

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

MAGISTARSKI RAD

**HEURISTIČKI ALGORITMI ZA PLANIRANJE
PRSTENASTIH TELEKOMUNIKACIJSKIH MREŽA**

Matija Mikac

Zagreb, 2004.

Magistarski rad je izrađen na Zavodu za telekomunikacije Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Mentor: prof.dr.sc. Mladen Kos

Magistarski rad ima 119 stranica i sadrži 107 slika i 47 tablica.

Povjerenstvo za ocjenu u sastavu:

1. prof.dr.sc. Damir Kalpić - predsjednik
2. prof.dr.sc. Mladen Kos - mentor
3. prof.dr.sc. Luka Neralić – Ekonomski fakultet Zagreb

Povjerenstvo za obranu u sastavu:

1. prof.dr.sc. Damir Kalpić - predsjednik
2. prof.dr.sc. Mladen Kos - mentor
3. prof.dr.sc. Luka Neralić – Ekonomski fakultet Zagreb
4. doc.dr.sc. Alen Bažant – zamjenik

Datum obrane: 30. siječanj 2004.

UVOD	1
1. PREGLED POSTUPKA PLANIRANJA PRSTENASTIH MREŽA.....	2
1.1. PRSTENASTE MREŽE	2
1.2. POSTUPAK PLANIRANJA TELEKOMUNIKACIJSKIH MREŽA	2
1.2.1. <i>Ulazni podaci.....</i>	3
1.2.2. <i>Topologija mreže.....</i>	4
1.2.3. <i>Topologija prstena</i>	7
1.2.4. <i>Usmjeravanje prometnih zahtjeva.....</i>	7
1.2.5. <i>Analiza topologije i pronaalaenje prstenova.....</i>	10
1.2.6. <i>Usmjeravanje prometnih zahtjeva kroz prstenove.....</i>	11
1.2.7. <i>Rezultati postupka planiranja</i>	12
1.3. PRISTUPI PLANIRANJU PRSTENASTIH MREŽA	13
2. JEDNOSTAVAN PRSTEN.....	16
2.1. STRUKTURA PRSTENA.....	16
2.2. OPĆENITI MODEL PRSTENA	17
2.3. DIMENZIONIRANJE JEDNOSTAVNOG PRSTENA	21
2.3.1. <i>Dimenzioniranje s mogućnošću dijeljenja prometnih zahtjeva</i>	24
2.3.2. <i>Dimenzioniranje bez mogućnosti dijeljenja prometnih zahtjeva</i>	26
2.4. ZAŠTITNI MEHANIZMI U PRSTENU	29
2.4.1. <i>Iskorištenost kapaciteta prstena pri zaštitnom preusmjeravanju zahtjeva.....</i>	30
2.5. PRSTENASTE MREŽNE STRUKTURE U PRIMJENI	31
2.5.1. <i>Sklopoljje</i>	31
2.5.2. <i>Izvedba jednostavnog prstena.....</i>	34
2.5.3. <i>Prstenovi sa zaštitom – samooporavljivi prstenovi</i>	35
2.5.4. <i>Princip korištenja prstenova u WDM/DWDM okruženju</i>	40
2.6. REZULTATI POSTUPKA DIMENZIONIRANJA JEDNOSTAVNOG PRSTENA	41
3. POSTUPCI PLANIRANJA MREŽA SASTAVLJENIH OD VIŠE PRSTENOVA	44
3.1. OPĆENITO O POSTUPKU PLANIRANJA PRSTENASTIH MREŽA	44
3.2. METODE ZA PLANIRANJE PRSTENASTIH MREŽA	45
3.2.1. <i>Provjera stupnja povezanosti u mreži.....</i>	45
3.2.2. <i>Smanjenje složenosti mrežne topologije</i>	47
3.2.3. <i>Određivanje prstenova u mreži</i>	48
3.2.4. <i>Usmjeravanje prometnih zahtjeva.....</i>	54
3.2.5. <i>Prekrivanje mreže prstenovima.....</i>	59
3.2.6. <i>Matematički model prekrivanja mreže prstenovima.....</i>	69
3.3. ALTERNATIVNI PRISTUPI RJEŠAVANJU PROBLEMA PREKRIVANJA MREŽE PRSTENOVIMA	71
3.4. POSEBNI MODELI PRSTENASTIH MREŽA.....	71
3.4.1. <i>Hijerarhijske prstenaste mreže</i>	71
3.4.2. <i>Koncentrirane prstenaste mreže.....</i>	72
3.5. ORGANIZACIJA POSTUPKA PLANIRANJA PRSTENASTIH MREŽA.....	73
4. RAZVIJENI ALATI ZA PLANIRANJE PRSTENASTIH MREŽA.....	75
4.1. ALAT ZA DIMENZIONIRANJE PRSTENOVA - MRINGSOLVER.....	75
4.1.1. <i>Specijalizirani model koncentriranih prstenastih mreža</i>	76
4.2. SPECIJALIZIRANI ALAT ZA DIMENZIONIRANJE SDH PRSTENOVA - MSDHRINGSOLVER	77
4.2.1. <i>Usmjeravanje nezaštićenih zahtjeva.....</i>	79
4.2.2. <i>Dijeljenje zahtjeva prema razinama.....</i>	82
4.3. ALAT ZA ANALIZU I PLANIRANJE PRSTENASTIH MREŽA - MRINGDESIGNER.....	84
4.3.1. <i>Određivanje prstenova u mreži</i>	84
4.3.2. <i>Usmjeravanje prometnih zahtjeva</i>	85
4.3.3. <i>Prekrivanje mreže prstenovima</i>	86
5. STUDIJSKI PRIMJERI PLANIRANJA PRSTENASTIH MREŽA.....	88
5.1. IZVEDBA POSTUPAKA KORIŠTENIH ZA DOBIVANJE REZULTATA	88

5.2.	OPIS TESTNIH MREŽA I STUDIJSKIH PRIMJERA.....	91
5.3.	REZULTATI ODREĐIVANJA PRSTENOVA U MREŽAMA	92
5.4.	REZULTATI POSTUPKA PLANIRANJA.....	93
5.4.1.	<i>Testna mreža 1 - studijski primjer 1</i>	93
5.4.2.	<i>Testna mreža 1 - studijski primjer 2</i>	102
5.4.3.	<i>Testna mreža 1 – rezultati matematičkog programa</i>	104
5.4.4.	<i>Testna mreža 1 - zaključak.....</i>	105
5.4.5.	<i>Testna mreža 2 – studijski primjeri 3 i 4.....</i>	105
5.4.6.	<i>Testna mreža 2 – rezultati matematičkog programa</i>	112
5.4.7.	<i>Testna mreža 2 – zaključak.....</i>	112
5.4.8.	<i>Testna mreža 3 - studijski primjer 5.....</i>	113
5.5.	ZAKLJUČAK O REZULTATIMA PLANIRANJA.....	116
	ZAKLJUČAK	117
	LITERATURA	118

Uvod

Tehnologije prstenastih telekomunikacijske mreže posjeduju svojstvo da mogu jednostavnim i provjerjenim mehanizmima osigurati zaštitu prometa u slučaju kvara u mreži. Zbog toga su često korištene u temeljnim mrežama u kojima je sprečavanje gubitka informacija od osobite važnosti. Zbog karakteristične strukture prstenastih mreža, postupak njihova planiranja razlikuje se od standardnih postupaka planiranja telekomunikacijskih mreža. U postupku planiranja prstenastih mreža javlja se niz specifičnih problema koje je potrebno riješiti. Najčešće se radi o optimizacijskim problemima koje je teško riješiti egzaktno, pa se zbog toga za njihovo rješavanje koriste različiti heuristički algoritmi koji kao takvi ne mogu garantirati optimalnost rješenja.

Problemi koji se razmatraju u ovome radu uključuju problem dimenzioniranja jednostavnih prstenova, problem određivanja prstenova na zadanim topologijama mreža, te problem prekrivanja mrežne topologije prstenastim strukturama. Svi navedeni problemi su detaljno analizirani te su predloženi postupci rješavanja. Prema predloženim postupcima izvedeni su algoritmi za rješavanje problema.

Rad je podijeljen u pet poglavlja.

Prvo poglavlje daje općeniti pregled postupka planiranja telekomunikacijskih mreža, uz naglasak na planiranje prstenastih mreža.

U drugom poglavlju detaljno je analizirana struktura jednostavnog prstena, te su izneseni problemi koji se javljaju u postupku njegova optimalnog dimenzioniranja. Dan je opis mehanizama zaštite prometa u prstenu, uz naglasak na mogućnost primjene prstenaste mrežne strukture u praksi. Uključen je detaljan prikaz korištenja prstenova u SDH/SONET prijenosnim sustavima. Problem optimalnog dimenzioniranja prstena je matematički formuliran i riješen korištenjem matematičkog programiranja, a dan je i opis heurističkih i egzaktnih algoritama za rješavanje različitih problema dimenzioniranja prstena. Rezultati dobiveni tim algoritmima su uspoređeni s optimalnim rezultatima matematičkog programa.

U trećem se poglavlju detaljno opisuje postupak planiranja mreža sastavljenih od više prstenova. Objasnjene su metode koje se mogu koristiti u različitim fazama izvedbe postupka planiranja. Izdvojeni su problem određivanja prstenastih struktura u zadanoj mrežnoj topologiji, problem usmjeravanja prometnih zahtjeva u prstenastim mrežama, te problem određivanja prstenova koji prekrivaju mrežu. Za svaki od problema je opisan postupak rješavanja korištenjem heurističkih ili egzaktnih algoritama. Na kraju je predložen model organizacije postupka planiranja prstenastih mreža koji uključuje rješavanje svih spomenutih problema.

Rezultati dobiveni na testnim mrežama korištenjem postupka predloženog u trećem poglavlju opisani su u četvrtom poglavlju rada. Rezultati su detaljno proanalizirani, a odabrana rješenja prikazana su grafički radi lakše interpretacije.

U petom poglavlju su opisani alati razvijeni kao praktični dio rada. Opisani su alat za dimenzioniranje prstenova *mRingSolver*, specijalizirani alat za dimenzioniranje SDH prstenova *mSDHRingSolver* i alat za analizu i planiranje prstenastih mreža *mRingDesinger*. Svi alati su korišteni u izvedbi istraživačkih projekata te je njihova funkcionalnost potvrđena na primjerima stvarnih mreža.

1. Pregled postupka planiranja prstenastih mreža

1.1. Prstenaste mreže

Krajem osamdesetih i početkom devedesetih godina prošlog stoljeća dolazi do znatnog povećanja potrebe za prijenosom veće količine informacija u telekomunikacijskim mrežama. U to doba postojeće mreže nisu mogle zadovoljiti sve postavljene zahtjeve, te dolazi do uvođenja novih tehnologija u temeljne mreže – uvođe se prijenosni sustavi temeljeni na sinkronim digitalnim hijerarhijama koje postaju standard za prijenos informacija korištenjem optičkih medija. Osim promjena u strukturi temeljne mreže, povećanje prometa koji se prenosi mrežom uvjetovalo je i promjenu postupka planiranja mreža. Prije svega to se odnosilo na korištenje novih mrežnih topologija koje mogu osigurati višestruku povezanost u mreži. Višestrukom povezanošću takve topologije omogućuju uspostavu alternativnih puteva u mreži u slučaju kvarova. Naime, uvođenje optičkih prijenosnih medija omogućilo je povećanje kapaciteta veza – kroz svako optičko vlakno prenosio se promet čiji bi gubitak uslijed kvarova mogao uzrokovati velike finansijske gubitke i korisnicima i ponuđaćima telekomunikacijskih usluga. Zbog toga se sve više pažnje počelo pridavati razvoju mehanizama za zaštitu prometa u slučaju kvarova.

Jedna od posljedica uvođenja mehanizama zaštite jest povećanje kapaciteta u mreži. Od samih početaka uvođenja sinkronih digitalnih hijerarhija, standardizirani su mehanizmi automatske zaštite u slučaju kvara u mreži. Jedan od najprihvaćenijih, a ujedno i najjednostavnijih mehanizama zaštite temelji se na korištenju prstenastih struktura za lokalnu zaštitu prometa. Spajanjem jednostavnih prstenova omogućena je izgradnja prstenastih mreža koje su u potpunosti otporne na jednostrukе (i u određenim slučajevima višestruke) kvarove u mreži.

Promjene u strukturi mreže dovele su i do značajnih promjena u postupcima planiranja mreža. Iako naizgled jednostavna, struktura prstena može biti vrlo složena kad je u pitanju postupak oblikovanja i planiranja mreže. Da bi se minimizirala cijena mreže, nužno je kvalitetno isplanirati niz parametara – odrediti minimalan kapacitet prstena, optimalno usmjeravanje prometnih zahtjeva, načine povezivanja prstenova u mreži itd. Brojni su problemi koji se javljaju u postupku planiranja prstenastih mreža – grubo se dijele na probleme određivanja prstenova u mreži (*REP – Ring Enumeration Problem*), određivanje prstenova koji 'prekrivaju' ponudenu mrežnu topologiju (*RCP – Ring Covering Problem*), probleme usmjeravanja zahtjeva kroz mrežu i unutar prstenova i probleme dimenzioniranja prstenova (*RDP – Ring Dimensioning Problem*). Osim što svaki od problema može biti poprilično složen, svi su u pravilu međusobno povezani, što najčešće onemogućuje određivanje optimalnog rješenja u realnim mrežama. Na jednostavnijim primjerima mreža, koji zanemaruju niz praktičnih zahtjeva i problema, moguće je postići relativno dobre rezultate korištenjem matematičkog programiranja, dok određivanje kvalitetnih rješenja za stvarne mreže može biti složeno. Naime, pri planiranju realnih mreža pokušava se, koliko je to moguće, odvojeno rješavati pojedine probleme, te ih ovisno o dobivenim rezultatima prilagodavati s ciljem da se pronađe bolje rješenje.

Ovaj rad daje detaljan pregled postupaka u planiranju prstenastih mreža, te predlaže pristup planiranju korišten u praktičnim projektima. Predloženi pristup implementiran je u nekoliko alata razvijenih na Zavodu za telekomunikacije, te primijenjen na primjerima iz prakse. Iako se ne radi o kompletno automatiziranom rješenju za dizajn prstenastih mreža, praktični dio rada pokazao se vrlo korisnim u izvedbi istraživačkih projekata, te u primjeni na razvoju SDH mreža.

1.2. Postupak planiranja telekomunikacijskih mreža

Planiranje prstenastih mreža uključuje neke specifičnosti, no u principu se ne razlikuje od općenitog postupka planiranja telekomunikacijskih mreža.

Razmotrimo prvo slučaj u kojem se izgrađuje nova mreža. Dakle, ne postoji mrežna infrastruktura. Ulazni podaci za izgradnju nove mreže su moguće lokacije čvorova mreže, te prometni zahtjevi

među čvorovima. Lokacije čvorova mogu biti točno definirane ako je već odlučeno gdje će se čvorovi postavljati. Ukoliko se definiraju samo moguće lokacije čvorova može se postići dodatna optimizacija i smanjenje troškova izgradnje mreže. Naime, može se pokazati da neki čvorovi uopće ne trebaju biti postavljeni, već se promet koji bi na njih dolazio jednostavno uputi na neki drugi čvor (naravno, ukoliko je to finansijski isplativije).

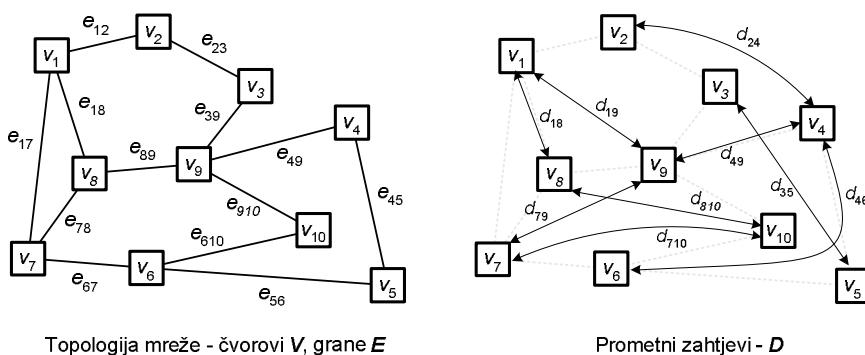
Izlazni podaci, dakle rezultati postupka planiranja, su mnogobrojni. Kao prvo, mora se odrediti prospojenost u mreži kako bi se u potpunosti definirala mrežna topologija – čvorovi i grane koje ih povezuju. Osim toga, potrebno je odrediti svojstva svakog pojedinog mrežnog elementa – za čvorove je potrebno odrediti kapacitete i zahtijevanu funkcionalnost čvora, za grane je potrebno odrediti prijenosne kapacitete, vrstu i količinu medija i sl. Ovisno o tome da li se izgrađuje mreža sa fiksnim ili dinamičkim usmjeravanjem, potrebno je i za sve prometne zahtjeve odrediti kako se usmjeravaju ili pak predložiti koje karakteristike moraju zadovoljavati čvorovi koji se koriste za dinamičko usmjeravanje. Ukoliko mreža mora osigurati zaštitu prometa u slučaju kvara, potrebno je detaljno definirati načine zaštite, alternativna usmjeravanja zahtjeva, dodatne kapacitete koji će omogućiti zaštitu itd.

Sličan postupak je i u redizajnu mreže koja je već izgrađena. Ulazni podaci u tom slučaju su postojeća topologija (lokacije čvorova i podaci o granama koje ih povezuju) i prometni zahtjevi. Izlazni podaci identični su izlaznim podacima u slučaju izgradnje nove mreže.

U nastavku su ukratko opisani različiti koraci u postupku dizajna telekomunikacijskih mreža (s naglaskom na mreže zasnovane na prstenastim zaštitnim strukturama).

1.2.1. Ulazni podaci

Ulazni podaci definiraju se ovisno o modelu mreže koji se koristi. Model korišten u ovom radu mrežu N definira kao trojku $N = (\mathbf{V}, \mathbf{E}, \mathbf{D})$. Skup čvorova \mathbf{V} sadrži čvorove v_i . Svaki čvor v_i definiran je skupom svojstava koja ovise o primjeni modela. Svojstva mogu biti lokacija čvora, klasifikacija čvora, cijena, razina povezanosti, te bilo kakve druge karakteristične informacije. Skup grana \mathbf{E} sadrži grane e_{ij} , gdje su indeksima i i j naznačeni rubni čvorovi grane. Drugim riječima, grana e_{ij} započinje čvorom v_i a završava čvorom v_j iz skupa čvorova \mathbf{V} . Svaka grana e_{ij} definirana je svojstvima – duljinom, cijenom, kapacitetom i slično. Skup prometnih zahtjeva \mathbf{D} sadrži prometne zahtjeve d_{ij} . Vrijednost pridružena zahtjevu d_{ij} definira prometni zahtjev između čvora v_i i čvora v_j . Prikaz modela izведен je za primjer mreže na slici 1.1.



Slika 1.1: Mreža - model

U primjeru mreže sa slike definirani su sljedeći skupovi:

$$\begin{aligned} \mathbf{V} &= \{ v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10} \} & N_V &= 10 \\ \mathbf{E} &= \{ e_{12}, e_{23}, e_{39}, e_{49}, e_{45}, e_{56}, e_{67}, e_{78}, e_{89}, e_{910}, e_{17}, e_{18}, e_{610} \} & N_E &= 13 \\ \mathbf{D} &= \{ d_{18}, d_{19}, d_{24}, d_{35}, d_{46}, d_{49}, d_{79}, d_{710}, d_{810} \} & N_D &= 9 \end{aligned}$$

Ulagni podaci pri planiranju mreže uvijek uključuju skup čvorova V i skup prometnih zahtjeva D . U skupu čvorova mogu biti moguće lokacije čvorova ili lokacije za koje je već određeno da će se koristiti kao čvorista – ovisno o tome u postupku planiranja primjenjuju se različite metode za selekciju čvorova. Prometni zahtjevi definiraju se s kraja na kraj, između para čvorova iz skupa V .

U određenim slučajevima definirana je kompletan topologija mreže – uz skup čvorova V , definiran je i skup grana E . U tim slučajevima najčešće se zahtjeva planiranje nadogradnje ili poboljšanja svojstava već izgrađene mreže.

1.2.2. Topologija mreže

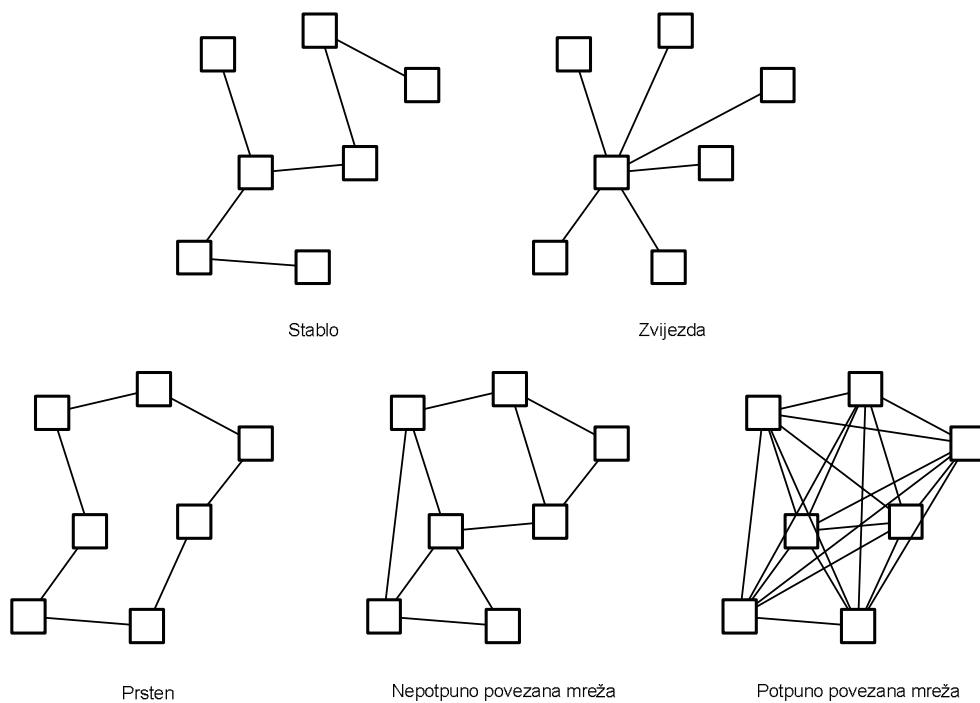
Određivanje topologije mreže uključuje nekoliko operacija. Ukoliko je potrebno, donosi se odluka o tome koji čvorovi će biti uključeni u konačno rješenje. U tom slučaju se iz skupa čvorova V određuje skup inicijalnih čvorova u mreži V_0 . U nekim slučajevima V_0 je zadan, pa se ta odluka preskače.

Slijedeći korak je određivanje prospojenosti mreže. Potrebno je odrediti skup grana E koji povezuje sve čvorove V_0 u mreži. Ovisno o zahtjevima i ograničenjima koja se postavljaju (npr. maksimalni broj grana koje mogu ulaziti i izlaziti iz nekog čvora, zahtijevani stupanj povezanosti mreže itd.) koriste se različite metode za određivanje skupa inicijalnih grana E . Skup čvorova V_0 i skup grana E određuju topologiju mreže $N = (V_0, E)$.

Postupci određivanja topologije mreže

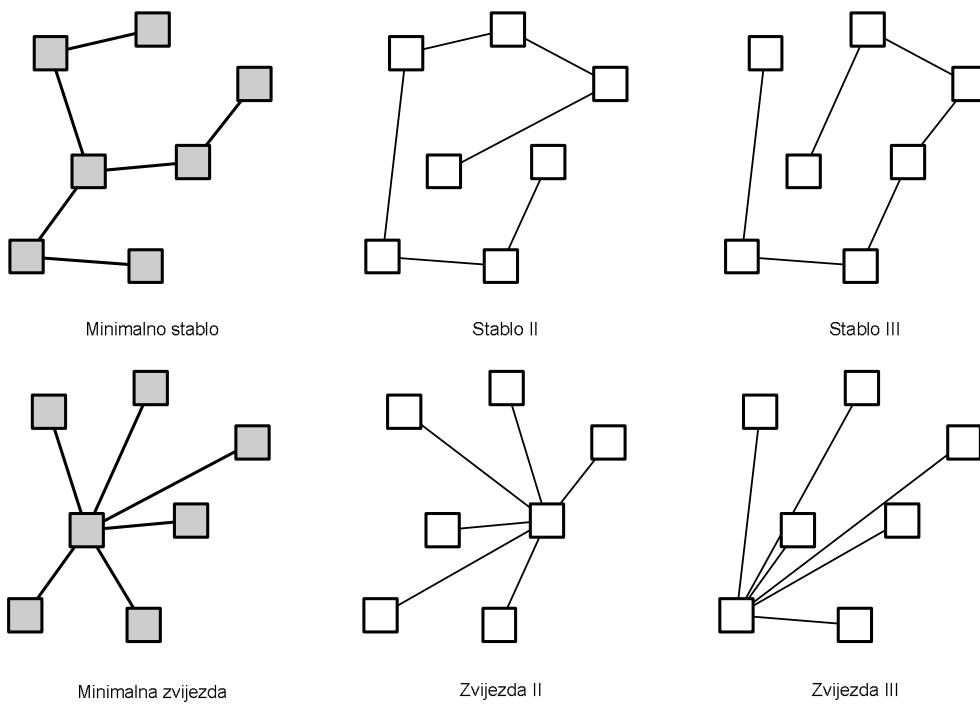
Ovisno o vrsti mreže i razini povezanosti koriste se različiti postupci za određivanje topologije mreža. U jednostruko povezanim mrežama, gdje se ne zahtjeva redundancija veza među čvorovima, koriste se jednostruko povezane strukture – stabla i zvijezde. Takve topološke strukture ne mogu osigurati alternativne puteve kojima bi se realizirali postavljeni prometni zahtjevi u slučaju kvara u mreži. Višestruko povezane mreže topološki su organizirane u složenje dvostruko i višestruko povezane strukture. Struktorno najjednostavnije višestruko povezane mreže su izgrađene od prstenova. Veća razina povezanosti u mrežama dodatno povećava redundanciju veza u mreži i u tom slučaju govorimo o povezanoj (*mesh*) mreži. Krajnji slučaj, kad su svi čvorovi međusobno povezani je potpuno povezana mreža (*full-connected mesh*). Slika 1.2. prikazuje primjere različitih mrežnih topologija.

Svaka od spomenutih struktura dobiva se analizom lokacija čvorova i određivanjem veza između čvorova u skladu sa zahtjevima. Naravno, parametri koji određuju cijenu mreže dodatno se analiziraju kako bi se postigla minimizacija troškova pri izgradnji mreže. Nije svejedno kakvo će stablo ili zvijezda biti izgrađeni, ni na koji će se način čvorovi povezati u prsten. Pri određivanju veza između čvorova najčešći parametar za minimizaciju troškova je duljina veza (npr. duljina bakrenog ili optičkog kabela i slično).



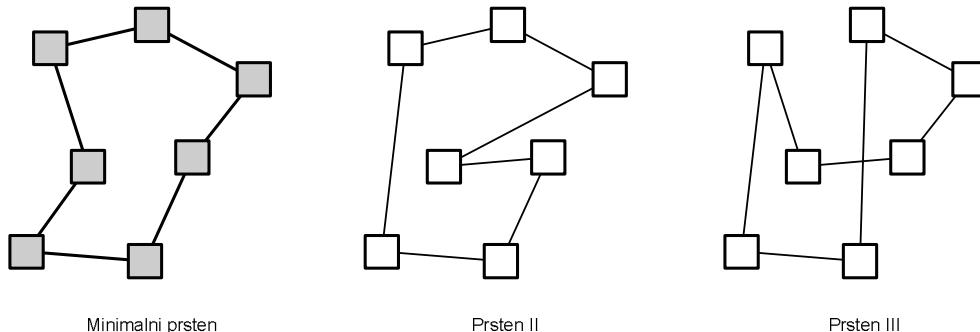
Slika 1.2: Topološke strukture – stablo, zvijezda, prsten, višestruko povezane mreže

Slika 1.3. prikazuje različite načine određivanja stabla odnosno zvijezde. Minimalno razapinjajuće stablo (*Minimum Spanning Tree*) je najkraće jednostruko povezano rješenje. Primjer zvjezdaste strukture također ukazuje na potrebu da se odredi 'centralni' čvor kako bi se duljina potrebnih veza za izvedbu strukture minimizirala.



Slika 1.3: Različite strukture stabla i zvijezde

Sličan je problem i pri određivanju topologije prstena. Čvorovi koje prsten sadrži mogu se povezati na različite načine. Ako vezama između čvorova pridijelimo težine, možemo odrediti najkraću zatvorenu strukturu (prsten) koji povezuje sve čvorove. Problem određivanja minimalnog prstena je identičan problemu trgovackog putnika (TSP – *Traveling Salesman Problem*). Slika 1.4. prikazuje nekoliko različitih realizacija prstena na istom skupu čvorova.



Slika 1.4: Različite realizacije prstena između čvorova u mreži

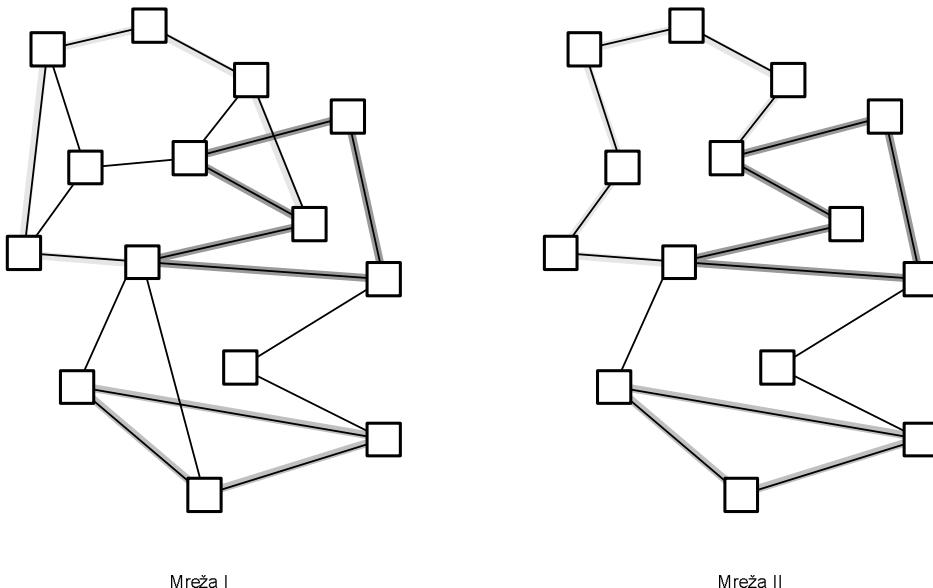
Složenije višestruko povezane topološke strukture dobivaju se kombiniranjem različitih metoda – spajanjem stabala, kombiniranjem prstenova, dodavanjem grana u prstenove, ukidanjem grana inicijalno potpuno povezane strukture i slično [1].

Neovisno o topologiji koja se koristi, cijena mreže u fazi određivanja topologije isključivo je ovisna o karakteristikama grana (npr. duljini). Čvorovi predstavljaju fiksani trošak (npr. troškovi postavljanja), što znači da se nastoji minimizirati cijena izvedbe grana koje se uključuju u rješenje.

Specifičnosti prstenaste mrežne topologije

Spomenuti postupci za određivanje višestruko povezane topologije u mreži mogu se koristiti u planiranju povezanih (*mesh*) i prstenastih mreža. Međutim, pri planiranju prstenastih mreža dobro je od početka voditi računa o tome da topologija što jednostavnije podrži korištenje prstenastih struktura, jer će strukturno jednostavnija topologija mreže osigurati jednostavnije provođenje ostalih faza u planiranju.

Svaka složenija višestruko povezana topologija omogućava korištenje prstenastih zaštitnih struktura. Nadogradnjava takvih mreža može biti prilično složeno i mukotrpno. Zbog toga se već u fazi planiranja topologije nastoji pojednostaviti strukturu mreže tako da identifikacija i realizacija pojedinačnih prstenova bude što jednostavnija. Shematski prikaz različitih pristupa u planiranju je dan na slici 1.5.



Slika 1.5: Prstenovi u povezanoj (*mesh*) mreži i planski generiranoj mreži

Rezultati dobiveni prilagodbom na *mesh* model razlikuju se od rezultata na planski izgrađenoj topologiji. Na primjerima *mesh* topologija rezultantni prstenovi najčešće će se preklapati i neki će čvorovi funkcionirati neovisno u različitim prstenovima, dok se kod planski pripremljene topologije susjedni prstenovi mogu jednostavno vezati prema definiranim pravilima, te nema preklapanja prstenova. Taj detalj vidljiv je i na slici 1.5. – za razliku od mreže II, u mreži I postoje preklapanja posebno označenih prstenova s ostalim strukturama u mreži.

1.2.3. Topologija prstena

Prsten je najjednostavnija dvostruko povezana struktura. U takvoj topološkoj strukturi svi čvorovi su povezani dvama neovisnim putevima, što se u praksi iskorištava za realizaciju jednostavnog, a vrlo efikasnog mehanizma automatske zaštite prometa u slučaju kvara. Zaštita funkcionira u potpunosti u slučaju jednostrukog kvara u mreži, dok u slučaju višestrukih kvarova nema garancije zaštite prometnih zahtjeva.

Kao što je u prethodnom odjeljku opisano, ovisno o težinama grana nastoji se odabrati najpovoljnije topološko rješenje povezanosti čvorova u prstenu. Ukoliko cijena grane ovisi o udaljenosti rubnih čvorova, optimalno rješenje je prsten bez 'preklapanja' grana – problem određivanja topologije takvog prstena identičan je TSP problemu. Uz postojeća ograničenja veličine prstenova, problem je moguće riješiti egzaktno.

1.2.4. Usmjeravanje prometnih zahtjeva

Dok u samom postupku određivanja topologije mreže cijenu i troškove izgradnje određuju najčešće duljine grana i karakteristike medija koji se koriste za izvedbu tih grana, prometni zahtjevi uvelike utječu na cijenu u dalnjem postupku dizajna mreže. Naime, ovisno o zahtijevanim prijenosnim kapacitetima grana cijena mreže raste – osim fizičke izvedbe grana koje moraju omogućiti prijenos zahtijevanih kapaciteta, nužno je uložiti i u izgradnju zahtijevanih sučelja na čvorovima. Ukoliko se npr. zahtijeva prijenos STM-1 signala nekom granom, nužno je da čvorovi povezani tom granom posjeduju sučelja koja prihvataju STM-1 signale. U suprotnom nije moguće realizirati komunikaciju. Dakle, cijena mreže direktno ovisi o kapacitetima grana u mreži – cijenu definiraju sučelja na rubnim čvorovima grana.

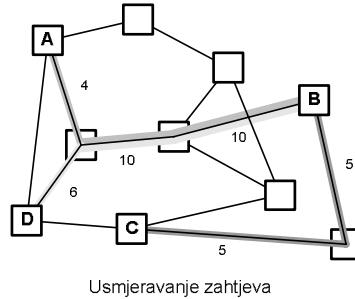
Općenito, razlikujemo mreže sa fiksnim usmjeravanjem i mreže sa dinamičkim usmjeravanjem prometa. Ovim radom se analiziraju mreže sa fiksnim usmjeravanjem – zahtjevi između određenog

para čvorova prenose se, u standardnom režimu rada mreže, uvijek istim putem. U slučaju kvara promet se preusmjerava alternativnim, također unaprijed definiranim putevima.

Ukoliko su zadani prometni zahtjevi – bilo skupom zahtjeva D ili prometnom matricom zahtjeva MD , jasno je da će se tražena komunikacija uspostavljati korištenjem grana mreže. Određivanje puteva kojima prometni zahtjevi prometuju od izvorišnog do odredišnog čvora naziva se usmjeravanjem zahtjeva. Kad se za sve zahtjeve odrede takvi putevi moguće je odrediti zahtijevane kapacitete grana u mreži. Slika 1.6. prikazuje primjer jednostavnog usmjeravanja tri prometna zahtjeva iz tablice 1.1.

Tablica 1.1: Prometni zahtjevi

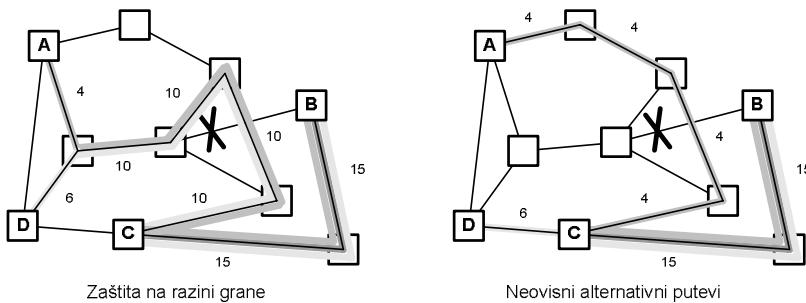
Izvorište	Odredište	Zahtjev
A	B	4
B	C	5
B	D	6



Slika 1.6: Usmjeravanje zahtjeva kroz mrežu

Ne ulazeći u stohastičku prirodu komunikacijskih procesa, pretpostavljamo da su zahtjevi definirani za isti slučaj, npr. za glavni prometni sat. Kapaciteti grana određuju se zbrajanjem zahtjeva koji prolaze kroz njih.

Primjer sa slike 1.6. rezultira određenim zahtijevanim kapacitetima grana. No, što će se dogoditi ako u mreži dođe do kvara? Svi zahtjevi koji su usmjeravani kroz dio mreže pogoden kvarom nepovratno se gube. Rješenje je u izvedbi zaštitnih mehanizama koji omogućavaju preusmjeravanje prometa alternativnim putevima u slučaju kvara u mreži. Primjer preusmjeravanja prikazan je na slici 1.7. Na slici su prikazana dva glavna načina realizacije alternativnih puteva – zaštita na razini puta i zaštita na razini grane. Kod zaštite na razini puta prometni zahtjev se usmjerava putem (bilo unaprijed, bilo dinamički određenim) neovisnim od inicijalnog. Kod zaštite na razini grane put se mijenja samo na dionici zahvaćenoj kvarom – grana na kojoj se dogodila pogreška zaobilazi se unaprijed definiranim putem.



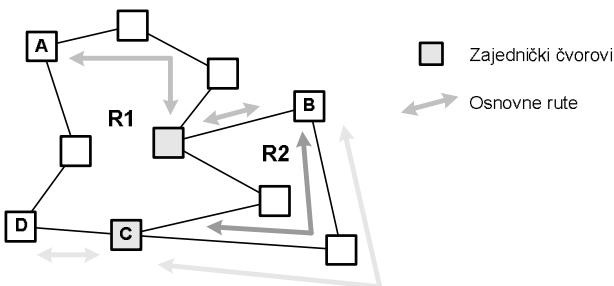
Slika 1.7: Kvar u mreži i mogućnosti preusmjeravanja zahtjeva

Jasno je da će zbog preusmjeravanja doći do povećanja zahtijevanih kapaciteta na granama koje se koriste za uspostavu alternativnih puteva.

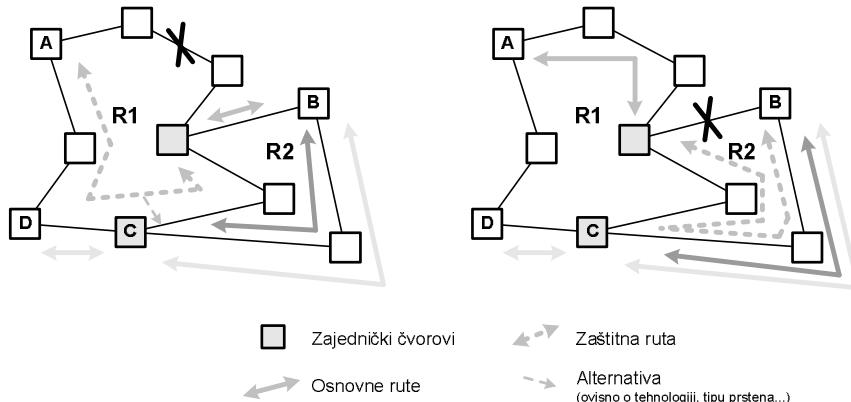
Budući da je cilj minimizirati kapacitete grana u mreži, problem usmjeravanja prometnih zahtjeva uz pružanje mogućnosti zaštite prometa vrlo je složen. Postoji niz različitih pristupa rješavanju tog problema. Ovisno o predviđenoj topologiji mreže rezultati i postupci se razlikuju. Jedan od pristupa u određivanju optimalnog usmjeravanja u *mesh* mreži je izvedba inicijalnog usmjeravanja korištenjem najkraćih puteva u mreži, dok se izvedba alternativnog usmjeravanja zasniva na

prilagođavanju kapacitetima preostalih grana – analiziraju se preopterećene i podopterećene grane te se pokušava izbalansirati promet kako bi rezultat bio što bolji. Najkraći putevi određuju se prema različitim težinskim vrijednostima – udaljenosti čvorova tj. duljini dionica, broju čvorova kroz koje zahtjev mora proći i slično.

U prstenastim mrežama pristup je nešto drugačiji. Budući da se mreža funkcionalno dijeli na više prstenova (svaki od njih posjeduje funkciju zaštite prometa u slučaju kvara) usmjeravanje se svodi na određivanje prstenova kroz koje zahtjevi prolaze. Za svaki zahtjev može se odrediti da li se usmjerava unutar vlastitog prstena ili mu je odredište u nekom drugom prstenu u mreži. Mreža se strukturno razdvaja na prstenove i usmjeravanje zahtjeva čija su izvođači i odredišta u različitim prstenovima svede se na traženje puta koji će najmanje povećati kapacitete svakog pojedinog prstena, drugim riječima koji će prolaziti kroz najmanji broj prstenova. Dodatno usmjeravanje zbog omogućavanja zaštite prometa nije potrebno obavljati jer svaki pojedini prsten u potpunosti zaštićuje prometne zahtjeve koji njime prolaze! Primjer usmjeravanja zahtjeva u prstenastoj mreži prikazan je na slici 1.8. U slučaju kvara u mreži pogodeni prstenovi aktiviraju zaštitne mehanizme i prometni zahtjevi su zaštićeni (slika 1.9.).



Slika 1.8: Usmjeravanje zahtjeva u prstenastoj mreži



Slika 1.9: Kvarovi i alternativno usmjeravanje u prstenastoj mreži

Iako na trenutak rješenje korišteno u prstenastim mrežama može izgledati idealnim, ipak to nije slučaj. Naime, korištenje prstena povećava ukupne zahtjeve za kapacitetima u mreži. Sve grane unutar prstena moraju biti jednakom dimenzionirane, bez obzira da li i koliko u standardnom režimu rada kroz njih prolazi prometnih zahtjeva. Može se dogoditi da pojedine grane, iako dimenzionirane na velike kapacitete, budu poprilično podopterećene. To je vrlo neefikasno i neisplativo, jer je cijena grane prije svega ovisna o instaliranom kapacitetu (medij + sučelja na rubnim čvorovima). Dodatna neisplativost postoji zbog samog principa izgradnje zaštitnih prstenova – iako je to na modelima 'nevidljivo', svaki prsten uključuje kapacitete za zaštitu što uvelike povisuje cijenu mreže. Da bi zaštita funkcionirala u svim slučajevima izvode se 'paralelni' zaštitni prstenovi takvi da je njihov zaštitni kapacitet jednak kapacitetu standardnih prstenova. No, činjenica je da su takvi modeli zaštite provjereni u praksi i gotovo su standardni dio svih temeljnih telekomunikacijskih mreža.

Isplativost izgradnje prstenastih mreža se može povećati dodjeljivanjem prioriteta prometnim zahtjevima. Moguće je, naime, definirati prometne zahtjeve nižeg prioriteta za koje se ne zahtijeva zaštita. Takvi prometni zahtjevi mogu se usmjeravati korištenjem zaštitnih kapaciteta u prstenovima, no u slučaju kvara u mreži njihov prijenos postaje upitan – ukoliko su prenošeni dijelom mreže koji se zbog kvara aktivirao kako bi zaštitio promet višeg prioriteta prekida se s njihovim prijenosom. Podopterećene grane u glavnim prstenovima se također mogu iskoristiti za prijenos specifičnog niskoprioritetnog prometa, čime se može povećati njihova iskoristivost. Takvo iskorištanje kapaciteta prstenastih mreža s ciljem povećanja iskoristivosti je često u praksi.

1.2.5. Analiza topologije i pronalaženje prstenova

Dok se pri izgradnji novih mreža može voditi računa o pametnom planiranju i modelima koji će olakšati izvedbu i povezivanje prstenastih struktura, na postojećim mrežnim topologijama potrebno je izvršiti detaljnu analizu i odrediti skup prstenastih struktura koje prekrivaju cijelu mrežu.

Nakon što se utvrdi da li topologija zadovoljava uvjet o višestrukoj povezanosti, cilj je pronaći sve prstenaste strukture u zadanoj mreži – postupak pronalaženja prstenova još se naziva i enumeracija prstenova. Sama po sebi enumeracija ne predstavlja optimizacijski problem, no zbog eksponencijalnog rasta broja rješenja s brojem čvorova i grana u mreži, javlja se praktični problem određivanja prstenova koji se mogu uključiti u rješenje. Problem se tretira kao REP (*Ring Enumeration Problem*), no bitno je razlučiti tip problema od ostalih (RDP, RCP). Kad su prstenovi u mreži pronađeni, sljedeći korak je odabir podskupa prstenova koji prekrivaju zadalu mrežnu topologiju (ovisno o parametrima postupka planiranja, prekrivanje može biti izvedeno na razini čvorova – u tom slučaju broj prstenova može biti manji, ili na razini grana mreže). Problem prekrivanja mreže prstenovima (RCP – *Ring Covering Problem*) je vrlo složen, te je optimalno rješavanje tog problema u pravilu teško izvedivo. Zbog toga se u raznim alatima za planiranje prstenastih mreža koriste različiti heuristički algoritmi, najčešće kratkovidni, koje rezultiraju podoptimalnim rješenjima.

Jedan od bitnih parametara u određivanju podskupa prstenova koji prekrivaju mrežu jest vrednovanje prstenova. Pri određivanju prstena koji se dodaje u rješenje uvijek se nastoji odabrati "najbolji" prsten. Parametri za određivanje 'najboljeg' prstena direktno utječu na konačne rezultate, te ih je nužno što kvalitetnije definirati. Da li kao najbolje rješenje odabirati prstene koji uspijevaju obraditi najviše zahtjeva, prstene koji imaju najveći kapacitet ili prstene koji imaju najbolju iskoristivost? Razni alati na tržištu postavljaju različite uvjete pri određivanju rješenja. Stoga je nužno provesti analizu rezultata dobivenih različitim kriterijima odlučivanja. Na taj se način smanjuje utjecaj kratkovidnosti metoda za rješavanje RCP-a, jer različiti kriteriji uvjetuju analizu različitih područja rješenja.

Ne manje važan utjecaj na rezultate problema prekrivanja mreže prstenovima ima i određivanje usmjeravanja zahtjeva kroz mrežu. Iako postoje različiti pristupi planiranju prstenastih mreža, većinom se provodi inicijalno usmjeravanje zahtjeva kako bi se odredilo kojim putem se zahtjevi usmjeravaju kroz predložene prstenove. Različiti principi usmjeravanja mogu dati različite rezultate prekrivanja mreže.

Također, jedan od problema u postupku planiranja je vrednovanje rezultata – razvoj složenijeg matematičkog modela je poprilično mukotrpan i nije izведен kao dio ovog rada. Stoga su dobiveni rezultati analizirani jedni u usporedbi s drugima kako bi se uočile prednosti i nedostaci korištenja različitih faza u postupku.

Alternativno, na manjim mrežama moguće je izvesti kombiniranje prstenova dobivenih postupkom enumeracije, te tako proanalizirati i vrednovati sve moguće kombinacije prekrivanja mreže. Usporedba tako dobivenih rezultata sa rezultatima predloženih metoda može pomoći u donošenju zaključaka, no ne može garantirati kvalitetu rezultata u većim i složenijim mrežama. Ukoliko se u postupku enumeracije odredi N_R prstenova u mreži, broj mogućih kombinacija prstenova je $2^{N_R} - 1$. Sve te kombinacije ne prekrivaju cijelu mrežu, tako da je neke od njih moguće zanemariti

i reducirati broj mogućih prekrivanja. Prekrivanje mreže se također može različito interpretirati – npr. ponekad se postavljaju zahtjevi da prstenovi moraju uključiti sve čvorove u mreži, ponekad samo čvorove koji su izvorišta i/ili odredišta prometnih zahtjeva, dok se ponekad zahtjeva uključivanje svih grana u prstenove. Zbog složenosti problema mnogi alati omogućavaju definiranje ograničenog broja izvedivih prstenova u mreži, čime se postupak može pojednostaviti i ubrzati (smanjuje se broj mogućih prekrivanja, a time i procesorsko vrijeme potrebno za obradu i proračune).

Najbolji uvid u složenost problema može dati tablični prikaz broja mogućih prstenova u mreži. Broj mogućih prekrivanja mreže prstenovima eksponencijalno raste s brojem mogućih prstenova, što je zorno prikazano u tablici 1.2.

Tablica 1.2: Mogući broj prstenova i prekrivanja mreže

Broj čvorova N_V	Broj grana N_E	Mogućih prstenova (REP)	Mogućih prekrivanja (RCP)	Broj znamenaka (RCP)
5	7	8	255	3
14	17	16	65535	5
15	22	256	1,157e+77	78
15	30	65536	2,000e+19728	19729
20	45	6,7108e+7	*	2020178
30	75	7,0368e+13	*	2.11831e+13

* vrlo veliki brojevi

Iz tablice je očigledno da je bilo kakav postupak potpune enumeracije kod većih mreža potpuno besmislen i neopravдан. Iako potpuna enumeracija u REP nije nužna, ipak se redukcijom broja mogućih prstenova može utjecati na konačna rješenja. Enumeracija u RCP bi omogućila kvalitetniju analizu rezultata dobivenih alternativnim heurističkim metodama, no zbog ogromne složenosti za realne mreže nije uvijek izvediva u praksi.

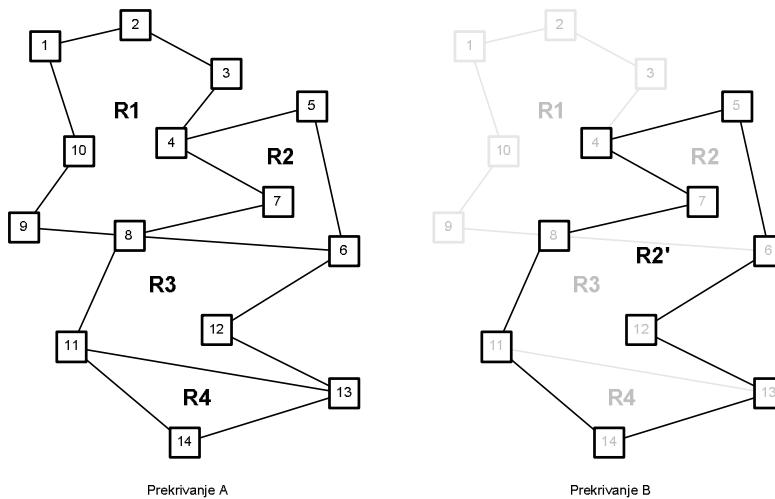
1.2.6. Usmjeravanje prometnih zahtjeva kroz prstenove

Nakon što se na postojećoj mrežnoj topologiji definiraju prstenovi potrebno je usmjeriti prometne zahtjeve tako da zahtijevani kapaciteti budu minimalni. Za svaki od definiranih prstenova prometni zahtjevi se analiziraju i grupiraju – zahtjevi se podijele na lokalne (*local demands*) i meduprstenaste (*inter-ring demands*) zahtjeve. Lokalni zahtjevi su zahtjevi između para čvorova u istom prstenu – usmjeravanje i zaštita takvih zahtjeva u potpunosti se obavlja u pridruženom prstenu. Preostali zahtjevi prenose se između čvorova koji se ne nalaze u istom segmentu mreže – svaki čvor nalazi se u drugom prstenu, te je za uspostavu prometa potrebno zahtjev usmjeriti između izvorišnog čvora u izvorišnom prstenu i odredišnog čvora u odredišnom čvoru. Ovisno o pozicijama izvorišnog i odredišnog čvora/prstena takav prometni zahtjev mora proći kroz dva (ukoliko su izvorišni i odredišni prsten susjedni prstenovi) ili više prstena. Primjer lokalnih i meduprstenastih zahtjeva opisan je na modelima mreže na slici 1.10.

U tablici 1.3. opisani su prometni zahtjevi i prstenovi koji se koriste pri njihovom usmjeravanju.

Tablica 1.3: Prometni zahtjevi – lokalni i međuprstenasti

Izvorište	Odredište	Prekrivanje A		Prekrivanje B	
		Tip zahtjeva	Usmjeravanje	Tip zahtjeva	Usmjeravanje
1	4	lokalni	R1	lokalni	R1
1	6	međuprstenasti	R1-R2	međuprstenasti	R1-R2'
1	10	lokalni	R1	lokalni	R1
3	8	lokalni	R1	lokalni	R1
3	14	međuprstenasti	R1-R2-R3-R4 ili R1-R3(8)-R4	međuprstenasti	R1-R2'
9	13	međuprstenasti	R1-R2-R3 ili R1-R3	međuprstenasti	R1-R2'
5	14	međuprstenasti	R2-R3-R4	lokalni	R2'
6	13	lokalni	R3	lokalni	R2'
11	14	lokalni	R4	lokalni	R2'



Slika 1.10: Modeli prekrivanja mreže prstenovima

Intuitivno se problem usmjeravanja ne čini složenim, no u stvari se radi o prilično složenom problemu. Potrebno je uskladiti optimalno usmjeravanje lokalnih zahtjeva u prstenu (što je općepoznati problem dimenzioniranja prstena – RDP) sa usmjeravanjem i zaštitom međuprstenastih zahtjeva. Postupak određivanja optimalnih kapaciteta u mreži ovisan je ponajviše o karakteristikama mrežne opreme koja se koristi. Ovisno o vrstama prstenova (ili kombinacijama različitih vrsta prstenova) i modelima povezivanja prstenova i zaštite međuprstenastog prometa, realiziraju se različiti postupci planiranja.

1.2.7. Rezultati postupka planiranja

Rezultati postupka planiranja mreže moraju sadržavati dovoljno informacija koje će omogućiti izvedbu realne mreže. To ne moraju biti detaljne specifikacije svakog pojedinog dijela opreme koji se koristi, već općenite informacije karakteristične za pojedinu mrežnu opremu i slično.

Rezultat uključuje resultantnu topologiju – skup odabralih čvorova V_R i grana E_R . Svakom čvoru iz skupa V_R definiraju se svojstva koja će dizajneru mreže dati dovoljno korisnih informacija pri realizaciji mreže. Osim informacija o topologiji predložene mreže, dio rezultata čine i informacije o funkcionalnim (topološkim) strukturama u mreži – podaci o predloženim prstenovima, njihovim kapacitetima i ostalim svojstvima. Podaci o usmjeravanju i zaštiti (na razini prstenova) prometnih zahtjeva također su uključeni u rezultat. Za svaki se prometni zahtjev definiraju putevi i skup čvorova kojima se zahtjev inicijalno usmjeruje. Inicijalno usmjeravanje koje se dobiva kao rezultat postupka je usmjeravanje koje se koristi u normalnom režimu rada mreže. Za slučaj kvara nije potrebno definirati puteve jer će se alternativni put uspostaviti na razini prstenova u mreži kad se kvar u mreži pojavi.

1.3. Pristupi planiranju prstenastih mreža

Dva su različita pristupa planiranju prstenastih mreža. Prvi pristup uključuje definiranje što kompletnijeg matematičkog modela i izvođenje različitih metoda matematičkog programiranja s ciljem dobivanja optimalnog rješenja. Zbog visoke složenosti modela mreže i procesa planiranja, često se radi o vremenski i prostorno vrlo zahtjevnim postupcima koji ne uspijevaju pronaći optimalno rješenje u realnom vremenu. Kako bi se procedura ubrzala pribjegava se pojednostavljenju matematičkog modela mreže – na taj način smanjuje se složenost problema, no isto tako gubi se i dio rješenja. Sama izrada matematičkog modela također može predstavljati problem, posebice ako se u obzir uzima niz ograničenja koja se mogu pojaviti u realnim mrežama.

Drugi pristup, češće prisutan u praksi, svodi se na kombiniranje različitih metoda u procesu dizajna podijeljenom na faze kao što je opisano u prethodnim poglavljima. Već i samo razdvajanje složenog postupka na faze smanjuje područje rješenja, što znači da postoji mogućnost da se optimalno rješenje odmah eliminira (to je u stvari vrlo čest slučaj). Idealni alat bi trebao postupak planiranja obaviti automatski, sa što manje zahtjeva za sudjelovanjem dizajnera u samom postupku. Međutim, takav način rada često dovodi do rezultata koje dizajner naknadno obraduje i prilagođava realnom stanju mreže. Postoje dvije krajnosti – alati koji u potpunosti automatiziraju postupak planiranja i alati koji zahtjevaju intenzivno sudjelovanje korisnika pri donošenju odluka. Bez obzira na princip funkcioniranja alata dobiveno rješenje gotovo sigurno neće biti optimalno. Za razliku od pristupa baziranog na matematičkom modelu, heuristički postupci koji se koriste u sklopu faza planiranja niske su razine složenosti, te daju rezultate u prihvatljivom vremenu. To omogućuje njihovo intenzivno korištenje radi naknadne obrade i izvedbe modifikacija na inicijalnom rješenju.

Tablica 1.4. prikazuje osnovne prednosti i nedostatke pojedinog pristupa.

Tablica 1.4: Usporedba različitih pristupa planiranju

Matematički model mreže	Planiranje prema fazama
- matematičko programiranje	- heuristike
- optimalno rješenje	- podoptimalna rješenja
- visoka složenost	- reducirana složenost
- sporo izvođenje	- brzi proračuni

Pristup planiranja prema fazama je interesantan prije svega zbog svoje primjenjivosti u stvarnim mrežama. Usporedbom rješenja koja se dobivaju različitim postavkama postupka planiranja mogu se pronaći kvalitetna rješenja koja zadovoljavaju zahtjeve dizajnera. Međutim, teško je tvrditi da se postupak planiranja može svesti na isključivo automatizirani proces s minimalnim utjecajem samog dizajnera – alati koji se koriste u postupku planiranja isključivo su tu kako bi pomogli dizajneru u njegovom radu, a ne eliminirali potrebu za uslugama dizajnera mreža.

Ukoliko se postupak planiranja razdvoji u faze, u sklopu svake faze omogućava se viša ili niža razina interakcije dizajnera i alata. Praktički je u svakoj fazi moguće dozvoliti dizajneru da u potpunosti definira uvjete i time utječe na proces planiranja. Ovisno o uvjetima i definiranim ulaznim podacima za svaku fazu, rezultati se razlikuju. Prednosti i nedostaci različitih pristupa za pojedine faze pobrojane su u tablici 1.5.

Tablica 1.5: Prednosti i nedostaci različitih pristupa za pojedine faze

FAZA		Prednosti	Nedostaci
Topologija	Automatizirano	- brza definicija grana u mreži	- nerealni rezultati - složena implementacija stvarnih ograničenja
	Manualno	- realan prikaz mreže	- problem unosa velike količine informacija
Određivanje prstenova	Automatizirano	- pronađi sve prstenove - redukcija postavljanjem ograničenja	- velik broj prstenova u realnim mrežama - uključujući i nerealna rješenja, složenost
	Manualno	- utjecaj dizajnera na rezultate - realne mrežne strukture	
Usmjeravanje zahtjeva	Automatizirano	- analiza rješenja uz različita usmjeravanja - brz i efikasan postupak	- složena implementacija realnih ograničenja
	Manualno	- unaprijed određeni putevi - sporo definiranje u većim mrežama	- neoptimalno usmjeravanje
Prekrivanje prstenovima	Automatizirano	- rezultati ovisni o ostalim fazama - različiti modeli i kriteriji selekcije	- kratkovidne metode
	Manualno	- realan prikaz mreže	- može se iskoristiti jedino za određivanje usmjeravanja

Općenito, svaka od faza može se izvoditi automatizmom ili uz interakciju dizajnera. Dizajner može u potpunosti preskočiti proračune u pojedinim fazama ručnim definiranjem svih potrebnih parametara i rezultata koje faza treba polučiti. Tablica za svaki od načina izvođenja navodi osnovne prednosti i nedostatke takvog pristupa.

