. *end-to-end*), pouzdanu isporuku paketa podataka, kontrolu protoka i zagušenja. Glavni problem koji se nameće TCP protokolu je razlikovanje između prisustva mobilnosti i zagušenja mreže. Dakle, modifikacije u okviru TCP protokola su neophodne kako bi se osigurala efikasna komunikacija.

Bezbednost

Bezbednost komunikacija u ad hoc bežičnim mrežama je od presudne važnosti, posebno u vojnim primenama. Jedinstvene karakteristike ovih mreža unose nove izazove u bezbednosnom dizajnu, kao što su otvorena mrežna arhitektura od tačke do tačke, deljeni (zajednički) bežični medijum, stroga ograničenja resursa i dinamična mrežna topologija. Nedostatak infrastrukture i sloboda kretanja čvorova čine ad hoc bežične mreže podložnijim fizičkim napadima, kao što su prisluškivanje, obmane, ometanje, napadi uskraćivanja servisa (engl. *Denial of Service, DoS*) i napad čoveka u sredini (engl. *Man-In-The-Middle, MITM*). To jasno ukazuje na potrebu za sigurnosnim rešenjima koja ostvaruju adekvatnu zaštitu i željene performanse unutar MANET mreže [57]. Međutim, njihova decentralizovana priroda mrežne kontrole čini ih robustnim protiv jedinstvene tačke otkaza usled otkaza jednog ili više čvorova.

Nepostojanje infrastrukture

Funkcionisanje ad hoc bežične mreže oslanja se na saradnju između nezavisnih i s kraja na kraj (engl. *peer-to-peer*) čvorova koji žele da komuniciraju jedni sa drugima, bez postojanja kičme mreže ili centralizovane kontrole. Dakle, svi čvorovi imaju isti opseg funkcija u okviru mreže. Ne postoje određene funkcije kao što su serveri i ruteri.

Ograničeni protok

Bežične veze koje povezuju čvorove ad hoc mreža imaju znatno manji kapacitet nego što je to slučaj u ožičenim mrežama. Efekti smetnji, buke, slabljenja i višestrukih pristupa znatno su izraženi, što izaziva varijaciju dostupnog propusnog opsega usled okolnih uslova. Stoga, realan protok često je mnogo manji od teoretskog maksimuma. Usled ovih karakteristika podrazumeva se da je zagušenje više pravilo nego izuzetak.

Mobilnost čvorova

Mobilnost povezana sa čvorovima, koja se smatra glavnom karakteristikom mobilnih ad hoc mreža, otvara mnoga pitanja kao što su sudari paketa, prekidi putanja, zastarele ruting informacije i teškoće u rezervaciji resursa. Njihovo razrešenje zahteva optimalan ruting protokol koji je u stanju da ih efikasno interpretira.

Iako ad hoc bežične mreže uživaju rast broja aplikacija i poseduju mnoge atraktivne karakteristike, istovremeno se suočavaju sa brojnim izazovima od kojih su samo najvažniji identifikovani.

4.3. Primena ad hoc bežičnih mreža

Ad hoc bežične mreže su veoma fleksibilne i pogodne za nekoliko vrsta primene, prvenstveno jer omogućavaju uspostavljanje privremene komunikacije bez unapred postavljene infrastrukture. Sledi spisak njihovih glavnih primena:

- Lične komunikacije (npr. mobilni telefoni, laptopovi, tablet uređaji),
- Zajedničke službe (npr. mreža taksi vozila, sportski stadioni),
- Hitne operacije (npr. policija, vatrogasci, spasilačke službe),
- Vojna okruženja (npr. bojišta),
- Konferencije (npr. sala za sastanke),
- Kompanijske mreže,
- Kućne mreže,
- Mreža vozila,
- Bežične senzorske mreže,
- Zdravstvo (npr. bolnice),
- Bežične meš mreže (veoma pouzdane mreže koje su usko vezane za ad hoc bežične mreže, ali u kojima čvorovi uglavnom nisu mobilni),
- Zajedničko i distribuirano računarstvo.

5. RUTING PROTOKOLI U MOBILNIM AD HOC MREŽAMA

Prilikom uspostavljanja komunikacije putem bežične mreže ako dva čvora nisu direktno povezana putem komunikacionog linka njihove međusobne poruke trebaju biti prosleđene putem središnjih (prelaznih) čvorova. Pronalaženje puta između dva čvora kojim se šalju poruke u komunikacionim mrežama zove se rutiranje. U tradicionalnim računarskim mrežama čvorovi sa namenom rutiranja paketa zovu se ruteri. Aplikacije na terminalima komuniciraju sa serverima, a paketi podataka se prosleđuju putem rutera ka svojim destinacijama. Za razliku od tradicionalnih računarskih mreža, bežične ad hoc mreže ne prave razliku između terminala, servera i rutera. Bežične ad hoc mreže se takođe razlikuju od tradicionalnih bežičnih mreža sa baznim stanicama, kao što su mobilni telefonski sistemi, u kojima se poruke između terminala prenose putem baznih stanica. U bežičnim ad hoc mrežama čvorovi su ne samo terminali, već funkcionišu i kao ruteri koji prosleđuju poruke ka drugim čvorovima koji nisu u direktnom bežičnom opsegu prenosa. Čvorovi formiraju samoorganizujuću mrežu bez centralizovane administracije i upravljanja. Dakle, bežične ad hoc mreže su distribuirani sistemi.

Ad hoc ruting protokol je konvencija, ili standard, koji kontroliše na koji način čvorovi odlučuju kako će usmeravati pakete podataka između ostalih čvorova u mobilnoj ad hoc mreži [58]. Efikasni mehanizmi za rutiranje u okviru MANET mreža su predmet brojnih istraživanja, od kojih su samo neki postali opšte prihvaćeni standard.

U ad hoc mrežama čvorovi nisu upoznati sa topologijom njihove mreže. Umesto toga, moraju je samostalno otkriti. Osnovna ideja je da svaki novi čvor objavi svoje prisustvo, i potom da osluškuje brodkast odgovore koje emituju njegovi susedi. Svaki čvor uči o čvorovima u svojoj neposrednoj blizini i o načinu kako da ih dosegne.

Ruting protokol ima dve osnovne funkcije:

- i. Pronalaženje izvodljivog puta za paketni prenos podataka od izvornog čvora ka odredišnom čvoru,
- ii. Identifikovanje i razmenu ruting informacija u vidu ruting tabele potrebne za uspostavljanje ruting putanja, otkrivanje prekida u putanjama, ponovno uspostavljanje pokidanih ili popravku puteva i optimizaciju iskorišćenosti propusnog opsega.

Čvorovi u ad hoc mreži funkcionišu kao ruteri koji otkrivaju i održavaju puteve ka drugim čvorovima u mreži. Odsustvo namenskih rutera otežava ostvarivanje bezbednosti u ad hoc bežičnim mrežama. Ovaj zadatak je još teži usled mobilnosti čvorova, ograničene procesorske snage i ograničene dostupnosti resursa, kao što su napajanje i propusni opseg.

5.1. Karakteristike ruting protokola

Ruting protokoli namenjeni za upotrebu u tradicionalnim ožičenim mrežama ne mogu se koristiti u ad hoc bežičnim mrežama. Specifične karakteristike ovih mreža zahtevaju posebne

ruting protokole koji se bave rešavanjem identifikovanih izazova u ovim mrežama [59, 60]. Takvi ruting protokoli treba da poseduju sledeće karakteristike (slika 5.1).



SLIKA 5.1. KARAKTERISTIKE RUTING PROTOKOLA U MOBILNIM AD HOC MREŽAMA

Distribuiranost

Ruting protokol treba da bude u potpunosti distribuiran, jer centralizovano rutiranje podrazumeva visok stepen kontrole radi poboljšanja pouzdanosti. Pošto su svi čvorovi mobilni, centralizovani ruting protokol je neprihvatljiv. Svaki čvor treba da vrši rutiranje koristeći podatke prikupljene od drugih čvorova.

Dinamičnost

Pored pretpostavke da se saobraćaj ravnomerno distribuira u okviru mreže, i da se održava rutiranje između svih čvorova u mreži, ruting protokol mora biti prilagodljiv konstantnim promenama topologije izazvanim mobilnošću čvorova. Topologija ad hoc bežične mreže je veoma dinamična zbog mobilnosti čvorova, tako da se oni mogu kretati unutar i van dometa jedni od drugih. Stoga, da bi povezivanje sa preskocima bilo održivo preko celokupne mreže, ruting protokol mora reagovati na promene topologije.

Nepostojanje petlji u putevima

Dobar ruting protokol treba da bude bez petlji u putevima, kako bi se izbeglo gubljenje paketa u komunikaciji i propusnog opsega bežičnih veza.

Komunikacija sa preskocima

Da bi se ostvarila komunikacija sa preskocima (engl. *multihop*) efikasan ruting protokol mora se izvršavati na svakom čvoru. Zbog decentralizovane prirode mreže svaki čvor je odgovoran za pronalaženje najboljeg puta do odredišta za pakete podataka koje dobija.

Očuvanje energije

Čvorovi u ad hoc bežičnim mrežama poseduju ograničenu količinu energije u vidu baterijskog napajanja, tako da ruting protokol treba biti visoko optimizovan sa ciljem očuvanja ovog resursa.

Bezbednost

Ruting protokoli u ad hoc mrežama podložni su brojnim napadima, kao što su lažno predstavljanje, prislušivanje mrežnog saobraćaja, presretanje i preusmeravanje poruka, i sl. Stoga, postoji potreba za uvođenjem preventivnih mera bezbednosti. Šifrovanje saobraćaja i autentifikacija mogu pomoći u rešavanju pitanja bezbednosti, i sprečiti većinu takvih napada.

Ravnomerna distribucija opterećenja

Kako broj čvorova koji postoje u određenoj geografskoj zoni varira, tako se menja i opterećenost bežičnog kanala. Dakle, ako broj čvorova raste, raste i opterećenost kanala. Dobar ruting protokol može savladati ove probleme putem ugrađenih mehanizama za ravnomernu distribuciju opterećenja preko celokupne mreže.

Kvalitet servisa

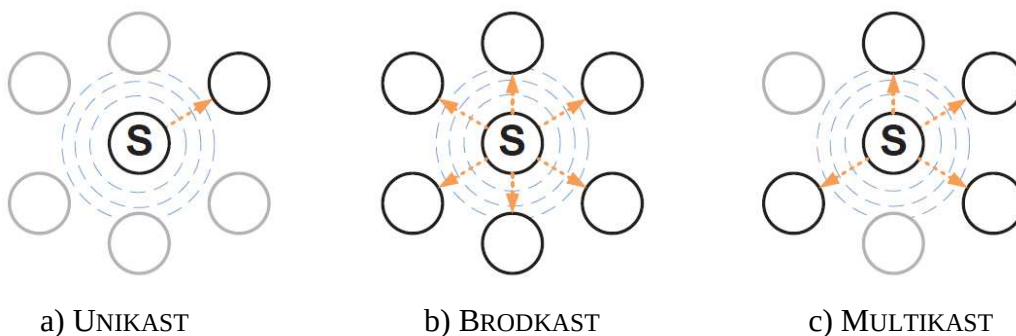
Ruting protokol treba posedovati određeni nivo kvaliteta usluga u realnom vremenu kako bi podržao trenutni saobraćaj.

Propusni opseg

S obzirom da je propusni opseg u ad hoc bežičnim mrežama ograničen, propusnost svake bežične veze se zasniva na saobraćaju koji svaka veza nosi i broju čvorova. Dakle, dobar ruting protokol treba da svede korišćeni propusni opseg na minimum.

5.2. Klasifikacija postojećih ruting protokola

Rutiranje je razmena informacija (paketa) od jednog čvora ka drugom. Glavni ciljevi ruting protokola su pronalaženje i održavanje puteva između čvorova u dinamičkoj topologiji sa eventualno unidirekcionim vezama, uz korišćenje minimalnih resursa. Protokol je skup standarda ili pravila za razmenu podataka između dva uređaja. Ruting protokoli svrstani su prema načinu prosleđivanja paketa u unicast, broadcast i multikast ruting protokole (slika 5.2).



SLIKA 5.2. KLASIFIKACIJA MANET RUTING PROTOKOLA ZASNOVANA NA NAČINU PROSLEĐIVANJA PAKETA

Unicast prosleđivanje znači komunikaciju jedan na jedan, odnosno, jedan izvor prenosi pakete podataka na jedno odredište. To je najveća klasa ruting protokola u ad hoc mrežama.

Broadcast je emitovanje od jednog ka svim ostalim čvorovima u mreži. Broadcast je osnovni režim rada na bežičnom kanalu. Naime, svaka poruka koja se prenosi na bežičnom kanalu primljena je na svim susedima koji se nalaze na rastojanju od jednog preskoka (engl. *hop*) od

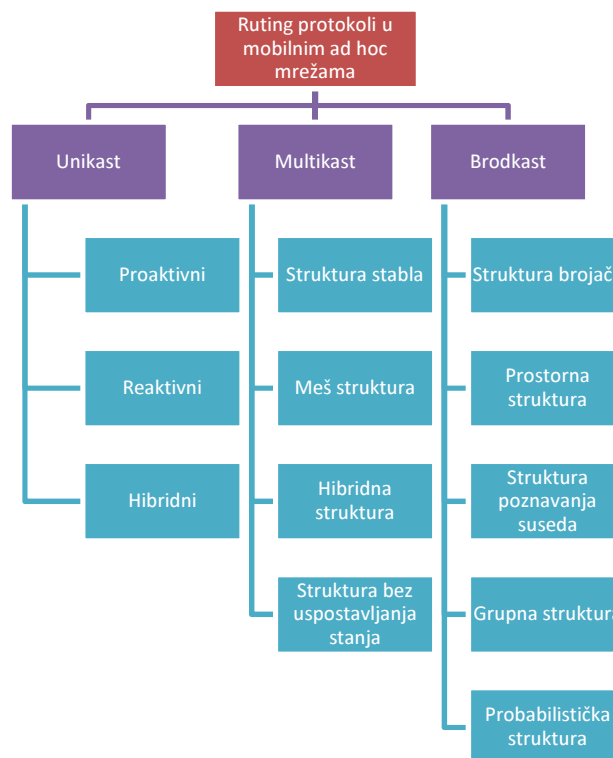
pošiljaoca. Najjednostavnije odašiljanje jedne poruke ka svih mrežnim čvorovima je „brodkast poplava“, ali to može dovesti do problema „brodkast oluje“ zbog suvišnog ponovnog reemitovanja iste poruke.

Multikast ruting protokoli koriste se kada jedan čvor treba da pošalje istu poruku, ili tok podataka, na više destinacija. Multikast je prenos podataka od jednog čvora do n prijemnika, što znači da će samo jedna kopija poruke biti dostavljena svim prijemnicima. Time će komunikacioni troškovi biti smanjeni u odnosu na slučaj kada bi jedinstvenu poruku unikast mehanizmom trebalo dostaviti višestrukim prijemnicima. Zbog svojih karakteristika, multikast je neretko pogodniji mehanizam komunikacije [61]. Ipak, najbolji rezultati ostvaruju se primenom odgovarajuće klase ruting protokola za njima odgovarajuće namene.

Ruting protokoli kreirani za upotrebu u mobilnim ad hoc bežičnim mrežama mogu se klasifikovati po mehanizmu ažuriranja ruting informacija u tri kategorije (slika 5.3):

- proaktivne (periodične),
- reaktivne (na zahtev), i
- hibridne.

Alternativa klasifikacija se zasniva na vremenskoj informaciji rutiranja koja s obzirom na životni vek bežične veze i odabranih putanja pretpostavlja njihov značaj. Pored toga, ruting protokoli se mogu klasifikovati prema tome da li je adresna topologija ravna ili hijerarhijska, ili na osnovu korišćenja određenih resursa. Ove kategorije nisu međusobno isključive, jer pojedini ruting protokoli mogu biti svrstani u više od jedne grupe. Istraživanja u ovom radu odnose se prvenstveno na klasifikaciju ruting protokola zasnovanu na mehanizmu ažuriranja ruting informacija.



SLIKA 5.3. KLASIFIKACIJA MANET RUTING PROTOKOLA

5.3. Unikast ruting protokoli

Rutiranje je najosnovnija komponenta u mrežama za ostvarivanje komunikacije i prenosa paketa. Da bi MANET mreža bila praktično upotrebljiva, funkcionalan i efikasan unikast ruting protokol je neophodan. Stoga, razvijeni su brojni različiti unikast ruting protokoli, koji se po mehanizmu ažuriranja ruting informacija mogu svrstati u proaktivne, reaktivne i hibridne.

5.3.1. Proaktivni ruting protokoli

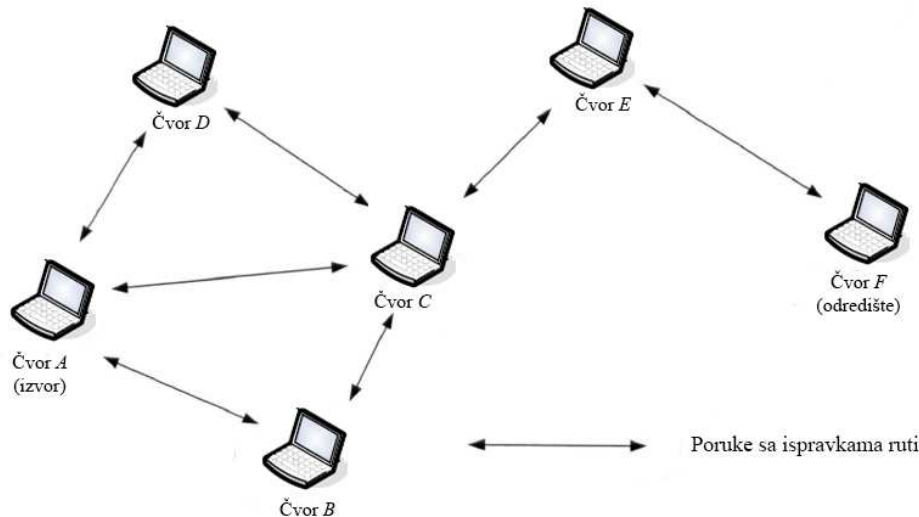
Proaktivni (periodični) ruting protokoli zasnovani su na parametrima iz tabele, jer svaki čvor održava jednu ili više tabela kako bi sačuvao mrežnu topologiju i ruting informacije. Ove informacije se često ažuriraju periodičnom razmenom ruting informacija, koje se uopšteno brodkastuju kroz celu mrežu. Proaktivni ruting protokoli međusobno se razlikuju po načinu otkrivanja i ažuriranja ruting informacija, kao i po vrstama informacija koje se čuvaju u ruting tabelama.

Proaktivni ruting protokoli održavaju aktuelnim tabele rutiranja kroz periodične kontrolne poruke. Stoga, proaktivni ruting protokoli uvek imaju kompletnu tabelu rutiranja i spremni su da razmenjuju pakete u bilo kom trenutku. Svaki čvor održava jednu ili više tabela za skladištenje ruting informacija. Čvorovi reaguju na promene u topologiji mreže propagiranjem ispravki kroz mrežu, sa ciljem održanja konzistentnosti. Oblasti u kojima se različiti ruting protokoli razlikuju su broj potrebnih tabela vezanih za rutiranje i metode kojima čvorovi šire promene u mrežnoj strukturi.

Ovaj tip protokola održava sveže liste destinacija i njihovih puteva periodičnim distribuiranjem ruting tabela kroz mrežu. Glavni nedostaci ovih algoritama su velika količina podataka za održavanje i spora reakcija na restrukturiranje i otkaze.

5.3.1.1. DSDV

Rutiranje destinacionih sekvenci zasnovano na vektoru razdaljine (engl. *Destination-Sequenced Distance Vector Routing, DSDV*) poboljšana je verzija distribuiranog Belman-Fordovog algoritma [62]. DSDV uvećava redni broj za svaku destinaciju, kako bi se osigurala brža konvergencija i kako bi se rešili problemi povratne petlje i približavanja beskonačnosti. DSDV je jedan od prvih protokola koji su predloženi kao prikladno rešenje za primenu u ad hoc bežičnim mrežama. DSDV koristi algoritam za proračun najkraće putanje na osnovu vektora razdaljine prilikom odabira putanje do odredišta. S obzirom da je DSDV ruting protokol zasnovan na parametrima iz tabele, ruting tabele se održavaju na svakom čvoru. Ruting tabele sadrže informacije o najkraćem rastojanju do svakog drugog čvora unutar mreže, kao i podatak o čvoru koji je prvi preskok na najkraćem putu ka svakom drugom čvoru u mreži. Ruting tabele se razmenjuju između suseda u redovnim intervalima, kako bi se održale aktuelne ruting informacije i topologija mreže. Pored toga, čvor prosleđuje svoju tabelu kada primeti značajnu promenu u lokalnoj topologiji u odnosu na poslednje ažuriranje. Stoga, sve ruting putanje do svih destinacija su uvek dostupne svakom čvoru, u svakom trenutku. Slika 5.4. pokazuje primer procesa ažuriranja trase i ruting tabelu čvora A.



SLIKA 5.4. DSDV PROCES AŽURIRANJA PUTANJE

TABELA 5.1. DSDV RUTING TABELA ČVORA A

Odredište	Sledeći preskok	Metrika	Redni broj
B	B	1	S180_ČvorB
C	C	1	S206_ČvorC
D	D	1	S060_ČvorD
E	C	2	S288_ČvorE
F	C	3	S120_ČvorF

Razmena ruting tabela između čvorova u mreži uvodi veliku količinu dodatnih kontrolnih poruka na mreži. Stoga, uvedene su dve metode koje se koriste za ažuriranje ruting tabela sa ciljem smanjenja količine kontrolnih poruka koje se prenose putem mreže. Prvi tip ažuriranih poruka je potpuna slika stanja (engl. *full dump*), koja se koristi kada su detektovane značajne promene u lokalnoj topologiji, ili u slučaju kada je potrebno preneti veći broj inkrementalnih paketa za ažuriranje. Dakle, ova vrsta poruka za ažuriranje sadrži sve trenutne ruting informacije. Alternativno, kada u lokalnoj topologiji nema značajnijih promena, koriste se inkrementalni paketi za ažuriranje koji sadrže samo informacije koje su se promenile od poslednje potpune slike stanja. Inkrementalne poruke za ažuriranje ruting tabele propagiraju se češće od poruka koje sadrže sliku kompletnog stanja.

Novi broj sekvence se koristi od strane odredišnog čvora kako bi se pokrenulo ažuriranje ruting tabele. Ovaj broj je uvek veći od prethodnog za određeni odredišni čvor. Na osnovu broja sekvence u ažuriranoj poruci, čvor može proslediti ili odbaciti poruku. Nakon što dobije ažuriranu tabelu, čvor ima dva izbora za ažuriranje svoje ruting tabele: ili će ažurirati svoju ruting tabelu na osnovu dobijenih informacija, ili će je zadržati neko vreme kako bi izabrao najbolje putanje iz višestrukih ažuriranih ruting tabela dobijenih od različitih susednih čvorova.

O održavanju putanja koje su raskinute brine krajnji čvor na prekinutoj putanji, koji inicira poruku ažuriranja ruting tabele. Ova poruka dodeljuje beskonačnu težinu prekinutoj putanji i novi redni broj koji je veći od sačuvanog rednog broja za tu destinaciju. Svaki čvor koji primi ažuriranu tabelu sa beskonačnom težinom neke putanje brzo je prosleđuje svojim susedima,

kako bi se informacija o prekinutoj putanji propagirala ka svim čvorovima u mreži. Na ovaj način, prekid samo jedne putanje dovodi do distribucije ažurirane ruting tabele na celoj mreži.

Očigledno, DSDV ruting algoritam ima brojne prednosti i mane. Pre svega, sve putanje ka svim destinacijama u mreži su uvek dostupne svim čvorovima. Održavanje aktuelne slike kompletne mrežne topologije na svim čvorovima propagira se kroz mrežu, što unosi veliku količinu dodatnog saobraćaja na mreži. Broj kontrolnih poruka zavisi od broja čvorova u datoj mreži i može se reprezentovati kompleksnošću reda $O(n^2)$, gde je n broj čvorova. Skalabilnost celokupne mreže je ugrožena ovom visokom složenošću. Kao rezultat toga, DSDV protokol je efikasan za formiranje malih ad hoc bežičnih mreža, ali nije skalabilan za velike mreže, jer se u tom slučaju veliki deo propusnog opsega koristi za ažuriranje ruting informacija. Velika količina vremena provodi se pri generisanju i ažuriranju ruting tabele, što je naročito primetno u vrlo dinamičnim topologijama. Još jedna mana je nestabilnost mreže tokom perioda pre nego što ažurirane ruting poruke stignu do svih čvorova u mreži, usled visoke stope mobilnosti među njima. Usled prisutnih nedostataka, DSDV ruting protokol ne koristi se u znatnoj meri.

5.3.1.2. WRP

Bežični ruting protokol (engl. *Wireless Ruting Protokol*, WRP) [63] koristi informacije prethodnika (dostupne ruting informacije) da obezbedi brzu konvergenciju i otkloni problem približavanja beskonačnosti. Kod WRP ruting protokola svaki čvor ima na raspolaganju put do svakog odredišnog čvora u mreži, u cilju održavanja aktuelne topologije cele mreže i ruting informacija među svim čvorovima. Svaki čvor održava četiri ruting tabele:

- tabelu razdaljina (engl. *Distance Table*, DT),
- tabelu ruta (engl. *Routing Table*, RT),
- tabelu težina veza (engl. *Link Cost Table*, LCT), i
- tabelu reemitovanih poruka (engl. *Message Retransmission List*, MRL).

Tabela razdaljina sadrži mrežni prikaz susednih čvorova, gde svaki element obuhvata razdaljinu i prethodni čvor prijavljen od strane susednih čvorova za određenu destinaciju. Svaka putanja sadrži najkraće rastojanje, prethodni čvor i sledeći čvor ka destinaciji, kao i zastavicu koja ukazuje na status rute. Tabela težina veza sadrži broj preskoka do odredišta, što se saznaje prenošenjem poruke kroz svaku vezu i beleženjem broja intervala ažuriranja između dva uzastopna periodična prolaza, sve dok se ne primi i poslednja uspešna ispravka sa datog linka. Svaki element tabele reemitovanih poruka se koristi za ažuriranu poruku koja treba biti reemitovana, pri čemu se zaseban brojač održava za svaki njen element. Kada se pošalje, ažurirana poruka sadrži svoj redni broj, reemisioni brojač i listu ispravki. Ako reemisioni brojač dosegne vrednost nula nakon unosa u ažuriranu poruku za koju nije primljena potvrda o prijemu, reemitovanje ažurirane poruke će biti obustavljeno.

Poruke sa ispravkama se koriste između susednih čvorova u mreži kako bi svoje ruting tabele održavali aktuelnim. One sadrže listu ispravki: odredište, rastojanje do odredišta, prethodnika odredišta i listu odgovora koja ukazuje na to koji čvorovi trebaju potvrditi svoju ispravku. Svaki čvor u mreži može menjati elemente svoje tabele razdaljina u vezi sa rastojanjem do odgovarajućih čvorova, za koje se kroz nove primljene poruke sa ispravkama može proveriti postojanje mogućih novih putanja putem drugih čvorova.

Glavna prednost WRP ruting protokola nalazi se u činjenici da su eliminisani problemi pojavljivanja ruting petlji i približavanja beskonačnosti, što se obavlja prilikom provere doslednosti prethodnih informacija generisanih od strane ostalih čvorova u mreži. Ovo eliminiše eventualne ruting petlje i istovremeno smanjuje broj potrebnih poruka ažuriranja ruting tabele kada dođe do raskida neke od veza. Međutim, kompleksnost održavanja više ruting tabela zahteva više memorije i veću procesorsku snagu od svakog čvora u ad hoc bežičnoj mreži. Takođe, postoji značajno povećanje količine kontrolnih poruka uključenih u proces ažuriranja ruting tabela usled povećanja mobilnosti čvorova. Usled toga, WRP nije pogodan za veoma dinamične mreže i mreže sa velikim brojem čvorova.

5.3.1.3. CGSR

Promenljivo rutiranje grupisanih zaglavlja i gejtveja (engl. *Cluster-head Gateway Switch Routing, CGSR*) [64] koristi hijerarhijsku mrežnu topologiju, za razliku od drugih srodnih ruting protokola koji se zasnivaju na parametrima iz tabele, a koji koriste ravnu topologiju. CGSR organizuje čvorove u klasteru i u svakom klasteru dinamički se odabire poseban čvor „glava klastera“ (engl. *Cluster Head, CH*) korišćenjem algoritma najmanje promenljivog klastera (engl. *Least Cluster Change, LLC*) [65]. LLC prati dva slučaja u kojima postoji promena glave klastera. Prvi je kada se u jednom klasteru pojave dve glave klastera. U tom slučaju jedan od čvorova mora se odreći svog statusa glave klastera. Drugi slučaj nastaje kada se jedan od čvorova slobodno pomera i izlazi van dometa svih glava klastera. Naime, cilj je da svi čvorovi unutar jednog klastera budu u komunikacionom dometu od jednog preskoka do glave klastera. Glava klastera drži sve potrebne informacije o drugim čvorovima, što uključuje putanje ka čvorovima u svom klasteru, kao i putanje ka drugim glavama klastera.

CGSR je efikasan način za alokaciju kanala između različitih klastera kroz unapređenje iskorišćenja prostora. Eksplicitan zahtev CGSR ruting protokola na nivou sloja i MAC šeme je da svaki klaster bude definisan sa jedinstvenim kodom višetrnog pristupa (engl. *Code Division Multiple Access, CDMA*), te je stoga svaki klaster u obavezi da iskoristi tehniku ponovne prostorne upotrebe kodova. Unutar svakog klastera, tehnika višestrukog pristupa podelom vremena (engl. *Time Division Multiple Access, TDMA*) se koristi sa prosleđivanje tokena.

Glava klastera izvršava CGSR ruting protokol i celokupna komunikacija prolazi kroz njega. Komunikacija između dve glave klastera odvija se kroz zajedničke čvorove (gejtveje) koji su u komunikacionom dometu obe glave klastera i članovi su oba klastera. Gejtveji komuniciraju pomoću različitih CDMA kodova na osnovu njihovih glava klastera. Glavni faktori koji utiču na rutiranje u ovim mrežama su prosleđivanje tokena (u glavama klastera) i raspoređivanje kodova (u gejtvejima). CGSR koristi šemu broja sekvence slično kao i kod DSDV ruting protokola, kako bi smanjio broj zastarelih stavki u ruting tabeli i kako bi se ostvarile ruting putanje bez petlji. Kod CGSR ruting protokola, paketi podataka se usmeravaju kroz brojne glave klastera i gejtvej čvorove na svom putu do odredišta, kao što je dato u sledećem primeru:

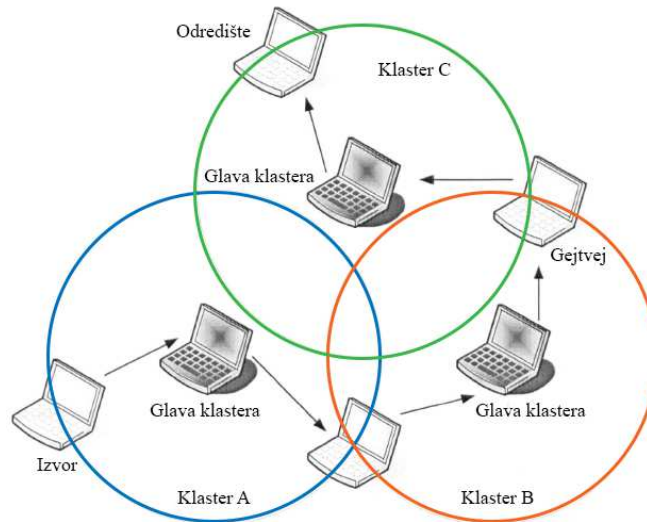
Glava klastera ↔ Gejtvej ↔ Glava klastera ↔ Gejtvej ↔ Glava klastera ↔ ...

Kod CGSR ruting protokola svaki čvor održava dve tabele:

- ruting tabelu koja sadrži odredišne glave klastera za svaki čvor u mreži, i

- ruting tabelu zasnovanu na vektoru razdaljine koja sadrži čvorove koji su prvi preskok ka doseganju svakog odredišnog klastera.

Glava klastera povremeno prenosi tabelu članova klastera što svakom čvoru omogućuje da ažurira svoju sopstvenu tabelu. Kada čvor ima paket podataka koji treba preneti, identifikuje najbližu glavu klastera ka odredištu na osnovu tabele sa članovima klastera i na osnovu ruting tabele. Na osnovu svoje ruting tabele, čvor može pronaći susedni čvor koji je prvi preskok ka dostizanju izabrane glave klastera, i potom započinje prenos paketa podataka ka tom čvoru (slika 5.5). Kako bi se održale aktuelne informacije o topologiji mreže, obe tabele moraju redovno biti ažurirane na čvorovima.



SLIKA 5.5. CGSR METOD OTKRIVANJA RUTA

Prednost CGSR ruting protokola je u smanjenoj veličini ruting tabele grupisanjem čvorova u klaster i čuvanjem samo jednog unosa za svaki odredišni klaster. Pored toga, CGSR omogućuje bolju iskorisćenost propusnog opsega i poseduje mogućnost za relativno lako sprovođenje prioritizacije prilikom raspoređivanja zahteva, usled činjenice da se CGSR zasniva na klasterovanju čvorova i usmeravanju prenosa preko glava klastera.

Glavni nedostaci ovog protokola su povećane dužine ruting putanja i nestabilnost u veoma mobilnim mrežama u kojima postoji visok stepen promene glava klastera, što degradira performanse. Pored toga, kompleksnost i suviše kontrolne poruke izazvane selekcijom glava klastera otežavaju kontinuirano održavanje strukture klastera u mobilnom okruženju. Drugi razlog za zabrinutost je potrošnja električne energije na glavama klastera, jer je stopa pražnjenja baterija na tim čvorovima znatno veća nego kod ostalih čvorova. Ovo može dovesti do čestih promena u izboru glave klastera, i stvoriti jedinstvenu tačku otkaza na glavama klastera i gejtvejima, što za posledicu može izazvati višestruke prekide veza. Sve ovo smanjuje skalabilnost mreže, što je veoma nepoželjno.

5.3.1.4. Babel

Babel je ruting protokol zasnovan na vektoru razdaljine razvijen za IP mreže sa komutacijom paketa [66]. Dizajniran je da bude robustan i efikasan za ožičene i bežične mreže. Babel je zasnovan na idejama postojećih DSDV, AODV i EIGRP ruting protokola, ali koristi tehniku

očekivanog troška prenosa (engl. *Expected Transmission Count, ETX*) za procenu težine linka, umesto jednostavne metrike koja se zasniva na broju preskoka. Pritom, koristi nekoliko tehnika kako bi se obezbedilo odsustvo ruting petlji.

Babel funkcioniše na IPv4 i IPv6 mrežama. Tokom upotrebe, pokazao se kao robustan i brzo konvergentan protokol. Dve implementacije Babel ruting protokola su javno dostupne: standardna zasebna implementacija i verzija koja je integrisana u okviru Quagga ruting paketa.

5.3.1.5. B.A.T.M.A.N.

Bolji pristup mobilnom ad hoc umrežavanju (engl. *Better Approach To Mobile Ad-hoc Networking, B.A.T.M.A.N.*) [66] je ruting protokol koji je trenutno u fazi razvoja od strane „Freifunk“ zajednice, namenjen da zameni postojeći OLSR ruting protokol.

Ključna modifikacija koju B.A.T.M.A.N. unosi je decentralizacija znanja o najboljem putu kroz mrežu, odnosno činjenica da nijedan čvor nema sve podatke. Ova tehnika eliminiše potrebu da se šire informacije o promeni mreže na svakom čvoru. Svaki čvor čuva informacije samo o pravcu sa koga je dobio podatke i šalje svoje podatke u skladu sa time. Na ovaj način, podaci se prenose od čvora do čvora, a paketi dobijaju pojedinačne, dinamički kreirane puteve. Time je formirana mreža zajedničke inteligencije.

Rane 2007. godine dizajneri B.A.T.M.A.N. ruting protokola započeli su eksperimente sa idejom rutiranja već na drugom nivou OSI referentnog modela (sloj podataka), umesto na trećem nivou (mrežni sloj). Za razliku od L3 ruting procesa sufiks *adv* (napredni) je izabran. Umesto slanja UDP paketa i manipulisanja ruting tabelom, kreira se virtuelni mrežni interfejs i paketi se transparentno transportuju. *Batman-adv* kernel modul je deo zvaničnog Linux kernela od verzije 2.6.38.

B.A.T.M.A.N. nema elemente klasičnih ruting protokola. On otkriva druge čvorove i pronalazi najbolji put do njih. Pritom, vodi evidenciju o novim čvorovima i obaveštava svoje susede o njihovom postojanju. Na redovnoj osnovi, svaki čvor šalje opštu poruku svima (engl. *broadcast*) i tako obaveštava sve svoje susede o svom postojanju. Susedi potom prenose ovu poruku svojim susedima i tako dalje. Ovim se informacija prenosi svakom čvoru u mreži. U cilju pronalaska najbolje putanje do određenog čvora, B.A.T.M.A.N. broji primljene inicijatorske poruke (engl. *originator-message*) i beleži sa kog susednog čvora poruka dolazi.

Poput ruting protokola koji se zasnivaju na vektoru razdaljine, B.A.T.M.A.N. ne pokušava da utvrdi čitav put, već pomoću inicijatorske poruke ustanovljava samo prvi korak u pravom smeru. Podaci se potom prenose prema sledećem čvoru u tom pravcu, koji dalje koristi isti mehanizam. Ovaj proces se ponavlja sve dok podaci ne stignu na odredište.

Razvoj B.A.T.M.A.N. ruting protokola podeljen je u nekoliko faza (verzija):

- i. U prvoj fazi, ruting algoritam je implementiran i testiran u smislu praktičnosti i prikladnosti za izvršavanje svojih zadataka. Za slanje i primanje inicijatorskih poruka, odnosno informacija o postojanju, izabran je UDP port 1966.
- ii. Prva verzija algoritma napravljena je na osnovu pretpostavke da čim neki čvor primi inicijatorske poruke sa drugog čvora, automatski može poslati podatke nazad.

Međutim, u bežičnim mrežama moguće je postojanje jednosmernih (unidirekcionih) veza. Stoga, u drugoj verziji algoritma poseban mehanizam je ubačen u protokol koji omogućava čvoru da utvrdi da li susedni čvor omogućava dvosmernu komunikaciju. Samo dvosmerne veze sa čvorovima se smatraju delom mreže, dok se jednosmerne veze sa čvorovima zanemaruju.

- iii. Najveća novina u trećoj verziji je podrška za raznovrsne mrežne uređaje. Sada, čvor koji izvršava B.A.T.M.A.N. ruting protokol može biti raspoređen na centralnom mestu, i može imati nekoliko ožičenih ili bežičnih mrežnih interfejsa koji su vezani za njega. Tada, B.A.T.M.A.N. može odašiljati mrežne podatke u više pravaca bez kašnjenja prilikom ponovljenog slanja paketa.