Automatizirano *određivanje topologije* moguće je izvesti korištenjem različitih metoda, no to najčešće rezultira prilično nerealnim rezultatima. Stoga se pri planiranju prstenastih mreža topologija u potpunosti manualno definira – na taj način osigurava se realan prikaz mreže.

Određivanje prstenova je postupak koji se ne može provesti u realnom vremenu zbog velikog broja mogućih rješenja. Iako je automatizirano rješavanje moguće izvesti, potpuna enumeracija se najčešće izbjegava. Postavljanjem realnih ograničenja (minimalna i maksimalna veličina prstena i slično) broj prstenova se može reducirati i na taj način ubrzati provedbu kasnijih faza u planiranju. Međutim, ponekad će biti najrealnije definirati prstenove u mreži manualno – na taj način se unaprijed ograničavaju moguća rješenja na ona koja je dizajner unaprijed predvidio.

Usmjeravanje zahtjeva se najčešće obavlja automatiziranim postupkom, uz mogućnost korištenja različitih modela usmjeravanja – na taj se način utječe na rezultate faze prekrivanja prstenova, te je moguća analiza i određivanje najboljeg modela usmjeravanja. Manualno definiranje puteva kojima se zahtjevi usmjeravaju koristi se u pravilu u već izgrađenim mrežama s ciljem optimiranja korištenja prstenastih struktura.

Najčešće se završna faza, *prekrivanje prstenova*, obavlja automatizirano. Prekrivanje mreže prstenovima moguće je definirati i manualno, a tome se pribjegava u slučaju kad je potrebno odrediti optimalno usmjeravanje zahtjeva – primjer je već izgrađena mreža sa definiranim prstenovima u kojoj je došlo do promjene u strukturi prometnih zahtjeva. U tom se slučaju optimalnim usmjeravanjem mogu smanjiti kapaciteti prstenova i na taj način omogućiti prijenos određenog broja budućih zahtjeva bez promjena u strukturi mreže.

Automatizirano i manualno izvođenje pojedinih faza moguće je kombinirati – najčešće korišteni pristupi navedeni su u tablici 1.6.

Tablica 1.6: Najčešće korišteni pristupi projektiranja mreže prema fazama

FAZA	Pristup A	Pristup B	Pristup C	Pristup D
Određivanje prstenova	Zadano	-	Zadano	Automatsko
Usmjeravanje	Zadano	Automatsko	Automatsko	Automatsko
Prekrivanje prstenovima	Automatsko	Zadano	Automatsko	Automatsko

U pregledu pristupa projektiranju mreže prema fazama izostavljena je faza određivanja mrežne topologije – u praksi se kompletna topologija zadaje na početku postupka planiranja ili pak se na temelju definiranih čvorova određuje neka višestruko povezana topološka struktura korištenjem različitih automatiziranih metoda.

Pristup A polazi od zadanog skupa mogućih prstenova i zadanog usmjeravanja prometnih zahtjeva, te na temelju tih informacija provodi postupak selekcije prstenova i određivanje podskupa prstenova koji prekrivaju mrežu. Ovakav pristup može se koristiti za verifikaciju postojećih mreža ili određivanje prekrivanja u mreži koja je djelomično projektirana od strane dizajnera.

Pristup B specifičan je po tome što je zadano prekrivanje mreže prstenovima (u tom slučaju, enumeracija prstenova nije bitna). Rezultat koji se traži je optimalno usmjeravanje zahtjeva kroz definirane strukture u mreži. Takav pristup koristi se s ciljem povećanja iskoristivosti mrežnih kapaciteta, a može poslužiti i za verifikaciju dimenzioniranja mreže.

Pristupi C i D su 'kompletni' postupci planiranja jer uključuju automatizirano određivanje usmjeravanja zahtjeva i prekrivanja mreže. Na taj način moguće je provjeriti više različitih načina usmjeravanja i različitih kriterija selekcije prstena za prekrivanje mreže. Time se mogu dobiti podoptimalna rješenja (u idealnom slučaju moguće je dobiti i optimalno rješenje). Razlika u pristupima je ta što pristup D uključuje i automatiziranu enumeraciju prstenova u mreži. Takav model može se vrlo dobro iskoristiti u dizajnu manjih mreža, dok se kod složenih mreža, a zbog zahtjevnosti procesa enumeracije, preporuča ograničavanje broja prstenova ili definiranje skupa prstenova unaprijed.

U ovom radu su analizirane sve spomenute faze dizajna, izloženi postupci rješavanja problema te prikazani rezultati dobiveni metodama implementiranim u praktičnom dijelu rada.

2. Jednostavan prsten

Prsten (engl. *ring* ili *cycle*) je najjednostavnija struktura koja povezuje čvorove omogućavajući uspostavljanje dva neovisna puta između svaka dva čvora. To jednostavno svojstvo iskorištava se za efikasnu zaštitu prometa u mreži u slučaju kvarova. Za razliku od složenijih struktura, preusmjeravanje prometa u slučaju kvara u prstenu je vrlo jednostavno i ne zahtijeva dodatnu analizu mreže i određivanje alternativnog puta među čvorovima.

Ovisno o tehnološkoj izvedbi prstenaste mreže, zaštita se realizira na različite načine – cilj je omogućiti zaštitu prometnih zahtjeva uz minimalne troškove. U praksi su troškovi izgradnje najčešće proporcionalni prometnim kapacitetima, pa se nastoji što više reducirati zahtjeve za kapacitetima. Jedan od načina smanjenja zahtijevanih kapaciteta je kvalitetno usmjeravanje prometnih zahtjeva unutar prstenova i unutar cijele mreže. Smanjivanje troškova prstenaste mreže svodi se na smanjivanje kapaciteta prstena. Unatoč jednostavnosti prstenaste strukture, problemi usmjeravanja zahtjeva i minimizacije kapaciteta prstena su vrlo složeni. Optimalne rezultate često je nemoguće postići u realnom vremenu te se stoga u postupcima planiranja prstenastih mreža koriste razne heurističke metode određivanja podoptimalnih rješenja navedenih problema.

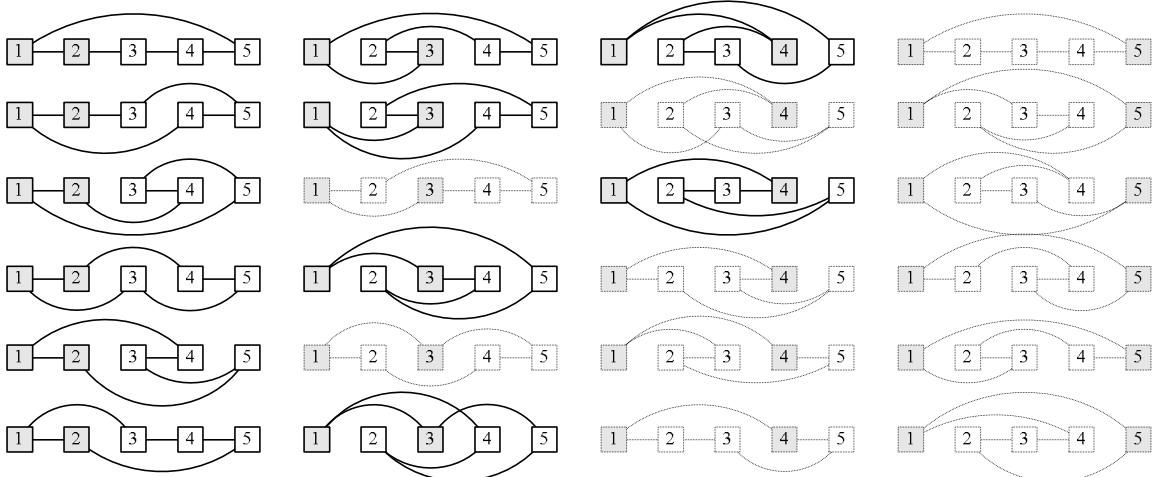
Ovo poglavlje analizira probleme dimenzioniranja jednostavnog prstena i opisuje nekoliko metoda za rješavanje tih problema. Daje se i pregled praktične primjene prstenova u telekomunikacijskim mrežama, s naglaskom na SDH/SONET transportnu tehnologiju. Navedeni su i neki problemi koji se javljaju u stvarnim mrežama, a koji nisu obuhvaćeni teoretskim modelima.

2.1. Struktura prstena

Topološki, prsten je jednostavna dvostruko povezana struktura – za povezivanje svaka dva čvora uključena u prsten postoje dva neovisna puta. Ukoliko se prsten sastoji od skupa čvorova V koji sadrži N čvorova, tada je moguće izvesti $\frac{(N-1)!}{2}$ različitih povezivanja svih čvorova u prsten.

Skup čvorova definiramo kao $V_N = \{v_i : i=1,2,\dots,N\}$. Svaki čvor v_i se može povezati sa $(N-1)$ čvorova v_j ($j \neq i$), svaki od tih $(N-1)$ sa preostalih $(N-2)$ v_k ($k \neq i, k \neq j$) čvorova itd. Kako promatramo samu topološku strukturu prstena, zanemaruјemo smjerove – tj. prsten u kojem su redom spojeni npr. $v_1-v_2-v_3-v_4-v_1$ izjednačavamo sa prstenom u kojem su spojeni $v_1-v_4-v_3-v_2-v_1$ – ukupni broj mogućih povezivanja čvorova u prsten dijelimo sa 2. Svaki od prstenova R koju uključuje sve čvorove iz skupa V_N definiran je i skupom E_N od N grana.

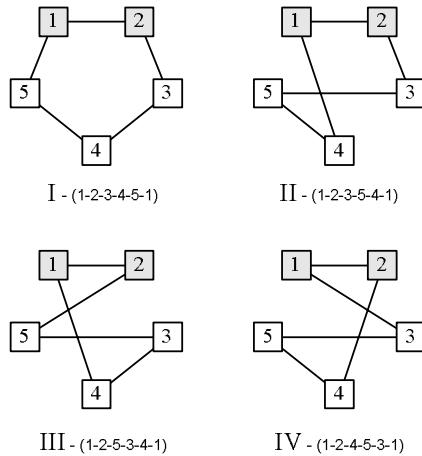
Na slici 2.1. su prikazane sva moguća povezivanja čvorova u prsten za skup čvorova V_5 .



Slika 2.1: Moguća povezivanja čvorova za skup od 5 čvorova

Topologije na slici 2.1. određivane su prema pravilu tako da je prva grana u prstenu grana između posebno naznačenih čvorova. Rezultantne topologije prikazane zasjenjeno pojavljuju se u prijašnjim koracima, te su zbog toga zanemarene. Nad skupom od 5 čvorova ukupno je moguće realizirati $4!/2=12$ topologija (24 moguće, s tim da se 12 ponavlja).

Lokacije čvorova nisu razmatrane u primjeru na slici 2.1., već su razmatrana samo moguća povezivanja čvorova – primjer nekoliko mogućih prstenova uz definirane lokacije čvorova dan je na slici 2.2.



Slika 2.2: Moguće topologije prstena prema pozicijama čvorova

Na slici 2.2. prikazano je nekoliko topologija prstena, vodeći računa o pozicijama čvorova. Vidljivo je da neka rješenja izgledaju prilično zbumujuće zbog 'preklapanja' grana. Očigledno je da rješenje I izgleda jednostavnije od rješenja III, no jednostavnost prikaza na papiru ne mora nimalo koincidirati sa jednostavnošću i efikasnošću realne izvedbe mreže!

Kako bi ipak dobili neku kvantitativnu mjeru pri planiranju mreže, pribjegavamo definiranju težinskih parametara za svaku granu, nakon čega u dalnjem postupku planiranja optimiziramo kompletну mrežu prema tom parametru. Najčešće se zadaje cijena grane, te se nastoji minimizirati cijenu mreže (sumu cijena svih grana) – koriste se parametri proporcionalni cijeni (duljina grane, broj vlakana, kapacitet...). Kako bi se što zornije prikazala problematika, u dalnjim razmatranjima u okviru ovog rada se koriste najjednostavniji prikazi (topologija I sa slike 2.2) – takav prikaz odgovara optimalnom u slučaju da je duljina grane proporcionalna cijeni. Prstenasta topologija I sa slike je dvostruko povezana topologija koja nudi najnižu cijenu (duljinu).

Promatrano na topologiji I sa slike, strukturno prsten \mathbf{R}_5 definiramo skupom čvorova V_5 i skupom grana $E_5 : \mathbf{R}_5 = \{V_5, E_5\}$. U primjeru - $V_5 = \{v_1, v_2, v_3, v_4, v_5\}$, $E_5 = \{e_{12}, e_{23}, e_{34}, e_{45}, e_{51}\}$. U oznakama grana e_{ij} indeksi i i j označavaju rubne čvorove v_i i v_j .

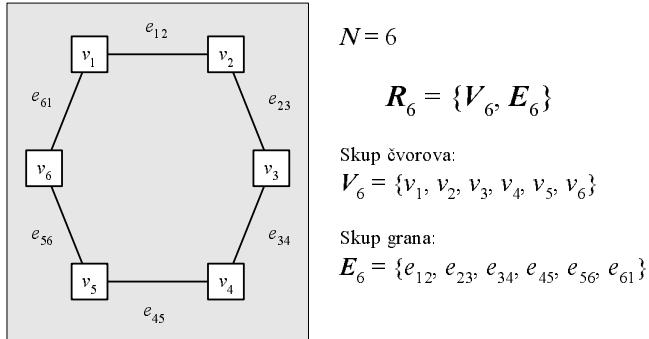
Ovisno o mogućoj komunikaciji između susjednih čvorova, prstene dijelimo na jednosmjerne (*unidirectional*) i dvosmjerne (*bidirectional*). U primjeru na slici 2.2. u slučaju jednosmernog prstena komunikacija između čvorova 1 i 3 odvijala bi se putem 1-2-3 za zahtjeve od 1 do 3, te putem 3-4-5-1 za zahtjeve od 3 do 1. Kad bi se radilo o dvosmjernom prstenu zahtjevi između čvorova 1 i 3 bi se u oba smjera prenosili putem 1-2-3 (3-2-1) ili 3-4-5-1 (1-5-4-3).

2.2. Općeniti model prstena

Prsten \mathbf{R} možemo definirati kao uređeni par koji sadrži skup čvorova V i skup grana E . Broj grana u prstenu jednak je broju čvorova. Prsten koji sadrži N čvorova označavamo sa \mathbf{R}_N , sa skupom čvorova V_N i grana E_N .

$$\begin{aligned}
\mathbf{R}_N &= \{\mathbf{V}_N, \mathbf{E}_N\} & |\mathbf{V}_N| = |\mathbf{E}_N| = N \\
\mathbf{V}_N &= \{v_i : i=1,2,\dots,N\} \\
\mathbf{E}_N &= \{e_{ij} : i=1,2,\dots,N; j=(i+1) \bmod N\}
\end{aligned}$$

Skup \mathbf{V}_N sadrži čvorove v_1, v_2, \dots, v_N . Skup grana \mathbf{E}_N sadrži grane $e_{12}, e_{23}, \dots, e_{N-1N}, e_{N1}$, gdje indeksi označavaju indekse rubnih čvorova određene grane. Numeriranje čvorova dogovorno se izvodi u smjeru kazaljke na satu. Model prstena \mathbf{R}_6 prema iznijetoj definiciji prikazuje slika 2.3.



Slika 2.3: Model prstena \mathbf{R}_6

Definicija modela prstena \mathbf{R}_N je općenita, ni po čemu vezana za telekomunikacijsku primjenu prstenaste strukture. Dodavanjem skupa prometnih zahtjeva \mathbf{D} prstenu \mathbf{R}_N , prsten postaje uredena trojka $\mathbf{R}_N = \{\mathbf{V}_N, \mathbf{E}_N, \mathbf{D}\}$. Skup zahtjeva \mathbf{D} sadrži K zahtjeva d_{ij} , od kojih je svaki definiran indeksom i izvorišnog čvora, indeksom j odredišnog čvora i zahtijevanim prometom. Uz uvjet $i < j$ stvara se pretpostavka o simetričnosti zahtjeva – budući da se zahtjev sa $j < i$ ne definira, pretpostavka je da su zahtjevi od v_i prema v_j i od v_j prema v_i identični. Bez uvjeta $i < j$, zahtjevi ne moraju biti simetrični.

Ovisno o korištenom modelu, moguće je dozvoliti ponavljanje zahtjeva između istih parova čvorova. U modelu koji je korišten u ovom radu prometni zahtjev d_{ij} je cijelobrojno definiran.

Da bi mreža bazirana na strukturi prstena mogla funkcionirati, potrebno je osigurati dovoljne kapacitete koji omogućavaju prijenos podataka prema definiranim zahtjevima. Prijenos zahtjeva kroz grane prstena moguć je samo ukoliko je kapacitet grane e_{ij} , $C(e_{ij})$ jednak ili veći od zbroja svih prometnih zahtjeva koji prolaze tom granom. Model prstena zbog toga nadopunjujemo kapacitetom grana $C(e_{ij})$, za sve grane u skupu \mathbf{E}_N .

Kako je prsten dvostruko povezana struktura, veza između svaka dva čvora može se uspostaviti korištenjem dva neovisna puta. Zbog jednostavnije analize, definiramo smjer kazaljke na satu (*clockwise direction*) i smjer suprotan smjeru kazaljke na satu (*counterclockwise direction*). Svaki zahtjev d_{ij} iz \mathbf{D} moguće je usmjeriti u jednom ili drugom smjeru. Isto tako moguće je i dijeljenje zahtjeva, pa se dio zahtjeva usmjerava u jednom, a dio u drugom smjeru. Svakom zahtjevu d_{ij} pridružujemo svojstvo smjera $x(d_{ij})$. Dogovorno definiramo vrijednost $x(d_{ij})=1$ u slučaju usmjeravanja zahtjeva u smjeru kazaljke na satu, odnosno $x(d_{ij})=0$ u slučaju usmjeravanja u smjeru suprotnom smjeru kazaljke na satu. Ukoliko je dozvoljeno dijeljenje zahtjeva $x(d_{ij})$ definira dio (postotak) zahtjeva koji se usmjerava u smjeru kazaljke na satu ($0 \leq x(d_{ij}) \leq 1$), dok se u smjeru suprotnom smjeru kazaljke na sati usmjerava $1-x(d_{ij})$ dio zahtjeva.

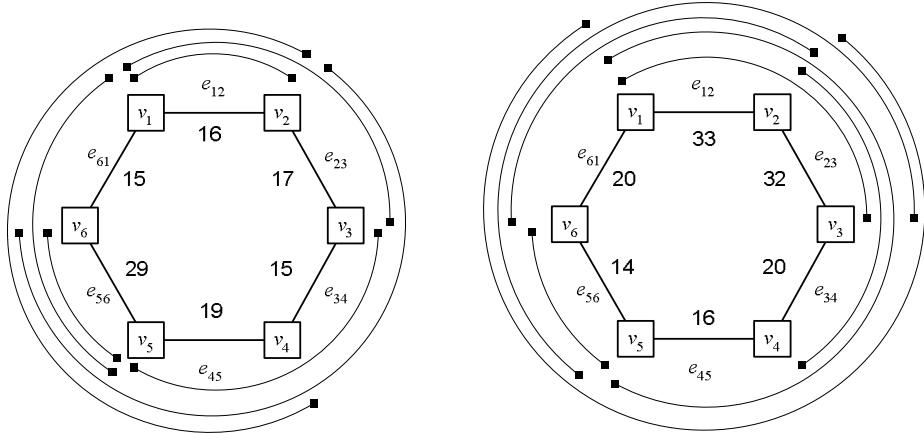
Prsten je osim skupovima \mathbf{V}_N , \mathbf{E}_N i \mathbf{D} karakteriziran još jednim bitnim svojstvom – kapacitetom prstena RC . Kapacitet prstena jednak je maksimalnom kapacitetu grana u prstenu. Kad se odredi kapacitet prstena sve grane u prstenu se dimenzioniraju na taj kapacitet. To se izvodi s ciljem omogućavanja zaštite u slučaju kvarova. Kapacitet prstena ne bi imao nikakvo posebno značenje u prstenastoj strukturi koja nije namijenjena omogućavanju zaštite prometa.

U tablici 2.1. je definiran skup prometnih zahtjeva D za primjer modela prstena R_6 .

Tablica 2.1: Prometni zahtjevi za model prstena

	izvoršni čvor	odredišni čvor	prometni zahtjev	smjer $x(d_{ij})$	
				primjer 1.	primjer 2.
d_{12}	v_1	v_2	5	1 (↻)	0 (↺)
d_{13}	v_1	v_3	7	1 (↻)	1 (↺)
d_{15}	v_1	v_5	11	0 (↺)	1 (↻)
d_{24}	v_2	v_4	4	0 (↺)	1 (↻)
d_{26}	v_2	v_6	10	1 (↻)	0 (↺)
d_{35}	v_3	v_5	5	1 (↻)	0 (↺)
d_{56}	v_5	v_6	4	1 (↻)	1 (↻)

Dva različita primjera usmjeravanja za taj model dana su na slici 2.4.



Slika 2.4: Različita usmjeravanja prometnih zahtjeva

Kapaciteti grana (suma prometnih zahtjeva koji se usmjeravaju kroz pojedine grane) za primjere usmjeravanja sa slike 2.4. dani su u tablici 2.2.

Može se uočiti razlika u maksimalnim zahtjevanim kapacitetima grana koji će omogućiti realizaciju tih usmjeravanja. Očito je da su zahtjevani kapaciteti pojedinih grana različiti. Kako je kapacitet prstena jednak maksimalnom kapacitetu grane u prstenu, kapaciteti prstena prema predloženim usmjeravanjima su 29, odnosno 33.

Postavlja se pitanje da li je, i kako, moguće izvesti usmjeravanje koje će smanjiti kapacitet prstena. Optimalno usmjeravanje i kapaciteti grana u tom slučaju definirani su u tablicama 2.3 i 2.4. Grafički je usmjeravanje prikazano na slici 2.5.

Tablica 2.2: Kapaciteti grana za primjere usmjeravanja

grana	primjer 1		primjer 2	
	zahtjevi	kapacitet	zahtjevi	kapacitet
e_{12}	d_{12}, d_{13}, d_{24}	16 (5+7+4)	$d_{13}, d_{15}, d_{26}, d_{35}$	33 (7+11+10+5)
e_{23}	d_{13}, d_{26}	17 (7+10)	$d_{13}, d_{24}, d_{15}, d_{35}, d_{12}$	32 (7+4+11+5+5)
e_{34}	d_{35}, d_{26}	15 (5+10)	d_{24}, d_{15}, d_{12}	20 (4+11+5)
e_{45}	d_{24}, d_{35}, d_{26}	19 (4+5+10)	d_{15}, d_{12}	16 (11+5)
e_{56}	$d_{56}, d_{15}, d_{26}, d_{24}$	29 (4+11+10+4)	d_{56}, d_{35}, d_{12}	14 (4+5+5)
e_{61}	d_{24}, d_{15}	15 (4+11)	d_{26}, d_{35}, d_{12}	20 (10+5+5)

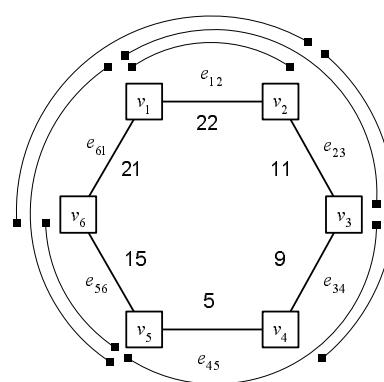
Dakle, optimalni kapacitet prstena RC iz primjera je 22. Određivanje optimalnog usmjeravanja u prstenu, s ciljem minimizacije kapaciteta tj. cijene prstena je osnovni problem u postupku dimenzioniranja jednostavnog prstena – problem se označava sa RDP (*Ring Dimensioning Problem*).

Tablica 2.3: Optimalno usmjeravanje

	izvorišni čvor	odredišni čvor	prometni zahtjev	Smjer
d_{12}	v_1	v_2	5	1 (\cup)
d_{13}	v_1	v_3	7	1 (\cup)
d_{15}	v_1	v_5	11	0 (\circlearrowleft)
d_{24}	v_2	v_4	4	1 (\cup)
d_{26}	v_2	v_6	10	0 (\circlearrowleft)
d_{35}	v_3	v_5	5	1 (\cup)
d_{56}	v_5	v_6	4	1 (\cup)

Tablica 2.4: Kapaciteti grana

grana	Optimalno usmjeravanje	
	zahtjevi	kapacitet
e_{12}	d_{12}, d_{13}, d_{26}	22 (5+7+10)
e_{23}	d_{13}, d_{24}	11 (7+4)
e_{34}	d_{35}, d_{26}	9 (5+10)
e_{45}	d_{24}, d_{35}, d_{26}	5 (4+5+10)
e_{56}	$d_{56}, d_{15}, d_{26}, d_{24}$	15 (4+11+10+4)
e_{61}	d_{24}, d_{15}	21 (4+11)



Slika 2.5: Optimalno usmjeravanje u prstenu

2.3. Dimenzioniranje jednostavnog prstena

Dimenzioniranje jednostavnog prstena je postupak koji obuhvaća određivanje usmjeravanja prometnih zahtjeva kroz prsten, te određivanje kapaciteta grana i kapaciteta prstena potrebnih za uspješnu realizaciju usmjeravanja tih zahtjeva.

Da bi se omogućio prijenos prometa prema definiranim zahtjevima mora biti zadovoljen uvjet da je kapacitet svake grane veći ili jednak sumi svih zahtjeva koji su usmjereni kroz granu.

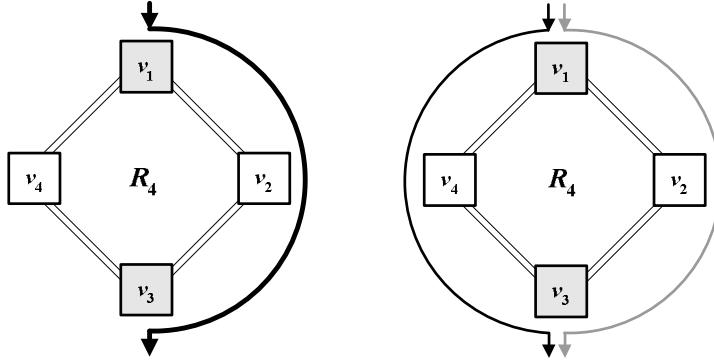
Svako planiranje mreže mora voditi računa o više parametara. S jedne je strane potrebno zadovoljiti sve zahtjeve, a s druge strane je interes učiniti to sa što manjim finansijskim ulaganjima. Pri dimenzioniranju prstena nije dovoljno odrediti proizvoljno usmjeravanje zahtjeva, pozbrnjati zahtjeve i odrediti kapacitete grana i prstena, već je nužno proanalizirati sve mogućnosti usmjeravanja u nastojanju da se smanji trošak izgradnje mreže.

Osnovna je pretpostavka da cijena izgradnje jednostavne prstenaste mreže direktno ovisi o kapacitetu prstena. Iako ta veza najčešće nije linearна, cilj je jasan – minimizirati kapacitet prstena. Promatramo li jednostavan prsten, tada troškove njegove izgradnje možemo podijeliti na fiksni trošak postavljanja optičkih kablova i ostale opreme, dok varijabilni trošak, kojeg je moguće reducirati pametnim planiranjem, predstavljaju sučelja na opremi – najbitniji dio tog troška su linijska sučelja koja ovise o kapacitetu prstena. Iz toga proizlazi ovisnost cijene prstena o njegovom kapacitetu.

Kako minimizirati cijenu izgradnje mreže? Budući da je cijena mreže proporcionalna kapacitetu prstena, problem se svodi na minimizaciju kapaciteta prstena. Jedini parametar kojim možemo utjecati na kapacitet grana je način usmjeravanja prometnih zahtjeva kroz prsten. Trebamo naći usmjeravanje koje će omogućiti prijenos prometnih zahtjeva uz najmanji kapacitet prstena, odnosno najmanji maksimalni kapacitet grane u prstenu. Minimizacija se može provesti samo kod dvosmjernih prstenova – kod jednosmjernih prstenova svi zahtjevi se moraju usmjeriti u istom smjeru, te nikakva smanjenja kapaciteta nisu moguća. Kako je svaki od zahtjeva dvosmjeren, on utječe na kapacitet svih grana u jednosmjernom prstenu, što znači da je kapacitet jednosmjernog prstena jednak sumi prometnih zahtjeva. Dakle, postupci dimenzioniranja prstenova i njihove optimizacije ne odnose se na jednosmrjne prstenove.

Uz definirano K zahtjeva postoji 2^K mogućih kombinacija usmjeravanja u prstenu ukoliko se ne omogući dijeljenje zahtjeva. Naime, svaki zahtjev može se usmjeriti u jednom od dva smjera, tako da se broj kombinacija svede na $2 \cdot 2 \cdot 2 \cdots 2 = 2^K$. Uz mogućnost dijeljenja zahtjeva broj kombinacija je beskonačan, odnosno ograničen ukoliko je dozvoljeno samo cjelobrojno dijeljenje zahtjeva. Cilj je odabrati optimalno usmjeravanje. Budući da se radi o eksponencijalnom rastu složenosti problema, jasno je da se za pronalaženje rješenja ne može efikasno iskoristiti enumeracijski algoritam koji bi proanalizirao sve kombinacije i pronašao optimum. Zbog toga se različitim relaksacijama problema pokušava suziti prostor pretraživanja uz nastojanje zadržavanja optimalnog rješenja unutar potprostora.

Neka je definiran prsten $\mathbf{R}_N = \{\mathbf{V}_N, \mathbf{E}_N, \mathbf{D}\}$. Skup \mathbf{D} neka sadrži $K=|\mathbf{D}|$ prometnih zahtjeva koje je potrebno usmjeriti kroz prsten. Smjer u kojem se zahtjev d_{ij} usmjerava definiran je varijabljom $x(d_{ij})$. Ovisno o mogućnostima dijeljenja zahtjeva varijabla $x(d_{ij})$ poprima vrijednosti $\{0, 1\}$ u slučaju kad dijeljenja zahtjeva nije omogućeno, odnosno vrijednosti iz skupa $[0, 1]$ kada je dijeljenje zahtjeva moguće. Primjer usmjeravanja bez dijeljenja i sa dijeljenjem zahtjeva dan je na slici 2.6.



Slika 2.6: Usmjeravanje bez dijeljenja i sa dijeljenjem zahtjeva

Za svaku granu e_{ab} (grana sa rubnim čvorovima v_a i v_b) definiramo skup $A(e_{ab})$ koji sadrži sve prometne zahtjeve koji, ukoliko su usmjeravani u smjeru kazaljke na satu, prolaze granom e_{ab} . Skup $B(e_{ab})$ sadrži ostale zahtjeve, tj. zahtjeve koji usmjeravani u smjeru kazaljke na satu ne prolazi granom e_{ab} . Budući da vrijedi $i < j$ za svaki d_{ij} , skup $A(e_{NI})$ će uvijek biti prazan jer ne postoji zahtjev koji se kroz tu granu usmjerava u smjeru kazaljke na satu. Skupove možemo zapisati kao:

$$\begin{aligned} A(e_{ab}) &= \left\{ \forall d_{ij} \in D : (i \leq a) \cup (j \geq b) \right\} \\ B(e_{ab}) &= D \setminus A(e_{ab}) \end{aligned} \quad (2.1)$$

Kapacitet pojedine grane uz tako definirane skupove odreduje se izrazom:

$$C(e_{ab}) = \sum_{d_{ij} \in A} d_{ij} \cdot x(d_{ij}) + \sum_{d_{ij} \in B} d_{ij} \cdot (1 - x(d_{ij})) \quad (2.2)$$

Prvi dio izraza predstavlja kapacitet koji je potreban za prijenos zahtjeva koji se prenose granom u smjeru kazaljke na satu, dok drugi dio izraza predstavlja kapacitet potreban za prijenos zahtjeva u smjeru suprotnom od kazaljke na satu. Smjerove i zahtjeve koji se usmjeravaju određujemo prema vrijednosti varijable $x(d_{ij})$.

Kapacitet prstena izražavamo, prema definiciji:

$$RC = \max_{e_{ab} \in E} C(e_{ab}) \quad (2.3)$$

Problem optimizacije kapaciteta prstena u literaturi se naziva problemom dimezioniranja prstena – RDP (*Ring Dimensioning Problem*). Razmatranjem predloženog modela prstena R_N funkciju cilja općenitog optimizacijskog problema minimizacije kapaciteta mreže izgrađene od jednostavnog prstena možemo zapisati kao:

RDP:

$$\min RC = \min \left(\max_{e_{ab} \in E} C(e_{ab}) \right) \quad (2.4)$$

Predloženi model može se jednostavno realizirati kao matematički program. Varijable odluke odgovaraju varijablama $x(d_{ij})$, dok se funkcija cilja definira prema RDP formulaciji. Ograničenja koja se uvode ovise o tome da li je omogućeno dijeljenje zahtjeva ili ne. Ukoliko je dijeljenje omogućeno, varijable odluke mogu poprimiti bilo koju vrijednost između 0 i 1, te se radi o čistom linearном programu koji se može riješiti u polinomnom vremenu. Ako dijeljenje nije dozvoljeno radi se o cijelobrojnem programu (*integer program*). Alternativno, moguće je dozvoliti dijeljenje zahtjeva uz uvjet da je to dijeljenje cijelobrojno – u tom slučaju također se radi o cijelobrojnem matematičkom programu.

Jasno je da se minimalni kapacitet dobiva u slučajevima s dozvoljenim dijeljenjem zahtjeva. Svako ograničenje – dozvoljeno dijeljenje zahtjeva u nekim diskretnim koracima, cijelobrojno dijeljenje zahtjeva ili pak zabrana dijeljenja zahtjeva – rezultira podoptimalnim rješenjem problema.

Međutim, ta rješenja su optimalna za realni slučaj mreže – moguće ih je realizirati u praksi, te u tom slučaju predstavljaju najbolje odnosno najjeftinije rješenje.

Korištenjem jezika za opis matematičkih problema AMPL [2] RDP možemo zapisati kao:

```

1: set DEMANDS;
2: param source {DEMANDS};
3: param destination {DEMANDS};
4: param demand {DEMANDS};
5: param N;

6: var X {i in DEMANDS} >=0, <=1;
7: var RingCap;

8: data "rdp1.dat";

9: param A {DEMANDS,1..N};
10: let {i in DEMANDS, j in 1..N}
11:     A[i,j]:= (if (source[i]<=j) && (destination[i]>=j+1) then 1 else 0);

12: minimize RDP: RingCap;
13: subject to Gr {i in 1..N}:
14:     RingCap >= sum{j in DEMANDS}demand[j]*(X[j]*A[j,i]+(1-X[j])*(1-A[j,i]));

```

Prvih pet redaka služi za definiranje skupa zahtjeva i parametara pojedinog zahtjeva. Definira se i parametar N koji označava veličinu mreže (broj čvorova/grana). U linijama 6 i 7 definiraju se varijable – vrijednosti koje solver mijenja tijekom određivanja optimuma – varijabla X se definira za svaki zahtjev i identična je varijabli x uvedenoj u ovom odjeljku u model jednostavnog prstena. Definira se i varijabla $RingCap$ koja predstavlja vrijednost kapaciteta prstena. Linija 7 služi za definiranje podataka kojima se model definira – skup zahtjeva i svi definirani parametri nalaze se u podatkovnoj datoteci. Linije 9, 10 i 11 služe za definiranje pomoćnog parametra A . Parametar se definira za svaki zahtjev za svaku granu – vrijednost parametra je 1 ukoliko zahtjev usmjeravan u smjeru kazaljke na satu prolazi pojedinom granom, odnosno 0 ukoliko ne prolazi. Parametar A ekvivalentan je definiranju skupova \mathbf{A} i \mathbf{B} u modelu jednostavnog prstena.

Linije 12,13 i 14 predstavljaju optimizacijski dio modela – postavlja se zahtjev za minimizacijom varijable $RingCap$, uz uvjet da je ona veća ili jednaka kapacitetu pojedine grane (izraz u liniji 14 predstavlja kapacitet grane, tj. sumu doprinosa svih prometnih zahtjeva grani i).

Osim samog matematičkog modela u AMPL-u, za rješavanje je potrebno definirati i podatke o prstenu. Uz veličinu prstena nužno je definirati prometne zahtjeve. Primjer datoteke s podacima koji se koriste kao ulaz u model prikazan je u nastavku. Radi se o zahtjevima za primjer prstena iz odjeljka 2.2.

```

param  N := 6;
param: DEMANDS:      source  destination    demand  :=
1      1      2      5
2      1      3      7
3      1      5     11
4      2      4      4
5      2      6     10
6      3      5      5
7      5      6      4;

```

Tako definiran matematički problem moguće je optimalno riješiti korištenjem nekog od alata za rješavanje matematičkih problema (npr. CPLEX – posjeduje mogućnost rješavanja standardnih linearnih programa, ali i cjelobrojnih programa). U AMPL okruženju dovoljno je pokrenuti optimizaciju naredbom solve u komadnom retku.

Slučaj s dozvoljenim dijeljenjem zahtjeva je čisti linearni program i može se riješiti egzaktno korištenjem standardnih algoritama. Slučaj bez dijeljenja zahtjeva je cjelobrojni linearni program čije se rješenje može bazirati na kombiniranju poznatih algoritama za prilagodbu rješenja linearog programa cjelobrojnom problemu. Dodatni problem u modelu s dozvoljenim dijeljenjem zahtjeva predstavljaju realna ograničenja koja ne omogućavaju dijeljenje zahtjeva u proizvoljnim koracima ili zahtijevaju cjelobrojna dijeljenja zahtjeva. Za rješavanje realnih modela najčešće je potrebno prilagoditi linearni program uvodenjem dodatnih uvjeta.

U nastavku su opisane efikasne metode za rješavanje RDP uz mogućnost dijeljenja zahtjeva (RDPwS – *RDP with splitting*), te heuristike za rješavanje problema dimenzioniranja prstena bez dijeljenja zahtjeva (RDPw/oS – *RDP without splitting*).

2.3.1. Dimenzioniranje s mogućnošću dijeljenja prometnih zahtjeva

Kao što je već napomenuto RDPwS se definira kao linearni program, te je rješiv korištenjem standardnih kombinatoričkih algoritama [3,4]. Rješenje RDPwS-a je optimalno. Svako novo ograničenje koje uvodi realniju sliku modela (ograničene mogućnosti dijeljenja zahtjeva po koracima (npr. cjelobrojno dijeljenje)) rezultira odstupanjem od optimalnog rješenja tj. dovodi do podoptimalnog rješenja problema. Ipak, mreža bazirana na takvima rezultatima je izvediva u realnosti, te predstavlja 'realni' optimum. Klasu RDPwS problema dijelimo na standardne RDPwS probleme i RDPwS probleme sa ograničenjima (cRDPwS – *constrained RDPwS*).

Standardni RDPwS

Prema modelu za definiranje RDP (2.4), RDPwS možemo zapisati kao:

RDPwS:

$$\min RC = \min \left(\max_{e_{ab} \in E} C(e_{ab}) \right) \quad (2.4a)$$

uz uvjet:

$$0 \leq x(d_{ij}) \leq 1 \quad (2.5)$$

Problem se kao matematički program zapisuje prema zapisu iz prošlog odjeljka. Rezultat – kapacitet prstena koji se dobiva korištenjem CPLEX-a jest 18.5.

Budući da se radi o linearном programu bez cjelobrojnih ograničenja, određivanje optimuma moguće je korištenjem egzaktnog algoritma. U literaturi [5] predložen je algoritam EXACT.

EXACT algoritam

Algoritam je u principu vrlo jednostavan. Svi prometni zahtjevi inicijalno se usmjere u smjeru kazaljke na sati ($x(d_{ij})=1$), a zatim se za svaki pojedini zahtjev provjerava da li se usmjeravanjem jednog njegovog dijela u suprotnom smjeru može smanjiti kapacitet prstena. Ukoliko je to moguće, mijenja se varijabla $x(d_{ij})$, te se postupak nastavlja za preostale zahtjeve. Kako bi se olakšala analiza rezultata, u svakoj iteraciji definira se pomoćna varijabla odluke $x_k(d_{ij})$, gdje k označava broj iteracije.

Skupovi $A(e_{ab})$ i $B(e_{ab})$ koji sadrže zahtjeve koji se kroz pojedine grane usmjeravaju u smjeru kazaljke na sat ili smjeru suprotnom od smjera kazaljke na sat definiraju se prema [5].

Sva inteligencija algoritma svodi se na provjeru maksimalnih kapaciteta grana kroz koje pojedini zahtjev prolazi pri usmjeravanju u smjeru kazaljke na satu a i maksimalnih kapaciteta grana kroz koje prolazi pri usmjeravanju u suprotnom smjeru b . Ukoliko je a veće od b , znači da je utjecaj na kapacitet prstena veći pri usmjeravanju u smjeru kazaljke na satu, te je moguće preusmjeriti dio ili cijeli zahtjev u suprotnom smjeru kako bi se kapacitet prstena smanjio. Budući da je cilj izjednačiti utjecaje zahtjeva u oba smjera ($a=b$), dio zahtjeva se usmjerava u suprotnom smjeru. Ukoliko utjecaj određenog zahtjeva na kapacitet prstena nije presudan, moguće je da se cijeli zahtjev usmjeri u smjeru suprotnom smjeru kazaljke na satu.

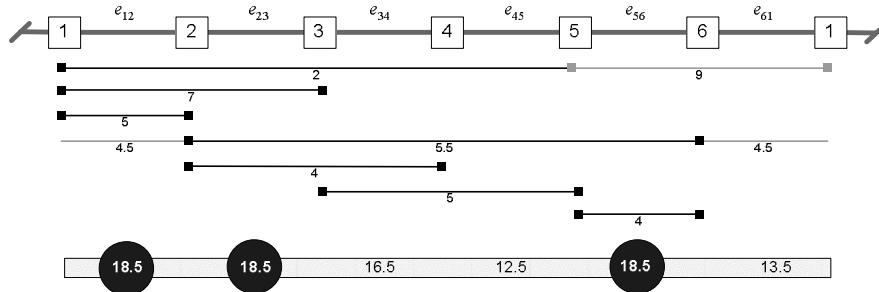
Princip rada algoritma prikazan je pseudokodom u nastavku:

```

algoritam EXACT;
    za  $\forall d_{ij}$  postavi  $x_0(d_{ij})=1$ 
    k=0;
    za  $\forall d_{ij}$ 
        k=k+1;
        a=max kapacitet grana kroz koje prolazi  $d_{ij}$  u smjeru  $\cup (e_{ab} \in A(e_{ab}))$ ;
        b=max kapacitet grana kroz koje prolazi  $d_{ij}$  u smjeru  $\cup$ ;
        c=a-b;
        ako  $c > 0$ 
            d=min(c/2,  $d_{ij}$ )
             $x_k(d_{ij}) = x_{k-1}(d_{ij}) - d/d_{ij}$ ;
            za sve ostale zahtjeve  $d_{xy}$  ( $x \neq i, y \neq j$ )
                 $x_k(d_{xy}) = x_{k-1}(d_{xy})$ 
    kraj

```

Rezultati dobiveni korištenjem algoritma EXACT su optimalni ukoliko se prije realizacije algoritma zahtjevi pravilno sortiraju – ukoliko za zahtjeve d_{i1j1} i d_{i2j2} vrijedi $i1 < i2$ onda prvi zahtjev prethodi drugom, a ukoliko je $i1 = i2$ onda prvi zahtjev prethodi drugom ukoliko vrijedi $j1 > j2$. Detaljna analiza i opis algoritma dana je u [5]. Rješenja koja daje algoritam mogu se iskoristiti u aproksimacijskom algoritmu za rješavanje RDPwoS 2-OPT. Prikaz funkciranja algoritma na primjeru prstena zadanog u odjeljku 2.2 prikazuje slika 2.7.



Slika 2.7: EXACT metoda – rezultat usmjeravanja

Na slici su redom navedeni zahtjevi sortirani prema definiranom kriteriju. Maksimalni kapaciteti grana su 18.5 jedinica. Zahtjevi d_{15} i d_{26} su, kao što je na slici vidljivo, dijeljeni i prenose se u oba smjera.

RDPwS sa ograničenjima

Cjelobrojno dijeljenje zahtjeva

Pri usmjeravanju s mogućnošću dijeljenja zahtjeva, tj. njihovog usmjeravanja u oba smjera unutar prstena, jedno od realnih ograničenja može biti zahtjev za cjelobrojnim dijeljenjem zahtjeva. Dakle, ukoliko se npr. zahtjev od 4 jedinice želi podijeliti dozvoljena dijeljenja su (0,4), (1,3), (2,2), (3,1) i (4,0).

Kao i kod RDPwS funkcija cilja je identična onoj osnovnog RDP problema (2.4):

cRDPwS:

$$\min RC = \min \left(\max_{e_{ab} \in E} C(e_{ab}) \right) \quad (2.4b)$$

Uvjeti koji se kod cRDPwS postavljaju su:

$$\begin{aligned} 0 \leq x(d_{ij}) \leq 1 \\ x(d_{ij}) \cdot d_{ij} \text{ je cjelobrojno} \end{aligned} \quad (2.5a)$$

Definiranim uvjetima dodatno se ograničavaju vrijednosti varijable odluke $x(d_{ij})$ na one kojih umnožak s veličinom prometnog zahtjeva (cijeli broj) jest cijeli broj.

RDPwS je u stvari relaksacija cRDPwS-a. cRDPwS spada u klasu cjelobrojnih matematičkih problema i nema garancije da se može riješiti u polinomnom vremenu.

Uvjet cjelobrojnog dijeljenja zahtjeva zahtjeva promjenu matematičkog programa. Matematički model za rješavanje cRDPwS-a može se u AMPL-u zapisati kao:

```

1: set DEMANDS;
2: param source {DEMANDS};
3: param destination {DEMANDS};
4: param demand {DEMANDS};
5: param N;

6: var X {i in DEMANDS} >=0, <=1;
7: var RingCap;
8: var LinkCap{1..N};
9: var Parts {DEMANDS} integer;

10: data "rdp1.dat";

11: param A {DEMANDS,1..N};
12: let {i in DEMANDS, j in 1..N}
13:   A[i,j]:= (if (source[i]<=j) && (destination[i]>=j+1) then 1 else 0);

14: PartDem {i in DEMANDS}:
15:   Parts[i]=demand[i]*X[i];

16: minimize RDP: RingCap;
17: subject to Gr {i in 1..N}:
18:   RingCap >= sum{j in DEMANDS} (Parts[j]*A[j,i]+demand[j]*(1-X[j])) ;

```

Model se mnogo ne razlikuje od prije prikazanog standarnog modela za RDPwS. Jedinu razliku predstavlja definiranje dodatne varijable *PartDem* za svaki zahtjev. Varijabla *PartDem* za određeni zahtjev poprima vrijednost koja je jednaka dijelu zahtjeva koji se prenosi u smjeru kazaljke na satu. Kako je pri deklaraciji varijable navedeno da ona mora biti cjelobrojna (*integer* ključna riječ u 9. retku), *solver* će minimizirati kapacitet ali uz uvjet cjelobrojnog dijeljenja zahtjeva.

Jedan od algoritama koji uspješno rješavaju problem cjelobrojnog dijeljenja zahtjeva (tzv. *integer splitting problem*) je INDES, iznesen u [6]. Autori su pokazali da rezultati algoritma maksimalno odstupaju za jednu jedinicu prometa od optimalnog rezultata pri dijeljenju zahtjeva bez ograničenja.

Dijeljenje zahtjeva prema razinama

Dijeljenje prometnih zahtjeva prema razinama predstavlja poseban slučaj cjelobrojnog dijeljenja zahtjeva. Vrlo čest slučaj u praksi, te na području dimenzioniranja jednostavnog prstena uz RDPwoS najznačajniji, slučaj je u kojem je dijeljenje zahtjeva dozvoljeno u točno definiranim koracima. Primjer je hijerarhija transportnih modula u SDH transportnom sustavu.

Prilagodbom INDES i EXACT algoritama može se realizirati dimenzioniranje prstena uz dijeljenje zahtjeva prema razinama.

2.3.2. Dimenzioniranje bez mogućnosti dijeljenja prometnih zahtjeva

Problem dimenzioniranja prstena uz zabranjeno dijeljenje zahtjeva intuitivno je poprilično jasan. Svaki od K prometnih zahtjeva iz skupa \mathbf{D} može se usmjeriti jednim od dva ponuđena smjera – u smjeru kazaljke na satu ili u smjeru suprotnom smjeru kazaljke na satu. Broj mogućih kombinacija usmjeravanja u tom slučaju je 2^K . Budući da broj kombinacija raste eksponencijalno s brojem zahtjeva, jasno je da određivanjem i analizom svih kombinacija ne možemo efikasno riješiti problem pronašlaska optimalnog usmjeravanja u realnom vremenu. Kako je u postupku dizajna

prstenastih mreža potrebno mnogo puta obaviti postupak dimenzioniranja, nužno je implementirati metode koje će rješavati probleme (optimalno ili podoptimalno) u realnom vremenu!

RDPw/oS kao linearni matematički model definiramo uz ograničavanje mogućih vrijednosti varijable $x(d_{ij})$ na 0 i 1. Funkcija cilja je identična općenitom RDP (2.4):

RDPw/oS:

$$\min RC = \min \left(\max_{e_{ab} \in E} C(e_{ab}) \right) \quad (2.4c)$$

No, uvjet vezan uz varijable odluke sada se definira kao:

$$x(d_{ij}) \in \{0,1\} \quad (2.5b)$$

Vrijednost varijable $x(d_{ij})$ jednaka 1 označava usmjeravanje zahtjeva d_{ij} u smjeru kazaljke na satu, dok vrijednost 0 označava usmjeravanje u smjeru suprotnom smjeru kazaljke na satu.

Prema tome, matematički program za RDPw/oS spada u klasu cjelobrojnih matematičkih programa i za njegovo optimalno rješavanje ne može se garantirati postojanje algoritma koji će to obaviti u polinomnom vremenu. Sam program identičan je modelu opisanom u odjeljku 2.3, uz jedinu razliku u definiranju varijable X u 6. retku modela. Ukoliko se varijabla X deklarira kao:

```
...
6: var X {i in DEMANDS} >=0, <=1 integer;
...
```

radit će se o modelu za rješavanje RDPw/oS – ključna riječ *integer* u AMPL-u uvjetuje cjelobrojnost varijabli. Rezultati dobivenim CPLEX-om daju optimalni kapacitet za primjer jednostavnog prstena iznosa 22.

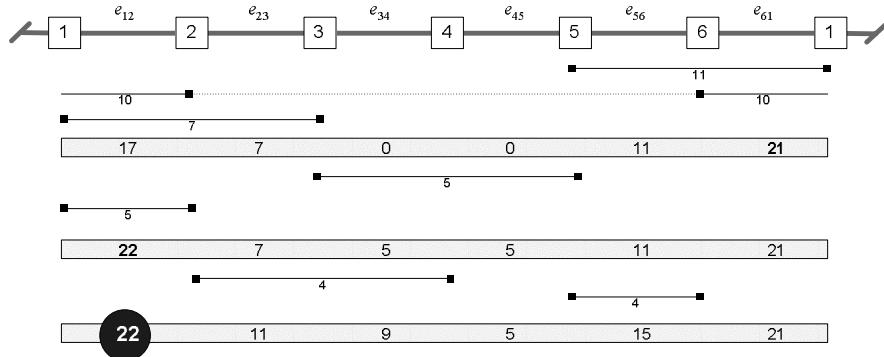
Heurističke metode za rješavanje RDPw/oS dijele su u dvije grupe: pohlepne (*greedy*) metode i adaptivne težinske metode (*weight-based*). Pohlepni postupci su kratkovidni i u pravilu daju podoptimalna rješenja lošija od onih koje daju adaptivne metode. Njihova je složenost nešto niža, a u određenim slučajevima daju relativno dobre rezultate. Stoga je njihova primjena u praksi česta. Osim heurističkim metodama, RDPw/oS se može efikasno rješavati i metodama koje se baziraju na rješenjima RDPwS problema – dobivena rješenja se prilagodavaju uvjetima o zabranjenom dijeljenju zahtjeva – ovisno o načinu na koji je pojedini zahtjev dijeljen donosi se odluka o smjeru u kojem će ga se usmjeriti.

Pohlepne metode

Princip funkcioniranja pohlepnih metoda je jednostavan [7]. Zahtjevi iz skupa \mathbf{D} redom se usmjeravaju u smjeru koji manje povećava kapacitet prstena. Dakle, za neki zahtjevi d_{ij} obavi se usmjeravanje u smjeru kazaljke na satu i usmjeravanje u smjeru suprotnom od smjera kazaljke na satu, te se za svaki od slučajeva proračunaju trenutni kapaciteti grana i odredi kapacitet prstena. Ovisno o tome za koji slučaj je kapacitet prstena niži definira se vrijednost varijable $x(d_{ij})$.

Postupak će ovisiti o redoslijedu analize zahtjeva – najbolji rezultati dobivaju se ukoliko se zahtjevi sortiraju prema samim vrijednostima od najvećeg do najmanjeg. Alternativno se zahtjevi mogu sortirati prema udaljenosti čvorova i drugim kriterijima.

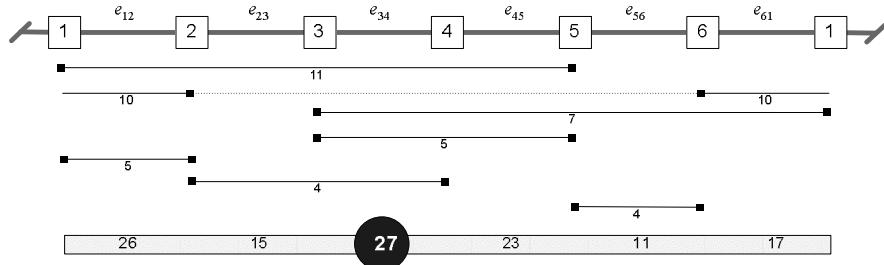
Metoda *One-phase greedy* svodi se na sortiranje zahtjeva od najvećeg prema najmanjem, te na opisano određivanje usmjeravanja zahtjeva. Najveći (prije se usmjerava) zahtjev se usmjerava u smjeru kojim prolazi kroz manji broj grana. Metoda *Two-phase greedy* je u potpunosti identična, osim razlike u prvom koraku u kom se najveći prometni zahtjev usmjerava u smjeru u kojem prolazi veći broja grana. Zbog različitog pristupa u prvom koraku rezultati se mogu prilično razlikovati. Slika 2.8. prikazuje postupak za metodu *OneGreedy* – zahtjevi su sortirani i usmjeravaju se tako da minimalno povećaju trenutni kapacitet prstena. Metoda je primijenjena na model prstena iz poglavlja 2.2.



Slika 2.8: OneGreedy metoda - postupak

U slučaju da zahtjev usmjeravanjem u bilo kojem smjeru jednako utječe na povećanje kapaciteta, usmjerava ga se u smjeru u kojem treba proći manje grane (primjer zahtjeva d_{26}). Metoda rezultira maksimalnim kapacitetom grane e_{12} od 22 jedinica, što odgovara optimalnom kapacitetu.

Rezultat metode TwoGreedy prikazan je na slici 2.9.



Slika 2.9: TwoGreedy metoda - postupak

Najveći zahtjev se usmjerava u smjeru suprotnom od smjera koji se definira u *OneGreedy* metodi. Naravno da to utječe na usmjeravanje ostalih zahtjeva – konačni rezultat je maksimalni kapacitet grane e_{34} od 27 jedinica. Očito je da ova metoda daje lošiji rezultat i to za 5 jedinice, što je više od 20% optimalnog rješenja. Iz tog najjednostavnijeg primjera može se zaključiti da kod pohlepnih metoda nema nikakve garancije kvalitete rezultata, te je njihova iskoristivost upitna. Prema rezultatima koje metode daju na različitim modelima prstena zaključuje se o smislenosti njihovog korištenja (zaključak je dan u odjeljku 2.6).

Adaptivne težinske metode

Za razliku od kratkovidnog pristupa pohlepnih metoda, adaptivne metode prilagođavaju težinske parametre grana u prstenu ovisno o stanju u mreži i posebnim ograničenjima dizajna. Zahtjevi se potom usmjeravaju u smjeru manje težine. Svakoj grani e_{ij} iz skupa E pridružuje se težinski parametar $w(e_{ij})$. Težine se po potrebi normaliziraju tako da je suma težina svih grana jednaka 1.

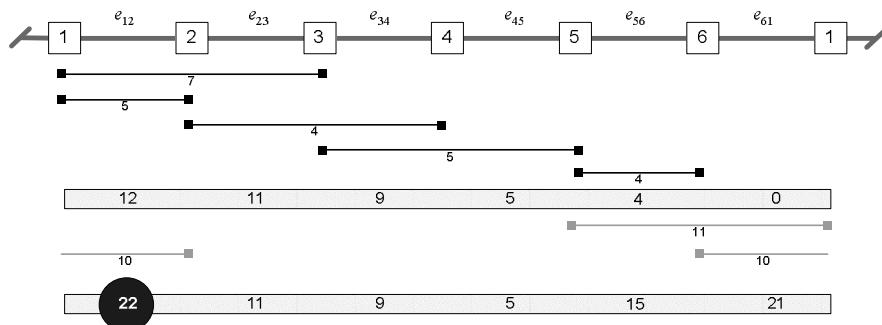
Najjednostavniji primjeri težinske metode su postupci u kojima se zahtjevi usmjeravaju u smjeru kojim trebaju proći manji broj grana (tzv. *min-hop* pristup u kojem se svim granama pridijele jednakе težine npr. $w(e_{ij})=1/N$ gdje je N broj čvorova odnosno grana) ili pak je određenim zahtjevima zabranjeno korištenje određenih grana (zabranjeni smjer/grana ima težinu 1 ili mnogo višu od ostalih).

U literaturi često prezentirana težinska metoda je *Dual-ascent* metoda [7]. U toj metodi težine grana se prilagodjavaju nakon svake iteracije, te se na temelju toga obavlja usmjeravanje. Rezultati će u pravilu biti bolji od rezultata dobivenih pohlepnim metodama.

RDPwS bazirane metode

2-OPT je jedna od metoda dimenzioniranja prstena bez mogućnosti dijeljenja zahtjeva bazirana na EXACT algoritmu za rješavanje RDPwS [5].

Algoritam se svodi na određivanje utjecaja povećanja kapaciteta u slučaju da se zahtjev u potpunosti usmjeri jednim ili drugim smjerom. Odluka se donosi tako da se definira smjer koji rezultira nižim kapacitetom prstena. Algoritam je vrlo jednostavan i brz, te daje dobre rezultate. Rezultantno usmjeravanje dobiveno 2-OPT metodom je optimalno i identično onom dobivenom metodom *OneGreedy*. Sam postupak kojim se dolazi do rješenja je drugačiji i bazira se na rješenju EXACT metode koje prikazuje slika 2.7. Odrede se kapaciteti grana koje uvjetuju zahtjevi koji se ne dijeli, te se analiziraju dijeljeni zahtjevi – određuje se smjer kojim zahtjev bez dijeljenja manje utječe na kapacitet, te se na osnovu tog kriterija definira rezultantni smjer za, u EXACT metodi, dijeljene zahtjeve.



Slika 2.10: 2-OPT metoda - postupak

Slika 2.10 prikazuje postupak koji se koristi za prilagodbu rješenja EXACT metode u izvođenju 2-OPT metode. Označeni su privremeni kapaciteti grana koji su odredeni usmjeravanjem zahtjeva koje EXACT metoda nije dijelila. Nakon toga analiziraju se EXACT metodom dijeljeni zahtjevi i donosi se odluka o smjeru. Rezultat je u ovom primjeru optimalan.