Pojedine neobične pojave i posebne okolnosti mogu se pojaviti tokom određivanja najboljeg puta kroz mrežu. Oni su otkriveni i rešeni u cilju sprečavanja ruting petlji koje mogu sprečiti podatke da stignu sa svoje odredište. Čvor sada može obavestiti mrežu da on obezbeđuje pristup Internetu. Ostali čvorovi koriste tu informaciju da procene da li postoji veza sa Internetom u njihovoj blizini i koliki propusni opseg je dostupan. Čvorovi mogu koristiti određeni gejtvej ili dozvoliti B.A.T.M.A.N. ruting protokolu da utvrdi koji gejtvej će koristiti na osnovu kriterijuma kao što je brzina veze.

Osim u bežičnim mrežama, B.A.T.M.A.N. ruting protokol može se koristiti u uobičajenim ožičenim mrežama, kao što je ethernet (engl. *Ethernet*). Tokom upotrebe, poslednja treća verzija B.A.T.M.A.N. ruting protokola pokazala je visok nivo stabilnosti, ali i nešto veće vreme konvergencije.

5.3.1.6. HSR

Ruting protokol zasnovan na hijerarhiji stanja (engl. *Hierarchical State Routing, HSR*) predložen u [67] tipičan je primer hijerarhijskog ruting protokola. HSR održava hijerarhijsku topologiju gde izabrane glave klastera (engl. *Cluster Head, CH*) na najnižem nivou postaju članovi sledećeg višeg nivoa. Na višem nivou, formiraju se superklasteri, i tako dalje. Čvorovi koji žele da komuniciraju sa čvorom van njihovog klastera pitaju njihovu glavu klastera da prosledi njihov paket na sledeći nivo, sve dok glava klastera odredišnog čvora ne bude u istom klasteru. Paket zatim putuje do odredišnog čvora.

HSR grupiše čvorove unutar jednog klastera na logičan način, umesto na geografski. Cilj je da čvorovi sa zajedničkim interesom budu grupisani zajedno, pod pretpostavkom da će intenzivno komunicirati unutar logičkog klastera. HSR ne precizira kako se klaster formira.

5.3.1.7. OLSR

Optimizovani ruting protokol zasnovan na stanju veze (engl. *Optimized Link State Routing Protocol, OLSR*) je IP ruting protokol razvijen za upotrebu u mobilnim ad hoc mrežama [29]. OLSR je proaktivni ruting protokol zasnovan na stanju veza koji koristi pozdravne poruke (engl. *Hello*) i poruke kontrole topologije (engl. *Topology Control, TC*) da otkrije, a zatim i da distribuira informacije o stanju veza širom mobilne ad hoc mreže. Pojedinačni čvor koristi ove informacije da izračuna sledeći preskok do odredišta, pritom koristeći ruting putanju sa najmanjim brojem preskoka.

U ožičenim mrežama, ruting protokoli zasnovani na stanju veze, kao što su otvoren protokol najkraće putanje (engl. *Open Shortest Path First, OSPF*) i protokol za međusistemske veze (engl. *Intermediate System-Intermediate System, IS-IS*), odabiru izabrani ruter na svakom linku za obavljanje odašiljanja informacija o topologiji. U bežičnim ad hoc mrežama postoji drugačiji pojam veze, odnosno paketi se odašilju putem jednog interfejsa. Dakle, drugačiji pristup je potreban u cilju optimizacije procesa plavljenja. Koristeći pozdravne poruke, OLSR protokol na svakom čvoru otkriva informacije o susednim čvorovima udaljenim do dva preskoka i obavlja distribuirani izbor skupa releja više čvorova (engl. *Multipoint Relay, MPR*). Čvorovi odabiru releje više čvorova, MPR-ove, tako da postoji put do svakog od njihovih suseda udaljenih do dva preskoka preko čvora izabranog kao MPR. MPR čvorovi potom odašilju poruke kontrole topologije (TC poruke) koje sadrže MPR identifikatore segmenata. Korišćenje MPR čvorova OLSR čini jedinstvenim među drugim ruting protokolima zasnovanim na stanju veze. Cilj korišćenja MPR čvorova je smanjenje reemitovanih kontrolnih poruka u istom regionu. Svaki čvor bira skup susednih čvorova sa jednim preskokom (MPR skup čvorova). Susedni čvorovi datog čvora koji nisu izabrani kao MPR čvorovi obrađuju pakete, ali ih ne prosleđuju. Samo MPR čvorovi prosleđuju pakete.

Skup odabranih MPR čvorova mora biti izabran tako da obezbedi pokrivanje svih susednih čvorova udaljenih do dva preskoka. Ovaj skup takođe mora biti minimalan, kako bi emisija brodkastovanih kontrolnih poruka bila smanjena. MPR skup čvora n treba da bude takav da svaki susedni čvor čvora n udaljen do dva preskoka ima dvosmernu vezu sa čvorovima unutar MPR skupa datog čvora n . Ove dvosmerni veze mogu se odrediti korišćenjem periodičnih pozdravnih poruka koje sadrže informacije o svim susedima i statusu njihovih veza. Dakle, ruta je niz preskoka od izvornog do odredišnog čvora, kroz MPR čvorove u okviru mreže. Izvorni čvor ne zna kompletnu putanju do odredišta, već samo informaciju o sledećem preskoku ka odredištu.

Pošto ruting protokoli zasnovani na stanju veze zahtevaju sinhronizaciju topologije preko mreže, OSPF i IS-IS ruting protokoli koriste pouzdani algoritam za obavljanje ove funkcije. S obzirom da je takav algoritam veoma teško dizajnirati za ad hoc bežične mreže, OLSR se ne bori sa problemom pouzdanosti, već jednostavno poplavljuje topološke informacije unutar mreže dovoljno često kako bi se osiguralo da baza podataka ne ostane nesinhronizovane duže vreme.

OLSR koristi pozdravne poruke da pronađe svoje susedne čvorove na razdaljini do dva preskoka putem njihovih odgovora. Pošiljaoc potom odabire svoje MPR čvorove na razdaljini od jednog preskoka, koji nude najbolje rute do čvorova udaljenih dva preskoka. Svaki čvor ima MPR identifikator segmenta, koji nabraja čvorove koji su ga izabrali kao MPR čvor. OLSR koristi poruke kontrole topologije (TC poruke) zajedno sa MPR prosleđivanjem kako bi preneo informacije o susednim čvorovima putem cele mreže. Poruka o terminalima i mrežnim udruženjima (engl. *Host and Network Association, HNA*) se koristi kod OLSR ruting protokola za širenje informacija o mrežnim putanjama, na isti način kao što TC poruke oglašavaju puteve između čvorova. Na slikama 5.6 i 5.7 prikazani su formati pozdravne poruke i poruke kontrole topologije respektivno.

0 (biti 0-9)				1 (biti 10-19)										2 (biti 20-29)							3 (30-31)	
0	1		9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4		9	0	1
Rezervisano										Interval emitovanja							Spremnost					
Kod veze				Rezervisano						Veličina poruke veze												
Adresa interfejsa susednog čvora																						
Adresa interfejsa susednog čvora																						
....																						
....																						
Kod veze				Rezervisano						Veličina poruke veze												
Adresa interfejsa susednog čvora																						
Adresa interfejsa susednog čvora																						

SLIKA 5.6. FORMAT POZDRAVNE PORUKE KOD OLSR RUTING PROTOKOLA

0 (biti 0-9)				1 (biti 10-19)										2 (biti 20-29)				3 (30-31)	
0	1		9	0	1	2	3	4	5	6	7	8	9	0	1		9	0	1
Redni broj oglašenog susednog čvora										Rezervisano									
Glavna adresa oglašenog susednog čvora																			
Glavna adresa oglašenog susednog čvora																			

SLIKA 5.7. FORMAT PORUKE KONTROLE TOPOLOGIJE KOD OLSR RUTING PROTOKOLA

S obzirom da je OLSR proaktivni ruting protokol putevi do svih destinacija u mreži poznati su pre upotrebe. Poznavanje puteva u ruting tabeli korisno je za neke sisteme i mrežne aplikacije, jer ne postoji period kašnjenja prilikom odašiljanja paketa usled procesa otkrivanja novih ruta. Dodatna prednost OLSR ruting protokola ogleda se u činjenici da se dodatne mrežne rute mogu ubrizgati u OLSR sistem putem HNA poruka koje omogućavaju povezivanje sa Internetom ili drugim mrežama u okviru OLSR MANET oblaka.

Međutim, originalna verzija OLSR ruting protokola ne sadrži nikakve odredbe za detekciju kontrole kvaliteta veza, već jednostavno pretpostavlja da je link aktivan ako su nedavno primljene pozdravne poruke. OLSR pretpostavlja da su veze aktivne ili neaktivne, što nije nužni slučaj u bežičnim mrežama, gde veze često ispoljavaju privremene prekide. Implementacija poput otvorenog koda OLSRd [68] koji se najčešće koristi na Linux baziranim mrežnim ruterima, proširena je sa detekcijom kontrole kvaliteta linka. Za male ožične pristupne tačke sa niskom procesorskom snagom, otvoreni kod OLSRd pokazao se upotrebljivim čak i u velikim meš mrežama sa hiljadama čvorova.

Činjenica da je OLSR proaktivni ruting protokol znači da se intenzivno koriste napajanje i mrežni resursi u cilju propagiranja podataka o eventualno neiskorišćenim rutama. Iako ovo ne predstavlja problem u ožičenim mrežama, ovo jeste značajan problem u MANET mrežama.

S obzirom da pripada ruting protokolima zasnovanim na stanju veza, OLSR zahteva razumno veliku količinu protoka i procesorske snage za izračunavanje optimalnih putanja u mreži. U manjim mrežama koje ne prelaze nekoliko stotina čvorova ovo ne predstavlja značajan problem, dok se sa povećavanjem broja čvorova ovi problemi sve više ispoljavaju. Dodatni utrošak ruting resursa generalno je veći nego kod reaktivnih ruting protokola, ali korišćenjem samo MPR čvorova za plavljenje topoloških informacija, OLSR postaje znatno efikasniji.

OLSR ruting protokol verzije 2 (OLSRv2) trenutno se razvija u okviru IETF radne grupe. Nova verzija protokola zadržaće ključne karakteristike originala, uključujući selekciju MPR čvorova. Ključne razlike su fleksibilnost i modularni dizajn korišćenjem zajedničkih komponenti kao što su format paketa i protokol za otkrivanje susednih čvorova (engl. *Neighborhood Discovery Protocol, NHDP*). Ove komponente razvijaju se sa ciljem da budu korišćene u narednim generacijama različitih IETF MANET ruting protokola. Razlike u upravljanju sa višestrukim adresama i interfejsima na čvorovima takođe će biti prisutne između OLSR i OLSRv2 ruting protokola.

5.3.2. Reaktivni ruting protokoli

Reaktivni ruting protokoli, takođe poznati kao ruting protokoli na zahtev, predloženi su isključivo za mobilne ad hoc bežične mreže. Osnovna karakteristika ovih mreža je njihova dinamična topologija. S obzirom da ruting protokoli prate promenu topologije, redovno ažuriranje globalne topologije je neophodno na svakom čvoru. Ponekad dobijene ruting informacije mogu isteći pre nego što se za njima ukaže potreba, što dodatno komplikuje stvari i utiče na rasipanje propusnog opsega. Koncept reaktivnih (na zahtev) ruting protokola osmišljen je kako bi se smanjila količina nepotrebnih ruting ispravki, a samim time i iskorišćenost kapaciteta linkova.

Reaktivni ruting protokoli, za razliku od proaktivnih, ne održavaju informacije o mrežnoj topologiji i putanji do svakog čvora unutar mreže. Putevi se uspostavljaju kada se za to ukaže potreba (na zahtev) korišćenjem procesa otkrivanja ruta. Generalno, kada je putanja potrebna sa izvora *S* propagira se zahtev unutar mreže kako bi se izgradila putanja do željenog odredišta *D*. Kada čvor *D* primi zahtev za rutom (engl. *Route Request, RREQ*) on šalje odgovor za rutom (engl. *Route Response, RREP*) nazad ka čvoru *S*. Kada se RREQ zahtev pošalje putem dvosmerne veze, RREP se šalje nazad kroz isti link. Kod reaktivnih ruting protokola koriste se osnovne funkcije ruting algoritama, što su otkrivanje ruta i njihovo održavanje. Jednom otkrivena ruta održava se dokle god je odredište dostupno duž svakog puta od izvora ili dokle god je ruta potrebna.

Reaktivni ruting protokoli se mogu kategorizovati prema ruting informacijama prenešenim u podacima podataka kao:

- navođenje tačne putanje (puna putanja), i
- preskok po preskok (od tačke do tačke).

Kod ruting protokola sa navođenjem tačne putanje (engl. *source routing*) kao što su DSR i ABR, svi paketi podataka nose kompletne ruting adrese od izvornog do odredišnog čvora. Dakle, paket podataka na osnovu svog zaglavlja se upućuje ka odredištu duž ranije uspostavljenog puta. Prednosti korišćenja ove vrste protokola su nepostojanje potrebe za održavanjem veza suseda među središnjim čvorovima, niti se mora održavati ruting informacija za svaku aktivnu rutu za prosleđivanje paketa ka odredištu na središnjim čvorovima.

Kod ruting pristupa „preskok po preskok“ (engl. *hop-by-hop*), kao što je AODV protokol, u paketu podataka treba navesti samo destinaciju i adresu sledećeg preskoka. Središnji čvorovi se koriste za prosleđivanje paketa ka odredištu duž puta. Korišćenje ruting tabele na svakom čvoru u mreži obezbeđuje prednost ruting pristupu „preskok po preskok“, u tome što

dinamičko ažuriranje mrežne topologije osigurava da čvorovi dobijaju najnovije informacije o topologiji i samim time mogu da prosleđuju pakete podataka duž najboljih puteva.

Reaktivni ruting pristup može se brzo prilagoditi promenama putanja i ne unosi prekomerno iskorišćenje resursa usled periodičnih kontrolnih poruka kada su putevi keširani u okviru ruting tabele ili kada se mreža nalazi u stanju mirovanja. Međutim, otkrivanje novih trasa na zahtev stvara prekomerno plavljenje mreže koje može dovesti do zagušenja, a neretko su loši putevi otkriveni. Kao rezultat, javlja se gubitak paketa prilikom prenosa i visoko vreme kašnjenja prilikom uspostavljanja novih ruta.

5.3.2.1. AODV

Rutiranje na zahtev zasnovano na vektoru razdaljine (engl. *Ad hoc On-demand Distance Vector*, *AODV*) [69] je reaktivni ruting protokol tipa „preskok po preskok“ koji uspostavlja putanju do odredišta na zahtev, odnosno trasa se uspostavlja samo kada je potrebno preneti pakete podataka sa izvornog čvora. Ovo je korisno u MANET mrežama, jer poznavanje potpune ruting tabele na svakom čvoru podrazumeva veliki utrošak kapaciteta linkova i memorije. AODV je, kako ime kaže, ruting protokol zasnovan na vektoru razdaljine. AODV izbegava problem brojanja do beskonačnosti, prisutan kod drugih ruting protokola zasnovanih na vektoru razdaljine, pomoću rednih brojeva na ispravkama ruti. AODV je sposoban za unicast i multicast rutiranje. AODV ruting mehanizam sastoji se od dva procesa: otkrivanje ruta i njihovo održavanje.

Kada izvorni čvor treba da pošalje podatke, ali nema validnu rutu do odredišta, on inicira proces otkrivanja rute kako bi pronašao putanju do odredišta. RREQ paket se jednostavno brodcastuje na celoj mreži. Svaki središnji čvor koji prosleđuje RREQ paket ka svojim susedima stvara obrnuti put nazad ka izvornom čvoru, pritom utiskivajući informaciju o sledećem preskoku u svojoj ruting tabeli. Kada RREQ paket stigne do odredišnog čvora ili do središnjeg čvora koji ima validnu rutu do odredišta, kreira se RREP paket koji se šalje nazad ka izvornom čvoru korišćenjem obrnutog puta. Usled toga, AODV je ograničen na korišćenje isključivo simetričnih veza. Validnost rute na središnjim čvorovima određuje se upoređivanjem svog rednog broja sa rednim brojem odredišta. Svaki čvor koji učestvuje u prosleđivanju RREP paketa nazad ka izvornom čvoru kreira putanju do odredišta utiskivanjem informacije o sledećem preskoku u ruting tabelu. Središnji čvorovi duž puta od izvora do odredišta ne moraju znati koji čvorovi se nalaze duž ruting putanje, izuzev informacije o narednom preskoku ka izvoru i odredištu.

Naredna faza je proces održavanja postojećih ruta. Nakon procesa otkrivanja rute i dokle god se data ruta koristi, središnji čvorovi duž aktivne rute održavaju aktivnu listu svojih suseda koji čine prve naredne preskoke putem periodične razmene pozdravnih (engl. *hello*) paketa. Takođe, kada ruta postane neaktivna, tj nema paketa podataka koji se šalju preko nje, aktivira se tajmer nakon koga se trasa smatra zastarelom i ističe. U drugom slučaju, ako čvor postane svestan da je došlo do prekida puta u okviru aktivne rute, generiše se RERR paket na mestu prekida veze. Ovaj paket se potom prenosi ka odgovarajućim čvorovima koji učestvuju u formiranju trase i do onih čvorova koji aktivno koriste datu rutu. Čvorovi pogođeni informacijom o nevažećem putu označavaju ga kao isteklog, jer im više nije koristan. Na ovaj način, RERR paket propagira se ka izvornom čvoru koji potom pokreće novu fazu otkrivanja putanje ka odredištu.

AODV je ruting protokol zasnovan na vektoru razdaljine, koja pronalazi i uspostavlja rute samo kada je sa izvornog čvora potrebno preneti pakete podataka. AODV koristi redne brojeve za identifikovanje najnovijih ruta. Kod AODV ruting protokola izvorni i središnji čvorovi čuvaju informaciju o sledećem preskoku za svaki tok saobraćaja. Kod „na zahtev“ ruting protokola, izvorni čvor brodkastuje RREQ paket unutar mreže kada put do željenog odredišta nije dostupan. Na osnovu jedne RREQ poruke izvorni čvor može dobiti više ponuđenih puteva ka odredištu. Glavna razlika između AODV i drugih „na zahtev“ ruting protokola je korišćenje rednog broja odredišta (engl. *Destination Sequence Number*, *DestSeqNum*) kako bi se odredila aktuelna ruta do odredišta. Čvor ispravlja svoju informaciju o putanji samo ako je *DestSeqNum* primljenog paketa veći ili jednak poslednjoj uskladištenoj *DestSeqNum* vrednosti na čvoru sa manjim brojem preskoka.

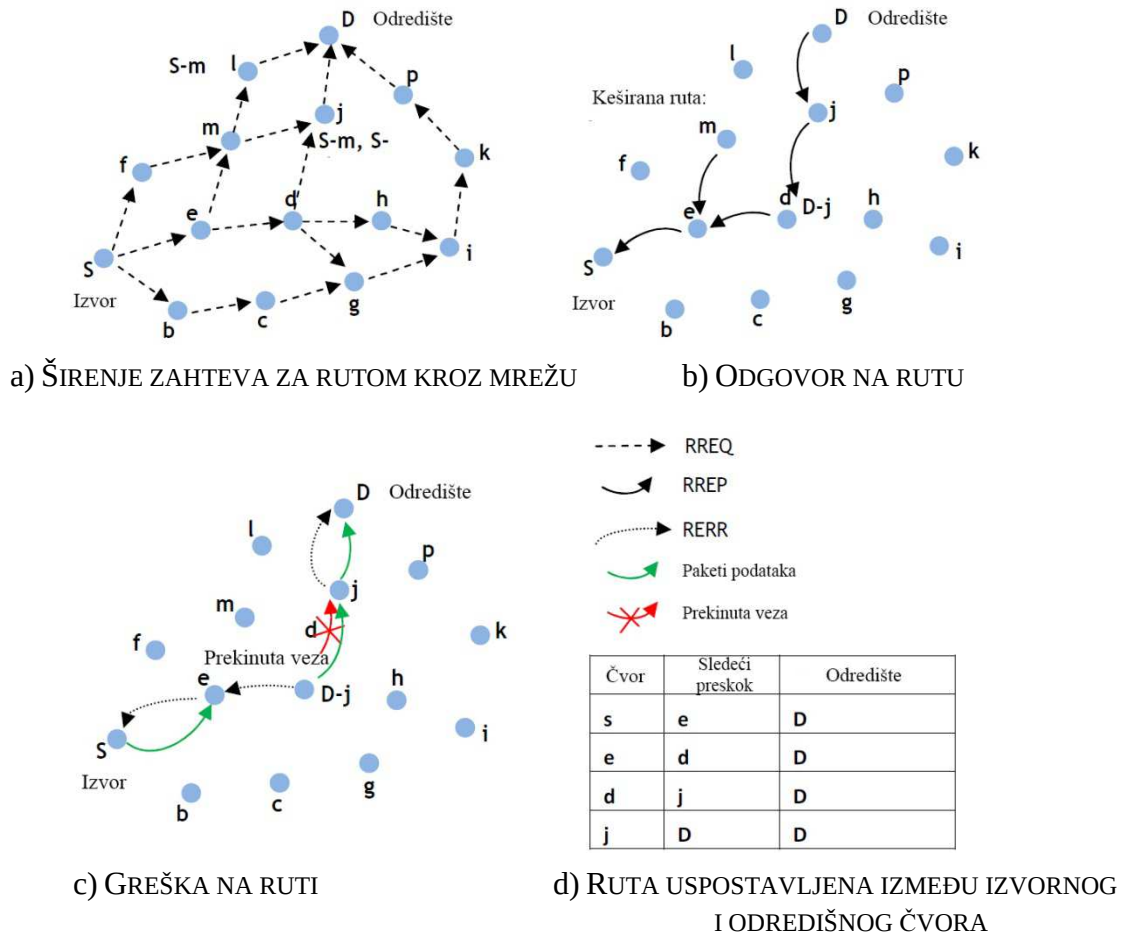
RREQ paket nosi informacije o identifikatoru izvora (engl. *Source Identifier*, *SrcID*), identifikatoru odredišta (engl. *Destination Identifier*, *DestID*), rednom broju izvora (engl. *Source Sequence Number*, *SrcSeqNum*), rednom broju odredišta (engl. *Destination Sequence Number*, *DestSeqNum*), brodkast identifikatoru (engl. *Broadcast Identifier*, *BcastID*), kao i o vremenu života (engl. *Time-to-Live*, *TTL*). *DestSeqNum* ukazuje na svežinu rute koja je prihvaćena od strane izvora. Kada središnji čvor dobije RREQ poruku, prosleđuje je ili odgovara sa RREP porukom ako ima validnu putanju do odredišta. Validnost rute na središnjem čvoru se određuje poređenjem rednog broj na središnjem čvoru sa rednim brojem odredišta u RREQ paketu. Ako je RREQ paket primljen više puta, što je označeno *BcastID*-*SrcID* parom, duplirane kopije paketa se odbacuju. Svi središnji čvorovi koji imaju validne rute do odredišta, ili direktnu komunikaciju sa odredišnim čvorom, šalju RREP pakete ka izvoru. Svaki središnji čvor prilikom prosleđivanja RREQ paketa unosi adresu prethodnog čvora i njen *BcastID*. Tajmer se koristi kako bi se obrisao unos u slučaju da se RREP poruka ne primi pre nego što tajmer istekne. Ovo pomaže u skladištenje samo aktivnih puteva na središnjim čvorovima. Kada izvorni čvor primi RREP paket, informacija o čvoru sa kojeg je primljen paket se čuva u cilju prosleđivanja paketa podataka ka ovom čvoru kao prvom preskoku prema destinaciji.

Kod AODV ruting protokola validnost rute na središnjim čvorima se određuje pomoću odgovarajućeg rednog broja destinacije, koji je veći ili jednak broju sadržanom u RREQ paketu. Ako čvor želi da šalje pakete, odgovor za rutom (engl. *Route Reply*, *RREP*) stavlja redni broj odredišta kao svoju razdaljinu u preskocima do odredišta u RREP poruku. Potom se RREP paket vraća izvornom čvoru duž trase kojom je primljena RREQ poruka. Kada izvorni čvor primi RREP paket započinje sa prenosom paketa podataka do odredišta. Ako tajmer otkrića rute istekne i ako izvorni čvor nije dobio RREP poruku, izvor reemituje RREQ poruku. Ovaj proces se ponavlja sve do nekog unapred utvrđenog maksimalnog broja pokušaja, nakon čega, ako se ne ruta se otkrije (izvor nije dobio RREP poruku) sesija se raskida.

U slučaju da dođe do prekida veze, a trasa je aktivna i neki od susednih čvorova koristi tu vezu, čvor generiše i propagira pakete greške na ruti (engl. *Route Error*, *RERR*) ka izvornom čvoru kako bi ga obavestio o prekidu veze. RERR paket sadrži IP adrese svih nedostupnih čvorova, jer se veza raskida i njihovi redni brojevi se uvećavaju za jedan. Čvor zatim brodkastuje paket i briše ove puteve iz svoje ruting tabele. Nakon prijema RERR poruke, ako izvorni čvor i dalje treba rutu za slanje paketa, reiniciraće proces pronalaženja nove putanje.

Proces otkrivanja i održavanja rute kod AODV ruting protokola predstavljen je na primeru prikazanom na slici 5.8. Izvorni čvor *S* inicira proces otkrivanja rute (slika 5.8.a)

generisanjem RREQ poruke koja se brodkastuje unutar mreže u cilju pronalaženja putanje do odredišnog čvora D . Pretpostavimo da RREQ poruka sadrži redni broj odredišta 3 i identifikator izvora S . Kada čvorovi b , e i f prime RREQ paket, proveravaju svoje ruting tabele kako bi utvrdili sledeći preskok ka odredištu. Ako u svojoj ruting tabeli nemaju putanju do odredišta, prosleđuju RREQ poruku ka svojim susedima c , d , i m . Pretpostavimo da čvorovi c i m poznaju puteve do odredišnog čvora D , kroz puteve $c-g-i-k-p-D$ i $m-l-D$ respektivno. Ako redni broj odredišta na središnjem čvoru m iznosi 4, odnosno 1 na čvoru c , onda je samo čvoru m dozvoljeno da pošalje unicast RREP paket duž trase do izvornog čvora S . To je posledica toga što središnji čvor c ima stariju rutu do čvora D u poređenju sa dostupnom rutom na čvoru S . S druge strane, čvor m ima noviju rutu do odredišta D u odnosu na trasu koja je na raspolaganju čvoru S . Ako RREQ paket na kraju stigne do odredišta preko relacije $e-d-j-D$ ili bilo koje druge alternativne trase, onda odredišni čvor D šalje unicast poruku RREP duž obrnutog puta do izvornog čvora S . U ovom slučaju izvorni čvor S može primiti više RREP paketa. Svi središnji čvorovi koji prime RREP poruku ažuriraju svoje puteve sa najnovijim rednim brojem odredišta. Oni takođe ažuriraju svoje ruting informacije ako to vodi do najkraćeg puta između izvornog i odredišnog čvora.



SLIKA 5.8. PROCES OTKRIVANJA I ODRŽAVANJA RUTE KOD AODV RUTING PROTOKOLA

Slika 5.8.c pokazuje proces održavanja rute usled prekinute veze. Kada veza pukne, što se utvrđuje usled odsustva pozdravnih paketa ili potvrde prijema istih, izvorni i odredišni čvorovi su obavešteni. Na primer, kada se veza između čvorova d i j raskine, oba čvora kreiraju RERR pakete koji obaveštavaju izvor, središnje čvorove duž puta i odredišni čvor o raskidanju veze. Čvorovi brišu odgovarajuće stavke iz njihovih ruting tabela. Izvorni čvor S

reinicijalizuje proces otkrivanja trase novim RREQ paketom koji sadrži novi brodkast identifikator i prethodni redni broj odredišta D .

Glavna prednost AODV ruting protokola je korišćenje „na zahtev“ pristupa prilikom uspostavljanja ruta i korišćenje rednog broja odredišta u cilju održanja i pronalaženja najnovijih putanja. Takođe, ruting protokol zasnovan na vektoru razdaljine je jednostavan i ne zahteva mnogo memorije za očuvanje ruting tabele.

Međutim, AODV zahteva više vremena za uspostavljanje rute, a početna komunikacija prilikom uspostavljanja rute je teža nego kod nekih drugih pristupa. Nedostatak ovog protokola je da središnji čvorovi mogu dovesti do nedoslednosti u ruting putanjama, ako je redni broj izvora veoma zastareo i ukoliko središnji čvorovi imaju veći, ali ne i poslednji redni broj odredišta, čime se stvraju zastareli unos. Takođe, brojni RREP paketi koji nastaju kao odgovor na svega jedan RREQ paket mogu dovesti do prekomernog generisanja kontrolnih poruka. Još jedna mana AODV ruting protokola je nepotrebno iskorišćenje protoka linkova usled periodičnih kontrolnih poruka.

Ako je RREQ paket poslat sa izvornog čvora ka svim susedima tokom otkrivanja rute, svaki sused dalje prosleđuje ovaj paket ka svojim susedima, bez uzimanja u obzir da li se ti čvorovi kreću van dometa pokrivanja. Vremenom, ovo dovodi do neuspeha prilikom slanja podataka duž otkrivene trase, jer lako može doći do raskida veze ka susednom čvoru koji se kreće van dometa pokrivanja prethodnog čvora duž ruting putanje.

5.3.2.2. HARP

Ruting protokol ugaone usmerenosti glave (engl. *Heading-direction Angle Routing Protocol, HARP*) [70] tako je nazvan zbog toga što koristi informacije o usmerenosti čvorova u mreži. Takve informacije mogu se dobiti od čvorova koji poseduju odgovarajuće instrumente i senzore, kao što je kompas koji pruža informacije o usmerenosti glave čvora u odnosu na magnetni sever. Ovaj protokol koristi se za smanjenje količine kontrolnih ruting poruka i za povećanje životnog veka veza između čvorova, uz pretpostavku da svaki čvor često razmenjuje informacije sa svojim susedima. Kod HARP ruting protokola svaki čvor klasifikuje svoje susede u osam različitih zona u skladu sa usmerenjem njihovih glava. Čvorovi se razvrstavaju u najmanje jednu od osam zona, bez obzira na njihovu lokaciju. Ovaj protokol pripada „na zahtev“ ruting tehnici. RREQ paket se prenosi sa čvora na jedan od njemu susednih čvorova koji ima sličnu ugaonu usmerenost glave kao i ostali HDA susedni čvorovi.

Kada izvorni čvor S šalje zahtev za rutom do odredišnog čvora D , pogledaće prvo u svojoj memoriji za rutom ka čvoru D . Ako je pronađe, čvor S će započeti sa slanjem paketa podataka ka čvoru D . Ako ruta ka čvoru D nije pronađena, vremenski period T_d će biti iniciran od strane izvornog čvora S , gde je T_d vreme potrebno da se pronađe odredište. Zatim, čvor S počinje da traži u svojoj memoriji susedni čvor koji ima istovetan ili sličan referentni ugao glave.

Ovaj protokol smanjuje količinu generisane saobraćaja usled uspostavljanja ruta i smanjuje iskorišćenost propusnog opsega, jer se ne zahteva od svih susednih čvorova da odgovore na RREQ poruku. Njegova glavna prednost je u tome što povećava vek trajanja veze između čvorova. Nedostatak se ogleda u činjenici da će izvorni čvor po prijemu poruke o grešci

ponovo generisati RREQ paket. Ograničeno slanje RREQ paketa eliminiše problem formiranja ruting petlji bez uzimanja u obzir da li se zna tačan put. Drugi nedostatak je klasifikacija različitih zona, koje nisu pogodne za primenu u mrežama visoke ili niske gustine. Ovaj protokol koristi se između ostalog kao tehnika mapiranja ose.

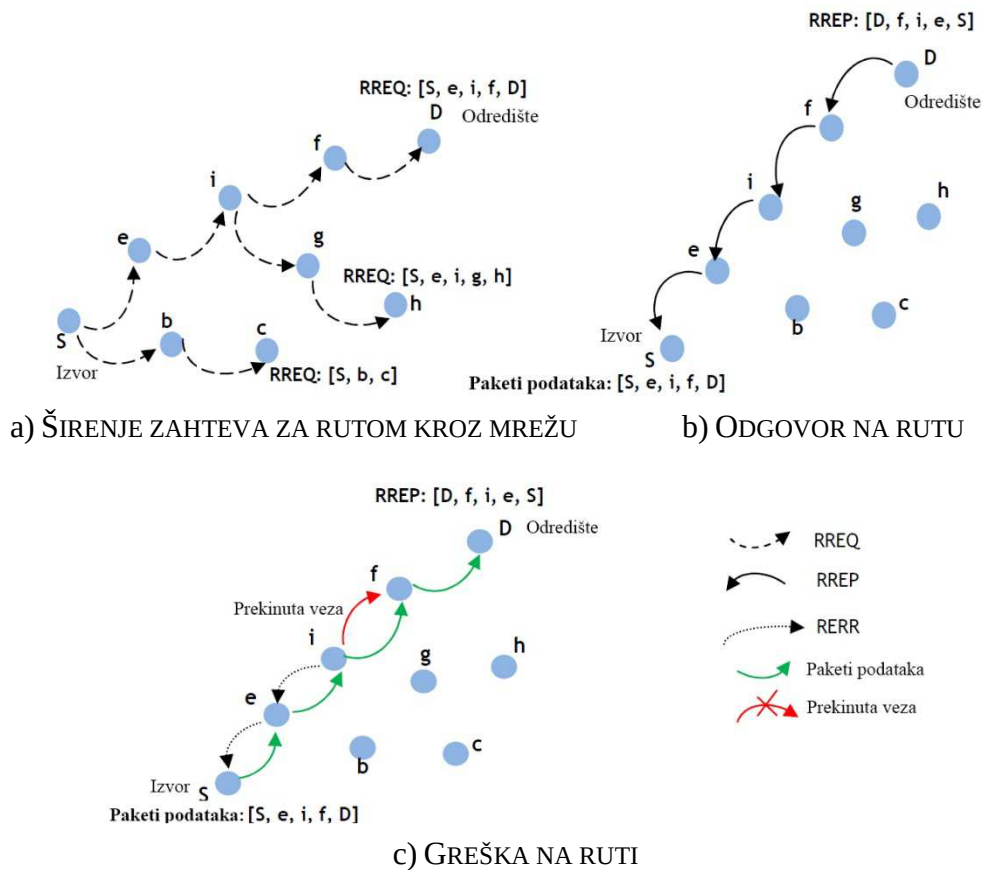
5.3.2.3. DSR

Dinamičko navođenje tačne putanje (engl. *Dynamic Source Routing, DSR*) [71] je ruting protokol razvijen na Carnegie Mellon Univerzitetu, SAD. DSR je dizajniran specifično za upotrebu u ad hoc bežičnim mrežama sa višestrukim preskocima. DSR odlikuje ruting tehnika navođenja tačne putanje umesto tehnike sledećeg preskoka kao što je to slučaj kod AODV ruting protokola, gde svaki paket koji treba biti prenešen nosi u svom zaglavlju kompletnu listu čvorova kroz koje mora proći do odredišta. Ključna prednost ruting tehnike navođenja tačne putanje ogleda se u tome što središnji čvorovi ne moraju održavati aktuelne ruting informacije da bi usmerili pakete podataka prema destinaciji, jer sami paketi u svom zaglavlju već sadrže sve ruting odluke. Ova činjenica, zajedno sa „na zahtev“ prirodom protokola, eliminišu potrebu za periodičnim porukama koje sadrže ispravke ruta i paketima za otkrivanje susednih čvorova prisutnih u drugim ruting protokolima.

Glavna razlika između DSR i AODV ruting protokola je u tome što DSR poruke sadrže punu putanju između izvornog i odredišnog čvora. Pored toga, kod DSR ruting protokola RREQ paket uključuje adrese svih čvorova koji su posećeni pre nego što se brodkastuje preko mreže. Svaki čvor koji primi RREQ paket prosleđuje ga svojim susedima samo ukoliko im paket već nije prosleđen ili ako susedni čvor nije odredišni čvor. Po prijemu RREQ paketa, odredišni čvor šalje RREP paket ka izvoru. RREP se prenosi istovetnim putem kojim je prešao i RREQ paket, ali u suprotnom smeru.

Ruting mehanizam DSR protokola sastoji se od dve faze: otkrivanje ruta i njihovo održavanje.

Proces otkrivanja ruta kod DSR ruting protokola inicira se kada čvor pokuša da pošalje paket podataka do odredišnog čvora za koji ne zna trasu. Proces otkrivanja nove rute zasniva se na širenju RREQ paketa kroz mrežu (slika 5.9.a) korišćenjem jednostavne poplave, odnosno svaki čvor koji primi RREQ paket prosleđuje ga ka susednim čvorovima, osim ako je to odredišni čvor ili središnji čvor koji već poznaje putanju do odredišta u svojoj ruting tabeli. Takvi čvorovi odgovaraju na zahtev sa RREP paketom koji se usmerava ka izvornom čvoru. Propagirani RREQ paketi izgrađuju ruting putanju. RREP je unikast paket koji se šalje nazad ka izvornom čvoru duž istog puta (slika 5.9.b). Trasa putem koje je prenet RREP paket se kešira na izvornom čvoru za buduću upotrebu. Nakon procesa otkrivanja rute, svaki paket podataka poslat sa izvora ka odredištu u svom zaglavlju sadrži informacije o kompletnoj putanji.



SLIKA 5.9. PROCES OTKRIVANJA RUTE KOD DSR RUTING PROTOKOLA

Proces održavanja ruta je odgovoran za detekciju promena unutar mrežne topologije koje utiču na korišćene rute. Kad god dođe do prekida veze (npr. detektovanjem neuspelog pokušaja prenosa podataka preko linka) RERR paket se šalje ka izvornom čvoru sa čvora gde je došlo do prekida veze (slika 5.9.c). Tokom prenošenja RERR paketa ka izvornom čvoru brišu se sve stavke u keširanim ruting tabelama duž puta koji sadrži neispravnu vezu. Po prijemu RERR paketa, izvorni čvor reinicijalizuje proces otkrivanja rute ukoliko je ovaj put i dalje neophodan i ukoliko ne postoji alternativni put koji se već nalazi u keš memoriji.

Postoje tri prednosti korišćenja tehnike rutiranja navođenjem tačne putanje i rednih brojeva:

- Sprečava se formiranje ruting petlji.
- Uklanja se potreba za znanjem aktuelnih ruting informacija na središnjim čvorovima prilikom prosleđivanja paketa.
- Izbegava se ponovno slanje istog ruting paketa i višestrukih prenosa na središnjim čvorovima.

Ove tehnike su implementirane u DSR ruting protokol u cilju poboljšanja njegovog učinka u korišćenju ruting informacija na središnjim čvorovima, prilikom generisanja njihovog odgovora ka izvornom čvoru nakon što dobiju RREQ pakete. U ovom slučaju, središnji čvorovi mogu imati put do sličnog odredišta. Druga optimizacija je uklanjanje kašnjenja prilikom vraćanja odgovora na RREQ zahteve.