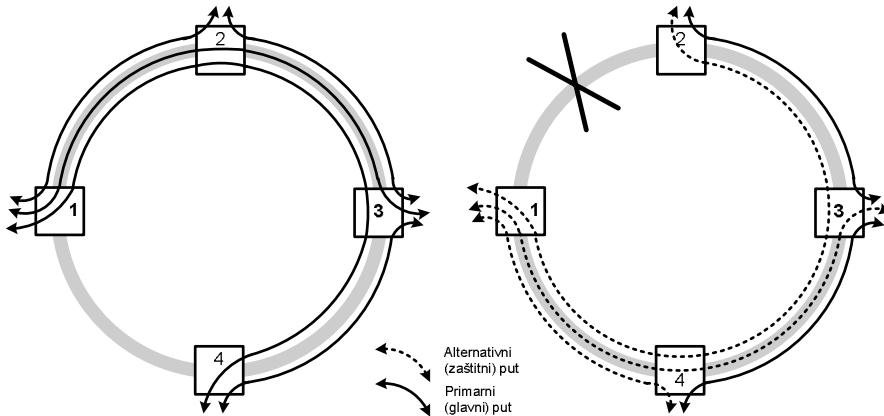
2.4. Zaštitni mehanizmi u prstenu

Model prstena i postupci dimenzioniranja opisani u ovom poglavlju nigdje eksplisitno ne vode računa o mogućnosti zaštite prometnih zahtjeva u slučaju kvarova. Prstenasta struktura omogućava uspostavu dva neovisna puta između dva čvora, te na taj način nudi mogućnost zaštite prometa. Međutim, dimenzioniranje prstenova svodi se na usmjeravanje zahtjeva bez analize vezane uz zaštitu prometa. Bez obzira na podopterećenost pojedinih grana u optimalno ili podoptimalno dimenzioniranim prstenovima, svaka se grana u prstenu dimenzionira na maksimalni kapacitet, tj. kapacitet prstena. Razlog tome je upravo zaštita prometa – ukoliko grane ne bi bile tako dimenzionirane, u određenim slučajevima ne bi bilo moguće izvesti zaštitu prometa.

Bitno je naglasiti da se zaštita nikad ne izvodi na jednostrukom prstenu – bez obzira da li se radi o jednosmjernom (*undirectional ring* – UR) ili dvosmjernom prstenu (*bidirectional ring* – BR). Na jednosmjernom jednostrukom prstenu efikasnu zaštitu za dvosmjernu komunikaciju između čvorova nije uopće moguće izvesti, dok bi kod dvosmjernog prstena to i bilo moguće, ali se u praksi ne koristi. Da bi se realizirala zaštita nužno je osim primarnog prstena definirati i zaštitni prsten.

Postoje dva osnovna principa zaštite u prstenu – zaštita na razini puta i zaštita na razini grane na kojoj se dogodio kvar.

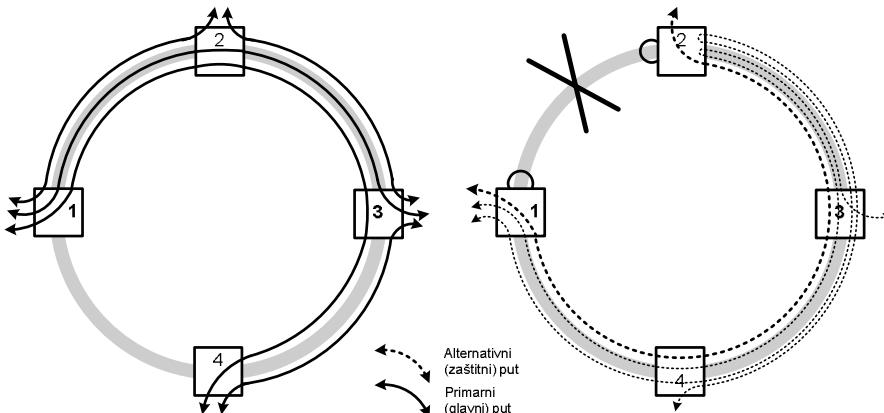
U slučaju kvara u prstenu sa zaštitom na razini puta svaki zahtjev koji je pogoden kvarom se usmjeri u alternativnom smjeru. Konceptualno je prikaz zaštite na razini puta dan na slici 2.11.



Slika 2.11: Koncept zaštite na razini puta u prstenu

U primjeru na slici su prepostavljeni dvosmjerni zahtjevi – komunikacija između čvorova obavlja se u oba smjera, a u ovom trenutku nije bitno na koji način se komunikacija odvija. U slučaju kvara na nekoj od grana (u primjeru grani između čvorova 1 i 2), svi prometni zahtjevi koji prolaze tom granom presumjeravaju se na alternativni put. Alternativni put obuhvaća sve grane koje primarni put nije obuhvatio. Zahtjevi koji su preusmjereni prikazani su crtkanom linijom.

Drugi princip zaštite prometa u prstenu je zaštita na razini multipleksne dionice tj. grane, pri čemu se dio prstena pogoden kvarom 'izbací' iz prstena i zaobide. Koncept je prikazan na slici 2.12.



Slika 2.12: Koncept zaštite na razini dionice u prstenu

Dok se zaštita na razini dionice/linije u standardnim mrežama može jednostavno obaviti korištenjem najkratčeg alternativnog puta koji zaobilazi dionicu u kvaru, u prstenastim mrežama ne preostaje drugo nego koristiti preostale grane prstena. U primjeru na slici prikazani su samo preusmjereni zahtjevi, dok zahtjevi između para čvorova 2 i 3, odnosno 3 i 4 nisu prikazani zbog jasnoće prikaza.

Rubni čvorovi grane u kvaru detektiraju kvar, te obavljaju tzv. *loopback* preusmjeravanje prometa, a ostali čvorovi primaju signalizacijsku informaciju te na osnovu nje prilagodavaju unutarnju logiku usmjeravanja. Na primjer, zahtjev između čvorova 3 i 1 usmjerava se kao i prije kvara, prema čvoru 2. Čvor 2 je već prije detektirao kvar na sljedećoj grani, te preusmjerava zahtjev natrag prema čvoru 3, koji zahtjev dalje prosljeđuje prema odredišnom čvoru 1. Preusmjeravanje zahtjeva između čvorova 1 i 2 svodi se na prijenos zahtjeva putem kao i kod zaštite na razini puta.

2.4.1. Iskorištenost kapaciteta prstena pri zaštitnom preusmjeravanju zahtjeva

Opisani koncepti zaštite u praksi se izvode korištenjem različitih vrsta prstenova. Iako se svaki model zaštite može iskoristiti u svim modelima prstenova, u praktičnoj primjeni izdvojena su dva

osnovna pravila. Tako se zaštita na razini puta koristi u zaštitnim jednosmjernim prstenovima, dok je zaštita na razini multipleksne dionice glavno obilježje zaštitnih dvosmjernih prstenova.

U odjeljku 2.3 analizirane su mogućnosti optimalnog usmjeravanja prometnih zahtjeva u prstenovima. Zaključeno je da u jednosmjernim prstenovima ne postoji mogućnost smanjenja zahtijevanih kapaciteta primjenom različitih načina usmjeravanja, te da je kapacitet jednosmjernog prstena uvijek jednak zbroju prometnih zahtjeva koje je potrebno prenijeti prstenom. S druge strane, u dvosmjernim prstenovima se različitim usmjeravanjem zahtjeva može utjecati na kapacitet prstena.

Već je napomenuto da izvedba efikasne zaštite prometa nije moguća bez korištenja zaštitnog prstena. Dakle, postojećem jednosmjernom ili dvosmjernom prstenu, s ciljem postizanja mogućnosti efikasne zaštite prometa, dodaje se kapacitetom jednak jednosmjeren odnosno dvosmjeren prsten. Odmah postaje jasno da se radi o popriličnom 'rasipanju' kapacitetima, što je ujedno jedan od glavnih nedostataka prstenastih zaštitnih topologija u odnosu na alternativne *mesh* topologije.

Bez obzira na vrstu prstena, radni, ali i zaštitni prsten dimenzionirani su na optimalni kapacitet. U najgorem slučaju sav kapacitet će biti iskorišten zbog preusmjeravanja uslijed kvara na nekoj od grana u mreži. S druge pak strane, u normalnom režimu rada u kojem nema nikakvih kvarova zaštitni kapacitet će biti u potpunosti neiskorišteni. Kako bi se ipak postigle neke uštede i smanjila neiskorištenost zaštitnih kapaciteta, postoji mogućnost iskoristavanja tih kapaciteta za prijenos prometnih zahtjeva nižeg prioriteta, koji će se u slučaju kvara možda izgubiti.

Prijenos zahtjeva nižeg prioriteta efikasniji je u modelu zaštite na razini puta. U tom je slučaju moguće da pojedini zahtjevi nižeg prioriteta i dalje budu prenošeni, dok se kod zaštite na razini dionice svi zahtjevi nižeg prioriteta gube (zbog specifičnog načina funkciranja tog oblika zaštite).

Iz svega navedenog proizlazi, da je korištenjem opisanih principa zaštite, dimenzioniranje kompletne zaštitne strukture prstena određeno dimenzioniranjem jednostavnog prstena. Ukoliko uspijemo naći optimalan kapacitet jednostavnog prstena, zaštitni prsten će također biti optimalno dimenzioniran identičnim kapacitetom.

2.5. Prstenaste mrežne strukture u primjeni

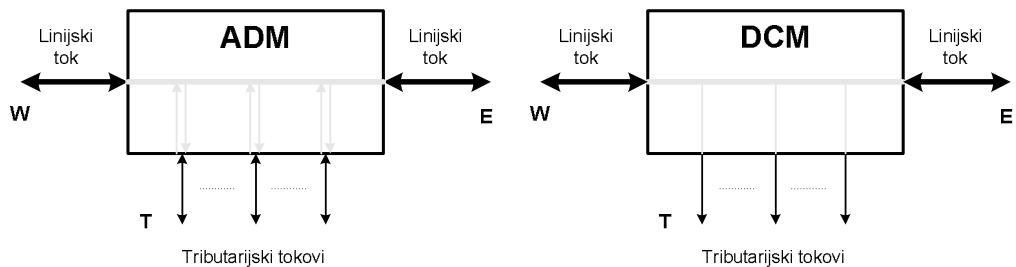
Kao što je već navedeno u uvodnom poglavlju, prstenaste mrežne strukture sa mogućnošću zaštite u slučaju kvarova su najveću primjenu doživjele u SDH i SONET prijenosnim sustavima. Zahtjevi za visokom pouzdanošću sustava i mogućnostima automatskog preusmjeravanja u slučaju kvarova mogli su se jednostavno realizirati korištenjem prstenastih struktura, pa prstenovi postaju prepoznatljiv model u izgradnji prijenosnih sustava baziranih na SDH tehnologijama. Ta integracija rješenja baziranih na prstenovima bila je moguća zahvaljujući uskladivosti predviđene SDH opreme – multipleksera i digitalnih prospojnika sa modelima prstenastih mreža. Osim u SDH/SONET mrežama prstenovi su, kao koncept, prihvaćeni i u WDM (*Wavelength Division Multiplex*) mrežama, te nekim specifičnim prijenosnim sustavima poput DPT-a (*Dynamic Packet Transport*). Razlike u implementaciji prstenova u različitim sustavima postoje, no konceptualno se radi o identičnim rješenjima, s tim da pojedine tehnologije uvode novosti kojima se 'preskaču' ograničenja drugih tehnologija (npr. DPT podržava više od 16 čvorova u jednom prstenu, dok je u SDH broj čvorova ograničen na 16).

2.5.1. Sklopovlje

Prstenovi se izvode korištenjem optičkih vlakana koja povezuju susjedne čvorove. Čvorovi se izvode kao multiplekseri ili prospojnici. Izvedbe multipleksera i prospojnika se razlikuju ovisno o korištenoj tehnologiji, no funkcionalnost unutar prstena je identična.

Multiplekseri – ADM, DCM

Multiplekseri su uređaji koji imaju dvije osnovne funkcije. Prva je dodavanje informacijskih tokova nižih brzina u tok veće brzine, a druga izdvajanje tokova nižih brzina iz toka veće brzine. Ovisno o tome koju funkciju obavljaju, razlikujemo multipleksere s dodavanjem i izuzimanjem tokova (ADM – *Add/Drop Multiplexer*) i multipleksere s izuzimanjem (DCM – *Drop & Continue Multiplexer*). U praksi se svaki ADM može konfigurirati da radi kao DCM. Tokovi nižih brzina koji se dodaju ili izuzimaju iz toka veće brzine nazivaju se pritocima (tributarijima), dok se tok veće brzine može nazivati linijskim tokom. Shematski prikazi ADM-a i DCM-a dani su na slici 2.13.



Slika 2.13: Shematski prikaz ADM i DCM

ADM se shematski i funkcionalno opisuje svojim istočnim E (*east*) i zapadnim W (*west*) ulazno-izlaznim sučeljima kroz koje prolazi informacijski tok velike brzine u kojeg se dodaju i iz kojeg se izuzimaju tokovi nižih brzina. Razdvajanjem na E i W definiraju se istočna linija (*east line*) i zapadna linija (*west line*). Tokovi nižih brzina (pritoci) ulaze i izlaze u i iz multipleksera kroz tributarijske ulaze i izlaze T.

Za razliku od ADM-a, DCM izdvaja dio toka sa linije, no ne mijenja sadržaj glavnog toka, tj. ne dodaje nove pritoke na mjesto izdvojenih, već je sadržaj toka nepromijenjen. Multiplekseri s izuzimanjem se koriste na specifičnim mjestima u mreži gdje se ne dodaju novi pritoci na linijski tok. Jedna od primjena je pri povezivanju susjednih prstenova korištenjem više čvorova.

U početnoj fazi razvoja SDH tehnologije, postojala su neka funkcionalna ograničenja koja su kasnije reducirana. Inicijalno je bilo osmišljeno da pojedini čvor može izdvajati i dodavati pritoke u dio toka koji mu je unaprijed eksplicitno dodijeljen (TSA – *Time Slot Assignment*). Kasnije je omogućeno dodavanje toka u slobodne dijelove toka koje su npr. oslobođili neki drugi čvorovi (TSI- *Time Slot Interchange*). Time je ukinuto ograničenje i omogućeno bolje iskorištavanje prometnog kapaciteta prstenova.

Prospojnici - DXC

Prospojnici (DXC – *Digital Crosconnect*) se koriste za prospajanje tokova različitih brzina. Posjeduju više ulaza i izlaza, na koje preusmjeravaju odnosno prospajaju tokove. Korištenje prospojnika u prstenastim mrežama je rijede nego li korištenje multipleksera. Postoje dva načina upotrebe prospojnika u prstenastim mrežama: korištenje prospojnika koji funkcioniра kao multiplekser, te korištenje prospojnika za povezivanje različitih prstenova i prospajanje tokova između njih. Korištenje prospojnika kao multipleksera najčešće se izbjegava zbog nešto više cijene samog uređaja, dok je korištenje za povezivanje prstenova vrlo često.

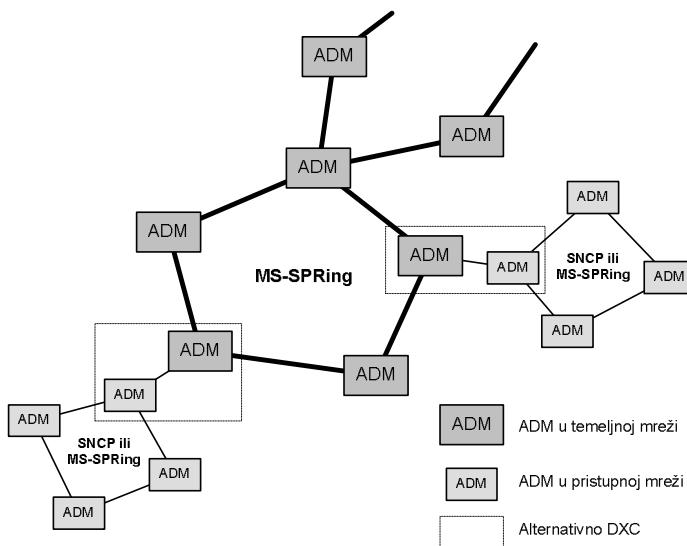
Ovaj rad analizira mreže prstenova i njihovu primjenu, te se ne orijentira na nijednu tehnologiju. Međutim, najčešća primjena prstenastih struktura u praksi je prisutna u SDH/SONET tehnologiji pa će i primjeri biti prikazani na SDH/SONET modelu. Funkcionalno identična situacija je u mrežama zasnovanim na WDM/DWDM tehnologiji. Postupak SDH multipleksiranja u potpunosti je standardiziran [12] i omogućava uključivanje različitih pritoka u složenije tokove.

Tablica 2.5: Standardizirane brzine u SDH/SONET sustavima

SDH okvir	SONET okvir	Brzina prijenosa
STM-0	STS-1, OC-1	51,84 Mbit/s
STM-1	STS-3, OC-3	155,52 Mbit/s
STM-4	STS-12, OC-12	622,08 Mbit/s
STM-16	STS-48, OC-48	2488,32 Mbit/s
STM-64	STS-192, OC-192	9953,28 Mbit/s
STM-256	STS-768, OC-768	39812,12 Mbit/s

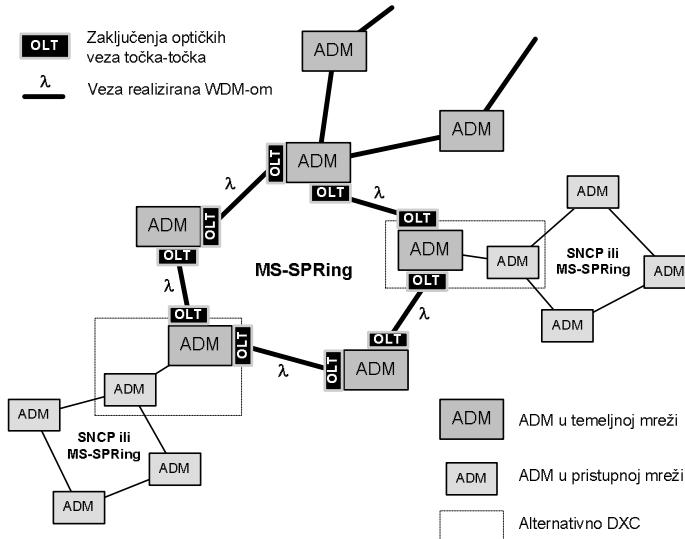
Okviri viših brzina STM-N formiraju se povezivanjem okvira manjih brzina STM-M ($M < N$).

U SDH sustavima, tok veće brzine može biti npr. 622Mbit/s, dok ulazni/izlazni tokovi mogu biti 2Mbit/s ili 155Mbit/s informacijski tokovi. Gornja granica brzina pomaknuta je čak do 40Gbit/s (tablica 2.5) što omogućava izgradnju temeljnih telekomunikacijskih mreža velikih brzina, a bez potrebe za uvodenjem WDM/DWDM tehnologija u postojeće mreže. Primjer jednostavne SDH mreže bazirane na prstenovima prikazuje slika 2.14. Logički se mreža može podijeliti na dva dijela: pristupni i temeljni dio. Oba dijela mreže mogu se izgraditi korištenjem prstenova. Temeljni dio ponekad se u praksi izvodi i kao višestruko povezana mreža u kojoj se koriste DXC i ADM čvorovi. Takva izvedba također osigurava kvalitetne mehanizme zaštite u slučaju kvara, no nauštrb sporije realizacije zaštite u odnosu na prstenaste mreže.



Slika 2.14: Jednostavna SDH mreža bazirana na prstenovima

Daljnje povećanje brzina u postojećim SDH mrežama bazirati će se na uvođenju WDM/DWDM veza između čvorova (od točke do točke – *point to point*), te proslijedivanju tokova velikih brzina prenesenih WDM sustavima na postojeću ili unaprijedenu SDH opremu. Primjer uvođenja WDM tehnologije u temeljni dio mreže prikazan je na slici 2.15.



Slika 2.15: SDH mreža uz uveden WDM u temeljni dio

Na primjeru uvođenja WDM-a sa slike ukazuje se na mogućnosti prijenosa prometa korištenjem WDM veza od točke do točke. Pritom je bitno naglasiti da po uvođenju takvih veza ostaje dovoljno prijenosnih kapaciteta za uporabu u ostalim vrstama mreža.

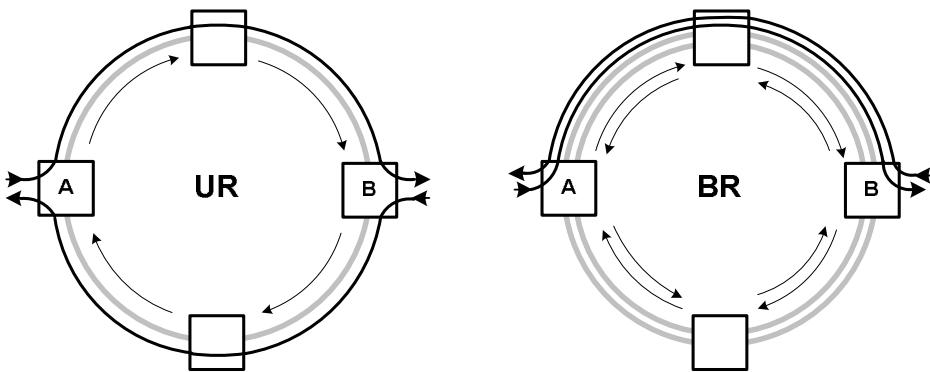
S druge strane, izgradnja novih mreža vjerojatno će se svesti na korištenje WDM/DWDM bazirane temeljne mreže i eventualnu implementaciju zaštitnih optičkih prstenova. Funkcioniranje zaštitnih prstenova u WDM/DWDM mrežama bilo bi moguće korištenjem optičkih ADM multipleksera (OADM) i optičkih prospojnika (OXC). U prijelaznom razdoblju koriste se i optičko-digitalni prospojnici (ODXC), no u budućoj sveoptičkoj mreži neće biti korišteni.

2.5.2. Izvedba jednostavnog prstena

Kao što je prethodno navedeno, prsten je izgrađen od čvorova koji mogu biti multiplekseri ili prospojnici. Čvorovi su u pravilu povezani optičkim vlaknima. Svaki prometni zahtjev može se usmjeriti jednim od ponuđenih smjerova – u smjeru kazaljke na satu i u smjeru suprotnom smjeru kazaljke na satu. Prometni zahtjevi mogu biti simetrični ili nesimetrični. O simetričnim prometnim zahtjevima govorimo u slučaju da su zahtjevi od čvora A do čvora B jednakim zahtjevima od čvora B do čvora A. Budući da je to najčešći slučaj, ovaj rad u matematičkom modelu i izvedbi algoritama i heuristika predviđa simetričnost zahtjeva. Ukoliko pak govorimo o nesimetričnim zahtjevima zahtjevi u svakom od smjerova nisu jednaki, tj. prometni zahtjevi od čvora A prema čvoru B se razlikuju od prometnih zahtjeva čvora B prema čvoru A.

Ovisno o izvedbi prstena razlikujemo jednosmjerni (*unidirectional ring* – UR) i dvosmjerni (*bidirectional ring* – BR). Kod jednosmjernog prstena svi prometni zahtjevi se usmjeravaju u istom smjeru, dok kod dvosmjernog prstena zahtjevi mogu biti usmjeravani u različitim smjerovima. U slučaju dvosmjernog prstena, različitim usmjeravanjima možemo utjecati na smanjenje kapaciteta, a samim time i troškove izgradnje prstena.

Kad se radi o jednosmjernom prstenu susjedni čvorovi se međusobno povežu jednim optičkim vlaknom. Kroz vlakno se svi prometni zahtjevi usmjeravaju u istom smjeru. Kod dvosmjernog prstena uobičajeno je korištenje dva optička vlakna za povezivanje susjednih čvorova. Pri tome se jedno vlakno koristi za prijenos prometnih zahtjeva u jednom, a drugo u drugom smjeru. Drugim riječima, zahtjevi između para čvorova u dvosmjernom prstenu se usmjeravaju istim putem, različitim smjerovima. Za razliku od usmjeravanja kod jednosmjernog prstena gdje se zahtjevi prenose istim smjerom, ali neovisnim putevima. UR i BR su shematski prikazani na slici 2.16.

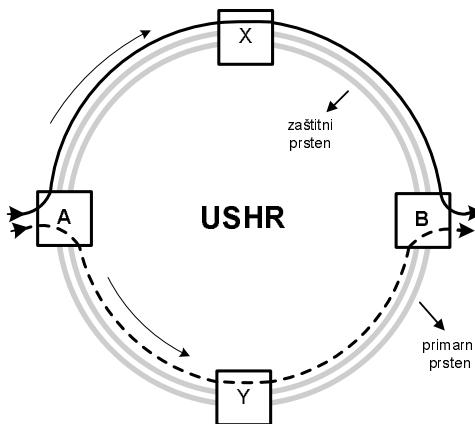


Slika 2.16: Shematski prikaz UR i BR

2.5.3. Prstenovi sa zaštitom – samooporavljivi prstenovi

Iako opisani jednosmjerni i dvosmjerni prstenovi mogu korektno funkcionirati, opisani modeli nikad se (ili vrlo rijetko) ne koriste u praksi. Korištenje prstenastih struktura dobiva smisao tek kad se u obzir uzme zaštita prometa i problematika funkcioniranja mrežnih struktura u slučaju kvara u mreži. Naime, prsten kao dvostruko povezana struktura predstavlja idealno rješenje za realizaciju automatske zaštite u slučaju kvara. Automatska zaštita se svodi na zaštitno preusmjeravanje (APS – *Automated Protection Switching*) prometa u trenutku kad se detektira kvar. Upravo je brza detekcija kvara i jednostavnost aktivacije zaštitnih mehanizama zaslužna za tako dobru prihvaćenost SDH prstenastih struktura pri realizaciji telekomunikacijskih mreža. Naime, ITU-T definira zahtjeve za detekciju kvara i uspostavu alternativnih puteva i ispravno funkcioniranje mreže tako da je dozvoljeno vrijeme prekida usluga maksimalno 50ms. U literaturi se ta vrijednost ponekad smatra premalom, no svi proizvođači opreme deklariraju upravo tih 50ms kao maksimalno vrijeme potrebno za aktivaciju zaštitnih mehanizama i nesmetan rad mreže.

Zaštita u slučaju kvarova u mreži nužna je za osiguranje zahtijevanih parametara za kvalitetom usluge u temeljnim telekomunikacijskim mrežama. Bez izvedbe brze i efikasne zaštite, u slučaju kvara dogodili bi se ogromni, za korisnike u potpunosti neprihvatljivi, informacijski gubici.

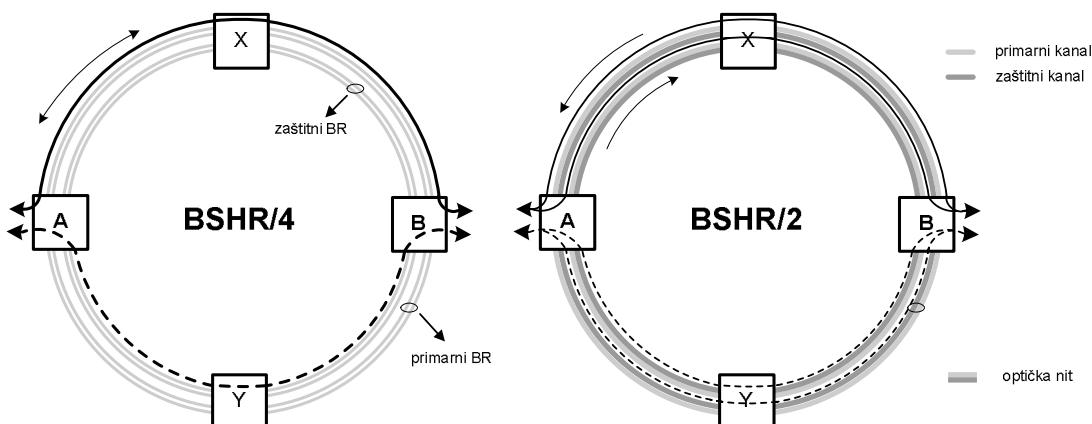


Slika 2.17: USHR model prstena

Rješenje problema zaštite bazirano je na samooporavljivim prstenovima (SHR – *self-healing ring*). Spomenuti jednosmjerni (UR) i dvosmjerni (BR) prstenovi se modifciraju dodavanjem zaštitnog prstena koji se koristi u slučaju kvara kako bi se prometni zahtjevi na koje utječe kvar usmjerili alternativnim (u slučaju prstena jedinim mogućim, sekundarnim) putevima. Na taj način se jednostavno 'zaobilazi' dio mreže u kojem se dogodio kvar. Bitno je napomenuti da mehanizmi zaštite funkcioniraju samo u slučaju jednostrukog kvara. Općenito, vjerojatnost pojave višestrukog kvara na određenom dijelu mreže je mala i najčešće se zanemaruje u razmatranjima. Ipak, kao

takva, uviјek postoji. Stoga se u postupku planiranja mreže posebno izvode i proračuni pouzdanosti mrežnih elemenata, kako bi se procijenila pouzdanost kompletne mreže.

Dodavanjem zaštitnog prstena, koji se u praksi izvodi dodavanjem još jednog optičkog vlakna (kod jednosmjernih prstenova) odnosno dva optička vlakna (kod dvosmjernih prstenova) za povezivanje susjednih čvorova u prstenu, izgraduju se jednosmjerni samooporavlјivi (USHR – *Unidirectional Self Healing Ring*) i dvosmjerni samooporavlјivi (BSHR – *Bidirectional Self Healing Ring*) prstenovi. Prikaz USHR-a je dan na slici 2.17, dok je BSHR prikazan na slici 2.18. Primjeri samooporavlјivih prstenova na slikama jasno omogućavaju razlučivanje postojanja primarnog (radnog (*working*)) i sekundarnog (zaštitnog (*protection*)) prstena.



Slika 2.18: BSHR modeli – BSHR s 4 vlakna i BSHR s dva vlakna

Jednosmjerni samooporavlјivi prsten izvodi se korištenjem dva optička vlakna, dok je dvosmjerni samooporavlјivi prsten najčešće izведен sa četiri optička vlakna. Osim modela koji koristi četiri vlakna, ponekad se koristi i izvedba dvosmjernog samooporavlјivog prstena korištenjem dva optička vlakna. U tom slučaju nije moguće fizički razlučiti radni i zaštitni prsten jer istim optičkim vlaknom prolaze i osnovni i zaštićeni prometni zahtjevi. Ovisno o broju vlakana koja se koriste uvode se oznake BSHR/2 za dvosmjerni prsten s dva vlakna, odnosno BSHR/4 za prsten sa četiri optička vlakna. Jednosmjerni prsten se uviјek izvodi s dva optička vlakna, a označava se s USHR/2 ili samo USHR.

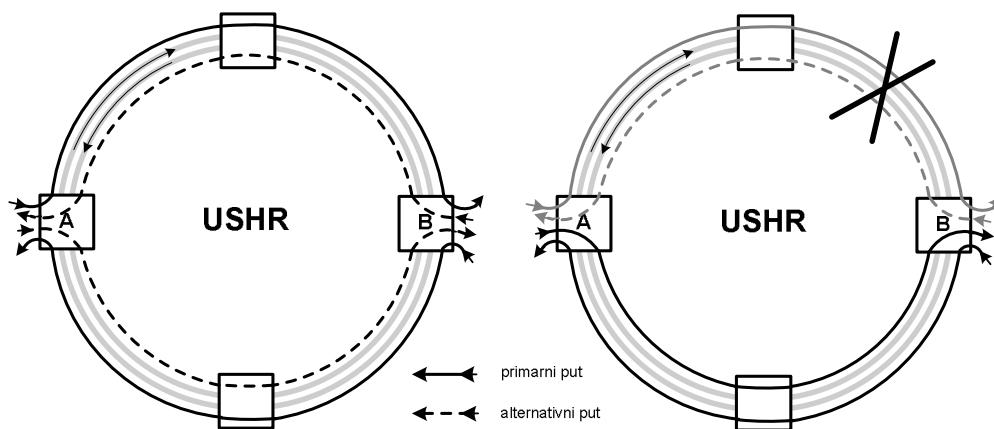
Bitno je naglasiti da postoje različiti načini izvedbe USHR i BSHR, ovisno o principu zaštite koja se koristi. Isto tako, ovisno o tehnologiji, koristi se različito nazivlje za gotovo identične izvedbe samooporavlјivih prstenova. U tablici 2.6 su pobrojani najčešće korišteni modeli samooporavlјivih prstenova, te njihovo nazivlje u SDH i SONET arhitekturama.

Tablica 2.6: Nazivlje u SDH i SONET arhitekturama

Opis	SDH naziv	SONET naziv
Jednosmjeran samooporavlјiv prsten (USHR) sa zaštitom na razini puta	SNCP (Subnetwork connection protected ring)	UPSR (Unidirectional Path Protected Self-healing Ring)
Dvosmjeran samooporavlјiv prsten s 2 vlakna (BSHR/2) sa zaštitom na razini grane	MS-SPRING 2f (Multiplex Section Shared Protection Ring with 2 fibers)	BLSR 2f (Bidirectional Link Protected Self-healing Ring with 2 fibers)
Dvosmjeran samooporavlјiv prsten s 4 vlakna (BSHR/4) sa zaštitom na razini grane	MS-SPRING 4f (Multiplex Section Shared Protection Ring with 4 fibers)	BLSR 4f (Bidirectional Link Protected Self-healing Ring with 4 fibers)

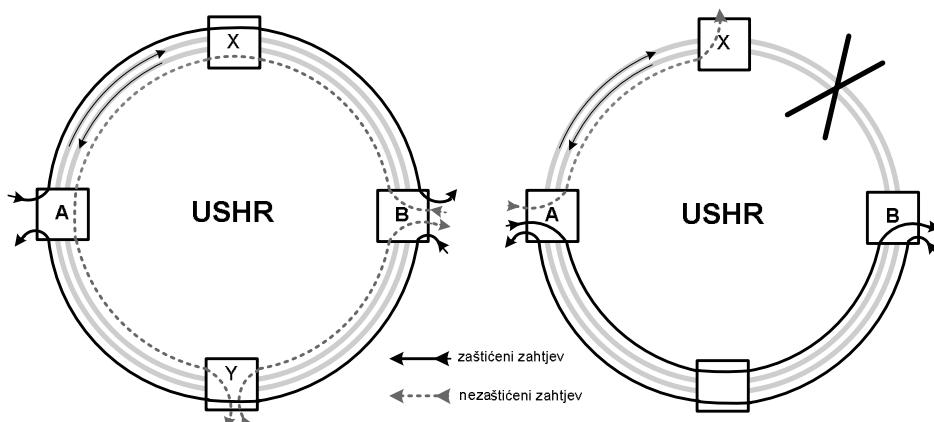
Zaštita na razini puta

Ukoliko se izvodi zaštita na razini puta govorimo o UPSR (*Unidirectional Path Protected SHR*) i BPSR (*Bidirectional Path Protected SHR*). UPSR se u SDH naziva još i SNCP (*Subnetwork Connection Protected Ring*). Drugi naziv za prstenove sa zaštitom na razini puta je DPR (*Dedicated Protection Ring*), zbog svojstva da nema dijeljenja kapaciteta radnog i zaštitnog prstena. Zaštita na razini puta svodi se na uspostavljanje alternativnog puta u slučaju kvara na mreži. Alternativni put se uspostavlja korištenjem zaštitnog prstena i to neovisnim dionicama. Zaštita može biti izvedena kao 1+1 i kao 1:1. Kod 1+1 zaštite, prometni zahtjevi se istodobno prenose i glavnim i zaštitnim prstenom. Odredišni čvorovi nadgledaju snagu i kvalitetu signala na ulazima, te se odlučuju za kvalitetniji signal. Shematski prikaz usmjeravanja u jednosmjernom samooporavljivom prstenu je dan na slici 2.19. Prema SDH standardima u SNCP prstenovima koristi se 1+1 zaštita.



Slika 2.19: Shematski prikaz – USHR sa zaštitom na razini puta

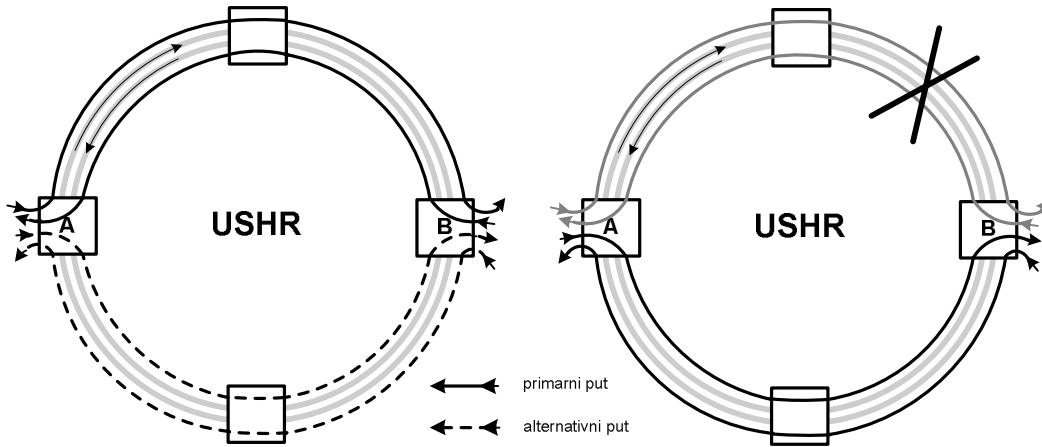
Nedostatak ovog pristupa je taj da je zaštitni prsten opterećen i u slučaju kad nema kvara u mreži. Korištenje 1:1 zaštite omogućava korištenje zaštitnog prstena za prijenos prometa nižeg prioriteta – u slučaju kvara u mreži prekida se promet nižeg prioriteta, te se zaštitni prsten koristi za uspostavu alternativnog puta i prijenos prometnih zahtjeva na koje je kvar utjecao (UPSR/SNCP uz korištenje 1:1 zaštite prikazan je na slici 2.20. – lijevo je prikazan prsten bez kvara kojim prometuju i nezaštićeni zahtjevi nižeg prioriteta između čvorova Y i B, a desno prsten nakon preusmjeravanja uslijed kvara grane – prikazana je i mogućnost uspostave niskoprioritetne jednosmjerne komunikacije između čvorova A i X).



Slika 2.20: USHR sa zaštitom na razini puta – 1:1 i nezaštićeni zahtjevi

Postoji i alternativni model korištenja jednosmjernog zaštitnog prstena – komunikacija u oba smjera se može obavljati istim putem, odnosno istim multipleksnim dionicama. U slučaju kvara

koji utječe na prometovanje zahtjeva, odabire se alternativni put kojim se provodi također dvosmjerna komunikacija. Izbor modela korištenja ovisi o samoj fizičkoj izvedbi mreže. Prikaz alternativnog modela dan je na slici 2.21.



Slika 2.21: Alternativni model zaštite na razini puta u USHR

SDH/SONET podržava zaštitu na razini puta korištenjem standardiziranih upravljačkih protokola. Ipak, kod zaštite na razini puta ne radi se o korištenju mehanizma automatskog zaštitnog prospajanja (APS). Signalizacija nužna za detekciju kvara i uspostavu alternativnog puta prenosi se u zaglavlju puta (POH – Path Overhead) virtualnog spremnika u SDH okviru.

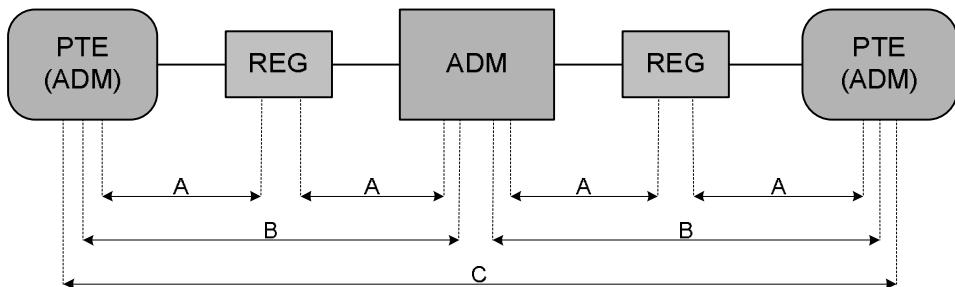
Prema SDH standardu inicijalno je zaštita na razini puta definirana za jednosmjerne prstenove UPSR, no standardizirana je i zaštita za dvosmjerne BPSR prstenove. Zaštita na razini puta u praksi se izvodi gotovo uvijek za jednosmjerne samooporavljive prstenove. Za dvosmjerne pak se prstenove u pravilu koriste APS mehanizmi zaštite na razini multipleksne dionice. Zaštita na razini puta koristi se i u *mesh* SDH mrežama. Moguća je izvedba na svim razinama virtualnih spremnika (dakle, teoretski do 2Mbit/s razine). No, jasno je da će takva vrsta zaštite zahtijevati prilične signalizacijske kapacitete i dodatno procesiranje u slučaju kvarova, pa se izbjegava korištenje na razini virtualnih spremnika nižih brzina. Bez obzira na područje primjene, zaštita na razini puta ne može se, zbog sporosti postupka, uspoređivati sa mnogo bržim APS metodama koje se koriste u samooporavljivim prstenovima sa zaštitom na razini dionice.

Zaštita na razini multipleksne dionice/linije

Zaštita na razini grane opisana u poglavlju 2.4. u SDH arhitekturi se naziva zaštitom na razini multipleksne dionice, dok se u SONET arhitekturi govori o zaštiti na razini linije. Multipleksna dionica (*multiplex section*) se nalazi između susjednih multipleksera, što je identično liniji (*line*) u SONET arhitekturi. Pri izvođenju zaštite na razini multipleksne dionice/linije, alternativni putevi se uspostavljaju tako da se u graničnim čvorovima (najблиžim kvaru) izvrši prospajanje koje omogućava zaobilazeњe dionice zahvaćene kvarom. Općenita definicija dionica u transportnim mrežama prema SDH/SONET standardima prikazana je na slici 2.22., a nazivlje korišteno u standardima navedeno je u tablici 2.7.

Tablica 2.7: Nazivlje koje se koristi za dionice prema SDH i SONET standardima

Oznaka	SDH naziv	SONET naziv
A	Regeneratorska dionica (Regenerator Section)	Dionica (Section)
B	Multipleksna dionica (Multiplex Section)	Linija (Line)
C	Put (Path)	Put (Path)

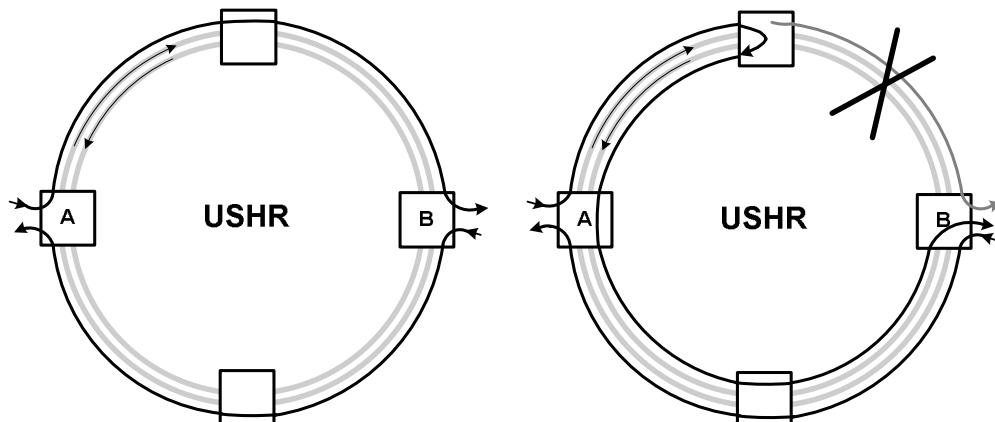


Slika 2.22: Shematski prikaz dionica u transportnim mrežama

Oznake na slici označavaju osnovne mrežne elemente: PTE (*Point of Termination*) predstavlja rubnu točku, ADM multipleksler, a REG regenerator.

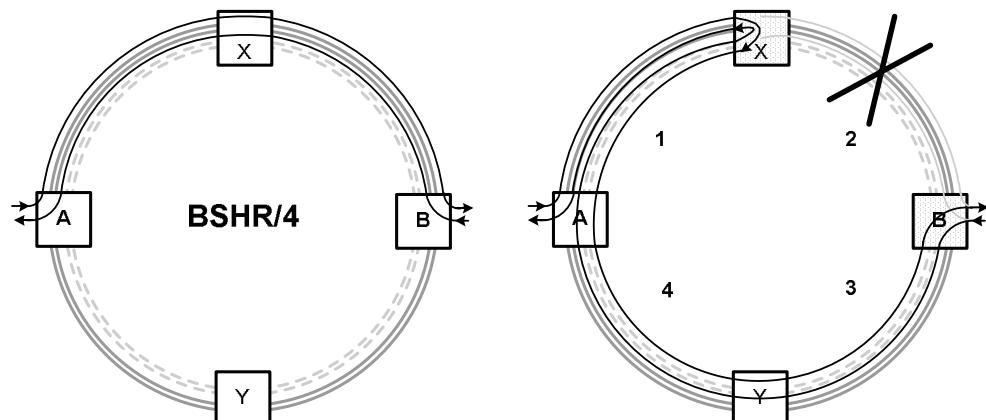
Prstenove sa zaštitom na razini multipleksne dionice, odnosno linije označavamo sa ULSR, (*Unidirectional Line Protection SHR*) ukoliko se radi o jednosmjernim samooporavljivim prstenovima, te MS-SPRing (*Multiplex Section-Shared Protection Ring*) odnosno BLSR (*Bidirectional Line Protection SHR*), ukoliko se radi o dvosmjernim samooporavljivim prstenovima.

Primjer zaštite na razini multipleksne dionice prikazan je za USHR na slici 2.23.



Slika 2.23: USHR sa zaštitom na razini dionice uz zatvaranje petlje

Slika 2.24. prikazuje zaštitu na razini multipleksne dionice za BSHR izveden korištenjem četiri optička vlakna.



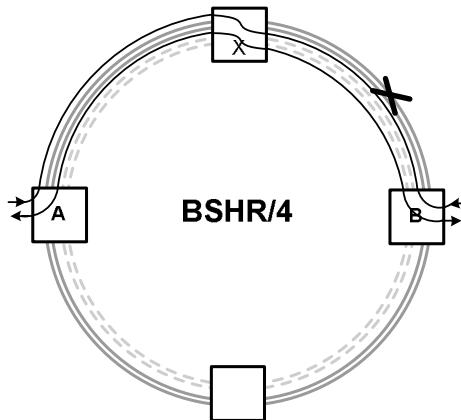
Slika 2.24: BSHR/4 sa zaštitom na razini dionice uz zatvaranje petlje

Vidljivo je da se u oba slučaja na rubnim čvorovima oko kvara obavlja zatvaranje petlje (tzv. *loopback* svojstvo ADM čvorova) i prometni zahtjevi se preusmjeravaju na zaštitne prstenove.

Kod jednosmjernog samooporavljivog prstena vidljivo je da u slučaju kvara zahtjev od čvora B do čvora A nastavlja prometovati kao i u slučaju bez kvara. Zahtjev u drugom smjeru se na graničnom čvoru preusmjeri na zaštitni jednosmjerni prsten kroz kojeg se može provesti do odredišnog čvora isključivo zbog svojstva zaštitnog prstena da mu je smjer suprotan smjeru definiranom u radnom prstenu. Kad bi oba prstena imala isti smjer zaštite bi bila moguća jedino u slučaju da se na ugroženoj grani kvar dogodi samo na radnom prstenu, a zaštitni ostane ispravan.

U dvosmjernom prstenu aktivacija *loopback* mehanizma preusmjerava dvosmjerni zahtjev – budući da se komunikacija u oba smjera odvija istim putem, kvar onemoguće usmjeravanja istim putem za bilo koji smjer, pa se cjelokupni (dvosmjerni) zahtjev usmjerava na zaštitni prsten. Drugi način zaštite u BSHR/4 na razini dionice, također prisutan u praksi, svodi se na preklapanje linijskog sučelja (tzv. *line switch*). Taj model prikazan je na slici 2.25.

Preklapanje linijskog sučelja moguće je samo u slučaju kad kvar na grani nema utjecaja na zaštitni prsten (npr. kvar na optičkim vlaknima u radnom prstenu, pri čemu na istoj dionici/grani zaštitni prsten nije oštećen i može funkcionirati). Pri tome se u graničnim čvorovima (na slici označeni kao X i B) promet jednostavno preusmjeri s dionice radnog prstena na dionicu zaštitnog prstena.



Slika 2.25: BSHR/4 sa zaštitom na razini dionice uz linijsko preklapanje

Ograničenje koje uvode SDH standardi odnosi se na broj kanala po prstenu. To je u stvari broj multipleksnih dionica koje se mogu nalaziti u pojedinom prstenu. Kako se u K1 i K2 oktetu po četiri bita koriste za definiranje kanala, proizlazi maksimalan broj dionica 16. Ukoliko svaki ADM čvor u prstenu obavlja svoje funkcije dodavanja i izdvajanja pritoka iz linijskog toka, znači da prsten može sadržavati ukupno 16 čvorova. Detalji i specifikacije postupaka zaštite mogu se pronaći u ITU-T specifikacijama [9], a pregled je dan i u [10,11,12].

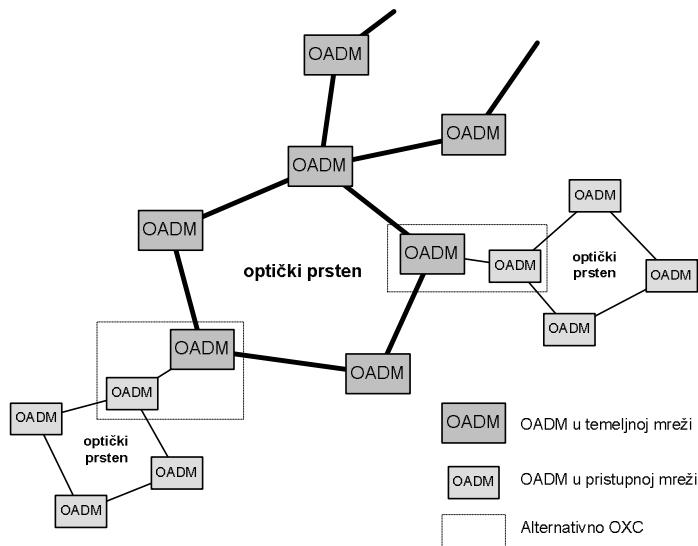
U praksi se javljaju sve navedene izvedbe prstena, no postoje neke pravilnosti u korištenju. Jednosmjerni prsteni sa zaštitom na razini puta (SNCP/UPSR) u pravilu se koriste u pristupnim dijelovima temeljnih mreža za sakupljanje prometa sa bliskih lokacija. U središnjem dijelu temeljnih mreža obično se koriste dvosmjerni prstenovi sa zaštitom na razini dionica (MS-SPRing/BLSR).

2.5.4. Princip korištenja prstenova u WDM/DWDM okruženju

U novije vrijeme na tržištu su se pojavili uređaji koji podržavaju najveće okvire definirane SDH/SONET standardima. SDH razine STM-64 i STM-256, kapaciteta 10Gbit/s odnosno 40 Gbit/s podržane svim novijim modelima SDH opreme omogućavaju povećanje brzina sustava do samih granica, nezamislivih pri osmišljavanju prvotnog SDH koncepta. Daljnje povećanje kapaciteta vjerojatno neće biti isplativo direktnim korištenjem SDH tehnologije već će se morati započeti s unaprednjem postojećih ili izgradnjom novih mreža velikih brzina. Naime, korištenje

valnog multipleksa omogućava višestruko povećanje prijenosnog kapaciteta na postojećim optičkim vlaknima, te se njegovim uvođenjem nastoji što bolje iskoristiti postavljenu infrastrukturu. Jedno od rješenja koje se polako uvodi u postojeće SDH mreže je korištenje WDM tehnologije za uspostavu veza između SDH čvorova. Na taj način bi se postojeća optička vlakna iskoristila za prijenos više valnih duljina te bi se time umnogostručila prijenosna svojstva vlakana. U tom slučaju postojeća mreža bi funkcionalala bez ikakvih preinaka, a preostali kapaciteti mogli bi se iskoristiti u drugim tipovima mreža ili za izgradnju novih mreža, pri čemu bi SDH bio aktivan dok za to postoji potreba. Scenarij djelomičnog uvođenja WDM tehnologije u postojeću SDH mrežu prikazan je u odjeljku 2.5.1 na slici 2.15. Taj scenarij predviđa uvođenje WDM veza točka-točka i iskorištavanje kapaciteta za korištenje u SDH/SONET mreži.

Mogući scenarij je i intenzivno korištenje WDM-a u temeljnoj mreži te izgradnja sveoptičke mreže koja bi među ostalim mogla podržavati i prstenaste samooporavljive strukture korištenjem optičkih multipleksera, OADM (slika 2.26). Konceptualno se sveoptička mreža izgradena od optičkih prstenova ne razlikuje od standardne SDH mreže – ADM čvorovi zamijenjeni su optičkim ADM čvorovima (OADM), dok se za povezivanje podmreža i prstenova koriste optički prospojnici (OXC).



Slika 2.26: Koncept sveoptičke mreže bazirane na prstenovima

Međutim, postoje značajne razlike u funkcioniranju optičkih prstenova – za razliku od SDH prstenova u kojima se različiti prometni zahtjevi dodaju na linjsko sučelje, u optičkim prstenovima zahtjevi se prenose različitim valnim duljinama. Kako cijena izgradnje ovisi o broju valnih duljina koje se koriste, u optičkim se prstenovima problem optimizacije svodi na pronalaženje optimalnog korištenja valnih duljina za ponuđene prometne zahtjeve.

Pitanje isplativosti nadogradnje SDH mreža tema je mnogobrojnih rasprava [13,14]. Neki rezultati sugeriraju nadogradnju kako bi se dostigla isplativost relativno nove opreme postavljene u posljednjih 10-tak godina, dok drugi predlažu kompletan redizajn mreže, uz korištenje postojećih optičkih vlakana, te izgradnju sveoptičke mreže. Naravno, osnovni preduvjet produženja korištenja SDH mreža je dopuna standarda mogućnostima prijenosa različitih signala nevezano uz zasad standardizirane diskretne brzine (*traffic grooming*). Na tim i sličnim dopunama standarda se intenzivno radi, no pitanje je kako će tržište prihvati novosti. Velik utjecaj na odluke imat će i spremnost proizvođača opreme da ulažu u nadogradnju postojeće opreme ako je isplativost upitna.

2.6. Rezultati postupka dimenzioniranja jednostavnog prstena

U okviru praktičnog dijela rada implementirane su sve spomenute metode dimenzioniranja prstenova. Dobiveni rezultati uspoređeni su sa rezultatima dobivenim korištenjem CPLEX-a za

rješavanje prezentiranog matematičkog programa. U nastavku je dan tablični pregled rezultata koji su grupirani prema klasama prstenova. Prstenovi se razlikuju prema broju čvorova/grana i broju definiranih zahtjeva i podijeljeni su u tri klase – male, srednje i velike prstenove. Kriterij za klasifikaciju je broj zahtjeva s obzirom na broj čvorova u prstenu. Naime, uz pretpostavku da su zahtjevi dvosmjerni i da nije moguće definirati više zahtjeva između istog para čvorova, maksimalan broj zahtjeva u prstenu je jednak $\frac{n \cdot (n - 1)}{2}$ – svaki čvor može uputiti zahtjev prema svim ostalima, s tim da se eliminiraju zahtjevi između istih para čvorova (zanemaruju se smjerovi zbog pretpostavke o simetričnosti zahtjeva u oba smjera). Klasifikacija je izvedena na slijedeći način:

1. *mali prstenovi* – broj zahtjeva jednak je broju čvorova u prstenu
2. *srednji prstenovi* – broj zahtjeva jednak je polovici mogućeg broja zahtjeva
3. *veliki prstenovi* – broj zahtjeva je maksimalan za definiran broj čvorova

Rezultati su dobiveni na slučajno generiranim modelima prstenova – ulazni podaci su broj čvorova u prstenu te broj zahtjeva i karakteristike zahtjeva (ograničenja u veličini zahtjeva). Zahtjevi se definiraju slučajnim izborom (uniformna distribucija) uz vrijednosti od 1 do 30. Za svaku veličinu prstena postupak se ponavlja deset puta, te rezultati predstavljaju srednje vrijednosti. Rezultati su prikazani u tablici 2.8. Detalji postupka određivanja rezultata dani su u [8].

Vremena potrebna za izvođenje svih iznesenih proračuna zadovoljavaju u slučaju korištenja na stvarnim mrežama. Za mreže sa većim brojem čvorova (u realnim mrežama ne pojavljuju se prstenovi sa više od 16 čvorova) i maksimalnim brojem zahtjeva vremena za 10 izvođenja nisu prelazila 1 sekundu na standardnom PC računalu. Korištenje CPLEX-a za rješavanje matematičkog programa se pokazalo nešto sporijim, no još uvijek dovoljno brzo za realne primjene. Ipak, složenost standardnih metoda koje CPLEX koristi za rješavanje matematičkih programa može u određenim slučajevima težiti eksponencijalnoj, pa se korištenje u praksi izbjegava. Rezultati dobiveni CPLEX-om su optimalni i služe za vrednovanje rezultata implementiranih metoda (u RDPw/oS).

Tablica 2.8: Rezultati dobiveni različitim metodama dimenzioniranja

N,K	UPSR/ SNCP	BLSR/MS-SPRing RDPw/oS				RDPwS	
		CPLEX	Greedy	DAscent	2-OPT	EXACT	INDES
Mali prstenovi							
5,5	57,1	41,8	41,8	45,2	42,9	36,4	36,6
10,10	107,8	66,5	67,4	78,4	70,6	61,5	61,9
15,15	143,6	90,2	92,6	103,9	93,3	88,3	88,6
20,20	181,6	114,0	118,4	139,0	119,1	111,7	112,1
40,40	329,0	192,6	207,0	240,7	198,3	189,6	189,7
Srednji prstenovi							
5,8	79,3	50,4	50,8	52,4	51,1	45,4	45,7
10,22	209,6	120,2	128,8	139,0	122,3	116,4	116,8
15,52	468,8	255,6	273,7	283,7	263,8	255,8	256,1
20,95	819,5	442,2	487,3	481,0	449,2	439,9	440,5
40,390	3239,6	**	1821,7	1793,8	1717,9	1706,3	1706,6
Veliki prstenovi							
5,10	96,2	59,3	59,8	61,7	60,7	53,6	54,0
10,45	418,6	218,2	238,6	233,5	222,6	215,1	215,4
15,105	917,4	486,2	504,5	504,5	494,3	484,1	484,6
20,190	1575,6	821,6	878,3	858,9	830,1	820,5	820,8
40,780	6201,4	**	3303,7	3250,5	3189,4	3177,0	3177,5

** zbog ograničenja akademске inačice CPLEX-a nije rješivo

Tablica 2.8. ukazuje na nekoliko činjenica:

1. Kapaciteti potrebnii za prijenos i zaštitu zahtjeva u dvosmjernim prstenovima mogu biti 20-48% niži od zahtijevanih kapaciteta u jednosmjernim prstenovima.
2. Smanjenje kapaciteta potrebnih za prijenos u modelu prstena s dijeljenjem zahtjeva u odnosu na model bez dijeljenja zahtjeva nije značajno u mrežama sa većim brojem zahtjeva – smanjenje nekad nije ni 1%, pa se odluka o korištenju tih modela u praksi donosi na temelju rekapitulacije realnih troškova opreme i konfiguriranja mreže. Kod mreža s manjim brojem zahtjeva smanjenje kapaciteta može biti značajno (čak i iznad 10%).
3. Pohlepne (*greedy*) metode pokazuju dobre rezultate samo u malim mrežama. *DualAscent* metoda kod srednjih i većih mreža daje bolje rezultate, nauštrb složenosti i potrebnog vremena izvođenja u odnosu na pohlepne metode.
4. 2-OPT metoda bazirana na EXACT algoritmu, iako niže složenosti od *DualAscent* metode, u pravilu daje najbolje rezultate za RDPw/oS za srednje i velike mreže.
5. Metode za dimenzioniranje uz dozvoljeno dijeljenje zahtjeva daju optimalna rješenja, što se za INDES metodu cjelobrojnog dijeljenja može zaključiti zbog odstupanja od EXACT metode za manje od jedne prometne jedinice.

Na temelju iznesenih činjenica može se zaključiti da je u praksi najbolje za RDPw/oS problem koristiti pohlepne metode za male mreže i 2-OPT metodu za veće mreže. Za rješavanje RDPwS problema odabiru se ili EXACT ili INDES metoda, ovisno o zadanim ograničenjima u planiranju mreže.