Pored ovih, još jedna velika prednost DSR ruting protokola je eliminisanje potrebe da se mreža opterećuje sa povremenim ažuriranim ruting porukama, tako što se rute uspostavljaju isključivo na zahtev. S druge strane, raskinute veze ne mogu biti lokalno popravljene putem

mehanizma periodičnog održavanja. Umesto toga, raskinuta veza izaziva slanje RERR paketa ka izvornom čvoru, koji potom pokreće novu fazu otkrivanja ruting putanje. Čvorovi koji prime RERR poruku takođe prihvataju informaciju o zastareloj ruting putanji koja koristi raskinutu vezu. Druga mana DSR ruting protokola je činjenica da ne radi dobro u velikim mrežama, jer svaki paket u svom zaglavlju prenosi informaciju o celokupnoj putanji do odredišnog čvora. Performanse DSR ruting protokola značajno se degradiraju sa povećanjem mobilnosti čvorova, ali je zato dobar za upotrebu u statičkim mrežama, kao i u sredinama sa niskim stepenom mobilnosti. Uprkos korišćenja tehnike pronalaženja ruta na zahtev, postoji značajan broj emitovanih kontrolnih paketa zahvaljujući ruting mehanizmu navođenja tačne putanje putem plavljenja RREQ paketa. Mehanizam uspostavljanja ruta kod DSR ruting protokola može izazvati dugo odlaganje u slanju paketa, kao rezultat potrebe da se paket prenese kroz novu vezu. Kod ruting tehnike navođenja tačne putanje korišćenje jednog od pozitivnih aspekata mobilnosti na kraju utiče na smanjenje efikasnosti mobilnosti. Proizvod je novi adaptivni mehanizam koja zahteva veze sa dužim vekom trajanja, kako bi se smanjili troškovi izazvani propagiranjem RREQ paketa u mreži.

5.3.2.4. DSRFLOW

Stanje toka kod dinamičkog navođenja tačne putanje (engl. *Flow State in the Dynamic Source Routing, DSRFLOW*) je skup dodataka koji pruža sve pogodnosti ruting tehnike navođenja tačne putanje, uz smanjenje utroška resursa koje je povezano sa ovom tehnikom. DSRFLOW radi tako što dozvoljava slanje većine paketa bez zaglavlja, koje kod DSR ruting protokola sadrži informacije o kompletnoj putanji, čime se znatno smanjuje prekomerna iskorišćenost resursa. Jedan od glavnih nedostataka DSR ruting protokola ogleda se u činjenici da što je duža trasa kojom se prenose paketi, veće je i zaglavlje samih paketa. Tehnika koja se koristi kod DSRFLOW ruting protokola zove se implicitno navođenje tačne putanje. Unapređena tehnika stanja toka kod DSR ruting protokola prvi put je opisana u [72]. Mehanizam implicitnog navođenja tačne putanje sada je uključen u DSR Internet nacrtu.

Osnovna ideja DSRFLOW ruting protokola je da uvede takozvanu tabelu protoka za svaki čvor, čime DSRFLOW postaje ruting protokol koji čuva informacije o sesijama (engl. *stateful routing*). Za svaki tok koji čvor prosleđuje postoji jedna stavka u tabeli protoka, koja minimalno mora da sadrži adresu sledećeg preskoka.

5.3.2.5. ABR

Ruting protokol zasnovan na povezivanju (engl. *Associatively-Based Routing Protocol, ABR*) [73, 74] je signalni „na zahtev“ ruting protokol razvijen na Univerzitetu u Kembridžu, UK. ABR je distribuirani ruting protokol namenjen za upotrebu u mobilnim ad hoc mrežama, sa ciljem kreiranja stabilnih bežičnih veza. Naime, veze mogu biti stabilne ili nestabilne u zavisnosti od njihove vremenske stabilnosti. Stabilnost veze se određuje prebrojavanjem periodičnih signala koje čvor prima od svojih suseda. Kod ABR ruting protokola svaki čvor proizvodi periodične pozdravne poruke kako bi predstavili svoje prisustvo susednim čvorovima. Čvorovi računaju broj primljenih pozdravnih poruka od svojih suseda i taj podatak se koristi za ažuriranje tabele asocijativnosti svakog čvora. Pošto čvorovi koriste vremensku stabilnost i tabelu asocijativnosti, u stanju su da kategorizuju svaku vezu sa susedom kao stabilnu ili nestabilnu, u zavisnosti od broja signalnih poruka koje odgovaraju datom susednom čvoru. Veza koja odgovara nestabilnom susednom čvoru naziva se

nestabilna veza, odnosno veza ka stabilnom susedu naziva se stabilna veza. ABR je ruting protokol sa navođenjem tačne putanje koji nema potrebu za povremenim slanjem ruting ispravki.

Ukoliko put do odredišta nije dostupan u keš memoriji izvornog čvora, inicira se proces otkrivanja nove rute propagiranjem RREQ paketa kroz mrežu. Središnji čvorovi po prijemu RREQ paketa dodaju im svoju adresu i svoju vrednost broja primljenih signala, i kao takve ih prosleđuju svojim susedima. Od momenta kada primi RREQ paket, odredišni čvor čeka tokom određenog perioda pre nego što izabere rutu koja ima maksimalan procenat stabilnih veza. Ukoliko je primljeno više pravaca koji imaju isti ukupni stepen stabilnosti, biće izabran najkraći put.

Ako do prekida veze dođe na nekom od središnjih čvorova, to se detektuje putem signalnih poruka. Potom čvor najbliži mestu gde se detektuje prekid inicira proces lokalne popravke putanje, lokalnim brodkastovanjem poruke za popravku rute sa ograničenim vremenom života. Ako ovaj čvor ne uspe da popravi prekinutu vezu, sledeći čvor na putu ka izvoru reinicijalizuje lokalnu poruku za popravku veze. Ovaj proces popravka veze nastavlja se duž svih središnjih čvorova prema izvornom čvoru. Ukoliko se dogodi da središnji čvorovi na prekinutoj putanji ne uspeju da poprave raskinutu vezu, obaveštava se izvorni čvor koji potom započinje proces otkrivanja nove putanje prema odredištu.

Glavna prednost ABR ruting protokola nalazi se u činjenici da preferira stabilne puteve u odnosu na puteve sa manjim brojem preskoka, što rezultuje u manjem broju prekinutih putanja i zauzvrat smanjuje opterećenost mreže usled učestalog pronalaženja novih puteva. S druge strane, izabrana stabilna putanja može biti znatno duža od putanje sa najmanjim brojem preskoka između izvornog i odredišnog čvora. Takođe, lokalni brodkast upit za lokalnu popravku veze može dovesti do dugih zastoja u prenosu paketa podataka.

5.3.2.6. SSA

Adaptivni ruting protokol zasnovan na stabilnosti signala (engl. *Signal Stability-Based Adaptive Routing Protocol, SSA*) odabire rute na osnovu snage radio signala između dva čvora i njihove stabilnosti, čime se identifikuju putevi koji postoje duže od ostalih [75]. Poput ABR ruting protokola, SSA koristi signalne poruke između čvorova, pri čemu izmerena jačina signala signalne poruke određuje stabilnost veze. Na osnovu izmerene jačine signala, veza se klasifikuje kao stabilna ili nestabilna. Ova tehnika zahteva korišćenje proširenog radio interfejsa na čvorovima kako bi bili u stanju da izmere jačinu primljenih signala. SSA ruting protokol sastoji se od dva međusobno povezana protokola:

- Protokola za prosleđivanje (engl. *Forwarding Protocol, FP*), i
- Dinamičkog ruting protokola (engl. *Dynamic Routing Protocol, DRP*).

Protokol za prosleđivanje obavlja rutiranje u cilju prosleđivanja paketa na putu do odredišta, dok dinamički ruting protokol održava ruting tabelu i tabelu stabilnosti signala (engl. *Signal Stability Table, SST*) kroz interakciju sa svojim procesima na drugim susednim čvorovima. Tabela stabilnosti signala sadrži broj primljenih signala i zasnovana je na snazi signala signalnih poruka svojih suseda. Ova tabela se koristi prilikom prosleđivanja zahteva za rutom (RREQ) sa čvorova na putanji do odredišta putem veza sa jakim signalom, kako bi se pronašle najstabilnije putanje od izvornog do odredišnog čvora. Ako čvor primi snažne signale u nekoliko poslednjih signalnih poruka, onda tu vezu smatra kao jaku i stabilnu.

Ako odredište nije dostupno u ruting tabeli, FP inicira zahtev za otkrivanje nove rute. Ključni zadatak u otkrivanju rute je da se RREQ paketi prosleđuju sledećem čvoru samo ako su primljeni preko jake veze i ako nisu prethodno obrađeni. Ukoliko se paket primi putem slabe veze, biće odbačen pre nego što bude obrađen. Ukoliko izvorni čvor ne pronađe stabilne veze neophodne za izgradnju putanje do odredišta, onda FP protokol propagira zahtev za otkrivanje rute putem mreže bez razmatranja stabilnosti veza. U tom slučaju, odredišni čvor bira vezu putem koje je pristigao prvi RREQ paket i aktivira slanje paketa odgovora na rutu (RREP) prema izvornom čvoru. DRP preskače duž veza između središnjih čvorova, i pritom ažurira njihove ruting tabele. Na taj način, RREQ paketi dolaze do odredišnog čvora preko veza sa najjačom stabilnošću signala.

Glavna prednost SSA ruting protokola jeste da se paketi između čvorova generalno prosleđuju duž stabilnih puteva. Međutim, ukoliko stabilne veze nisu dostupne za prosleđivanje RREQ paketa između dva čvora, onda izvorni čvor propagira RREQ pakete preko mreže bez obzira na stabilnost veza, iako je moguće da će neka od njih nakon kratkog perioda biti na raspolaganju i jaka. Još jedna mana je da kada čvor detektuje prekinutu vezu, odašilje paket sa greškom na ruti (RERR) ka izvornom čvoru, obaveštavajući ga o prekidu putanje. Kada RERR paket stigne na izvorni čvor, on šalje paket za brisanje rute kako bi obavestio sve čvorove na prekinutom putu, i pokreće proces pronalaženja nove putanje do odredišta. Ovo izaziva plavljenje mreže RREQ paketima, što dovodi do smanjenja raspoloživog propusnog opsega.

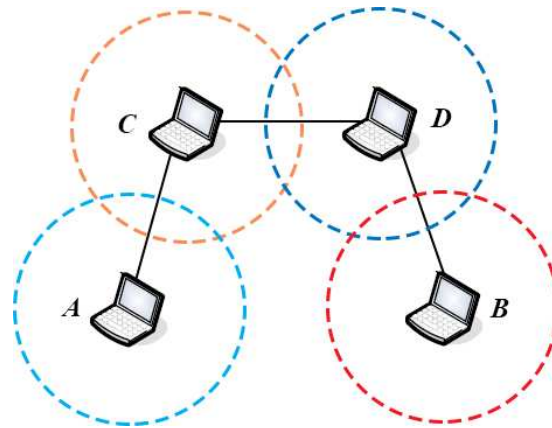
5.3.2.7. DYMO

Dinamički „na zahtev“ MANET ruting protokol (engl. *DYnamic Manet On-demand Routing, DYMO*) je naslednik popularnog AODV ruting protokola i sadrži mnoge od njegovih prednosti [76]. DYMO je donekle lakši za implementaciju i dizajniran je sa budućim poboljšanjima u vidu.

DYMO može raditi i kao proaktivni i kao reaktivnog ruting protokol, odnosno putevi mogu biti otkriveni samo kada je to potrebno. U oba slučaja novi putevi se otkrivaju putem sledećih koraka:

1. Posebna poruka sa zahtevom za rutu (RREQ) se brodkastuje kroz MANET mrežu. Svaka RREQ poruka čuva spisak svih čvorova kroz koje je prošla, tako da svaki čvor koji primi RREQ poruku može odmah saznati put nazad do izvora ove poruke.
2. Kada RREQ poruka stigne na svoje odredište, poruka sa odgovorom na rutu (RREP) se prosleđuje nazad ka izvornom čvoru, ukazujući na to da je put do odredišta pronađen. Na svom putu nazad ka izvoru, RREP poruka vraća se istim putem kojim je stigla RREQ poruka, što omogućuje svim središnjim čvorovima kroz koje prolazi da snime ovu rutu.

Dakle, čim RREP poruka stigne do izvornog čvora, dvosmerni put je uspešno ostvaren i snimljen na svim središnjim čvorovima kroz koje prolazi, i razmena paketa podataka može početi.



SLIKA 5.10. PRIMER ZA PREDSTAVLJANJE DYMO RUTING PROTOKOLA

U sledećim koracima prikazano je ostvarivanje komunikacije između čvora A i čvora B kod DYMO rutinog protokola na primeru sa slike 5.10:

1. Čvor A želi da pošalje pakete podataka ka čvoru B. Međutim, čvor A ne zna putanju do čvora B, tako da brodkastuje RREQ poruku za pronalaženje puta ka čvoru B, koja sadrži samo informacije o čvoru A.
2. Čvor C prima RREQ poruku sa čvora A, pamti informaciju o tome kako da dosegne čvor A (direktno), a zatim u RREQ paket dodaje informacije o sebi i ponovo emituje paket.
3. Čvor D prima RREQ paket sa čvora C, pamti informaciju o tome kako da dosegne čvor C (direktno) i čvor A (preko čvora C), a zatim u RREQ paket dodaje informacije o sebi i ponovo emituje paket.
U isto vreme, čvor A takođe dobija RREQ paket od čvora C. Ispitivanjem sadržanih informacija unutar paketa otkriva blok sa informacijama kako da dosegne do samog sebe, što mu nije od koristi. Stoga, čvor A odbacuje RREQ paket i ne emituje ga ponovo kao što to čini čvor D.
4. Čvor B prima RREQ paket sa čvora D, i pamti informacije o tome kako da dosegne čvor D (direktno), čvor C (preko čvora D) i čvor A (takođe preko čvora D). Shvativši da je odredište RREQ poruke, čvor B kreira RREP paket koji sadrži informacije o samom sebi. Potom označava RREP paket sa odredištem A, i znajući da čvor D ima putanju ka čvoru A, prosleđuje ga ka čvoru D.
U isto vreme, čvor C takođe dobija RREQ paket od čvora D, ali po istoj logici kao i čvor A nešto ranije, ignoriše i odbacuje RREQ paket.
5. Čvor D prima RREP paket koji čvor B šalje čvoru A, pamti informacije o tome kako da dosegne čvor B (direktno) i dodaje informacije o sebi unutar paketa. Znajući da čvor A može da se dosegne preko čvora C, šalje RREP paket ka čvoru C.
6. Čvor C prima RREP paket namenjen čvoru A od čvora D, pamti informacije o tome kako da dosegne čvor D (direktno) i čvor B (preko čvora D), a zatim dodaje informacije o sebi unutar RREP paketa. Znajući da čvor A može direktno doseći, šalje RREP paket ka čvoru A.
7. Čvor A prima RREP paket poslat od čvora C i pamti sve informacije o tome kako da dosegne čvor C (direktno), čvor D (preko čvora C) i što je najvažnije kako da dosegne čvor B (preko čvora C). Sada čvor A zna kako može doći do čvora B, i konačno može započeti sa prenosom paketa podataka preko čvora C.
8. Čvor C prima paket podataka namenjen čvoru B od čvora A. S obzirom da zna da čvor D može dostići čvor B, prosleđuje mu paket.

9. Čvor *D* prima paket podataka namenjen čvoru *B*. S obzirom da zna da čvor *B* može direktno dosegnuti, prosleđuje mu paket.
10. Čvor *B* prima paket podataka. S obzirom da zna povratnu putanju do čvora *A*, može mu odgovoriti sa novim paketom. Ovaj proces se ponavlja sve dok se komunikacija u potpunosti ne završi ili dok ne dođe do promene u mrežnoj topologiji (npr. čvor *C* napusti mrežu ili se neki drugi čvor *E* pridruži), pri čemu može biti neophodno ponovo pretražiti mrežu za novom rutom.

5.3.2.8. TORA

Privremeno uređen ruting algoritam (engl. *Temporally-Ordered Routing Algorithm, TORA*) [77] je ruting protokol razvijen za upotrebu u bežičnim meš mrežama i mobilnim ad hoc mrežama. Razvijen je od strane Vinsent Parka na Univerzitetu u Merilendu i Pomorskoj istraživačkoj laboratoriji (engl. *Naval Research Laboratory*), SAD. Parkov algoritam licenciran je od strane Nova Inženjeringa (engl. *Nova Engineering*) koji ga predstavljaju na tržištu.

TORA pokušava da postigne visok stepen skalabilnosti koristeći ravan, nehijerarhijski ruting algoritam. U svom radu algoritam u najvećoj mogućoj meri pokušava da potisne generisanje dalekosežnih kontrolnih poruka. Da bi se ovo postiglo, TORA ne koristi pristup korišćenja najkraćih ruting putanja koji je zastupljen kod većine ruting protokola ovog tipa.

TORA gradi i održava usmereni aciklični graf (engl. *Directed Acyclic Graph, DAG*) smešten na odredišnom čvoru. Ne postoje dva čvora koja mogu imati istu visinu. Informacije mogu teći sa čvorova sa većim visinama ka čvorovima sa nižim visinama. Stoga, informacije se mogu posmatrati kao fluid koji može teći samo nizbrdo. Održavanjem skupa visina čvorova tokom vremena TORA ostvaruje rutiranje sa preskocima bez petlji, jer informacije ne mogu teći uzbrdo i na taj način se vratiti na neki od već korišćenih čvorova.

Ključni koncept dizajna TORA ruting protokola je lokalizacija kontrolnih poruka na veoma mali skup čvorova blizu pojave topološke promene. Da bi se ovo postiglo, čvorovi održavaju ruting informacije o susednim čvorovima na razdaljini do jednog preskoka. Protokol obavlja tri osnovne funkcije:

- Kreiranje ruta,
- Održavanje ruta, i
- Brisanje ruta.

Tokom faza kreiranja i održavanja ruta, čvorovi koriste metriku visina da uspostave usmereni aciklični graf (DAG) smešten na odredištu. Nakon toga, određuju se veze na osnovu metrike relativne visine susednih čvorova. Usled mobilnosti čvorova, vremenom DAG graf se slama i tada proces održavanja ruta ponovo uspostavlja DAG graf smešten na odredišnom čvoru. Tajming je važan faktor kod TORA ruting protokola, jer metrika visina zavisi od logičkog vremena neuspeha linka.

Ukoliko čvor koji zahteva vezu ka odredištu nema nizvodne susede šalje upit (engl. *Query, QRY*) i postavlja zastavicu unutar paketa. QRY paket sadrži identifikator odredišnog čvora. Odgovor na upit se zove ažurirani paket (engl. *Update, UPD*). UDP paket sadrži upetostručenu visinu susednog čvora koji odgovara na upit, kao i odredišno polje koje govori za koji odredišni čvor je ova ispravka namenjena. Čvor koji primi QRY paket radi sledeće:

- Ako je njegova zastavica postavljena unutar paketa, to znači da ne mora da prosledi QRY paket svojim susedima, jer je on inicirao ovu komunikaciju. Stoga, odbacuje ovaj paket kako bi sprečio dodatno trošenje kapaciteta linkova.
- Ako čvor nema nizvodne veze i ako zastavica nije postavljena, postavlja svoju zastavicu i reemituje QRY poruku.

Čvor koji primi UPD paket ažurira vrednost visine svog suseda u tabeli i čini jednu od sledećih radnji:

- Ako bit refleksije visine susednog čvorova nije postavljen i ako je njegova zastavica postavljena, čvor postavlja svoju visinu prema odredištu za vrednost veću za jedan od svojih suseda. Čvor potom briše svoju zastavicu i šalje UPD poruku svojim susedima, tako da trasa može prolaziti kroz njega.
- Ako putanja ka susedu nije validna (što je označeno sa bitom refleksije) ili ako zastavica nije postavljena, čvor samo ispravlja vrednost visine susednog čvora u svojoj tabeli.

Svaki čvor održava tabelu susedstva koja sadrži visine susednih čvorova. U početku, visina svih čvorova je prazna (engl. *NULL*). Čvor koji zahteva rutu ka odredištu brodkastovaće QRY poruku. QRY poruka propagira se putem središnjih čvorova sve dok ne dosegne do čvora koji ima rutu do odredišta. Ovaj čvor potom šalje UPD poruku nazad ka izvornom čvoru. Proces održavanja ruta reaguje na promene topologije sa ciljem oporavka puteva.

Proces brisanja ruta kod TORA ruting protokola obuhvata brodkastovanje paketa za brisanje (engl. *Clear*, *CLR*) kroz mrežu, kako bi se izbrisali nevažeci putevi. Naime, kada čvor otkrije oblast, postavlja svoju visinu i visine svih svojih suseda za odredište u svojoj tabeli na praznu vrednost (*NULL*) i generiše paket za brisanje (*CLR*). *CLR* paket se sastoji od reflektovanog referentnog nivoa i odredišnog identifikatora. Kada čvor dobije *CLR* paket, ako referentni nivo unutar paketa odgovara njegovom referentnom nivou, čvor postavlja svoju visinu i visine svih svojih suseda za odredište u svojoj tabeli na praznu vrednost i reemituje *CLR* paket. Ako referentni nivo unutar primljenog *CLR* paketa ne odgovara sopstvenom referentnom nivou, čvor u svojoj tabeli postavlja visine suseda koje odgovaraju datom referentnom nivou na praznu vrednost i ažurira status njihovih veza na neusmerene.

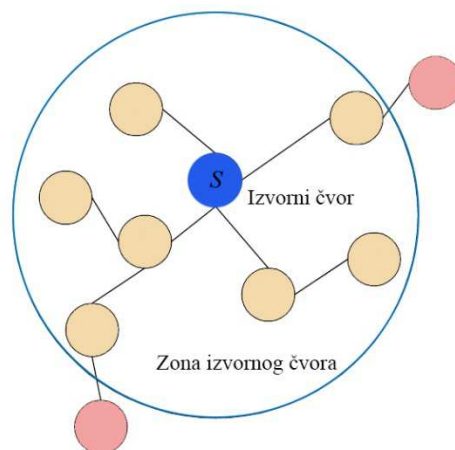
5.3.3. Hibridni ruting protokoli

Hibridni ruting protokoli pokušavaju da iskombinuju najbolje karakteristike proaktivnih i reaktivnih ruting algoritama. Često se sastoje od dva klasična ruting protokola: jednog proaktivnog i jednog reaktivnog. Hibridni ruting protokoli dele mrežu u oblasti koje se nazivaju zone, koje mogu biti preklapajuće ili nepreklapajuće u zavisnosti od algoritma koji se koristi kod određenog hibridnog ruting protokola. Ruting zona određenog čvora može se definisati kao udaljenost od tog čvora ili kao deo određenog geografskog područja. Proaktivni ruting protokol funkcioniše unutar zona, i odgovoran je za uspostavljanje i održavanje puteva do odredišta koja se nalaze unutar zona. S druge strane, reaktivni ruting protokol odgovoran je za uspostavljanje i održavanje puteva ka odredištima koja se nalaze izvan zona.

5.3.3.1. ZRP

Zonski ruting protokol (engl. *Zone Routing Protocol, ZRP*) [78] je hibridni ruting protokol koji objedinjuje tehnike reaktivnih i proaktivnih ruting protokola. Zona rutiranja određuje se radijusom zone, koji je zapravo minimalni broj preskoka između čvorova. Svaki čvor ima svoju zonu rutiranja i istovremeno je član drugih ruting zona, koje se međusobno preklapaju. ZRP koristi proaktivni ruting protokol unutar svake zone. Reaktivni ruting protokol se koristi za slanje paketa ka čvorovima koji se nalaze izvan date ruting zone. Veličina izabrane zone utiče na komunikacione performanse MANET mreže. ZRP je generalno pogodniji za primenu u mrežama sa velikim brojem čvorova

ZRP koristi činjenicu da je komunikacija između čvorova u ad hoc mrežama najčešće lokalizovana, čime su promene u topologiji čvorova u neposrednoj blizini od presudnog značaja. ZRP koristi ove osobine za definisanje okvira za komunikaciju čvora sa drugim postojećim protokolima. Lokalni skupovi susednih čvorova nazivaju se zone i definisani su za sve čvorove. Veličina zone zasniva se na p faktoru, koji je definisan kao broj preskoka do granice zone (slika 5.11). Kod ZRP ruting protokola postoje različite preklapajuće zone, koje pomaže u optimizovanju trase.

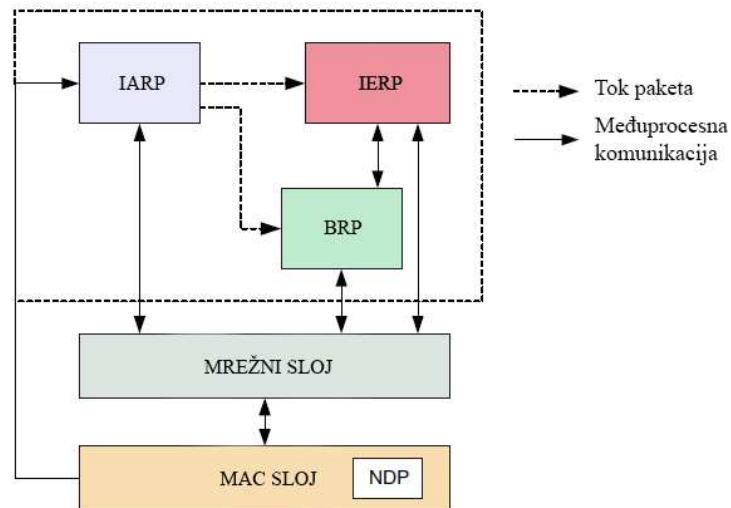


SLIKA 5.11. PRIMER ZRP ZONE IZVORNOG ČVORA SA FAKTOROM $P = 2$

ZRP predstavlja hibridni okvir protokola, što omogućuje korišćenje adekvatne ruting strategije u zavisnosti od različitih situacija. ZRP se može optimizovati da u potpunosti iskoristi prednosti bilo kog zastupljenog protokola. ZRP se zasniva na tri podprotokola:

- Proaktivni unutarzonski ruting protokol (engl. *Intrazone Routing Protocol, IARP*),
- Reaktivni međuzonski ruting protokol (engl. *Interzone Routing Protocol, IERP*), i
- Granični protokol za razrešenje (engl. *Bordercast Resolution Protocol, BRP*).

Arhitektura ZRP ruting protokola prikazana je na slici 5.12.



SLIKA 5.12. ARHITEKTURA ZRP RUTING PROTOKOLA

IARP koristi proaktivnu tehniku za otkrivanje prisustva i odsustva susednih čvorova, unutar jedne ruting zone. Svaki čvor održava svoju ruting zonu, kako bi se osiguralo da svi čvorovi unutar jedne zone imaju aktuelnu ruting tabelu sa svim ostalim čvorovima u zoni. Zahtevi za rutom izvan zone se propagiraju uz korišćenje perifernog faktora ρ , umesto tehnike plavljenja mreže.

IERP funkcioniše vrlo slično kao „na zahtev“ ruting protokoli u potrazi za ruting informacijama ka čvorovima koji se nalaze van svoje trenutne zone. IERP pokreće proces otkrivanja rute kada trasa do odredišta nije lokalno dostupna. Ova poruka će se prosledivati između središnjih čvorova sve dok se ne dosegne odredišni čvor ili dok se u keš memoriji na nekom od središnjih čvorova ne pronađe put do odredišta. Potom će poruka sa odgovorom na rutu biti poslata nazad ka izvornom čvoru čime se završava proces otkrivanja putanje.

S obzirom da su lokalne informacije unutar zone poznate IARP protokolu, otkrivanje trasa između zona postaje jednostavno. BRP protokol se koristi za usmeravanje upita za rutom ka regionima mreže koji još uvek nisu obuhvaćeni upitom. Ako se čvor nalazi unutar ruting zone koja je dobila upit, ovaj čvor postaje pokriven. Dakle, upit za rutom ne ulazi u druge zone, već samo u one zone koje se nalaze između izvorne i odredišne zone čime se značajno pojednostavljuje kontrola preopterećenja i smanjuje se broj dupliranih upita.

Proces otkrivanja ruta kod ZRP ruting protokola zasniva se na pretraživanju ruting tabele i/ili međuzonske pretrage. Kada se putanja prekine usled mobilnosti čvorova, ako je izvor mobilnost unutar zone, to će se tretirati kao događaj promene stanja veze i ruting ispravka koja se koristi u proaktivnom rutiranju će obavestiti sve ostale čvorove unutar zone. S druge strane, ako je izvor mobilnosti rezultat graničnog čvora ili čvora unutar neke druge zone, onda se obavlja zahtev za popravkom rute u vidu međuzonske pretrage za rutom, ili u najgorem slučaju, izvorni čvor se obaveštava o prekidu trase.

Pošto ZRP u osnovi koristi različite ruting protokole, njihove karakteristike će biti drugačije. Neki delovi trase zavise od pravilne konvergencije rutiranja, dok ostali delovi zavise od preciznosti otkrića međuzonskih ruta. Ovo otežava ostvarivanje stabilnosti celokupnog ZRP ruting protokola. Bez pravilne kontrole upita, ZRP se zapravo može ponašati lošije od standardnog ruting protokola zasnovanog na plavljenju mreže.

ZRP generalno pokazuje bolje performanse zahvaljujući korišćenju obe proaktivne i reaktivne ruting šeme. ZRP poseduje prednost u ograničavanju širenja periodičnih proaktivnih ruting ispravki između čvorova unutar jedne ruting zone. ZRP takođe ograničava pretragu reaktivnih ruta između odabranih perifernih čvorova. Međutim, ZRP je potencijalno neefikasan u slučajevima kada RREQ paket treba propagirati putem čitave mreže, kao i zbog činjenice da upotreba hijerarhijskog rutiranja prilikom pronalaženja puta do odredišta može biti podoptimalna. Sa povećanjem veličine ruting zone, veća količina memorije je potrebna na svakom čvoru usled povećanja količine topoloških informacija.

5.3.3.2. HSLS

Maglovito pronicljiv ruting protokol zasnovan na stanju veze (engl. *Hazy Sighted Link State, HSLS*) je bežični meš ruting protokol razvijen od strane CUWiN fondacije [79]. Ovaj algoritam omogućuje komunikaciju putem digitalnog radija u meš mreži kako bi se poruke prosledile do čvorova koji su van domašaja direktnog radio kontakta. Dodatni utrošak resursa usled generisanja sistemskih poruka teorijski je optimalan, pri čemu se koristi proaktivno i reaktivno rutiranje zasnovano na stanju veze kako bi se ograničilo ažuriranje mrežne topologije.

HSLS je razvijen da bude skalabilan za uporebu u mrežama sa preko hiljadu čvorova, pri čemu na većim mrežama počinje da prevazilazi efikasnost ostalih ruting protokola. Ovo je postignuto korišćenjem pažljivo osmišljene ravnoteže učestalosti ažuriranja i obima ispravki, kako bi se optimalno propagirale informacije o stanju veza. Za razliku od tradicionalnih metoda, HSLS ne preplavljuje mrežu sa informacijama o stanju veza, već pokušava da se izbori sa pokretnim čvorovima koji menjaju veze sa ostatkom mreže. HSLS ne zahteva da svaki čvor ima isti pogled na mrežu.

Ruting protokoli zasnovani na stanju veze pronalaze optimalne rute, pri čemu smanjuju rasipanje prenosnog kapaciteta. HSLS ruting protokol poseduje karakteristike različitih ruting šema: proaktivnih, reaktivnih, kao i algoritmama koji prihvataju podoptimalno rutiranje. Ove ruting strategije su pomešane u cilju ograničavanja količine poruka koje sadrže ispravke stanja veza u vremenu i prostoru. Ograničavanjem vremena života paketa (engl. *time-to-live*) vrednost iskorišćenog kapaciteta linkova se smanjuje. Ograničavanjem perioda kada se odašilju ažurirane proaktivne ruting informacije, nekoliko ispravki mogu biti prikupljene i prenešene odjednom, pri čemu se ostvaruje ušteda prenosnog kapaciteta linkova.

HSLS koristi različite ruting tehnike:

- Po definiciji, ruting protokoli zasnovani na stanju veze koriste dostupne informacije kako bi pronašli najbolju putanju, tako da rutiranje bude optimalno.
- Podoptimalno rutiranje dešava se prirodno, jer udaljeni čvorovi ređe dobijaju informacije.
- Minimiziranje broja proaktivnih ispravki nije trivijalan zadatak. Šema HSLS ruting protokola je nastala modifikacijom dva postojeća ruting algoritma zasnovana na stanju veze. Prvi, kratkovidi ruting algoritam zasnovan na stanju veze (engl. *Near-Sighted Link-State Routing*) je ograničen u prostoru, odnosno u broju preskoka čvorova putem kojih ruting informacije mogu biti prenešene. Drugi, diskretni ruting algoritam zasnovan na stanju veze (engl. *Discretized Link-State Routing*), ograničava vremenski interval tokom kojeg ruting informacije mogu biti prenešene. Stoga, kao rezultat dobijena je periodična proaktivna ispravka sa fraktalnim stepen od dva

brojem preskoka čvorova za prenos podataka (npr. broj preskoka od 1, 2, 1, 4, 1, 2, 1, 8, ...).

- Reaktivno rutiranje javlja se usled toga što neuspešan pokušaj korišćenja susedne veze dovodi do toga da tajmer istekne. Tokom svakog uzastopnog neuspeha, reakcija se eskalira na prošireni skup čvorova u meš mreži.

Tvorci HSLs ruting protokola radili su na finom podešavanju ovih stavki definisanjem mere globalnog mrežnog otpada. Ovo uključuje otpad koji nastaje emitovanjem ispravki ruting putanja, i otpad proistekao iz neefikasnih prenosnih puteva. Ukupno dodatno trošenje resursa (engl. *overhead*) definisano je kao razlika iznosa protoka koji se koristi u odnosu na vrednost minimalnog potrebnog propusnog opsega da bi se prosledili paketi putem najkraćeg rastojanja (mereno u broju preskoka čvorova), uz pretpostavku da čvorovi u datom trenutku imaju informacije o kompletnoj mrežnoj topologiji.

Potom su napravili neke razumne pretpostavke i iskoristili matematičku optimizaciju da bi pronašli vremena za slanje ispravki o stanju veza, kao i širinu skupa čvorova koju bi ispravke o stanju veza trebale pokrivati. U osnovi, otkrili su da bi rast ovih vrednosti tokom vremena trebao biti vezan za stepen broja dva.

Lokalno slanje ruting ispravki neophodno je svaki put kada se veza izgubi. Ovo je reaktivni deo HSLs algoritma. Lokalne ruting ispravke ponašaju se isto kao i funkcija isteka tajmera. Inače, svaki put kada se kašnjenje od poslednjeg ažuriranja udvostruči, čvor šalje ruting informaciju koja udvostručuje broj preskoka čvorova koji se razmatra. To traje do neke gornje granice. Gornja granica daje mreži globalnu veličinu i osigurava fiksnu vrednost maksimalnog vremena odziva za mrežu bez mobilnih čvorova.

HSLs algoritam ima nekoliko specijalnih funkcija koje otklanjaju probleme zajedničke za radio mreže, kao što su jednosmerne veze i ruting petlje. Konkretno, HSLs preusmerava sve konekcije na susedne čvorove kada izgubi vezu sa susednim čvorom. Pritom, reemituje sva svoja susedstva kada se to desi. Ovo je naročito korisno za veze sa velikim brojem preskoka, koje su inače najmanje pouzdane u radio mrežama.

HSLs algoritam je u osnovi veoma jednostavan. HSLs uspostavlja prilično dobre puteve u realnom vremenu, i znatno smanjuje broj i veličinu poruka koje se odašilju u cilju povezivanja i održavanja mreže, u poređenju sa mnogim drugim protokolima. Mnogi jednostavniji MANET ruting protokoli jednostavno preplavljaju celu mrežu sa ruting informacijama kad god dođe do promene veze. Ruting informacije i prenos podataka su decentralizovani, pa stoga imaju visoku pouzdanost i performanse.

HSLs daje veoma brzu i relativno preciznu pretpostavku o tome da li je određeni čvor u mreži, jer je kompletna, mada uglavnom nesinhronizovana ruting tabela prisutna u svakom čvoru. Međutim, ova pretpostavka nije adekvatna za visoko pouzdane sisteme.

HSLs ima dobre osobine skalabilnosti. Asimptotska skalabilnost ukupnog preopterećenja (engl. *overhead*) iznosi $O(n^{3/2})$ što je znatno niže u odnosu na standardne ruting protokole zasnovane na stanju veze kod kojih ova vrednost iznosi $O(n^2)$, gde je n broj čvorova u mreži.

Osnovni nedostatak HSLs ruting protokola ogleda se u činjenici da se zahtevaju čvorovi sa velikim količinama memorije radi očuvanja ruting tabela. Takođe, HSLs ruting protokol retko šalje udaljene ruting ispravke, tako da čvorovi nemaju najnovije informacije o tome da

li je neki udaljeni čvor i dalje prisutan. Ovo pitanje u izvesnoj meri prisutno je u svim ruting protokolima zasnovanim na stanju veze, jer ruting tabela može sadržati informacije o čvoru koji više nije prisutan na mreži. Međutim, pojedini ruting protokoli imaju mehanizme koji propagiraju informacije sa susednih čvorova od čvora koji je napustio mrežu, pa uskoro svi čvorovi unutar mreže saznaju tu informaciju. Kod HSLS ruting protokola, udaljene ruting informacije tek povremeno se šalju. Stoga, HSLS može zakazati u nekim okolnostima koje zahtevaju visoku pouzdanost.

5.3.3.3. HWMP

Hibridni bežični meš ruting protokol (engl. *Hybrid Wireless Mesh Protocol, HWMP*) definisan IEEE 802.11s standardom je ruting protokol namenjen za upotrebu u bežičnim meš mrežama [80]. Zasniva se na upotrebi proaktivnog AODV ruting protokola i ruting tehnike zasnovane na formiranju stabla. HWMP se oslanja na ravnopravnom link upravljačkom protokolu (engl. *Peer Link Management*) pomoću kojeg svaka meš tačka otkriva i prati susedne čvorove. Ukoliko je bilo koji od ovih čvorova povezan sa spoljašnjim ožičenim kanalom, u tom slučaju nema potrebe za HWMP ruting protokolom, koji bira puteve među onima koji su dobijeni objedinjavanjem svih meš tačaka u jednu kompozitnu mapu.

HWMP je hibridni ruting protokol, jer podržava dve različite metode za odabir ruta. Iako su ovi protokoli veoma slični, u slučaju korišćenja IEEE 802.11s protokola koriste se MAC adrese za rutiranje, umesto IP adresa.

HWMP je napravljen sa ciljem da zameni različite vendorske protokole koji se inače koriste za istu svrhu, dozvoljavajući korišćenje softvera sa otvorenim kodom (engl. *open source*) na čvorovima.

5.3.3.4. OORP

OrderOne ruting protokol (engl. *OrderOne Routing Protocol, OORP*) je ruting algoritam namenjen za rad u bežičnim meš mrežama [58]. OORP ostvaruje komunikaciju između bežičnih čvorova putem razumno efikasnih puteva.

Tvorci OORP ruting protokola navode da je protokol u stanju da obradi više hiljada mobilnih čvorova, što je ograničavajući problem kod većine drugih ruting protokola. OORP koristi hijerarhijske algoritme kako bi minimizirao ukupnu količinu generisanog kontrolnog saobraćaja neophodnog za kreiranje ruta između čvorova. Količina ovog saobraćaja je ograničena na vrednost između 1 i 5 % od ukupnog kapaciteta linka između dva čvora u bilo kojoj mreži i ne raste sa povećanjem veličine mreže.

Osnovna ideja OORP ruting algoritma je da organizuje mrežu u strukturu stabla. U početku, čvorovi se u susreću u korenu stabla kako bi uspostavili početnu rutu. Trasa se zatim udaljava od korena pod različitim isečenim uglovima. Kada više ne postoje uglovi koji se mogu iseći, gotovo optimalna ruta postoji. Ovaj put se stalno održava. Svaki proces se obavlja uz minimalnu lokalnu komunikaciju i kreiranje veoma malih ruting tabela. Od 2004. godine OORP je patentiran kao zaštićena intelektualna svojina.

Osnovna pretpostavka za izvršavanje OORP ruting protokola je da svaki čvor unutar mreže ima jedinstveno ime, barem jednu vezu, i memoriju određenog kapaciteta za održanje liste susednih čvorova. Mrežni čvorovi formiraju hijerarhiju, tako što svaki čvor odabire roditelja. Roditelj je susedni čvor koji je sledeći najbolji preskok ka većini drugih čvorova. Ovaj metod stvara hijerarhiju oko čvorova koji imaju veću verovatnoću da će ostati prisutni, i koji imaju veći kapacitet, a uz to su bliži topološkom centru mreže. Memorijska ograničenja malih čvorova se ogledaju u njihovoj maloj ruting tabeli, što ih automatski sprečava da budu predodređeni za centralno čvorište. Na vrhu stabla, jedan ili dva čvora nisu u stanju da pronađu čvorove bolje povezane od samih sebe, te stoga postaju roditelji za celu mrežu. Ovakav hijerarhijski algoritam ne zahteva kompleksan ruting algoritam, niti razmenu velike količine kontrolnih poruka.

Svi čvorovi unutar mreže informaciju o putu do sebe prosleđuju čvoru koji je koren stabla. Stoga, čvor koji traži putanju do odredišta šalje zahtev ka čvoru koji je korena stabla, i od njega uvek dobija informaciju o traženoj putanji. Komercijalna verzija OORP ruting protokola koristi Dijkstrin algoritam za kontinuirano optimizovanje i održavanje ruta. S obzirom da su čvorovi mobilni i da se mrežna topologija neprestano menja, OORP stalno prilagođava ruting putanje.

Pod pretpostavkom da pojedini čvorovi u mreži imaju dovoljnu količinu memorije za skladištenje informacija o svim čvorovima u mreži, ne postoji praktično ograničenje u veličini mreže. Pošto je količina protoka rezervisanog za kontrolne poruke ograničena da bude manja od 5 % od ukupnog kapaciteta veze bez obzira na veličinu mreže, količina kontrolnog protoka ne bi trebalo da se povećava sa porastom veličine mreže. Većina ruting protokola skalira ili smanjenjem količine generisanih proaktivnih ruting informacija ili reaktivnim „na zahtev“ rutiranjem. OORP ruting protokol koristi obe proaktivne i reaktivne metode. Pravilno podešen, OORP ruting protokol može se koristiti u mreži sa preko 10.000 čvorova, iako koristi ograničen propusni opseg veza od svega 5 % za ruting informacije.

OORP na pouzdan način utvrđuje da neki čvor više nije u mreži i pritom generiše samo neznatnu količinu kontrolnih poruka, što je veoma vredna osobina ruting protokola u ad hoc bežičnim mrežama.

Međutim, centralni čvorovi nose dodatni teret, jer moraju imati dovoljno memorije za skladištenje informacija o svim čvorovima unutar mreže. Na određenom broju čvorova, mreža će stoga prestati da skalira. Ukoliko su svi čvorovi u mreži niskog kapaciteta, mreža može biti preplavljena paketima usled učestalih promena. Olakšavajuća okolnost je što u skoro svim realnim mrežama, što se više približavamo od perifernih ka centralnim čvorovima propusni opseg veza raste.

OORP ruting protokol ne uključuje mehanizme za bezbednost i autentifikaciju. Međutim, bezbednost i autentifikacija mogu se obezbediti unutar integratora protokola. Tipične mere bezbednosti uključuju šifrovanje, kao i potpisivanje paketa i uvećavanje brojača paketa prilikom njihovog slanja kako bi se sprečio napad ponovljenog slanja poruka (engl. *Replay Attack*).

5.4. Poređenje unikast ruting protokola

Proaktivni ruting protokoli gube dosta vremena u ažuriranju ruting tabela na čvorovima. Stoga, kada se mrežna topologija često menja, većina aktuelnih ruting informacija u ruting tabelama različitih čvorova može biti pogrešna. Dakle, proaktivni ruting protokoli se preporučuju za upotrebu u ad hoc mrežama sa umerenom dinamikom.

Prednost proaktivnog ruting pristupa ogleda se u činjenici da se trasa do odredišta može odmah izabati, bez odlaganja. Nedostaci su generisanje veće količine kontrolnog saobraćaja i samim time većeg korišćenja propusnog opsega, čak i kada se mreža nalazi u stanju mirovanja, što može dovesti do zagušenja mreže. Pored toga, proaktivni ruting protokoli sporije reaguju na topološke promene unutar mreže.

Kod proaktivnih ruting protokola osnovni problem je ažuriranje aktuelne ruting tabele na svim čvorovima unutar mreže. Proaktivni ruting protokoli zahtevaju periodičnu razmenu ruting informacija između čvorova. Međutim, ova razmena informacija može izazvati brodcast oluju usled ogromnog broja brodcastovanih poruka kada je mobilnost čvorova visoka.

Reaktivni ruting protokoli unose kašnjenje u komunikaciji usled određivanja trase. Preporučuju se za upotrebu u ad hoc mrežama kod kojih čvorovi imaju visok stepen mobilnosti.

Prednost reaktivnog ruting pristupa su niža vrednost propusnog opsega koja se koristi za održavanje ruting tabele. Takođe, energetski su efikasniji. Međutim, ostvaruju znatno veće kašnjenje prilikom uspostavljanja komunikacionih puteva.

Reaktivni ruting protokoli suočavaju se sa problemom skaliranja kada u mreži postoji veliki broj čvorova. Obim ovog problema zavisi od toga koji reaktivni ruting protokol se koristi i od posmatranog scenarija.

Proaktivni ruting pristup se oslanja na mehanizam ažuriranja ruting tabele koji podrazumeva stalno propagiranje ruting informacija, bez obzira da li se i koliko često ove rute koriste. Međutim, ovo nije slučaj kod reaktivnih ruting protokola. Kada čvor koji koristi reaktivni ruting protokol želi da ostvari komunikaciju sa udaljenim čvorom do koga ne poznaje putanju, moraće sačekati dok takva trasa ne bude otkrivena. S druge strane, kod proaktivnog ruting protokola sve ruting informacije su stalno na raspolaganju u ruting tabeli svakog čvora, bez obzira na to da li se koriste. Ova osobina, iako korisna, generiše značajnu količinu kontrolnog saobraćaja unutar mreže i znatnu potrošnju energije. Pošto su propusni opseg i baterijsko napajanje ograničeni resursi unutar MANET mreže, ovo postaje ozbiljno ograničenje.

Prednosti i nedostaci proaktivnih, reaktivnih i hibridnih ruting pristupa dati su u tabeli 5.2 [13].

TABELA 5.2. SUMARNE KARAKTERISTIKE PROAKTIVNIH, REAKTIVNIH I HIBRIDNIH RUTING PROTOKOLA

<i>Karakteristika</i>	<i>Proaktivni (zasnovani na parametrima iz tabele)</i>	<i>Reaktivni (na zahtev)</i>	<i>Hibridni</i>
Organizacija mreže	Ravna/Hijerarhijska	Ravna	Hijerarhijska
Širenje topoloških informacija	Periodično	Na zahtev	Oba metoda
Dostupnost ruta i kašnjenje	Uvek su dostupne, bez kašnjenja	Dostupne su po potrebi. Postoji kašnjenje	Oba metoda
Suočavanje sa mobilnošću čvorova	Periodične ispravke	Održavanje ruta	Oba metoda
Količina generisanih kontrolnih poruka	Visoka	Niska	Srednja

Kada je reč o izboru najoptimalnijeg ruting protokola unutar MANET mreže, reaktivni pristup je bolji kada je mobilnost čvorova visoka i kada je količina saobraćaja koja se prenosi između čvorova mala. S druge strane, proaktivni ruting pristup je preporučljiv kada je mreža generalno statična i kada je količina generisanog saobraćaja između čvorova visoka. Ako je potrebno ostvariti balans između različitih modela mobilnosti čvorova i različitih količina generisanog saobraćaja među čvorovima unutar jedne mreže, preporučuje se upotreba hibridnih ruting protokola koji kombinuju obe klase protokola, proaktivne i reaktivne.

Tabela 5.3 daje sumarno poređenje najvažnijih karakteristika između različitih ruting protokola [14].

TABELA 5.3. SUMARNO POREĐENJE KARAKTERISTIKA RAZLIČITIH RUTING PROTOKOLA

<i>Proto- kol</i>	<i>Ruting tehnika</i>	<i>Metrike</i>	<i>Otkriva- nje susednih čvorova</i>	<i>Opora- vak putanje</i>	<i>Skladi- šte za rute</i>	<i>Rešava- nje proble- ma povrat- nih petlji</i>	<i>Više- stru- ke puta- nje</i>	<i>Količina generisa- nih kontrol- nih poruka</i>	<i>Odlike</i>
DSDV	Proakti- vna / Plavlje- nje	Najkraći put	Pozdrav- ne poruke	Periodi- čni brodkast	Ruting tabela	Da. Pomoću rednih brojeva	Da	Visoka	Distri- buirani algori- tam
WRP	Proakti- vna	Najkraći put	Pozdrav- ne poruke	Periodi- čne ispravke	Ruting tabela	Da. Pomoću rednih brojeva i provere informa- cija od prethod- nika	Da	Visoka. Raste sa poveća- njem mobilno- sti čvorova	Koristi po- zdravne poruke
CGSR	Proakti- vna	Najkraći put	Povre- meno brodka- stuje	Periodi- čne ispravke	Ruting tabela	Da. Ograni- čen obim prenosa	Ne	Niska. Odabir glava klastera	Glava klastera je kritičan

			tabelu članova klastera			ka klasteru		izaziva povećanje troškova	čvor
OLSR	Proaktivna	Najkraći put	Pozdravne poruke	Periodične ispravke	Ruting tabela	Ne uvek.	Ne	Visoka	Koristi MPR čvorove za rutiranje
AODV	Reaktivna / Plavljenje	Najnovija ruta, najkraći put	Pozdravne poruke	Lokalna popravka	Ruting tabela	Da. Pomoću rednih brojeva	Ne	Visoka	Vodi računa samo o sledećem preskoku u ruti
DSR	Reaktivna / Plavljenje	Najkraći put	Nema	Nova ruta, obaveštava izvor	Keš za rute	Da. Navođenje tačne putanje	Da	Visoka	U potpunosti na zahtev
ABR	Reaktivna / Zasnova na stabilnosti bežične veze	Najjača povezanost	Periodične signalne poruke između susednih čvorova	Lokalni brodkast	Ruting tabela	Da. Pomoću rednih brojeva	Ne	Umerena	Visoko kašnjenje prilikom oporavka ruta
SSA	Reaktivna	Najjača snaga signala	Pozdravne poruke	Nova ruta, obaveštava izvor	Ruting tabela	Da.	Da	Umerena	Koristi tabelu stabilnosti signala
TORA	Reaktivna	Najkraći put, relativna visina čvorova	Pozdravne poruke	Obrnuta veza	Ruting tabela	Da.	Da	Visoka	Kontrolni paketi lokalizovani su na području topološke promene
ZRP	Hibridna	Najkraći put	Unutarzonske pozdravne poruke	Započinje ispravku na mestu prekida	Međuzonske, unutarzonske tabele	Da. Pomoću rednih brojeva	Da	Niska u spoljnoj zoni	Ruting opseg definisan je brojem preskoka

TREĆI DEO

***SOPSTVENI RUTING PROTOKOL
U MOBILNIM AD HOC MREŽAMA***

6. SOPSTVENI RUTING PROTOKOL

Ruting protokol za MANET mreže predstavljen u ovom radu nazvan je „Optimizovano rutiranje zasnovano na stanju veze - Belman-Ford“ (engl. *Optimized Link State Routing Protocol - Bellman-Ford, OLSR-BF*). Originalni proaktivni OLSR ruting protokol [29] predstavljen u poglavlju 5.3.1.7 iskorišćen je kao osnova za razvoj novog ruting protokola, prvenstveno usled njegove opšte prihvaćenosti, široke zastupljenosti i dokazano dobrih karakteristika u poređenju sa drugim proaktivnim ruting protokolima.

OLSR-BF je po svojoj prirodi proaktivni ruting protokol, te su kod njega putevi do svih odredišta uvek na raspolaganju. Proces otkrivanja ruta je zasnovan na brodkastovanju pozdravnih poruka i poruka kontrole topologije (TC). Pored toga, OLSR-BF se zasniva na konceptu MPR čvorova.

Kod OLSR-BF ruting protokola svaki čvor bira grupu susednih čvorova kao MPR čvorove. Dakle, za svaki čvor postoji grupa susednih čvorova koji se zovu MPR selektori. Samo čvorovi izabrani kao MPR čvorovi, mogu prosleđivati kontrolne informacije unutar mreže. Zahvaljujući tome, smanjena je količina emitovanih kontrolnih poruka i minimizirano je opterećenje linkova. Ruting informacije se periodično ažuriraju putem kontrolnih poruka poslatih od strane MPR čvorova. Funkcionisanje OLSR-BF ruting protokola ne zavisi od centralnog entiteta.

Prethodnik novopredloženog OLSR-BF ruting protokola, OLSR, se smatra kao dobar ruting protokol za guste i velike mreže zahvaljujući optimizaciji MPR čvorova. Slično kao i njegov prethodnik, OLSR-BF je ruting protokol zasnovan na preskocima. Dakle, svaki čvor koristi svoje lokalne ruting informacije za prosleđivanje paketa podataka do odredišta.

Uopšteno govoreći, ruting protokoli mogu se klasifikovati u dve široke kategorije na osnovu toga da li ruting protokol bira najbolju putanju izračunavanjem stanja svake veze na putu i potom pronalazi putanju koja ima najnižu ukupnu metriku do odredišta, ili bira najbolji put za rutiranje zasnovan na metrici razdaljine (udaljenost) i interfejsa (vektor). Na osnovu toga, razlikuju se dva opšteprihvaćena ruting algoritma za prosleđivanje paketa:

- ruting algoritmi zasnovani na stanju veze (engl. *Link-State, LS*) (npr. Dijkstra algoritam), i
- ruting algoritmi zasnovani na vektoru razdaljine (engl. *Distance Vector, DV*) (npr. Belman-Ford algoritam).

Iz trećeg poglavlja rada u kome je predstavljena teorija grafova uviđa se da za pronalažanje najkraćih rastojanja među čvorovima generalno postoje dva metoda:

- i. Pronalaženje najkraćih putanja od jednog izvora:
 - Dijkstrin algoritam (najbrži), i
 - Belman-Fordov algoritam (radi i sa negativnim težinama grana, ali se sporije izvršava od Dijkstrinog algoritma).
- ii. Pronalaženje najkraćih putanja među svim čvorovima unutar mreže:
 - Flojd-Voršalov algoritam (jednostavan za programiranje, ali ujedno i najsporiji).

Novopredloženi OLSR-BF ruting protokol u mnogome se zasniva na postojećem OLSR ruting protokolu [29], uz izuzetak korišćenja Belman-Fordovog algoritma umesto Dijkstrinog algoritma prilikom proračuna najkraćih rastojanja između čvorova u odnosu na izvorni čvor. Belman-Ford je algoritam zasnovan na vektoru razdaljine koji koristi dinamičko programiranje. Glavna prednost Belman-Fordovog algoritma u odnosu na Dijkstrin algoritam ogleda se u mogućnosti korišćenja grana sa negativnim težinama.

Osnovni razlog zbog kojeg su se tvorci originalne verzije OLSR ruting protokola [29] odlučili za korišćenje Dijkstrinog algoritma ogleda se u činjenici da Dijkstrin algoritam ima manje vreme izvršavanja prilikom proračuna najkraćih udaljenosti između čvorova u odnosu na izvorni čvor. Međutim, iako nešto kompleksniji i u vidu resursa zahtevniji, Belman-Fordov algoritam donosi znatna poboljšanja u vidu svakodnevnog funkcionisanja i širine upotrebe novopredloženog OLSR-BF ruting protokola u mobilnim bežičnim ad hoc mrežama.

Korišćenjem unidirekcionih veza između čvorova MANET mreža može se predstaviti usmerenim težinskim grafom, gde su grane usmerene od jednog ka drugom susednom čvoru (asimetrične, unidirekzione veze). Ukoliko između posmatranih čvorova postoji i grana usmerena u suprotnom smeru, veza između posmatranih čvorova smatra se simetričnom (bidirekciona veza).

Negativne težine grana omogućuju kreiranje asimetričnih veza između čvorova. Belman-Fordov algoritam podržava grafove sa negativnim težinama grana, što znači da podržava jednosmerne (usmerene) bežične veze. Korišćenjem Belman-Fordovog algoritma kod OLSR-BF ruting protokola omogućeno je namensko korišćenje asimetričnih veza, kako bi se iskoristile unidirekzione veze između susednih čvorova koje bi inače kod tradicionalnog OLSR ruting protokola bile odbačene.

Izvedba novopredloženog OLSR-BF ruting protokola urađena je u okviru OPNET Modeler mrežnog simulatora. Naime, OPNET Modeler mrežni simulator unutar instalacionog direktorijuma na čvrstom disku (engl. *hard drive*) čuva datoteke svih podržanih modela čvorova i protokola. Specifično, unutar direktorijuma `\\OPNET\models\std\manet\` čuvaju se datoteke vezane za MANET mreže. Među njima nalazi se i model originalnog OLSR ruting protokola, čiji se izvorni algoritam za proračun ruting tabele nalazi u okviru zasebne datoteke.

Korišćenjem Microsoft Visual Studio programskog alata i C/C++ programskog editora zamenjen je Dijkstrin algoritam sa Belman-Fordovim algoritmom unutar `\\OPNET\models\std\manet\olsr_rte.pr.c` datoteke (engl. *file*). Originalna verzija Belman-Fordovog algoritma neznatno je prilagođena za primenu u okviru OPNET Modeler mrežnog simulatora. Izvorni kod Belman-Fordovog algoritma u C programskom jeziku predstavljen je u celosti u prilogu ove disertacije.

U nastavku je predstavljen način funkcionisanja i opis formata paketa poruka novopredloženog OLSR-BF ruting protokola.

6.1. Osnovne funkcionalnosti OLSR-BF rutinog protokola

Osnovne funkcionalnosti OLSR-BF rutinog protokola zasnivaju se na sledećim procesima:

- otkrivanje susednih čvorova, i
- širenje topoloških informacija.

Svaki čvor otkriva susedne čvorove do kojih ima direktnu vezu. Svaki čvor šalje pozdravne poruke, koje sadrže listu suseda poznatih datom čvoru i stanje njihovih veza (veza može biti asimetrična, simetrična, MPR ili izgubljena). Pozdravna poruka se brodkastuje ka svim susedima udaljenim do jednog preskoka. Susedni čvorovi ne prosleđuju dalje primljenu pozdravnu poruku.

Pomoću emitovanja pozdravnih poruka čvor može otkriti susedne čvorove na udaljenosti do dva preskoka. Parametar „vreme-zadržavanja-suseda“ (engl. *Neighbour-hold-time*) se povezuje sa svakim otkrivenim susednim čvorom. Ovaj parametar omogućuje brisanje zastarelih suseda iz liste susednih čvorova.

Svaki čvor unutar mreže održava informacije o topologiji dobijene putem TC poruka. Svaki čvor koji je izabran kao MPR čvor širi TC poruke. TC poruke se brodkastuju po celoj mreži i prosleđuju se putem MPR čvorova. Zahvaljujući upotrebi MPR čvorova opterećenost mreže sa kontrolnim saobraćajem je smanjena. TC poruka se koristi za širenje MPR liste selektora. Na ovaj način, MPR čvor se proglašava kao čvor udaljen jedan preskok od MPR selektora čvorova, odnosno na takav način da je svaki MPR selektor čvor dostupan direktno putem MPR čvora.

6.2. OLSR-BF format paketa

OLSR-BF paketi su sadržani u okviru protokola za korisničke datagrame (engl. *User Datagram Protocol, UDP*). U ovom radu razmatra se samo Internet protokol verzije 4 (engl. *Internet Protocol version 4, IPv4*), što znači da se koriti 32-bitni adresni prostor. Svaki OLSR-BF paket enkapsulira jednu ili više poruka. Osnovna struktura jednog OLSR-BF paketa (slika 6.1) sadrži:

0 (biti 0-9)				1 (biti 10-19)										2 (biti 20-29)							3 (30-31)	
0	1		9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4		9	0	1
Dužina paketa										Redni broj paketa												
Tip poruke				Vreme validnosti						Veličina poruke												
Adresa pošiljaoca																						
Vreme života				Broj preskoka						Redni broj poruke												
Poruka																						
Tip poruke				Vreme validnosti						Veličina poruke												
Adresa pošiljaoca																						
Vreme života				Broj preskoka						Redni broj poruke												
Poruka																						

SLIKA 6.1. FORMAT OLSR-BF PAKETA

- i. Zaglavlje paketa (engl. *Packet Header*) sastoji se od dva polja:
 - Dužina paketa (engl. *Packet Length*): Dužina paketa izražena u bajtovima [B].
 - Redni broj paketa (engl. *Packet Sequence Number, PSN*): Definiše redni broj svakog OLSR-BF paketa. PSN se inkrementira za jedan svaki put kada se novi OLSR-BF paket prenosi.
- ii. Zaglavlje poruke (engl. *Message Header*) sastoji se od sedam polja:
 - Tip poruke (engl. *Message Type*): Ukazuje na tip poslate poruke. Tipovi poruka u rasponu od 0 do 127 su rezervisani.
 - Vreme validnosti (engl. *Validity Time, Vtime*): Ovo polje pokazuje za koliko vremena nakon prijema poruke čvor mora uzeti u obzir informacije koje su sadržane u poruci kao validne, osim ukoliko je u međuvremenu primljena novija poruka sa ažuriranim informacijama. Vreme validnosti predstavlja se sa četiri najviša bita u *Vtime* polju.

$$Vtime = C * (1 + a/16) * 2^b \text{ [sekund]}$$

gde su:

a – celobrojna vrednost predstavljena sa četiri najviša bita u *Vtime* polju,
b – celobrojna vrednost predstavljena sa četiri najniža bita u *Vtime* polju,
C – konstanta koja iznosi 1/16 deo sekunde (0.0625 sekundi).

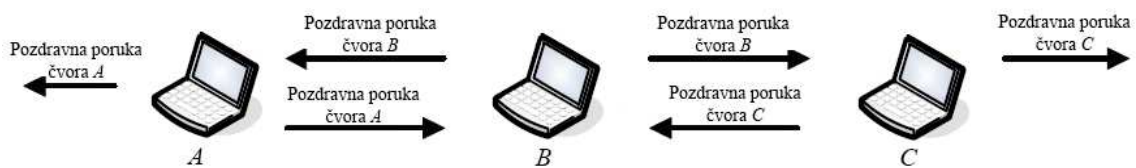
- Veličina poruke (engl. *Message Size*): Ovo polje predstavlja veličinu date poruke računatu u bajtovima, a meri se od početnog polja „tip poruke“ sve do početka sledećeg polja „tipa poruke“ (ili ako ne postoji sledeća poruka, do kraja paketa).
- Adresa pošiljaoca (engl. *Originator Address*): Ovo polje sadrži adresu čvora koji je prvobitno generisao ovu poruku. Ovo polje ne treba mešati sa izvornom adresom iz IP zaglavlja paketa, koja se menja svaki put na IP adresu središnjeg čvora koji je reemitovao ovu poruku. Ovo polje se nikada ne menja tokom reemitovanja paketa.
- Vreme života (engl. *Time-to-Live, TTL*): Ovo polje sadrži informaciju o maksimalnom broju preskoka tokom kojih će poruka biti reemitovana. Pre nego što se poruka reemituje, vrednost TTL polja se umanjuje za 1. Kada čvor primi poruku sa TTL vrednošću koja iznosi 0 ili 1, ovu poruku neće dalje reemitovati. U praksi, čvor neće dobiti poruku sa TTL vrednošću 0. Dakle, postavljanjem ovog polja, pošiljaoc poruke može ograničiti radijus doseganja poruke unutar mreže.
- Broj preskoka (engl. *Hop Count*): Ovo polje sadrži informaciju o broju preskoka koje je poruka dostigla. Pre nego što se poruka reemituje, vrednost ovog polja mora se uvećati za 1. U početku slanja, inicijator ove poruke postavlja vrednost polja na 0.
- Redni broj poruke (engl. *Message Sequence Number*): Prilikom generisanja poruke, izvorni čvor dodeljuje jedinstveni redni broj za svaku poruku i smešta ga unutar ovog polja. Redni broj se povećava za 1 za svaku narednu poruku koja potiče sa ovog čvora. Redni brojevi poruka se koriste kako bi se obezbedilo da se data poruka ne reemituje više od jednog puta sa bilo kog čvora unutar mreže.

6.3. OLSR-BF poruke

Proces funkcionisanja OLSR-BF rutinog protokola zasnovan je na konstantnoj razmeni različitih tipova kontrolnih poruka.

6.3.1. Pozdravne poruke

OLSR-BF se zasniva na otkrivanju susednih čvorova udaljenih do jednog preskoka. Svaki čvor otkriva susedne čvorove sa kojima ima direktnu (simetričnu ili asimetričnu) vezu. Proces otkrivanja susednih čvorova vrši se brodkastovanjem pozdravnih poruka (slika 6.2). Pozdravne poruke se generišu sa svakog čvora unutar mreže.



SLIKA 6.2. GENERISANJE I ŠIRENJE POZDRAVNIH PORUKA

Pozdravna poruka otkriva:

- spisak adresa susednih čvorova sa kojima izvorni čvor pozdravne poruke ima simetričnu ili asimetričnu vezu.
- spisak adresa susednih čvorova koji su aktivni.
- listu susednih čvorova koji su izabrani kao MPR čvorovi.

Pozdravna poruka se šalje putem opšteg formata paketa OLSR-BF poruke. U polje koje se odnosi na tip poruke upisuje se *HELLO_MESSAGE*, a TTL polje se postavlja na vrednost 1. Format pozdravne poruke prikazan je na slici 6.3.

0 (biti 0-9)				1 (biti 10-19)										2 (biti 20-29)							3 (30-31)	
0	1		9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4		9	0	1
Rezervisano										Interval emitovanja							Spremnost					
Kod veze				Rezervisano						Veličina poruke veze												
Adresa interfejsa susednog čvora																						
Adresa interfejsa susednog čvora																						
....																						
....																						
Kod veze				Rezervisano						Veličina poruke veze												
Adresa interfejsa susednog čvora																						
Adresa interfejsa susednog čvora																						

SLIKA 6.3. FORMAT POZDRAVNE PORUKE

Opis polja pozdravne poruke:

- Interval emitovanja pozdravne poruke (engl. *HELLO emission interval, Htime*): Ovo polje određuje interval emitovanja pozdravnih poruka koji ovaj čvor koristi na datom interfejsu. Interval emitovanja pozdravnih poruka predstavljen je sa četiri najviša bita *Htime* polja i njegovog eksponenta (četiri najniža bita u *Htime* polju). Drugim rečima:

$$Htime = C \cdot (1 + a/16) \cdot 2^b \text{ [sekund]}$$

gde su:

- a – celobrojna vrednost predstavljena sa četiri najviša bita u $Htime$ polju,
- b – celobrojna vrednost predstavljena sa četiri najniža bita u $Htime$ polju,
- C – konstanta koja iznosi 1/16 deo sekunde (0.0625 sekundi).

- Spremnost (engl. *Willingness*): Ovo polje određuje spremnost čvora da prosledi saobraćaj za druge čvorove. Ako se spremnost čvora definiše kao *WILL_NEVER*, taj čvor nikada neće biti izabran kao MPR čvor.
- Kod veze (engl. *Link Code*): Ovo polje određuje informacije o povezanosti između interfejsa pošiljaoca i liste susednih interfejsa. Takođe, ovo polje navodi informacije o statusu suseda. Kodovi veze koji su nepoznati čvoru tiho se odbacuju. Tako na primer, kod veze prikazuje da li je susedni čvor odabran kao MPR čvor od strane pošiljaoca.
- Veličina poruke veze (engl. *Link Message Size*): Predstavlja veličinu poruke veze izraženu u bajtovima, a meri se od početka polja „kod veze“ pa sve do sledećeg polja „koda veze“ (ili ako ne postoje više tipova veza, do kraja poruke).
- Adresa interfejsa susednog čvora (engl. *Neighbour Interface Address*).

Kada čvor primi pozdravnu poruku, može ažurirati informaciju o susedu sa adresom čvora koji šalje poruku. Takođe, ukoliko je to potrebno iz bezbednosnih razloga, čvor može ignorisati pozdravne poruke.

6.3.2. Poruke kontrole topologije

MPR čvorovi brodkastuju poruke kontrole topologije (TC poruke). Slika 6.4 ilustruje proces brodkastovanja TC poruke od strane MPR čvora. TC poruke nose listu suseda koji su izabrali emisioni čvor kao MPR čvor. Informacije koje prenose TC poruke neophodne su za izračunavanje ruting tabele na čvorovima.



SLIKA 6.4. BRODKASTOVANJE TC PORUKA

Format TC poruke prikazan je na slici 6.5.

0 (biti 0-9)				1 (biti 10-19)										2 (biti 20-29)				3 (30-31)	
0	1		9	0	1	2	3	4	5	6	7	8	9	0	1		9	0	1
Redni broj oglašenog susednog čvora										Rezervisano									
Glavna adresa oglašenog susednog čvora																			
Glavna adresa oglašenog susednog čvora																			

SLIKA 6.5. FORMAT TC PORUKE

Opis polja TC poruke:

- Redni broj oglašenog susednog čvora (engl. *Advertised Neighbour Sequence Number, ANSN*): Redni broj je povezan sa oglašenim skupom susednih čvorova. Svaki put kada čvor detektuje promenu u oglašenom skupu suseda, on povećava ovaj redni broj. Ovaj broj se smešta unutar ANSN polja TC poruke, kako bi se pratile najnovije informacije. Kada čvor primi TC poruku, na osnovu vrednosti ANSN polja može odlučiti da li su primljene informacije o oglašenim susednim čvorovima izvornog čvora novijeg datuma od informacija koje već poseduje.
- Rezervisano (engl. *Reserved*): Ovo polje rezervisano je za buduću upotrebu, i postavljeno je na vrednost „0000000000000000“ u skladu sa standardom RFC 3626 [29].
- Glavna adresa oglašenog susednog čvora (engl. *Advertised Neighbour Main Address*): Ovo polje sadrži glavnu adresu oglašenog susednog čvora. Sve glavne adrese oglašenih suseda izvornog čvora nalaze se unutar TC poruke. Ukoliko se dosegne maksimalna dozvoljena veličina poruke nametnuta od strane mreže, dok još uvek nisu ubačene sve oglašene adrese suseda unutar TC poruke, biće generisano više TC poruka dok se ne pošalju sve oglašene adrese suseda. Dodatne glavne adrese susednih čvorova mogu biti uključene unutar TC poruke ukoliko se želi ostvarivanje redundantnosti.

6.3.3. Poruke o terminalima i mrežnim udruženjima

Poruka o terminalima i mrežnim udruženjima (engl. *Host Network Association, HNA*) sadrži informacije o mrežnim adresama i njihovim maskama. HNA poruka se brodkastuje periodično tokom svakog HNA intervala (engl. *HNA Interval, HNA_INTERVAL*).

Iako HNA poruka nije deo jezgra OLSR-BF ruting protokola, koristi se kao dodatna funkcionalnost za proširenje mogućnosti uvođenja spoljašnjih ruting informaciju u OLSR-BF MANET mrežu. TC i HNA poruke su slične, jer se obe koriste za označavanje dostupnosti određenog čvora. Osim toga, obe poruke se šire i prosleđuju kroz mrežu putem MPR čvorova. Bitna razlika je u tome što informacije dobijene putem nove TC poruke mogu otkazati prethodno dobijene informacije, ako je novi redni broj veći. S druge strane, informacije koje dolaze iz HNA poruka mogu biti zamenjene tek kada se njihov životni vek završi.

Struktura HNA poruke prikazana je na slici 6.6.

0 (biti 0-9)				1 (biti 10-19)										2 (biti 20-29)							3 (30-31)	
0	1		9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4		9	0	1
Adresa mreže																						
Mrežna maska																						

SLIKA 6.6. FORMAT HNA PORUKE

Opis polja HNA poruke:

- Adresa mreže (engl. *Network Address*): Ovo polje se koristi da proglasi IPv4 adresu mreže od 32 bita.
- Mrežna maska (engl. *Netmask*): Ovo polje se koristi da proglasi masku date IPv4 adrese mreže, takođe dužine 32 bita.

HNA poruke se šire po celoj mreži putem MPR čvorova. Ove poruke prenose informacije o mrežnim adresama i njihovim maskama. U okviru ad hoc mreža mogu postojati pristupni ruteri (engl. *Access Router*, *AR*) koji su zaduženi za ostvarivanje veze između MANET mreže sa ožičenim mrežama. Pristupni ruteri podržavaju klasične ožičene i ad hoc mrežne ruting protokole.

Čvorovi koji prime HNA poruku vode registre sa informacijama iz ove poruke. Promenljiva *A_gateway_addr* proglašava adresu izvornog čvora HNA poruke. *A_network_addr* proglašava informaciju koja je dobijena iz polja adrese mreže unutar HNA poruke. *A_netmask* promenljiva se koristi za proglašavanje informacije dobijene iz polja mrežne maske unutar HNA poruke. Konačno, promenljiva *A_time* se koristi kao tajmer za prečišćavanje zapisa koji su istekli.

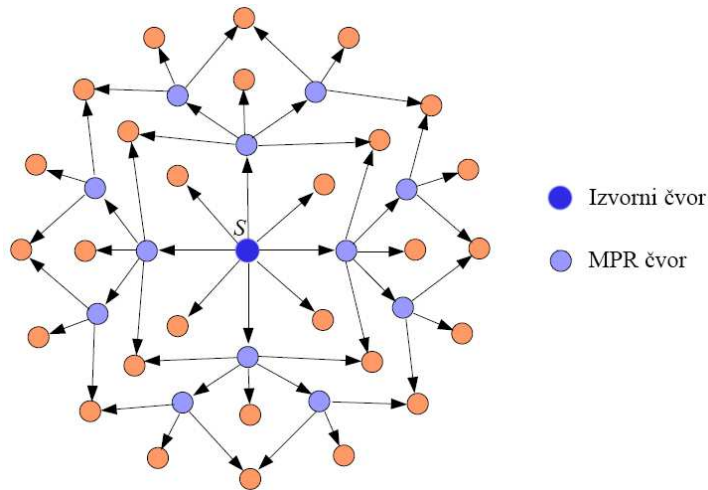
Obrada primljene HNA poruke se vrši putem sledećeg algoritma:

- i. Ukoliko predajnik (ne inicijator) HNA poruke nije susedni čvor udaljen do jednog preskoka, poruka se odbacuje.
- ii. U suprotnom, HNA poruka se obrađuje i informacije o mrežnoj adresi i njenoj masci se razmatraju:
 - a) Ako postoji registar gde je:
A_gateway_addr = adresa inicijatora
A_network_addr = adresa mreže
A_netmask = mrežna maska

ažurira se vreme održavanja:
 $A_time = \text{trenutno vreme} + \text{vreme validnosti}.$
 - b) Ako u HNA poruci ne postoji registar sa traženim informacijama, onda se promenljive *A_gateway_addr*, *A_network_addr* i *A_netmask* ažuriraju korišćenjem informacija iz HNA poruke. Na isti način inicijalizuje se *A_time* tajmer.

6.4. MPR čvorovi

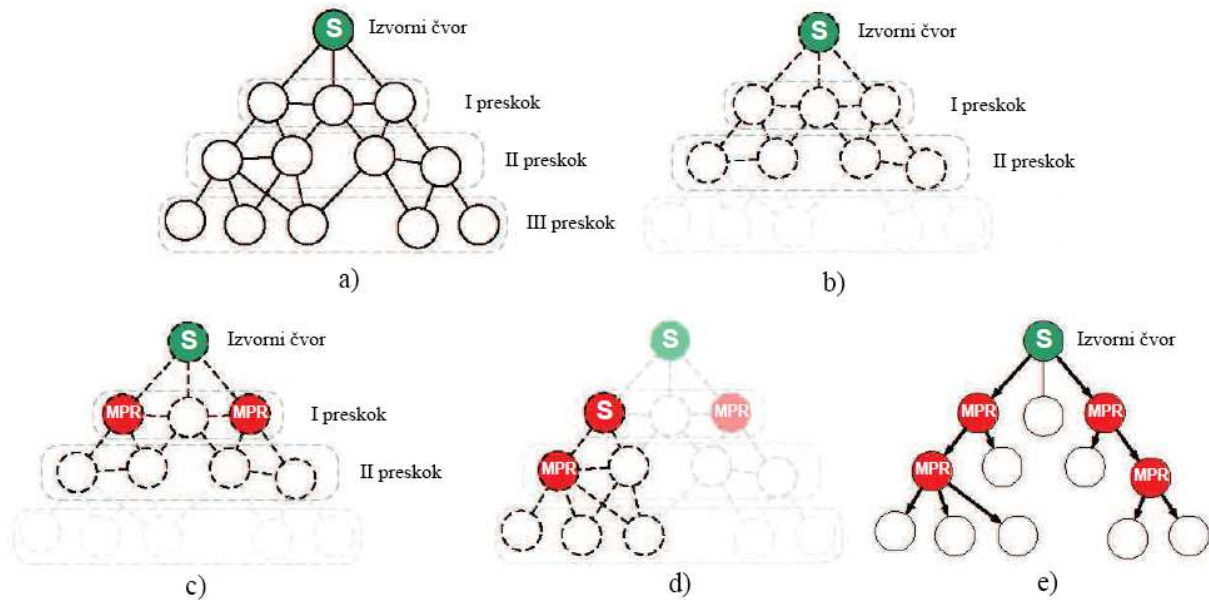
Cilj korišćenja MPR čvorova je smanjenje opterećenja prilikom širenja kontrolnih poruka kroz mrežu. Svaki čvor bira skup susednih čvorova koji mogu prosleđivati svoje poruke ka drugim čvorovima udaljenim do dva preskoka. MPR čvorovi moraju imati veze sa svim čvorovima udaljenim do dva preskoka. Tako formiran skup čvorova predstavlja MPR skup datog čvora. Kao što je prikazano na slici 6.7, susedi izvornog čvora *S* koji nisu u okviru MPR skupa primaju i obrađuju informacije, ali ne prosleđuju informacije koje potiču sa čvora *S*.