3. Postupci planiranja mreža sastavljenih od više prstenova

U ovom poglavlju razmatraju se problemi koji se javljaju u postupku planiranja mreža sastavljenih od više prstenova. Dan je kratak pregled različitih pristupa planiranju, te su detaljno prikazane različite metode koje se koriste u postupku. Sve opisane metode ujedno su i implementirane u praktičnom dijelu rada. Bitno je naglasiti da je postupak planiranja mreže vrlo složen i gotovo nemoguće optimalno rješiv, te se stoga pribjegava raznim metodama koje kao rezultat daju rješenja koja se mogu realizirati u stvarnim mrežama. Jedan od problema u vrednovanju rezultata planiranja svakako je i taj da se korišteni modeli razlikuju od autora do autora, tako da usporedba rezultata biva bespredmetna. Matematički modeli koji se mogu naći u literaturi često definiraju specifičnosti, a katkad zanemaruju neka svojstva mreže ili pak se baziraju na raznim ograničavajućim pretpostavkama. Kao dio ovog rada razvijen je postupak za kompletno planiranje prstenastih mreža – da bi se rezultati mogli uspoređivati predložen je model prema kojem se mreže vrednuju i odlučuje koje rješenje više zadovoljava postavljene zahtjeve. U razvoju metoda koje se koriste u postupku uočen je niz pravilnosti pri dobivanju rezultata. Prema spomenutom modelu vrednovanja doneseni su sudovi o tome da li je i u kojoj mjeri pojedina izvedba neke od faza u postupku planiranja bolja ili lošija u odnosu na druge, također razvijene, izvedbe.

U prvom dijelu poglavlja dan je odvojen pregled različitih metoda koje se mogu koristiti u postupku planiranja, da bi u drugom dijelu bio sagledan kompletan postupak baziran na tim metodama. Prema dobivenim rezultatima iznose se neki zaključci koji mogu pomoći u dalnjem razvoju kvalitetnijih postupaka, kojih bi cilj trebao biti pronalaženje realnijih rezultata. Naime, kroz suradnju na tematski sličnim projektima sa timovima mrežnih dizajnera uočeni su problemi koji mogu onemogućiti realizaciju rezultantnih mreža u praksi, pa je stoga nadogradnju postojećih metoda nužno provesti. S druge strane, pokazalo se da takvi dizajnerski timovi najčešće ne žele ovisiti o automatskim metodama planiranja, već žele imati što veću ulogu u samom procesu – takav pristup je najčešće prisutan u praksi, te olakšava izvedbu alata za planiranje mreža jer se velik dio dizajnerskih problema eliminira predefiniranošću parametara od strane dizajnera. U tom slučaju, dizajnerima preostaje jedan jedini zanimljiv problem – dimenzioniranje pojedinih prstenova u mreži. Taj problem je već obrađen u prethodnom poglavlju.

3.1. Općenito o postupku planiranja prstenastih mreža

Postupci planiranja prstenastih mreža dostupni u literaturi uglavnom primjenjuju dva različita pristupa problemu. Prvi pristup pretpostavlja da su određeni parametri unaprijed definirani, dok drugi pristup na osnovu zadanih ulaznih podataka određuje sve parametre mreže.

Usporedba dva pristupa, te pozitivne i negativne strane svakog od njih prikazuje tablica 3.1.

Pristup "A" može se nazvati i analitičkim pristupom zbog visoke razine predodređenosti rezultata postupka u tom slučaju. Naime, definiranjem određenih ulaznih parametara prije provodenja postupka u stvari se ostvaruje mogućnost analize mreže u ovisnosti o njima. Na taj način dizajneri mreža mogu provjeriti realne modele koje koriste, a ujedno i svaku novu mrežu napraviti što realnijom. Osim topologije mreže moguće je i strukturno definirati mrežu (npr. odrediti ograničen skup prstenova koji se moraju koristiti u dalnjem postupku, ili čak unaprijed odrediti i završno prekrivanje mreže prstenovima), te na taj način preskočiti odredene faze u postupku planiranja. Dva najčešće korištena oblika pristupa "A" su:

- 1) Definiranje topologije mreže i određivanje skupa prstenova koji se mogu koristiti za prekrivanje – u dalnjem postupku potrebno je odrediti usmjeravanje zahtjeva i odabrati prstenove koji prekrivaju mrežu. Postupak je sličan pristupu "B" jer uključuje i problem prekrivanja mreže prstenovima RCP (*Ring Covering Problem*).
- 2) Definiranje topologije mreže i prekrivanja prstenovima – u dalnjem postupku potrebno je usmjeriti zahtjeve kroz prstenove i prema tome odrediti karakteristike prstenova. Ovaj pristup karakterističan je za mnoge alate – radi se o tipičnom analitičkom pristupu.

Tablica 3.1: Usporedba osnovnih pristupa postupku planiranja

	PRISTUP "A"	PRISTUP "B"
Zadano	topologija prstenovi zahtjevi	topologija zahtjevi
Rezultat	kapaciteti prstenova specifikacija elemenata mreže usmjeravanje zahtjeva	struktura mreže, kapaciteti prstenova specifikacija elemenata mreže usmjeravanje zahtjeva
Problemi	RDP/RLP RCP nad definiranim prstenovima	RDP/RLP enumeracija prstenova RCP
Prednosti	realni ulazi – utjecaj dizajnera redukcija problema	šire područje rješenja
Mane	pristup dizajnera?	realizacija u praksi?

Pristup "B" je u stvari postupak sinteze kompletne mreže na temelju osnovnih ulaznih podataka. Na temelju zadane osnovne topologije (čvorovi + grane) i definiranih zahtjeva nužno je u tijeku postupka planiranja riješiti nekoliko problema – odrediti moguće prstenove u mreži (enumeracija prstenova), usmjeriti zahtjeve i odrediti prstenove koji prekrivaju mrežu (RCP). Alati koncipirani na tom pristupu nude mogućnost automatiziranog planiranja kompletne prstenaste mreže.

Pristup "A" češće je korišten u praksi, prije svega zbog ograničenja koja se javljaju u pristupu "B" – iako postoji niz metoda koje se mogu iskoristiti u takvom postupku planiranja, još uvijek su dobiveni rezultati upitnih praktičnih vrijednosti. To je i jasno s obzirom da unos dodatnih ulaznih informacija u postupak kod pristupa "A" daje realniji prikaz mreže, te promjenom niza parametara omogućava veći utjecaj dizajnera na sam postupak. Na taj način dizajneri unose stečeno iskustvo u postupak planiranja, što nijedan alat za automatizirano planiranje ne može učiniti.

3.2. Metode za planiranje prstenastih mreža

Neovisno o pristupu rješavanju problema, alati koji se koriste u planiranju svoju funkcionalnost baziraju na istim postupcima i metodama. U nastavku se opisuju najbitnije metode i funkcije, te se daju primjeri njihovog korištenja.

Model mreže koji se koristi u analizi definiran je u prvom poglavlju rada u odjeljku 1.2.1.

3.2.1. Provjera stupnja povezanosti u mreži

Nakon što je definirana kompletna topologija mreže (skup čvorova V i skup grana E), potrebno je proanalizirati mrežu i utvrditi da li je na takvoj topologiji uopće omogućena izvedba prstenastih struktura. Drugim riječima, potrebno je utvrditi da li je mreža ili samo njen dio barem dvostrukе povezanosti. Dvostruka povezanost znači da je u mreži moguće uspostaviti alternativne puteve, tj. spojiti neka dva čvora korištenjem dva različita puta.

Postupak utvrđivanja stupnja povezanosti je jednostavan. Stupanj povezanosti $cl(v_i)$ pojedinog čvora v_i ($v_i \in V$) definiran je brojem grana kojima je čvor vezan s ostatkom mreže. Dakle, ukoliko je čvor samo jednom granom vezan s ostatkom mreže govorimo o razini povezanosti $cl(v_i)=1$. Da bi bio zadovoljen zahtjev o mogućoj uspostavi alternativnih puteva u mreži svi čvorovi u mreži moraju biti barem stupnja povezanosti $cl(v_i)=2$. Taj uvjet je nužan, ali ne i dovoljan, te ga zapisujemo kao:

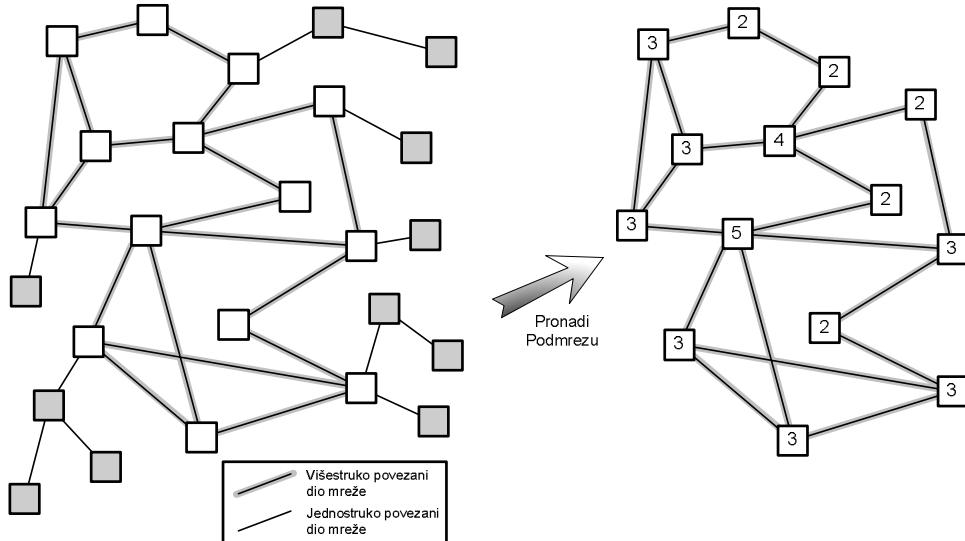
$$N=(V, E) \text{ omogućuje uspostavu alternativnih puteva} \Leftrightarrow cl(v_i) \geq 2 \text{ za } \forall v_i \in V \quad (3.1)$$

Metodu koja provjerava zadatu mrežu i kao rezultat daje odluku o tome da li mreža omogućava uspostavu alternativnih puteva naziva se *ProvjeriVišestrukuPovezanost*.

Alternativno, postoji mogućnost da je dio mreže N' ($N' \subset N$) definiran skupom čvorova V' i skupom grana E' višestruko povezan, pa se planiranje može obavljati samo na tom dijelu mreže. Praktični primjer tog slučaja bio bi kompletna mreža koja uključuje jednostruko povezani pristupni dio mreže i temeljnu mrežu. Analiza i određivanje dijela mreže koji zadovoljava uvjete višestruke povezanosti izvodi se u slučaju da mreža N ne zadovoljava uvjet (3.1). Kako odrediti višestruko povezani dio mreže? Određivanje se svodi na eliminaciju jednostruko povezanih čvorova i njima pridruženih grana – princip je dan pseudo-kodom:

```
algoritam PronadiVišestrukuPovezanuPodmrezu()
1:    $N = (V, E)$ 
2:    $N' = (V', E') = N$ 
3:   dok (postoji  $v_i \in V'$   $cl(v_i) = 1$ )
4:      $v_i$  izbaci iz  $V'$ 
5:     dok (postoji  $e_i \in E'$  kome je  $v_i$  rubni čvor)
6:       izbaci  $e_i$  iz  $E'$ 
7:   ako  $N' \neq N'$   $N'$  je višestruko povezan dio
kraj
```

Postupak pronalaženja višestruko povezane podmreže opisanom metodom je prikazan na slici 3.1.

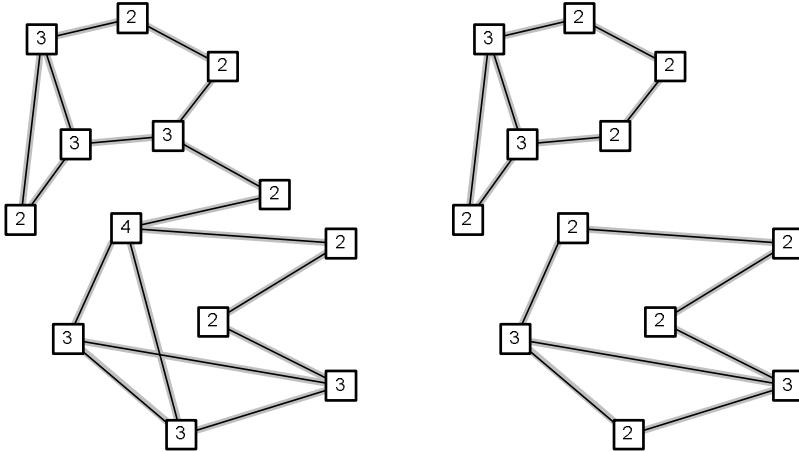


Slika 3.1: Analiza topologije i pronalaženje višestruko povezane podmreže

Rezultat provjere je informacija o tome da li zadana topologija zadovoljava nužan uvjet višestruke povezanosti, te topologija dijela mreže N' koji zadovoljava uvjet u slučaju da mreža N ne zadovoljava (dobiva se korištenjem metode *PronadiVišestrukuPovezanuPodmrezu*). Rezultat provjere povezanosti za mrežu na slici bio bi negativan zbog postojanja jednostruko povezanih rubnih čvorova. Postupak analize i pronalaženja višestruko povezane podmreže daje rezultat prikazan na desnom dijelu slike.

Na slici su posebno naznačeni čvorovi koji nisu dio višestruko povezanih struktura i koji se uklanjuju u postupku određivanja višestruko povezane podmreže. Na resultantnoj mreži naznačeni su stupnjevi povezanosti čvorova, te je vidljivo da svi zadovoljavaju uvjet (3.1) – stupanj povezanosti im je veći ili jednak 2.

Uvjet (3.1) je nužan uvjet postojanja višestruke povezanosti u mreži. Ipak, zadovoljavanje tog uvjeta nije dovoljno da bi se zaključilo da ulazna topologija predstavlja višestruku povezanu mrežu. Primjeri koje prikazuje slika 3.2 također zadovoljavaju nužni uvjet, ali ne predstavljaju višestruko povezane mreže.



Slika 3.2: Topologije koje ne predstavljaju višestruko povezane mreže

Lijevi dio slike prikazuje mrežu u kojoj svi čvorovi imaju stupanj povezanosti veći ili jednak 2, ali sama mreža nije višestruko povezana. Desni dio prikazuje topologiju koja se sastoji od dvije neovisne podmreže, od kojih je svaka višestruko povezana. Ipak, sama topologija ne definira višestruko povezanu mrežu.

Postavlja se pitanje kako obaviti analizu takvih topologija i donijeti zaključke o njihovoj nepravilnosti, te kako eventualno definirati nužan i dovoljan uvjet o višestrukoj povezanosti mreže.

Slučaj prikazan u desnom dijelu slike može se riješiti analizom podskupova mreže. Počinje se od slučajno izabrane grane, dodaju se rubni čvorovi grane, te se za svaki od njih traže susjedi. Novopronadeni čvorovi dodaju se u skup provjerenih čvorova P . Postupak se ponavlja dok je moguće odrediti susjede ili dok svi čvorovi ne budu dodani u skup P . Ukoliko po završetku postupka vrijedi $P=N$, topologija je kompaktna, tj. ne sastoji se od više višestruko povezanih struktura.

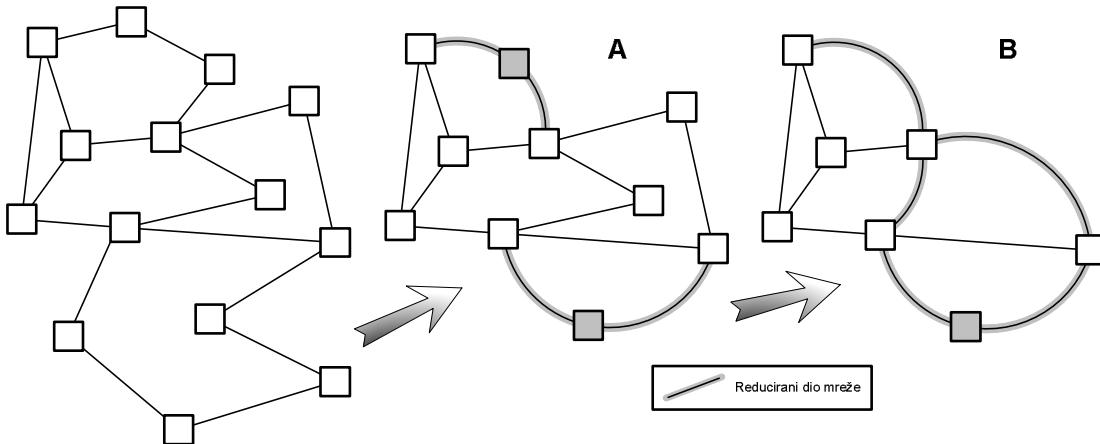
Osnovno svojstvo višestruko povezane mreže koje nalaže postojanje mogućnosti uspostave bar dva neovisna puta između svaka dva čvora u mreži može se iskoristiti za verifikaciju mreže prikazane u lijevom dijelu slike. Ukoliko za pojedini par čvorova nije moguće uspostaviti dva neovisna puta (npr. najkraći put sp i najkraći put sp' s ograničenjem da ne smije uključivati grane uključene u sp).

Upravo svojstvo moguće uspostave dva neovisna puta između svaka dva para čvorova predstavlja dovoljan uvjet za razmatranje mogućih zaštitnih mehanizama u mreži. Ukoliko se između svih parova čvorova u mreži ne mogu uspostaviti bar dva neovisna puta, nije moguće ostvariti zaštitu svih čvorova u slučaju kvara u mreži.

3.2.2. Smanjenje složenosti mrežne topologije

Ovisno o zadanoj topologiji mreže $N=(V,E)$, ponekad postoji mogućnost smanjenja složenosti topologije što u pravilu dovodi do smanjenja procesorskog vremena potrebnog za analizu mreže ili izvedbu nekog algoritma nad zadatom mrežom. Ukoliko je neku mrežu $N=(V,E)$ moguće svesti na mrežu $N'=(V',E')$ pri čemu je broj čvorova ili grana u N' manji nego u N , mrežu N' nazivamo sličnom mreži N . Iskorištavanje mogućnosti svođenja mreže N na njoj sličnu mrežu N' ovisi o postupku u okviru kog se koristi. U analizi prstenastih mreža može pomoći pri postupku određivanja prstenova u mreži i nekim drugim slučajevima.

Na slici 3.3. prikazana je inicijalna mrežna topologija i njoj slične topologije A i B.



Slika 3.3: Smanjenje složenosti topologije

Algoritam *SlicnaMreza* kao ulazne parametre uzima skup čvorova i skup grana u inicijalnoj mreži. Rezultat je skup čvorova i skup grana slične mreže, kao i informacija o eliminaciji pojedinih čvorova i grana kako bi se slična mreža N' kasnije mogla uspješno rekonstruirati u prvobitni oblik, tj. mrežu N .

```

algoritam SlicnaMreza
1:   dok postoji cvor  $v_i$  za koji vrijedi  $cl(v_i) = 2$ 
2:     ako ne postoji grana koja povezuje susjede od  $v_i$ 
3:       grupiraj grane koje izlaze iz čvora  $i$  pretvori ih u novu granu
4:         Obrisi čvor  $v_i$ 
5:         Spremi podatke potrebne za interpretaciju
6:       ponavljam
kraj

```

Postupak pretraživanja grafa ponavlja se sve dok postoje čvorovi stupnja povezanosti 2. Postupak opisan danim algoritmom svodi se na eliminaciju pojedinačnih čvorova. Alternativno bi se algoritam mogao izvesti rekursivno pri čemu bi se problem sveo na traženja lanca čvorova koji se potom reducira na jednu granu ili čvor i dvije grane. Uvjet iz drugog retka algoritma zorno je prikazan na topologiji B na slici 3.3. Taj uvjet provjerava postojanje 'iste' grane između paru čvorova. U mrežnom modelu u ovom radu višestruke grane između dva čvora nisu definirane te ih se stoga ne uključuje u razmatranje. Upravo to je razlog korištenja spomenutog uvjeta u algoritmu.

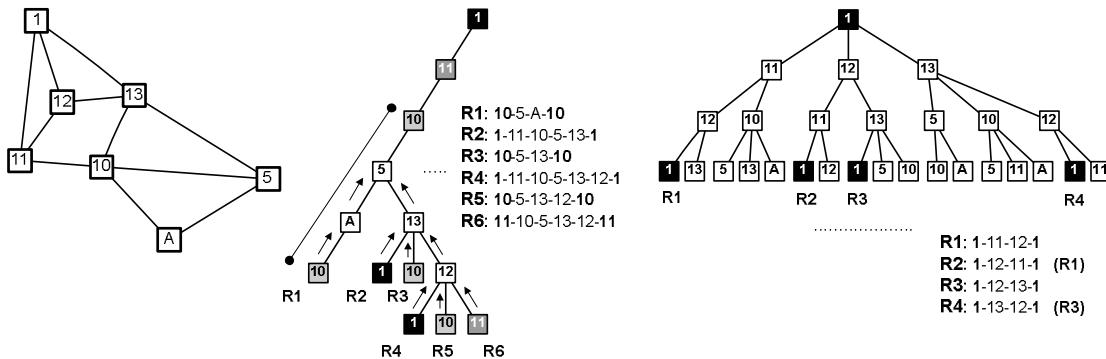
3.2.3. Određivanje prstenova u mreži

Nakon što se utvrdi da neka topologija zadovoljava uvjet višestruke povezanosti postupak planiranja prstenaste mreže nastavlja se određivanjem svih prstenova u mreži (tzv. enumeracija prstenova – *ring/cycle enumeration*). Problem određivanja prstenova skraćeno se naziva REP (*Ring Enumeration Problem*), i moguće ga je egzaktno riješiti. Ne radi se o kombinatoričkom problemu već o čistom problemu enumeracije. Međutim, kako broj mogućih prstenova u mreži raste eksponencijalno sa složenošću topologije, problem nije uvijek rješiv u realnom vremenu. Dva su osnovna načina određivanja prstenova u mreži – pretraživanje grafa mreže i kombiniranje tzv. elementarnih prstenova. Egzaktno rješavanje REP-a moguće je korištenjem obje metode, no metoda kombiniranja elementarnih prstenova niže je izvedbene složenosti (jednostavnije strukture i slično) od metode pretraživanja grafa, pa se koristi u praktičnom dijelu rada. Kako bi se smanjilo procesorsko vrijeme potrebno za određivanje prstenova, uvođe se ograničenja na moguće veličine

prstenova (minimalni i maksimalni broj čvorova), kao i na minimalni broj prstenova koji se traži. Pokazalo se da takva ograničenja znatno ubrzavaju izvedbu algoritama.

Pretraživanje grafa mreže

Metoda pretraživanja grafa s ciljem određivanja svih prstenova u mreži je vrlo intuitivna i jasna. Mreža se topološki prikazuje grafom $N=(V,E)$. Pretraživanje se svodi na izgradnju stabla za pretraživanje, te analizu pronađenih struktura. Pretraživanje se zaustavlja kad se pojavi čvor koji je već uključen u put na stablu. Dobivena struktura je uvijek prsten jer se u postupku grananja uzimaju u obzir uvjeti i ograničenja koja odmah eliminiraju drugačije strukture. Postupak je prikazan za jednostavnu mrežu – bitno je naglasiti da je primjenjena metoda smanjenja složenosti grafa kako bi se pojednostavio prikaz postupka. Pretraživanje je moguće prema stablu u dubinu i širinu. Dio postupka prikazan je na slici 3.4.



Slika 3.4: Pretraživanje grafa

Svaki od načina pretraživanja daje iste rezultate. Ovisno o topologiji mreže vrijeme potrebno za provođenje pretraživanja će biti manje ili veće. U lijevom dijelu slike prikazano je pretraživanje stabla po dubini, a u desnom dijelu pretraživanje po širini. Izvedba pretraživanja po dubini je efikasnija jer brže dovodi do različitih prstenova (različitim dubinama). Pretraživanje po širini pronalazi najprije prstenove s manjim brojem grana, te je često ponavljanje istih prstenova (različita orijentiranost grana). Dodatnim ograničenjima može se eliminirati ponavljanje prstenova.

Kombiniranje elementarnih prstenova

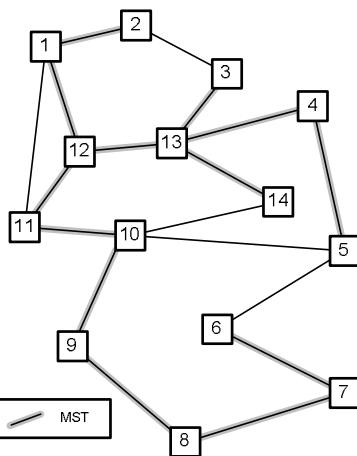
Elementarni prstenovi su osnovni prstenovi u mreži koji se dobivaju jednostavnim postupkom dodavanja grana na minimalno razapinjuće stablo [15]. Prvi korak u postupku je određivanje minimalnog razapinjućeg stabla (MST) na mreži $N=(V,E)$. Nakon toga se svaka granu e_i iz E koja nije dio MST-a pojedinačno dodaje na MST i analizira dobivena topologija – struktura koja se dobije može biti ili stablo ili struktura koja sadrži višestruko povezani dio. Prsten koji se može pojaviti naziva se elementarnim prstenom. Analiza i određivanje elementarnog prstena obavlja se korištenjem prije opisane *PronadiVisestrukoPovezanuPodmrezu* metode. Maksimalni broj elementarnih prstenova u mreži jednak je $|E|-|V|+1$, gdje su $|E|$ i $|V|$ kardinalni brojevi skupova grana i čvorova, odnosno brojevi čvorova i grana u mreži. Maksimalan broj elementarnih prstenova je jednak broju grana koje nisu uključene u razapinjuće stablo – broj grana u stablu je $|V|-1$, a ukupan broj grana $|E|$.

Nakon što se odredi skup elementarnih prstenova, njihovim kombiniranjem određuju se svi prstenovi koji se mogu formirati na zadanoj mreži N . Kombiniranje elementarnih prstenova izvodi se prema principu da se grane zajedničke elementarnim prstenovima eliminiraju, pa ostaju samo grane koje se nalaze ili u jednom ili u drugom prstenu (isključivo logičko ILI – oznaka operacije u nastavku je \oplus). Dobivene topološke strukture ne moraju nužno izgrađivati prsten – rezultati mogu

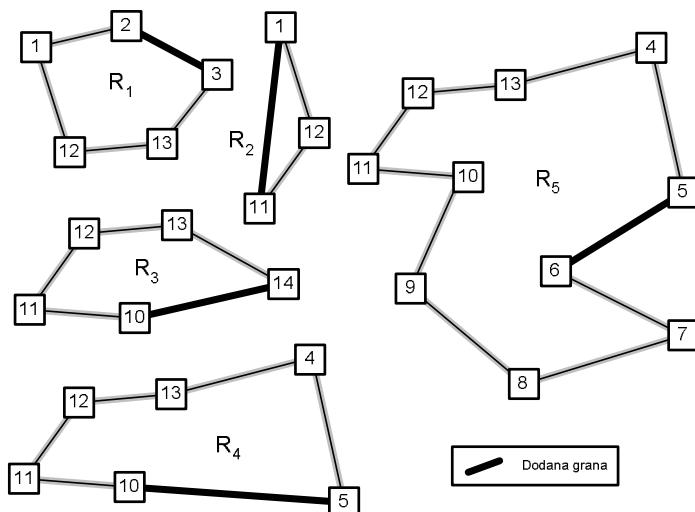
biti svakojaki – jednostruko povezane strukture, kombinacije povezanih ili odvojenih prstenova i slično. Maksimalan broj prstenova koji se mogu dobiti kombiniranjem K elementarnih prstenova je $\sum_{k=1}^K \binom{K}{k} = \sum_{k=0}^K \binom{K}{k} - 1 = 2^K - 1 = 2^{|E|-|V|+1} - 1$. Skup svih pronađenih prstenova označavamo sa RES (*Ring Enumeration Set*).

Postupak određivanja elementarnih prstenova i njihovog kombiniranja objašnjen je na mreži prikazanoj na slici 3.5. Mreža se sastoji od 14 čvorova i 18 grana. Prema prethodnom zaključku, maksimalan broj elementarnih prstenova u mreži je $(18-14+1)=5$. Maksimalan broj prstenova dobivenih kombiniranjem elementarnih jednak je $2^5 - 1 = 31$.

Na slici 3.5. posebno je naznačeno minimalno razapinjuće stablo, koje sadrži 13 grana. Preostalih 5 grana ($e_{23}, e_{1,11}, e_{10,14}, e_{5,10}, e_{56}$) nije uključeno u stablo i one se pojedinačno dodaju kako bi se eventualno izgradili elementarni prstenovi. Rezultati dodavanja grana u strukturu stabla dani su na slici 3.6. Svi pet grana u ovom slučaju sa stablom izgrađuju elementarne prstenove (R_1, R_2, R_3, R_4, R_5) koji su prikazani na slici.



Slika 3.5: Mreža i minimalno stablo



Slika 3.6: Određivanje elementarnih prstenova

Budući da se kombiniranje elementarnih prstenova izvodi korištenjem logičke operacije 'isključivo ILI' nad granama prstenova, najadekvatniji oblik zapisa strukture prstena prikazan je u tablici 3.2.

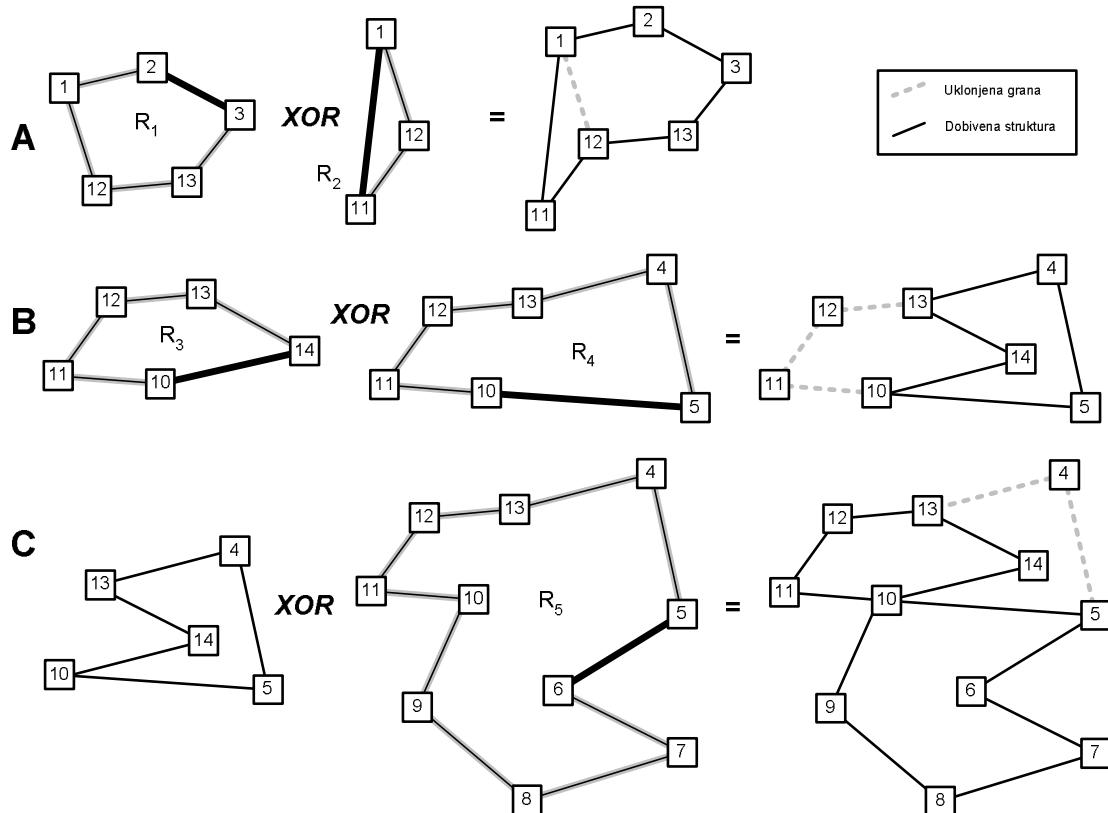
Svaki prsten prikazuje se slijedom nula i jedinica, pri čemu jedinica naznačuje da je određena grana sadržana u prstenu.

Tablica 3.2: Zapis strukture prstenova u postupku enumeracije

Prsten	e ₁₂	e ₂₃	e _{3,13}	e ₄₅	e ₅₆	e ₆₇	e ₇₈	e ₈₉	e _{9,10}	e _{10,11}	e _{11,12}	e _{1,11}	e _{12,13}	e _{13,14}	e _{10,14}	e _{5,10}	e _{1,12}	e _{4,13}
R ₁	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0
R ₂	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0
R ₃	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0	0
R ₄	0	0	0	1	0	0	0	0	0	1	1	0	1	0	0	1	0	1
R ₅	0	0	0	1	1	1	1	1	1	1	0	1	0	0	0	0	0	1

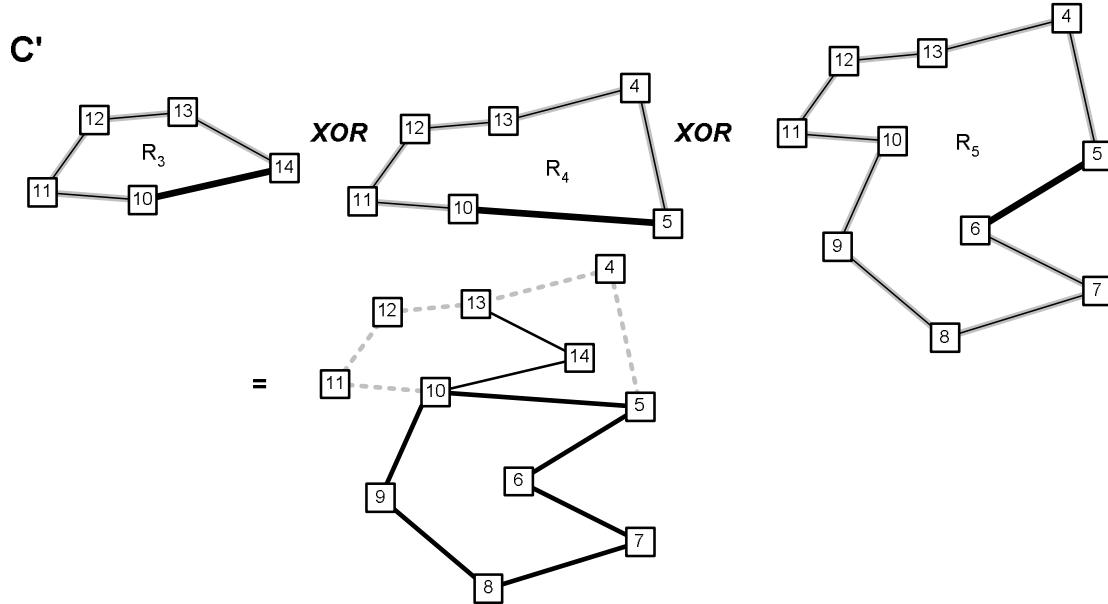
Slika 3.7 prikazuje tri primjera kombiniranja elementarnih prstenova.

U primjeru A kombiniraju se elementarni prstenovi R_1 i R_2 . Rezultat je novi prsten $R_A=R_1 \oplus R_2$, pri čemu je grana zajednička za oba elementarna prstena – $e_{1,12}$ uklonjena iz rješenja, a zadržane su sve ostale grane koje se pojavljuju u jednom od elementarnih prstenova. Identičan postupak primijenjen je i u primjeru B – kombinacija elementarnih prstenova R_3 i R_4 rezultira novim prstenom $R_B=R_3 \oplus R_4$. U ovom primjeru uklonjene su tri grane zajedničke elementarnim prstenovima. U primjeru C radi s o kombinaciji tri elementarna prstena – R_3 , R_4 i R_5 . Umjesto R_3 i R_4 koristi se u primjeru B dobivena kombinacija $R_3 \oplus R_4$. Rezultat je $R_C=R_B \oplus R_5$, odnosno $R_C=R_3 \oplus R_4 \oplus R_5$. Ovisno o definiciji operacije 'isključivo ILI' za više od dva operanda, rezultat se može razlikovati – u primjeru C za više operanada 'isključivo ILI' se interpretira kao 'modulo 2' zbrajanje. Drugi način interpretacije (rezultat operacije je 0 ako je broj pojavljivanja grane u ulaznim prstenovima veći ili jednak 2) prikazuje slika 3.8. kao primjer C'.



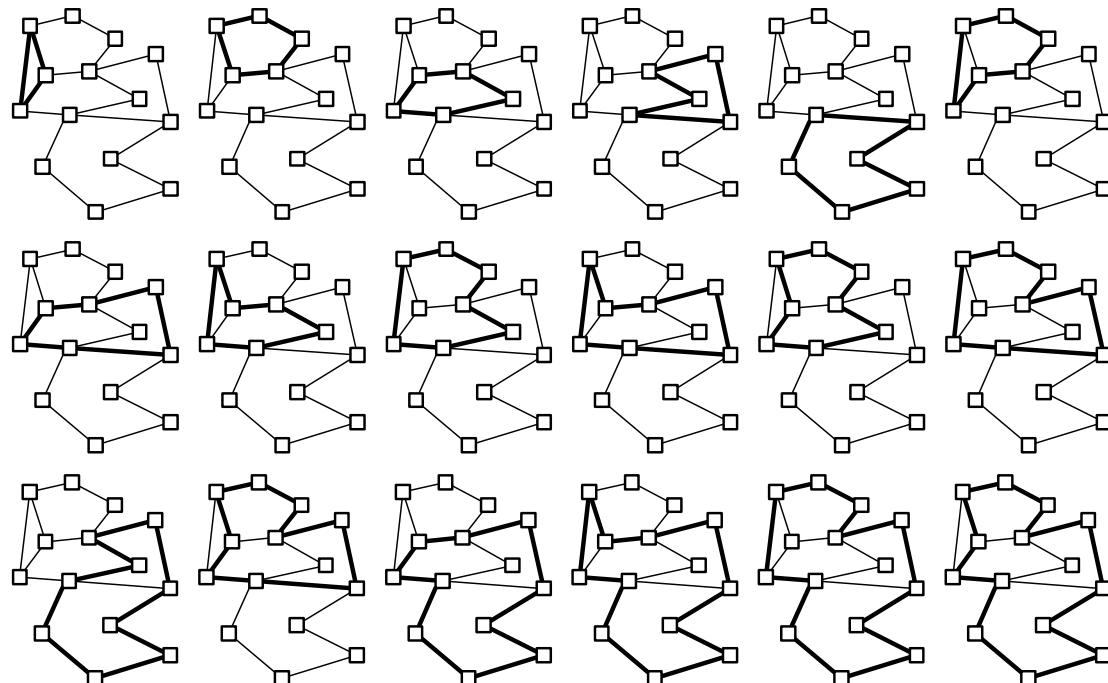
Slika 3.7: Kombiniranje elementarnih prstenova

Rezultat R_C nije jednostruki prsten već se radi o kombinaciji dva prstena (R_{C1} i R_{C2}). Očito je da vrijedi da je $R_{C1}=R_3$, odnosno $R_{C2}=R_4 \oplus R_5$. U samom postupku određivanja svih prstenova u mreži R_C se u ovom slučaju zanemaruje – topološka struktura ne zadovoljava uvjete za jednostruki prsten. Bitno je naznačiti da se rješenja sadržana u R_C ne gube - R_{C1} kao elementarni prsten je već formiran, a R_{C2} se dobije kombinacijom dva prstena!



Slika 3.8: Alternativna definicija kombiniranja prstenova

Uz korištenje alternativne definicije operacije 'isključivog ILI' na prstenovima, rješenje dobiveno u primjeru C' (slika 3.8) sadrži prsten i dvije grane koje se mogu eliminirati korištenjem metode PronadiVisestrukoPovezanuPodmrežu opisane u poglavljju 3.2.1. Rješenje je u stvari identično prstenu koji se dobije kombiniranjem elementarnih prstena R_4 i R_5 , pa ga se zanemaruje jer je isti prsten već pronađen.



Slika 3.9: Prikaz svih pronađenih prstenova

U tablici 3.3. su simbolički prikazane sve operacije koje se mogu obaviti nad definiranim elementarnim prstenovima iz primjera. Posebno su naznačeni rezultati koji predstavljaju jednostrukе prstenove i koji čine dio rješenja metode za određivanje svih prstenova u mreži.

Tablica 3.3: Moguće kombinacije elementarnih prstenova

Struktura	Grane - $e_{ij}(i,j)$																#	?	
	12	23	3,13	45	56	67	78	89	9,10	10,11	11,12	1,11	12,13	13,14	10,14	5,10	1,12	4,13	
R ₁	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	5 DA*
R ₂	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	3 DA*
R ₃	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0	0	5 DA*
R ₄	0	0	0	1	0	0	0	0	0	1	1	0	1	0	0	0	1	0	6 DA*
R ₅	0	0	0	1	1	1	1	1	1	1	0	1	0	0	0	0	0	1	10 DA*
R ₁ ⊕R ₂	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	6 DA
R ₁ ⊕R ₃	1	1	1	0	0	0	0	0	0	1	1	0	0	1	1	0	1	0	8 DA
R ₁ ⊕R ₄	1	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0	1	1	9 DA
R ₁ ⊕R ₅	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	1	13 DA
R ₂ ⊕R ₃	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	0	1	0	6 DA
R ₂ ⊕R ₄	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	1	1	7 DA
R ₂ ⊕R ₅	0	0	0	1	1	1	1	1	1	0	1	1	0	0	0	0	1	1	11 DA
R ₃ ⊕R ₄	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	0	1	5 DA
R ₃ ⊕R ₅	0	0	0	1	1	1	1	1	0	0	0	0	1	1	0	0	1	1	9 DA
R ₄ ⊕R ₅	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1	0	0	6 DA
R ₁ ⊕R ₂ ⊕R ₃	1	1	1	0	0	0	0	0	1	0	1	0	1	1	0	0	0	0	7 DA
R ₁ ⊕R ₂ ⊕R ₄	1	1	1	1	0	0	0	0	1	0	1	0	0	0	1	0	1	0	8 DA
R ₁ ⊕R ₂ ⊕R ₅	1	1	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	1	12 DA
R ₁ ⊕R ₃ ⊕R ₄	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	10 NE
R ₁ ⊕R ₃ ⊕R ₅	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	0	1	1	14 NE
R ₁ ⊕R ₄ ⊕R ₅	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	11 NE
R ₂ ⊕R ₃ ⊕R ₄	0	0	0	1	0	0	0	0	0	0	1	1	0	1	1	1	1	1	8 NE
R ₂ ⊕R ₃ ⊕R ₅	0	0	0	1	1	1	1	1	0	1	1	0	1	1	0	1	1	1	12 NE
R ₂ ⊕R ₄ ⊕R ₅	0	0	0	0	1	1	1	1	0	1	1	0	0	0	0	1	1	0	9 NE
R ₃ ⊕R ₄ ⊕R ₅	0	0	0	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	12 NE
R ₁ ⊕R ₂ ⊕R ₃ ⊕R ₄	1	1	1	1	0	0	0	0	0	1	1	1	1	1	1	0	1	1	11 NE
R ₁ ⊕R ₂ ⊕R ₃ ⊕R ₅	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0	0	1	15 NE
R ₁ ⊕R ₂ ⊕R ₄ ⊕R ₅	1	1	1	0	1	1	1	1	0	1	1	1	1	0	0	0	1	0	12 NE
R ₁ ⊕R ₃ ⊕R ₄ ⊕R ₅	1	1	1	0	1	1	1	1	1	0	0	0	1	1	1	1	1	0	14 NE
R ₂ ⊕R ₃ ⊕R ₄ ⊕R ₅	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	0	12 NE
R ₁ ⊕R ₂ ⊕R ₃ ⊕R ₄ ⊕R ₅	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	0	0	13 NE

Stupac '#' tablice označava broj grana u resultantnoj strukturi, a stupac '?' ukazuje na to da li se radi o prstenu ili ne. Postupak rezultira pronalaskom svih 18 mogućih prstenova u zadanoj mreži. Grafički prikaz svih prstenova, poredanih prema broju grana, dan je na slici 3.9.

Kako je maksimalan broj mogućih prstenova eksponencijalno ovisan o broju čvorova i grana u mreži, postupak enumeracije može zahtijevati dosta procesorskog vremena. Npr. za mrežu sa 25 čvorova i 40 grana maksimalan broj prstenova bio bi 2^{16} , za mrežu od 35 čvorova i 60 grana čak 2^{26} itd. Zbog toga se pri implementaciji metode dodaju ograničenja – definira se maksimalni broj prstenova koje je potrebno odrediti, minimalni i maksimalni broj čvorova po prstenu. Mogu se odrediti i vremenska ograničenja, tako da se nakon nekog vremena postupak prekine i ponude do tog trenutka pronađeni prstenovi. U praksi se to pokazalo dovoljno dobrim, jer najčešće nema

smisla provoditi daljnje faze postupka planiranja na ogromnim skupovima mogućih prstenova. Ograničenja na nekoliko stotina prstenova veličina najčešće prisutnih u realnim mrežama su i više nego dovoljna.

3.2.4. Usmjeravanje prometnih zahtjeva

Realizacija svakog definiranog prometnog zahtjeva mora biti moguća u resultantnoj mreži. To znači da se za dobivenu topologiju mora moći odrediti put između zadanog para čvorova. Ukoliko je mreža izgrađena od prstenova taj put će prolaziti određenim brojem prstenova. Međutim, definirani put neće nužno biti identičan putu kojim će se prometni zahtjev doista usmjeravati. Naime, usmjeravanje zahtjeva unutar svakog pojedinog prstena izvodi se ovisno o tipu prstena i rješava se kao dio, u poglavlju 2.3 objašnjenoj, problema dimenzioniranja prstena (RDP).

U automatiziranom postupku planiranja prstenastih mreža potrebno je prije određivanja prstenova koji će se koristiti odrediti mjeru koja će omogućiti vrednovanje različitih prstenova i samim time odrediti njihovo uključivanje u rezultat. Pri analitičkom pristupu planiranja mreže, u slučajevima kad su prstenovi koji se koriste unaprijed definirani, takva mjera nije potrebna.

Ipak, u oba slučaja postoji potreba za korištenjem algoritama za usmjeravanje prometnih zahtjeva. Ako su prstenovi već definirani, algoritmom usmjeravanja se određuje kojim prstenovima će prometnih zahtjev prolaziti, a ako određivanje prstenova tek predstoji algoritmi usmjeravanja koriste se da bi se predodredili putevi kojima će zahtjevi prolaziti, te se na temelju tih usmjeravanja određuju prstenovi koji najbolje mogu prenijeti zahtjeve tim putevima. Usmjeravanje u mrežama s definiranim prstenovima nije razmatrano jer se najčešće svodi na trivijalni slučaj pronalaženja minimalnog puta i određivanje prstenova koji sadrže grane tog puta. Predloženi algoritmi mogu se primijeniti i u tom slučaju.

Postupci usmjeravanja najčešće se svode na traženje najkraćeg puta uz zadavanje eventualnih ograničenja. Tako se osim najkraćeg puta (SP – *Shortest Path*) za usmjeravanje može koristiti drugi najkraći put (SSP – *Second Shortest Path*), drugi neovisni najkraći put (DSSP – *Disjoint Second Shortest Path*) i sl. Takvi postupci izvode se neovisno za svaki pojedini zahtjev i nema nikakve ovisnosti između usmjeravanja različitih zahtjeva.

Ponekad je bolje koristiti prilagodljive algoritme za traženje najkraćeg puta – u tom slučaju analiziraju se svi zahtjevi i usmjeravanje pojedinog ovisi i o prije usmjerjenim zahtjevima. U ovom radu analizirana su dva jednostavna algoritma kojima je za cilj minimizirati odnosno maksimizirati broj korištenih grana u mreži. Minimiziranje broja korištenih grana postiže se usmjeravanjem takvim da se zahtjevi pokušaju provesti kroz što više već korištenih grana. Na taj način te grane postaju opterećenije, no u mreži postoji i određeni broj grana koje se uopće ne koriste. To može utjecati na eliminaciju određenog broja prstenova. Maksimiziranje broja grana postiže se usmjeravanjem takvim da se zahtjevi usmjeravaju putevima koji još nisu korišteni (ako takvi postoje). Budući da usmjeravanje trenutnog zahtjeva ovisi o svim prethodnim usmjeravanjima, različitim redoslijedima zahtjeva dobivaju se različiti rezultati.

Vrednovanje rezultata usmjeravanja bespredmetno je ukoliko se ne sagledava kompletan postupak planiranja, dakle krajnji rezultat – prekrivanje mreže prstenovima. U postupku implementacije moguće je provjeriti očekivanja – na temelju kalkulacija o opterećenosti grana, prosječnoj opterećenosti i broju grana koje se ne koriste uočavaju se možebitni problemi i analiziraju dobiveni rezultati.

U nastavku su ukratko opisani algoritmi koji se koriste i koji su implementirani u praktičnom dijelu rada. Dan je i pregled spomenutih karakterističnih vrijednosti koje upućuju na 'efikasnost' algoritama. Ponovno, generalno gledajući ti parametri ne garantiraju kvalitetu prekrivanja mreže prstenovima, ali se neke pravilnosti mogu pronaći.

Bitno je naglasiti da nakon što se odrede prstenovi koji prekrivaju mrežu usmjeravanje zahtjeva ne mora nužno pratiti puteve definirane u ovoj fazi. Zahtjevi se usmjeravaju unutar prstena prema pravilima koja će rezultirati minimalnim kapacitetima prstenova, što ne mora uopće odgovarati

dionicama puta koji je definiran prije određivanja prstenova. Ta činjenica navodi na zaključak da bi se naknadnim preusmjeravanjima (nakon određivanja prstenova) možda moglo poboljšati rezultate postupka.

Najkraći putevi

Usmjeravanje najkraćim putevima određuje se standardnim Dijkstrinim algoritmom [4]. Složenost algoritma je $O(N^2)$, gdje je N broj čvorova u mreži. Težina grana definira se proizvoljno – u praksi se radi najčešće o duljini grane ili pak se težina postavlja na 1, što se koristi u određivanju puteva koji prolaze najmanjim brojem grana (*min-hop* putevi). U slučaju postojanja puteva jednakih težina vrši se usporedba prema alternativnom kriteriju (npr. ako je primarni kriterij duljina, alternativni je broj *hop-ova*). Implementirana metoda nazvana je SPR.

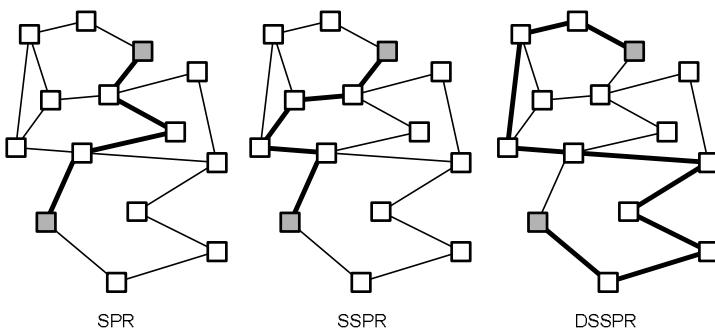
Korištenje najkraćeg puta za usmjeravanje zahtjeva u pravilu će koristiti najmanji broj grana u mreži (uz *min-hop* određivanje puta). Ovisno o strukturi mreže i zahtjeva, a s ciljem smanjivanja broja prstenova koji prekrivaju mrežu i prenose zahtjeve, korištenje drugih puteva koji prolaze većim brojem grana može dovesti do boljih konačnih rezultata.

Da bi se proanalizirao i taj utjecaj, koriste se dva algoritma – algoritam za određivanje drugog najkraćeg puta, te algoritam za određivanje neovisnog drugog najkraćeg puta.

Neovisni drugi najkraći put svodi se na ponavljanje Dijkstrinog algoritma uz zanemarivanje grana koje su dobivene kao najkraći put. Implementirana metoda nazvana je DSSPR.

Drugi najkraći put može se odrediti bilo kojom metodom za određivanje k -tog najkraćeg puta. U praktičnom dijelu rada implementiran je algoritam prema [4]. Metoda je nazvana SSPR. Algoritam se svodi na traženje najkraćeg puta Dijkstrinim algoritmom, te pronaalaženje drugog najkraćeg puta eliminacijom pojedinih grana uključenih u najkraći put. Složenost algoritma je $O(N^3)$, jer se radi o N ponavljanja standardnog Dijkstrinog algoritma.

Na slici 3.10 su prikazani rezultati algoritama kod usmjeravanja jednog zahtjeva na testnoj mreži (slika 3.5) između čvorova 3 i 9.



Slika 3.10: Usmjeravanja najkraćim putevima

Može se uočiti da DSSPR usmjeravanje koristi velik broj grana s ciljem da zaobide grane koje se koriste u SPR. Gledano s aspekta sigurnosti prometa, neovisnost puteva je postignuta na razini grana – postoje zajednički čvorovi kojih kvar može onemogućiti komunikaciju – u tom slučaju potrebno je dodatno preusmjeravati promet uz dana ograničenja. Prezentirane metode nisu namijenjene određivanju alternativnih puteva već definiranju različitih mogućih usmjeravanja unutar mreže. Cilj je dobiti određeni skup usmjeravanja nad kojima se vrše daljnji proračuni i analiziraju konačni dobiveni rezultati.

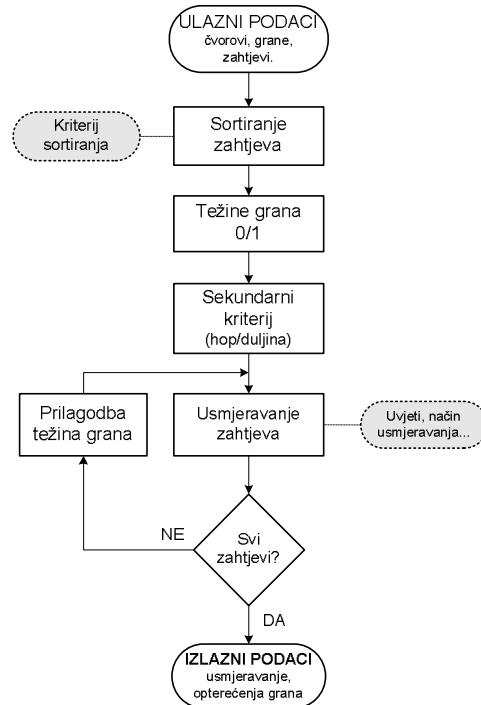
Adaptivno usmjeravanje

Budući da je uočeno da se na rezultate kasnijih faza postupka može utjecati različitim modelima usmjeravanja, dodane su dvije metode adaptivnog usmjeravanja. Osnovna razlika metoda adaptivnog usmjeravanja i metoda za određivanje najkraćih puteva opisanih u prethodnom poglavlju jest u tome da je pri adaptivnom usmjeravanju svako usmjeravanje ovisno o prethodnim usmjeravanjima, te se postupak mora provesti za sve zahtjeve, što kod određivanja najkraćih puteva nije slučaj.

Prva metoda MxFER (*MaX Free Edges Routing*) ima za cilj usmjeriti zahtjeve tako da se iskoristi što manje grana u mreži, tj. da što više grana ostane neiskorišteno. Druga metoda MiFER (*MIn Free Edges Routing*) ima suprotan cilj – zahtjeve se nastoji usmjeriti tako da se iskoristi što više grana, tj. da je što manje grana neiskorišteno. Općeniti princip funkciranja metoda za adaptivno usmjeravanje prikazan je blok dijagramom na slici 3.11.

Rezultati koji se dobivaju ovise o redoslijedu usmjeravanja – jedan od parametara je definiranje načina sortiranja zahtjeva. Nakon što su zahtjevi poredani prema kriteriju prvi zahtjev se usmjeri najkraćim ili drugim najkraćim putem. Potom se redefiniraju težine grana – kod MxFER sve korištene grane dobivaju težinu 0, a neiskorištene težinu 1. Kod MiFER korištene grane dobivaju težinu 1, a neiskorištene težinu 0. Korištenjem kombinacija 0 i 1 za težine grana algoritmi će sigurno odabrati grane s težinama 0. Nakon što su težine grana redefinirane nastavlja se s usmjeravanjem zahtjeva. Kod MxFER algoritmi za najkraće puteve birati će grane koje su već korištene, a kod MiFER grane koje još nisu korištene.

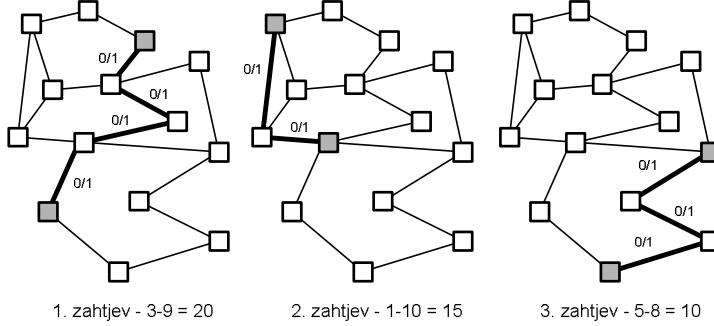
Budući da će zbog težina 0 i 1 veći broj puteva imati jednaku težinu, bitno je definirati sekundarni kriterij odabira grana u rješenje. Sekundarni kriterij najčešće je broj *hop-ova* ili duljina grana.



Slika 3.11: Shematski prikaz postupka adaptivnog usmjeravanja

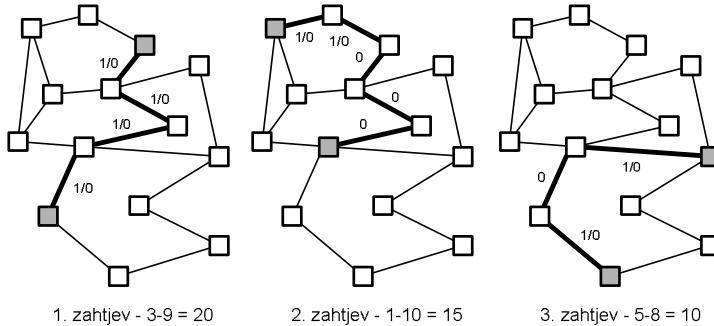
Primjer usmjeravanja u testnoj mreži uz definiranje tri zahtjeva pri korištenju MxFER i MiFER dan je na slikama 3.12 i 3.13. Prije provedbe algoritama usmjeravanja zahtjevi se poredaju od najvećeg prema najmanjem. Prikazane su dvije izvedbe MxFER metode – razlikuju se u sekundarnom kriteriju koji se koristi. Na odabranim granama označeni su parametri težine – oznaka 1/0 označava

da je prilikom odabira grane težina promijenjena sa 1 u 0. Već korištene grane ne mogu mijenjati težinu, te je ona kod MiFER uvijek na 1, a kod MxFER uvijek na 0.



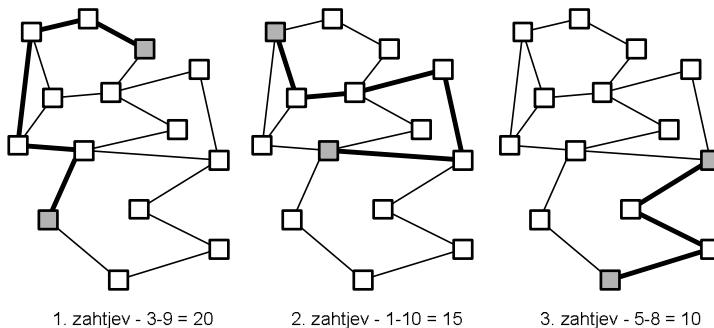
Slika 3.12: Primjer usmjeravanja za MiFER metodu

Na slici 3.12 je vidljivo da MiFER metoda odabire najkraće puteve za usmjeravanje zahtjeva, vodeći pri tome računa o tome da put sadrži što manje već korištenih grana. U primjeru se korištene grane ne ponavljaju. Sekundarni kriterij u izvedbe prikazane metode je minimalni broj *hop-ova* – ovu izvedbu metode označavamo sa MiFER1.



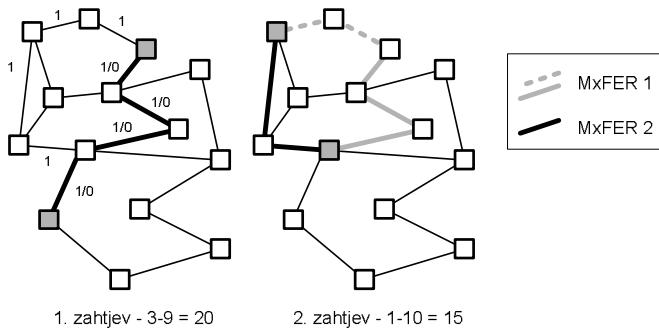
Slika 3.13: Primjer usmjeravanja za MxFER metodu

Kod metode MiFER2 sekundarni kriterij je maksimalni broj *hop-ova* – rezultati dobiveni metodom MiFER2 su prikazani na slici 3.14. Broj nekorištenih grana u ovom slučaju se smanjuje (kod MiFER1 to je 9, dok je kod MiFER2 5) – očito da primjena drugačijeg sekundarnog kriterija 'raspršuje' zahtjeve po mreži što je i cilj metode. Ipak, ostaje upitna vrijednost metode i njen utjecaj na kasnije faze dizajna.