SLIKA 6.7. ODABIR MPR ČVOROVA

U isto vreme, MPR čvorovi održavaju informacije o skupu suseda udaljenih do jednog preskoka koji su ga izabrali kao MPR čvor. Ovaj skup je poznat kao MPR skup selektora. Ove informacije potiču iz pozdravnih poruka primljenih od susednih čvorova udaljenih do jednog preskoka.

Susedni čvorovi udaljeni do dva preskoka otkrivaju se putem pozdravnih poruka. Na primeru sa slike 6.8.a pozdravne poruke izvornom čvoru omogućuju delimični pogled na mrežu (slika 6.8.b). Na osnovu toga, OLSR-BF ruting protokol određuje MPR čvorove za dati izvorni čvor, tako što određuje minimalni skup susednih čvorova udaljenih do jednog preskoka neophodnih da se dosegne svaki susedni čvor udaljen do dva preskoka (slika 6.8.c). Tako izabrani MPR čvorovi na osnovu njihovog lokalnog pogleda na mrežnu topologiju, takođe određuju svoje MPR čvorove (slika 6.8.d). Da bi se odredila globalna ruting tabela, svaki MPR čvor kroz mrežu šalje poruku kontrole topologije (TC) koja sadrži njegove susede udaljene do jednog preskoka. Samo čvorovi izabrani kao MPR čvorovi šalju TC poruke (slika 6.8.e) čime se znatno smanjuje količina generisanog kontrolnog saobraćaja unutar mreže. Na kraju procesa svi čvorovi unutar mreže, bili oni MPR čvorovi ili ne, poseduju ruting tabelu sa najkraćom putanjom do svakog čvora unutar mreže. OLSR-BF ruting protokol optimizuje plavljenje mreže brodkast paketima tako što izgrađuje mrežnu topologiju u obliku stabla za svaki izvorni čvor (slika 6.8.e).



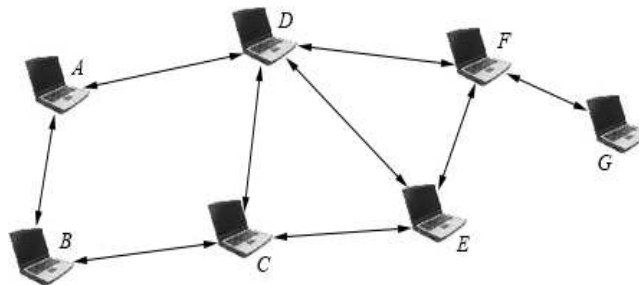
SLIKA 6.8. ODABIR MPR ČVOROVA KOD OLSR-BF RUTING PROTOKOLA

Svaki čvor bira skup MPR čvorova udaljenih do jednog preskoka. MPR čvorovi moraju biti u stanju da dosegnu sve susede udaljene do dva preskoka od izvornog čvora. Kada izvorni čvor odabere MPR čvorove, status veza sa tim čvorovima menja se iz stanja simetrične (engl. *Symetric Link*, *SYM_LINK*) ili asimetrične (engl. *Asymetric Link*, *ASYM_LINK*) veze u MPR veza (engl. *MPR Link*, *MPR_LINK*) u tabeli susedstva. Takođe, vrednost MPR rednog broja (engl. *MPR Sequence Number*, *MPR_Seq_Num*) se uvećava za jedan.

Skup MPR suseda se ponovo izračunava u sledećim slučajevima:

- Prilikom otkrivanja promene u stanju susedstva na udaljenosti od jednog preskoka. To može biti raskinuta veza ili dodavanje novog susednog čvora.
- Prilikom otkrivanja promene u stanju susedstva na udaljenosti od dva preskoka. To može biti raskinuta veza ili dodavanje novog susednog čvora između suseda na razdaljini od jednog preskoka i drugog suseda na razdaljini od dva preskoka.

Na primeru mreže od 7 čvorova (slika 6.9) u tabeli 6.1 predstavljeni su odabrani MPR čvorovi i MPR skupovi selektora za svaki čvor unutar mreže.



SLIKA 6.9. MANET MREŽA OD 7 ČVOROVA

TABELA 6.1. ODABRANI MPR ČVOROV I MPR SKUPOVI SELEKTORA

Čvor	MPR čvorovi	MPR skup selektora
A	D	/
B	C	/
C	D	B, D, E
D	C, F	A, C, E, F
E	C, D, F	/
F	D	D, E, G
G	F	/

6.5. Informaciona skladišta

OLSR-BF je rutinog protokol zasnovan na stanju veze, te stoga održava raznovrsne baze (skladišta) informacija. Skladišta informacija se ažuriraju putem dobijenih kontrolnih poruka, i u njima se čuvaju informacije koje se mogu koristiti prilikom generisanja novih kontrolnih poruka. Sledi kratak pregled različitih skladišta informacija koja se koriste kod OLSR-BF rutinog protokola (slika 6.10).

Skup veza (engl. *Link Set*)

Ova baza se održava za izračunavanje stanja veza sa susedima. To je jedina baza podataka koja radi na specifičnim „interfejs na interfejs“ vezama.

Skup suseda (engl. *Neighbor Set*)

Svi registrovani susedi udaljeni do jednog preskoka nalaze se u okviru ove baze. Podaci se ažuriraju dinamički na osnovu informacija iz skupa veza. Registrovani su susedi sa oba tipa veza, simetričnih i asimetričnih.

Skup suseda udaljenih dva preskoka (engl. *2 Hop Neighbor Set*)

Svi susedni čvorovi udaljeni do dva preskoka, ne računajući lokalni čvor, koji mogu biti dosegnuti putem susednih čvorova udaljenih jedan preskok, registrovani su u okviru ove baze.

Skup releja više čvorova (engl. *Multipoint Relay Set, MPR Set*)

Svi MPR čvorovi izabrani od strane lokalnog čvora registrovani su u ovoj bazi. MPR koncept objašnjen je u poglavlju 6.4.

Skup selektora releja više čvorova (engl. *Multipoint Relay Selector Set*)

Svi susedi koji su izabrali ovaj čvor kao svoj MPR čvor evidentiraju se u ovoj bazi.

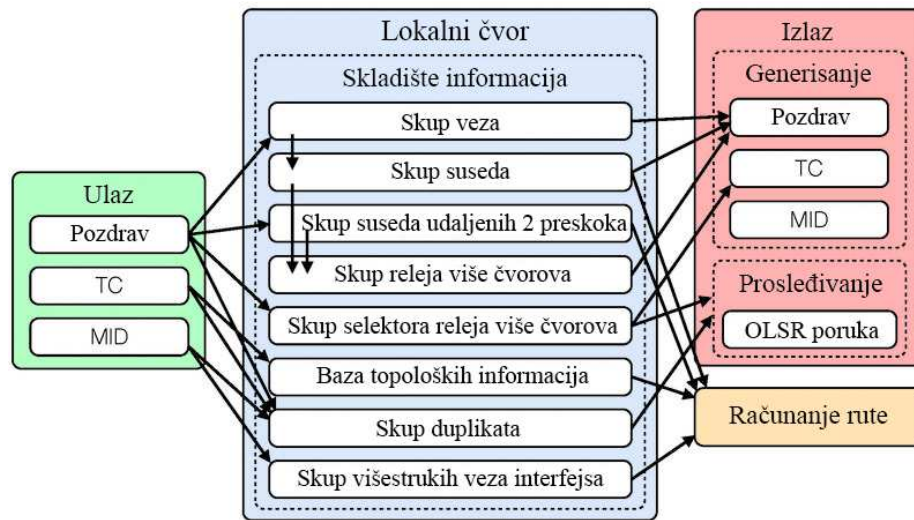
Baza topoloških informacija (engl. *Topology Information Base*)

Ovo skladište sadrži informacije o statusu svih veza koje su dobijene od čvorova u OLSR-BF rutinog domenu.

Skup duplikata (engl. *Duplicate Set*)

Ova baza podataka sadrži informacije o nedavno procesuiranim i prosleđenim porukama.

Skup višestrukih veza interfejsa (engl. *Multiple Interface Association Set*)
 Skup višestrukih veza interfejsa sadrži informacije o čvorovima koji koriste više od jednog komunikacionog interfejsa. Adrese svih interfejsa takvih čvorova čuvaju se u ovoj bazi.



SLIKA 6.10. INFORMACIONA SKLADIŠTA KOD OLSR-BF RUTING PROTOKOLA

Slika 6.10 daje pregled informacionih skladišta kod OLSR-BF ruting protokola i njihovih odnosa u obradi i generisanju poruka, a potom i u proračunu rute. Primljene pozdravne poruke prenose informacije o ispravkama što dovodi do promena u skupu veza (engl. *Link Set*). Ove informacije dalje izazivaju ispravke u skupu suseda (engl. *Neighbor Set*), koji potom aktivira pokretanje ponovnog proračuna skupa releja više čvorova (engl. *Multipoint Relay Set*, *MPR Set*). Skup suseda udaljenih dva preskoka takođe se ažurira na osnovu primljenih pozdravnih poruka, što opet dovodi do pokretanja proračuna MPR skupa. Konačno, MPR skup selektora se ažurira prema informacijama dobijenim u pozdravnoj poruci. Primljene TC poruke izazivaju ispravke u bazi topoloških informacija (engl. *Topology Information Base*), dok se skup višestrukih veza interfejsa (engl. *Multiple Interface Association Set*) ažurira putem primljenih MID poruka. Sve primljene poruke će biti registrovane i u skupu duplikata, izuzev ako se tamo već nalaze njihovi zapisi.

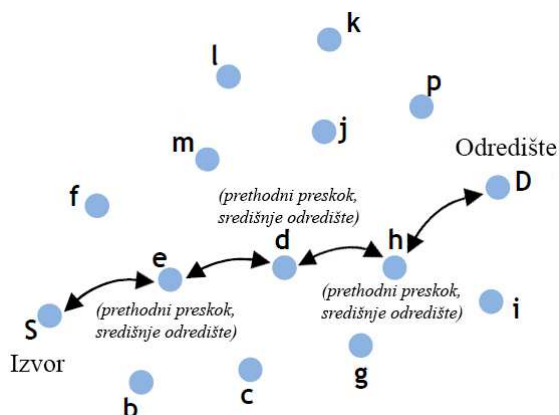
Prilikom generisanja pozdravne poruke koriste se informacije iz skupa veza, skupa suseda i skupa releja više čvorova (MPR). Prilikom generisanja TC poruke zadaje se upit u okviru skupa selektora releja više čvorova. Kod prosleđivanja kontrolnog saobraćaja koristi se MPR skup selektora i skup duplikata.

Konačno, proračun rute se zasniva na informacijama dobijenim od skupa suseda, skupa suseda udaljenih dva preskoka, baze topoloških informacija i skupa višestrukih veza interfejsa.

6.6. Proračun ruting tabele

Svaki čvor održava ruting tabelu. Ruting tabela se izračunava na osnovu informacija dobijenih od susednih čvorova i na osnovu mrežne topologije. Čvorovi koji prime TC poruku čuvaju informacije o parovima povezanih čvorova (prethodni preskok, čvor), gde su „čvorovi“ adrese koje se nalaze u TC listi poruka. Da bi izvorni čvor *S* pronašao putanju do udaljenog odredišnog čvora *D*, zapis (prethodni preskok, *D*) mora biti pronađen u okviru

ruting tabele. Nakon što je zapis pronađen, prethodni preskok postaje središnje odredište (engl. *Intermediate Destination, Destination_inter*) i sada se traži sledeći par (prethodni preskok, središnje odredište). Ovaj proces se sukcesivno izvršava sve dok se ne pronađe čvor koji je prethodni preskok, a nalazi se u skupu suseda izvornog čvora. Na slici 6.11 prikazan je proračun kompletne putanje od izvornog čvora *S* do odredišnog čvora *D*.



SLIKA 6.11. PRORAČUN PUTANJE OD IZVORNOG ČVORA *S* DO ODREDIŠNOG ČVORA *D*

U nastavku je dat primer proračuna putanje, koji se koristi kao deo procesa održavanja ruting tabele:

1. Sve stavke u okviru ruting tabele se uklanjaju.
2. Dodaju se nove stavke u okviru ruting tabele koje ukazuju na simetrične ili asimetrične veze ka susedima udaljenim do jednog preskoka ($h = 1$). Ovi čvorovi su deklarirani kao odredišni čvorovi unutar ruting tabele. Za svaki odredišni čvor unutar ruting tabele dodaje se novi zapis koji sadrži odredište i adresu sledećeg preskoka, pri čemu se IP adresa susednog čvora koristi za označavanje obe ove vrednosti, kao i ruting rastojanje (engl. *Route Distance, R_dist*) koje se definiše sa vrednošću 1.
3. Nove stavke se dodaju za čvorove koji se nalaze na udaljenosti većoj od jednog preskoka. Proces počinje sa čvorovima koji se nalaze na razdaljini od dva preskoka ($h = 2$). Potom se h vrednost uvećava za 1 i proces se nastavlja. Proces se završava kada više nema stavki za dodavanje u datoj iteraciji.

Za svaku stavku u okviru tabele topologije, ako adresa odredišta nije pronađena u registru ruting tabele i ako njen poslednji preskok pripada odredišnoj adresi zapisa sa razdaljinom h , sledeći zapis se dodaje u ruting tabelu:

- Odredišna adresa se postavlja na adresu odredišta iz tabele topologije.
 - Sledeći preskok se postavlja na vrednost sledećeg preskoka u zapisu rute čije odredište je isto kao i adresa poslednjeg preskoka.
 - Udaljenost se postavlja na vrednost $h+1$.
4. Nakon proračuna ruting tabele, sve stavke iz tabele topologije koje su korišćene za proračun ruta biće obrisane, čime se oslobađa deo iskorišćene memorije.

Ruting tabela se ponovo proračunava svaki put kada se otkrije promena u topologiji. Dakle, novi proračun ruting tabele vrši se kada postoji promena u:

- Skupu susednih čvorova, ili
- Skupu susednih čvorova udaljenih dva preskoka.

Tabela topologije ponovo se kreira kada se pojavi ili izgubi veza sa susednim čvorom, ili kada se deo topologije kreira ili ukloni.

7. SIMULACIJA

7.1. Metodologija istraživanja

U ovom poglavlju objašnjena je metodologija sprovođenja ovog istraživanja. Obrazložene su izabrane metode istraživanja i upoređene su sa drugim mogućim metodama. Takođe, obrazložena je opravdanost njihovog izbora.

7.1.1. Pretpostavke

Sledeće pretpostavke, koje su široko usvojene u literaturi [81, 82, 83], korišćene su i u ovom istraživanju:

- Svaki mobilni čvor ima dovoljno energetske resurse da funkcioniše tokom izvođenja celokupnog procesa simulacije. Ni u jednom trenutku mobilnom čvoru neće ponestati energije, niti će doći do njegovog otkaza usled nedostatka napajanja. Tokom celokupnog izvođenja eksperimenata bežični primopredajnici su aktivni u svim trenucima.
- Broj čvorova u datoj topologiji ostaje nepromenjen tokom izvođenja simulacije. Ne postoje čvorovi koji se pridružuju, niti čvorovi koji napuštaju mrežu tokom izvođenja simulacije. Ipak, i dalje se može javiti efekat podeljenosti mreže, tako da mreža možda neće biti povezana sve vreme. Međutim, ni u jednom trenutku čvor ne napušta niti se dodaje u simulacionu oblast. Na ovaj način omogućuje se proučavanje ponašanja predloženih algoritama pod istim uslovima.
- Različiti simultani prenos paketa putem mreže mogu ometati jedni druge, odnosno utiču jedni na druge ako se nađu u neposrednoj blizini. S druge strane, čvor će uvek uspešno dekodirati prenešene signale pod uslovom da je u prenosnom opsegu izvora i da nema smetnji u njihovom prenosu. Svi podaci koji postoje u mreži, potekli su iz date mreže.
- Svi mobilni čvorovi su homogeni, odnosno svi bežični čvorovi su opremljeni sa IEEE 802.11 primopredajnicima iste nominalne snage prenosa.
- Svi čvorovi učestvuju u formiranju jedinstvenog ruting protokola unutar mreže. Svaki čvor ponaosob sposoban je da prosleđuje pakete ka drugim čvorovima u mreži.
- Proces otkrivanja putanje može biti pokrenut sa bilo kog izvornog čvora koji ima pakete podataka koji trebaju biti prenešeni.

Dodatne pretpostavke biće navedene u narednim poglavljima kada se za to ukaže potreba.

7.1.2. Opravdanost metodologije istraživanja

Nakon donošenja odluke o domenu ove disertacije, a to je unapređenje postojećeg OLSR ruting protokola u MANET mrežama, još u ranim fazama ovog istraživanja donešena je odluka o metodologiji korišćenoj u cilju testiranja, merenja i procene mehanizama i tehnika koje će biti korišćene tokom izrade ove disertacije. U ovom poglavlju razmotrene su različite

metode istraživanja MANET mreža i objašnjen je izbor simulacije kao odgovarajućeg naučnog metoda za potrebe izrade ove disertacije. Opravdana je upotreba OPNET Modeler i ns-3 mrežnih simulatora, i date su informacije o postupcima koji su ispraćeni u cilju smanjenja mogućnosti za nastankom grešaka prilikom izvođenja simulacija.

Istraživanje funkcionalnosti MANET mreža može se izvršiti na jedan od tri uobičajena metodološka načina:

- simulacija,
- analitičko modelovanje, i
- realno okruženje.

Nakon pažljivog razmatranja, kao i kod većine drugih akademskih projekata, simulacija je izabrana kao pogodan metod proučavanja u ovom istraživanju, jer pruža okruženje sa specifičnim prednostima nad ostalim metodama [84]. Ovo uključuje:

- Simulacija omogućuje ponovljive scenarije i istovremeno istraživanje različitih metrika. Ovo omogućuje razvoj i modifikaciju ruting protokola u istovetnom scenariju, koji će pomoći u dubljem razumevanju uticaja pojedinih parametara na performanse i krajnje rezultate.
- Simulacija omogućuje izolaciju parametara. Ovo omogućuje proučavanje efekta pojedinačnog parametra, kao što je mobilnost, gustina čvorova, količina saobraćaja ili propusni opseg, dok svi ostali parametri ostaju nepromenjeni.
- Simulacija omogućuje širok spektar scenarija i mrežnih konfiguracija koje se procenjuju na razumnom nivou, u ograničenom vremenskom okviru i budžetu.

U momentu kada je ovaj istraživački rad započet, jedna od opcija za razmatranje je bila i analitičko modelovanje. U slučaju MANET mreža sa višestrukim preskocima, analitičko modelovanje se pokazalo kao grubo u svojoj prirodi, što ga čini nedovoljnim za korišćenje u ovom istraživanju sa razumnim stepenom tačnosti. Naime, u mobilnim mrežama brojni faktori su uključeni u razvoj analitičkog modela, pri čemu odnosi između ovih faktora još uvek nisu u potpunosti shvaćeni. Takvi faktori uključuju brzinu mobilnosti čvorova, opterećenost saobraćaja i veličinu mreže. Štaviše, precizni efekti svakog od ovih faktora na mrežne performanse nisu precizno definisani, što još više opravdava korišćenje metoda simulacije za istraživanje mobilnih mreža.

Poslednja alternativa modelu simulacije je korišćenje realnog bežičnog okruženja. Za potrebe ovog istraživanja potrebno je proučiti veći broj različitih MANET mreža. U slučaju korišćenja realnog okruženja, moguće postavke su ograničene na fizičke modele i dostupnost komponenti. Istina je da korišćenje fizičkog okruženja pruža realno posmatranje bilo koje izabrane tehnike. Međutim, troškovi i komplikacija u postavljanju realnog fizičkog okruženja smanjile bi stepen izvodljivosti ovog rada. Kao razuman kompromis između tačnih povratnih informacija iz realnog fizičkog modela i nesigurnog ishoda analitičkog modela, simulacija je izabrana kao pogodan metodološki model za izradu ove disertacije.

Izbor metodologije istraživanja je nedovoljan za započinjanje eksperimenata koji se sprovode u okviru ovog istraživanja. Još jedan izbor koji je morao biti napravljen je odabir simulatora koji će se koristiti za izvršavanje simulacija. Osnovni preduslov je mogućnost prilagođenja simulatora potrebama istraživanja i mogućnost primene predloženih mehanizama i tehnika. Za ove namene postoji širok spektar simulatora, među kojima su najpopularniji OPNET Modeler, ns-2, ns-3 i GloMoSim.

OPNET Modeler [85] je komercijalni mrežni simulator koji podržava različite tehnologije LAN i WAN mreža, i implementira više od 400 funkcija za modelovanje protokola, elemenata mreže i dinamičkog ponašanja mreže. Pripada klasi simulatora vođenih događajima u diskretnom vremenu, a implementacija simulatora se zasniva na objektno-orijentisanom programiranju kombinovanim sa analitičkim modelom. Izvorni kod programa nije javno dostupan. Raspoloživ je za rad u Windows i Linux okruženjima. Za potrebe simulacije MANET okruženja obezbeđena je podrška za AODV, DSR, GRP, OLSR i TORA ruting protokole.

Mrežni simulator verzije 2 (engl. *Network Simulator version 2, ns-2*) [86] je simulator vođen događajima u diskretnom vremenu. Razvijen je metodama objektno-orijentisanog programiranja na programskom jeziku C++ sa korisničkim interpreterom objektno-orijentisanih OTcl (engl. *Object-oriented Tool Command Language*) skriptova. Programiranje scenarija simulacije obavlja se pomoću OTcl-a, a efikasna obrada generisanih događaja omogućena je zahvaljujući C++ jezgru simulatora. Ns-2 je softver sa slobodnom distribucijom i slobodno dostupnim izvornim kodom. Razvijen je za Unix i Windows platforme. Vizuelni prikaz scenarija i toka simulacija omogućen je kroz mrežni animator (engl. *Network Animator, NAM*). Analiza rezultata može se obaviti pomoću različitih aplikacija koje omogućavaju grafički prikaz performansi mreže.

Ns-3 [87] je mrežni simulator sa otvorenim kodom (engl. *open source*) na čijem se razvoju i unapređenju i dalje aktivno radi. Ns-3 za cilj ima značajno proširenje funkcionalnosti postojećeg ns-2 simulatora. Razvijen i napisan u programskim jezicima C++ i Python sa mogućnošću pisanja novih skripti, a njegov kod je dostupan za Linux, Unix, OS X i Windows operativne sisteme. Model MANET mreža sastavni je deo ns-3 simulatora.

GloMoSim je mrežni simulator koji je namenski razvijen za potrebe simulacije skalabilnih bežičnih mreža [88]. Simulacije su vođene diskretnim događajima i mogu se izvoditi paralelno. Obezbeđuje podršku za AODV, DSR, Fisheye i druge MANET ruting protokole. Slobodno je dostupan akademskim institucijama koje imaju licencu za njegovo korišćenje, a komercijalno je dostupan kao QualNet softverski paket.

Među njima izabran je OPNET Modeler mrežni simulator kao alat za simulaciju i implementaciju predloženog OLSR-BF ruting protokola. OPNET je izabran, jer pored mogućnosti korišćenja postojećih modela i protokola, omogućuje kreiranje i simuliranje sopstvenih protokola. OPNET Modeler je mrežni simulator vođen događajima u diskretnom vremenu. OPNET obezbeđuje široku podršku za simulaciju TCP, ruting, multikast i drugih protokola putem ožičenih i bežičnih (lokalnih i satelitskih) mreža. OPNET simulator je dokazan alat za simulaciju korišćen u velikom broju prethodnih studija MANET mreža [16, 89, 90, 91].

OPNET Modeler je posebno koristan, jer već ima nekoliko realizovanih MANET ruting protokola koji se mogu koristiti i modifikovati prema potrebi. Tako je na osnovu postojećeg modela OLSR ruting protokola kreiran novopredloženi OLSR-BF ruting protokol za MANET mreže. OLSR je posebno interesantan, jer poseduje već gotove modele koji mogu lako biti modifikovani i iskorišćeni za implementaciju modifikovanih ruting protokola.

Procesu izrade izmena u OPNET simulatoru za prihvatanje novog predloženog ruting protokola posvećena je posebna pažnja u cilju garantovanja da će primenjeni algoritam funkcionisati onako kako je napravljen i da simulator neće ispoljiti neželjene efekte. To je

postignuto kroz temeljne provere korišćenih modela i algoritama koje OPNET simulator poseduje. Takođe, izvršeno je pažljivo parcijalno testiranje realizovanih funkcija. Osim toga, realne implementacije funkcija ruting protokola uključene su u simulacijama koje se sprovode tokom ovog istraživanja, kako bi se postigla aproksimacija što je moguće bliža stvarnom ponašanju sistema.

7.2. OPNET Modeler

Optimizovani mrežni tehnički alat (engl. *Optimized Network Engineering Tool, OPNET*) je originalno razvijen na Masačusetskom institutu za tehnologiju (engl. *Massachusetts Institute of Technology, MIT*), SAD, 1987. godine za modelovanje mreže i simulaciju različitih protokola. OPNET Modeler sastoji se od različitih integrisanih alata koji podržavaju modelovanje i simulaciju ožičenih, satelitskih, mobilnih i fiksni bežičnih mrežnih sistema [85].

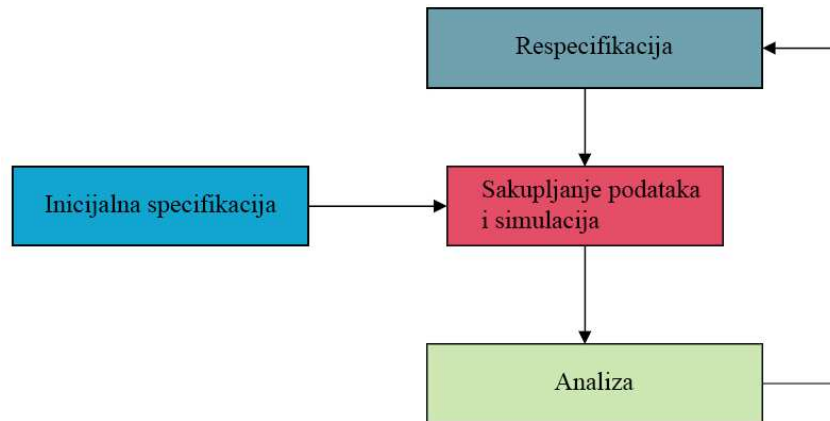
OPNET Modeler mrežni simulator jedan je od vodećih simulatora u svojoj oblasti, specijalizovan za istraživanje i razvoj mreža. OPNET omogućava da se osmisle i istraže komunikacione mreže, uređaji, protokoli i aplikacije sa velikom fleksibilnošću. OPNET poseduje grafički interfejs za izgradnju modela različitih mrežnih entiteta, od fizičkog sloja do aplikativnih procesa. Sve komponente su modelovane objektno-orijentisanim pristupom koji omogućuje intuitivno i jednostavno mapiranje realnih sistema. OPNET simulacija se zasniva na događajima, odnosno OPNET kreira pakete za prenos podataka umesto da simulira tokove bita. OPNET koristi „vremenske pečate događaja“ koji povezuju svaki simulirani događaj sa određenim vremenom simulacije. Stoga, simultani događaji koji se dešavaju na različitim čvorovima mogu biti precizno modelovani i evidentirani. OPNET Modeler sadrži biblioteku modela koja sadrži standardne mrežne modele i protokole (npr. ATM, MANET, TCP/IP, RIP, OSPF, i druge). Model mobilnih ad hoc bežičnih mreža integrisan je u osnovnom OPNET Modeler paketu u cilju simuliranja MANET ruting protokola AODV, DSR, GRP, OLSR i TORA.

7.2.1. Arhitektura

OPNET Modeler predstavlja sveobuhvatno razvojno okruženje za modelovanje i procenu performansi komunikacionih mreža i distribuiranih sistema. OPNET Modeler paket sastoji se od niza alata, pri čemu je svaki fokusiran na pojedine aspekte faza modelovanja [89]. Ovi alati svrstavaju se u tri glavne kategorije, koje odgovaraju trima fazama modelovanja i simulacije projekata:

- Specifikacija,
- Sakupljanje podataka i simulacija, i
- Analiza.

Ove faze se izvode u odgovarajućem nizu, pri čemu se uglavnom formira ciklus sa povratkom u fazu specifikacije nakon faze analize. Faza specifikacije je zapravo podeljena na dva dela: inicijalna specifikacija i respecifikacija, pri čemu samo faza respecifikacije pripada ciklusu (slika 7.1).



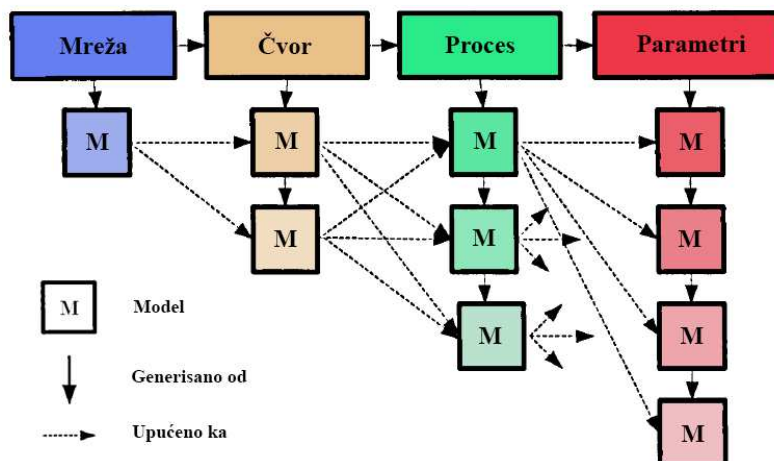
SLIKA 7.1. PROCES SIMULACIJE PROJEKTA KOD OPNET SIMULATORA

7.2.2. Hijerarhijsko modelovanje

OPNET Modeler ima četiri alata (editora) pomoću kojih predstavlja sistem koji se modeluje:

- Mrežu,
- Čvor,
- Proces, i
- Parametre.

Ovi alati su organizovani u hijerarhijsku organizaciju (slika 7.2) koja podržava koncept ponovne upotrebe modela na različitom mrežnom nivou. Modeli razvijeni u jednom sloju mogu se koristiti od strane drugog modela na višem mrežnom sloju [89].



SLIKA 7.2. HIJERARHIJSKA ORGANIZACIJA KOD OPNET SIMULATORA

Mrežni editor se koristi za kreiranje fizičke topologije komunikacione mreže. Njime se definiše položaj i interkonekcija komunikacionih entiteta, koji se sastoje od čvorova i veza. Konkretno mogućnosti svakog čvora su realizovane u zasebnom modelu sa skupom atributa koji je priključen svakom modelu čvora.

Modeli čvorova su izraženi kao povezani moduli dve različite kategorije. Prvi skup modula ima predefinisane karakteristike i skup ugrađenih parametara, kao što su generatori paketa i radio prijemnici. Drugi skupovi su programabilni moduli, koji se nazivaju procesori i redovi.

Svaki čvor je definisan kao blok struktuiranog dijagrama protoka podataka, pri čemu su njegove funkcionalnosti definisane modelom procesa. Paketi se prenose između različitih modula putem fiksnih žica korišćenjem toka paketa i numeričkih signala. Čvor može biti fiksni ili mobilan. OPNET Modeler podržava sledeće modele čvorova u okviru MANET mreže:

- MANET stanice: Ovi modeli čvorova mogu se koristiti za generisanje sirovih IP paketa preko bežične mreže. Oni čvorovi opremljeni su sa IEEE bežičnim primopredajnicima koji im omogućuju primanje i slanje paketa. MANET stanice mogu se konfigurisati da rade kao mobini čvorovi unutar određenog MANET ruting protokola.
- Bežične mrežne radne stanice i serveri: Ovi modeli čvorova koriste se za generisanje aplikativnog saobraćaja, kao što je FTP, E-mail ili HTTP, putem IP paketa podataka preko bežične mreže. Ovi čvorovi mogu se konfigurisati da rade kao mobini čvorovi unutar određenog MANET ruting protokola.
- Profili: Opisuju aktivnosti korisnika u pogledu aplikacija koje se koriste. Moguće je koristiti više različitih profila u datoj mreži. Profili mogu predstavljati različite grupe korisnika kako bi oslikali tipične aplikacije koje se koriste za svaku grupu korisnika.
- Aplikacije: Profil je konstruisan korišćenjem različitih definicija za svaku korišćenu aplikaciju. Za svaku aplikaciju moguće je definisati različite parametre korišćenja, kao što su vreme početka, vreme trajanja i ponovljivost izvođenja. Na taj način, moguće je korišćenje različitih aplikacija sa različitim parametrima upotrebe unutar jedne mreže.
- Grupa prijemnika (engl. *Receiver Group, Rxgroup*): Čvor koji označava grupu prijemnika se koristi za izračunavanje skupa mogućih prijemnika sa kojima određeni čvor može komunicirati. Ovaj uslužni čvor može znatno ubrzati simulaciju eliminisanjem prijemnika koji se ne podudaraju.
- Zadatak (engl. *task*): Prilagođene, nestandardne aplikacije se konfigurišu pomoću čvora koji označava konfiguraciju zadatka.
- Mobilnost: Ovi čvorovi se koriste za definisanje profila mobilnosti pojedinih čvorova. Čvorovi koji definišu mobilnost kontrolišu kretanje mobilnih čvorova unutar MANET mreže na osnovu zadatih parametara.

Modeli procesa se koriste da opišu tok logike i ponašanja modela procesa i redova. Komunikacija među različitim procesima ostvaruje se sistemskim prekidima (engl. *interrupt*). Procesni modeli predstavljeni su u Proto-C jeziku, koji se sastoji od dijagrama prelaza stanja, biblioteke kernel procedura i standardnog C programskog jezika. Editor procesa koristi pristup dijagrama prelaza stanja kako bi podržao bilo koji tipa protokola, resursa, aplikacije, algoritma ili polise svrstavanja paketa u red čekanja. Stanja i prelazi grafički definišu napredovanje procesa kao odgovor na različite događaje. Unutar svakog stanja, generalna logika može se odrediti korišćenjem biblioteke unapred definisanih funkcija i standardnog C programskog jezika. Roditelj proces može stvoriti nove procese koji se nazivaju decom procesima, u cilju obavljanja podzadataka.

7.3. Metrike

Postoje različite vrste metrika za vrednovanje performansi različitih ruting protokola. Različite metrike ukazuju na različite karakteristike mrežnih performansi unutar jedne mreže.

U ovom radu razmotrene su najznačajnije metrike za poređenje ukupnih mrežnih performansi. Metrike su izabrane na osnovu iskustava brojnih sličnih studija koje se mogu naći u literaturi [58, 91]. Ove metrike su propusni opseg, kašnjenje, opterećenost mreže kontrolnim paketima, količina generisanih pozdravnih poruka i broj proračuna ruting tabele. Ovi parametri su važni za razmatranje procene efikasnosti ruting protokola u MANET mreži. Kako bi se proverila efikasnost ruting protokola u pronalaženju putanje prema odredištu, posmatra se količina generisanih kontrolnih poruka kroz mrežu. Ako ruting protokol pritom ima veliko kašnjenje, vrlo je verovatno da taj ruting protokol nije efikasan kao ruting protokoli koji imaju nisku vrednost kašnjenja. Slično važi i za veličinu propusnog opsega, koja predstavlja količinu uspešno isporučenih paketa u jedinici vremena. Ruting protokol koji ostvaruje visoku vrednost propusnog opsega smatra se efikasnijim od ruting protokola sa niskom vrednošću propusnog opsega. Kvalitativno poređenje različitih metrika ima veliki uticaj na izbor efikasnog ruting protokola u MANET mreži i direktno se odnosi na upotrebnu vrednost predloženog MANET ruting protokola.

U ovom simulacionom okruženju svaka od navedenih metrika je izračunata kao prosečna vrednost svih čvorova u mreži.

7.3.1. Propusni opseg (propusnost)

Propusni opseg (engl. *throughput*) se definiše kao ukupna količina podataka koja dostiže prijemnik od strane pošiljaoca [92] u jedinici vremena. Veličina propusnog opsega se predstavlja u broju prenešenih bita u sekundi [bit/s]. Brojni faktori unutar mreže utiču na veličinu propusnog opsega, kao što su promene u topologiji mreže, nepouzdana komunikacija između čvorova, ograničen kapacitet veza i ograničeni energetske resursi. Visoka propusna moć je veoma poželjna u svakoj mreži. Veličina propusnog opsega se može predstaviti sledećim izrazom:

$$\text{Propusni opseg} = \frac{\text{Broj isporučenih paketa} * \text{Veličina paketa} * 8}{\text{Trajanje simulacije}}$$

Istraživanje veličine propusnog opsega, naročito kada se razmatra u različitim mrežnim scenarijima, pomaže da se utvrdi koliko pojedini ruting protokoli dozvoljavaju aplikacijama da optimizuju korišćenje raspoloživog propusnog opsega.

7.3.2. Kašnjenje

Ukupno kašnjenje (engl. *delay*) predstavlja vreme od momenta generisanja paketa na izvoru do prijema paketa na odredištu [91]. Dakle, ovo je vreme koje je potrebno da paket pređe preko mreže. Određene aplikacije, npr. prenos glasa, osetljivije su od drugih na kašnjenje paketa, te stoga zahtevaju nisku prosečnu vrednost kašnjenja u mreži. Usled slabe snage signala čvorova, čestih kreiranja i raskidanja veza, i mobilnosti čvorova povećava se ukupno kašnjenje unutar MANET mreže.

Postoje četiri tipa kašnjenja koja doprinose kašnjenju:

- kašnjenje u prenosu (engl. *transmission delay*) – vreme potrebno predajniku da preda sve bitove paketa podataka.

- propagaciono kašnjenje (engl. *propagation delay*) – vreme koje je potrebno da se prenese jedan bit od izvora do odredišta.
- kašnjenje usled procesiranja (engl. *processing delay*) – vreme koje je potrebno da se na izvornoj strani procesira paket pre predaje, kao i na bilo kom međuvorstu koji se nalazi na putu prenosa podataka, plus dodatno vreme koje je na odredištu potrebno za procesiranje poruke pre nego što se preda aplikaciji.
- kašnjenje zbog čekanja u redovima (engl. *queuing delay*) – kašnjenje koje se javlja zbog čekanja u redovima čekanja u bilo kom čvoru na prenosnom putu.

Matematički model ukupnog mrežnog kašnjenja može se predstaviti sledećim izrazom, pri čemu kašnjenje zbog čekanja u redovima nije uključeno, jer mrežno kašnjenje nije povezano sa njim:

$$T_{ukup} = N * (T_{pren} + T_{prop} + T_{proc})$$

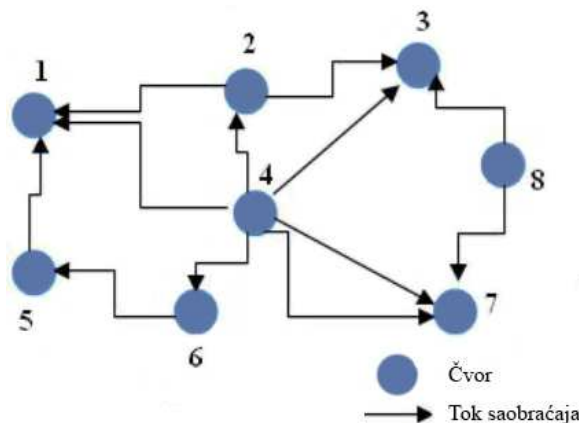
gde su:

- N – skalarna veličina (broj čvorova)
- T_{ukup} – ukupno kašnjenje
- T_{pren} – kašnjenje u prenosu
- T_{prop} – propagaciono kašnjenje
- T_{proc} – kašnjenje usled procesiranja

7.3.3. Opterećenost mreže kontrolnim saobraćajem

Opterećenost mreže kontrolnim saobraćajem (engl. *network load*) predstavlja ukupan protok podataka (opterećenje) dostavljen sa viših slojeva na MAC sloj, koji su prihvaćeni i koji čekaju na prenos na svim bežičnim čvorovima unutar mreže [93]. Ova vrednost ne uključuje saobraćaj podataka sa viših slojeva koji je odbačen bez čekanja u redovima, zbog popunjenosti redova čekanja ili usled prevelikih paketa podataka. Opterećenost mreže povećava se sa količinom saobraćaja koja postoji na mreži. Povećanje efikasnosti određene mreže ostvaruje se smanjenjem njene opterećenosti.

Visoka opterećenost mreže utiče na performanse MANET ruting protokola tako što usporava isporuku paketa [94] i dovodi do povećanja broja sudara kontrolnih paketa. Model opterećenja mreže prikazan je na slici 7.3.



SLIKA 7.3. OPTEREĆENOST MREŽE KONTROLNIM SAOBRAĆAJEM

Opterećenost mreže kontrolnim saobraćajem može se predstaviti sledećim matematičkim modelom:

$$NL = \frac{R_d}{BS - BU - (R_d - S_d)}$$

$$NL = \frac{\int_{t_0}^t f_{R_d(s)} ds}{BS - BU(t_0) - \int_{t_0}^t f_{R_d(s)} ds - \int_{t_0}^t g_{S_d(s)} ds}$$

gde su:

- NL – opterećenost mreže
- R_d – primljeni podaci
- S_d – poslani podaci
- $BU(t_0)$ – bafer nedostupnosti
- BS – veličina bafera
- $f_{R_d(t)}$ – vreme prijema pri učestanosti t
- $g_{S_d(t)}$ – vreme slanja pri učestanosti t
- $\int_{t_0}^t f_{R_d(s)} ds$ – primljeni podaci u intervalu $[t_0, t]$

Ruting protokoli koriste kontrolne pakete za uspostavljanje ruta putem kojih se prenose paketi podataka. Kontrolni paketi su odvojeni od paketa podataka, ali dele isti komunikacioni kanal. Zbog ograničenog kapaciteta bežičnih kanala u mobilnim ad hoc mrežama veliki broj kontrolnih paketa može dovesti do zagušenja unutar mreže i loših mrežnih performansi. Količina propusnog opsega konzumirana od strane svih generisanih kontrolnih paketa datog ruting protokola pomaže da se utvrdi njegova sklabilnost. Količina generisanih kontrolnih paketa je poznata kao unutrašnja efikasnosti određenog ruting protokola. Niža vrednost količine generisanih kontrolnih paketa veoma je tražena optimizacija u okviru MANET mreža.

Mobilni čvorovi su suočeni sa ograničenim izvorom napajanja, te je stoga ušteda energije važan faktor koji treba razmotriti u implementaciji MANET mreža. Osim toga, ograničenost prostiranja radio signala, višestruko korišćenje istovetnog radio kanala, i veličina mreže su faktori koje takođe treba uzeti u obzir. Ovi faktori ograničavaju čvorove da direktno komuniciraju između izvora i odredišta. Sa porastom broja čvorova u mreži komunikacija između izvora i odredišta sve više se oslanja na središnje čvorove. Većina ruting protokola oslanja se na susedne čvorove za usmeravanje saobraćaja, te stoga povećanje broja suseda izaziva još veću stopu generisanja kontrolnog ruting saobraćaja u mreži.

Borba za prenosne slotove između različitih čvorova takođe postaje izraženija sa porastom veličine mreže. Učestalost emisija je povećana zbog čestih prekida veza usled mobilnosti čvorova. Način na koji se ruting protokol suočavanju sa ovim izazovima usled zagušenja mreže i niskog kapaciteta linkova je važan pokazatelj efikasnosti datog ruting protokola.

7.4. Simulaciono okruženje

Cilj eksperimenata je merenje mrežnih performansi predloženog OLSR-BF ruting protokola u odnosu na postojeće AODV, DSDV i OLSR ruting protokole u različitim mrežnim topologijama i usled različitih uslova unutar mreže. Istraživanje je sprovedeno pomoću mrežnih simulatora diskretnih događaja OPNET Modeler [85] i ns-3 [87]. Simulacija je fokusirana na performanse MANET ruting protokola, pri čemu su beležene vrednosti propusnog opsega, kašnjenja i opterećenosti mreže kontrolnim saobraćajem. Za te potrebe kreirana su dva simulaciona scenarija sa različitim brojem mobilnih čvorova. Čvorovi su nasumično postavljeni u okruženju površine $1000 \times 1000 \text{ m}^2$. U prvom scenariju MANET mreža se sastoji od 20 mobilnih čvorova, dok je u drugom scenariju broj mobilnih čvorova povećan na 50. Ove vrednosti veličine mreže u kombinaciji sa količinom generisanog saobraćaja i uz kontrolisanje broja izvornih čvorova unutar MANET mreže osiguravaju testiranje pri različitim mrežnim opterećenjima.

Mrežni simulator OPNET Modeler poseduje predefinisane modele tradicionalnih AODV i OLSR ruting protokola. Na osnovu postojećeg modela OLSR ruting protokola kreiran je novi model za simulaciju OLSR-BF ruting protokola. U periodu kada je ova studija izvedena simulacioni model tradicionalnog DSDV ruting protokol nije postojao u okviru OPNET Modeler mrežnog simulatora. Stoga je za potrebe simulacije DSDV ruting protokola korišćen ns-3 mrežni simulator. Rezultati eksperimenata sprovedeni nad DSDV ruting protokolom u okviru ns-3 mrežnog simulatora unešeni su u OPNET Modeler i upoređeni su sa rezultatima eksperimenata sprovedenim nad ostalim ruting protokolima.

Mrežna topologija zasnovana je na upotrebi bežičnih čvorova. Svaki čvor u mreži podešen je da izvršava istovetni ruting protokol. U svakom scenariju proverava se ponašanje AODV, OLSR i OLSR-BF ruting protokola sa podrazumevanim parametrima u okviru OPNET Modeler mrežnog simulatora za svaki od protokola ponaosob (tabele 7.3 i 7.4). Istovetno se odnosi i na DSDV ruting protokol unutar ns-3 mrežnog simulatora (tabela 7.5).

Svaki scenario je izvođen u trajanju od 900 sekundi. Rezultati simulacija uporedno su prikazani na priloženim graphicima. Glavni cilj ovih simulacija je modelovanje ponašanja MANET ruting protokola. Za analizu različitih ruting protokola generisan je konstantan FTP saobraćaj unutar mreže.

Model mobilnosti i bežični mrežni parametri su identični za sve čvorove u svakom scenariju. Za model mobilnosti koristi se model mobilnosti slučajne putanje (engl. *Random Waypoint Mobility Model*, *RWP*) definisan u objektu konfiguracije mobilnosti mreže. Svi čvorovi nasumično se kreću unutar definisanog bežičnog domena. Brzina svakog mobilnog čvora definisana je uniformnom raspodelom između 0 i 10 m/s.

Objekat konfiguracije grupe prijemnika (engl. *Receiver Group*, *Rxgroup*) se koristi za ograničenje mogućeg skupa prijemnika na osnovu udaljenosti između čvorova. Prag rastojanja je postavljen na vrednost od 250 metara. Autor u [46] došao je do zaključka da je najčešća dužina puteva za slične oblasti dugačka 4 preskoka. Na prenosnom opsegu od 250 metara, minimalno rastojanje potrebno da pokrije ovaj broj preskoka je 1000 metara, pa je izabrana mreža oblast od $1000 \times 1000 \text{ m}^2$.

Ove vrednosti su izabrane na osnovu brojnih primera iz literature u kojima su vršena merenja performansi MANET ruting protokola [58, 95, 96], i predstavljeni su u tabeli 7.1.

TABELA 7.1. PARAMETRI SIMULACIJE

Parametri simulacije	Vrednost
Simulator	OPNET Modeler verzije 14.5, Ns-3 verzije 3.17
Veličina topologije	1000x1000 m ²
Veličina mreže	20, 50 čvorova
Model mobilnosti	Slučajna putanja (RWP)
Brzina mobilnosti čvorova	0 – 10 m/s
Tip prenošenog saobraćaja	FTP
Dužina trajanja simulacije	900 sekundi
Adresiranje	IPv4
Bežični standard	IEEE 802.11b
Brzina prenošenja podataka	11 Mbit/s
Transmisioni opseg	250 m
Ruting protokol	AODV, DSDV, OLSR, OLSR-BF

Parametri bežične mreže su podešeni sa podrazumevanim vrednostima dostupnim u OPNET Modeler simulatoru (tabela 7.2), i istovetni su za sve čvorove tokom izvođenja svih eksperimenata u oba scenarija. Slično važi i za DSDV ruting protokol uz određene promene u konfiguraciji ns-3 mrežnog simulatora kako bi vrednosti svih parametara bile izjednačene sa vrednostima definisanim u okviru OPNET Modeler simulatora. Svi čvorovi su opremljeni sa primopredajnicima koji koriste IEEE 802.11b standard u komunikaciji preko bežičnih kanala protoka od 11 Mbit/s.

TABELA 7.2. PARAMETRI BEŽIČNE MREŽE

Parametar	Vrednost
Fizičke karakteristike	IEEE 802.11b (HR/DSSS PHY)
Brzina prenošenja podataka	11 Mbit/s
Limit kratkih ponovnih pokušaja	7
Limit dugih ponovnih pokušaja	4
Maksimalni životni vek prijema	0.5 sekundi
Veličina bafera	256000 bita
Mogućnost rominga	Nije omogućena

TABELA 7.3. PARAMETRI AODV RUTING PROTOKOLA

Parametar	Vrednost
Interval slanja pozdravnih poruka	Uniformna (1, 1.1) sekundi
Dozvoljen gubitak pozdravnih poruka	2 paketa

Maksimalni broj preskoka	35
Vreme kretanja paketa na čvoru	0.04 sekundi
Granica u generisanju ruting grešaka na čvoru	10 paketa/sekundi
Bafer kratkih prekida	2
TTL inkrement	2
TTL granica	7
Lokalna adresa TTL	2
Veličina reda za čekanje paketa	Beskrajna
Lokalna popravka	Omogućena
Adresiranje	IPv4

TABELA 7.4. PARAMETRI OLSR I OLSR-BF RUTING PROTOKOLA

Parametar	Vrednost
Spremnost	Podrazumevana
Interval slanja pozdravnih poruka	2 sekunde
Interval slanja TC poruka	5 sekundi
Vreme držanja susednih veza	6 sekundi
Vreme držanja topoloških informacija	15 sekundi
Vreme držanja dupliciranih poruka	30 sekundi
Adresiranje	IPv4

TABELA 7.5. PARAMETRI DSDV RUTING PROTOKOLA

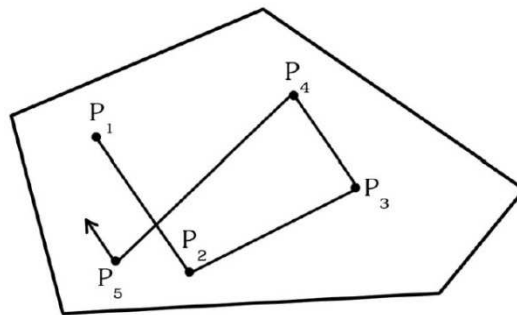
Parametar	Vrednost
Interval slanja periodičnih ispravki	15 sekundi
Obračunavanje intervala opterećenja	Omogućeno
Interval obračunavanja	6 sekundi
Faktor opterećenja	0.875
Baferovanje	Omogućeno
Maksimalna dužina reda čekanja	100 paketa
Maksimalno vreme čekanja	30 sekundi
Maksimalna dužina reda čekanja po odredištu	5 paketa
Agregacija ruta	Nije omogućena
Adresiranje	IPv4

Prostor mogućih vrednosti parametara za izvođenje simulacija teoretski je neograničen. Jedina ograničenja odnose se na prostor i vreme, kao i na proračunsku snagu računara na kome se izvode simulacije.

7.4.1. Model mobilnosti čvorova

Simulacija MANET mreža zahteva temeljnu pokrivenost svih aspekata korišćenog mrežnog protokola. Da bi se simulirala mobilna mreža mora se razmotriti model mobilnosti čvorova. Tokom istraživanja računarskih mreža korišćeni su brojni predloženi modeli mobilnosti [97]. Model mobilnosti slučajne putanje (engl. *Random Waypoint Mobility Model*, RWP) jedan je od najpopularnijih modela mobilnosti korišćen u istraživanju MANET mreža [98].

Kao što se sa slike 7.4 može uočiti, model počinje definisanjem mrežne topologije kao skupa čvorova koji su nasumično postavljeni u zatvorenom simulacionom prostoru, koji je takođe poznat kao oblast simulacije. Posle toga, svaki čvor nasumično bira destinaciju unutar simulacione oblasti i putuje prema njoj sa određenom brzinom. Kada stigne do odredišta, čvor pauzira tokom unapred definisanog vremenskog intervala, pre nego što odabere novu destinaciju. Proces se ponavlja do kraja izvođenja simulacije.



SLIKA 7.4. PRIMER MODELA MOBILNOSTI SLUČAJNE PUTANJE ČVORA P

Brzina svakog čvora je navedena u skladu sa ravnomerno raspoređenim vrednostima između 0 i V_{max} , gde je V_{max} maksimalna brzina čvora. Dužina pauze i V_{max} su konstantne vrednosti i unose se u simulator. U početnoj upotrebi modela mobilnosti slučajne putanje [99], povećanje mobilnosti je simulirano povećanjem parametra maksimalne brzine ili smanjenjem perioda trajanja pauze.

Ostali modeli mobilnosti predloženi za istraživanje u okviru MANET mreža uključuju varijaciju RWP modela mobilnosti pod nazivom slučajna putanja na granici (engl. *Random Waypoint on the Border*, RWPB) [100]. Kod ovog modela, inicijalna raspodela čvorova je blizu granica simulacione oblasti. Drugi, često korišćeni model je Markovijanov model mobilnosti putanje (engl. *Markovian Waypoint Model*, MWP) [101]. MWP dodaje ograničenje u pogledu sledećeg odredišta čvora u zavisnosti od njegove trenutne pozicije.

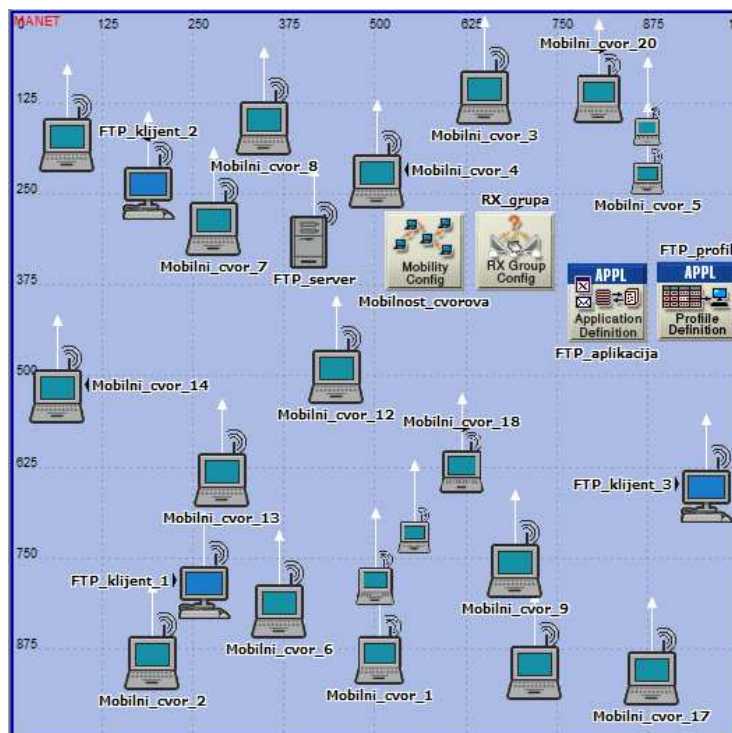
U ovom radu, RWP je izabran kao model mobilnosti čvorova iz nekoliko razloga. Pre svega ovaj rad ima za cilj da prouči učinak predloženog OLSR-BF rutinog protokola pod uticajem ograničenog skupa parametara kako bi se omogućio prihvatljiv stepen kontrole nad eksperimentima, pri čemu se izbegavaju eventualni neželjeni sporedni efekti koji mogu nastati usled korišćenja kompleksnih modela mobilnosti čvorova. Konačno, sve do momenta kada je ovo istraživanje započeto, nije postojao opšteprihvaćeni model mobilnosti predložen za simuliranje realnih mobilnih ad hoc mreža.

8. REZULTATI SIMULACIJA

Za potrebe ovog istraživanja u okviru OPNET Modeler i ns-3 mrežnih simulatora kreirana su dva scenarija MANET mreža sa različitim brojem mobilnih čvorova.

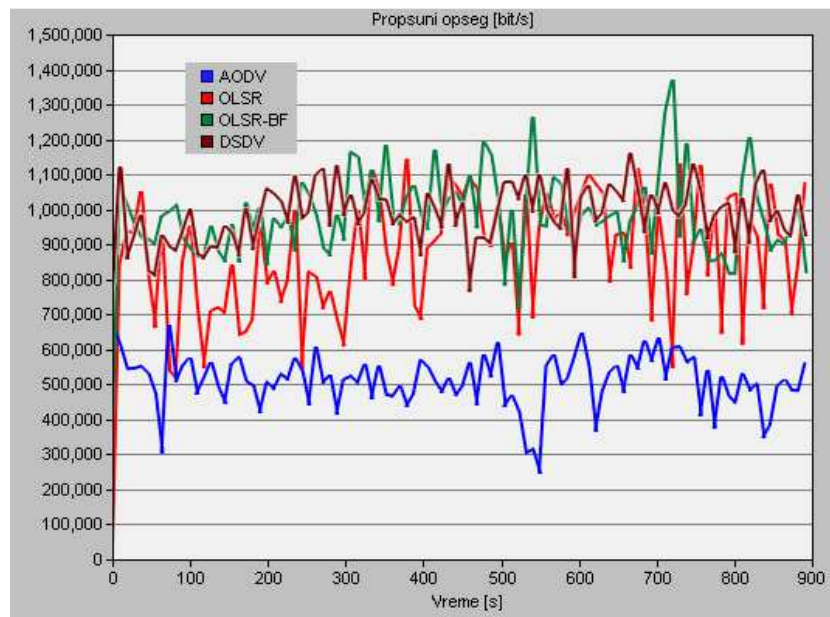
8.1. Mreža od 20 mobilnih čvorova

U ovom scenariju koristi se 20 mobilnih čvorova i jedan FTP server (slika 8.1). U mreži se koristi IPv4 adresiranje. TCP performanse MANET mreže analizirane su generisanjem protokola za prenos datoteka (engl. *File Transfer Protocol*, *FTP*). Model mobilnosti svakog čvora je izabran kao RWP, unutar mreže veličine 1000x1000 m². Na svim mobilnim čvorovima postavlja se jedinstveni ruting protokol, a istovetni eksperimenti su sprovedeni za AODV, DSDV, OLSR i OLSB-BF ruting protokole, pri čemu su merene sledeće metrike: propusni opseg, kašnjenje i opterećenost mreže kontrolnim saobraćajem. Specifično za OLSR i OLSR-BF ruting protokole merene su još i vrednosti količine generisanih pozdravnih poruka i broja proračuna ruting tabele.



SLIKA 8.1. MREŽNA TOPLOGIJA SA 20 MOBILNIH ČVOROVA

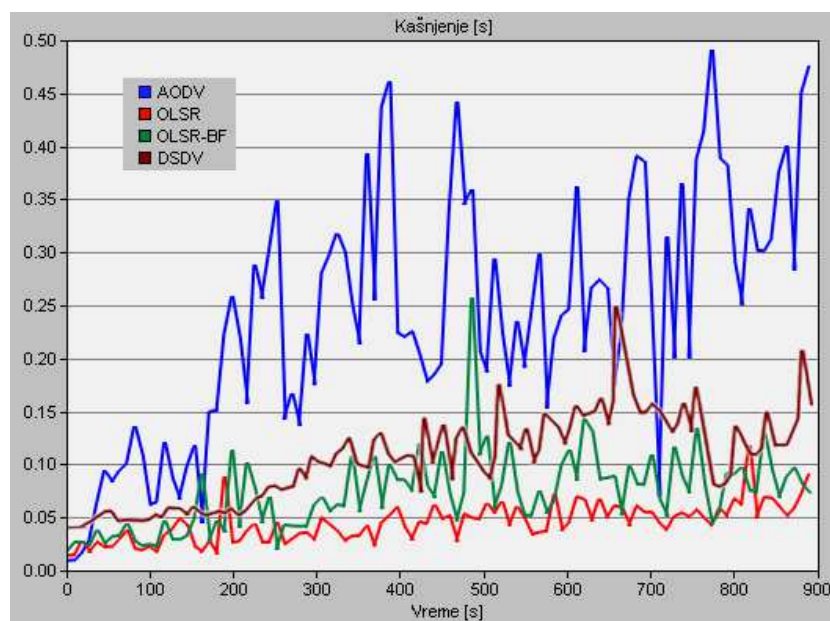
8.1.1. Propusni opseg



SLIKA 8.2. PROPUSNI OPSEG [BIT/S]

Slika 8.2 poredi vrednosti propusnog opsega za AODV, DSDV, OLSR i OLSR-BF ruting protokole u mreži sa 20 mobilnih čvorova. Uočljivo je da OLSR-BF pokazuje povećanu propusnu moć u odnosu na originalni OLSR ruting protokol, bez obzira na činjenicu što istovremeno ostvaruje i najvišu vrednost opterećenja mreže kontrolnim saobraćajem koja je zabeležena i predstavljena u poglavlju 8.1.3. DSDV takođe ostvaruje više vrednosti propusnog opsega u odnosu na OLSR, ali znatno manje nego li OLSR-BF. Takođe, evidentno je da reaktivni ruting protokol AODV ostvaruje najslabije rezultate propusnog opsega.

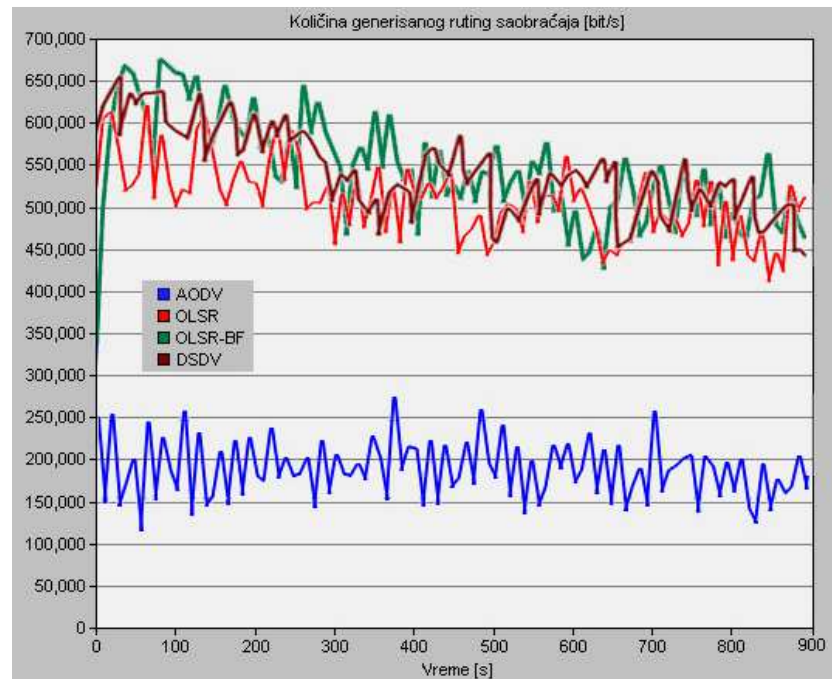
8.1.2. Kašnjenje



SLIKA 8.3. KAŠNjenje [S]

Slika 8.3 prikazuje kašnjenje za AODV, DSDV, OLSR i OLSR-BF ruting protokole u mreži sa 20 mobilnih čvorova. Ova metrika predstavlja vreme između generisanja paketa na izvornom čvoru i njegovog prijema na odredišnom čvoru. Evidentno je da reaktivni AODV ruting protokol ostvaruje znatno višu vrednost kašnjenja u odnosu na proaktivne ruting protokole DSDV, OLSR i OLSR-BF. Razlog niske vrednosti kašnjenja kod proaktivnih ruting protokola je poznavanje puta do odredišnog čvora pre zahteva za prenosom paketa.

8.1.3. Opterećenost mreže kontrolnim saobraćajem

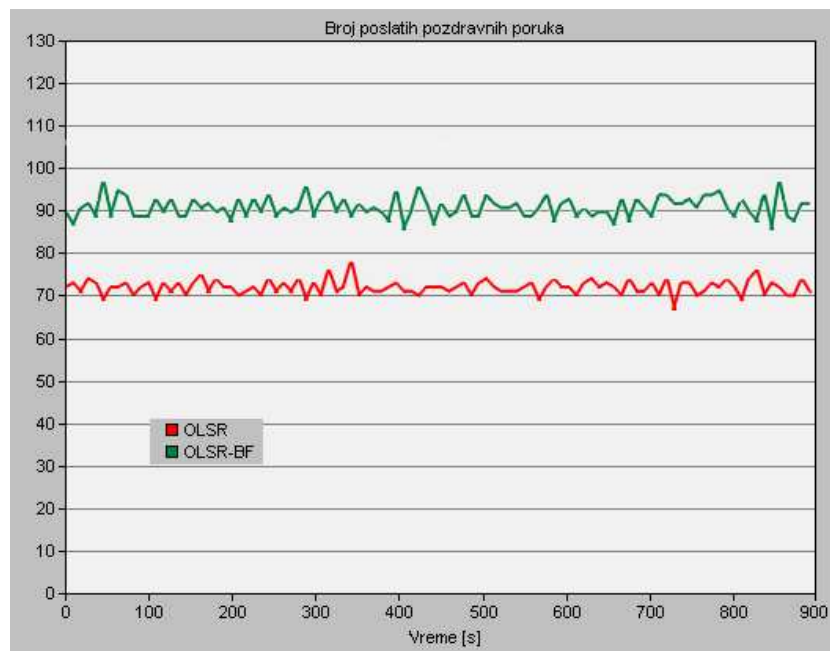


SLIKA 8.4. OPTEREĆENOST MREŽE KONTROLNIM SAOBRAĆAJEM [BIT/S]

Slika 8.4 pokazuje količinu ruting saobraćaja poslatog u mreži sa 20 mobilnih čvorova za sva četiri ruting protokola. Ova vrednost uključuje poslate pozdravne poruke, kao i poslate i prosleđene TC poruke. Evidentna je značajna razlika između proaktivnog i reaktivnog ruting pristupa. Iz rezultata eksperimenata jasno je da proaktivni ruting protokoli DSDV, OLSR i OLSR-BF grade kompletne ruting tabele odmah na početku simulacije. S druge strane, reaktivni ruting protokol AODV putanje do odredišta proračunava kada postoji zahtev za prenos paketa podataka. OLSR-BF generiše najviše vrednosti opterećenja mreže usled čestog ažuriranja ruting tabele, i odabira MPR čvorova prilikom svake promene mrežne topologije. DSDV ima sličnu karakteristiku usled činjenice da se poruke ažuriranja ruting tabele propagiraju kroz celu mrežu.

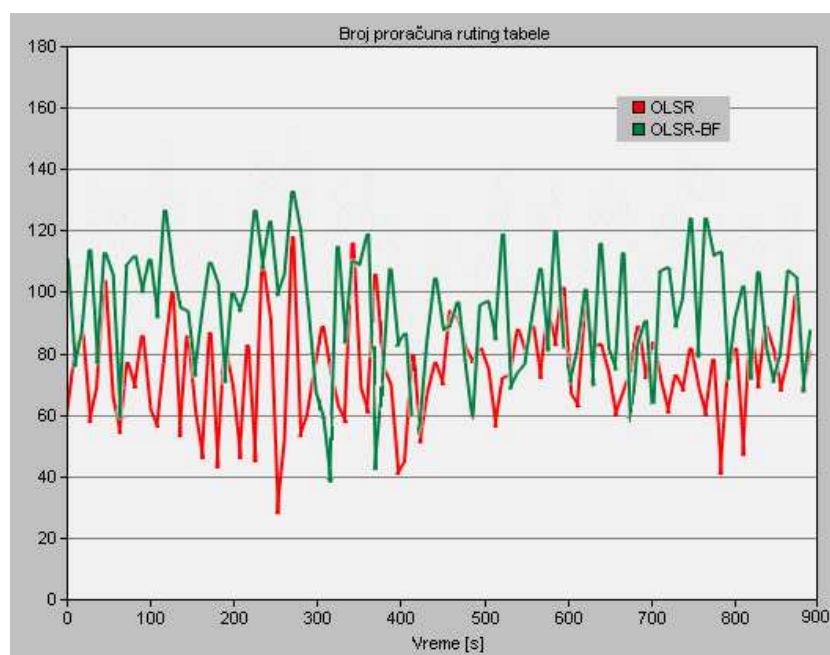
8.1.4. Broj poslatih pozdravnih poruka

Broj poslatih pozdravnih poruka od strane 20 mobilnih čvorova kod OLSR i OLSR-BF ruting protokola prikazan je na slici 8.5. Evidentno je da je broj poslatih pozdravnih poruka veći kod modifikovanog OLSR-BF ruting protokola za oko 20%.



SLIKA 8.5. BROJ POSLATIH POZDRAVNIH PORUKA

8.1.5. Broj proračuna ruting tabele

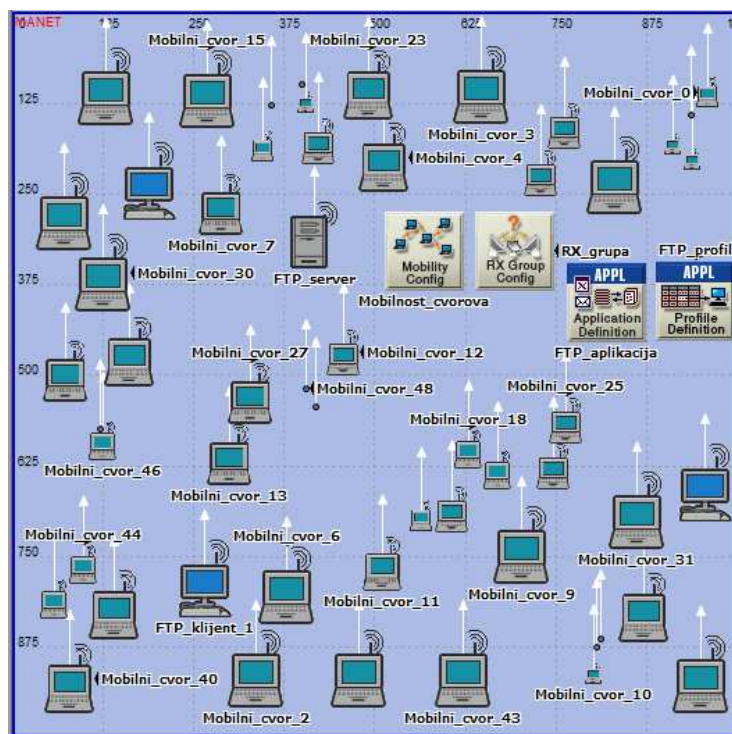


SLIKA 8.6. BROJ PRORAČUNA RUTING TABELE

Slika 8.6 pokazuje ukupan broj proračuna ruting tabele tokom simulacije mreže sa 20 mobilnih čvorova. Proračun ruting tabele kod proaktivnih ruting protokola OLSR i OLSR-BF dešava se svaki put kada se na čvoru detektuje promena susedstva ili topološka promena. Provera stanja susedstva se dešava tokom obrade periodičnih pozdravnih i TC poruka. Veliki broj proračuna ruting tabele uzrokovan je prvenstveno usled mobilnosti čvorova unutar mreže.

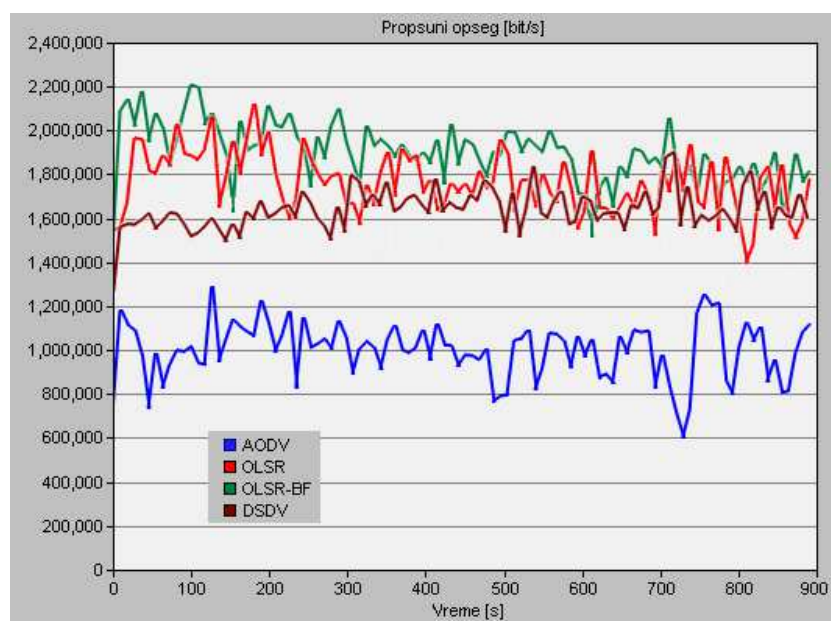
8.2. Mreža od 50 mobilnih čvorova

U drugom scenariju povećan je broj mobilnih čvorova na 50 (slika 8.7), dok svi ostali parametri u okviru MANET mreže ostaju istovetni. Razlog povećanja broja mobilnih čvorova je simulacija mreže sa većom gustinom mobilnih čvorova na istoj površini. U ovom scenariju ponavljaju se eksperimenti za svaki od rutinog protokola u odnosu na istovetne parametre.



SLIKA 8.7. MREŽNA TOPLOGIJA SA 50 MOBILNIH ČVOROVA

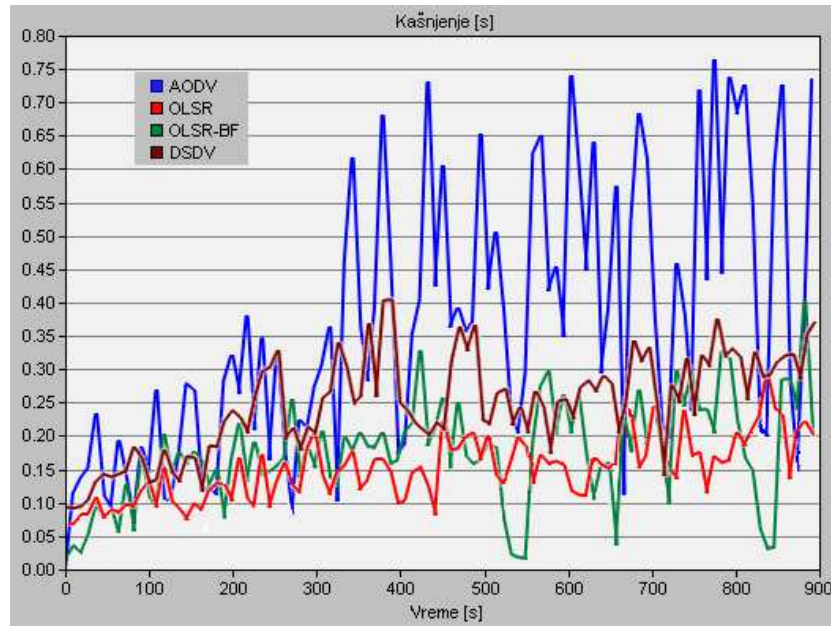
8.2.1. Propusni opseg



SLIKA 8.8. PROPUSNI OPSEG [BIT/S]

Slika 8.8 poredi vrednosti propusnog opsega za AODV, DSDV, OLSR i OLSR-BF ruting protokole u mreži sa 50 mobilnih čvorova. Uočava se da je ukupna vrednost propusnog opsega uvećana sa povećanjem broja čvorova unutar mreže, jer sada postoji veći broj dostupnih čvorova za prosleđivanje paketa podataka do odredišta. Uz zadržavanje stepena mobilnosti i dužine paketa konstantnim, OLSR-BF pruža veću propusnu moć od tradicionalnih proaktivnih ruting protokola, i znatno višu vrednost od reaktivnog AODV ruting protokola.

8.2.2. Kašnjenje

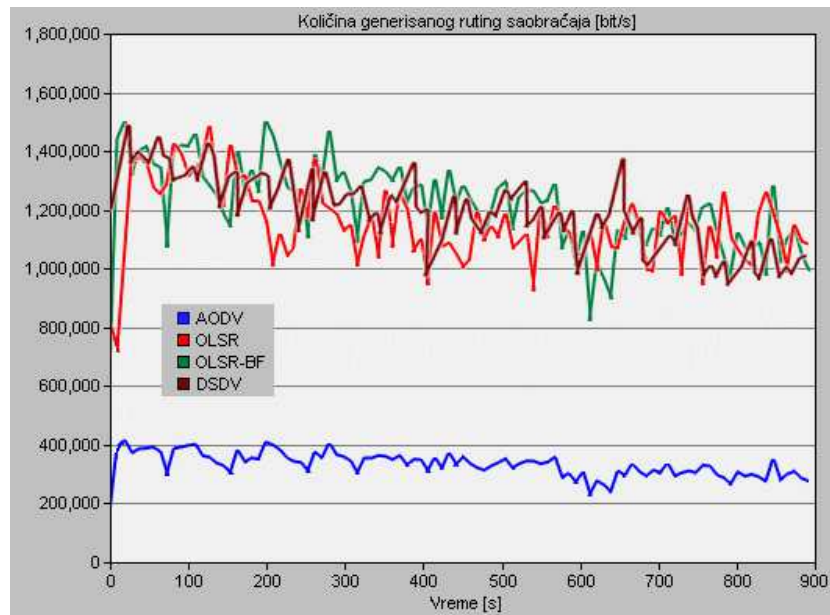


SLIKA 8.9. KAŠNJENJE [S]

Slika 8.9 prikazuje kašnjenje za AODV, DSDV, OLSR i OLSR-BF ruting protokole u mreži sa 50 mobilnih čvorova. Povećani broj čvorova unutar mreže omogućuje veći broj preskoka i različitih putanji do odredišta, što uvećava kašnjenje kod reaktivnog ruting protokola AODV, usled agresivnog brodkastovanja paketa za uspostavljanje putanje do odredišta. S druge strane, proaktivni ruting protokoli DSDV, OLSR i OLSR-BF ostvaruju veću vrednost propusnog opsega uz održavanje niskog vremena kašnjenja u mreži. U poređenju sa scenarijom mreže sa manjim brojem čvorova, AODV jeste povećao propusnu moć, ali uz znatno povećanje ukupnog kašnjenja u prenosu paketa podataka do odredišta.