Slika 3.14: Utjecaj sekundarnog kriterija u metodi MiFER

Slika 3.13 pokazuje rezultate dobivene MxFER metodom uz sekundarni kriterij maksimalnog broja *hop-ova*. Izvedbu uz taj kriterij označavamo sa MxFER1. Alternativna izvedba uz kriterij minimalnog broja *hop-ova* prikazana je na slici 3.15.



Slika 3.15: Izvedba MxFER metode uz promjenu sekundarnog kriterija

Na slici su posebno naznačene grane koje se koriste za usmjerenje kod MxFER1 izvedbe. Sekundarni kriterij utječe na izvedbu ukoliko postoje putevi s jednakim brojem neiskorištenih grana – MxFER1 i MxFER2 u ovom primjeru usmjeravaju drugi zahtjev različitim putevima, ali uz jednak broj neiskorištenih grana (crtkani prikaz na istoj slici). Ako je sekundarni kriterij maksimalni broj *hop-ova*, tada se usmjerenje sigurno djelomično poklapa sa usmjerenjem nekog drugog zahtjeva. Jasno je da u tom slučaju opterećenje pojedinih grana raste, no zbog takve agregacije zahtjeva utjecaj na kasniji tijek postupka planiranja može biti veći.

Zbog nemogućnosti efikasnog prikaza usmjerenja svih zahtjeva u mreži, rezultati su prodiskutirani prema karakterističnim vrijednostima dobivenim za izvođenje svih algoritama. Rezultati su prikazani u tablici 3.4. Testna mreža je prikazana na slici 3.5. Nad mrežom (14 čvorova, 18 grana) je definirano 25 slučajno generiranih zahtjeva. Rezultati su zapisani u formatu $a/b/c$, gdje je a broj neiskorištenih grana u mreži, b prosječno opterećenje iskorištenih grana, a c prosječno opterećenje svih grana u mreži.

Tablica 3.4: Rezultati različitih metoda usmjerenja

Metoda	težina=HOP						težina=duljina					
SPR	1/95/90						2/104/92					
SSPR	2/156/139						1/147/139					
DSPR	0/153/153						0/166/166					
Kriterij sortiranja zahtjeva												
	max-min	min-max	SP max	SSP max	DSP max	max-min	min-max	SP max	SSP max	DSP max		
MxFER1	5/162/117	5/169/115	5/168/121	5/197/142	5/169/122	5/165/119	5/177/128	5/183/132	5/183/132	5/169/122		
MxFER2	5/152/110	5/152/110	5/168/121	5/159/114	5/142/102	5/151/109	5/152/110	5/168/121	5/159/114	5/148/106		
MiFER1	0/97/97	0/91/91	0/92/92	0/95/95	0/92/92	0/97/97	0/91/91	0/92/92	0/95/95	0/93/93		
MiFER2	1/105/99	0/91/91	1/98/93	1/102/96	0/95/95	0/97/97	0/91/91	0/95/95	1/102/96	0/95/95		

Očito je da postupci rezultiraju različitim rješenjima usmjerenja zahtjeva. Usmjerenje zahtjeva korištenjem najkraćih puteva (SPR) daje najniže prosječno opterećenje grana u mreži (90), uz jednu neiskorištenu granu. Niska prosječna opterećenja daje i MiFER1 metoda, no kod usmjerenja dobivenih tom metodom iskorištenе su sve grane. Ostale metode najkraćih puteva (SSPR i DSPR) povećavaju opterećenost grana u mreži, kao i prosječnu opterećenost. MxFER metoda rezultira opterećenistima korištenih grana u rangu SSPR i DSPR, ali uz nekorištenje većeg broja grana što smanjuje prosječnu opterećenost grana u mreži. Veća opterećenost grana javlja se u rezultatima metode MxFER1. U testnoj mreži od 18 grana, MxFER ostavlja 5 grana neiskorišteno, što je gotovo 28% grana. Ovisno o izvedbi metoda za prekrivanje mreže prstenovima taj rezultat bi mogao imati veći utjecaj na smanjenje broja prstenova.

3.2.5. Prekrivanje mreže prstenovima

U automatiziranom postupku planiranja prstenastih mreža glavni problem je određivanje podskupa prstenova **RCS** (*Ring Cover Set*) koji prekrivaju mrežu. Radi se o problemu prekrivanja mreže prstenovima RCP (*Ring Covering Problem*). Postupak ovisi o svim prethodnim fazama planiranja i svaka od njih može imati manji ili veći utjecaj na krajnje rezultate. Ulazni podaci u postupak određivanja prstenova koji prekrivaju mrežu su:

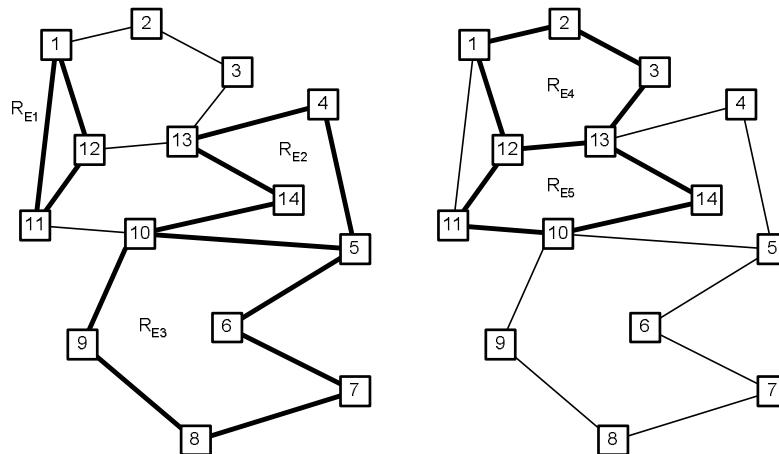
- skup mogućih prstenova u mreži (**RES**)
- prometni zahtjevi i njihovo usmjeravanje na razini mreže

Ti podaci mogu biti definirani od strane dizajnera mreže ili pak se određuju metodama opisanim u prethodnim poglavljima.

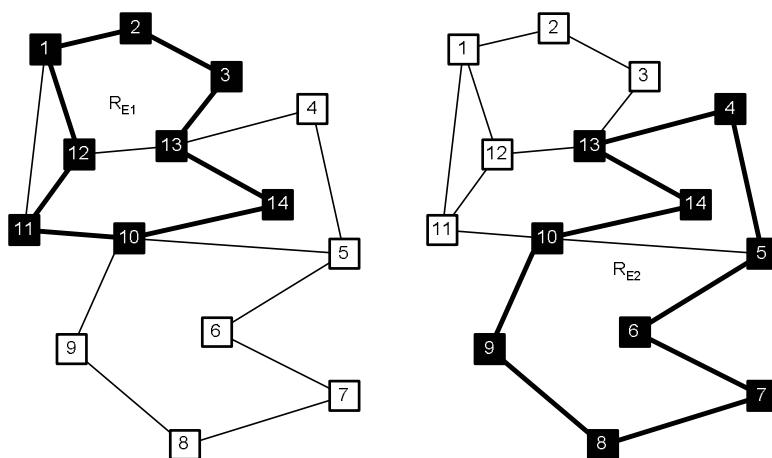
Moguće je definirati tri načina prekrivanja mreže prstenovima:

- prekrivanje grana
- prekrivanje čvorova
- prekrivanje zahtjeva

Modeli prekrivanja grana i čvorova u mreži prikazani su na jednostavnom primjeru mreže na slikama 3.16 i 3.17.

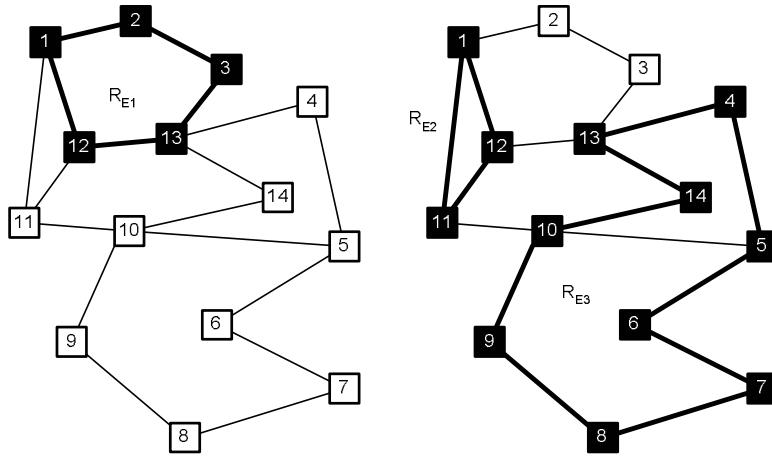


Slika 3.16: Primjer prekrivanja svih grana u mreži



Slika 3.17: Primjer prekrivanja svih čvorova u mreži I

Može se uočiti razlika u prekrivanju – dok se kod prekrivanja grana sve grane uključuju u neki od prstenova, kod prekrivanja čvorova postoji mogućnost da pojedine grane nisu prekrivene.



Slika 3.18: Primjer prekrivanja svih čvorova u mreži II

Posljedica je da kod prekrivanja grana postoji više grana zajedničkim različitim prstenovima, tj. postoji više preklapanja među prstenovima. U primjerima je prekrivanje svih grana u mreži izvedeno korištenjem 5 prstenova, dok je prekrivanje čvorova izvedeno sa samo dva prstena. Jedan od alternativnih načina prekrivanja čvorova prikazan je na slici 3.18, pri čemu se prekrivanje svih čvorova izvodi pomoću tri prstena.

Poseban slučaj je prekrivanje mreže prema zahtjevima – prstenovi koji prekrivaju mrežu omogućavaju realizaciju svih zahtjeva, no ne moraju nužno prekrivati sve grane ili sve čvorove u mreži. Ukoliko svi čvorovi sudjeluju u generiranju ili prihvaćanju prometnih zahtjeva, prekrivanje prema zahtjevima se svodi na prekrivanje prema čvorovima.

Najčešće korišten pristup za rješavanje RCP-a je korištenje standardnih pohlepnih (*greedy*) metoda. Sama izvedba i uvjeti korišteni u metodama razlikuju se od autora do autora, što usporedbe rezultata čini teškim. Vrednovanje rezultata dobivenih spomenutim metodama također je složeno, i ovisi prije svega o kriteriju za vrednovanje kompletne mreže. Kako *greedy* metode u pravilu pronalaze lokalne optimume, teško je pretpostaviti koliko su rezultati uistinu kvalitetni. Izrada matematičkog modela koji bi bio uskladiv sa tim metodama također nije trivijalna, te nije izvedena kao dio ovog rada. Da bi se ipak dobio neki odnos među rezultatima, korišten je jednostavan matematički model predložen u [15]. Alternativno je moguće izvesti vrednovanje rezultata za primjer jednostavne mreže – prekrivanje prstenovima svelo bi se na enumeraciju mogućih rješenja, vrednovanje svakog od njih, te na kraju usporedbu sa rezultatima dobivenim implementiranim metodama.

Nužno je razlučiti dva kriterija vrednovanja:

- kriterij vrednovanja pojedinog prstena
- kriterij vrednovanja resultantne mreže

Da bi se moglo odrediti koji prsten je bolji za uključivanje u konačno rješenje potrebno je odrediti težinu svakog pojedinog prstena. Adekvatno definiranje težinske funkcije za pojedinačne prstenove može predstavljati problem. Nakon što se odrede svi prstenovi, potrebno je odrediti težinsku vrijednost cijele mreže kako bi se mogla izvesti usporedba s ostalim rezultatima. Definiranje težinske funkcije za vrednovanje kompletne mreže također može biti problematično.

Kao dio rada analizirana su dva principa prekrivanja prstenovima – oba su izvedena *greedy* metodama uz uvjet prekrivanja prstenovima svih prometnih zahtjeva – svi zahtjevi se mogu realizirati korištenjem resultantnih prstenova, bez garancije da su sve grane i/ili čvorovi u mreži prekriveni.

Prvi princip prekrivanja je karakterističan zbog toga jer se detaljno razmatra svaki pojedini prometni zahtjev – zahtjev se usmjerava kroz prstene koji se uključuju u rješenje, po potrebi ga se

rastavlja u više zahtjeva ukoliko se radi o međuprstenastom zahtjevu, te ga se usmjerava unutar prstenova koji se uključuju u rješenje. Takav pristup rješavanju problema omogućava detaljnu analizu rezultata usmjeravanja svakog pojedinog zahtjeva. Njegova najveća prednost je ta da se ovisno o definiranim uvjetima može smanjiti ovisnost rezultata o inicijalnom usmjeravanju prometnih zahtjeva na razini mreže.

Drugi princip se u potpunosti oslanja na inicijalno usmjeravanje zahtjeva na razini mreže. Prilikom određivanja prstenova koji se uključuju u mrežu vodi se računa jedino o težinama grana u mreži – analizom težina grana za svaki pojedini prsten može se odrediti efikasnost prstena i prema tom (i sličnim) parametrima odrediti prstenove koji predstavljaju najbolja rješenja u pojedinim iteracijama postupka.

Definiranje kriterija vrednovanja prstenova

Rezultati prekrivanja mreže prstenovima ovise o podacima koji ulaze u postupak: prstenovima koje je moguće realizirati (skup mogućih prstenova \mathbf{RES}) i usmjeravanju prometnih zahtjeva.

Ulagano usmjeravanje zahtjeva nije definirano na razini prstenova već na razini kompletne mreže, kao da se radi o *mesh* mreži. Na temelju informacija o usmjeravanju moguće je odrediti unutar kojih od ponudenih prstenova se pojedini zahtjevi mogu zatvoriti. Generalno gledajući, cijena pojedinog prstena će ovisiti o njegovoj veličini (broju čvorova), kapacitetu (samim time i usmjeravanju zahtjeva kroz prsten) i prometu koji izlazi i ulazi u prsten (međuprstenasti promet). Često se analizira i iskoristivost prstena kao parametar njegove cijene – kako bi se smanjili nepotrebni troškovi nastojeći se realizirati prstene sa što višom razinom iskoristivosti – iskoristivost se u praksi koristi kao mjera efikasnosti prstena. Doduše, inzistiranje na izgradnji što više iskoristivih prstenova može biti dvosjekli mač u slučaju većeg porasta prometnih zahtjeva u realnoj mreži – u dobro iskorištene prstene teško je uklopiti nove zahtjeve bez povećanja kapaciteta (drugim riječima, bez dodatnih ulaganja). Najčešće se u postupku planiranja o tome vodi računa pa se i definiraju prometni zahtjevi za više razdoblja i radi dodatna analiza isplativosti.

Dakle, cijenu prstena \mathbf{R} , P_R možemo definirati kao funkciju veličine prstena $|R|$, kapaciteta $C(R)$, prometnih zahtjeva $d_{ij} \in D$, tipa zahtjeva $t(d_{ij})$ (tip zahtjeva unutar prstena može se definirati kao lokalan ukoliko su izvorišni i odredišni čvor dio prstena, odnosno međuprstenasti ukoliko samo dio puta prometnog zahtjeva prolazi prstenom) i njihovog usmjeravanja unutar prstena $x(d_{ij})$ (analogno modelu jednostavnog prstena, x označava smjer kojim se zahtjev usmjerava unutar prstena):

$$P_R = f_x(|R|, C(R), D, t(D), x(D))$$

Već na prvi pogled je jasno da određivanje cijene prstena nije nimalo jednostavno – iako možemo definirati ovisnosti cijene i pojedinih karakterističnih vrijednosti, pitanje je definiranja same funkcije f_x . U literaturi se obično definiraju težinske funkcije koje najčešće ne obuhvaćaju sve utjecaje, ali omogućuju vrednovanje prstenova [10,16]. Budući da o tom kriteriju ovise svi daljnji postupci, obično se realizira nekoliko težinskih funkcija, pa se konačna rješenja uspoređuju i odabire najbolje. Na taj način se može smanjiti pohlepnost *greedy* metoda i eventualno pronaći više lokalnih optimuma i odlučiti se za najbolji.

Težinske funkcije se većinom baziraju na određivanju razine iskoristivosti prstena – ukoliko je prsten dobro iskorišten najvjerojatnije predstavlja dobro rješenje. Iskoristivost se pokazuje kao dobra mjera koja definira utjecaj samih prometnih zahtjeva, ali i kapaciteta prstena. Ipak, na taj način se ne ostvaruje vrednovanje međuprstenastog prometa koji može imati izravan utjecaj na težinu kompletne mreže (u postupku planiranja nastojeći se smanjiti promet između prstenova). Definicija iskoristivosti prstena može biti različita, ovisno o principu koji se koristi pri određivanju prstenova koji prekrivaju mrežu.

Karakteristični parametri prstena koji se koriste za definiranje težinskih funkcija definiraju se na sličan način u oba spomenuta pristupa. Različitim kombinacijama iznijetih funkcija moguće je realizirati alternativne težinske funkcije, no one ovdje nisu razmatrane. Samo definiranje funkcija

ovisit će u realnom postupku planiranja i o dodatnim parametrima koje dizajneri mogu definirati. Tako je na primjer za svaki prsten moguće definirati funkcije koje će realnije vrednovati mrežu (npr. težina ovisna o praktičnoj realizaciji prstena – tipovima i cijenama veza, kapacitetima čvorova i sl.). Bitno je naglasiti da se ne radi o uskladivim težinskim funkcijama – prstenovi vrednovani istim funkcijama mogu se usporedavati, no konačni rezultati prekrivanja ne mogu se vrednovati na osnovu različito definiranih težina prstenova. To bi se moglo izvesti tek uz normiranje funkcija, no kao što će se u nastavku pokazati to i nije nužno jer se na razini mreže definiraju alternativni kriteriji vrednovanja.

Unutarprstenasto usmjeravanje i kriterij vrednovanja

Nakon što se za svaki prsten $R \in RES$ odredi koji zahtjevi prolaze kroz njega ($D_R \subseteq D$), obavlja se usmjeravanje zahtjeva unutar prstena – na taj način se odredi kapacitet prstena $C(R)$ i smjerovi usmjeravanja. Zahtjevi koji prolaze kroz prsten nalaze se u dva skupa – skupu lokalnih zahtjeva D_{Lokal} i skupu međuprstenastih zahtjeva D_{Inter} (vrijedi $D_R = D_{Lokal} \cup D_{Inter}$). Za svaku granu $e \in E_R$ određuje se opterećenje $w(e)$, te se na osnovu tih podataka definiraju težinske funkcije. Opterećenje grane $w(e)$ jednako je sumi prometnih zahtjeva koji prolaze tom granom. Kapacitet prstena u stvari je jednak maksimalnom opterećenju grana $C(R) = \max(w(e))$ ukoliko se radi o dvosmjernom prstenu, odnosno sumi zahtjeva ukoliko se radi o jednosmjernom prstenu. Na taj način definiraju se sve karakteristične vrijednosti prstena koji se koriste za definiranje težine prstena.

U nastavku su navedene težinske funkcije korištene za vrednovanje prstenova pri realizaciji pristupa koji se oslanja na unutarprstenasto usmjeravanje i analizu prometnih zahtjeva – dodatne težinske funkcije mogu se pronaći u [10,16].

$$f_1 = \frac{|D_{Lokal}|}{|D_{Inter}|} \cdot C(R) \quad (3.2)$$

$$f_2 = \frac{|R| \cdot C(R)}{\sum_{\forall e_i \in E_R} w(e_i)} \quad (3.3)$$

$$f_3 = \alpha \cdot \frac{\sum_{\forall e_i \in E_R} w(e_i)}{|R| \cdot C(R) - \sum_{\forall e_i \in E_R} w(e_i)} + (1 - \alpha) \cdot \frac{2 \cdot |D| - |D_{Inter}|}{2 \cdot |D|} \quad (3.4)$$

$$f_4 = \frac{\sum_{\substack{d_{ij} \in D_{Lokal} \\ d_{ij} \in D_{Inter}}} d_{ij}}{\sum_{d_{ij} \in D_{Inter}} d_{ij}} \quad (3.5)$$

Težinska funkcija f_1 ovisi o broju lokalnih i međuprstenastih zahtjeva koji se usmjeravaju prstenom, te o kapacitetu prstena. Tom funkcijom preferiraju se prstenovi sa što većim brojem lokalnih zahtjeva i što većim kapacitetom – najčešće će to rezultirati prstenovima sa velikim lokalnim prometom. Različitim interpretacijama lokalnih i međuprstenastih zahtjeva mogu se varirati vrijednosti funkcije. Rezultantne mreže dobivene selekcijom prstenova ovim kriterijem najčešće se baziraju na jednom većem prstenu kojim prometuje velik broj zahtjeva, te manjim prstenovima koji skupljaju preostale zahtjeve i po potrebi ih dovode do čvorova koji su elementi glavnog prstena.

Težinska funkcija f_2 rezultira težinom koja ovisi o veličini i iskoristivosti prstena. Zbog direktnog utjecaja veličine prstena na težinu, ova će težinska funkcija rezultirati većim prstenovima većeg kapaciteta. Funkcija se može promatrati kao mjera redundancije u prstenu – brojnik označava instalirane kapacitete u prstenu (ukupno instaliran kapacitet – svaka grana dimenzionira se na kapacitet jednak kapacitetu prstena), a nazivnik radne kapacitete (suma težina grana). Ukoliko se

nastoji izgraditi mrežu s višom redundancijom koristi se funkcija f_2 , dok se u slučaju zahtjeva za što efikasnjom mrežom kao težinska funkcija može koristiti invertirana težinska funkcija $f_2'=1/f_2$, koja predstavlja iskoristivost mreže. Težinska funkcija f_3 definirana je prema [10] uz korištenje parametra α kojim je moguće mijenjati utjecaj parametra iskoristivosti mreže i veličine prstena. Prvi pribrojnik predstavlja mjeru efikasnosti mreže, a drugi ovisnost o ukupnom broju zahtjeva i broju međuprstenastih zahtjeva. Povećanje parametra α povećava utjecaj iskoristivosti mreže na rezultate.

Funkcija f_4 koristi se za odabir prstenova u kojima prevladavaju lokalni zahtjevi nad međuprstenastima. Za razliku od funkcije f_1 u ovoj se funkciji kao mjerilo odnosa lokalnih i međuprstenastih zahtjeva koriste njihove vrijednosti, a ne samo broj zahtjeva.

Prekrivanje prema težinama grana i vrednovanje prstenova

Slični kriteriji vrednovanja prstenova koriste se i pri izvedbi prekrivanja mreže prstenovima bez dodatne analize zahtjeva unutar prstenova. Jedina razlika je u tome što zbog nepostojanja informacija o unutarprstenastom usmjeravanju zahtjeva nije moguće odrediti da li se pojedini zahtjev zatvara lokalno unutar prstena ili ne. Za svaki prsten $R \in RES$ prema podacima o prstenu i ulaznim podacima o usmjeravanju zahtjeva na razini mreže mogu se odrediti slijedeći parametri: veličina prstena $|R|$, težine svih grana u prstenu $w(e)$, te maksimalne težine grana $\max w(e)$. Dakle, moguće je odrediti sve karakteristične parametre kao i kod prvog pristupa rješavanju. No, bitno je naznačiti da značenje parametara nije u potpunosti identično – na vrijednosti parametara u izvedbi bez analize unutarprstenastog prometa isključivo utječe inicijalno usmjeravanje prometnih zahtjeva na razini mreže. Nakon provedbe postupka prekrivanja, vrijednosti bi trebale biti više ili manje uskladive, no inicijalno određivanje je potpuno drugačije.

Težinske funkcije korištene pri ovakvom pristupu rješavanja problema su definirane kao:

$$g_1 = \frac{\sum_{\forall e_i \in E_R} w(e_i)}{\max_{e_i \in E_R} w(e_i) \cdot |R|} \quad (3.6)$$

$$g_2 = \max_{e_i \in E_R} w(e_i) \cdot |R| - \sum_{\forall e_i \in E_R} w(e_i) \quad (3.7)$$

$$g_3 = \sum_{\forall e_i \in E_R} w(e_i) \quad (3.8)$$

$$g_4 = \frac{\left(\sum_{\forall e_i \in E_R} w(e_i) \right)^2}{\max_{e_i \in E_R} w(e_i) \cdot |R|} \quad (3.9)$$

Funkcija g_1 je u stvari izraz identičan funkciji f_2' spomenutoj u prethodnom odjeljku. Radi se o izrazu koji predstavlja efikasnost ili iskoristivost prstena – omjer realno korištenih kapaciteta (u normalnom režimu rada) i ukupno instaliranih kapaciteta. Primjena funkcije g_1 najčešće će rezultirati manjim prstenovima sa visokom razinom iskoristivosti.

Težinska funkcija g_2 određuje absolutnu razliku između instaliranih kapaciteta i realno korištenih kapaciteta u prstenu. Direktnom primjenom funkcije g_1 u postupku traženja adekvatnih prstenova dobivat će se kao rezultat podopterećeni, najčešće veliki, prstenovi. Zbog toga se u postupku koristi i invertirana funkcija $g_2'=1/g_2$. Tako definirana funkcija g_2' kao rezultat daje manje i dobro iskoristive prstenove.

Korištenjem funkcije g_3 kao rezultat se dobivaju prstenovi koji prenose velik promet – funkcija predstavlja jednostavnu sumu korištenih kapaciteta grana.

Funkcija g_4 je kombinacija funkcija g_1 i g_3 , te služi za određivanje prstenova koji su dobro iskorišteni, a ujedno prenose veći promet.

Definiranje kriterija vrednovanja mreže

Nakon što se odrede prstenovi koji prekrivaju mrežu (skup RCS) potrebno je na neki način odrediti kvantitativnu mjeru vrijednosti te mreže i po potrebi odlučiti između više ponudenih rješenja. Već je prije spomenuto da usporedbe rješenja baziranih na kriterijima vrednovanja pojedinih prstenova ne dolaze u obzir zbog nenormiranosti težinskih funkcija. I uz normiranje težinskih funkcija za prstenove rezultati ne bi nužno davali dobre odnose među resultantnim mrežama. Ipak, podaci o mreži koji su dostupni nakon provedbe postupka prekrivanja dovoljni su da se odrede karakteristični parametri mreže koji mogu poslužiti za definiranje što realnijeg kriterija vrednovanja cijele mreže.

Vrednovanje mreže ovisno o prstenastom i međuprstenastom prometu

U realnim mrežama cijena bi ovisila o cijenama pojedinih prstenova, tj. cijena mreže P_N bila bi suma cijena prstenova P_R koji prekrivaju mrežu uz dodatak cijene međuveza P_{IC} između prstenova:

$$P_N = \sum_{\forall R \in RCS} P_R + P_{IC}$$

Kao što je izneseno u poglavlju 2.3, realna cijena prstena proporcionalna je kapacitetu prstena. Budući da su prstenovi potpuno određeni u prethodnom koraku postupka (neovisno o pristupu rješavanja problema), kao mjera za cijenu prstena P_R može se uzeti kapacitet prstena. Slično se kao mjera za cijenu međuveza može uzeti kapacitet ostvarenih međuveza u mreži. U postupku za određivanje prekrivanja mreže bez analize unutarprstenastih zahtjeva nije moguće odrediti međuprstenasti promet, pa u tom slučaju vrednovanje mreže na opisani način nije izvedivo, već se mora zanemariti P_{IC} . Prema iznesenom, komponente za određivanje cijene mreže mogu se definirati:

$$P_{R(BSHR)} = \max_{e_i \in E_R} w(e_i) \quad (3.10)$$

$$P_{R(USHR)} = \sum_{\forall d_{ij} \in D_R} d_{ij} \quad (3.11)$$

$$P_{IC} = \sum_{\forall R \in RCS} \left(\sum_{\forall d_{ij} \in D_{Inter}} d_{ij} \right) \cdot \alpha \quad (3.12)$$

$P_{R(BSHR)}$ se kao mjerilo cijene pojedinog prstena definira u slučaju da se radi o dvosmjernim prstenovima, dok se $P_{R(USHR)}$ definira u slučaju kad su prstenovi jednosmjerni (kapacitet prstena je jednak sumi prometnih zahtjeva). Za međuprstenasti promet definira se cijena P_{IC} koja je određena sumom međuprstenastih zahtjeva u svim prstenovima. Parametar α omogućava definiranje dodatne vrijednosti međuprstenastih zahtjeva. Time se model za određivanje cijene približava realnom modelu mreže u kojoj međuprstenasti promet uvjetuje dodatne troškove izgradnje – troškovi su najčešće proporcionalni kapacitetu pomnoženom s nekim faktorom.

Vrednovanje mreže prema iskoristivosti i redundanciji

Često korišten kriterij vrednovanja mreža jest određivanje iskoristivosti i redundancije u mreži. Naime, kad se cijela mreža prekrije prstenovima, za svaku granu u mreži je moguće odrediti težinu odnosno radno opterećenje grane $w(e)$. Isto tako moguće je odrediti i maksimalna opterećenja grana (ekvivalentno kapacitetu dvosmjernog prstena) za svaki pojedini prsten $\max w(e)$. Prema tim podacima jednostavno se može odrediti iskoristivost mreže kao omjer ukupnog radnog opterećenja grana (suma opterećenja svih prekrivenih grana) i ukupno instaliranih kapaciteta u mreži. Tako definirana iskoristivost mreže zapisuje se kao:

$$\mu_N = \frac{\sum_{\forall R \in RCS} \left(\sum_{\forall e_i \in E_R} w(e_i) \right)}{\sum_{\forall R \in RCS} \left(|R| \cdot \max_{e_i \in E_R} w(e_i) \right)}$$

Ovisno o kriteriju koji dizajneri postave, cijena mreže može se iskazati proporcionalna iskoristivosti, dakle $P_N \sim \mu_N$. Ukoliko se analizira redundancija u mreži, cijena se definira obrnuto proporcionalna iskoristivosti, tj. $P_N \sim 1/\mu_N$. Izraze za odredivanje cijene mreže zapisujemo kao:

$$P_{N1} = \mu_N = \frac{\sum_{\forall R \in RCS} \left(\sum_{\forall e_i \in E_R} w(e_i) \right)}{\sum_{\forall R \in RCS} \left(|R| \cdot \max_{e_i \in E_R} w(e_i) \right)} \quad (3.13)$$

$$P_{N2} = \frac{1}{\mu_N} = \frac{\sum_{\forall R \in RCS} \left(|R| \cdot \max_{e_i \in E_R} w(e_i) \right)}{\sum_{\forall R \in RCS} \left(\sum_{\forall e_i \in E_R} w(e_i) \right)} \quad (3.14)$$

Cijena mreže može se iskazati i drugačijim odnosima karakterističnih parametara prstena. Na primjer odnosom ukupnih slobodnih kapaciteta i ukupnog opterećenja u standardnom načinu rada:

$$P_{N3} = \frac{\sum_{\forall R \in RCS} \left(\left(|R| \cdot \max_{e_i \in E_R} w(e_i) \right) - \sum_{\forall e_i \in E_R} w(e_i) \right)}{\sum_{\forall R \in RCS} \left(\sum_{\forall e_i \in E_R} w(e_i) \right)} \quad (3.15)$$

Ovisno o zahtjevima i kriterijima dizajnera odabire se najpovoljnije rješenje mreže. Na odabir najpovoljnijeg rješenja mogu utjecati i drugi alternativni parametri.

Alternativni parametri vrednovanja mreže

Za mrežu dobivenu postupkom prekrivanja prstenovima mogu se definirati još neki dodatni parametri koji mogu pomoći dizajneru u odluci o izboru rješenja. Mogu se definirati:

- prosječna veličina prstena u rješenju,
- broj prstenova u rješenju,
- ukupni broj i kapacitet međuprstenastih zahtjeva,
- ukupni broj i kapacitet lokalnih zahtjeva i sl.

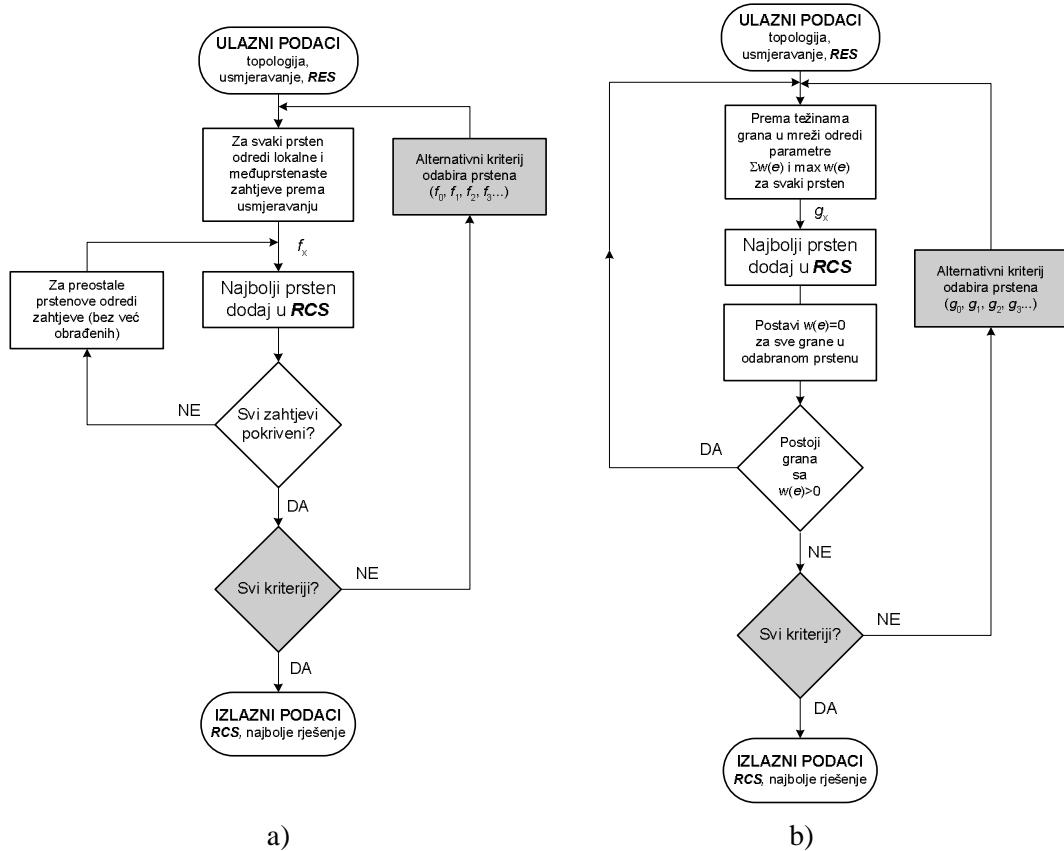
Kombiniranjem i usporedbom tih parametara dizajner dobiva dodatne informacije o mreži, što može utjecati na njegovu odluku o rješenju.

Pohlepna (*greedy*) metoda

Prema literaturi [10,16], mnogi od alata dostupnih na tržištu pri rješavanju RCP problema koriste pohlepne metode koje kao takve baš i nisu popularne u optimizacijskim postupcima. Njihov osnovni nedostatak je da u potpunosti suzuju područje u kojem traže rješenje i gotovo uvijek nalaze lokalni optimum za kojeg nema nikakve garancije da je dovoljno blizu globalnom optimalnom

rješenju. Međutim u rješavanju RCP-a pokazale su se vrlo dobrom izborom, a njihova kratkovidnost uspješno se prevladava korištenjem različitih, prije opisanih, težinskih funkcija. Kako se radi o relativno brzim metodama, nikakav problem ne predstavlja njihovo višestruko izvođenje i usporedba rezultata dobivenih korištenjem raznih težinskih funkcija. Ovisno o principu koji se koristi pri prekrivanju mreže postupci su prikazani blok dijagramima na slici 3.19.

Prikaz izvođenja *greedy* metode uz dodatnu analizu unutarprstenastog i međuprstenastog prometa je dan na slici 3.19 a). Slika 3.19 b) prikazuje izvođenje *greedy* metode kod koje se prekrivanje obavlja prema inicijalno definiranom usmjeravanju zahtjeva na razini mreže.

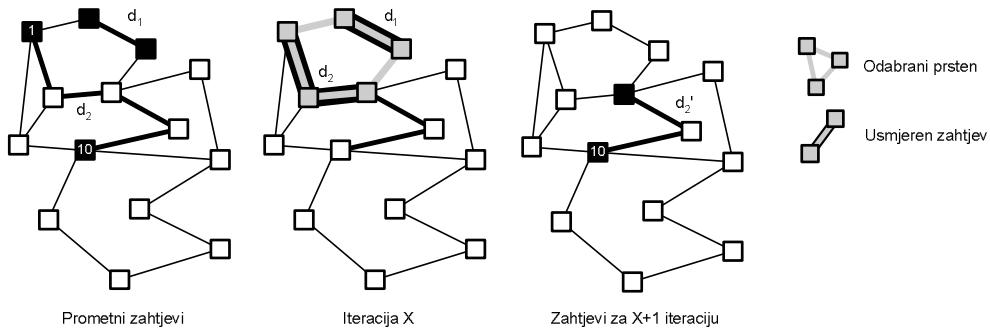


Slika 3.19: Shematski prikaz *greedy* metode za rješavanje RCP uz različite principe određivanja prstenova

Ulagni podaci u postupak za određivanje prekrivanja su topologija, skup izvedivih/mogućih prstenova na zadanoj topologiji, prometni zahtjevi i njihovo usmjeravanje.

Postupak uz analizu unutarprstenastog prometa

Prvi korak postupka je analiza svih zahtjeva i prstenova, te određivanje lokalnih i međuprstenastih zahtjeva za svaki pojedini prsten. Nakon što se odredi usmjeravanje zahtjeva unutar prstena (ovisno o tipu prstena – SNCP ili MS-SPRing), moguće je odrediti opterećenja pojedinih grana i kapacitet prstena. Na taj način dolazi se do vrijednosti svih parametara potrebnih za definiranje težine prstena. Za svaki prsten se, prema definiranom kriteriju (3.2, 3.3, 3.4, 3.5), odredi težina. Sve težine se usporede, odabere najbolji prsten i doda u skup prstenova koji prekrivaju mrežu \mathbf{RCS} .



Slika 3.20: Princip obrade lokalnih i međuprstenastih zahtjeva

Zahtjevi koji su zatvoreni (lokalni) u odabranom prstenu eliminiraju se iz dalnjeg razmatranja. Zahtjevi koji su djelomično usmjereni unutar prstena (izvorište i/ili odredište nisu u prstenu, tj. radi se o međuprstenastim prometnim zahtjevima), se dijele i u daljnji postupak uključuju kao 'parcijalni'. Prikaz lokalnog i međuprstenastog zahtjeva, te postupak daljnje obrade istih prikazan je na slici 3.20. Prvi dio slike (lijevo) prikazuje dva definirana prometna zahtjeva d_1 i d_2 . U x -toj iteraciji postupka odabran je prikazani prsten. U tom prstenu d_1 predstavlja lokalni zahtjev, dok je d_2 međuprstenasti zahtjev. Prije prelaska na sljedeću iteraciju iz dalnjeg se razmatranja eliminira lokalni zahtjev d_1 i dio zahtjeva d_2 koji je usmjerjen odabranim prstenom. Preostali dio zahtjeva d_2 se definira kao novi zahtjev d_2' koji se analizira u nastavku postupka. U dalnjem postupku d_2' može postati lokalni zahtjev nekog novoodabranog čvora. Takav zahtjev se u postupku smatra lokalnim zahtjevom, no ne uključuje se u skup tzv. pravih lokalnih zahtjeva. Svojstvo pravih lokalnih zahtjeva jest da nisu generirani kao dio nekog već obrađenog zahtjeva.

Ukoliko su svi zahtjevi obrađeni postupak se prekida i iznosi se rješenje preklapanja prstenovima. Ako postoje zahtjevi koji još nisu obrađeni postupak se ponavlja (ponovno se odrede lokalni i međuprstenasti za sve preostale prstenove, odrede parametri prstena i proračunaju nove težinske vrijednosti) sve dok svi zahtjevi ne budu obuhvaćeni rješenjima.

Dva su bitna koraka u postupku koji određuju u kojem će smjeru algoritam krenuti i kakve će rezultate dati:

- način odredivanja lokalnih i međuprstenastih zahtjeva
- određivanje težine pojedinog prstena

Različite izvedbe tih koraka uvjetovati će pronalazak različitih rješenja prekrivanja mreže. Variranjem uvjeta ostvaruje se proširenje skupa mogućih rješenja, što omogućuje vrednovanje kompletnih rješenja i donošenje odluke o najboljem rješenju od ponudenih.

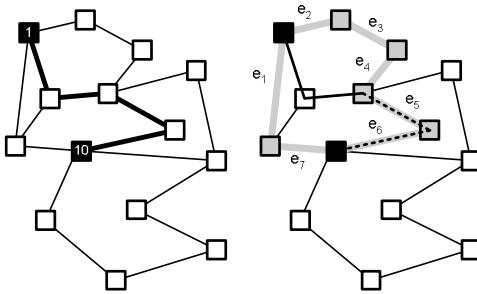
Dodatno proširenje skupa rješenja ostvaruje se korištenjem različitih metoda usmjeravanja – ponavljanjem metode za različite ulazne podatke o usmjeravanju zahtjeva također se dobivaju različita rješenja za prekrivanje mreže.

Proširenje promatranog područja rješenja problema može se ostvariti primjenom drugog početnog izbora u *greedy* algoritmu – umjesto odabira prstena s najboljim težinskim vrijednostima, kao početni izbor uzima se prsten s drugom najboljom težinom. Odluka o boljem rješenju donosi se nakon definiranja cijelog **RCS** skupa. Dio postupka koji realizira različite načine odredivanja težina prstena na blok dijagramu je zasjenjen. Sam proračun težina izvodi se korištenjem prije definiranih težinskih funkcija.

Određivanje lokalnih i međuprstenastih zahtjeva

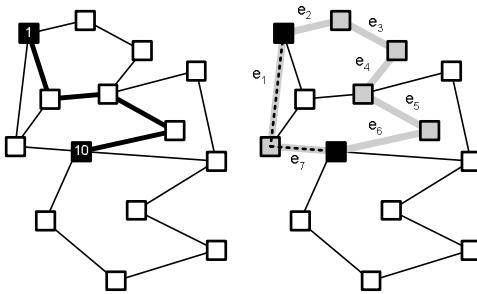
Dva su principa određivanja lokalnih i međuprstenastih zahtjeva.

Prvi princip se u potpunosti bazira na prethodno definiranom usmjeravanju. Ukoliko je usmjeravanje zahtjeva izvedeno tako da su sve grane na putu usmjeravanja sadržane u prstenu zahtjev se definira kao lokalni, a ukoliko se grane djelomično preklapaju sa prstenom zahtjev je međuprstenasti. Primjer određivanja tipa zahtjeva je prikazan na slici 3.21.



Slika 3.21: Primjer određivanja tipa zahtjeva – prvi princip

Drugi princip lokalne zahtjeve definira samo na razini čvorova – ukoliko su izvorišni i odredišni čvor dio prstena zahtjev se definira kao lokalni, neovisno o njegovom usmjeravanju. Međuprstenasti zahtjevi su svi ostali – identično prvom principu grane puta usmjeravanja djelomično se preklapaju sa prstenom. Primjer takvog određivanja lokalnih zahtjeva je dan na slici 3.22.



Slika 3.22: Primjer određivanja tipa zahtjeva – drugi princip

Primjeri usmjeravanja zahtjeva između čvorova 1 i 10 na mreži prikazanoj na slikama ukazuju na različite rezultate pri korištenju spomenutih principa. U prvom slučaju, pri kojem se za određivanje tipa zahtjeva koristi isključivo informacija o predloženom usmjeravanju, zahtjev se definira kao međuprstenasti i prolazi tek granama e_5 i e_6 označenog prstena. U drugom slučaju zahtjev se definira kao lokalni, bez obzira na predloženo usmjeravanje. Jedini kriterij u tom slučaju je prisutnost izvorišnog i odredišnog čvora u označenom prstenu.

U postupak određivanja lokalnih i međuprstenastih zahtjeva moguće je uključiti prepostavke koje utječu na način određivanja – npr. maksimalan broj zahtjeva koje prsten može posluživati, maksimalno opterećenje grane unutar prstena (ekvivalentno ograničavanju maksimalnog mogućeg kapaciteta prstena) i slično.

Može se prepostaviti da će se korištenjem drugog spomenutog principa smanjiti broj međuprstenastih zahtjeva, te vrlo vjerojatno smanjiti broj prstenova potrebnih za realizaciju svih prometnih zahtjeva. S druge strane, takav pristup poprilično umanjuje utjecaj modela usmjeravanja zahtjeva na rezultate prekrivanja mreže prstenovima.

Postupak prekrivanja baziran na inicijalnim usmjeravanjima zahtjeva

Za razliku od prethodno opisanog postupka u kojem se svaki pojedini zahtjev dodatno analizira i usmjerava unutar razmatranih prstenastih struktura, ovaj pristup isključuje bilo kakvu analizu prometnih zahtjeva – ulazni podaci o usmjeravanju zahtjeva na razini mreže su dovoljni za nastavak postupka i traženje podskupa prstenova koji će prekrivati mrežu.

Postupak je shematski prikazan blok-dijagramom na slici 3.19 b).

U prvoj iteraciji određuju se karakteristični parametri za svaki od mogućih prstenova, te se prema tim parametrima odredi težina svakog prstena (jedini podaci koji se mogu dobiti iz ulaznih

podataka o usmjeravanju zahtjeva su težine grana $w(e)$, te na osnovu njih određen kapacitet dvosmjernog prstena $\max w(e)$ – dakle, analiza jednosmjernih prstenova ovim postupkom nije moguća). Prema težini (određenoj izrazima 3.6, 3.7, 3.8, 3.9) se odabire najbolji prsten i dodaje ga se u skup prstenova koji će prekrivati mrežu. Kako bi se spriječilo višestruko uključivanje opterećenja grana, svima grana uključenim u resultantni prsten težine se postavljaju na 0. Ukoliko u mreži postoje grane s težinama većim od 0, postupak se ponavlja sve dok sve opterećene grane nisu uključene u rješenje.

Ovakav postupak mnogo je jednostavniji od postupka koji uključuje analizu prometnih zahtjeva. No, jednostavnost postupka utječe i na nižu informativnost rješenja. Budući da je upravo detaljno izvješće o usmjeravanju svakog prometnog zahtjeva u mreži od vrlo velike važnosti za dizajnere mreže, može se zaključiti da je praktična vrijednost ovog pristupa rješavanju problema manja.

Ipak, dodatnom analizom rezultata i njihovom sintezom sa ulaznim podacima o usmjeravanju zahtjeva moguće je odrediti detalje o usmjeravanju zahtjeva i time omogućiti korištenje ovog pristupa u praksi. Mogući problem, prije svega zbog izvedbe u realnim mrežama, predstavlja veća vjerojatnost da su prometni zahtjevi prenošeni većim brojem prstenova u odnosu na postupak koji uključuje analizu zahtjeva unutar prstenova.

3.2.6. Matematički model prekrivanja mreže prstenovima

Dosadašnje izlaganje vezano uz prekrivanje mreže prstenovima isključivo se temeljilo na objašnjenjima principa izvođenja različitih metoda koje se u tu svrhu koriste. U nastavku se definira matematički model koji se može iskoristiti za određivanje optimalnog prekrivanja korištenjem matematičkog programa. U literaturi se mogu naći različiti, manje ili više složeni, matematički modeli [15,17].

Najjednostavniji model predlaže [15]. Prema tom modelu optimizacija se obavlja na skupu mogućih prstenova \mathbf{RES} i to tako da se svakom prstenu jednokratno dodijeli težina, te se potom prelazi na optimacijski postupak kojim se određuje podskup prstenova s najnižom ili najvišom težinom, već ovisno o definiranim kriterijima. Jedini uvjet koji se postavlja je da sve grane u mreži budu prekrivene. Za svaki prsten $R \in \mathbf{RES}$ definira se težina $C(R)$. Definira se vektor redak \mathbf{C} koji sadrži težine svih prstenova, $\mathbf{C} = [C(R_1) \ C(R_2) \dots \ C(R_{|\mathbf{RES}|})]$. Grane koje su sadržane u pojedinom prstenu definiraju se matricom \mathbf{A} s elementima a_{ij} ($i = 1..|\mathbf{RES}|, j = 1..|E|$) gdje a_{ij} dobiva vrijednost 1 ukoliko je grana e_j sadržana u prstenu R_i iz \mathbf{RES} . U suprotnom a_{ij} poprima vrijednost 0. Problem se svodi na određivanje varijabli x za svaki prsten R , koje označavaju da li je prsten odabran u skup \mathbf{RCS} . Za potrebe modela definira se vektor stupac \mathbf{x} koji sadrži varijable x za svaki pojedini prsten, $\mathbf{x} = [x(R_1) \ x(R_2) \ \dots \ x(R_{|\mathbf{RES}|})]$. Težina mreže se definira matrično kao umnožak vektora \mathbf{C} i \mathbf{x} . Funkcija cilja kod tako definiranog RCP se definira kao:

$$\begin{aligned} & \text{RCPEdges:} \\ & \min \mathbf{C} \cdot \mathbf{x} \end{aligned} \tag{3.16}$$

Uvjet je da odabrani prstenovi uključuju sve grane u mreži, što se može zapisati matrično kao:

$$\mathbf{A} \cdot \mathbf{x} \geq \mathbf{I} \tag{3.17}$$

gdje \mathbf{I} predstavlja vektor stupac koji sadrži jedinice.

Vrijednosti koje se uključuju u matricu \mathbf{A} moguće je jednostavno odrediti na modelu za određivanje prstenova koji se koristi u ovom radu. 'Binarni' prikaz prstenova iz tablice 3.3 jednostavno se, bez dodatne obrade, može preslikati u matricu.

Da bi se model mogao opisati AMPL-om matrični zapis se prilagođava. Funkcija cilja definira se kao suma parametara:

RCPEdges:

$$\min \sum_{i=1}^{|RES|} C(R_i) \cdot x(R_i) \quad (3.16a)$$

Uvjet se može zapisati kao $|E|$ nejednadžbi:

$$\sum_{j=1}^{|RES|} a_{ij} \cdot x(R_j) \geq 1 \text{ za } i=\{1,2, \dots, |E|\} \quad (3.17a)$$

Varijacijama u definiciji matrice A moguće je realizirati različite načine preklapanja mreže – ukoliko se u A umjesto informacija o granama sadržanim u prstenovima definiraju informacije o čvorovima sadržanim u prstenu problem će se svesti na određivanje prstenova koji će prekrivati čvorove (na taj način može se, kao što je u prethodnom odjeljku opisano, smanjiti broj prstenova potrebnih za prekrivanje zbog mogućnosti zanemarivanja određenih grana).

Izraz (3.17a) definira sumu koja pokazuje da li je i -ta grana uključena u rješenje. Ukoliko je i -ta grana uključena u prsten R_j , tada je vrijednost $a_{ij}=1$. Dovoljno je da postoji jedan prsten koji je odabran u rješenje, a sadrži i -tu granu, pa da uvjet bude zadovoljen.

U nastavku je dan model opisan u AMPL-u.

```

1: set RINGS;
2: set EDGES;
3: param A {RINGS,EDGES};
4: param C {RINGS};
5: data "rcp1.dat";
6: var X {RINGS} <=1, >=0 integer;
7: minimize RCP: sum {i in RINGS} C[i]*X[i];
8: subject to Obj {i in EDGES}:
9:     sum{j in RINGS}A[j,i]*X[j]>=1;
```

Funkcija cilja definirana je prema (3.16a) u retku 7, dok su uvjetne nejednadžbe definirane u recima 8 i 9, prema (3.17a). Podaci se definiraju u podatkovnoj datoteci. Primjer za jednostavnu mrežu sa tri prstena i sedam grana je dan u nastavku:

```

set RINGS := R1 R2 R3;
set EDGES := e1 e2 e3 e4 e5 e6 e7;

param A:      e1      e2      e3      e4      e5      e6      e7 := 
R1    1        1        1        1        1        0
R2    0        1        1        0        0        1
R3    1        0        0        0        1        1        1;

param C:=
    R1      100
    R2      90
    R3      87;
```

Specifičnost (ujedno i nedostatak) ovakvog modela je što se na temelju inicijalnih težina prstenova određuje konačno rješenje. Dakle, nema nikakve prilagodbe težina ovisno o odabranim prstenovima i eventualnim promjenama koje se događaju vezano uz usmjeravanje zahtjeva kroz prstenove. Ipak, ukoliko se težinski kriterij pogodno definira, mogu se dobiti zanimljivi rezultati prekrivanja mreže. Druga specifičnost modela je uvjet o prekrivanju grana u mreži. Taj princip može se dobro iskoristiti u nadogradnji postojećih mreža (definirana topologija onakva kakva je

već realizirana), dok je kod izgradnje novih mreža po tom pitanju potreban dodatan angažman dizajnera mreže.

Alternativni modeli mogu zahtijevati kao uvjet samo prekrivenost čvorova – to je dovoljan uvjet za uspostavu svih prometnih zahtjeva koji u određenim slučajevima može smanjiti troškove izgradnje mreže jer eliminira nekorištene grane. Prikazan model jednostavno se prilagodava takvim zahtjevima.

3.3. Alternativni pristupi rješavanju problema prekrivanja mreže prstenovima

Predložen, u ovom radu analiziran i kao praktični dio rada implementiran, pristup rješavanju problema prekrivanja mreže prstenovima bazira se na iskorištanju ulaznih podataka o inicijalnom usmjeravanju zahtjeva u mreži. Opravdanost tog pristupa garantirana je, uvijek bitnim, utjecajem predviđenog prometa na topologiju mreže. Međutim, isto tako se mogu razviti i alternativni pristupi koji se u prvom redu svode na određivanje podskupa prstenova koji zadovoljavaju određene uvjete (neovisne o samim prometnim zahtjevima), a nakon toga usmjeravanje prometnih zahtjeva kroz definirane prstenove. Ovisno o uvjetima koji se postavljaju, problem prekrivanja mreže prstenastim strukturama može biti vrlo složen, te bi mogao biti predmetom istraživanja nekog drugog rada. Najjednostavniji uvjeti koji se mogu postaviti bili bi uvjeti o nužnosti prekrivanja svih grana ili svih čvorova u mreži. Alternativno bi se na rezultate moglo utjecati ograničavanjem mogućnosti preklapanja prstenova – npr. zabranom mogućnosti da dva prstena imaju zajedničke više od jedne grane i slično. Mogao bi se i definirati uvjet o povezivanju prstenova samo jednim zajedničkim čvorom, dakle bez zajedničkih grana itd.

U literaturi se mogu naći detaljniji pregledi različitih pristupa. Zaključak na koji se najčešće nailazi jest da je pristup koji se inicijalno orientira na prometne zahtjeve najadekvatnije i najčešće primjenjivano rješenje. Iako su i ostale metode primjenjivane u određenim slučajevima, pokazalo se da je njihova primjenjivost ograničena na specijalne slučajeve mreže.

U ovom radu takvi alternativni pristupi rješavanju problema nisu razmatrani.

3.4. Posebni modeli prstenastih mreža

Bitno drugačiji pristupi rješavanju problema planiranja prstenastih mreža primjenjuju se u posebnim modelima prstenastih mreža. U takvim se modelima pojedine faze prisutne u standardnom postupku planiranja mogu izbjegći. Specijalni modeli su često korišteni u praksi zbog specifičnosti postupaka planiranja korištenih od strane dizajnera – kao što je i prije spomenuto, dizajneri moraju imati utjecaj na postupak planiranja, što standardni postupci često mogu zanemariti. Dva posebna modela prstenastih mreža koji su razmatrani ovim radom su:

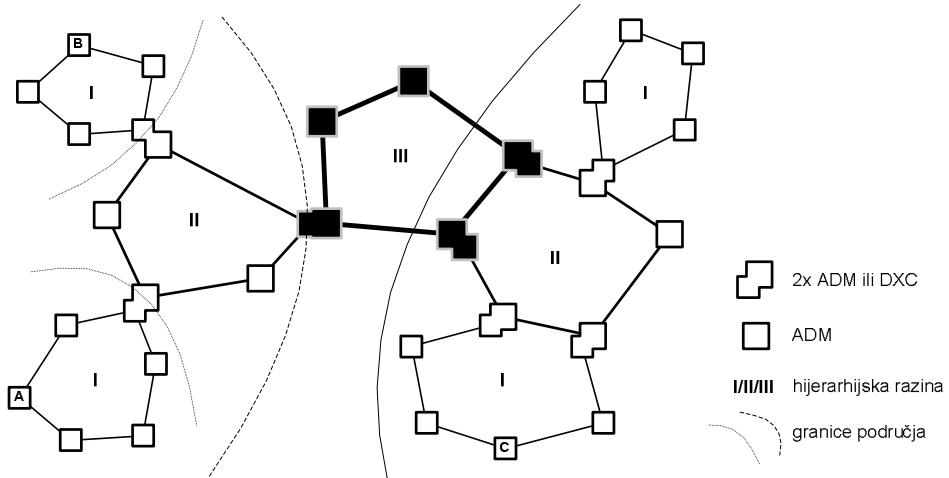
- hijerarhijske prstenaste mreže, te
- koncentrirane prstenaste mreže.

Oba modela u potpunosti eliminiraju probleme određivanja prstenova i odabira prstenova za prekrivanje mreže. Već samo zadavanje topologije takvih mreža definira sve potrebne prstenove, te se problemi planiranja svode na određivanje usmjeravanja prometnih zahtjeva kroz definirane strukture. Izvedbe mreža bazirane na oba spomenuta modela se često javljaju u praksi. U nastavku su ukratko opisana oba modela, te su navedene mogućnosti njihove primjene u praksi, kao i specifičnosti u postupku planiranja takvih mreža.

3.4.1. Hijerarhijske prstenaste mreže

Hijerarhijske prstenaste mreže uskladive su sa standardnim pristupom izgradnje telekomunikacijskih mreža, u kojem se mreža dijeli na nekoliko dijelova – npr. podjela na pristupni (*access*) i temeljni (*backbone*) dio mreže. U takvim mrežama definira se hijerarhija među prstenovima, te je komunikacija u mreži moguća samo između prstenova susjednih hijerarhijskih razina. Iako je takvu organizaciju moguće izvesti i na logičkoj razini, u praksi se najčešće i na fizičkoj razini mreže izgrađuju vodeći računa o hijerarhiji. Slika 3.23 prikazuje primjer

hijerarhijske prstenaste mreže sa tri razine – rubni prstenovi niže razine predstavljaju pristupni dio mreže, prstenovi više razine koriste se za prikupljanje prometa iz pristupnih prstenova, dok temeljni dio mreže predstavljaju prsteni najviše razine. Takav model mreže vrlo je čest u praksi, s tim da se temeljni dio mreže (najviša razina) često realizira i kao *mesh* mreža.



Slika 3.23: Hijerarhijska prstenasta mreža

Komunikacija između čvorova hijerarhijske mreže realizira se prema hijerarhiji unutar definiranih područja. Na slici su naznačene granice hijerarhijskih područja najniže razine koja se spajaju na isti prsten više razine. Ukoliko postoji potreba za komunikacijom između čvorova koja pripadaju različitim područjima ona se realizira kroz prstenove više razine. Na primjeru sa slike, komunikacija između čvorova A i B ostvaruje se prolaskom zahtjeva kroz prsten II. hijerarhijske razine. Komunikacija između čvorova A ili B sa čvorom C uspostavlja se i kroz prsten najviše, III. hijerarhijske razine.