8.2.3. Opterećenost mreže kontrolnim saobraćajem

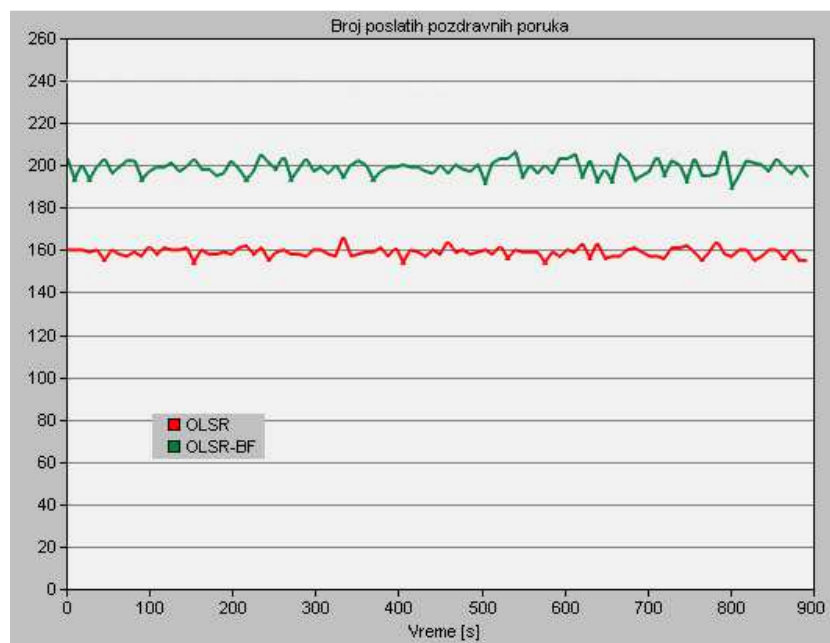
Slika 8.10 prikazuje količinu ruting saobraćaja poslatog u mreži sa 50 mobilnih čvorova za sva tri ruting protokola. Uočava se da kod proaktivnih ruting protokola DSDV, OLSR i OLSR-BF vrednost opterećenja mreže kontrolnim saobraćajem počinje sa naglom visokom vrednošću koja se potom lagano snižava tokom trajanja simulacije. Razlog tome je što se kod proaktivnih ruting protokola celokupna ruting tabela sa putanjama do svih čvorova uspostavlja na svakom čvoru unutar mreže, bez obzira na činjenicu da li se date putanje koriste.



SLIKA 8.10. OPTEREĆENOST MREŽE KONTROLNIM SAOBRAĆAJEM [BIT/S]

Povećanje vrednosti opterećenja mreže kontrolnim saobraćajem usled povećanja broja mobilnih čvorova, nastaje kao posledica činjenice da paketi treba da prođu preko većeg broja središnjih čvorova do svog odredišta. Dakle, zbog povećanog broja mobilnih čvorova povećava se opterećenje unutar mreže.

8.2.4. Broj poslatih pozdravnih poruka

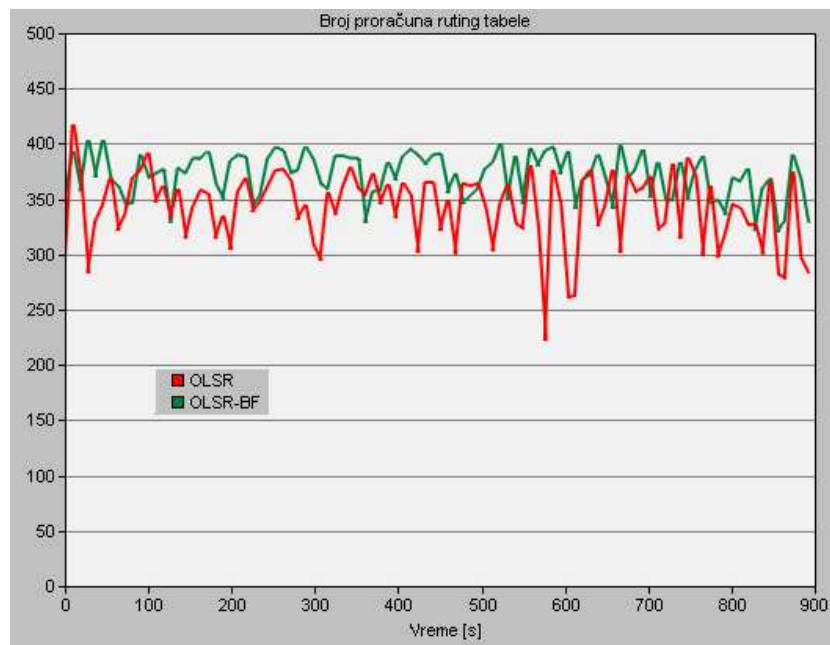


SLIKA 8.11. BROJ POSLATIH POZDRAVNIH PORUKA

Broj poslatih pozdravnih poruka od strane 50 mobilnih čvorova kod OLSR i OLSR-BF rutinog protokola prikazan je na slici 8.11. Uočava se da je ova vrednost znatno viša u odnosu na

scenario mreže sa manjim brojem mobilnih čvorova, dok je odnos između OLSR-BF i OLSR ruting protokola ostao nepromenjen.

8.2.5. Broj proračuna ruting tabele



SLIKA 8.12. BROJ PRORAČUNA RUTING TABELE

Slika 8.12 pokazuje ukupan broj proračuna ruting tabele tokom simulacije mreže sa 50 mobilnih čvorova. Uočljivo je znatno povećanje dobijenih vrednosti kod oba proaktivna ruting protokola OLSR i OLSR-BF u odnosu na scenario sa manjim brojem čvorova.

9. ANALIZA REZULTATA EKSPERIMENATA

U ovom radu kreirano je bežično mobilno ad hoc mrežno okruženje za merenje performansi predloženog ruting protokola. Za izvođenje eksperimenata korišćeni su OPNET Modeler i ns-3 mrežni simulatori. Prilikom izvođenja eksperimenata prenošen je FTP saobraćaj korišćenjem IPv4 adresiranja, i pritom su beležene vrednosti propusnog opsega, kašnjenja i količine generisanog kontrolnog saobraćaja. Radi poređenja rezultata eksperimenti su izvedeni i za druge postojeće ruting protokole AODV, DSDV i OLSR. Rezultati eksperimenata grafički su predstavljeni u prethodnom osmom poglavlju.

9.1. Analiza posmatranih metrika na performanse ruting protokola

9.1.1. Propusni opseg

Propusni opseg je odnos ukupne količine paketa podataka koja dostiže prijemnik u datom vremenskom periodu. Kada broj čvorova u mreži raste, propusni opseg se takođe povećava i time se povećavaju ukupne performanse ruting protokola.

U slučaju scenarija mreže sa 20 mobilnih čvorova proaktivni ruting protokoli imaju visoku propusnost. DSDV, OLSR i OLSR-BF ruting protokoli zasnovani su na stanju veza, što znači da su putevi do odredišta uvek na raspolaganju. Razlog za visok protok u poređenju sa reaktivnim AODV ruting protokolom nalazi se upravo u činjenici da su putevi do odredišta lako dostupni zahvaljujući prirodi proaktivnih ruting protokola.

U scenariju mreže sa većim brojem mobilnih čvorova odnos u veličini propusnog opsega između posmatranih ruting protokola blago je promenjen. Svi ruting protokoli pokazuju porast vrednosti propusnog opsega sa porastom broja čvorova. OLSR i OLSR-BF imaju sličan oblik ponašanja vrednosti propusnog opsega; DSDV ostvaruje nešto niže vrednosti, dok AODV pokazuje znatno manje vrednosti propusnog opsega. Razlog povećanja propusne moći kod OLSR i OLSR-BF ruting protokola nastaje usled korišćenja MRP čvorova, koji povećavaju efikasnost i smanjuju količinu generisanog kontrolnog saobraćaja kontrolisanjem i lokalizovanjem plavljenja mreže, što ih čini efikasnijim u procesu održavanja veza bez potrebe da svi čvorovi unutar mreže učestvuju u tome. OLSR i OLSR-BF ruting protokoli pogodni su za primenu u gustim mrežama sa relativno velikim brojem čvorova, što je vidljivo iz činjenice da oba ruting protokola ostvaruju više vrednosti propusnog opsega u scenariju sa većim brojem čvorova.

Primećuje se da DSDV, OLSR i OLSR-BF ruting protokoli daleko nadmašuju AODV reaktivni ruting protokol u oba scenarija. S obzirom da su DSDV, OLSR i OLSR-BF proaktivni ruting protokoli, putevi su lako i uvek dostupni za saobraćaj. Proaktivni ruting protokoli održavaju konzistentne putanje u mreži što za posledicu ima nisko kašnjenje. Nisko kašnjenje u mreži dovodi do visoke propusne moći i obrnuto.

Ruting protokol koji ima veću propusnu moć uopšteno daje najbolje rezultate. Navedeni rezultati pokazuju da OLSR-BF ostvaruje veću propusnu moć od tradicionalnih ruting protokola.

9.1.2. Kašnjenje

U svim izvedenim eksperimentima DSDV, OLSR i OLSR-BF ruting protokoli ostvarili su nižu vrednost kašnjenja u odnosu na reaktivni AODV ruting protokol. DSDV, OLSR i OLSR-BF su proaktivni ruting protokoli, što znači da su putevi u mreži uvek dostupni, bez obzira da li u mreži postoje paketi podataka koji trebaju biti prenešeni. Periodične ruting ispravke održavaju puteve aktuelnim unutar mreže. Odsustvo kašnjenja prilikom procesa otkrića ruta objašnjava relativno nisko ukupno kašnjenje DSDV, OLSR i OLSR-BF ruting protokola. U posmatranim scenarijima DSDV, OLSR i OLSR-BF ruting protokoli ostvarili su konzistentno kašnjenje u najvećoj meri zbog svojih proaktivnih karakteristika.

Kod proaktivnih DSDV, OLSR i OLSR-BF ruting protokola čvor koji želi da pronađe put do odredišta treba samo da pogleda u svojoj ruting tabeli. Kod reaktivnog AODV ruting protokola čvor mora pokrenuti proces otkrivanja nove putanje, ukoliko važeća putanja do odredišta ne postoji. Pošto pretraživanje postojeće ruting tabele zahteva znatno manje vremena nego pretraživanje mreže za novom putanjom, proaktivni ruting protokoli se znatno bolje ponašaju u mrežama osetljivim na kašnjenje.

U scenariju mreže sa 50 mobilnih čvorova AODV trpi značajnu degradaciju performansi usled visokog stepena kašnjenja. Jedan od osnovnih razloga za degradaciju kašnjenja kod AODV ruting protokola pripisuje se procesu otkrivanja ruta. Izgradnja ruting putanja kod AODV ruting protokola ne može se brzo uspostaviti, što dovodi do potencijalno dugotrajnog odlaganja prilikom slanja paketa podataka. Uopšteno, AODV ruting protokol povećava kašnjenje usled povećanja broja čvorova. Takođe, uočljivo je da vrednost kašnjenja kod AODV ruting protokola ima veće oscilacije u pogledu različitog broja čvorova.

Sa povećanjem broja čvorova unutar mreže odnos između posmatranih ruting protokola ostaje nepromenjen, uz napomenu da su rezultati reaktivnog ruting protokola AODV pogoršani u odnosu na scenario mreže sa manjim brojem čvorova. Kod DSDV ruting protokola uočljiv je nešto viši stepen kašnjenja u poređenju sa scenarijom mreže sa manjim brojem čvorova, ali sveukupno ostaje na relativno niskom nivou. Zaključuje se da su OLSR i OLSR-BF izloženi veoma niskom kašnjenju u svim scenarijima, kao i da je funkcija njihovog kašnjenja konzistentna. AODV ima visoko kašnjenje pogotovo u mreži sa većim brojem mobilnih čvorova. Navedeni rezultati pokazuju da u pogledu vremena kašnjenja, OLSR-BF ruting protokol ostvaruje neznatno lošiji učinak od svog prethodnika.

9.1.3. Opterećenost mreže kontrolnim saobraćajem

U scenariju mreže sa 20 mobilnih čvorova AODV ima najmanju vrednost opterećenja mreže. AODV ima manje opterećenje mreže zbog uspostavljanja putanja do odredišta na zahtev, odnosno samo kada postoji potreba za prenos paketa podataka do odredišta. Usled toga, AODV nema potrebu za ažuriranjem ruting tabele. Proaktivni ruting protokoli DSDV, OLSR i OLSR-BF ostvaruju znatno više mrežno opterećenje usled činjenice da konstantno održavaju ruting tabelu na čvorovima, što izaziva dodatno opterećenje na mreži.

Superiornost AODV ruting protokola potiče iz njegove reaktivne prirode. Kao reaktivni ruting protokol, AODV šalje ruting saobraćaj u mrežu samo kada izvor ima podatke za slanje, čime se eliminišu nepotrebne kontrolne ruting poruke. Međutim, količina kontrolnih poruka blago se povećava kao posledica višestrukih odgovora na pojedinačne zahteve za uspostavljanje rute. Osim toga, kada dođe do otkaza nekog od središnjih čvorova na putu do odredišta, poruka o grešci (RRER) se propagira prema svim susednim čvorovima. Ovo inicira ponovni postupak procesa otkrića novog puta do odredišta, čime se povećava količina generisanog kontrolnog ruting saobraćaja.

Slično važi i za scenario mreže sa 50 mobilnih čvorova, gde AODV opet ima najmanje opterećenje mreže. Primetno je da OLSR-BF generiše najveći iznos ruting saobraćaja u mreži, blisko praćen sa DSDV i OLSR ruting protokolima. DSDV, OLSR i OLSR-BF su proaktivni ruting protokoli i kao takvi konstantno plave mrežu sa kontrolnim i ruting saobraćajem kako bi održali validnim ruting tabele. To objašnjava ogromno opterećenje kontrolnim saobraćajem kod proaktivnih ruting protokola. Ova konstatacija odnosi se na oba posmatrana scenarija. U mrežama sa niskom stopom kapaciteta linkova AODV je znatno upotrebljiviji od proaktivnih ruting protokola.

Dobijeni rezultati pokazuju da je AODV stabilan i efikasan ruting protokol u pogledu opterećenja mreže. Najgori učinak u smislu količine generisanih kontrolnih poruka uočava se kod OLSR-BF ruting protokola.

9.1.4. Iskorišćenost resursa

Efikasnost iskorišćenja resursa kod proaktivnih DSDV, OLSR i OLSR-BF ruting protokola zavisi od broja čvorova u mreži. Naime, kod proaktivnih ruting protokola svi čvorovi moraju stalno održavati informacije o celoj mreži. Ruting protokoli moraju čuvati topološke informacije, informacije o MPR skupu selektora i moraju konstantno ažurirati informacije o stanju veza i susednih čvorova. Proaktivni ruting protokoli održavaju informacije o putevima koji se možda nikada neće koristiti. Generisanje kontrolnog saobraćaja povećava potrebnu obradu informacija na svakom čvoru. Zbog svega toga, potrošnja energije je veća nego kod reaktivnih ruting protokola.

Kod OLSR i OLSR-BF ruting protokola rute se održavaju putem TC poruka, kod DSDV protokola koriste se ažurirane ruting poruke, a kod AODV protokola rute se održavaju povremenim odašiljanjem pozdravnih poruka. OLSR i OLSR-BF pokušavaju da minimiziraju ovaj saobraćaj korišćenjem MPR čvorova, tako da samo ovi čvorovi emituju TC poruke. Pored toga, koriste pozdravne poruke za održavanje statusa susedstva. Kod DSDV ruting protokola poruke sa ispravkama ruti mogu sadržati sliku kompletnog stanja ili inkrementalne ispravke. S druge strane, kod AODV ruting protokola samo čvorovi koji učestvuju u komunikaciji povremeno šalju pozdravne poruke sa ograničenjem opsega prostiranja do jednog preskoka. Pored toga, veličina pozdravnih poruka kod AODV ruting protokola znatno je manja nego kod proaktivnih ruting protokola.

Proaktivni ruting protokoli DSDV, OLSR i OLSR-BF zahtevaju više propusnog opsega i energetske resurse u odnosu na reaktivni AODV ruting protokol.

9.1.5. Skalabilnost

AODV ruting protokol funkcioniše bolje u mrežama sa statičkim izvorima saobraćaja, pri čemu se komunikacija odvija po ustaljenim putanjama između čvorova. S druge strane, proaktivni ruting protokoli DSDV, OLSR i OLSR-BF imaju prednost u mrežama sa velikom gustinom čvorova i visoko sporadičnim saobraćajem. Zajedničko za sve ruting protokole je njihova ograničena skalabilnost usled povećanja veličine mreže. Kod DSDV, OLSR i OLSR-BF ruting protokola veličina ruting tabele raste nelinearno, a kontrolne poruke mogu blokirati pakete podataka. U slučaju AODV ruting protokola problem koji se javlja je plavljenje mreže ogromnom količinom paketa za pretragu puteva.

9.2. Analiza posmatranih faktora na performanse ruting protokola

9.2.1. Veličina mreže

Veličina mreže je faktor koji ima važnu ulogu u određivanju ukupnih performansi ruting protokola. Za neke ruting protokole efekat povećanja veličine mreže u smislu broja čvorova smatra se kao prednost, dok za druge protokole predstavlja nedostatak.

U scenariju mreže sa 20 mobilnih čvorova proaktivni ruting protokoli DSDV, OLSR i OLSR-BF optimalno funkcionišu. Zahvaljujući činjenici da održavaju ruting tabelu aktuelnom, važeće rute se mogu lako pronaći i koristiti. OLSR i OLSR-BF ruting protokoli pomoću MPR čvorova u umereno mobilnim sredinama imaju visoku verovatnoću održavanja važećih putanja do odredišta [102]. S druge strane, reaktivni AODV ruting protokol ima znatno višu vrednost kašnjenja koja degradira njegove ukupne performanse. Sa porastom veličine mreže povećava se kašnjenje kod AODV ruting protokola, što za posledicu ima i nižu ostvarenu vrednost propusnog opsega u odnosu na proaktivne ruting protokole.

U scenariju mreže sa 50 mobilnih čvorova proaktivni ruting protokoli i dalje ostvaruju bolje rezultate. DSDV, OLSR i OLSR-BF ponovo imaju manju stopu kašnjenja zbog svoje proaktivne prirode. Proaktivni ruting protokoli ostvaruju dobre performanse u velikoj mreži, pre svega zahvaljujući svojoj proaktivnoj prirodi i činjenici da pakete podataka prosleđuju odmah po prijemu duž putanja koje se održavaju aktuelnim unutar ruting tabele. U velikim mrežama čvorovi uglavnom komuniciraju sa svojim susedima prilikom otkrivanja mogućih putanja. Ako lokalna komunikacija dominira, putevi će ostati konstantni sa porastom veličine mreže. Međutim, često korišćenje istih putanja u velikoj mreži za posledicu ima veću verovatnoću da se dogodi neupeh u komunikaciji usled otkaza pojedinih linkova. Ovaj problem se smatra značajnim faktorom koji utiče na krajnje performanse ruting protokola.

9.2.2. Mobilnost čvorova

Mobilnost unosi stres u ruting protokol usled raskida veza i troškova proisteklih usled ponovnog otkrivanja ruta. Raskid veza nastaje kao posledica nevažećih puteva do čvorova koji nestaju sa mreže ili se udaljavaju iz komunikacionog opsega. Kada dođe do raskida veze, povećava se broj kontrolnih ruting paketa i time se povećava opterećenost mreže. Pored toga, pojedini ruting protokoli ne nose se dobro sa različitim saobraćajnim obrascima. Ovo dodaje

značajan teret na ruting protokole, što izaziva značajnu degradaciju njihovih performansi tokom intenzivne mobilnosti čvorova. Dakle, mobilnost čvorova ima važnu ulogu u određivanju performansi ruting protokola. Usled prisustva mobilnosti dolazi do raskidanja veza što dovodi do reemitovanja paketa unutar mreže. Akumulacija reemitovanja poruka povećava opterećenje mreže, što dovodi do smanjenja protoka i ukupnih performansi ruting protokola. Usled povećanja mobilnosti čvorova nastaje veći broj raskinutih veza, što dovodi do povećanja opterećenosti mreže kontrolnim saobraćajem.

Svaki ruting protokol ima različite pristupe za rešavanje problema mobilnosti čvorova u MANET mrežama. Dinamična topologija MANET mreža izaziva prekide putanja. Kada se to desi, ruting protokol mora pronaći nove puteve. DSDV, OLSR i OLSR-BF ruting protokoli periodično ažuriraju topološke informacije, tako da se novi putevi izračunavaju odmah po prijemu informacije o prekidu nekog od puteva. AODV je reaktivni protokol, tako da ne poseduje mehanizam za momentalni oporavak trasa. Stoga, proces otkrivanja novog puta mora biti iniciran. Prilikom distribuiranja mrežnog saobraćaja DSDV, OLSR i OLSR-BF ruting protokoli imaju manje troškove od reaktivnog ruting protokola, jer informacije o svim putevima čuvaju u okviru ruting tabele. S druge strane, AODV prvo mora otkriti rutu pre nego što se informacije mogu prenositi. Dakle, u slučaju mreža sa sporadičnim saobraćajem AODV ima velike kontrolne troškove po svakom prenešenom paketu podataka. U suprotnom, kada saobraćaj ima dugoročnije trajanje (tj. saobraćaj je statičan) AODV ostvaruje znatno bolje rezultate, jer se količina generisanih kontrolnih paketa smanjuje sa svakim prenešenim paketom podataka.

9.3. Komparativna analiza ruting protokola

U ovom poglavlju predstavljena je komparativna analiza rezultata simulacija različitih ruting protokola sa njihovim prednostima i manama:

- DSDV, OLSR i OLSR-BF su proaktivni ruting protokoli koji održavaju puteve na svakom čvoru u mreži, dok je AODV reaktivni ruting protokol koji pronalazi puteve na zahtev, odnosno kada je potrebna trasa do odredišta.
- Kontrolni ruting saobraćaj se kod DSDV, OLSR i OLSR-BF protokola periodično brodkastuje kako bi se ruting informacije održavale ažurnim, dok se kod AODV protokola samo pozdravne poruke prosleđuju ka susednim čvorovima u cilju održanja lokalnih veza.
- DSDV, OLSR i OLSR-BF ruting protokoli koriste koncept rednih brojeva za ažuriranje najnovijih informacija o rutama. Isti koncept je zastupljen i kod reaktivnog AODV ruting protokola.
- Zbog periodičnih ruting ispravki koje se emituju kod DSDV, OLSR i OLSR-BF protokola, propusni opseg se koristi čak i kada u mreži ne postoji prenos paketa podataka. To nije slučaj sa AODV ruting protokolom koji propagira samo pozdravne poruke ka svojim susedima.
- Za slanje paketa podataka na odredište kod DSDV, OLSR i OLSR-BF ruting protokola nema potrebe za otkrivanjem rute, jer se svi putevi već nalaze u ruting tabelama svakog čvora. S druge strane, AODV ruting protokol mora otkriti put pre slanja podataka, što unosi određeno kašnjenje prilikom slanja paketa podataka.
- Količina generisanog kontrolnog saobraćaja kod DSDV, OLSR i OLSR-BF ruting protokola raste sa porastom veličine mreže. Kod AODV ruting protokola količina

generisanog kontrolnog saobraćaja je manja, jer se održavaju samo male tabele sa lokalnim vezama.

- Proaktivni ruting protokoli DSDV, OLSR i OLSR-BF se teško bore sa visokim stepenom mobilnosti čvorova zbog nepostojanja alternativnih puteva u ruting tabelama koje postaju ustajale. Kod AODV ruting protokola ovaj problem je manje izražen, jer se putevi pronalaze na zahtev.
- Propusni opseg kod DSDV, OLSR i OLSR-BF protokola varira usled brodkastovanja periodičnih ruting ispravki. Sa povećanjem pokretljivosti čvorova ove oscilacije postaju još izraženije. Kod AODV ruting protokola ne oglašavaju se ruting ispravke, te je stoga propusni opseg unutar mreže znatno stabilniji.

9.4. Sumarna analiza rezultata eksperimenata

Tehnika koja se koristi u poređenju rezultata simulacija je tehnika grafičkog poređenja rezultata eksperimenata i zaključaka proisteklih iz prethodnih radova u literaturi.

Reaktivni AODV ruting protokol je simuliran u dva scenarija mreže sa različitim brojem mobilnih čvorova, pri čemu su beležene vrednosti propusnog opsega, kašnjenja i opterećenja mreže. Uočljivo je da kašnjenje kod AODV ruting protokola pokazuje velike oscilacije u oba scenarija, dok vrednosti propusnog opsega i opterećenja mreže prikazuju znatno stabilnije vrednosti. Opterećenje mreže je nisko u oba posmatrana scenarija. Propusni opseg znatno je viši u scenariju sa većim brojem mobilnih čvorova.

Istovetni scenariji mreža korišćeni su za analizu učinka proaktivnih OLSR i OLSR-BF ruting protokola, pri čemu su beležene vrednosti propusnog opsega, kašnjenja, mrežnog opterećenja, broja generisanih pozdravnih poruka i broja proračuna ruting tabele. Opterećenje i propusni opseg kod oba OLSR i OLSR-BF ruting protokola znatno su viši u scenariju sa većim brojem mobilnih čvorova. Kašnjenje je relativno nisko i pokazuje isto ponašanje u oba scenarija.

Proaktivni DSDV ruting protokol simuliran je u ns-3 mrežnom simulatoru u dva scenarija mreže sa različitim brojem mobilnih čvorova, pri čemu su beležene vrednosti propusnog opsega, kašnjenja i opterećenja mreže. Eksperimenti izvedeni nad DSDV ruting protokolom ekvivalentni su sa eksperimentima sprovedenim nad ostalim ruting protokolima u okviru OPNET Modeler mrežnog simulatora. Ponašanje performansi DSDV protokola slično je sa OLSR i OLSR-BF ruting protokolima uz napomenu da je u scenariju sa većim brojem mobilnih čvorova ostvaren niži stepen povećanja propusnog opsega, dok je stepen povećanja kašnjenja u blagom porastu.

Poređenjem različitih ruting protokola u scenariju sa 20 mobilnih čvorova uočava se da OLSR-BF ruting protokol ostvaruje najvišu vrednost propusnog opsega u mreži. Što se tiče kašnjenja, rezultati proaktivnih ruting protokola su ujednačeni i imaju niske vrednosti, dok su kod AODV ruting protokola vrednosti kašnjenja visoke. S druge strane, AODV nudi nisku vrednost opterećenja mreže u poređenju sa proaktivnim ruting protokolima. U malim mrežama sa niskom mobilnošću čvora dobijeni rezultati favorizuju korišćenje proaktivnih ruting protokola.

Kada se broj mobilnih čvorova poveća na 50, uz zadržavanje istovetnog stepena mobilnosti i ostalih parametara bežične mreže, proaktivni ruting protokoli održavaju visoku vrednost

propusnog opsega uz nisku vrednost kašnjenja. S druge strane, AODV ostvaruje najniže vrednosti opterećenja mreže, ali uz visoku vrednost kašnjenja. U velikoj mreži sa 50 čvorova eksperimentalni rezultati favorizuju upotrebu proaktivnih ruting protokola usled visokog propusnog opsega i malog vremena kašnjenja. Rezultati eksperimenata ukazuju na degradaciju performansi proaktivnih DSDV, OLSR i OLSR-BF ruting protokola sa povećanjem stepena mobilnosti čvorova.

Sumarni pregled rezultata eksperimenata za oba scenarija mreže sa različitim brojem mobilnih čvorova prikazan je u tabeli 9.1.

TABELA 9.1. SUMARNI PREGLED REZULTATA EKSPERIMENATA

Broj čvorova	Parametar	AODV	DSDV	OLSR	OLSR-BF
20	Propusni opseg	Umeren	Visok	Visok	Najviši
	Kašnjenje	Visoko	Nisko	Najniže	Nisko
	Opterećenost mreže kontrolnim saobraćajem	Niska	Visoka	Visoka	Najviša
	Broj poslatih pozdravnih poruka	/	/	Visok	Najviši
	Broj proračuna ruting tabele	/	/	Visok	Najviši
50	Propusni opseg	Umeren	Visok	Visok	Najviši
	Kašnjenje	Najviše	Umereno	Najniže	Nisko
	Opterećenost mreže kontrolnim saobraćajem	Umerena	Visoka	Visoka	Najviša
	Broj poslatih pozdravnih poruka	/	/	Visok	Najviši
	Broj proračuna ruting tabele	/	/	Visok	Najviši

S obzirom da su DSDV, OLSR i OLSR-BF ruting protokoli zasnovani na stanju veze, i usled činjenice da održavaju aktuelnim ruting tabele, ostvaruju visoke vrednosti generisanog kontrolnog saobraćaja, što povećava potrošnju propusnog opsega i tako povećava opterećenost mreže. Održavanje i ažuriranje ruting tabele za celu mrežu zahteva intenzivnu komunikaciju između čvorova, kao i plavljenje periodičnih ruting ispravki unutar mreže. Efekat mobilnosti čvorova u mreži izaziva dodatno generisanje kontrolnog saobraćaja kod proaktivnih ruting protokola usled čestih promena stanja veza. Opterećenost mreže kontrolnim saobraćajem kod AODV ruting protokola povećava se usled prenošenja RREQ i RREP paketa tokom procesa otkrića novih ruta. Sa porastom broja mobilnih čvorova unutar mreže povećava se i stopa generisanih kontrolnih poruka kod DSDV; OLSR i OLSR-BF ruting protokola. Efekat mobilnosti čvorova izraženiji je kod proaktivnog ruting pristupa. Proaktivni ruting protokoli su preferirani u odnosu na reaktivne kada se razmatra propusna moć ruting protokola.

Učinak reaktivnog AODV ruting protokola u smislu opterećenja mreže je znatno bolje u odnosu na proaktivne ruting protokole DSDV, OLSR i OLSR-BF. S druge strane, proaktivni ruting protokoli imaju manje kašnjenje u odnosu na AODV ruting protokol. Povećanje

veličine mreže ne utiče na performanse DSDV, OLSR i OLSR-BF ruting protokola zbog njihove proaktivne prirode, što znači da koriste raspoložive puteve do odredišta iz svojih ruting tabela. U slučaju propusnog opsega OLSR-BF ostvaruje najviše rezultate u oba scenarija. Proaktivni ruting protokoli imaju visoku propusnost u odnosu na AODV reaktivni ruting protokol usled činjenice da konstantno održavaju ruting tabele na svim čvorovima unutar mreže. AODV zahteva više vremena za pronalaženje trase do odredišta. Ako na aktivnom putu dođe do prekida ponovo se pokreće čitav proces uspostavljanja nove rute, jer AODV ne održava ruting informacije unutar mreže.

DSDV, OLSR i OLSR-BF su proaktivni ruting protokoli koji za obavljanje rutiranja na svakom čvoru koriste jednu ili više ruting tabela. Kao rezultat čuvanja celokupnih ruting tabela na svim čvorovima, proaktivni ruting protokoli ostvaruju znatno niže kašnjenje u odnosu na reaktivni AODV ruting protokol. Pronalaženje putanja do odredišta brže je kod proaktivnog nego li kod reaktivnog ruting pristupa. Pored toga, OLSR i OLSR-BF ruting protokoli imaju prednost u korišćenju MPR čvorova za slanje kontrolnih poruka prema drugim čvorovima, čime se smanjuje količina poslatih kontrolnih poruka. Dakle, za mreže sa malim brojem čvorova umerene mobilnosti proaktivni ruting protokoli DSDV, OLSR i OLSR-BF pokazuju bolje performanse u odnosu na AODV ruting protokol. U izuzetno mobilnim mrežama sa velikim brojem čvorova, za pretpostaviti je da DSDV, OLSR i OLSR-BF ruting protokoli neće biti toliko efikasni.

Proaktivni ruting protokoli funkcionišu dobro u oba mrežna scenarija i ostvaruju najnižu prosečnu vrednost kašnjenja. DSDV, OLSR i OLSR-BF su proaktivni ruting protokoli, tako da kada paket stigne na čvor odmah se prosleđuje duž puteva koji se čuvaju unutar ruting tabele. Stoga, DSDV, OLSR i OLSR-BF imaju veću propusnu moć od reaktivnih ruting protokola. Drugi očigledan razlog je to što TCP saobraćaj zahteva poznavanje trase od izvora do odredišta i obrnuto, što je znatno zahtevnije za AODV reaktivni ruting protokol.

OLSR i OLSR-BF ruting protokoli ostvaruju konzistentan protok i kašnjenje tokom izvođenja eksperimenata. DSDV ih blisko prati uz nešto višu vrednost kašnjenja u prenosu paketa podataka. Pošto je propusni opseg zavistan od ovih parametara, konzistentna vrednost propusnog opsega je očekivana. Ovo pokazuje superiornost proaktivnih ruting protokola u odnosu na reaktivni AODV ruting protokol. Treba napomenuti da sa porastom veličine mreže raste i veličina ruting tabele kod DSDV, OLSR i OLSR-BF ruting protokola, što može dovesti do eventualnog zagušenja i drugih problema u bežičnim mrežama. Prednost čuvanja kompletne ruting tabele kod proaktivnih ruting protokola može dovesti do opšte degradacije performansi MANET mreže.

Rezultati simulacija dali su važna saznanja o performansama različitih ruting protokola. Prisustvo mobilnosti, opterećenosti saobraćajem i različitih veličina mreže ima različite reakcije na performanse MANET ruting protokola.

10. ZAKLJUČAK

U ovom radu kvantifikovane su vrednosti performansi različitih mobilnih ad hoc ruting protokola u pogledu njihovog propusnog opsega, kašnjenja i opterećenosti mreže kontrolnim saobraćajem. Posmatrane metrike kao što su propusna moć i kašnjenje odnose se na pouzdanost ruting protokola. Efekat opterećenja mreže bavi se efikasnošću iskorišćenja mrežnih resursa od strane ruting protokola. Unutar mreže poželjno je da ruting protokol bude istovremeno pouzdan i efikasan.

Dublje razumevanje performansi MANET ruting protokola dobijeno je sprovođenjem različitih simulacionih eksperimenata. Rezultati eksperimenata dali su jasnu sliku o tome koji ruting protokol je najbolji za primenu u datom modelu mreže. Pored komparativne analize, eksperimentalni rezultati daju jasan uvid u različite parametre koji utiču na performanse različitih ruting protokola.

Cilj ovog rada jeste poređenje učinka novopredloženog OLSR-BF ruting protokola u odnosu na postojeće AODV, DSDV i OLSR ruting protokole. Za procenu performansi ovih protokola korišćeni su OPNET Modeler i ns-3 mrežni simulatori u kojima je kreiran model mobilnih ad hoc mreža. Za potrebe izvođenja eksperimenata kreirana su dva scenarija sa različitim brojem mobilnih čvorova. Unutar mreže generisan je FTP saobraćaj sa izvornih čvorova ka zajedničkom odredištu (FTP server).

Rezultati eksperimenata pokazuju da DSDV, OLSR i OLSR-BF ruting protokoli nadmašuju reaktivni AODV ruting protokol u pogledu veličine propusnog opsega i niskog kašnjenja u mrežama niske mobilnosti i visoke stope saobraćaja. S druge strane, AODV je potpuna suprotnost u odnosu na proaktivne ruting protokole i generiše znatno manje opterećenje mreže kontrolnim saobraćajem, ali je podbacio u svim drugim aspektima koji favorizuju korišćenje proaktivnih DSDV, OLSR i OLSR-BF ruting protokola. OLSR-BF ima najlošiji učinak u pogledu količine generisanih kontrolnih poruka, te stoga nije pogodan za primenu u mrežama sa niskim kapacitetom linkova. Konačno, uzimajući u obzir rezultate ovog istraživanja može se zaključiti da izbor odgovarajućeg ad hoc ruting protokol treba vršiti na osnovu mrežne topologije i količine saobraćaja koji se putem mreže prenosi.

Kvantitativno proučavanje performansi ruting protokola pokazuje da OLSR-BF ruting protokol dominira u mrežama visoke gustine i niske mobilnosti čvorova. Međutim, OLSR-BF ruting protokol zahteva stalno korišćenje određenog kapaciteta na linkovima u cilju ažuriranja mrežne topologije i ruting tabele. Takođe, njegove performanse degradiraju se usled povećanja stepena mobilnosti čvorova. Sličan efekat prisutan je i kod postojećih DSDV i OLSR ruting protokola. Uopšteno, učinak novopredloženog OLSR-BF ruting protokola efikasniji je od originalnog OLSR ruting protokola u oba posmatrana scenarija.

Eksperimentalna analiza pokazuje da je predloženi OLSR-BF ruting protokol bolji u pogledu veličine propusnog opsega od svog direktnog prethodnika OLSR i od tradicionalnog DSDV ruting protokola, kao i da je daleko je bolji od reaktivnog AODV ruting protokola. Međutim, nije dokazano da ovo važi za sve mreže, jer performanse ruting protokola mogu varirati u različitim mrežnim okruženjima. Na osnovu izvedenih simulacija i analitičke studije

zaključuje se da performanse i efikasnost MANET mreža u velikoj meri zavise od izbora adekvatnog ruting protokola.

Topologija MANET mreža može biti dinamična i nepredvidljiva. Tradicionalni ruting protokoli koji se koriste kod ožičenih mreža ne mogu biti direktno primenjeni na mobilne ad hoc bežične mreže, jer brojne usvojene pretpostavke ne važe u mobilnom bežičnom okruženju. Na primer, jedna od pretpostavki je da čvor može primiti bilo koju brodkastovanu poruku od strane drugih čvorova unutar iste podmreže (engl. *subnet*). Međutim, ovo nije slučaj kod čvorova u bežičnoj MANET mreži, usled činjenice da je propusni opseg kod ove vrste mreža ograničen. Dakle, model MANET mreža uvodi brojne izazove za ruting protokole.

U ovoj disertaciji predstavljeni su i klasifikovani postojeći ruting protokoli koji se koriste u okviru MANET mreža. Razmotrene su različite kategorije ruting protokola i detaljnije su predstavljeni njihovi najznačajniji predstavnici. Uočeno je da se svaki ruting protokol ponaosob bolje ponaša u određenom okruženju. Nijedan od njih nije savršen za primenu u svim opsezima mobilnosti čvorova, količine generisanog saobraćaja, broja čvorova, i sl. Zbog svojih karakteristika, tradicionalni ruting protokoli koji se koriste za ožičene mreže ne preporučuju se za upotrebu u MANET mrežama.

Tri glavne grupe unicast ruting protokola su proaktivni, reaktivni i hibridni ruting protokoli.

Osnovna karakteristika proaktivnih ruting protokola je činjenica da svaki čvor održava puteve do svih čvorova u mreži. Čvorovi povremeno ažuriraju ove informacije, bez obzira da li se date putanje koriste. S druge strane, kod reaktivnih ruting protokola čvorovi izračunavaju putanje samo između onih čvorova koji žele da komuniciraju. Ova vrsta ruting protokola efikasnije koristi propusni opseg bežičnih veza i ograničene resurse čvorova.