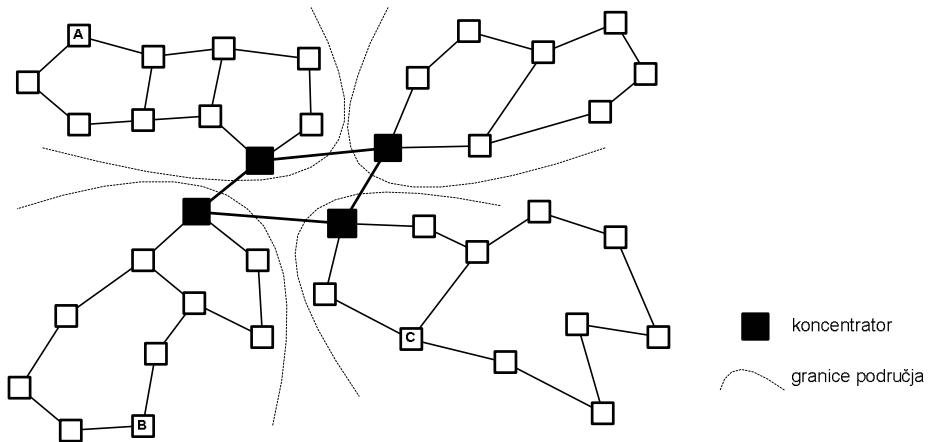
Specifičnosti u postupku planiranja

Zbog karakterističnog načina funkcioniranja, postupak planiranja hijerarhijskih prstenastih mrež relativno je jednostavan. Nakon što se definiraju prstenovi postupak usmjeravanja i dimenzioniranja prstenova izvodi se prema jednostavnim pravilima.

U svakom prstenu definirani su čvorovi koji se koriste za povezivanje sa prstenima više ili niže hijerarhijske razine. Ukoliko se utvrdi da su izvorišni i odredišni čvor elementi istog prstena zahtjev se usmjerava lokalno tim prstenom. U slučaju da je odredišni čvor u drugom prstenu, analizira se odredišni prsten i određuju prstenovi kroz koje se zahtjev mora usmjeriti da bi dospio u odredišni prsten. Na taj način usmjeravanje zahtjeva na razini prstenova je jednoznačno određeno, te preostaje optimizacija usmjeravanja zahtjeva s ciljem smanjenja kapaciteta i troškova izgradnje takvih mreža.

3.4.2. Koncentrirane prstenaste mreže

Model koncentriranih prstenastih mreža je konceptualno sličan onom hijerarhijskim prstenastim mrežama. Najveća razlika između modela postoji u samoj primjeni, odnosno prometnim karakteristikama mreža u kojima se koriste. Model koncentriranih mreža koristi se u mrežama kod kojih se promet može koncentrirati prije prijenosa u drugi dio mreže. Jednostavan primjer koncentrirane prstenaste mreže prikazan je na slici 3.24.



Slika 3.24: Koncentrirana prstenasta mreža

Podjela koncentrirane mreže na područja drugačija je nego li kod hijerarhijskih mreža. Osnovna pretpostavka je da nema druge komunikacije u pojedinom području osim komunikacije između čvorova i koncentratorskog čvora 'zaduženog' za to područje. Kao što je na slici vidljivo područja mogu biti sastavljena od više prstenova – realizacija usmjeravanja do koncentratorskog čvora ovisi o korištenoj tehnologiji i vrsti prstenova, te o tome najviše ovisi dimenzioniranje mreže.

Specifičnosti u postupku planiranja

Zbog karakteristike da čvorovi određenog područja komuniciraju samo sa koncentratorskim čvorom alat za planiranje može jednostavno odrediti same koncentratorske čvorove, kao i granice područja u mreži. Usmjeravanje zahtjeva od izvođačnih i odredišnih čvorova do koncentratora njihovih područja ovisi o korištenim modelima prstenova. Neovisno o tome da li se prstenovi formiraju prema fizičkoj topologiji ili ne, dolazit će do povećanja zahtijevanih kapaciteta u granama prstenova idući od 'periferije' područja prema koncentratoru.

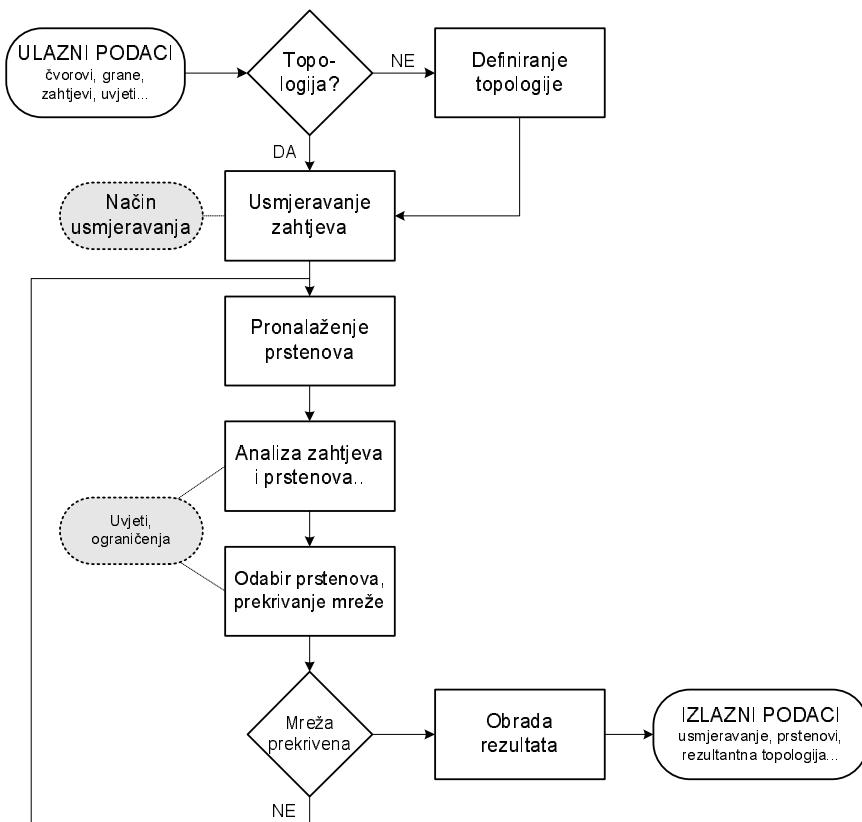
Dimenzioniranje dijela mreže koji povezuje koncentratorske čvorove ovisi o topologiji tog dijela mreže – ukoliko se radi o prstenastoj mreži primjenjuju se postupci za optimizaciju kapaciteta prstenova opisani u odjeljku 2.3. Ukoliko pak se radi o višestruko povezanoj mreži u kojoj se ne koriste prstenovi usmjeravanje i dimenzioniranje se realizira korištenjem raznih algoritama za usmjeravanje u mreži (postoji mogućnost primjene algoritama iz odjeljka 3.2.4).

Spomenuti modeli specijalnih mreža odlikuju se jednostavnijim metodama planiranja, te je to jedan od razloga njihove česte primjene u praksi. Međutim, nedostaci koji se u takvim mrežama mogu uočiti (npr. nepotrebno povećanje kapaciteta grana i prstenova najbližih koncentratorskim čvorovima u koncentriranim optičkim mrežama) mogu se u toku eksploatacije mreža pokazati vrlo problematičnim. Zbog toga se nastoji razviti što kvalitetnije alate koji omogućavaju planiranje standardnih modela prstenastih mreža.

3.5. Organizacija postupka planiranja prstenastih mreža

U prethodnim odjeljcima iznesen je detaljan pregled različitih metoda koje se koriste u postupcima planiranja prstenastih mreža. Metode su najčešće smještane u kontekst zajedno s ostalim metodama kako bi se naznačile mogućnosti njihova korištenja u sklopu postupka planiranja mreže. Ovim poglavljem daje se pregled najčešće primjenjivanog postupka planiranja prstenastih mreža. Svaka od opisanih metoda predstavlja jednu od faza u okviru postupka.

Postupak planiranja podijeljen na nekoliko faza je shematski prikazan dijagramom toka - slika 3.1.



Slika 3.1: Shematski prikaz postupka planiranja prstenastih mreža

Prije početka postupka, analiziraju se ulazni podaci i po potrebi definira kompletna topologija mreže (ukoliko nije unaprijed definirana). Prije analize topologije i određivanja struktura koje zadovoljavaju postavljene zahtjeve, provodi se usmjeravanje definiranih prometnih zahtjeva. Usmjeravanjem zahtjeva određuje se 'važnost' pojedinih grana u mreži. Pronalaženje svih prstenova u mreži (enumeracija prstenova) i određivanje prstenova koji prekrivaju mrežu su operacije karakteristične za postupak planiranja prstenastih mreža. Ukoliko se radi o velikim mrežama složenih topologija, definiraju se ograničenja koja omogućuju određivanje prstenova u realnom vremenu. Nakon što su prstenovi pronađeni, započinje procedura selekcije prstenova koji zadovoljavaju postavljene uvjete i koji će se iskoristiti za prekrivanje mreže. Postupak se svodi na ponavljanje procedure i izbor prstenova najboljih karakteristika sve dok se mreža ne prekrije. Budući da postupak planiranja ovisi o nizu parametara i uvjeta, najčešće ga se ponavlja s različitim postavkama tih parametara. Na taj način se dobiva više rezultantnih mreža, te se dodatnom analizom između njih odabire najbolje ponuđeno rješenje. Dobiveno rješenje uključuje specifikaciju prstenova koji prekrivaju mrežu, podatke o usmjeravanju zahtjeva itd. Odluka o izboru rješenja provodi se analizom karakterističnih vrijednosti i rezultata težinskih funkcija (3.10-3.15).

Opisani postupak planiranja je razrađen i implementiran u praktičnom dijelu rada.

4. Razvijeni alati za planiranje prstenastih mreža

Različiti problemi koji se javljaju u postupku planiranja prstenastih telekomunikacijskih mreža opisani ovim radom najčešće se ne mogu ručno rješavati. Za iole ozbiljniju analizu i izradu kalkulacija neophodna je pomoć računala i specijaliziranih programske alata. Korištenje takvih alata može olakšati i ubrzati postupak. Kao praktični dio ovog rada razvijani su alati u kojima se na jednom mjestu pokušalo integrirati metode za rješavanje što većeg broja spomenutih problema. Rezultat je nekoliko aplikacija – *mRingSolver*, *mSDHRingSolver* i *mRingDesigner*. U ovom poglavlju opisane su funkcije alata, te su dani primjeri njihove primjene.

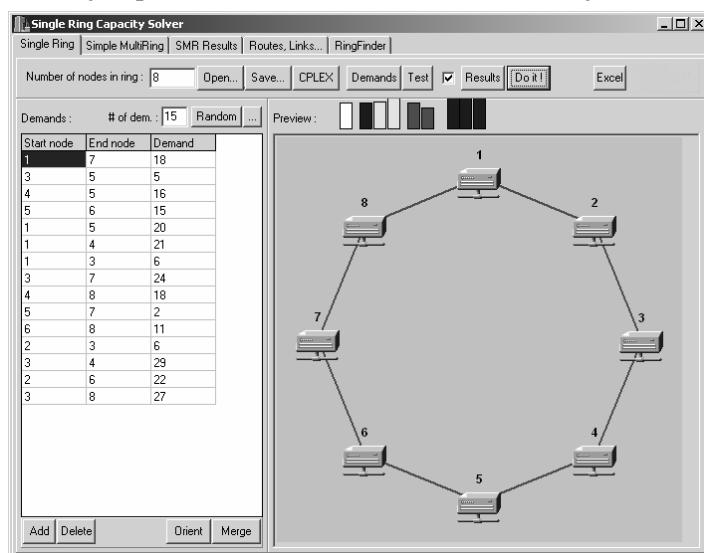
4.1. Alat za dimenzioniranje prstenova - *mRingSolver*

mRingSolver je aplikacija za analizu različitih algoritama za dimenzioniranje jednostrukih prstenova. Jednostavnim sučeljem omogućava definiranje veličine prstena i prometnih zahtjeva koji se moraju optimalno usmjeriti prstenom. Rezultati koji se dobivaju primjenom implementiranih algoritama prikazuju se tablično i grafički kako bi se olakšala njihova usporedba. Algoritmi implementirani u alatu pobrojani su u tablici 4.1.

Tablica 4.1: Implementirani algoritmi

RLPw/oS	RLPwS
<i>OneGreedy</i>	EXACT
<i>TwoGreedy</i>	INDES
kombinacije <i>greedy</i> postupaka	<i>EqualSplit</i>
<i>Dual Ascent</i>	
2-OPT	
Enumeracija	

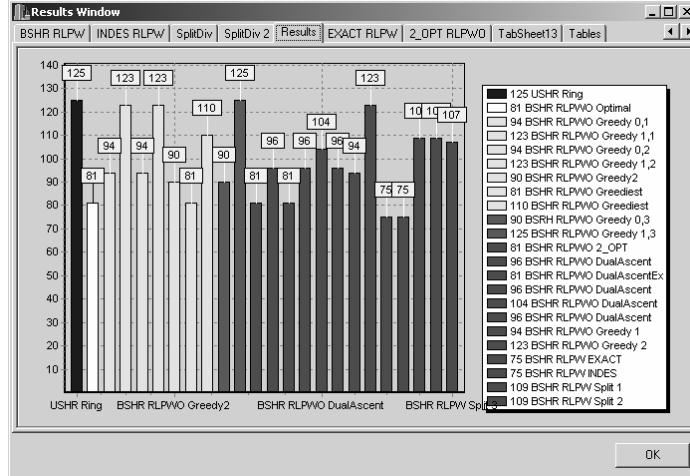
Uz navedene algoritme implementirana je i enumeracijska metoda koja se može primijeniti za dobivanje optimalnog rezultata u prstenovima bez dozvoljenog dijeljenja prometnih zahtjeva. Metoda je efikasna na manjim prstenovima, za modele sa 20-tak zahtjeva.



Slika 4.1: *mRingSolver* – osnovno sučelje

Svi algoritmi prilagođeni su za izvođenje nad jednostavnim modelom prstena, uz proizvoljno definiranje prometnih zahtjeva. Prometni zahtjevi se definiraju cjelobrojno i vrijednost koja im se

pridružuje označava broj prometnih jedinica. Primjeri osnovnog sučelja i usporednog prikaza rezultata dobivenih različitim algoritmima dani su na slikama 4.1 i 4.2.



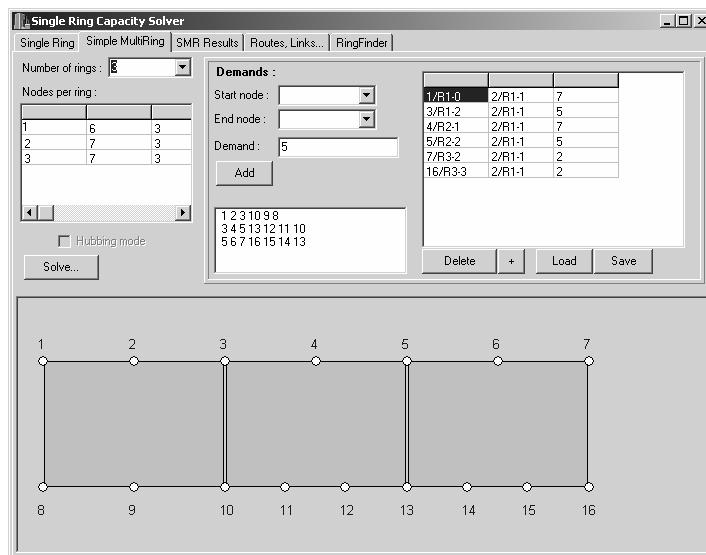
Slika 4.2: *mRingSolver* – usporedni prikaz rezultata

Rezultati dimenzioniranja prstenova izneseni u poglavlju 2 dobiveni su korištenjem ovog alata. Osim određivanja usmjeravanja prometnih zahtjeva unutar prstenova, alata omogućava pripremu podataka za korištenje u AMPL modelu matematičkog programa za rješavanje RDP-a.

4.1.1. Specijalizirani model koncentriranih prstenastih mreža

mRingSolver uključuje i dodatne funkcije koje omogućavaju rješavanje usmjeravanja zahtjeva u posebnom modelu prstenastih mreža, koncentriranim prstenastim mrežama (naziv korišten u literaturi je *hubbed ring network*). Takve mreže su obično izgrađene od više prstenova takvih da se promet koncentriра prema jednom čvoru. Usmjeravanje u takvim prstenovima se svodi na pronaalaženje optimalnog usmjeravanja prometnih zahtjeva do koncentratorskog tzv. *hub* čvora. Cilj je minimizirati potrebne kapacitete grana.

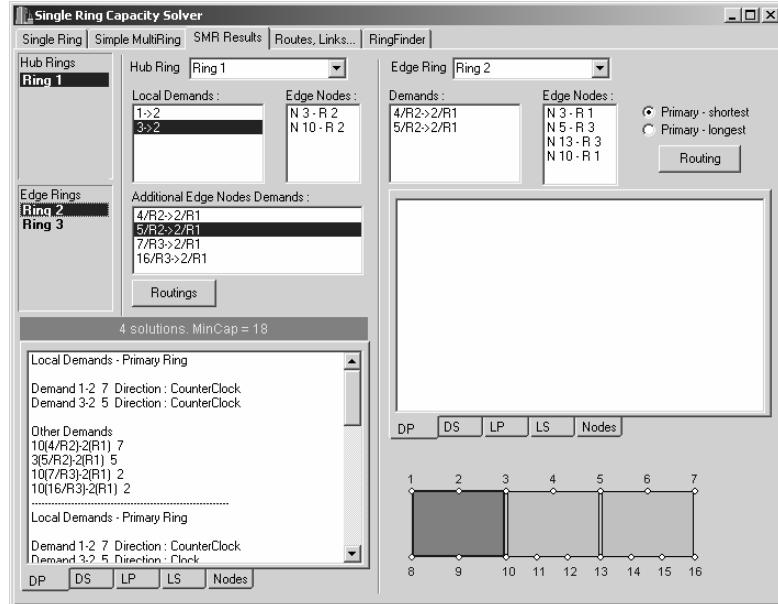
Alat omogućava definiranje podmreža sastavljenih od više prstenova. Pretpostavka je da unutar takve podmreže sav promet ide prema jednom ili više *hub*-ova. Sučelje za definiranje grupe prstenova prikazano je na slici 4.3.



Slika 4.3: *mRingSolver* – definiranje višeprstenaste podmreže

Omogućeno je definiranje mreže prema generičkoj strukturi – definira se broj prstenova i broj čvorova po prstenovima. Pretpostavlja se povezivanje prstenova u dva zajednička čvora. Zahtjevi se definiraju između proizvoljnih čvorova u mreži. Nakon što se na taj način definira mreža i odrede prometni zahtjevi pristupa se određivanju optimalnog usmjeravanja zahtjeva.

Slika 4.4 prikazuje sučelje za provođenje postupka optimalnog usmjeravanja i analizu rezultata.



Slika 4.4: *mRingSolver* – analiza rezultata usmjeravanja

Alat analizira definirane prometne zahtjeve i prema njima određuje koji od prstenova u mreži prikupljaju promet (*hub rings*), a koji ga šalju (*edge rings*). Za svaki od prstenova provodi se postupak optimalnog usmjeravanja, a rezultati se detaljno prikazuju – osim usmjeravanja svakog pojedinog zahtjeva daje se i pregled opterećenja grana, te ulazno-izlazne karakteristike čvorova.

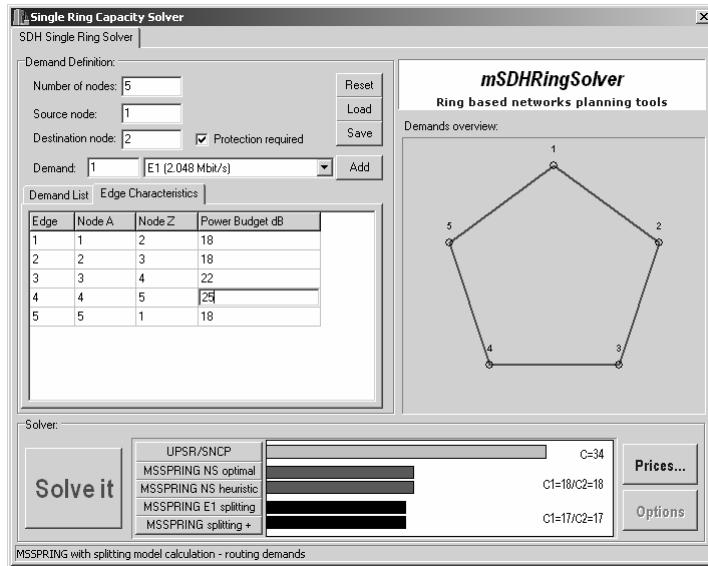
4.2. Specijalizirani alat za dimenzioniranje SDH prstenova - *mSDHRingSolver*

Osim alata *mRingSolver* koji se koristi za analizu standardnog modela prstena i algoritama za usmjeravanje zahtjeva i dimenzioniranje prstenova, izvedena je i specijalizirana inačica alata nazvana *mSDHRingSolver*. *mSDHRingSolver* koristi se za analizu i dimenzioniranje SDH izvedbe jednostavnog prstena. Uz pretpostavku korištenja ADM čvorova i definiranje zahtjeva u prometnim razinama definiranim SDH standardom (podržane SDH prometne razine dane su u tablici 4.2), specijalizirana inačica osim rješavanja problema dimenzioniranja nudi i rezultate u vidu specifikacije ADM čvorova i kompletног SDH prstena.

Tablica 4.2: *mSDHRingSolver* – podržane prometne razine

Signal	Broj E1 jedinica	Kapacitet
E1	1 E1	2.048 Mbit/s
E2	4 E1	8.448 Mbit/s
E3	21 E1	34.368 Mbit/s
E4	63 E1	139.264 Mbit/s
STM-0	21 E1	51.840 Mbit/s
STM-1	63 E1	155.520 Mbit/s
STM-4	252 E1	622.080 Mbit/s
STM-16	1008 E1	2488.320 Mbit/s
STM-64	4032 E1	9953.280 Mbit/s

Kao ulazni podatak se, osim veličine prstena i prometnih zahtjeva, definiraju i dozvoljeni gubici zbog gušenja signala na multipleksnim dionicama. Primjer definiranja karakteristika multipleksnih dionica je dan na slici 4.5.



Slika 4.5: Definiranje karakteristika multipleksnih dionica - *mSDHRingSolver*

Nakon provedenih proračuna, SDH prsten karakterizira se:

- kapacitetom prstena u E1 prometnim jedinicama
- realnim kapacitetom (linijsko sučelje prema SDH standardu) ovisno o načinu izvedbe (SNCP, MS-SPRING 2f/4f)
- karakteristikama grana u prstenu (odabir najpovoljnijeg modela izvedbe optičkog vlakna, a uz zadovoljavanje uvjeta o dozvoljenim gubicima na multipleksnim dionicama)

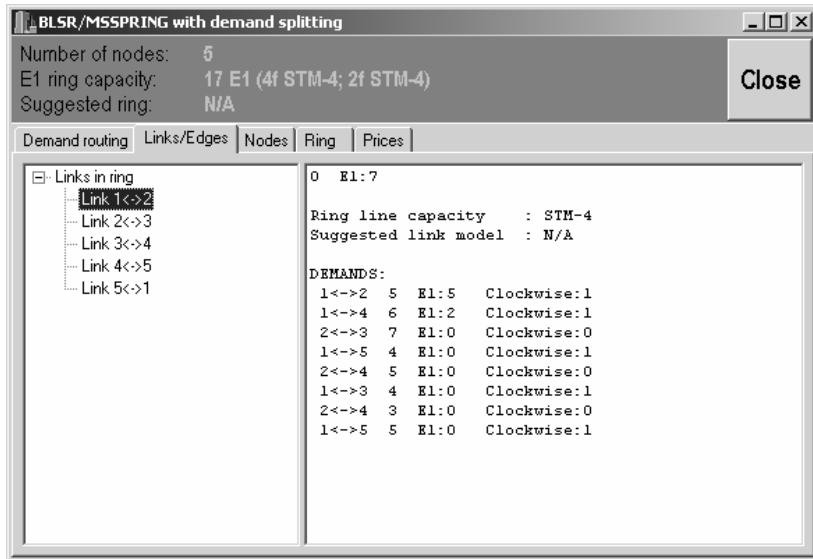
Svaki multiplekser u prstenu karakterizira se:

- svojstvima tributarijskih priključaka
- tipom multipleksera (odabir najpovoljnijeg ovisno o zahtijevanom linijskom sučelju)
- vrstama tributarijskih priključaka (ovisno o tipu multipleksera i zahtijevanim svojstvima priključaka)

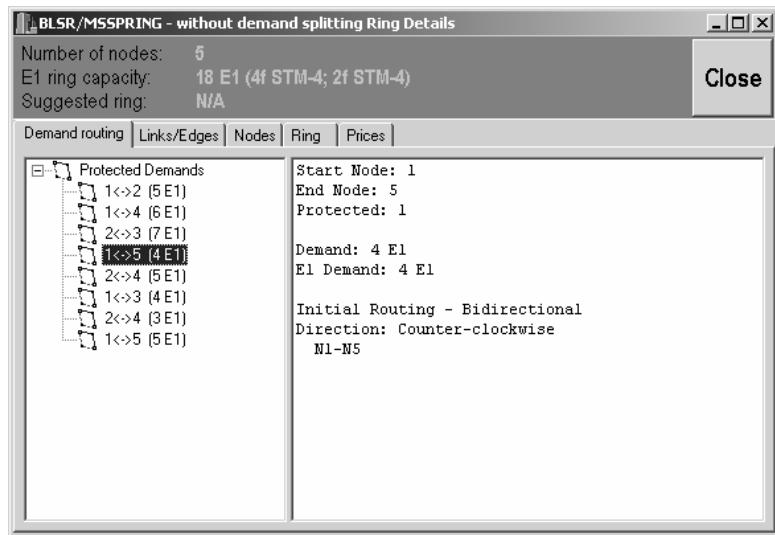
Osim razlike u primjenjenom modelu i složenosti izlaznih rezultata, *mSDHRingSolver* implementira dvije dodatne mogućnosti koje *mRingSolver* ne podržava:

- mogućnost analize i usmjeravanja nezaštićenih prometnih zahtjeva u SNCP prstenovima
- dijeljenje prometnih zahtjeva u MS-SPRING prstenovima prema definiranim SDH razinama

Proračuni kapaciteta i usmjeravanja zahtjeva se obavljaju istovremeno za različite tipove prstenova – SNCP, te MS-SPRING bez dijeljenja i sa dijeljenjem prometnih zahtjeva (dijeljenje na E1 razini i dijeljenje na ponuđenim razinama uz ograničenja). Po završetku postupaka dobiva se pregled rezultata usmjeravanja i detaljnih karakteristika mrežnih elemenata za tip prstena po želji. Primjer prikaza rezultata dan je na slikama 4.6 i 4.7.



Slika 4.6: *mSDHRingSolver* – prikaz rezultata – karakteristike dionica



Slika 4.7: *mSDHRingSolver* – prikaz rezultata – usmjeravanje zahtjeva

4.2.1. Usmjeravanje nezaštićenih zahtjeva

U praksi se pri korištenju jednosmjernih zaštitnih prstenova (SNCP prstenovi u SDH tehnologiji) mogu iskoristiti dodatni kapaciteti namijenjeni zaštiti prometa u slučaju kvara za prijenos niskoprioritetnog, nezaštićenog prometa. *mSDHRingSolver* omogućava definiranje nezaštićenih zahtjeva koji se u SNCP prstenovima usmjeravaju tako da najmanje utječu na povećanje proračunatog kapaciteta prstena. U normalnom režimu rada prstenom se uz zaštićene zahtjeve prenose i niskoprioritetni zahtjevi. U slučaju kvara, obavlja se preusmjeravanje zaštićenih zahtjeva, a niskoprioritetni zahtjevi mogu biti izgubljeni. Iako bi se prijenos nezaštićenih zahtjeva mogao koristiti i u MS-SPRing prstenovima, osnovna prednost korištenja u SNCP je ta da, zbog zaštite na razini puta, pri preusmjeravanju ugroženih zahtjeva postoji mogućnost da pojedini niskoprioritetni zahtjevi i dalje prometuju. Kod MS-SPRing to nije moguće jer se zaštita realizira na razini multipleksne dionice, te preusmjeravanje utječe na cijeli prsten što rezultira gubitkom svih nezaštićenih zahtjeva. Postupak usmjeravanja nezaštićenih zahtjeva s ciljem minimizacije povećanja kapaciteta prstena je ekvivalentan određivanju optimalnog usmjeravanja prometnih zahtjeva bez mogućnosti dijeljenja zahtjeva. U *mSDHRingSolver*-u se rješava ili enumeracijskom

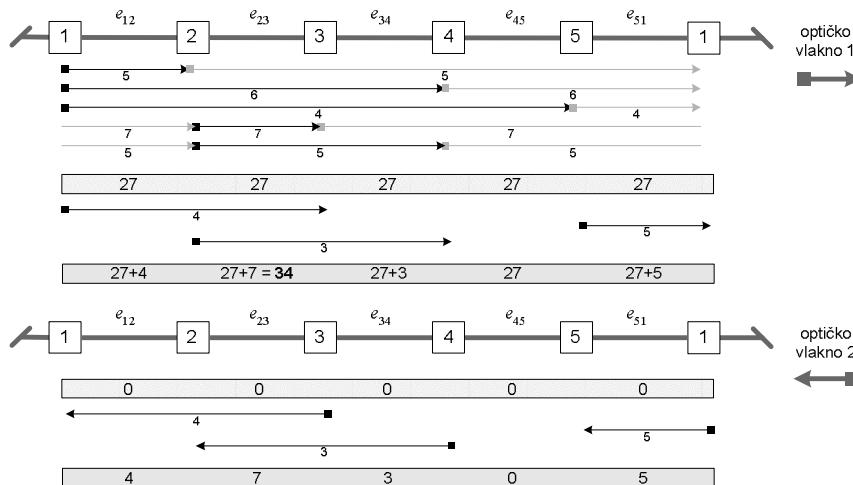
metodom (za manji broj zahtjeva) ili 2-OPT metodom koja pokazuje najbolje rezultate od svih heurističkih algoritama.

Neka je u SNCP prstenu izgrađenom od 5 čvorova definirano 5 zaštićenih i 3 nezaštićena prometna zahtjeva prema tablici. Kad bi svi zahtjevi bili zaštićeni, kapacitet prstena bi bio jednak sumi zahtjeva, i u ovom primjeru bi iznosio 39. Međutim usmjeravanje nezaštićenih zahtjeva moguće je izvesti kombiniranim korištenjem oba vlakna u prstenu.

Tablica 4.3: Definirani zahtjevi u primjeru

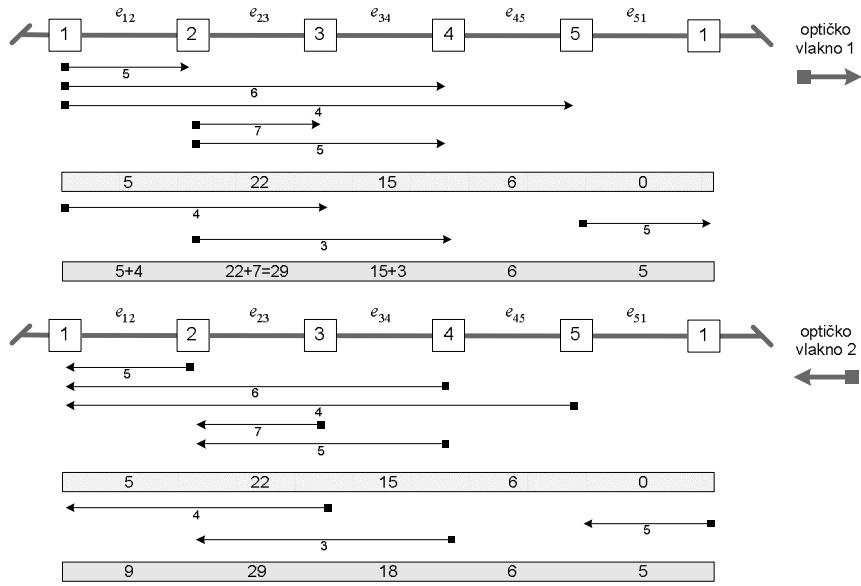
izvorišni čvor	odredišni čvor	prometni zahtjev	zaštita
1	2	5	DA
1	4	6	DA
1	5	4	DA
2	3	7	DA
2	4	5	DA
1	3	4	NE
1	5	5	NE
2	4	3	NE

Alat podržava dva modela SNCP prstena, kao što je opisano u poglavlju 2.5.3. Prvi model u režimu normalnog rada koristi samo jedno optičko vlakno za prijenos štićenih zahtjeva u oba smjera (istim smjerom prstena, ali različitim putevima), dok drugi model koristi oba optička vlakna i zahtjeve prenosi u oba smjera istim putem.



Slika 4.8: Usmjeravanje zaštićenih i nezaštićenih zahtjeva u SNCP – model 1

Usmjeravanje i iskoristivost za svako optičko vlakno je prikazano za prvi model na slici 4.8, a za drugi model na slici 4.9. Neovisno o modelu, nezaštićeni zahtjevi usmjeravaju se korištenjem oba optička vlakna, s time da se usmjeravanje nastoji izvesti tako da dodatno povećanje kapaciteta bude minimalno.



Slika 4.9: Usmjeravanje zaštićenih i nezaštićenih zahtjeva u SNCP – model 2

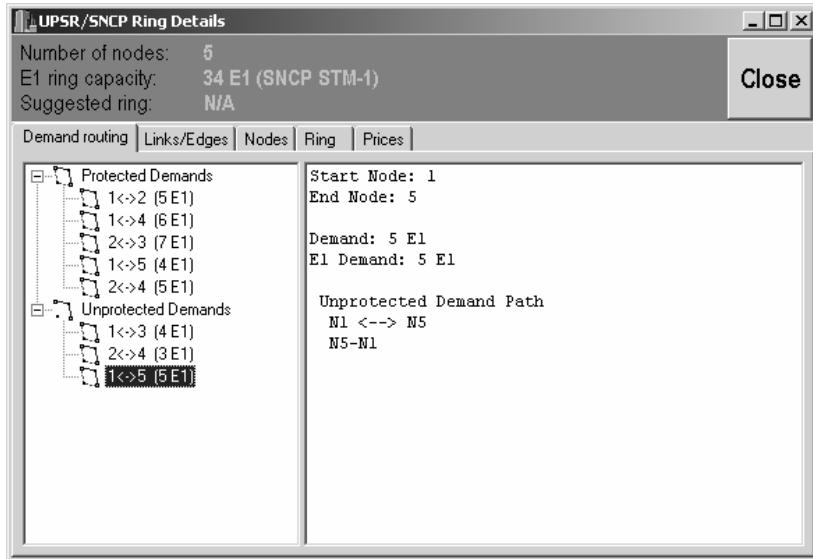
Na slikama je vidljivo da su optička vlakna različito opterećena, ovisno o korištenom modelu. U modelu u kojem se zaštićene zahtjeve koristi samo jedno optičko vlakno (slika 4.8), zahtijevani kapaciteti grana su maksimalni i jednaki sumi zaštićenih zahtjeva. Nezaštićeni zahtjevi se usmjeravaju kroz oba vlakna i to na isti način u oba modela, te neovisno o modelu povećavaju kapacitet grane e_{23} za 7 prometnih jedinica.

Može se uočiti da predloženo usmjeravanje nezaštićenih zahtjeva smanjuje opterećenje u odnosu na opterećenje koje bi se pojavilo u slučaju zaštite tih zahtjeva ili usmjeravanja korištenjem jednog optičkog vlakna. U tom slučaju povećanje opterećenja i kapaciteta grana/prstena bilo bi jednakо sumi nezaštićenih zahtjeva.

Iz prikazanih rezultata nameće se još jedan praktičan zaključak o primjeni različitih SNCP modela. U prvom modelu (slika 4.8) drugo optičko vlakno u prstenu nije uopće opterećeno zaštićenim zahtjevima u normalnom režimu rada, pa u tom slučaju možemo govoriti o zaštitnom optičkom vlaknu, odnosno zaštitnom prstenu. U drugom modelu (slika 4.9) oba vlakna su jednakо opterećena u normalnom načinu rada.

Na prikazanom primjeru, alat za dimenzioniranje vraća kao rezultat maksimalni kapacitet grana (kapacitet prstena) u SNCP jednak 34. Ta vrijednost dobiva se jednostavnim dodavanjem zahtijevanog kapaciteta za prijenos nezaštićenih zahtjeva (u našem primjeru 7) na kapacitet SNCP prstena (suma zaštićenih zahtjeva, tj. 27).

Prikaz rezultata usmjeravanja za nezaštićene zahtjeve vidljiv je na slici 4.10.



Slika 4.10: *mSDHRingSolver* – prikaz rezultata usmjeravanja nezaštićenih zahtjeva

4.2.2. Dijeljenje zahtjeva prema razinama

Problem dimenzioniranja prstena u kojem je omogućeno dijeljenje zahtjeva (RDPwS) može se riješiti optimalno u slučaju dozvoljenog proizvoljnog dijeljenja. U slučaju dozvoljenog samo cjelobrojnog dijeljenja zahtjeva problem je složeniji, no može se doći do podoptimalnog ili optimalnog rješenja (INDES algoritam). Poseban slučaj predstavlja realna izvedba dijeljenja zahtjeva – ukoliko se prometni zahtjevi definiraju prema prometnim razinama, svaka pojedina razina je nedjeljiva. Tako npr. zahtjev od 1 STM-1 nije moguće u praksi podijeliti na npr. 30+33 E1 zahtjeva, iako bi to moglo smanjiti kapacitet mreže.

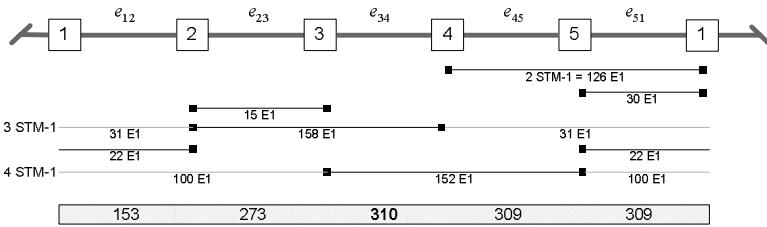
Zbog toga je prilagodbom EXACT i INDES algoritma izведен algoritam koji optimalno dijeli zahtjeve ali vodeći računa o definiranim prometnim razinama. Algoritmi se koriste u mrežama u kojima su zahtjevi definirani različitim prometnim razinama. Ukoliko su svi prometni zahtjevi istih razina koriste se standardni EXACT i INDES algoritmi bez prilagodbi.

U primjeru prstena od 5 čvorova definirani su zahtjevi kao što je prikazano tablicom.

Tablica 4.4: Definirani zahtjevi u primjeru

izvorišni čvor	odredišni čvor	prometni zahtjev	prometni zahtjev – E1
1	4	2 STM-1	126
1	5	30 E1	30
2	3	15 E1	15
2	4	3 STM-1	189
2	5	22 E1	22
3	5	4 STM-1	252

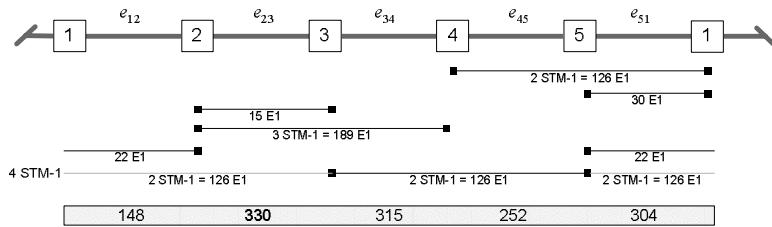
U SNCP modelu prstena zahtijevani kapacitet je 634 E1 jedinica. U MS-SPRing modelu bez dijeljenja zahtjeva optimalni kapacitet je 441 E1. Uz dijeljenja zahtjeva na E1 razini zahtijevani kapacitet je 310 E1. Rezultat usmjeravanja uz dijeljenja na E1 razini prikazan je shematski na slici.



Slika 4.11: Dijeljenje zahtjeva na E1 razini

U praksi je nemoguće izvesti dijeljenje na E1 razini ukoliko svi zahtjevi nisu definirani na E1 razini. Ukoliko su pritoci definirani na različitim razinama nije moguće demultiplexiranje na E1 razinu, bar ne bez nepotrebnih dodatnih troškova. Dijeljenje je moguće samo na razinama na kojima su zahtjevi definirani. Drugim riječima, 2 STM-1 zahtjeva ne mogu se podijeliti npr. 100 E1 u jedan smjer, a preostali 152 u drugi smjer.

Rezultat usmjeravanja uz dijeljenje prema definiranim razinama daje zahtijevani kapacitet prstena od 330 E1, što je smanjenje kapaciteta za oko 25% u odnosu na model bez dijeljenja zahtjeva. Rezultat je prikazan na slici.



Slika 4.12: Dijeljenje zahtjeva po razinama

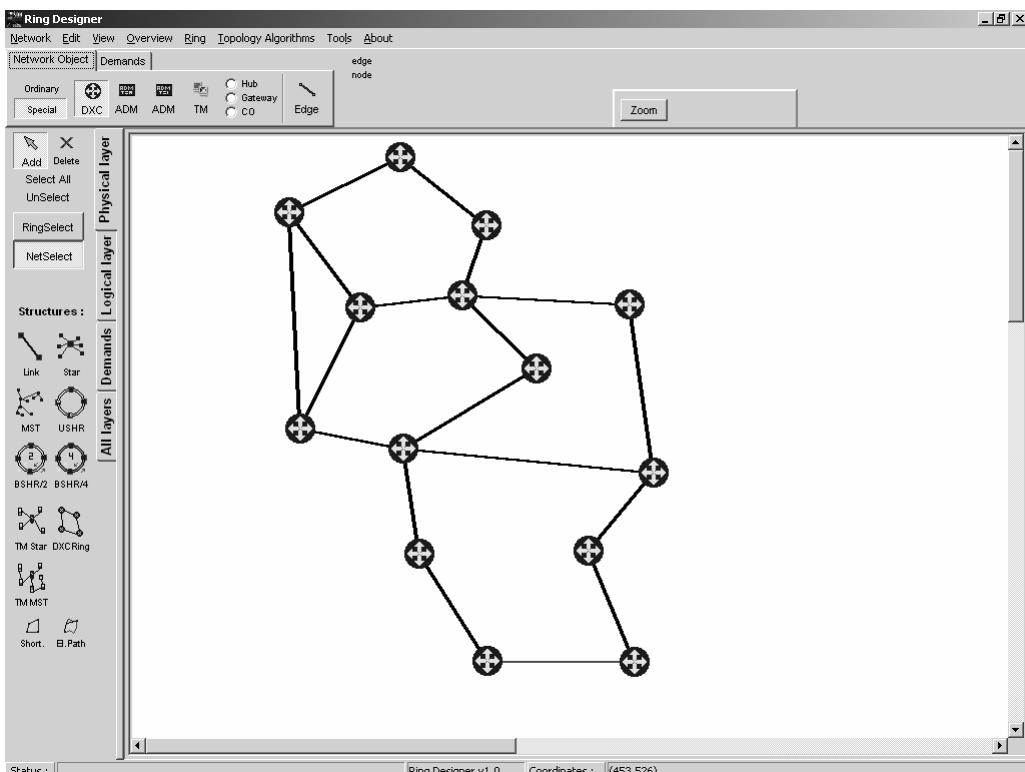
Rezultati koje daju prilagodeni algoritmi nisu nužno optimalni, no u praktičnoj primjeni su se pokazali kvalitetnim jer omogućavaju uštedu na kapacitetima mreže.

4.3. Alat za analizu i planiranje prstenastih mreža - mRingDesigner

mRingDesigner je složeniji alat koji služi za analizu i planiranje višestruko povezanih mreža. Alat je razvijen na bazi alata za izradu višestruko povezanih topologija koji je bio dio diplomskog rada autora. Alat omogućava definiranje topologije mreže (uz mogućnost automatizirane izrade topologije koja nudi višestruku povezanost) i prometnih zahtjeva, te na tako definiranim topologijama izvođenje sljedećih postupaka:

- pretraživanje topologije i određivanje prstenova prema zadanim kriterijima
- određivanje usmjeravanja prometnih zahtjeva
- određivanje podskupa prstenova za prekrivanje mreže
- analizu rezultata dobivenih različitim kombinacijama usmjeravanja i vrednovanja rješenja prekrivanja mreže

Osnovno sučelje za unos topologije mreže dano je na slici 4.13.



Slika 4.13: *mRingDesigner* – osnovno sučelje za definiranje mrežne topologije

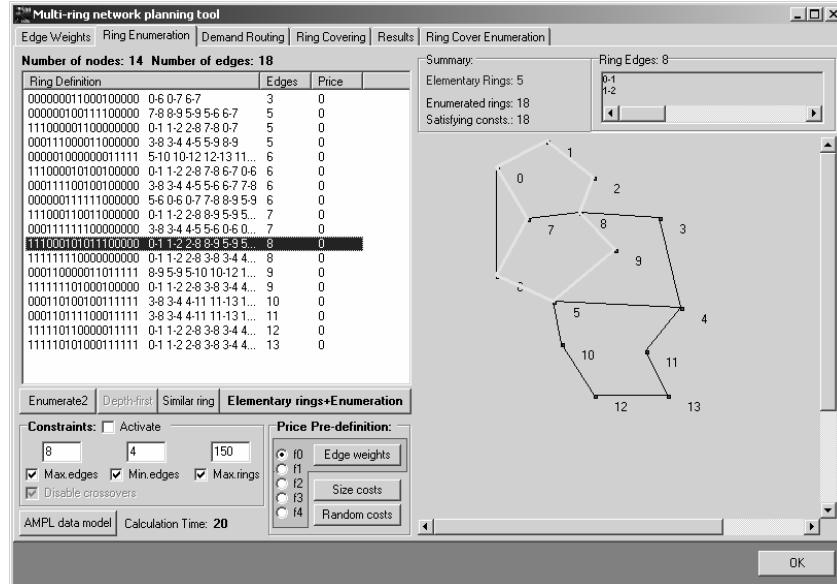
U nastavku je dan pregled funkcionalnosti alata vezan isključivo uz temu ovog rada. Ostale mogućnosti alata u ovom se prikazu zanemaruju.

4.3.1. Određivanje prstenova u mreži

Oba postupka za određivanje prstenova u mreži opisana u poglavljju 3.2.3 su implementirana u *mRingDesigner*-u. Zbog nešto jednostavnije izvedbene strukture preporuča se korištenje metode bazirane na određivanju elementarnih prstenova korištenjem minimalnog stabla.

Kako područje mogućih rješenja eksponencijalno raste s potencijom koja odgovara razlici broja grana i čvorova u mreži (2^{E-N+1}), jasno je da u većim mrežama određivanje prstenova nije izvedivo u realnom vremenu. Stoga se pribjegava ograničavanju rezultata – *mRingDesigner* omogućava definiranje minimalne i maksimalne veličine prstena, te ograničavanje broja prstenova koje treba pronaći. Na taj način se rezultati postižu u realnom vremenu.

Sučelje koje se koristi za definiranje ograničenja i analizu rezultata postupka određivanja prstenova u mreži dano je na slici 4.14.

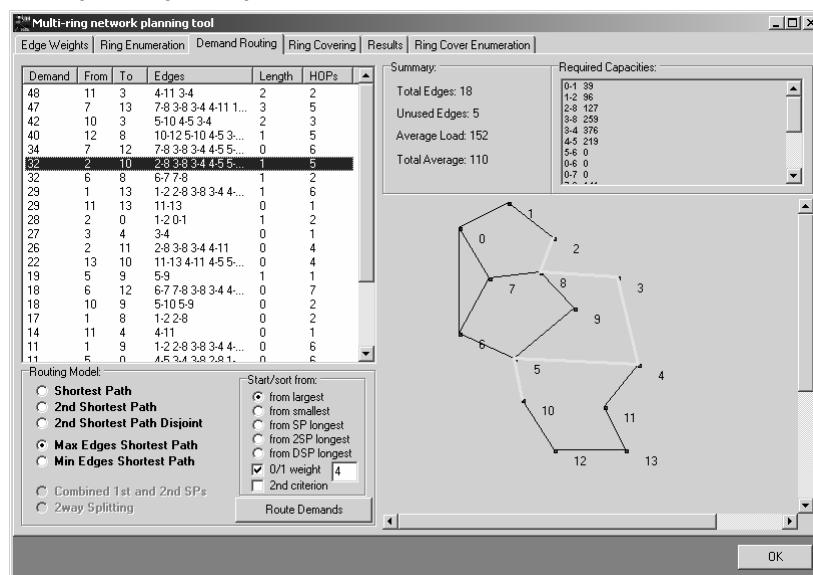


Slika 4.14: *mRingDesigner* – određivanje prstenova u mreži

Alat podržava snimanje rezultata enumeracije u AMPL uskladivom formatu. Težina prstenova za AMPL model mogu se definirati na različite načine – slučajnim vrijednostima, prema težinama ili duljinama grana, te korištenjem težinskih funkcija.

4.3.2. Usmjeravanje prometnih zahtjeva

Svi principi usmjeravanja prometnih zahtjeva opisani u poglavljju 3.2.4 su implementirani u *mRingDesigner*-u. Nakon što se odabere princip usmjeravanja i definiraju dodatni parametri (kriterij – duljina ili broj *hop*-ova, način sortiranja zahtjeva itd.) *mRingDesigner* vraća rezultate usmjeravanja, grafički ih prikazuje i proračunava prosječnu opterećenost grana u mreži. Primjer prikaza rezultata usmjeravanja dan je na slici 4.15.

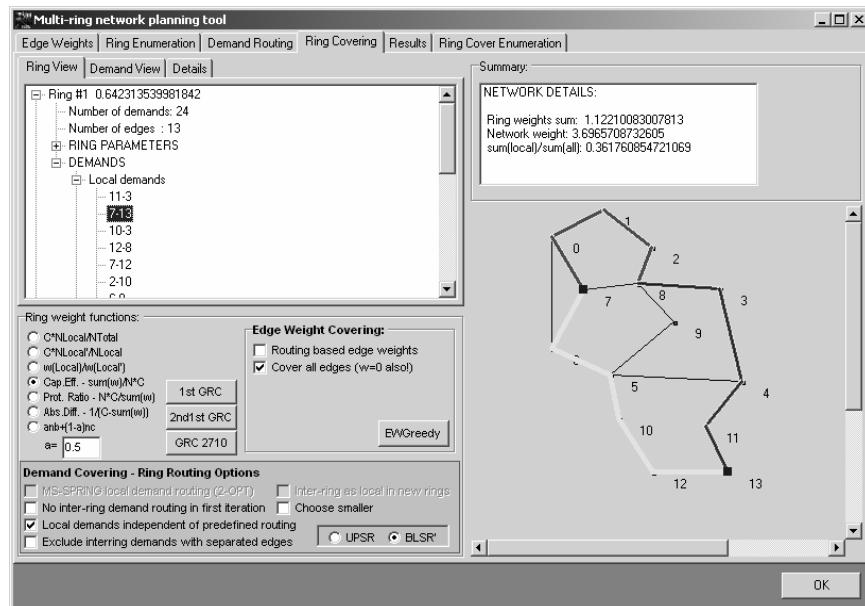


Slika 4.15: *mRingDesigner* – rezultati usmjeravanja prometnih zahtjeva

4.3.3. Prekrivanje mreže prstenovima

Postupci za prekrivanje mreže prstenovima opisani u poglavlju 3.2.5 implementirani su u alatu. Oba opisana postupka su podržana – moguće je određivanje prekrivanja uz detaljnu analizu usmjeravanja zahtjeva unutar prstenova, te određivanje prekrivanja temeljnog na inicijalnom usmjeravanju zahtjeva. Alat omogućava korištenje različitih modela vrednovanja prstenova (za svaki od postupaka definirano je nekoliko težinskih funkcija). Odabir prstenova vrši se iz skupa definiranog postupcima iz odjeljka 4.3.1, a sam odabir u određenim slučajevima ovisi i o inicijalnom usmjeravanju zahtjeva koje je rezultat postupaka opisanih u odjeljku 4.3.2.

Primjer dobivenih rezultata dan je na slici 4.16.



Slika 4.16: *mRingDesigner* – određivanje prekrivanja mreže prstenovima

Osim odabira prstenova koji prekrivaju mrežu alat kao rezultat daje i detaljan prikaz zahtjeva koji se djelomično (meduprstenasti zahtjevi) ili u potpunosti (lokalni zahtjevi) usmjeravaju unutar prstena. Za svaki od zahtjeva dana je i grafička interpretacija na definiranom modelu mreže.

Nakon što su prstenovi koji prekrivaju mrežu određeni, potrebno je nekako vrednovati i usporediti dobivena rješenja. Da bi se omogućila što detaljnija analiza rezultata za dobivenu mrežu određuju se karakteristične vrijednosti. Alat za resultantnu mrežu vraća sljedeće vrijednosti:

- broj prstenova
- opterećenje mreže
- zahtijevane kapacitete
- iskoristivost i redundanciju u mreži
- prosječnu iskoristivost resultantnih prstenova
- prosječnu duljinu prstenova
- udio lokalnih zahtjeva u ukupnom prometu
- broj grana prekrivenih prstenovima

Korisnik alata prema dobivenim vrijednostima može procijeniti kvalitetu pojedinog rješenja, te se odlučiti za najbolje. Dakle, alat neće sam donijeti odluku o najboljem rješenju već će to prepustiti dizajneru mreže.

Kao što je već pri opisu samog postupka za rješavanje RCP navedeno, izvođenja svih međufaza postupka mogu utjecati na konačne rezultate. Alat *mRingDesigner* definira niz parametara variranjem kojih je moguće utjecati na konačne rezultate. U tablici 4.5 su navedeni parametri koje je moguće definirati u pojedinim fazama postupka.

Tablica 4.5: Parametri kojima se može utjecati na rezultate postupka

Faza	Parametri
Određivanje prstenova	<ul style="list-style-type: none"> ○ minimalni broj čvorova u prstenu ○ maksimalni broj čvorova u prstenu ○ broj prstenova
Usmjeravanje prometnih zahtjeva	<ul style="list-style-type: none"> ○ slučajno generiranje prometnih zahtjeva ○ način usmjeravanja ○ kriterij određivanja težina grana ○ korištenje sekundarnog kriterija – da/ne ○ SP, SSP, DSSP ○ MxFER, MiFER + redoslijed zahtjeva
Prekrivanje uz analizu unutarprstenastog prometa	<ul style="list-style-type: none"> ○ kriteriji vrednovanja prstenova ○ postojanje meduprstenastih zahtjeva u prvom prstenu – da/ne ○ način usmjeravanja lokalnih zahtjeva unutar prstena ○ zabrana usmjeravanja meduprstenastih zahtjeva sa razdvojenim granama u istom prstenu – da/ne ○ korištenje sekundarnog kriterija – da/ne ○ razlikovanje lokalnih i pravih lokalnih zahtjeva
Prekrivanje prema inicijalnom usmjeravanju	<ul style="list-style-type: none"> ○ kriteriji vrednovanja prstenova ○ prekrivanje svih grana – da/ne (ili samo opterećenih) ○ korištenje sekundarnog kriterija – da/ne

Utjecaj parametara na izvođenje postupaka može biti različit. Kako se većinom radi o heurističkim algoritmima nema nikakvih garancija da će utjecaj istog parametra biti jednak u različitim primjerima mreža. Najbitnije jest da se promjenom parametara može utjecati na rezultate. Zadatak dizajnera je proučiti utjecaje te korišteni model što bolje prilagoditi svojim potrebama.

Alati koji su razvijani kao praktičan dio ovog rada predstavljaju solidnu osnovu za razvoj još naprednijih alata koji bi u konačnici trebali omogućiti djelomično automatizirano planiranje različitih vrsta prstenastih mreža. Ti alati trebali bi obuhvatiti i planiranje specijaliziranih tipova prstenastih mreža. Postojeći alati su primjenjeni u okviru nekoliko projekata Zavoda za telekomunikacije i njihovo korištenje omogućilo je bržu analizu predloženih rješenja.

5. Studijski primjeri planiranja prstenastih mreža

Metode koje se koriste u postupku planiranja prstenastih mreža implementirane su u opisanim programskim alatima. Njihovim povezivanjem realiziran je automatiziran postupak planiranja koji prema definiranim uvjetima obavlja pretraživanje mrežne topologije s ciljem određivanja prstenova sadržanih u mreži, inicijalno usmjeravanje prometnih zahtjeva i određivanje prstenova koji prekrivaju mrežu. U ovom se poglavlju iznose rezultati postupka planiranja dobiveni opisanim programskim alatima. Postupak je proveden za tri testne mreže, uz variranje početnih uvjeta i osnovnih parametara postupka. Za svaku mrežu provedeni su testovi i dobivena različita rješenja. Osim toga, testiranja su izvedena i korištenjem različitih težinskih funkcija. Na taj način je realizirano mnogo izvedivih rješenja, pri čemu su za svako određene karakteristične vrijednosti. Analizom i usporedbom karakterističnih vrijednosti odlučuje se o kvaliteti rješenja, te se pokušava izdvojiti najbolja od ponuđenih.

Na osnovu analize izneseni su zaključci te komentirani rezultati dobiveni uz različite početne uvjete. Rezultati dobivenim metodama implementiranim u praktičnom dijelu rada uspoređeni su s rezultatima koje daje jednostavan matematički model opisan u odjeljku 3.2.5.

5.1. Izvedba postupaka korištenih za dobivanje rezultata

Za dobivanje rezultata su korištena dva postupka bazirana na pohlepnim metodama za prekrivanje mreže prstenovima:

- Postupak A – prekrivanje mreže uz detaljnu analizu prometnih zahtjeva i usmjeravanje unutar prstenova
- Postupak B – prekrivanje mreže temeljeno na inicijalnom usmjeravanju prometnih zahtjeva

Oba postupka omogućavaju određivanje sljedećih karakterističnih vrijednosti za resultantne mreže:

- broja prstenova $|RCS|$
- ukupnog opterećenja mreže u standardnom režimu rada $\Sigma w(e_i)$ ($\forall e_i \in E$)
- ukupno zahtijevanih kapaciteta u mreži uz korištenje BSHR prstenova, $P_{N(BSHR)}$
- sume BSHR kapaciteta prstenova, C_{BSHR}
- prosječne veličine prstena, $l(R)$
- broja grana u rješenju, N_{EC}
- iskoristivosti i redundancije BSHR modela - μ_{BSHR} , r_{BSHR} – iskoristivost se određuje kao relativno opterećenje mreže – omjer ukupnog opterećenja i ukupnog kapaciteta u mreži
- prosječne iskoristivosti BSHR prstenova u mreži - μ_R

Postupak A omogućava i analizu USHR prstenova – moguće je određivanje ukupno zahtijevanih USHR kapaciteta $P_{N(USHR)}$, te prosječne iskoristivosti i redundancije u USHR prstenovima (μ_{USHR} , r_{USHR}). Rezultati vezani uz USHR prstenove su zanemareni i nisu analizirani, jer zbog karakteristika postupka B nije moguće dobiti ekvivalentne vrijednosti koje bi mogle poslužiti za usporedbu rezultata.

Zahvaljujući analizi prometa unutar prstenova, postupak A kao jednu od karakterističnih vrijednosti mreža vraća udio lokalnih zahtjeva L u ukupnom broju prometnih zahtjeva.

Bitno je naglasiti da se u izvođenju postupka A ne provodi lokalna optimizacija prometa u BSHR prstenovima, već se, zbog jednostavnosti izvedbe, zahtjevi usmjeravaju unutar prstena smjerom koji prolazi kroz manje grana (*min-hop* usmjeravanje). Optimizacijom usmjeravanja unutar BSHR prstenova moglo bi se ostvariti smanjenje ukupno instaliranih kapaciteta, te povećati iskoristivost mreže.

Testiranja obuhvaćaju izvođenje oba spomenuta postupka nad testnim mrežama i izvođenje matematičkog programa prema modelu opisanom u odjeljku 3.2.5. Nedostaci matematičkog modela su već izneseni prilikom njegova opisa. Dodatni problemi javljaju se prilikom interpretacije

dobivenih rezultata – budući da u matematičkom programu nema mogućnosti prilagodbe osnovnih parametara za vrijeme postupka dobiveni rezultati su neuskladivi sa rezultatima dobivenim izvođenjem pohlepnih metoda. Naime, matematički program se bazira na optimizaciji mreže prema početno definiranim parametrima. Bez obzira na medurezultate (u našem primjeru, odabir prstena u rješenje) definirani parametri se neće mijenjati, tj. ne dolazi do prilagođavanja parametara. U slučaju testnih mreža, parametri koji ulaze u matematički program su težine prstenova. Njih je moguće definirati različitim težinskim funkcijama, te je to jedini način utjecanja na rezultate dobivene matematičkim programom. Rezultat matematičkog programa je izbor prstenova koji prekrivaju sve grane u mreži. Taj rezultat se koristi samo kao kontrolni parametar za provjeru rezultata dobivenih ostalim metodama. Eventualnom dodatnom analizom tih rezultata moglo bi se odrediti karakteristične vrijednosti dobivenih rješenja, ali je zbog spomenutih nedostataka modela smislenost dodatne obrade rezultata upitna.

Definiranjem uvjeta pohlepnih postupaka utječe se na određivanje različitih rezultantnih mreža, što doprinosi proširenju područja rješenja.

Postupkom A mogu se odrediti karakteristični parametri prstena – veličina prstena $|R|$, kapacitet $C(R)$, ukupno opterećenje grana $\sum w(e_i)$ ($\forall e_i \in R$), te karakteristike usmjeravanih prometnih zahtjeva (broj lokalnih N_{Local} i međuprstenastih zahtjeva N_{Inter} , vrijednosti zahtjeva).

U postupku A primjenjuju se sljedeće težinske funkcije:

$$\begin{aligned}
f_1 &= C(R) \cdot \frac{N_{Local}}{N_{Inter}} & f_2 &= C(R) \cdot \frac{N'_{Local}}{N_{Local}} \\
f_3 &= \frac{\sum d_{Local}}{\sum d_{Inter}} & f_4 &= \frac{\sum_{e_i \in E_R} w(e_i)}{|R| \cdot C(R)} \\
f_5 &= \frac{|R| \cdot C(R)}{\sum_{e_i \in E_R} w(e_i)} & f_6 &= \alpha \cdot \frac{\sum_{e_i \in E_R} w(e_i)}{|R| \cdot C(R) - \sum_{e_i \in E_R} w(e_i)} + (1 - \alpha) \cdot \frac{2 \cdot |D| - N_{Inter}}{2 \cdot |D|} \\
f_7 &= \frac{\sum d'_{Local}}{\sum d_{Inter}}
\end{aligned}$$

Funkcije f_1 i f_2 su varijante težinske funkcije definirane u (3.2). Radi se o funkcijama koje težinu definiraju proporcionalno kapacitetu prstena, uz što veći broj lokalnih zahtjeva. N_{Local} predstavlja broj lokalnih zahtjeva u prstenu, a N_{Inter} broj međuprstenastih zahtjeva. N'_{Local} je karakterističan parametar vezan uz implementaciju postupka – jednak je broju 'pravih' lokalnih zahtjeva. Detalji o određivanju tzv. pravih zahtjeva opisani su u odjeljku 3.2.5. Korištenjem f_1 i f_2 odabiru se prstenovi većih kapaciteta i sa većim brojem prometnih zahtjeva. Funkcija f_3 je funkcija identična (3.5), dok je f_7 varijanta iste funkcije ali uz razmatranje 'pravih' lokalnih zahtjeva umjesto lokalnih. Funkcije f_3 i f_7 definiraju težinu prstena ovisno o prometnom udjelu (ne broju) lokalnih zahtjeva u odnosu na međuprstenaste zahtjeve. Na taj način se odabiru prstenovi sa dominantnim lokalnim prometom. Obje funkcije koje razmatraju 'prave' lokalne zahtjeve (f_2 , f_7) imaju za cilj smanjiti međuprstenasti promet, odnosno odabrati prstenovi sa što više lokalnog prometa. Funkcija f_4 definira težinu prstena kao iskoristivost ili efikasnost, te je po definiciji jednaka invertiranoj funkciji (3.3). Korištenjem te funkcije odabirat će se najbolje iskorišteni prstenovi, što bi moglo uvjetovati i dobro iskorištenje cijele mreže. Funkcija f_5 kao mjeru težine uvodi redundanciju prstena, kao u (3.3). Funkcija f_6 je izvedena prema (3.4) i omogućava definiranje težine kombiniranjem dva utjecaja – iskoristivosti prstena i udjela međuprstenastog prometa. Na taj način dobiva se kvalitetna definicija težine prstena jer, za razliku od ostalih težinskih funkcija, na rezultat utječu oba bitna parametra prstena.

Postupkom B moguće je za svaki prsten R odrediti tri parametra – veličinu prstena $|R|$, maksimalno opterećenje grane $\max w(e_i)$ koje odgovara kapacitetu prstena, te ukupno opterećenje grana u prstenu $\sum w(e_i)$ ($\forall e_i \in R$). Ti parametri su dovoljni za određivanje težina prstenova korištenjem težinskih funkcija:

$$g_1 = \frac{\sum_{\forall e_i \in E_R} w(e_i)}{\max_{e_i \in E_R} w(e_i) \cdot |R|}$$

$$g_2 = \max_{e_i \in E_R} w(e_i) \cdot |R| - \sum_{\forall e_i \in E_R} w(e_i)$$

$$g_3 = \sum_{\forall e_i \in E_R} w(e_i)$$

$$g_4 = \frac{\left(\sum_{\forall e_i \in E_R} w(e_i) \right)^2}{\max_{e_i \in E_R} w(e_i) \cdot |R|}$$

$$g_5 = \frac{\left(\sum_{\forall e_i \in E_R} w(e_i) \right)^3}{\max_{e_i \in E_R} w(e_i) \cdot |R|}$$

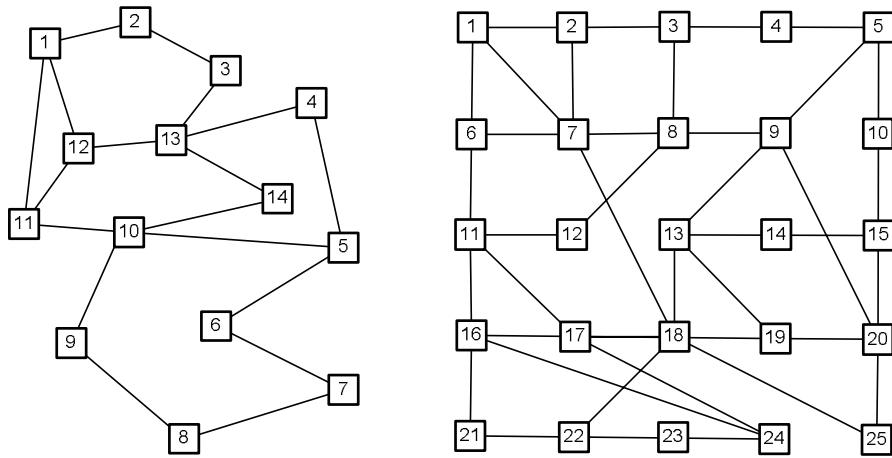
$$g_6 = \frac{1}{\max_{e_i \in E_R} w(e_i) \cdot |R| - \sum_{\forall e_i \in E_R} w(e_i)}$$

Težinska funkcija g_1 je definirana prema (3.6) i predstavlja iskoristivost ili efikasnost prstena. Rezultat odabira te metode biti će efikasni prsteni. Funkcija g_2 kao težinu definira absolutnu vrijednost slobodnih kapaciteta u prstenu. Može se pretpostaviti da će rezultati dobiveni korištenjem te metode biti vrlo slabe razine iskoristivosti jer će se odabirati prstenovi sa absolutno najvećim neiskorištenim kapacitetima. Najčešće će se raditi o velikim prstenovima, pa rezultati u većini slučajeva neće biti zadovoljavajući. Funkcija g_3 kao težinu prstena definira ukupno opterećenje – rezultat će biti prometno najviše opterećeni prstenovi. Funkcije g_4 i g_5 su definirane kao umnošci g_1 i g_3 , a njima se nastoji pojačati utjecaj količine prometa koji se prenosi prstenom – u većini slučajeva g_4 i g_5 će rezultirati sličnim rezultatima, dok su razlike moguće samo kod velikih prstenovima velikog kapaciteta. Definira se i g_6 kao invertirana funkcija g_2 – očekivani rezultati će sadržavati prstenove koji su absolutno najviše iskoršteni – pretpostavka je da će se raditi o malim prstenovima manjih kapaciteta.

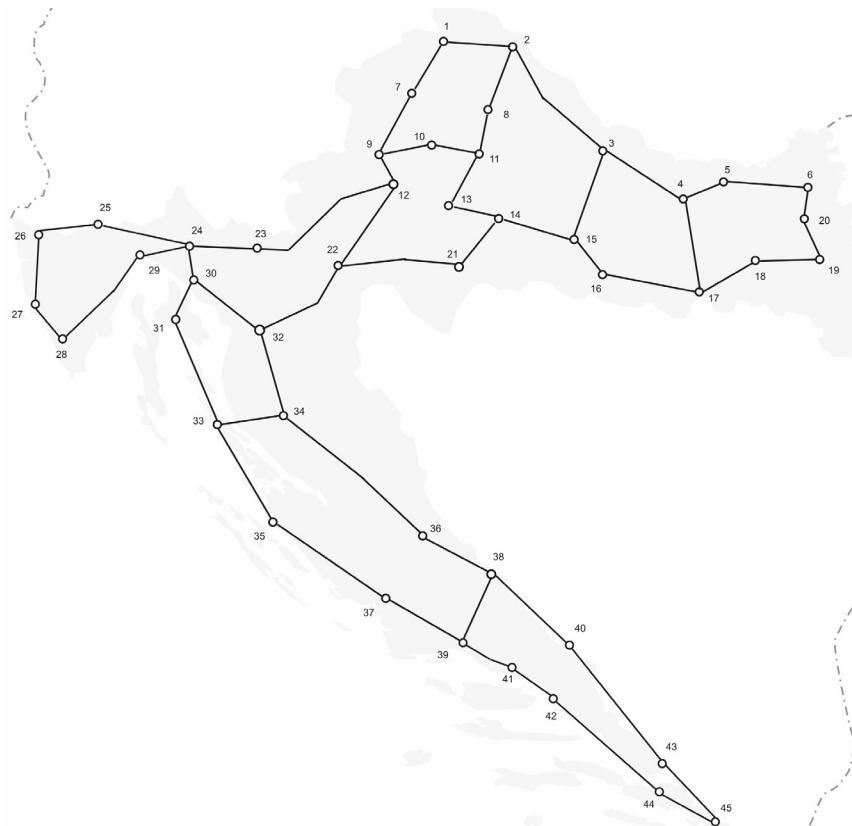
Težinske funkcije se u tabličnim prikazima rezultata označavaju uvedenim oznakama. Svaka resultantna mreža karakterizira se prije pobrojanim karakterističnim vrijednostima, te se prema njima nastoji analizirati ponuđeno rješenje i donijeti sud o izboru najboljeg. Definiranjem kontrolne funkcije dizajneri mreže mogu uvesti prioritete različitim karakterističnim vrijednostima i korištenjem proizvoljnih faktora definirati važnost svake od njih. Definiranje takve kontrolne funkcije u ovom radu nije izvedeno, već je obavljena analiza rezultata neovisnim vrednovanjem pojedinačnih karakterističnih vrijednosti mreže.

5.2. Opis testnih mreža i studijskih primjera

Testne mreže su prikazane na slikama 5.1 i 5.2. Testna mreža 1 izgrađena je od 14 čvorova i 18 grana – to je mreža koja se koristi u većini razmatranja u ovom radu. Testna mreža 2 je izgrađena od 25 čvorova i 40 grana i služi isključivo kao model za testiranje i analizu postupka planiranja. Testna mreža 3 je izvedena na temelju stvarnog modela transportne mreže. Izgrađena je od 45 čvorova i 54 grane.



Slika 5.1: Testne mreže 1 i 2



Slika 5.2: Testna mreža 3

Prometni zahtjevi u testnim mrežama definirani su slučajnim odabirom krajnjih čvorova (uz onemogućeno ponavljanje zahtjeva između istog para čvorova). Vrijednost pojedinog zahtjeva je također slučajno odabrana – može poprimiti vrijednosti između 1 i 50. Slučajni izbor ravna se prema uniformnoj razdiobi.

Analiza rezultata je obavljena za pet studijskih primjera. Primjeri su definirani u tablici 5.1. Za svaki test definirana je testna mreža, broj i način definiranja prometnih zahtjeva, suma prometnih zahtjeva, te oznake koje se koriste kod prikaza i analize rezultata. Za *SP1* definira se 25 prometnih zahtjeva što je oko 27% od ukupno mogućeg broja zahtjeva. Za *SP2* definira se 50 prometnih zahtjeva (55%) s ciljem analize utjecaja povećanja broja prometnih zahtjeva na rezultate planiranja. U *SP3* i *SP4* definira se 100 (33%) odnosno 200 (66%) prometnih zahtjeva, dok je u *SP5* definirano 500 prometnih zahtjeva (oko 50% od ukupno mogućeg broj prometnih zahtjeva).

Tablica 5.1: Karakteristike testova

Studijski primjer	Mreža	Prometni zahtjevi	Suma zahtjeva	Oznaka rezultata
SP1	Testna mreža 1	25, slučajno generirani	624	A
SP2	Testna mreža 1	50, slučajno generirani	1247	B
SP3	Testna mreža 2	100, slučajno generirani	2847	C
SP4	Testna mreža 2	220, slučajno generirani	4897	D
SP5	Testna mreža 3	500, slučajno generirani	13246	E

5.3. Rezultati određivanja prstenova u mrežama

U testnim mrežama moguće je redom definirati po 5, 16 i 10 elementarnih prstenova, te njihovim kombiniranjem dobiti 31, 65535 i 1023 složenijih struktura. Broj prstenova koji nastaju takvim kombiniranjem je jednak 18, 4371 i 61.

U slučaju većih mreža broj prstenova može biti poprilično velik (tablica 1.2 u odjeljku 1.2.5) te se stoga broj i veličina prstenova koji se razmatraju može ograničiti. To je učinjeno u postupku planiranja za testnu mrežu 2.

Rezultati postupka određivanja prstenova prikazani su tablično u tablici 5.2.

Tablica 5.2: Rezultati određivanja prstenova u testnim mrežama

Mreža	Svi prstenovi				Svi prstenovi 4-16			
	Broj	Vrijeme	MIN	MAX	Broj	Vrijeme	MIN	MAX
Mreža 1	18	20 ms	3	13	17	20 ms	4	13
Mreža 2	4317	671 s	3	24	2022	110 s	4	16
Mreža 3	61	520 ms	5	37	30	140 ms	5	16

Tablica prikazuje rezultate enumeracije. Prikazan je broj dobivenih prstenova, vrijeme trajanja postupka pretraživanja, te najmanji (MIN) i najveći (MAX) broj čvorova u određenim prstenovima. U prvom dijelu prikazani su rezultati pretraživanja koji uključuju sve prstenove koji se mogu definirati na zadanim mrežnim topologijama. Drugi dio ograničava veličinu prstenova – razmatraju se samo prstenovi koji uključuju 4 do 16 čvorova. Pretpostavka o veličini analiziranih prstenova od 4 do 16 čvorova je realna. U praksi se rijetko koriste prstenovi s manje od 4 čvora, a prstenovi veći od 16 čvorova nisu podržani SDH standardima. Svi postupci su izvođeni uz takvo ograničenje veličine prstenova.

Zbog velikog broja prstenova i vremenski zahtjevnog postupka određivanja prstenova, za testnu mrežu 2 je dodatno proanalizirana mogućnost određivanja ograničenog broja prstenova. Rezultati su vidljivi u tablici 5.3.

Tablica 5.3: Određivanje ograničenog broja prstenova u testnoj mreži 2

Broj prstenova	Sve veličine prstenova			Prstenovi 4-16 čvorova
	Vrijeme	MIN	MAX	Vrijeme
150	140 ms	3	18	140 ms
300	230 ms	3	20	690 ms
1000	3,7 s	3	21	3,7 s
1500	6 s	3	22	9 s
2000	15 s	3	23	49 s

Očito da je postupak određivanja manjeg broja prstenova izvediv u realnom vremenu za svaki oblik mreže. Uvjetima o veličini prstenova i broju traženih prstenova mogu se odrediti podskupovi prstenova koji su dovoljni za provođenje postupka određivanja prstenova koji prekrivaju mrežu. Zanimljivo je primijetiti da za određivanje 2000 prstenova veličina 4-16 čvorova treba više nego dvostruko manje vremena nego li za određivanje svih prstenova te veličine, iako je ukupni broj takvih prstenova 2022, dakle nešto više od 1% veći nego u slučaju s definiranim ograničenjima na pretraživanje. Razlog tome leži u načinu implementacije enumeracijske metode – kod pretraživanja svih prstenova postoji mogućnost da velik broj preostalih prstenova nije uskladiv s ograničenjima.

5.4. Rezultati postupka planiranja

Kako je na rezultate postupka za određivanje prekrivanja mreže prstenovima moguće utjecati variranjem početnih uvjeta, a karakteristične vrijednosti rezultantnih mreža ne mogu se iskoristiti za donošenje jednoznačne odluke o najboljem rješenju, prikaz rezultata izведен je tablično za svaki od testova. Za svaki od rezultata iznesene su sve karakteristične vrijednosti, pri čemu su značajnije posebno označene i komentirane. Daje se i grafička interpretacija nekih od ponuđenih rješenja kako bi se zornije ukazalo na različite mogućnosti prekrivanja. Usmjeravanje prometnih zahtjeva nije prikazano kao dio rezultata, no moguće ga je dobiti korištenjem programskih alata opisanih u poglavlju 4.

Za studijski primjer 1 iznose se detaljni rezultati dobiveni različitim varijacijama postupaka, te se daje naglasak na mogućnosti utjecanja na rezultate promjenom kontrolnih parametara. Za ostale primjere nisu dani tako detaljni rezultati, već su izdvojeni rezultati koji mogu ukazati na pravilnosti i olakšati donošenje suda o kvaliteti ponuđenih rješenja.

Studijski primjeri koji se odnose na iste mrežne topologije su dodatno proanalizirani kako bi se uočile sličnosti u ponuđenim rješenjima.

Za svaku testnu mrežu se, osim korištenjem predloženih postupaka A i B, prekrivanje određuje i korištenjem matematičkog programa. Težine prstenova koje predstavljaju ulazne podatke za matematički model definiraju se prema težinskim funkcijama korištenim u postupku B.

5.4.1. Testna mreža 1 - studijski primjer 1

U tablici 5.4 su prikazani rezultati dobiveni standardnim pohlepnim postupkom A za SPI – definirana su sljedeća ograničenja: prstenovi su ograničeni na veličine od 4 do 16 čvorova, inicijalno usmjeravanje prometnih zahtjeva obavlja se najkraćim putevima, te je definiran sekundarni kriterij odabira manjih prstenova u slučaju identičnih vrijednosti težinskih funkcija.

Tablica 5.4: SP1 – postupak A - rezultati 1

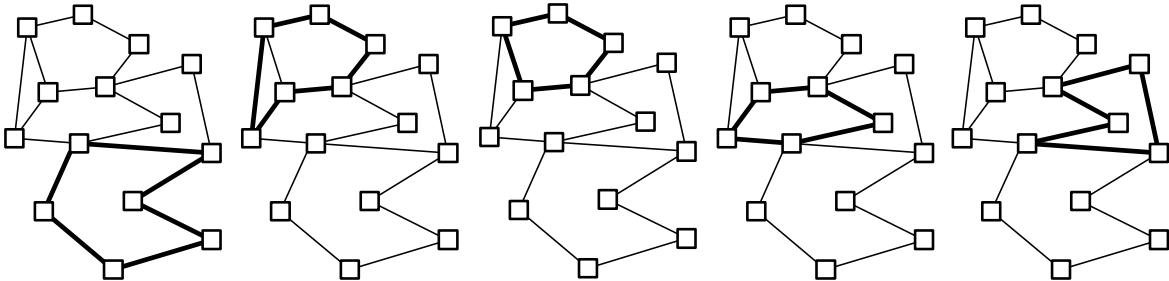
	f	$ RCS $	$\Sigma w(e_i)$	$P_{N(USHR)}$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	r_{BSHR}	N_{EC}
A_1	f_1	5	1702	6792	4673	5.4	36%	856	0.364	0.374	2.746	18
A_2	f_2	5	1702	6756	4218	5.4	32%	705	0.404	0.365	2.478	18
A_3	f_3	5	1702	7035	4017	8.2	84%	415	0.424	0.332	2.360	18
A_4	f_4	6	1702	7762	4193	6.7	60%	654	0.406	0.425	2.464	18
A_5	f_5	6	1702	12163	8271	10.6	52%	788	0.206	0.190	4.859	18
A_6	f_6	4	1702	7451	3824	8.8	80%	399	0.445	0.519	2.246	18
A_7	f_7	5	1702	7086	4128	8.4	88%	428	0.413	0.319	2.425	18

Rezultati dobiveni uz ista ograničenja, ali uz primjenu modificirane pohlepne metode koja kao prvi prsten u rješenje odabire drugi po težini, prikazani su u tablici 5.5.

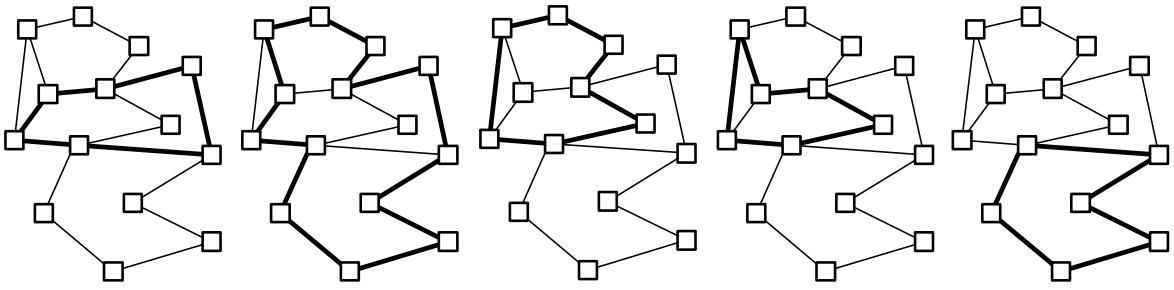
Tablica 5.5: SP1 – postupak A - rezultati 2

	f	$ RCS $	$\Sigma w(e_i)$	$P_{N(USHR)}$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	r_{BSHR}	N_{EC}
A_8	f_1	6	1702	6980	4914	5.5	36%	878	0.346	0.329	2.887	18
A_9	f_2	7	1702	9767	5578	6.57	60%	757	0.305	0.265	3.277	18
A_{10}	f_3	7	1702	7800	4702	7.71	60%	591	0.362	0.311	2.763	18
A_{11}	f_4	5	1702	7953	3740	7.6	48%	526	0.456	0.453	2.197	18
A_{12}	f_5	8	1702	9674	7728	8.37	48%	872	0.220	0.202	4.541	18
A_{13}	f_6	4	1702	8126	3913	9.3	56%	434	0.435	0.481	2.299	18
A_{14}	f_7	7	1702	7091	4188	7.28	72%	446	0.406	0.349	2.461	18

U ovom slučaju bolji rezultati (promatrujući smanjenje potrebnih BSHR kapaciteta) u odnosu na standardnu metodu dobivaju se za težinske funkcije f_4 i f_5 . Pregledom obje tablice teško je utvrditi najbolje od ponuđenih rješenja. Ukoliko se promatra iskoristivost mreže, prosječna iskoristivost prstena i minimizacija instaliranih BSHR kapaciteta, tada bi najbolje rješenje moglo biti prekrivanje dobiveno modificiranom pohlepnom metodom uz težinsku funkciju f_4 – rezultat A_{11} , a u izbor bi mogla ući i rješenja A_6 i A_3 . Obzirom na maksimalan udio lokalnog prometa u resultantnoj mreži najbolja rješenja dobivaju se primjenom težinske funkcije f_7 – najbolja prekrivanja u tom slučaju predstavljaju A_7 i A_{14} . Na slikama 5.3 i 5.4 su kao primjeri prikazana prekrivanja A_2 i A_{11} . Iako je A_{11} prema kriterijima bolje rješenje, prekrivanje A_2 čini se realnijim – manje je preklapanja prstenova i ne radi se o velikim prstenovima. Bez obzira na to, za tako malu mrežu kao što je testna mreža 1, 5 prstenova (što je 27% od ukupnog broja prstenova koje je moguće realizirati) ne predstavljaju zadovoljavajuće rješenje.



Slika 5.3: SP1 – prikaz rezultata A_2



Slika 5.4: SP1 – prikaz rezultata A_{11}

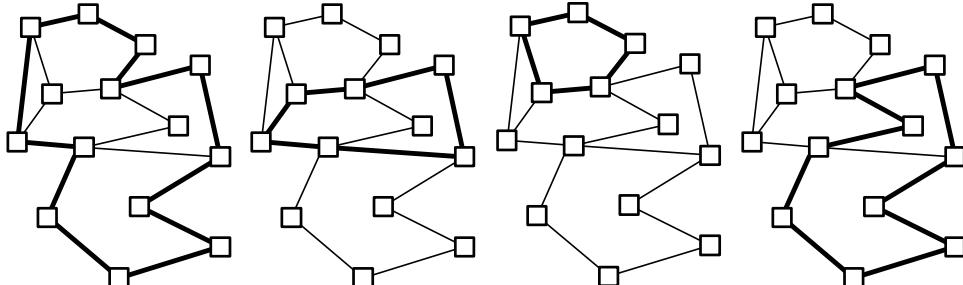
Slike 5.3 i 5.4 mogu ukazati na karakteristike težinskih funkcija koje se koriste. Rezultat A_2 dobiven je težinskom funkcijom f_2 koja u pravilu bira manje prstenove, dok se u A_{12} koristi funkcija f_4 koja odabire prstenove prema njihovoj iskoristivosti – najčešće su u početku izabrani veći prstenovi koji efikasno prenose prometne zahtjeve, a na kraju manji prstenovi koji prekrivaju preostale prometne zahtjeve. Kod A_{11} može se primjetiti da su svi čvorovi u mreži prekriveni korištenjem prva tri prstena – preostali prsteni ulaze u rješenje samo zbog karakteristične izvedbe postupka koji završava tek kad su svi zahtjevi prekriveni – dodatnim uvjetima taj nedostatak postupka može se eliminirati.

Očito da razlike u odabiru prvog prstena utječu na rezultate, no utjecaj nije dominantan i ne rezultira znatnjim promjenama. Pokazalo se da inicijalno usmjeravanje nema preveliki utjecaj na rezultate, no može utjecati na iskoristivost mreža i povećanje opterećenja u standardnom načinu rada. Veći utjecaj mogu imati kontrolni parametri. Tablica 5.6 sadrži rezultate dobivene uz uvjet da se u prstenove ne uključuju međuprstenasti zahtjevi koji su inicijalno usmjereni razdvojenim granama unutar prstena. Uočljiva su smanjenja zahtijevanih kapaciteta, što poboljšava iskoristivost mreže. U rješenju A_{14} smanjuje se broj prstenova (slika 5.5), no prstenovi se još uvijek preklapaju i stvara se nepotrebna redundancija u prekrivanju čvorova.

Tablica 5.6: SP1 – postupak A – rezultati 3

	f	$ RCS $	$\sum w(e_i)$	$P_{N(USHR)}$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	r_{BSHR}	N_{EC}
A_{15}	f_2	6↑	1702	6582↓	3408↓	6.7↓	85%•	379	0.499↑	0.377↑	2.002	18
A_{16}	f_6	4•	1702	7790↑	3742↓	8.8•	72%↓	401	0.455↑	0.572↑	2.199	18
A_{17}	f_7	4↓	1702	6656↓	3410↓	8↓	88%•	365	0.499↑	0.423↑	2.004	18

• ista vrijednost ↑ porast ↓ smanjenje



Slika 5.5: SP1 – prikaz rezultata A_{17}

Svi dosadašnji rezultati su pri usmjeravanju zahtjeva unutar prstenova poštivali inicijalno usmjeravanje. Ukoliko se kao kontrolni parametar uključi mogućnost da se lokalno unutar prstena usmjeravaju zahtjevi kojima su i izvorište i odredište sadržani u prstenu, dobivaju se rezultati prikazani tablicom 5.7. Utjecaj takvog načina usmjeravanja odražava se na vrijednosti karakterističnih parametara mreže posebno naznačenih u tablici – iako je inicijalno usmjeravanje identično kao i u prethodnim slučajevima, dolazi do promjene ukupnog opterećenja grana zbog

naknadne promjene usmjeravanja prometnih zahtjeva. Utjecaj je vidljiv i na broju grana uključenih u rješenje – za razliku od prethodno iznesenih rezultata, ovdje postoje rješenja koja ne koriste sve grane u mreži. U tablici su označene promjene u odnosu na početne rezultate.

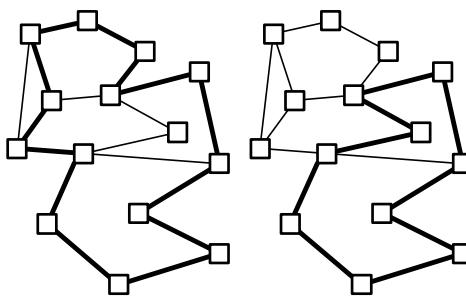
Tablica 5.7: SP1 – postupak A – rezultati 4

	f	$ RCS $	$\Sigma w(e_i)$	$P_{N(USHR)}$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	r_{BSHR}	N_{EC}
A_{18}	f_2	$2\downarrow$	2363 \clubsuit	8182 \downarrow	4321 \downarrow	9 \downarrow	96% $\uparrow\uparrow$	349 \downarrow	0.547 \uparrow	0.397 \uparrow	1.829	16 \clubsuit
A_{19}	f_4	$4\downarrow$	1789 \clubsuit	7268 \downarrow	4030 \downarrow	6.5 \downarrow	40% \downarrow	625 \downarrow	0.444 \uparrow	0.409 \downarrow	2.253	18
A_{20}	f_6	$3\downarrow$	2000 \clubsuit	6905 \downarrow	3686\downarrow	7.3 \uparrow	96% \uparrow	362 \downarrow	0.543 \uparrow	0.437 \uparrow	1.843	17 \clubsuit
A_{21}	f_7^*	$2\downarrow$	2045 \clubsuit	7026 \downarrow	3648\downarrow	9 \uparrow	96% \uparrow	340 \downarrow	0.561 \uparrow	0.534 \uparrow	1.784	16

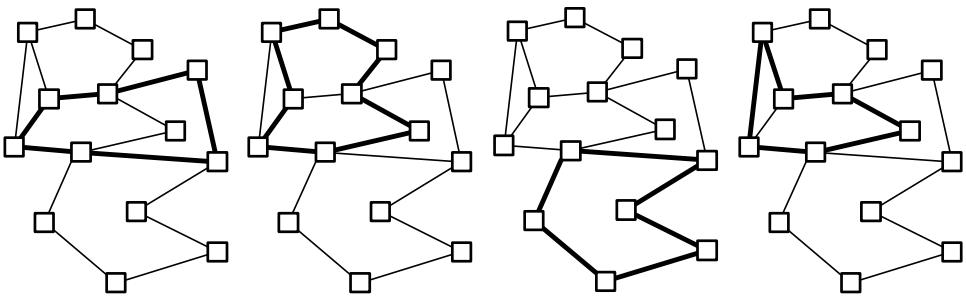
• ista vrijednost \uparrow porast \downarrow smanjenje \clubsuit posebna naznaka

Rezultat uz korištenje težinske funkcije f_7 (označen sa *) je dobiven korištenjem modificirane pohlepne metode, dakle uz izbor drugog najboljeg prstena u prvoj iteraciji postupka. Rješenja iz tablice su prikazana na slikama 5.6, 5.7, 5.8 i 5.9. Sva rješenja sastavljena od dva prstena svode se na jedan veći prsten kojim se odvija velik promet (većinu prometa čine lokalni zahtjevi, što se vidi iz parametra udjela lokalnog prometa na razini mreže). U svim rezultatima uočljivo je smanjenje zahtijevanih BSHR kapaciteta, smanjenje ukupnih kapaciteta prstenova, te povećanje iskoristivosti mreže. Povećanje iskoristivosti mreže nije nužno uzrokovano smanjenjem zahtijevanih kapaciteta, već na njega može utjecati i povećanje opterećenja uslijed promjena u usmjeravanju prometnih zahtjeva. Zanimljivo je primijetiti da uz korištenje funkcije f_4 dolazi do smanjenja udjela lokalnih prometnih zahtjeva, što samo potvrđuje da iskoristivost prstenova ne ovisi o broju i udjelu lokalnih zahtjeva. S druge strane, preostale tri razmatrane težinske funkcije direktno su ovisne o lokalnim zahtjevima, te je očito povećanje njihovog udjela u prometu u mreži. Smanjenje broja prstenova u rješenjima A_{18} , A_{20} i A_{21} može se objasniti upravo tom ovisnošću – zbog dodatnog usmjeravanja unutar prstenova, težinske funkcije pridružene tim rješenjima inicijalno odabiru prstenove sa velikim lokalnim prometom. Kao što je vidljivo na slikama, to su u pravilu veliki prstenovi koji sadrže većinu prometa (lokalnog). Preostali prstenovi su manji i prenose preostale prometne zahtjeve.

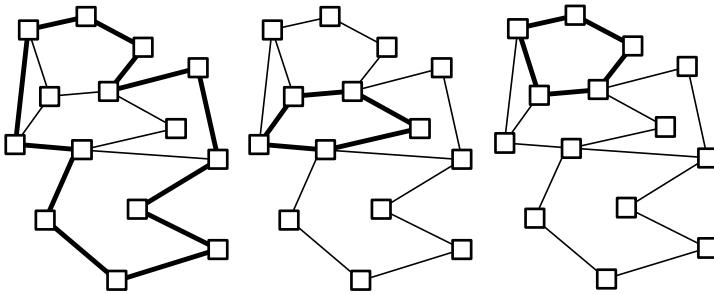
Rješenje A_{19} je specifično. Za dobivanje tog rješenja koristi se težinska funkcija f_4 (opterećenje prstena). U tom rješenju dolazi do smanjenja ukupno zahtijevanih kapaciteta, ali se ujedno smanjuje i udio lokalnog prometa (unatoč usmjeravanju na razini prstenova) u mreži. Prosječna iskoristivost prstenova u odnosu na rješenje A_4 se također smanjuje, no dolazi do porasta iskoristivosti mreže. Zbog poboljšanja ta dva parametra – ukupno zahtijevanih kapaciteta i iskoristivosti mreže se to rješenje može smatrati boljim od rješenja A_4 dobivenog korištenjem iste težinske funkcije.



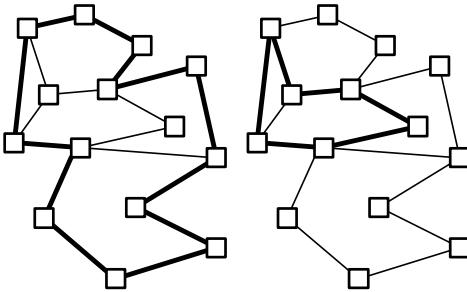
Slika 5.6: SP1 – prikaz rezultata A_{18}



Slika 5.7: SP1 – prikaz rezultata A_{19}



Slika 5.8: SP1 – prikaz rezultata A_{20}



Slika 5.9: SP1 – prikaz rezultata A_{21}

Prije prelaska na analizu rezultata dobivenih postupkom B, sagledani su rezultati postupka A. U tablici su odabrana po četiri najbolja rješenja od ponuđenih, za svaki definirani kriterij. Bitno je naglasiti da se variranjem uvjeta mogu dobiti još neki rezultati koji bi mogli ući u izbor najboljih po pojedinim kriterijima, no ovdje su razmatrani samo prethodno izneseni rezultati.

Tablica 5.8: SP1 – postupak A – najbolja rješenja

Kriterij	Najbolja rješenja
Najmanji zahtijevani BSHR kapaciteti	A_{15} (3408), A_{17} (3410), A_{21} (3648), A_{11} (3740)
Najveća iskoristivost mreže	A_{21} (0.561), A_{18} (0.547), A_{20} (0.543), A_{15} (0.499)
Najveća prosječna iskoristivost prstenova	A_{16} (0.572), A_{21} (0.534), A_6 (0.519), A_{17} (0.423)
Najmanji broj prstenova	A_{21} (2), A_{18} (2), A_{20} (3), A_{16} (4)
Najveći udio lokalnih prometnih zahtjeva	A_{21} (96%), A_{20} (96%), A_{18} (96%), A_{17} (88%)

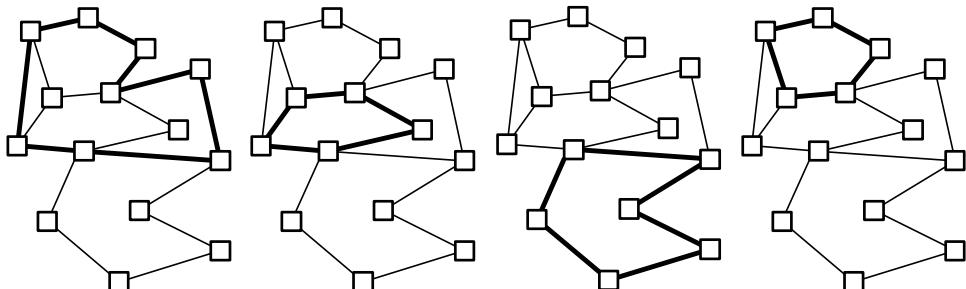
Rješenje A_{21} se pojavljuje u užem izboru za svih pet definiranih kriterija, pa ga možemo smatrati najboljim ponuđenim rješenjem. Rješenja A_{17} i A_{20} ulaze u uži izbor po tri kriterija, te stoga također mogu ući u razmatranje. Općenito, najbolja rješenja dobivena su korištenjem težinskih funkcija f_7 , f_2 i f_4 .

Za razliku od postupka A, postupak B ne uključuje analizu usmjeravanja zahtjeva unutar prstenova, već prije svega ovisi o inicijalnom usmjeravanju prometnih zahtjeva. Rezultati dobiveni korištenjem postupka B ne uključuju podatke o lokalnim zahtjevima.. U tablici 5.9 su izneseni rezultati za različita inicijalna usmjeravanja – skup prstenova koji se analiziraju ograničen je na prstenove koji sadrže od 4 do 16 čvorova.

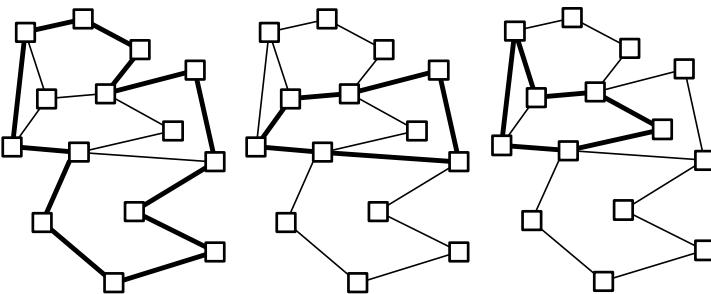
Tablica 5.9: SP1 – postupak B – rezultati 1

f	$ RCS $	$\sum w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	T_{BSHR}	N_{EC}	
USMJERAVANJE: SP - HOP										
A_{22}	g_1	4	1702	3490	6	540	0.488	0.431	2.051	18
A_{23}	g_2	5	1702	6783	10.8	580	0.251	0.183	3.985	18
A_{24}	g_3	3	1702	3372	8	348	0.505	0.448	1.981	18
A_{25}	g_4	3	1702	3434	8	362	0.496	0.439	2.018	18
USMJERAVANJE : SSP - HOP										
A_{26}	g_1	4	2491	4217	5.75	761	0.591	0.551	1.693	18
A_{27}	g_3	3	2491	4795	8	580	0.519	0.459	1.925	18
A_{28}	g_5	3	2491	4143	7	586	0.601	0.564	1.663	18
A_{29}	g_6	5	2491	4433	5.4	828	0.562	0.477	1.779	18
USMJERAVANJE: DSSP – duljina										
A_{30}	g_1	4	3092	5660	6.5	699	0.546	0.377	1.831	18
A_{31}	g_2	4	3092	9176	10.5	828	0.337	0.261	2.968	18
A_{32}	g_3	3	3092	6189	8.33	653	0.499	0.421	2.002	18
A_{33}	g_4	3	3092	5792	8	681	0.534	0.430	1.873	18
A_{34}	g_5	5	3092	5707	7	772	0.542	0.426	1.846	18

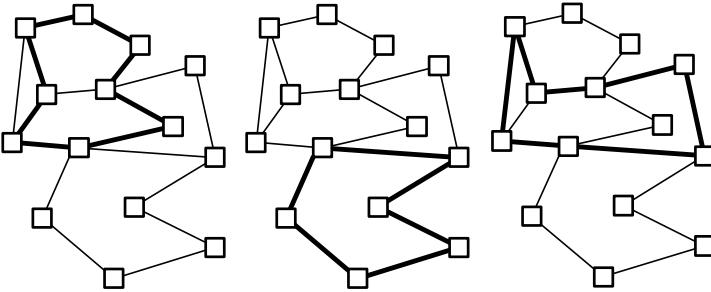
Odmah se mogu uočiti niski zahtijevani kapaciteti u rješenjima A_{22} , A_{24} i A_{25} . Rješenja A_{24} i A_{25} ujedno karakteriziraju i niski pojedinačni kapaciteti prstenova. Najveća iskoristivost mreže postiže se u rješenju A_{28} . No, uzrok povećanju iskoristivosti u tom slučaju isključivo leži u velikom povećanju opterećenja u mreži zbog alternativnih postupaka usmjeravanja! Neka od spomenutih rješenja prikazana su na slikama 5.10, 5.11 i 5.12.



Slika 5.10: SP1 – prikaz rezultata A_{22}



Slika 5.11: SP1 – prikaz rezultata A_{24}



Slika 5.12: SP1 – prikaz rezultata A_{28}

Prikazana rješenja prekrivaju sve grane u mreži. Radi se o prekrivanjima korištenjem 3 ili 4 prstena, što može predstavljati dobra rješenja. Očito je da su preklapanja prstenova smanjena, što je uvjetovano samom izvedbom postupka B – grane koje se uključe u neki od prstenova u nastavku ne utječu na težinske funkcije, pa je njihovo uključivanje u veći broj prstenova manje vjerljivo.

Budući da određivanje prstenova koji prekrivaju mrežu korištenjem postupka B isključivo ovisi o inicijalnom usmjeravanju, logično je pretpostaviti da će promjena inicijalnog usmjeravanja imati veći utjecaj na rezultate nego li u postupku A. To je lako uočljivo u tablici 5.9, koja ukazuje na povećanja zahtijevanih kapaciteta u slučaju korištenja alternativnih metoda usmjeravanja – korištenjem sekundarnih puteva prometni se zahtjevi 'raspršuju', te je ukupno opterećenje grana u mreži veće. To utječe na povećanje zahtijevanih kapaciteta. Zbog porasta zahtijevanih kapaciteta, ali i opterećenja grana u mreži, mijenja se iskoristivost mreže i prosječna iskoristivost prstenova. Iako u određenim slučajevima ti parametri ukazuju na vrlo dobra rješenja, ne može se zanemariti preveliko povećanje zahtijevanih kapaciteta.

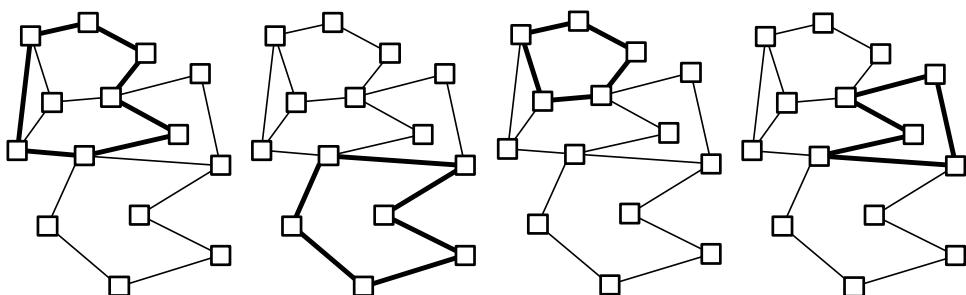
Prepostavka je da će i adaptivne metode usmjeravanja utjecati na rezultate prekrivanja. To je analizirano za oba implementirana adaptivna usmjeravanja, uz korištenje različitih početnih uvjeta koji određuju redoslijed usmjeravanja prometnih zahtjeva. Korištenjem adaptivnog usmjeravanja MxFER metodom dobivaju se rezultati prikazani u tablici 5.10. Jedini kontrolni parametar K koji može utjecati na određivanje rezultata korištenjem postupka B omogućava izvedbu prekrivanja mreže uz uvjet da su sve grane pokrivene ($K=0$), odnosno uz uvjet da se prekrivaju samo opterećene grane ($K=1$). Budući da MxFER metoda usmjerava prometne zahtjeve tako da se iskoristi što manje grana u mreži, mogu se uočiti razlike u rezultatima u odnosu na rezultate uz korištenje standardnih načina usmjeravanja. Rezultati su dobiveni uz uvjet $K=1$, dakle uz uvjet prekrivanja samo opterećenih grana mreže. Rezultati usmjeravanje MxFER metodom ovise o redoslijedu usmjeravanja prometnih zahtjeva. U tablici su dani rezultati uz korištenje različitih redoslijeda usmjeravanja prometnih zahtjeva. U prvom dijelu zahtjevi se obrađuju od najvećeg prema najmanjem, dok se u nastavku zahtjevi sortiraju od najduljeg prema najkraćem (kako bi se već prvim zahtjevom iskoristilo što više grana i time sve ostale zahtjeve pokušalo usmjeriti iskorištenim granama). Duljine puteva prometnih zahtjeva prema kojima se obavlja sortiranje određuju se postupcima usmjeravanja kako je naznačeno u zaglavljima u tablici.

Tablica 5.10: SP1 – postupak B – rezultati 2

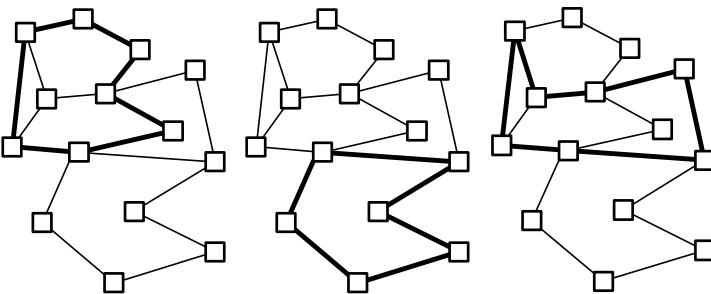
<i>f</i>	$ RCS $	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	N_{EC}
USMJERAVANJE: MxFER – od najvećeg zahtjeva								
A_{35}	g_1	4	2322	6059	5.5	1073	0.383	0.376
A_{36}	g_2	4	2322	10507	11.2	877	0.221	0.163
A_{37}	g_3	2	2322	6002	9	622	0.387	0.323
A_{38}	g_4	3	2322	5692	7	644	0.408	0.301
USMJERAVANJE: MxFER – od SP HOP najduljeg								
A_{39}	g_1	4	2497	5504	5.8	879	0.454	0.343
A_{40}	g_3	3	2497	5397	8	483	0.463	0.317
USMJERAVANJE: MxFER – od SSP HOP najduljeg								
A_{41}	g_1	4	2253	5404	5.8	897	0.417	0.384
A_{42}	g_3	2	2253	5820	9	623	0.387	0.376
A_{43}	g_4	3	2253	4699	7	621	0.479	0.391
USMJERAVANJE: MxFER – od DSSP najduljeg								
A_{44}	g_1	4	2161	4332	5.8	703	0.499	0.381
A_{45}	g_3	3	2161	4641	7.3	588	0.466	0.390
A_{46}	g_4	3	2161	4353	6.7	654	0.496	0.440
A_{47}	g_5^*	3	2140	4486	6.3	692	0.477	0.418

*uz sekundarni kriterij usmjerenja

Iz rezultata je vidljivo da uz korištenje MxFER usmjerenja dolazi do povećanja zahtjeva za kapacitetima u mreži. S druge strane, smanjuje se broj grana koje se koriste. Ovi pomalo kontradiktorni rezultati mogu se jednostavno objasniti – usmjerenje prometnih zahtjeva obavlja se s ciljem smanjenja broja opterećenih grana – kapaciteti prstenova rastu, a broj grana koje se uključuju u prekrivanje se smanjuje. Utjecaj smanjenja broja grana je na mreži s malim broj grana poput ove zanemariv, a kapaciteti ovise prije svega o prometnim zahtjevima. Na slikama su prikazani rezultati A_{44} i A_{45} .



Slika 5.13: SP1 – prikaz rezultata A_{44}



Slika 5.14: SP1 – prikaz rezultata A_{45}

Rezultati prekrivanja su vrlo slični prije prikazanima. No, same karakteristične vrijednosti rezultantnih mreža se razlikuju, prije svega zbog velikog utjecaja načina usmjeravanja na postupak B. Korištenjem MiFER usmjeravanja dobivaju se rezultati prekrivanja koji uključuju sve grane u mreži, a broj prstenova je u pravilu veći nego u rezultatima uz korištenje MxFER usmjeravanja. Nekoliko rezultata je prikazano u tablici 5.11.

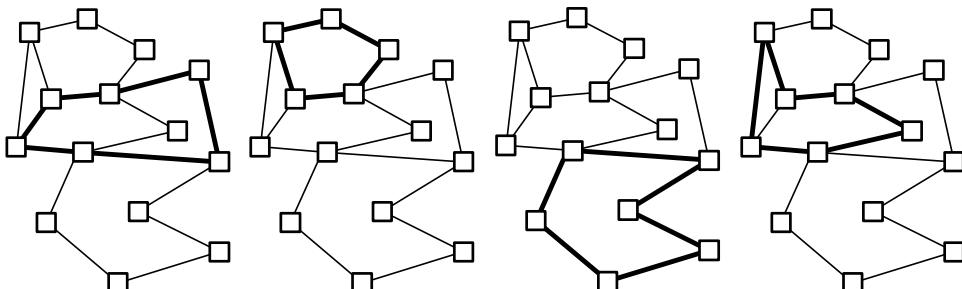
Tablica 5.11: SP1 – postupak B – rezultati 3

f	$ RCS $	$\sum w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	N_{EC}
USMJERAVANJE: MiFER – od najmanjeg zahtjeva								
A_{48}	g_4	3	1716	3522	8	363	0.487	0.419
A_{49}	g_4^*	3	1718	3390	8	354	0.507	0.424
USMJERAVANJE: MiFER – od najvećeg zahtjeva								
A_{50}	g_4	4	1882	3327	6.75	405	0.566	0.438
A_{51}	g_4^*	3	1965	3756	8.33	410	0.524	0.445
USMJERAVANJE: MiFER – od SP najduljeg								
A_{52}	g_1	4	1906	3135	5.75	542	0.608	0.599

*uz sekundarni kriterij usmjeravanja

Očito je da usmjeravanja MiFER metodom daju vrlo dobre rezultate. Iako je bilo očekivano da će MxFER metoda nuditi bolje rezultate zbog manjeg broja korištenih grana, pokazuje se da na primjeru testne mreže 1 bolje rezultate daje MiFER.

Prstenovi koji su odabrani za prekrivanje mreže u A_{52} prikazani su na slici 5.15.



Slika 5.15: SP1 – prikaz rezultata A_{52}

Izbor najboljih rezultata dobivenih postupkom B prema ključnim kriterijima prikazan je u tablici. Usporedba rezultata sa onima dobivenim postupkom A se može izvesti, no bitno je još jednom naglasiti razliku u načinu obrade prometnih zahtjeva u tim postupcima. Prema dostupnim rezultatima za postupak B nije moguće utvrditi udio lokalnog prometa u mreži, pa usporedba bez tog kriterija nije potpuna.

Tablica 5.12: SP1 – postupak B – najbolja rješenja

Kriterij	Najbolja rješenja
Najmanji zahtijevani BSHR kapaciteti	A_{52} (3135), A_{50} (3327), A_{24} (3372), A_{49} (3390)
Najveća iskoristivost mreže	A_{52} (0.608), A_{29} (0.601), A_{26} (0.591), A_{50} (0.566)
Najveća prosječna iskoristivost prstenova	A_{52} (0.599), A_{50} (0.566), A_{29} (0.564), A_{26} (0.551)
Najmanji broj prstenova	A_{47} (2), A_{29} (3), A_{33} (3), A_{24} (3)

Budući da je na prvom mjestu prema tri kriterija kao najbolje prekrivanje dobiveno postupkom B odabire se A_{52} . Usporedi li se to prekrivanje s prekrivanjima koja su predložena kao najbolja za postupak A odmah se može uočiti razlika – prekrivanja dobivena postupkom A uvek sadrže bar jedan veliki prsten kojim prometuje velik dio prometnih zahtjeva, dok se u A_{52} radi o nekoliko manjih prstenova. Dobri rezultati se dobivaju korištenjem težinskih funkcija g_1 , g_4 i g_6 . Sličnost funkcija g_1 , g_4 i g_5 često rezultira jednakim rezultatima, no to prije svega ovisi o opterećenju grana u prstenu (g_3).

5.4.2. Testna mreža 1 - studijski primjer 2

Testiranje se vrši nad identičnom mrežom kao i u SP1, ali uz dvostruko veći broj prometnih zahtjeva. Definirana ograničenja u postupcima su identična kao i u *studijskom primjeru 1*: prstenovi su ograničeni na veličine od 4 do 16 čvorova, inicijalno usmjeravanje prometnih zahtjeva obavlja se najkraćim putevima, te se koristi sekundarni kriterij odabira manjih prstenova u slučaju identičnih vrijednosti težinskih funkcija.

Izneseni su samo bitniji rezultati dobivenim korištenjem oba postupka. U tablici 5.13 su prikazani rezultati postupka A, a u tablici 5.14 postupka B.

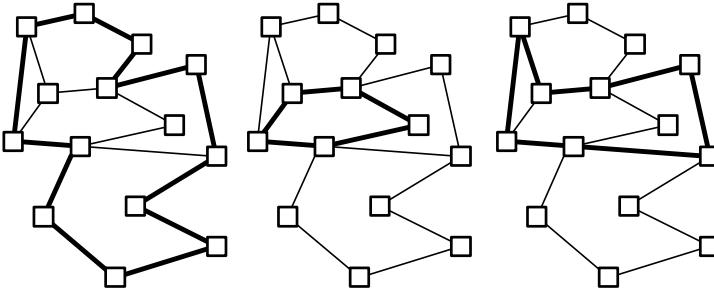
Tablica 5.13: SP2 – postupak A – rezultati

<i>f</i>	RCS	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	<i>L</i>	C_{BSHR}	μ_{BSHR}	μ_R	N_{EC}
SP HOP usmjeravanje									
<i>B</i> ₁	<i>f</i> ₁	5	3294	6489	5.4	38%	1202	0.508	0.477
<i>B</i> ₂	<i>f</i> ₂	5	3294	6174	5.4	40%	1139	0.533	0.478
<i>B</i> ₃	<i>f</i> ₄	5	3294	5717	5.4	42%	1039	0.576	0.497
Obrada lokalnih zahtjeva									
<i>B</i> ₄	<i>f</i> ₃ *	4	3920	5692	6.8	76%	661	0.689	0.437
<i>B</i> ₅	<i>f</i> ₄	3	3920	5063	8	76%	516	0.774	0.588
<i>B</i> ₆	<i>f</i> ₇	4	3920	5172	6.8	76%	557	0.758	0.475
Obrada lokalnih zahtjeva, zabrana međuprstenastih u prvom prstenu									
<i>B</i> ₇	<i>f</i> ₂	5	4221	7817	7.8	98%	686	0.540	0.300
<i>B</i> ₈	<i>f</i> ₃	5	4092	7532	6.8	88%	728	0.543	0.422
<i>B</i> ₉	<i>f</i> ₆ *	5	3379	5461	5.6	44%	983	0.618	0.605

*drugi najbolji prsten u prvoj iteraciji

Od rezultata dobivenih postupkom A izdvojeni su najbolji iz svake grupe. Rezultati u prvoj grupi dobiveni su bez dodatne analize prometnih zahtjeva uz usmjeravanje prometnih zahtjeva najkraćim putevima. Iz te grupe odabrano je rješenje B_3 – iako rješenje ne uključuje usmjeravanje zahtjeva unutar prstenova, iskoristivost mreže i prosječna iskoristivost prstenova su relativno visoke (u odnosu na SP1). Prstenovi odabrani u B_3 identični su onim izabranim u A_2 . Sam redoslijed uključivanja prstenova u rješenje je drugačiji, ali za prikaz rezultata može poslužiti slika 5.3.

Druga grupa rezultata dobivena je uz detaljnu analizu zahtjeva i usmjeravanje lokalnih zahtjeva unutar prstenova. To rezultira povećanjem udjela lokalnih zahtjeva u prometu mreže, te smanjenjem zahtijevanih kapaciteta u mreži, unatoč povećanju opterećenja grana. Odabранo je rješenje B_5 - rezultat čine tri prstena prikazana na slici 5.16.



Slika 5.16: SP2 – prikaz rezultata B_5

Uz onemogućeno usmjeravanje međuprstenastih zahtjeva u prvom odabranom prstenu, odabранo je prekrivanje B_9 . Rezultat sadrži 5 prstenova odabranih na neznatno drugačiji način u odnosu na A_2 .

Rezultat B_5 predstavlja najbolji izbor obzirom na većinu definiranih kriterija – mali broj prstenova, najniži prosječni kapacitet prstena, najniži zahtijevani kapaciteti, najbolja iskoristivost mreže i prosječna iskoristivost prstenova.

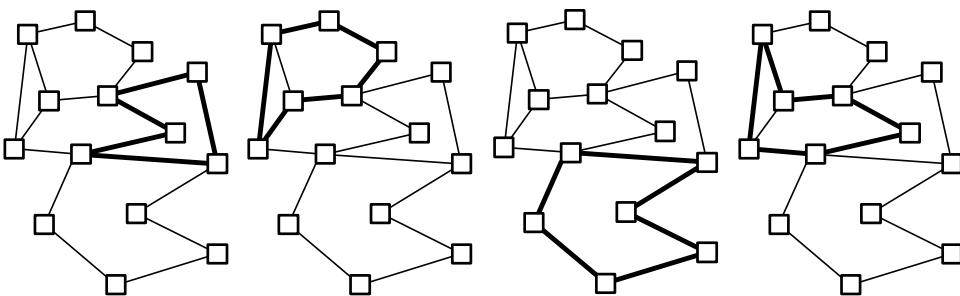
Rezultati prekrivanja korištenjem postupka B za studijski primjer 2 dani su u tablici 5.14.

Tablica 5.14: SP2 – postupak B - rezultati

f	$ RCS $	$\sum w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	N_{EC}
USMJERAVANJE: SP HOP								
B_{10}	g_1	4	3294	5689	5.8	998	0.579	0.559
B_{11}	g_4	3	3294	5763	7.3	753	0.572	0.543
B_{12}	g_5	4	3294	5737	6.5	794	0.574	0.461
USMJERAVANJE: SSP HOP								
B_{13}	g_5	4	4870	7788	6.5	1224	0.625	0.555
USMJERAVANJE: MiFER – od SP najduljeg								
B_{14}	g_1	4	3435	5846	5.8	1022	0.588	0.560
B_{15}	g_3	3	3435	6662	8	745	0.516	0.467
B_{16}	g_4	3	3435	5967	8	698	0.576	0.497
USMJERAVANJE: MiFER – od najmanjeg								
B_{17}	g_1	4	3309	5683	5.8	997	0.582	0.561
USMJERAVANJE: MiFER – od najvećeg								
B_{18}	g_1	4	3432	5965	5.8	1044	0.575	0.561
B_{19}	g_4	3	3432	5877	8	695	0.584	0.506
B_{20}	g_4^*	3	3528	5837	8	695	0.600	0.515

*uz sekundarni kriterij usmjeravanja

Kao najbolji od rezultata postupka B odabrani su B_{10} zbog niskih zahtijevanih kapaciteta i dobre iskoristivosti mreže i prstenova, B_{17} zbog najnižih zahtijevanih kapaciteta, te B_{20} zbog manjeg broja odabranih prstenova, te najnižeg prosječnog kapaciteta prstenova u rješenju. Prekrivanje B_{10} je prikazano na slici 5.17. B_{17} prekriva mrežu korištenjem identičnih prstena, ali uz nešto izmijenjeni redoslijed.



Slika 5.17: SP2 – prikaz rezultata B_{10}

U ovom studijskom primjeru uočljivo je da su rezultati dobiveni postupkom B prema parametru zahtijevanih kapaciteta lošiji od rezultata dobivenih postupkom A. Za studijski primjer 1 odnos je bio drugačiji, te su rezultati postupka B bili bolji. Može se zaključiti da postupak A daje bolje nešto bolje rezultate uz povećan broj prometnih zahtjeva – uzrok tome leži u dodatnoj analizi prometnih zahtjeva koja se provodi u postupku A.