Kod proaktivnog pristupa osnovni problem je visok stepen iskorišćenja resursa i protoka kada to nije neophodno. Kod reaktivnog pristupa glavni problem je kašnjenje prilikom uspostavljanja novih puteva. Kod oba proaktivna i reaktivna pristupa javlja se problem skalabilnosti. U cilju ublažavanja ovih problema nastala je nova hibridna vrsta unicast ruting protokola. Hibridni ruting protokoli kombinuju oba proaktivna i reaktivna pristupa kako bi postigli bolje performanse. Međutim, nijedan od postojećih ruting protokola nije pogodan za MANET mreže sa velikim brojem čvorova, ukoliko svaki od čvorova ima različitu brzinu kretanja i količinu generisanog saobraćaja.

Na osnovu razumevanja prednosti i slabosti postojećih ruting protokola, predložen je novi proaktivni ruting protokol OLSR-BF. Cilj novog ruting protokola je unapređenje postojećeg OLSR ruting protokola u smislu boljeg iskorišćenja postojećih bežičnih veza unutar MANET mreže.

Tokom poslednjih decenija, napredak u informacionim tehnologijama stvorio je osnov za povećano interesovanje i istraživačku delatnost u oblasti ad hoc bežičnih mreža. Ova tehnologija omogućuje ostvarivanje komunikacije između bežičnih čvorova koji su raspoređeni na ad hoc i privremeni način. Svi čvorovi unutar ad hoc mreže istovremeno preuzimaju ulogu terminala i rutera u komunikaciji s kraja na kraj mreže u ime drugih čvorova. Ad hoc mreže su veoma dinamične prirode, odnosno čvorovi mogu da se pridruže ili da napuste mrežu u svakom trenutku, i poseduju fleksibilnost kretanja unutar same mreže. Činjenica da mobilne ad hoc mreže mogu biti brzo raspoređene sa minimalnim prethodnim

planiranjem, troškovima, i bez potrebe za postojećom infrastrukturom čini ovu tehnologiju veoma atraktivnom i pogodnom za brojne primene, uključujući vojne, hitne i spasilačke operacije.

Iako mobilne ad hoc mreže predstavljaju obećavajuću tehnologiju koja nudi širok spektar mogućih korisnih primena, ova tehnologija je još uvek u nezreloj fazi. Postoji još mnogo pitanja i izazova koje treba rešiti, koji uglavnom nastaju iz urođene nepouzdanosti bežične komunikacije, dinamične prirode ovih mreža, ograničene dostupnosti resursa u vezi sa protokom i napajanjem, kao i sa potencijalno velikim razmerama ovih mreža. Ovi izazovi zahtevaju da se mrežni protokoli na svim nivoima u steku mreže, koji su inače prvobitno dizajnirani za ožične mreže, izmene i optimizuju kako bi se prilagodili karakteristikama bežičnog okruženja.

Iz ovog istraživanja zaključuje se da ne postoji niti jedan ruting protokol koji je superioran u pogledu svih performansi. Jedan protokol može biti superioran u pogledu niske stope generisanog kontrolnog saobraćaja, dok drugi može biti superioran u pogledu visoke propusne moći i niskog vremena kašnjenja. Izbor određenog ruting protokola zavisi od namene mreže.

Konačno, činjenica da li je priroda ruting protokola proaktivna ili reaktivna ima dalekosežne posledice na performanse ruting protokola u različitim scenarijima. Glavne razlike ogledaju se prvenstveno u načinu otkrivanja i održavanja ruting putanja, što diktira dalje ponašanje ruting protokola. Generalno, proaktivni ruting protokoli su pogodni za primenu u mrežama sa visokim kapacitetom veza, dok reaktivni ruting protokoli bolje funkcionišu u mrežama niskih kapaciteta.

10.1. Budući rad

Iako je upotreba simulacije kao sredstva za vršenje istraživanja složenih pitanja i sistema pogodna i efikasna, mana ovog pristupa je nepreciznost. To je zbog činjenice da je veoma teško precizno modelovanje fizičkog okruženja, a modeli koji se koriste u većini simulatora predstavljaju samo njihovu aproksimaciju i pojednostavljenje. Tako na primer, domet prenosa određenog omnidirekcionog bežičnog interfejsa predstavljen je savršenim krugom radijusa R . Radio veza je savršeno primljena u tom opsegu, dok signal ne može biti primljen izvan njega. S druge strane, realni ospezi pokrivanja radio signala obično imaju neuniformne i nekružne oblasti zračenja [103], sa prostornom i vremenskom fluktuacijom signala koja može izazvati nagle promene u mrežnom povezivanju i opsegu prenosa signala. U [104] pokazano je da su modeli za propagaciju radio signala koji se često koriste u istraživanju mobilnih ad hoc mreža netačni i da mogu imati negativan uticaj na rezultate simulacija. Netačnosti proizilaze iz upotrebe pojednostavljenih pretpostavki u ovim modelima, kao što su idealno ravan teren, identičan opseg pokrivanja radio signala na svim čvorovima, predstavljanje jačine signala jednostavnom funkcijom rastojanja, i posmatranje svih radio veza kao simetričnih.

Nekoliko zanimljivih pitanja i otvorenih problema koji zahtevaju dalja istraživanja pojavili su se tokom ovog istraživanja:

- Većina akademskih istraživanja MANET mreža koriste metod simulacije za procenu performansi predloženih algoritama [105, 106, 107]. Međutim, metodom simulacije nije moguće ispitati velike scenarije zbog vremenskih ograničenja i složenosti, čime

se naglašava značaj analitičkih modela. Postojeći analitički modeli [108, 109, 110] razmatraju pojedine, ali definitivno ne sve karakteristike MANET mreža. Dakle, razvoj analitičkih modela za MANET mreže koji uzimaju u obzir sve njihove bitne karakteristike, uključujući mobilnosti, su poželjni. Ovim bi se omogućilo istraživanje performansi ruting protokola u scenarijima MANET mreža koje možda nije moguće razmatrati pomoću metoda simulacije, kao što su velike mreže koje rade pod teškim opterećenjem.

- Simulacija je važno sredstvo za proučavanje MANET mreža. Međutim, simulacija zahteva određene pretpostavke kako bi njihov model bio realizovan na prihvatljivom nivou. Shodno tome, ovaj model ne može obuhvatiti sve faktore koji mogu uticati na performanse MANET mreža. Štaviše, neke važne karakteristike poput energije i radio propagacije veoma je teško precizno modelovati u simuliranom okruženju.
- Stoga, nameće se potreba za razvojem poboljšane verzije simulatora koji bi mogao simulirati besprekornu interakciju mobilnih čvorova između dve ili više heterogene mobilne ad hoc mreže radi sudije njihovog učinka. Ovo bi definitivno uspostavilo nova savremena istraživanja u oblasti MANET mreža.
- Iako je simulacija dragoceno sredstvo u proceni performansi MANET sistema, često se zahteva pojednostavljenje određenih modela usled njihove složenosti (npr. modeli radio propagacije ili mobilnosti čvorova). Kao rezultat toga, simulacioni model možda neće uspeti u oponašanju svih faktora koji mogu uticati na performanse sistema. Do sada je bilo malo aktivnosti u implementaciji i merenju performansi realnih MANET sistema. U budućem radu, bilo bi korisno u realnom okruženju ponoviti eksperimentalna merenja i verifikovati rezultate simulacija objavljene u ovoj disertaciji. Pored proveravanja rezultata, rezultati prikupljeni u realnom okruženju bili bi od posebne koristi za razvoj budućih simulacionih modela.
- Evaluacija performansi novopredloženog OLSR-BF ruting protokola sprovedena u kontekstu istraživanja odnosi se na poređenje sa postojećim AODV, DSDV i OLSR ruting protokolima. Prirodni nastavak ovog rada bilo bi poređenje performansi sa drugim ruting protokolima, kao što DSR, ZRP i drugi.

Drugi alternativni pravac ovog istraživanja će istražiti izvodljivost razvoja novog ruting algoritma koji će se baviti poboljšanjem ograničenja koja su uočena kod ad hoc ruting protokola predstavljenih u ovom istraživanju. Tako na primer, OLSR-BF ruting protokol je superioran u odnosu na druge ruting protokole u mnogim aspektima, kao što je veličina propusnog opsega, ali unosi problem plavljenja mreže sa kontrolnim ruting saobraćajem radi otkrivanja i održavanje ruta čak i kada se ne koriste. U mrežama sa velikim propusnim opsegom veza ovo ne predstavlja ograničavajući problem, ali u scenarijima mreža niskih kapaciteta linkova ovaj problem može dovesti do zagušenja mreže. Novi algoritam će nastojati da uspostavi ravnotežu između ovih neslaganja.

Tokom izrade ove disertacije u okviru IETF radne grupe razvijao se zvaničan OLSR ruting protokol druge verzije (OLSRv2) [111]. Svakako, od interesa bi bilo uporediti efikasnost učinka ovog ruting protokola nakon što bude ozvaničen. Unapređenja unešena u novom OLSRv2 ruting protokolu eventualno bi mogla biti iskorišćena prilikom razvoja nove modifikovane verzije OLSR-BF ruting protokola.

Eksperimentalno istraživanje se fokusira na mobilnost čvorova, veličinu mreže i saobraćajno opterećenje mreže. Mobilnost se smatra najuticajnijim faktorom, te je stoga raspoređena u svim posmatranim scenarijima. FTP je jedini tip saobraćaja koji se koristi u svim scenarijima. Međutim, postoje i drugi faktori koji mogu imati značajan uticaj na performanse protokola.

Druge vrste saobraćaja poput e-mail komunikacije, baza podataka, HTTP saobraćaja i paketa govora (engl. *Voice Over IP*, *VoIP*) su mogućnost za uključanje u budućim istraživačkim scenarijima. Eksperimentalni scenariji bi se takođe mogli izvesti i potvrditi pomoću drugih alata za simulaciju, kao što je npr. GloMoSim.

Učinak četiri MANET ruting protokola ocenjen je pomoću OPNET Modeler i ns-3 mrežnih simulatora. Buduća studija nad ovim ruting protokolima mogla bi koristiti različite pokazatelje performansi. Širok obim studija može se uraditi nad istim ruting protokolima korišćenjem IPv6 adresiranja, jer je u ovom radu obuhvaćen samo IPv4 adresni opseg. IPv6 je najnovija verzija IP adresiranja i način rutiranja se uopšteno razlikuje od IPv4 protokola. Obimna studija može se uraditi razmatranjem drugih MANET ruting protokola i procenom njihovih performansi korišćenjem različitih metrika. Bilo bi interesantno kada bi se mogli predstaviti namerno izazvani neuspesi veza između čvorova tokom izvođenja eksperimenata. Istovetni eksperimenti koji su izvedeni u ovom radu mogli bi se ponoviti uz promenu parametara simulacije, uz korišćenje drugih modela mobilnosti, itd. Sve u svemu, rad na području proučavanja MANET mreža je uvek dugotrajan, ali zanimljiv.

11. LITERATURA

- [1] Xiaoyan Hong, Kaixin Xu, Mario Gerla, *Scalable Routing Protocols for Mobile Ad Hoc Networks*, IEEE Network, vol. 16, issue 4, pp. 11-21, July/August 2002.
- [2] Imrich Chlamtac, Marco Contib, Jennifer J.-N. Liuc, *Mobile Ad Hoc Networking: Imperatives and Challenges*, Journal of Ad Hoc Networks, vol. 1, issue 1, pp. 13-64, 2003.
- [3] R. Ramanathan, J. Redi, *A Brief Overview of Ad Hoc Networks: Challenges and Directions*, IEEE Communications Magazine, vol. 40, issue 5, pp. 20-22, 2002.
- [4] Q. Li, D. Rus, *Sending Messages to Mobile Users in Disconnected Ad-hoc Wireless Networks*, Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom), ACM, New York, pp. 44-55, 2000.
- [5] E. M. Royer, C.-K. Toh, *A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks*, IEEE Communications Magazine, vol. 6, issue 2, pp. 46-55, 1999.
- [6] C. Shen, C. Srisathapornphat, C. Jaikaeo, *An Adaptive Management Architecture for Ad Hoc Networks*, IEEE Communications Magazine, vol. 41, issue 2, pp. 108-115, 2003.
- [7] Vinh Pham, Ph.D. Thesis, *Performing Gateway Load Balancing in MANETs*, Norwegian University of Science and Technology, February 2012.
- [8] C. E. Perkins, Ed, *Ad Hoc Networking*, Addison Wesley, 2001.
- [9] Ming-Shen Jian, *Routing Distribution and Selection in Mobile Wireless Networks*, VDM Verlag Dr. Müller, 2009.
- [10] M. S. Jian, P. L. Wu, C. N. Lee, *On-demand Flow Regulated Routing for Ad-hoc Wireless Networks*, Wireless Communication and Mobile Computing, vol. 6, no. 4, pp. 401-412, 2006.
- [11] D. Turgut, B. Turgut, S. K. Das, R. Elmasri, *Balancing Loads in Mobile Ad Hoc Networks*, Proceedings of the 10th International Conference on Telecommunications 2003, vol. 1, pp. 490-495, February 2003.
- [12] Tao Lin, Ph.D. Thesis, *Mobile Ad-hoc Network Routing Protocols: Methodologies and Applications*, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, March 2004.
- [13] Hongbo Zhou, *A Survey on Routing Protocols in MANETs*, Michigan State University, Technical Report: MSU-CSE-03-08, March 2003.
- [14] Saud Rugeish Alotaibi, Ph.D. Thesis, *Stability of Secure Routing Protocol in Ad hoc Wireless Network*, Faculty of Technology, De Montfort University, United Kingdom, England, 2010.

- [15] Palaniappan Annamalai, *Comparative Performance Study of Standardized Ad-Hoc Routing Protocols and OSPF-MCDS*, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, October 2005.
- [16] Reza Qarehbaghi, *Performance Analysis and Comparison of Three Wireless Ad-Hoc Network Routing Protocols*, School of Engineering Science, Simon Fraser University, 2010.
- [17] Md. Anisur Rahman, Md. Shohidul Islam, Alex Talevski, *Performance Measurement of Various Routing Protocols in Ad-hoc Network*, Proceedings of the International MultiConference of Engineers and Computer Scientists, pp. 321-323, 2009.
- [18] Patrick Sondi, Dhavy Gantsou, Sylvain Lecomte, *Performance Evaluation of Multimedia Applications Over an OLSR-Based Mobile Ad Hoc Network Using OPNET*, Proceedings of the 12th International Conference on Computer Modelling and Simulation, pp. 567-572, 2010.
- [19] Murizah Kassim, Ruhani Ab. Rahman, Roihan Mustapha, *Mobile Ad Hoc Network (MANET) Routing Protocols Comparison for Wireless Sensor Network*, IEEE International Conference on System Engineering and Technology (ICSET), pp. 148-152, June 2011.
- [20] Mostafa Fazeli, Hasan Vaziri, *Assessment Of Throughput Performance Under OPNET Modeler Simulation Tools in Mobile Ad Hoc Networks (MANETs)*, International Conference on Computational Intelligence, Communication Systems and Networks (CICSYN), pp. 328-331, 2011.
- [21] Syed Iftikhar Hussain Shah, Syed Hassan Shaheed, M.Sc. Thesis, *Performance Evaluation of MANET Routing Protocols*, Blekinge Institute of Technology, Sweden, February 2011.
- [22] Mladen Marinović, Seminarski rad, Mreže računala, *Teorija Grafova*, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva.
- [23] B. Bollobás, *Modern Graph Theory*, Springer-Verlag New York Inc., 1998.
- [24] R. J. Wilson, *Introduction to Graph Theory*, Longman Group Ltd., 1999.
- [25] T. H. Cormen, C. E. Leiserson, R. L. Rivest, *Introduction to Algorithms*, MIT Press, 1999.
- [26] R. Sedgewick, *Algorithms in C*, Addison-Wesley Publishing Company Inc., 1996.
- [27] MANET working group, [Online] <http://datatracker.ietf.org/wg/manet/charter>, Posećeno 11.12.2012.
- [28] Pai-Hsiang Hsiao, Adon Hwang, H. T. Kung, Dario Vlah, *Load-Balancing Routing for Wireless Access Networks*, Proceeding of IEEE Infocom, pp. 986-995, 2001.
- [29] T. Clausen, P. Jacquet, *Optimized Link State Routing Protocol (OLSR)*, IETF RFC 3626, October 2003.

- [30] S. Corson, J. Macker, *Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*, IETF RFC 2501, January 1999.
- [31] *Information Processing Systems – Open Systems Interconnection – Basic Reference Model*, ISO/Draft International Standard (DIS) 7498 (ISO/TC97/SC16 N890), February 1982, as revised by: *Changes to DIS 7498*, (ISO/TC97/SC16 N1226), June 1982.
- [32] *IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11-1997, 1997.
- [33] G. Anastasi, E. Borgia, M. Conti, E. Grego, *IEEE 802.11 Ad Hoc Networks: Performance Measurements*, Proceedings of the 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW), pp. 758-763, May 2003.
- [34] M. S. Gast, *802.11 Wireless Networks: The Definitive Guide*, 2nd Edition: O'Reilly, April 2005.
- [35] D.-C. Chang, T.-H. Shiu, *Digital GFSK Carrier Synchronization*, Proceedings of IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), pp. 1523-1526, December 2006.
- [36] C. Lo, E. Masry, L. B. Milstein, *Design and Analysis of a Fast Frequencyhopped DBPSK Communication System, Description and Hop Timing Tracking Loop Analysis*, Proceedings of IEEE Transactions on Communications, vol. 41, pp. 1552-1564, October 1993.
- [37] T. Ozugur, J. A. Copeland, M. Naghshineh, P. Kermani, *Next-Generation Indoor Infrared LANs: Issues and Approaches*, Proceedings of IEEE Personal Communications, vol. 6, pp. 6-19, December 1999.
- [38] T. Keller, L. Hanzo, *Orthogonal Frequency Division Multiplex Synchronization Techniques for Wireless Local Area Networks*, Proceedings of the 7th International Symposium of Personal Indoor and Mobile Radio Communications, pp. 963–967, October, 1996.
- [39] T.-D. Chiueh, S.-M. Li, *Trellis-Coded Complementary Code Keying for Highrate Wireless LAN Systems*, IEEE Communications Letters, vol. 5, pp. 191-193, May 2001.
- [40] J. Lorincz, D. Begusic, *Physical Layer Analysis of Emerging IEEE 802.11n WLAN Standard*, Proceedings of the 8th International Conference on Advanced Communication Technology (ICACT), vol. 1, pp. 189-194, February 2006.
- [41] D. Hongning, N. Kam-Wing, W. Min-You, *An Overview of MAC Protocols With Directional Antennas in Wireless Ad Hoc Networks*, Proceedings of the International Conference on Wireless and Mobile Communications (ICWMC), pp. 84-91, July 2006.

- [42] T. Fuhrmann, *Scalable Routing in Sensor Actuator Networks With Churn*, Proceedings of 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks (SECON), vol. 1, pp. 33-39, September 2006.
- [43] J. S. Ahn, P. B. Danzig, Z. Liu, L. Yan, *Evaluation of TCP Vegas: Emulation and Experiment*, Proceedings of the Conference on Applications, Technologies, Architectures and Protocols for Computer Communication (SIGCOMM), pp.185-195, 1995.
- [44] P. P.-K. Lam, S. C. Liew, *UDP-Liter: an Improved UDP Protocol for Real-Time Multimedia Applications Over Wireless Links*, Proceedings of 1st International Symposium on Wireless Communication Systems, pp. 314-318, September 2004.
- [45] A. Ahuja, S. Agarwal, J. P. Singh, R. Shorey, *Performance of TCP Over Different Routing Protocols in Mobile Ad-Hoc Networks*, Proceedings of IEEE Vehicular Technology Conference (VTC), pp. 2315-2319, May 2000.
- [46] Stylianos Papanastasiou, Ph.D. Thesis, *Investigating TCP Performance in Mobile Ad Hoc Networks*, Faculty of Information and Mathematical Sciences, University of Glasgow, October 2006.
- [47] S. Floyd, T. Henderson, A. Gurtov, *The NewReno Modification to TCP's Fast Recovery Algorithm*, IETF RFC 3782, April 2004.
- [48] M. Mathis, J. Mahdavi, S. Floyd, A. Romanow, *TCP Selective Acknowledgment Options*, IETF RFC 2018, October 1996.
- [49] J. Border, M. Kojo, J. Griner, G. Montenegro, Z. Shelby, *Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations*, IETF RFC 3135, June 2001.
- [50] M. Conti, G. Maselli, G. Turi, S. Giordano, *Cross-Layering in Mobile Ad Hoc Network Design*, IEEE Computer Society, vol. 37, issue 2, pp. 48-51, 2004.
- [51] K. Chen, S. H. Shah, K. Nahrstedt, *Cross-Layer Design for Data Accessibility in Mobile Ad-hoc networks*, Wireless Personal Communications, vol. 21, pp. 49-76, 2002.
- [52] C. S. R. Murphy, B. S. Manoj, *Ad Hoc Wireless Networks: Architectures and Protocols*, Prentice Hall PTR, New Jersey, May 2004.
- [53] M. Alchaita, M. Al-Akaidi, J. Ivins, *New On-demand Routing Approaches for Ad Hoc Networks*, Proceedings of the 6th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, June 2005.
- [54] Prashant Dewan, Partha Dasgupta, *Trusting Routers and Relays in Ad hoc Networks*, Proceedings of the International Conference on Parallel Processing Workshops (ICPPW), pp. 351-358, October 2003.
- [55] Mohamed G. Gouda, Eunjin Jung, *Certificate Dispersal in Ad-Hoc Networks*, Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS), pp. 616-623, 2004.

- [56] Krishna Paul, Romit Roy Choudhuri, S. Bandyopadhyay, *Survivability Analysis of Ad Hoc Network Architecture*, Proceedings of the European Commission International Workshop on Mobile and Wireless Communication, vol. 1818, pp. 31-46, 2000.
- [57] Xing Fei, Wang Wenye, *Understanding Dynamic Denial of Service Attacks in Mobile Ad Hoc Networks*, IEEE Military Communications Conference (MILCOM), pp. 1-7, October 2006.
- [58] Xin Wang, *Mobile Ad-Hoc Networks: Protocol Design*, InTech, 2011.
- [59] X. Zou, B. Ramamurthy, S. Magliveras, *Routing Techniques in Wireless Ad Hoc Networks – Classification and Comparison*, Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI), Orlando, Florida, USA, July 2002.
- [60] Jameela Al-Jaroodi, *Security Issues In Wireless Mobile Ad Hoc Networks (MANET)*, Technical Report TR02-10-07, University of Nebraska-Lincoln, USA, 2002.
- [61] H. Dhillon, H. Q. Ngo, *CQMP: A Mesh-based Multicast Routing Protocol With Consolidated Query Packets*, Proceedings of the IEEE Wireless Communications & Networking Conference (WCNC), New Orleans, Louisiana, USA, March 2005.
- [62] V. D. Park, M. S. Corson, *A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks*, Proceedings of the IEEE Conference on Computer Communications (INFOCOM), Kobe, Japan, pp.1405-1413, April 1997.
- [63] S. Murthy, J. J. Garcia-Luna-Aceves, *An Efficient Routing Protocol for Wireless Networks*, ACM/Baltzer Mobile Networks and Applications, Special Issue on Routing in Mobile Communications Networks, vol. 1, no. 2, October 1996.
- [64] C.-C. Chiang, H. K. Wu, W. Liu, M. Gerla, *Routing in Clustered Multihop Mobile Wireless Networks With Fading Channel*, Proceedings of the IEEE Singapore International Conference on Networks (SICON), Singapore, April 1997.
- [65] D. B. Johnson, D. A. Maltz, *Dynamic Source Routing in Ad-Hoc Wireless Networks*, Mobile Computing, Kluwer Academic Publishers, 1996.
- [66] M. Abolhasan, B. Hagelstein, J. C.-P. Wang, *Real-World Performance of Current Proactive Multi-Hop Mesh Protocols*, IEEE APCC, Shanghai, China, October 2009.
- [67] A. Iwata, C. Chiang, G. Pei, M. Gerla, T. Chen, *Scalable Routing Strategies for Ad Hoc Wireless Networks*, IEEE Journal on Selected Areas in Communications, Special Issue on Ad Hoc Networks, vol. 17, no. 8, pp. 1369-1379, 1999.
- [68] Andreas Tønnesen, Andreas Hafslund, Øivind Kure, *The UniK – OLSR Plugin Library*, OLSR Interop and Workshop, San Diego, California, USA, August 2004.
- [69] C. Perkins, E. Belding-Royer, S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, IETF RFC 3561, July 2003.
- [70] N. Nikaein, C. Bonnet, N. Nikaein, *Hybrid Ad Hoc Routing Protocol – HARP*, Proceeding of International Symposium on Telecommunications (IST), 2001.

- [71] D. Johnson, D. Maltz, Y-C. Hu, J. Jetcheva, *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)*, draft-ietf-manet-dsr-09, IETF Internet Draft, April 2003.
- [72] Yih-Chun Hu, David B. Johnson, *Implicit Source Routes for On-Demand Ad Hoc Network Routing*, Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc), pp. 1-10, ACM, Long Beach, CA, October, 2001.
- [73] C.-K. Toh, *A Novel Distributed Routing Protocol to Support Ad-Hoc Mobile Computing*, Proceedings of the IEEE 15th Annual International Phoenix Conference on Computers and Communications, pp. 480-486, March 1996.
- [74] C.-K. Toh, *Long-Lived Ad-Hoc Routing Based on the Concept of Associativity*, IETF Internet Draft, March 1999.
- [75] R. Dube, C. D. Rais, K-Y.Wang, S. K. Tripathi, *Signal Stability-Based Adaptive Routing (SSA) for Ad-Hoc Mobile Networks*, IEEE Personal Communications Journal, pp. 36-45, February 1997.
- [76] I. Chakeres, C. Perkins, *Dynamic MANET On-Demand (DYMO) Routing*, draft-ietf-manet-dymo-12, Mobile Ad Hoc Networks Working Group Internet Draft, February 2008.
- [77] V. Park, S. Corson, *Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification*, draft-ietf-manet-tora-spec-03, IETF Internet Draft, November 2000.
- [78] Z. Haas, M. Pearlman, P. Samar, *The Zone Routing Protocol (ZRP) for Ad Hoc Networks*, draft-ietf-manet-zone-zrp-04, IETF Internet Draft, July 2002.
- [79] C. Santivanez, R. Ramanathan, *Hazy Sighted Link State Routing Protocol (HSLS)*, BBN Technical Memorandum, no. 1301, August 2001.
- [80] A. O. Lim, X. Wang, Y. Kado, B. Zhang, *A Hybrid Centralized Routing Protocol for 802.11s WMNs*, Journal of Mobile Networks and Applications, vol. 13, issue 1-2, pp. 117-131, April 2008.
- [81] M. Bani-Yassein, M. Ould-Khaoua, L. M. Mackenzie, S. Papanastasiou, *Performance Analysis of Adjusted Probabilistic Broadcasting in Mobile Ad Hoc Networks*, International Journal of Wireless Information Networks, pp. 1-14, 2006.
- [82] W. Peng, X. C. Lu, *On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks*, Proceedings of the Annual Workshop on Mobile and Ad Hoc Networking and Computing (MobiHoc), Boston, Massachusetts, USA, pp. 129-130, 2000.
- [83] S. Ni, Y. Tseng, Y. Chen, J. Sheu., *The Broadcast Storm Problem in a Mobile Ad Hoc Networks*, Proceeding of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), pp. 151-162, 1999.
- [84] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, S. Suri, *Towards Realistic Mobility Models for Mobile Ad Hoc Networks*, Proceedings of the 9th ACM International Conference on Mobile Computing and Networking (MobiCom), San Diego, California, USA, pp. 217-229, 2003.

- [85] OPNET Modeler Network Simulator, [Online] <http://www.opnet.com>, Posećeno 09.11.2012.
- [86] Network Simulator ns-2, [Online] <http://www.isi.edu/nsnam/ns>, Posećeno 17.12.2012.
- [87] Network Simulator ns-3, [Online] <http://www.nsnam.org>, Posećeno 28.05.2013.
- [88] GloMoSim – Global Mobile Information Systems Simulation Library, [Online] <http://pcl.cs.ucla.edu/projects/glomosim>, Posećeno 18.01.2013.
- [89] Anipakala Suresh, *Performance Analysis of Ad Hoc On-Demand Distance Vector routing (AODV) Using OPNET Simulator*, University of Bremen, Germany, April 2005.
- [90] Jiwa Abdullah, Ph.D. Thesis, *QoS Routing for Mobile Ad Hoc Networks Using Genetic Algorithm*, Department of Electronic and Electrical Engineering of Loughborough University, UK, 2007.
- [91] Sajjad Ali, Asad Ali, *Performance Analysis of AODV, DSR and OLSR in MANET*, Department of Electrical Engineering With Emphasis on Telecommunication Blekinge Institute of Technology, Sweden, 2009.
- [92] Uyen Trang Nguyen, Xing Xiong, *Rate-adaptive Multicast in Mobile Ad-hoc Networks*, IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob), vol. 3, pp. 352-360, August 2005.
- [93] Suhaimi Bin Abd Latif, M. A. Rashid, F. Alam, *Profiling Delay and Throughput Characteristics of Interactive Multimedia Traffic over WLANs using OPNET*, 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW), vol. 2, pp. 929-933, May 2007.
- [94] Julian Hsu, Sameer Bhatia, Ken Tang, Rajive Bagrodia, Michael J. Acrkhe, *Performance of Mobile Ad Hoc Networking Routing Protocols in Large Scale Scenarios*, IEEE Military Communications Conference (MILCOM), vol. 1, pp. 21-27, 2004.
- [95] Sunil Kumar, *Reactive and Proactive Routing Protocols for Wireless Mesh Network using Multimedia Streaming*, International Conference on Recent Advances and Future Trends in Information Technology (iRAFIT), no. 7, pp. 13-17, April 2012.
- [96] Vahid Ayatollahi Tafti, Abolfazl Gandomi, *Performance of QoS Parameters in MANET Application Traffics in Large Scale Scenarios*, World Academy of Science, Engineering and Technology, vol. 72, February 2011.
- [97] T. Camp, J. Boleng, V. Davies, *A Survey of Mobility Models for Ad Hoc Network Research*, Wireless Communication and Mobile Computing (WCMC), Special Issue on Mobile Ad Hoc Networking Research, Trends and Applications, vol. 2, issue 5, pp 483-502, 2002.
- [98] C. Bettstetter, C. Wagner, *The Spatial Node Distribution of the Random Waypoint Mobility Model*, Proceedings of German Workshop on Mobile Ad Hoc Networks (WMAN), Ulm, Germany, pp. 48-51, 2002.

- [99] G. Resta, P. Santi, *An Analysis of the Node Spatial Distribution of the Random Waypoint Model for Ad Hoc Networks*, Proceedings of ACM Workshop on Principles of Mobile Computing (POMC), pp. 44-50, 2002.
- [100] E. Hyytiä, P. Lassila, J. Virtamo, *Spatial Node Distribution of the Random Waypoint Mobility Model with Applications*, IEEE Transactions on Mobile Computing, vol. 6, pp. 680-694, June 2006.
- [101] E. Hyytiä, P. Lassila, J. Virtamo, *A Markovian Waypoint Mobility Model with Application to Hotspot Modeling*, IEEE International Conference on Communications (ICC), vol. 3, pp. 979-986, June 2006.
- [102] A. Zaballo, A. Vallejo, G. Corral, J. Abella, *Ad-Hoc Routing Performance Study Using OPNET Modeler*, OPNETWORK, Washington DC, USA, August 2006.
- [103] G. Zhou, T. He, S. Krishnamurthy, J. A. Stankovic, *Impact of Radio Irregularity on Wireless Sensor Networks*, Proceedings of the 2nd International Conference on Mobile Systems, Applications and Services (MobiSys), New York, USA, pp. 125-138, 2004.
- [104] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, C. Elliott, *Experimental Evaluation of Wireless Simulation Assumptions*, Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), New York, USA, pp. 78-82, 2004.
- [105] A. Al-Maashri, M. Ould-Khaoua, *Performance Analysis of MANET Routing Protocols in the Presence of Self-Similar Traffic*, Proceedings of the 31st IEEE Conference on Local Computer Networks, pp. 801-807, November 2006.
- [106] H. Dubois-Ferriere, M. Grossglauser, M. Vetterli, *Age Matters: Efficient Route Discovery in Mobile Ad Hoc Networks Using Encounter Ages*, Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), New York, USA, pp. 257-266, 2003.
- [107] J. Li, M. Prasant, *PANDA: A Novel Mechanism for Flooding Based Route Discovery in Ad Hoc Networks*, Wireless Networks, vol. 12, pp. 771-787, 2006.
- [108] R. Groenevelt, P. Nain, G. Koole, *The Message Delay in Mobile Ad Hoc Networks*, Journal of Performance Evaluation, vol. 62, pp. 210-228, October 2005.
- [109] P. Jacquet, A. M. Naimi, G. Rodolakis, *Routing on Asymptotic Delays in IEEE 802.11 Wireless Ad Hoc Networks*, IEEE WiOpt, Riva Del Garda, Italy, 2005.
- [110] L. Xin, Q. Zhen, H. Bo, J. Yuehui, C. Shanzhi, *Theoretical Analysis on the Performance of Two Broadcast Schemes in Mobile Ad Hoc Networks*, Proceedings of the 6th International Conference on ITS Telecommunications, pp. 686-690, June 2006.
- [111] T. Clausen, C. Dearlove, P. Jacquet, U. Herberg, *The Optimized Link State Routing Protocol version 2*, draft-ietf-manet-olsrv2-17, IETF Internet Draft, October 2012.

PRILOG

Izvorni kod Belman-Fordovog algoritma u C programskom jeziku

```
#define _CRT_SECURE_NO_DEPRECATED
#include <stdio.h>

/* Struktura koja predstavlja težinsku granu (vezu) u mreži. */
typedef struct {
    int u, v, w;
} Edge;

int x = 1; /* Polazni čvor od koga se računa najkraće rastojanje ka
            ostalim čvorovima u mreži.
            Ova vrednost je uvek jednaka 1, jer se izvršava na lokalnom
            čvoru koji se uvek predstavlja kao čvor broj 1. */

int n; /* Broj čvorova u mreži. */
int e; /* Broj grana (veza) u mreži. */
Edge edges[16384]; /* Dovoljno veliko za  $n \leq 2^7 = 128$  čvorova. */
int d[32]; /*  $d[i]$  je minimalno rastojanje od čvora s do čvora i */
int g; /* Parametar postojanja negativne povratne petlje u mreži:
        0 – petlja ne postoji >> nastavi sa izvršavanjem
        algoritma
        1 – petlja postoji >> nemoguće je izračunati najkraće
        udaljenosti! */

int f; /* Parametar postojanja matrice povezanosti čvorova:
        1 – matrica postoji >> nastavi sa izvršavanjem
        algoritma
        0 – matrica ne postoji >> nemoguće je izračunati
        najkraće udaljenosti! */

#define INFINITY 100000

/* Podfunkcija za štampanje rezultata algoritma unutar izlaznog fajla:
najkrace_udaljenosti_cvorova.txt. */
void printDist() {
    int i;

    FILE *fout = fopen("najkrace_udaljenosti_cvorova.txt", "w");

    fprintf(fout, "Postojanje matrice povezanosti cvorova: %d\n\n", f);
    fprintf(fout, "Postojanje negativne povratne petlje: %d\n\n", g);
    fprintf(fout, "Najkrace rastojanje od polaznog cvora %d:\n", x);
    for (i = 0; i < n; ++i)
        fprintf(fout, "do cvora %d:\t", i + 1);
    fprintf(fout, "\n");
}
```

```

for (i = 0; i < n; ++i)
    fprintf(fout, "%d\t\t", d[i]);
fprintf(fout, "\n\n");

fclose(fout);
}

/* Osnovna funkcija koja pronalazi najkraće udaljenosti od izvornog ka svim ostalim
čvorovima unutar mreže korišćenjem Belman-Fordovog algoritma.
Ova funkcija je sposobna da otkrije postojanje negativnih povratnih petlji unutar mreže. */
void belman_ford(int s) {
    int i, j;

    /* Korak 1: Inicijalizacija udaljenosti od izvornog ka svim ostalim čvorovima unutar
mreže ka beskonačnosti. */
    for (i = 0; i < n; ++i)
        d[i] = INFINITY;
    d[s] = 0;

    /* Korak 2: Relaksacija svih grana (veza) n - 1 puta.
Jednostavna najkraća putanja od izvornog ka bilo kom drugom čvoru unutar mreže
može imati najviše n - 1 grana (veza). */
    for (i = 0; i < n - 1; ++i)
        for (j = 0; j < e; ++j)
            if (d[edges[j].u] + edges[j].w < d[edges[j].v])
                d[edges[j].v] = d[edges[j].u] + edges[j].w;

    /* Korak 3: Provera postojanja negativnih povratnih petlji unutar mreže.
Korak 2 garantuje pronalaženje najkraćih rastojanja, ako unutar mreže ne postoje
negativne povratne petlje.
Ukoliko se u koraku 3 dobije kraći put, onda u mreži postoji negativna povratna
petlja. */
    for (i = 0; i < e; ++i)
        if (d[edges[i].u] + edges[i].w < d[edges[i].v]) {
            g = 1;

            FILE *fout = fopen("najkrace_udaljenosti_cvorova.txt", "w");

            fprintf(fout, "Postojanje matrice povezanosti cvorova: %d\n\n", f);
            fprintf(fout, "Postojanje negativne povratne petlje: %d\n\n", g);
            fprintf(fout, "Greska: Graf sadrzi negativnu povratnu petlju!\n\n");

            fclose(fout);
            break;
        }
        else g = 0;
}

int main(int argc, char *argv[]) {

```

```
int i, j;
int w;

FILE *fin = NULL;
fin = fopen("matrica_povezanosti_cvorova.txt", "r");

/* U ovom koraku proverava se postojanje matrice povezanosti čvorova. */
if(fin == NULL) {
    f = 0;

    FILE *fout = fopen("najkrace_udaljenosti_cvorova.txt", "w");

    fprintf(fout, "Postojanje matrice povezanosti cvorova: %d\n\n", f);
    fprintf(fout, "Postojanje negativne povratne petlje: %d\n\n", g);
    fprintf(fout, "Greska: Ne postoji matrica povezanosti cvorova!\n\n");

    fclose(fout);
    return 0;
}
else f = 1;

fscanf(fin, "%d", &n);
e = 0;

for (i = 0; i < n; ++i)
    for (j = 0; j < n; ++j) {
        fscanf(fin, "%d", &w);
        if (w != 0) {
            edges[e].u = i;
            edges[e].v = j;
            edges[e].w = w;
            ++e;
        }
    }
fclose(fin);

/* Poziva se Belman-Fordov algoritam za proračun najkraćih rastojanja od izvornog
čvora na kome se ovaj algoritam izvršava.
Ovaj čvor je uvek čvor broj 1.*/
belman_ford(x-1);

/* Ako u datoj mreži postoji matrica povezanosti čvorova i ako ne postoji negativna
povratna petlja, dobija se rezultat izvršenja Belman-Fordovog algoritma za proračun
najkraćih rastojanja od datog polaznog čvora broj 1 do svih ostalih čvorova unutar
mreže. */
if (f == 1 && g == 0)
    printDist();

return 0;
}
```