5.4.3. Testna mreža 1 – rezultati matematičkog programa

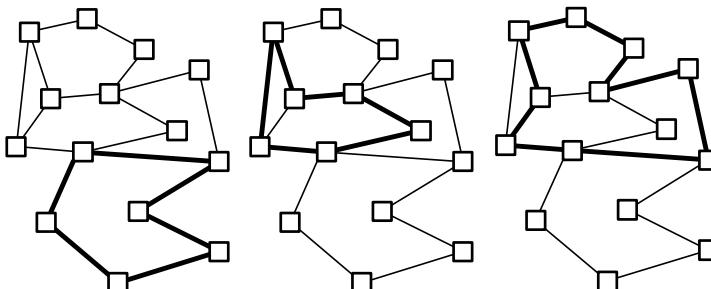
Jedini parametar kojim je moguće utjecati na rezultate matematičkog programa za određivanje optimalnog prekrivanja mreže prstenovima jest težina prstenova koji mogu ući u rješenje. Težine prstenova u testiranjima su određene težinskim funkcijama g_x koje se koriste u postupku B. Incijalno usmjeravanje prometnih zahtjeva koje se koristi jest usmjeravanje najkraćim putevima uz glavni kriterij broj *hop-ova*.

Budući da je analizom rezultata u studijskim primjerima 1 i 2 uočeno da najbolje rezultate daje korištenje težinskih funkcija g_1 , g_3 i g_6 , u analizi rezultata matematičkog programa koriste se samo te funkcije. Rezultati se dobivaju u obliku vektora X koji označava prstenove koji su izabrani u rješenje. Prikaz rezultata – broj prstenova i popis prstenova koji su uključeni u rješenje – je dan u tablici 5.15.

Tablica 5.15: Rezultati matematičkog programa na testnoj mreži 1

Težinska funkcija	25 prometnih zahtjeva	50 prometnih zahtjeva
g_1	3 prstena – R_4, R_{10}, R_{15}	3 prstena – R_{10}, R_{13}, R_{15}
g_3	3 prstena – R_4, R_7, R_{13}	3 prstena – R_4, R_7, R_{13}
g_5	4 prstena – R_3, R_4, R_7, R_{10}	3 prstena – R_4, R_7, R_{13}

Iako su rezultati matematičkog programa teško usporedivi s rezultatima dobivenim postupcima A i B, nije naodmet proučiti prekrivanja koja se dobivaju kao rezultati. Grafička interpretacija prekrivanja koje se tri puta pojavilo kao rezultat dana je na slici 5.18.



Slika 5.18: Primjer rezultata matematičkog programa na testnoj mreži 1

5.4.4. Testna mreža 1 - zaključak

Rekapitulacijom rezultata za studijske primjere 1 i 2 izvršene nad testnom mrežom 1, mogu se uočiti sličnosti u rješenjima prekrivanja za oba testa. Ipak, prisutne su manje razlike u odabiru težinskih funkcija koje dovode do najboljih rješenja – u postupku A najbolja rješenja za $SP1$ dobivena su korištenjem funkcija f_2 , f_4 i f_7 , dok je u $SP2$ najbolje rješenje dobiveno korištenjem funkcije f_4 . Očito je da težinske funkcije ovisne o odnosu lokalnih i meduprstenastih prometnih zahtjeva (f_2 , f_3 , f_7) ne daju zadovoljavajuća rješenja u primjerima s velikim brojem prometnih zahtjeva. S druge strane, težinske funkcije ovisne o iskoristivosti prstenova (f_4 , f_6) rezultiraju dobrim rješenjima neovisno o broju prometnih zahtjeva. Postupak B daje u oba testa najbolja rješenja uz korištenje težinske funkcije g_1 – funkcija definira težinu prstena proporcionalno iskoristivosti, što je ekvivalentno primjerni funkciji f_4 u postupku A.

Kao što je bilo očekivano, postoji utjecaj prometnih zahtjeva na rezultate – rješenja prekrivanja koja se dobivaju u $SP2$ nisu uvijek jednaka rješenjima dobivenim $SP1$. To je u potpunosti realno i ne predstavlja nedostatak. Analizom rezultata može se uočiti da u većini prekrivanja postoji ponavljanje čvorova i grana u različitim prstenovima. Na primjeru male mreže kao što je testna mreža 1 nije teško zaključiti da bi za prekrivanje bila dovoljna dva prstena. U $SP2$ nijedna od analiziranih metoda nije ponudila rješenje koje bi uključivalo dva prstena. Dodatnom analizom primjene najboljih prekrivanja za $SP1$, a uz obradu prometnih zahtjeva iz $SP2$ i obratno mogao bi se donijeti točniji zaključak o prednostima i nedostacima postupaka. Uz variranje prometnih zahtjeva u $SP1$ (ali bez povećanja broja zahtjeva kao kod $SP2$) mogla bi se provesti dodatna analiza skalabilnosti metode za prekrivanje mreže prstenovima.

5.4.5. Testna mreža 2 – studijski primjeri 3 i 4

Studijski primjer 3 se provodi nad testnom mrežom 2. Budući da je broj mogućih prstenova u toj mreži vrlo velik (4317), testiranja su izvršena uz definiranje 150 prstenova veličine od 4 do 16 čvorova. Da bi se proanalizirao utjecaj mogućnosti odabira iz većeg skupa prstenova, pojedina testiranja su vršena uz definiranje 300 i 600 prstenova veličine od 4 do 16 čvorova. Zbog velikog broja definiranih prstenova i prometnih zahtjeva, postupak određivanja rezultata je vremenski zahtjevniji nego za slučaj manjih mreža. Ta povećana zahtjevnost posebice je uočljiva u izvođenju složenijeg postupka A. Zbog toga se kod proračuna vodilo računa i o vremenu izvođenja, pa je taj podatak iznesen u tablicama (stupac T).

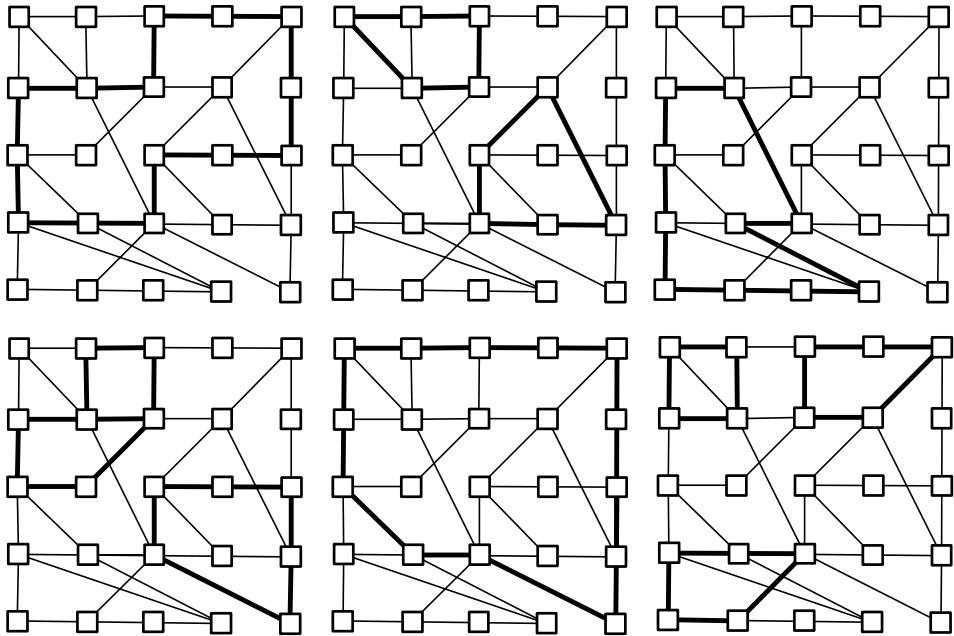
Testiranje je provedeno korištenjem svih metoda za oba postupka, no izneseni su samo najznačajniji rezultati. Naglašeno je korištenje težinskih funkcija koje su za testnu mrežu 1 polučile najbolje rezultate.

Neki od rezultata dobivenih postupkom A uz definiranih 150 prstenova prikazani su u tablici 5.16. Primarno su analizirani rezultati koji koriste inicijalno usmjeravanje najkraćim putevima, jer se je u prethodnim testovima ($SP1$ i $SP2$) pokazalo da je utjecaj inicijalnog usmjeravanja u postupku A uz dodatnu obradu lokalnih zahtjeva malen.

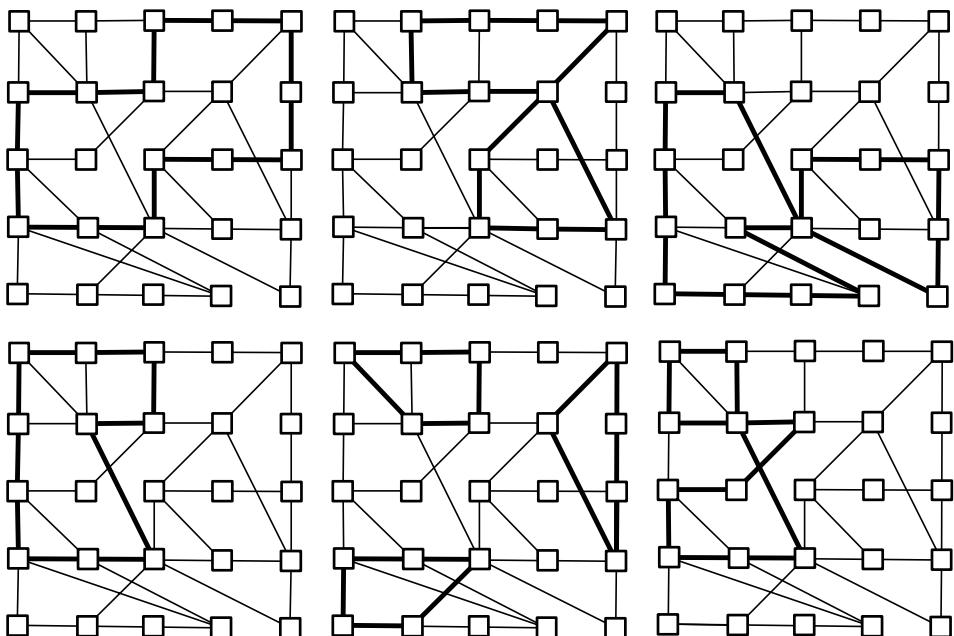
Tablica 5.16: $SP3$ – postupak A – rezultati za 150 prstenova

	f	$ RCS $	$\Sigma w(e)$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}
C_1	f_2	12	10438	23988	7.7	63%	2316	0.435	0.320	~800 ms	38
C_2	f_3	8	10624	20834	8	61%	1676	0.510	0.431	~800 ms	38
C_3	f_4	11	9956	15675	6.9	49%	2080	0.635	0.495	~800 ms	38
C_4	f_6	11	10014	15116	7.1	52%	1858	0.662	0.496	~800 ms	37

Od rezultata dobivenih uz definiranih 150 prstenova izdvojeni su C_4 i C_3 zbog najnižih zahtijevanih kapaciteta i visoke iskoristivosti. Prekrivanje C_3 je prikazano na slici 5.19, a prekrivanje C_4 na slici 5.20. Redoslijed određivanja prstenova nije u potpunosti poštivan – na nekim od prikaza mreže naznačeno je više resultantnih prstenova.



Slika 5.19: SP3 – prikaz rezultata C_3



Slika 5.20: SP3 – prikaz rezultata C_4

Rezultati pokazuju da je prekrivanje moguće izvesti relativno malim brojem prstenova. Rješenja se razlikuju, no neki od prstenova se javljaju u oba prekrivanja.

U tablicama 5.17 i 5.18 su izdvojeni rezultati dobiveni uz definiranih 300 i 600 prstenova. Rezultati su dobiveni uz iste uvjete kao i oni u tablici 5.16. Za rezultate dobivene istim težinskim funkcijama naznačene su promjene karakterističnih parametara.

Iz rezultata je vidljiv jedan od osnovnih nedostataka korištenja heurističkih algoritama – promjena broja prstenova koji mogu ući u rješenje omogućila je povećanje broja mogućih rješenja, ali nije dovela do boljih rezultata u svim slučajevima. Kako se postupak svodi na izvođenje pohlepne metode određivanja prstenova, postoji mogućnost da se izgube dobra rješenja dobivena primjenom postupka nad manjim inicijalnim skupom mogućih čvorova. Ipak, može se uočiti da se primjenom

težinskih funkcija f_4 i f_6 (ovise o iskoristivosti prstenova) dobivaju relativno dobra rješenja koja karakterizira manji broj prstenova za prekrivanje u odnosu na rješenje dobivena uz 150 prstenova, te povećanje prosječne veličine prstenova.

Tablica 5.17: SP3 – postupak A – rezultati za 300 prstenova

f	$ RCS $	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}	
C_5	f_3	8•	10776↑	23437↑	9.6↑	60%↓	1841↑	0.460↓	0.321↓	< 3 s	38
C_6	f_4	8↓	9235↓	15321↓	7.1↑	41%↓	1887↓	0.602↓	0.493↓	< 3s	38
C_7	f_6	10↓	9441↓	15391↑	7.3↑	42%↓	1934↓	0.613↓	0.529↑	< 3s	39↑
C_8	f_7	10	11718	24820	8.1	64%	1950	0.472	0.359	< 3s	38

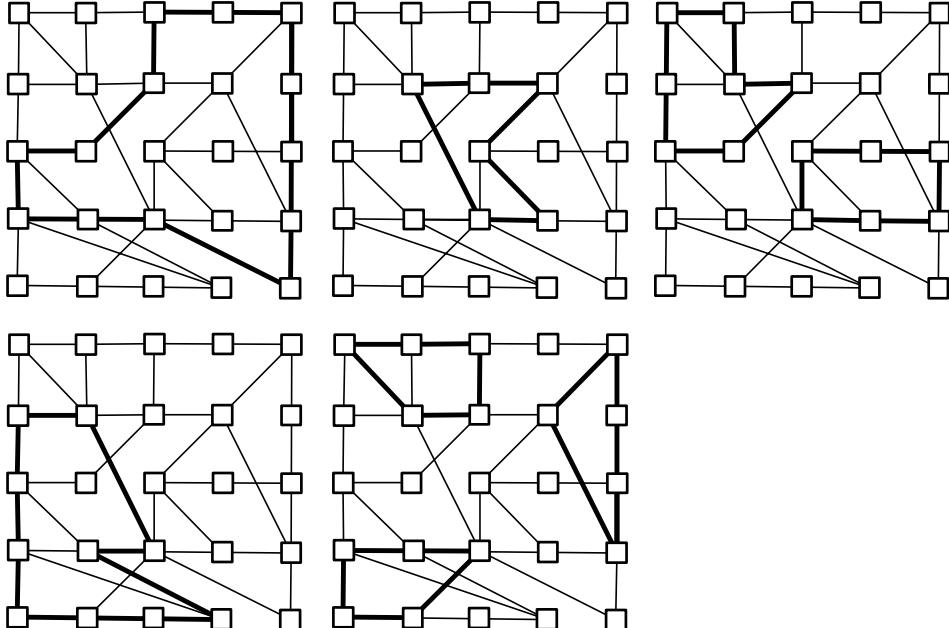
• ista vrijednost ↑ porast ↓ smanjenje

Tablica 5.18: SP3 – postupak A – rezultati za 600 prstenova

f	$ RCS $	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	L	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}	
C_9	f_3	6↓	11317	20504↓	11.7↑	66%↑	1386↓	0.552↑	0.426↓	< 5 s	38
C_{10}	f_3^*	6↓	11672	20286↓	11.3↑	65%↑	1374↓	0.575↑	0.381↓	< 5 s	38
C_{11}	f_4	10↓	11346	14615↓	8.2↑	71%↑	1360↓	0.776↑	0.570↑	< 5 s	39
C_{12}	f_6	10↓	11498	14780↓	8.2↑	64%↑	1462↓	0.772↑	0.618↑	< 5 s	39
MxFER usmjeravanje od najvećeg zahtjeva + obrada lokalnih zahtjeva											
C_{13}	f_4	7	12574	17412	8.7	68%	1596	0.722	0.507	< 5 s	36

* drugi najbolji prsten u prvoj iteraciji • ista vrijednost ↑ porast ↓ smanjenje

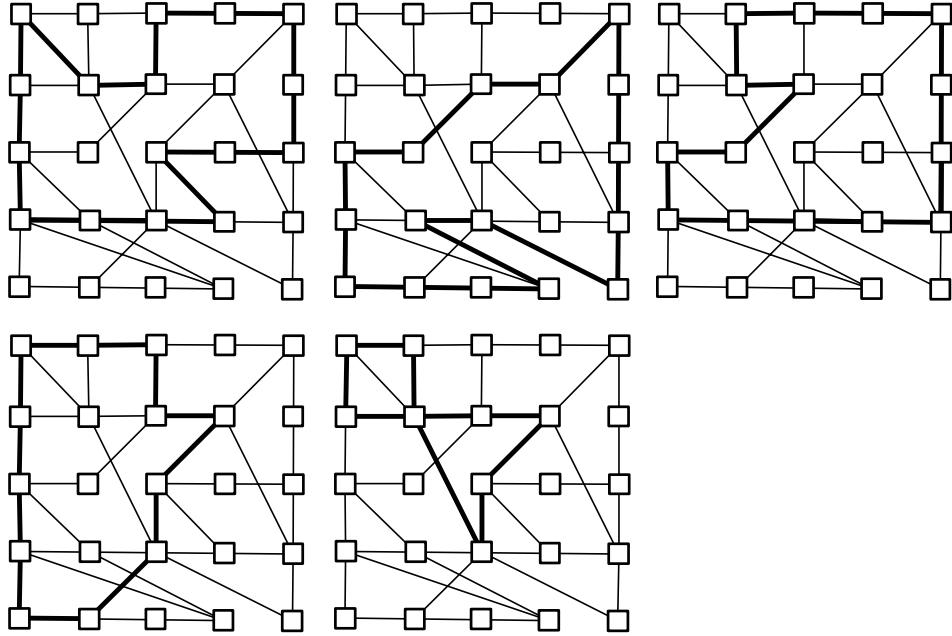
Iz tablice 5.17 izdvojeno je i prikazano na slici 5.21 prekrivanje C_6 zbog najnižih zahtijevanih kapaciteta i dobrih ostalih svojstava (mali broj prstenova, visoka iskoristivost). Kao što je i očekivano za postupak A, prvi odabrani prsten sadrži veći broj čvorova, te prenosi velik broj prometnih zahtjeva.



Slika 5.21: SP3 – prikaz rezultata C_6

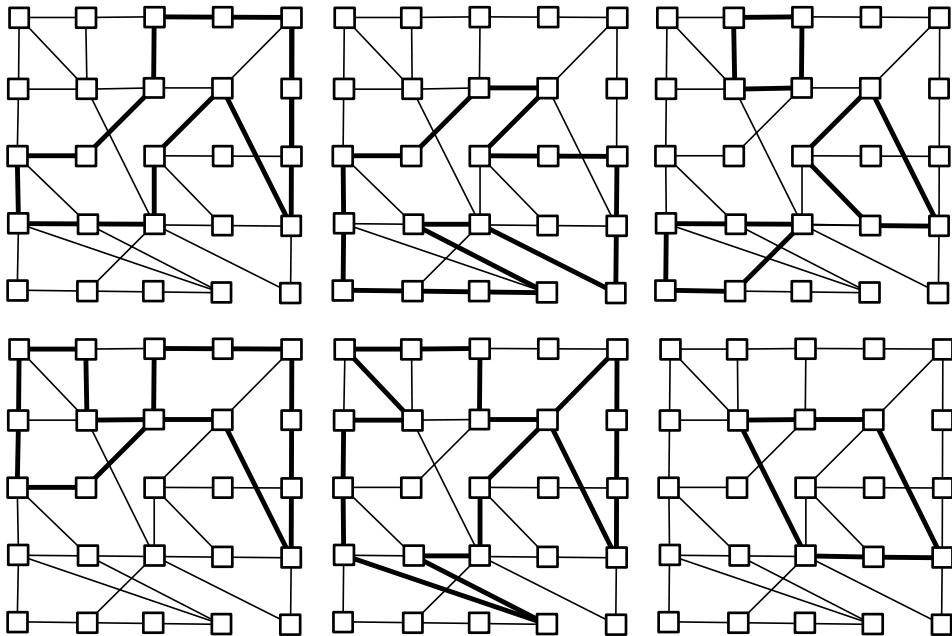
Od prekrivanja dobivenih analizom 600 prstenova izabrana su rješenja C_{10} i C_{11} . C_{10} je prikazan na slici 5.22 - sadrži 6 prstenova, od čega su čak četiri vrlo velika. Iako su prosječni kapaciteti

prstenova mali, iskoristivost takvih prstenova je niska te se gotovo sigurno može reći da C_{10} ne bi bio primjenjen u praksi.



Slika 5.22: SP3 – prikaz rezultata C_{10}

Prekrivanje dobiveno rezultatom C_{11} je prikazano na slici 5.23. Rezultat C_{11} predstavlja najbolje rješenje u pogledu ukupno zahtijevanih kapaciteta u mreži. Sastoji se od 10 prstenova, od čega su tri prstena nešto veća.



Slika 5.23: SP3 – prikaz rezultata C_{11}

Rezultati postupka nad skupom od 600 prstenova pokazuju poboljšanja u odnosu na rezultate uz definiranih 150 prstenova. Broj prstenova u rezultatima se smanjuje, no prosječna veličina prstenova u rješenju je veća. Opterećenja prstenova se smanjuju, a povećava se iskoristivost mreže. Rezultati su bolji nego oni dobiveni na skupu od 300 prstenova, što još jednom ukazuje na nepravilnosti koje su posljedica načina funkcioniranja primjenjenih metoda. Kao i u rezultatima

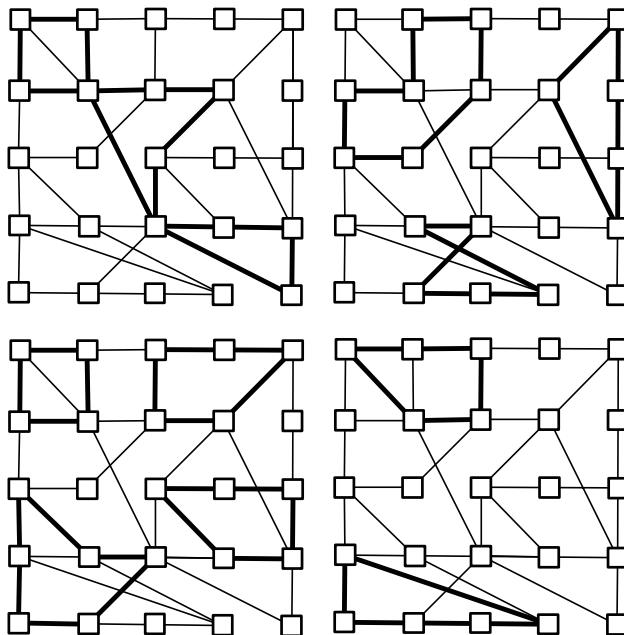
nad testnom mrežom 1, pokazalo se da težinske funkcije ovisne o iskoristivosti prstenova daju najbolja rješenja.

Tablica 5.19 prikazuje rezultate dobivene postupkom B uz enumeracijom određenih 150 prstenova. Dani su rezultati uz različite modele usmjeravanja – osim najkraćim putevima, usmjeravanje je za potrebe testiranja izvedeno i korištenjem adaptivnih metoda.

Tablica 5.19: SP3 – postupak B – rezultati za 150 prstenova

f	$ RCS $	$\sum w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}	
USMJERAVANJE: SP - HOP										
C_{14}	g_1	11	8624	18007	5.1	3462	0.479	0.454	$\sim 200\ ms$	39
C_{15}	g_3	7	8624	25603	8.7	2180	0.337	0.353	$\sim 200\ ms$	39
C_{16}	g_4	8	8624	20651	7.9	2226	0.418	0.340	$\sim 200\ ms$	39
C_{17}	g_5	9	8624	20060	6.9	2718	0.430	0.351	$\sim 200\ ms$	39
USMJERAVANJE : MiFER, od najmanjeg, prema duljini grana										
C_{18}	g_1	12	8693	18142	5.1	3487	0.479	0.437	$\sim 200\ ms$	40
C_{19}	g_4	9	8693	20726	7.6	2240	0.419	0.325	$\sim 200\ ms$	40
C_{20}	g_5	10	8693	20205	6.7	2737	0.430	0.336	$\sim 200\ ms$	40
USMJERAVANJE : MiFER, od SSP HOP najdužeg										
C_{21}	g_1	12	8777	15909	5.1	3108	0.552	0.448	$\sim 700\ ms$	40
USMJERAVANJE : MxFER, od najvećeg, prema duljini grana										
C_{22}	g_1	8	10348	24022	5.5	4569	0.431	0.408	$\sim 200\ ms$	35
C_{23}	g_5	7	10348	26967	6.6	3869	0.384	0.366	$\sim 200\ ms$	33

Iz dobivenih rezultata izdvojen je C_{21} , koji predstavlja prekrivanje uz najniže zahtijevane kapacitete. Grafički prikaz tog prekrivanja dan je na slici 5.24.



Slika 5.24: SP3 – prikaz rezultata C_{21}

Isti postupci, ali uz određenih 300 i 600 prstenova prikazani su u tablicama 5.20 i 5.21. Na primjeru povećanja broja prstenova može se lako uočiti problem karakterističan za korištenje pohlepnih metoda – u nekim slučajevima rješenja dobivena na većem broju prstenova mogu biti lošija od onih

dobivenih na manjem skupu izvedivih prstenova. U tablici 5.20 označena su pozitivna ili negativna kretanja rezultata u odnosu na rezultate nad 150 prstenova.

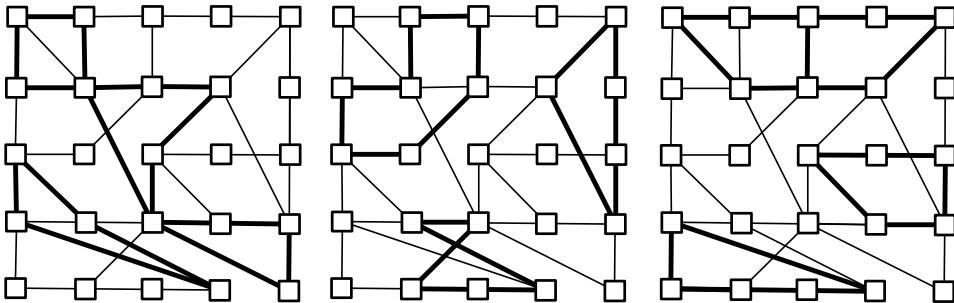
Tablica 5.20: SP3 – postupak B – rezultati za 300 prstenova

f	$ RCS $	$\sum w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}	
USMJERAVANJE: SP - HOP										
C_{24}	g_1	$10 \downarrow$	8624	$17646 \downarrow$	$6.3 \uparrow$	$2628 \downarrow$	$0.489 \uparrow$	$0.382 \downarrow$	$\sim 800 \text{ ms}$	$40 \uparrow$
C_{25}	g_3	$5 \downarrow$	8624	$28954 \uparrow$	$14.2 \uparrow$	$1977 \downarrow$	$0.298 \downarrow$	$0.246 \downarrow$	$\sim 800 \text{ ms}$	$39 \bullet$
C_{26}	g_4	$6 \downarrow$	8624	$20594 \downarrow$	$9.3 \uparrow$	$1972 \downarrow$	$0.419 \uparrow$	$0.395 \uparrow$	$\sim 800 \text{ ms}$	$39 \bullet$
C_{27}	g_5	$8 \downarrow$	8624	$18702 \downarrow$	$7.1 \uparrow$	$2509 \downarrow$	$0.461 \uparrow$	$0.386 \uparrow$	$\sim 800 \text{ ms}$	$39 \bullet$
USMJERAVANJE : MIFER, od najmanjeg, prema duljini grana										
C_{28}	g_1	$10 \downarrow$	8693	$17722 \downarrow$	$6.3 \uparrow$	$2646 \downarrow$	$0.491 \uparrow$	$0.387 \downarrow$	$\sim 800 \text{ ms}$	$40 \bullet$
C_{29}	g_4	$8 \downarrow$	8693	$20839 \downarrow$	$8 \uparrow$	$2069 \downarrow$	$0.418 \downarrow$	$0.330 \uparrow$	$\sim 800 \text{ ms}$	$40 \bullet$
C_{30}	g_5	$8 \downarrow$	8693	$19050 \downarrow$	$7.3 \uparrow$	$2421 \downarrow$	$0.456 \uparrow$	$0.415 \uparrow$	$\sim 800 \text{ ms}$	$40 \bullet$
USMJERAVANJE : MxFER, od najvećeg, prema duljini grana										
C_{31}	g_1	$11 \downarrow$	8777	$16167 \uparrow$	$4.9 \downarrow$	$3222 \uparrow$	$0.543 \uparrow$	$0.480 \uparrow$	$< 2 \text{ s}$	$40 \bullet$
C_{32}	g_1^*	$11 \downarrow$	8855	$17298 \uparrow$	$4.9 \downarrow$	$3450 \uparrow$	$0.512 \downarrow$	$0.484 \uparrow$	$< 2 \text{ s}$	$40 \bullet$
USMJERAVANJE : MxFER, od najvećeg, prema duljini grana										
C_{33}	g_1	$7 \downarrow$	11976	$26706 \uparrow$	$6.4 \uparrow$	$3748 \downarrow$	$0.448 \uparrow$	$0.377 \downarrow$	$\sim 800 \text{ ms}$	$35 \bullet$
C_{34}	g_5	$7 \bullet$	11976	$26319 \downarrow$	$6 \downarrow$	$4288 \uparrow$	$0.455 \uparrow$	$0.412 \uparrow$	$\sim 800 \text{ ms}$	$35 \uparrow$

*uz sekundarni kriterij usmjerenja • ista vrijednost ↑ porast ↓ smanjenje

Može se uočiti da je većina prekrivanja uz skup od 300 prstenova realizirana s manjim brojem prstenova nego li uz definiran početni skup od 150 prstenova. Iako je u većini slučajeva zahtijevani kapacitet mreže smanjen, ipak se ne dobiva rješenje sa zahtijevanim kapacitetima nižim od onih u rješenju C_{21} . Opterećenja mreže se u većini rješenja povećavaju, kao i prosječna opterećenja prstenova. Broj grana koje su prekrivene ostaje nepromijenjen ili se povećava. Prekrivanja koja uključuju mali broj prstenova karakteriziraju veći prstenovi (npr. C_{25} prekrije mrežu korištenjem samo 5 prstenova, ali uz prosječno 14.2 čvora po prstenu). Takva rješenja najčešće se neće primjenjivati u stvarnim mrežama.

Od prikazanih, rješenje C_{31} zahtijeva najmanje instaliranih kapaciteta u mreži, uz dobru iskoristivost mreže. Prikaz prstenova dan je na slici 5.25. Vidljivo je da se radi o malim prstenovima (prosječna veličina ispod 5 čvorova).



Slika 5.25: SP3 – prikaz rezultata C_{31}

Tablica 5.21 prikazuje rezultate dobivene nad skupom od 600 ponuđenih prstenova. Slični kao i na rezultatima nad 300 prstenova, vidljivo je da dolazi do smanjenja broja prstenova za prekrivanje, dok zahtijevani kapaciteti u mreži rastu. Porast prosječne iskoristivosti prstena bilježi se u

rješenjima u kojima se koriste težinske funkcije g_1 , g_4 i g_5 koje težinu prstena definiraju upravo prema iskoristivosti.

Tablica 5.21: SP3 – postupak B – rezultati za 600 prstenova

f	$ RCS $	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}	
USMJERAVANJE: SP - HOP										
C_{35}	g_3	4↓	8624	25704↑	14.8↑	1696↓	0.336↓	0.295↓	< 2 s	39
USMJERAVANJE : MiFER, od najmanjeg, prema duljini										
C_{36}	g_1	11↓	8693	19231↑	5.6↑	3195↓	0.451↓	0.415↓	< 2 s	40
C_{37}	g_4	7↓	8693	21192↑	8.9↑	1977↓	0.411↓	0.358↑	< 2 s	40
C_{38}	g_5	7↓	8693	19200↓	8.7↑	2215↓	0.453↑	0.378↑	< 2 s	40
USMJERAVANJE : MiFER, od SSP najduljeg										
C_{39}	g_1^*	12•	8855	17272↑	4.8	3450↑	0.513↓	0.488↑	< 2 s	40
USMJERAVANJE : MxFER, od najvećeg prema duljini grana										
C_{40}	g_5	6↓	11976	27404↑	8.2↑	3498↓	0.437↓	0.360↓	< 2 s	35

*uz sekundarni kriterij usmjerenja • ista vrijednost ↑ porast ↓ smanjenje

Utjecaj kontrolnih parametara na rezultate prekrivanja može u određenim slučajevima biti vrlo značajan. Primjer uočen u ovom testu odnosi se na izvođenja postupka B uz MiFER usmjerenje od najmanjeg zahtjeva prema najvećem. Ukoliko se težina grana u postupku određivanja puteva definira brojem hop-ova, te se zabrani provjera sekundarnog kriterija u slučaju puteva istih težina, dobivaju se bolji rezultati (tablica 5.22).

Tablica 5.22: SP3 – postupak B – utjecaj kontrolnih parametara

f	$ RCS $	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	T	N_{EC}	
USMJERAVANJE: MiFER, od najmanjeg prema broju HOP-ova – 150 prstenova										
C_{41}	g_1	13↑	8693	17542↓	4.8	3598	0.496↑	0.444↑	~200 ms	40
C_{42}	g_4	9•	8693	18906↓	6.9	2267	0.460↑	0.410↑	~200 ms	40
C_{43}	g_5	12↑	8693	16327↓	5.4	2925	0.532↑	0.422↑	~200 ms	40
USMJERAVANJE: MiFER, od najmanjeg prema broju HOP-ova – 300 prstenova										
C_{44}	g_1	11↑	8624	17435↓	5.2	3030	0.499↑	0.449↑	~200 ms	40

• ista vrijednost ↑ porast ↓ smanjenje

U svim iznesenim rješenjima dolazi do smanjenja zahtijevanih kapaciteta u mreži, uz povećanje iskoristivost mreže i prosječne iskoristivosti prstenova.

Sagledavanjem dobivenih rezultata u SP3, mogu se izdvojiti rješenja C_4 , C_6 i C_{11} dobivena postupkom A, te C_{21} , C_{31} i C_{43} dobivena postupkom B. Ta prekrivanja su odabrana zbog najmanjih vrijednosti zahtijevanih kapaciteta u mreži, te zbog visokih vrijednosti iskoristivosti mreže i prosječne iskoristivosti prstenova. Najniža vrijednost zahtijevanih kapaciteta postignuta korištenjem postupka A je niža od one dobivene postupkom B. Kao što je primjećeno na testnoj mreži 1, u oba postupka se najbolja rješenja dobivaju primjenom težinskih funkcija ovisnih o iskoristivosti prstena (f_4 i f_6 u postupku A, te g_1 , g_4 i g_5 u postupku B).

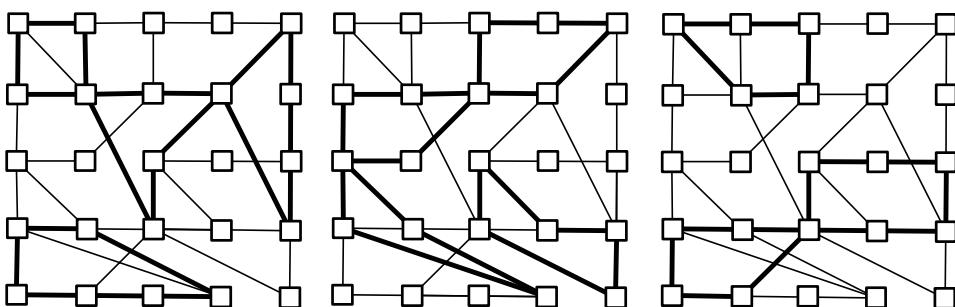
Utjecaj povećanja broja zahtjeva na rezultate

U SPI i SP2 detaljnije su analizirani rezultati prekrivanja nad istom testnom mrežom ali uz različito definirane prometne zahtjeve. Pokazalo se da postoje sličnosti u rezultatima, kao i pravilnosti u korištenju pojedinih težinskih funkcija.

Zbog opsežnosti ispitivanja, nad testnom mrežom 2 ne vrši se detaljna analiza prekrivanja uz veći broj zahtjeva, već se na modelu sa dvostruko većim brojem zahtjeva provode metode koje su uz manji broj zahtjeva polučile dobre rezultate. Postupak je proveden uz definirani skup od 150 mogućih prstenova. Umjesto 100 prometnih zahtjeva ($SP3$), u $SP4$ definira se 200 prometnih zahtjeva.

Rezultati su potvrđili otprije utvrđene pravilnosti. Najbolji rezultati prekrivanja dobivaju se korištenjem težinskih funkcija proporcionalnim iskoristivosti prstenova. Za prekrivanja se predlaže 8-15 prstenova. Očit je porast broja predloženih prstenova što se može objasniti većom vjerojatnošću pojave lokalnog prometa u manjim prstenovima, a uslijed povećanja broja prometnih zahtjeva. Postoje neke razlike u predloženim prekrivanjima, što je očekivano obzirom na drugačije definirane prometne zahtjeve.

Primjer rezultata D_B dobivenog postupkom B uz korištenje težinske funkcije g_1 prikazan je na slici 5.26. Rezultat je sličan rezultatima dobivenim u primjeru $SP3$.



Slika 5.26: $SP4$ – prikaz rezultata D_B dobivenog postupkom B

5.4.6. Testna mreža 2 – rezultati matematičkog programa

Zbog složenosti modela testne mreže 2 rezultati dobiveni matematičkim programom nisu prikazivani grafički, već je rezultat iskorišten samo kako bi opravdali rezultati prekrivanja dobiveni definiranim postupcima. Uz definiranje težina prstenova korištenjem težinskih funkcija g_1 , g_3 i g_5 , definirani su ulazni podaci za opis modela mreže u AMPL-u. Korištenjem CPLEX solver-a dobiveni su optimalni rezultati, ali uz već prije opisane nedostatke. Korištenjem funkcija g_1 i g_5 rezultat prekrivanja dobiven matematičkim programom uključuje 7 prstenova, a korištenjem funkcije g_3 9 prstenova. Broj prstenova uključenih u prekrivanje je uskladiv s brojem prstenova dobivenih postupcima A i B.

5.4.7. Testna mreža 2 – zaključak

Testna mreža 2 predstavlja specifičan model mreže namijenjen isključivo testiranju i analizi rješenja prekrivanja. Zbog složene topološke strukture, nad takvom mrežom je moguće definirati velik broj prstenova, što omogućava analizu rezultata prekrivanja uz variranje veličine skupa mogućih prstenova. Rezultati su pokazali da povećanje skupa mogućih prstenova ne dovodi nužno do poboljšanja svih karakterističnih parametara rješenja. U postupku B povećanje broja mogućih prstenova nije dovelo do boljih rješenja u pogledu zahtijevanih kapaciteta mreže, već se radilo isključivo o poboljšanjima iskoristivosti mreže i prstenova. Detaljnija analiza prometnih zahtjeva karakteristična za postupak A ipak je rezultirala smanjivanjem zahtijevanih kapaciteta mreže, a ujedno i poboljšanjima ostalih karakterističnih vrijednosti mreža.

Utjecaj prometnih zahtjeva na odabir prstenova koji prekrivaju mrežu već je uočen u mnogo jednostavnijoj testnoj mreži 1. U slučaju testne mreže 2, definiranjem 200 prometnih zahtjeva definira se 66% ukupno mogućih prometnih zahtjeva u mreži, što utječe na 'raspršenost' zahtjeva u mreži. To može dovesti do povećanja lokalnog prometa u manjim prstenovima, te samim time do

povećanja njihove iskoristivosti. To će rezultirati odabirom manjih prstenova u rješenja prekrivanja.

5.4.8. Testna mreža 3 - studijski primjer 5

Testna mreža 3 je najrealniji model mreže koji se koristi za analizu rezultata u ovom radu. Radi se o primjeru transportne mreže na području Republike Hrvatske. Promet u mreži definiran je slučajno generiranjem 500 prometnih zahtjeva. Iako stvarni promet u takvoj mreži pokazuje pravilnosti (lokalni i međuzupanijski promet), to u testiranjima nije uzeto u obzir. Jedini cilj je bio provjeriti ispravnost funkcioniranja postupaka za određivanje prstenova koji prekrivaju mrežu.

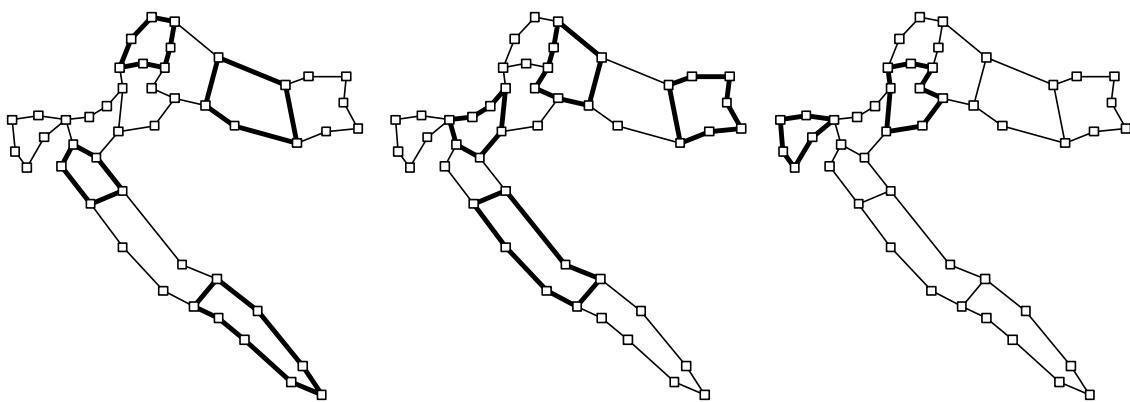
U tablici 5.23 su navedeni neki od rezultata dobivenih postupkom A.

Tablica 5.23: SP5 – postupak A

<i>f</i>	$ RCS $	$\Sigma w(e)$	$P_{N(BSHR)}$	$I(R)$	<i>L</i>	C_{BSHR}	μ_{BSHR}	μ_R	<i>T</i>	N_{EC}	
SP HOP usmjeravanje, bez dodatne obrade lokalnih zahtjeva											
<i>E</i> ₁	<i>f</i> ₁	10	81227	184107	6.6	12%	28405	0.441	0.485	< 3 s	54
<i>E</i> ₂	<i>f</i> ₄	10	81227	173398	6.8	13%	25842	0.468	0.486	< 3 s	54
<i>E</i> ₃	<i>f</i> ₇	11	81277	180876	7.4	15%	22859	0.441	0.446	< 3 s	54
SP HOP usmjeravanje, dodatna obrada lokalnih zahtjeva											
<i>E</i> ₄	<i>f</i> ₁	10	81227	183879	6.6	12%	28378	0.442	0.486	< 4 s	54
<i>E</i> ₅	<i>f</i> ₄	10	81227	172984	6.8	13%	25784	0.469	0.488	< 4 s	54
<i>E</i> ₆	<i>f</i> ₇	9	81227	161192	7.6	16%	19060	0.528	0.494	< 4 s	54
<i>E</i> ₇	<i>f</i> _{7*}	10	81227	161162	8.1	22%	18692	0.533	0.526	< 4 s	54

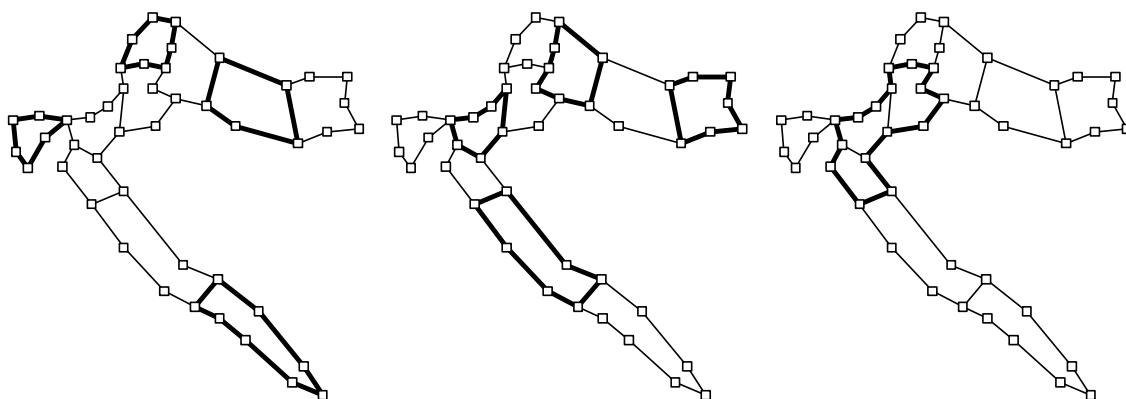
*drugi najbolji prsten u prvoj iteraciji

Među rezultatima je karakteristično ponavljanje nekoliko načina prekrivanja – iako se radi o istim rješenjima prekrivanja, zbog različitih interpretacija prometnih zahtjeva karakteristične vrijednosti takvih rješenja se razlikuju. Na slici 5.27 je prikazano rješenje *E*₄ – rezultati prekrivanja u tom slučaju su karakteristični jer sadrže svih 10 elementarnih prstenova koje je moguće izgraditi nad testnom mrežom.



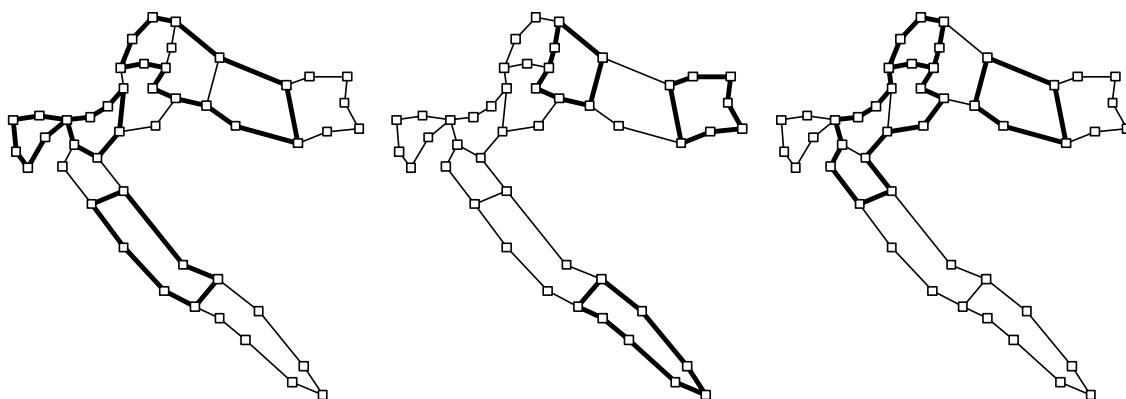
Slika 5.27: SP5 – prikaz rezultata *E*₄

Slika 5.28 prikazuje rezultat prekrivanja E_6 – rješenje je izabранo kao jedno od boljih s obzirom na zahtijevane kapacitete u mreži, a i zbog dobrog faktora iskoristivosti mreže.



Slika 5.28: SP5 – prikaz rezultata E_6

Rješenje prekrivanja E_7 karakteriziraju najniži zahtijevani kapaciteti. Usporedbom s ostalim prikazanim rješenjima može se uočiti odabir većeg prstena koji povezuje Slavoniju i središnji dio Hrvatske. No, to je najbolje ponuđeno rješenje za definirane prometne zahtjeve – osim što ima najniže zahtijevane kapacitete, te najviše faktore iskoristivosti mreže i prstenova, E_7 omogućuje promet u mreži uz najveći udio lokalnih zahtjeva. Zanimljivo je da je najbolje rješenje dobiveno korištenjem težinske funkcije f_7 . Ista funkcija nudila je dobra rješenja u testnoj mreži 1, dok rješenja dobivena tom funkcijom za testnu mrežu 2 nisu ulazila u uži izbor. Funkcija određuje težinu prstenova prema udjelu pravog lokalnog prometa, što više dolazi do izražaja u karakterističnim topologijama mreža poput ove u testnoj mreži 3. Korištenje funkcije f_4 koja težinu prstena definira proporcionalno iskoristivosti prstena također daje zadovoljavajuća rješenja.



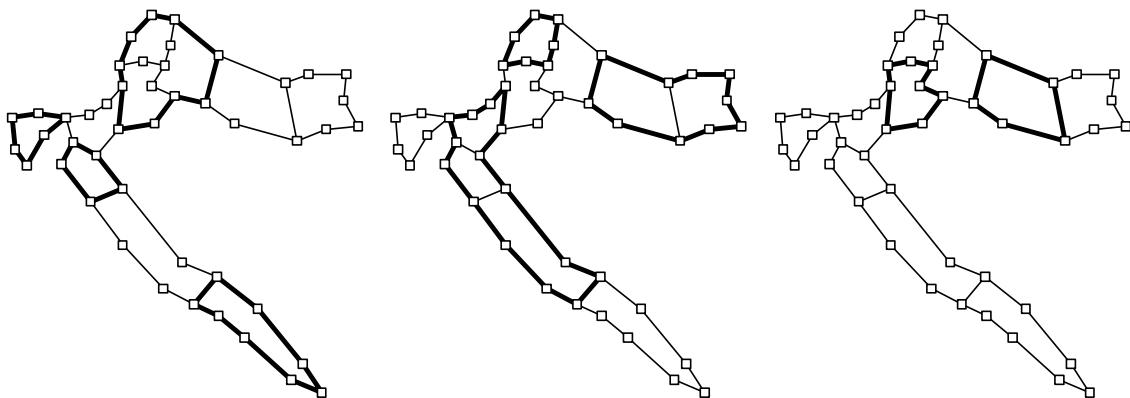
Slika 5.29: SP5 – prikaz rezultata E_7

Rezultati prekrivanja dobiveni postupkom B izneseni su u tablici 5.24.

Tablica 5.24: SP5 – postupak B

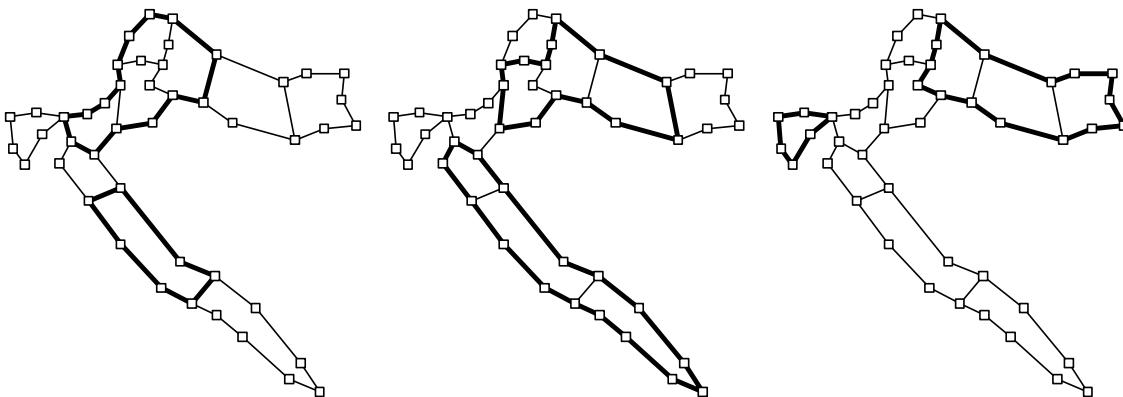
<i>f</i>	$ RCS $	$\Sigma w(e_i)$	$P_{N(BSHR)}$	$I(R)$	C_{BSHR}	μ_{BSHR}	μ_R	<i>T</i>	N_{EC}	
USMJERAVANJE: SP HOP										
E_8	g_1	10	81227	173398	6.8	25842	0.468	0.487	$\sim 500\ ms$	54
E_9	g_4	8	81227	184586	8.9	18399	0.440	0.396	$\sim 500\ ms$	54
E_{10}	g_5	9	81277	173331	8	17997	0.469	0.407	$\sim 500\ ms$	54
USMJERAVANJE: MiFER, od najmanjeg										
E_{11}	g_1	10	81237	175492	6.8	26150	0.463	0.450	$\sim 500\ ms$	54
E_{12}	g_5	9	81237	171593	8.1	17500	0.473	0.409	$\sim 500\ ms$	54
USMJERAVANJE: MiFER, od najvećeg										
E_{13}	g_5	9	81427	170653	8.1	17500	0.478	0.413	$\sim 800\ ms$	54
USMJERAVANJE: MiFER, od SP najduljeg										
E_{14}	g_5	9	81875	171141	8.1	17426	0.477	0.412	$\sim 500\ ms$	54

Najbolji rezultat dobiven postupkom B je prekrivanje E_{13} . Prikaz prekrivanja dan je na slici 5.30. Najbolje prekrivanje ostvareno je korištenjem funkcije g_5 .



Slika 5.30: SP5 – prikaz rezultata E_{13}

Samо као илustrација različitih mogućnosti prekrivanja mreže, na slici 5.31 je prikazano rješenje koje se dobiva korištenjem težinske funkcije g_3 . Rezultantna mreža ima nisku iskoristivost što se može objasniti velikim kapacitetima velikih prstenova – opterećenje grana ovisi isključivo o inicijalnom usmjeravanju, a kapaciteti prstenova su veliki zbog velikog broja prometnih zahtjeva koji njima prolaze.



Slika 5.31: SP5 – prekrivanje korištenjem 6 prstenova

Studijskim primjerom 5 željelo se provjeriti metode određivanja prstenova na primjeru stvarne mreže. Naime, u postupku planiranja teoretski modeli se ponekad teško prilagodavaju na realne slučajeve. Ovog puta to nije slučaj jer su rezultati zadovoljavajući.

Rezultati dobiveni matematičkim programom

U AMPL modelu studijskog primjera 5 težine prstenova su definirane korištenjem težinskih funkcija g_1 , g_3 i g_5 . Rezultati prekrivanja su uključivali redom 6, 10 i 7 prstenova. Prekrivanje dobiveno funkcijom g_3 identično je rješenju E_4 prikazanom na slici 5.27. Prekrivanja dobivena funkcijama g_1 i g_5 uključuju veće prstenove, slično rješenju prikazanom na slici 5.31.

Može se zaključiti da su rješenja dobivena postupcima A i B uskladiva prema broju prstenova sa rješenjima dobivenim matematičkim programom.

5.5. Zaključak o rezultatima planiranja

Ovim poglavljem obavljena je detaljna analiza rezultata postupka planiranja prstenastih mreža. Naglasak je stavljen na rezultate prekrivanja mreža prstenovima. Analiza je obavljena nad tri testne mreže, od kojih je svaka karakterističnih svojstava. Testna mreža 1 je mala mreža koja je iskorištena za detaljan prikaz mogućih rješenja, dok testna mreža 2 predstavlja složeni model visoke razine povezanosti. Testna mreža 2 je zbog velikog broja prstenova poslužila za detaljniju analizu utjecaja faze određivanja prstenova (enumeracije) na kompletni postupak određivanja prstenova koji prekrivaju mrežu. Model testne mreže 3 je izведен kao primjer stvarne prijenosne telekomunikacijske mreže na području Republike Hrvatske.

Testiranja su provedena kroz pet studijskih primjera. U sklopu svakog testiranja analizirane su sličnosti i razlike u rezultatima dobivenim korištenjem različitih postupaka za određivanje prekrivanja. Rezultati su kvalitativno usporedivani i sa rezultatima dobivenim korištenjem matematičkog programa za optimizaciju modela mreže. Svi korišteni postupci implementirani su u programskim alatima opisanim u poglavljju 4, te su primijenjeni u istraživačkim projektima Zavoda za telekomunikacije.

Rezultati su pokazali da se korištenjem implementiranih postupaka mogu dobiti dobri rezultati prekrivanja mreža prstenovima. Analizirani su utjecaji različitih izvedbi ostalih faza postupka (usmjeravanje prometnih zahtjeva, ograničenja u postupku određivanja prstenova u mreži), ali i utjecaji kontrolnih parametara vezanih uz sam postupak određivanja prstenova koji prekrivaju mrežu (različite težinske funkcije, obrada prometnih zahtjeva na razini prstena itd.).

Ono što je pokazano rezultatima jest da je moguće odrediti velik broj prekrivanja variranjem raznih ulaznih parametara. Svako resultantno prekrivanje vrednuje se analizom karakterističnih vrijednosti. Budući da nije definirana kontrolna funkcija kojom bi se odredili utjecaji različitih karakterističnih vrijednosti na konačnu odluku o najboljem rješenju, odluka se prepušta dizajnerima mreže.

Zaključak

Ovim magistarskim radom dan je detaljan pregled postupaka koji se mogu primijeniti u planiranju prstenastih telekomunikacijskih mreža. Obradeni su problemi specifični za planiranje prstenastih mreža: problem dimenzioniranja jednostavnog prstena, problem određivanja prstenova u mreži i problem određivanja prekrivanja mreže prstenovima. Svaki problem je detaljno opisan te su predloženi heuristički i egzaktni algoritmi za njihovo rješavanje. Uz to, predloženi su matematički modeli za rješavanje optimizacijskih problema dimenzioniranja jednostavnih prstenova i prekrivanja mreže prstenastim strukturama.

Svi opisani algoritmi za rješavanje navedenih problema implementirani su u prateće programske alate koji su izrađeni za potrebe analize rezultata.

Cilj rada je bio razviti heurističke algoritme za rješavanje problema koji se mogu javiti u postupku planiranja prstenastih mreža s naglaskom na rješavanje problema prekrivanja mreže prstenastim strukturama. Budući da nije moguće jednoznačno odrediti kriterije vrednovanja rezultantnih mreža, predložen je pristup koji se svodi na usporedbu i analizu različitih karakterističnih vrijednosti mreža. Analiza tako dobivenih i vrednovanih rezultata pokazala je da je cilj uspješno ostvaren, jer su dobiveni zadovoljavajući rezultati za sve primjere testnih mreža. Uočene su pravilnosti u korištenju različitih implementiranih metoda i kontrolnih parametara postupka, te su odredene težinske funkcije koje doprinose kvaliteti rezultata.

Unapređenje postupka planiranja prstenastih mreža moglo bi se postići nadogradnjom jednostavnog matematičkog modela mreže iznesenog u radu, odnosno definiranjem novog složenijeg matematičkog modela, koji bi uključivao sva bitna svojstva mreže – osim topologije mreže, model bi trebao uključivati i prometne zahtjeve. Na taj način mogao bi se optimizacijski problem mogao matematički definirati. što bi omogućilo optimalno rješavanje problema prekrivanja mreže prstenovima, uključujući optimalno usmjeravanje prometnih zahtjeva. Tek po razvoju takvog složenog modela moglo bi se obaviti točno vrednovanje rezultata dobivenih algoritmima opisanim u ovom radu. Kako takav model nije razvijen u sklopu ovog rada, izneseni zaključci o rezultatima odnose se samo na usporedbe rezultata dobivenih implementiranim algoritmima.

Programski alati koji uključuju sve implementirane algoritme korišteni su u planiranju mreža u okviru istraživačkih projekta Zavoda za telekomunikacije. Budući da su algoritmi inicijalno razvijani za jednostavni model prstena, prije primjene u stvarnim mrežama bila je potrebna prilagodba i uvođenje niza realnih ograničenja. Prilagođeni algoritmi pokazali su dobre rezultate, što potiče daljnji razvoj alata koji će omogućiti dizajnerima telekomunikacijskih mrež interaktivno sudjelovanje u postupku, uz mogućnost potpune automatizacije postupka.

Literatura

- [1] Mikac M., "Algoritmi za optimizaciju prijenosne telekomunikacijske mreže", diplomski rad, 2000.
- [2] Fourer R., Gay D.M., Kernighan B.W., "AMPL: A Modeling Language for Mathematical Programming", Scientific Press, 1993.
- [3] Papadimitrou C.H., Steiglitz K., "Combinatorial Optimization: Algorithms and Complexity", Dover Publications, 1998. (1982. Prentice-Hall)
- [4] Lawler E., "Combinatorial Optimization: Networks and Matroids", Dover Publications, 2001.
- [5] Myung Y.S., Kim H.G., Tcha D.W., "Optimal Load Balancing on SONET Bidirectional Rings", Operations Research, Vol.45, No.1, pp.148-153, 1997.
- [6] Lee C.Y., Chang S.G., "Balancing Loads on SONET Rings with Integer Demand Splitting", Computers Operations Research, Vol.24, No.3, pp. 221-229, 1997.
- [7] Cosares S., Saniee I., "An Optimization Problem Related to Balancing Loads on SONET Rings", Telecommunication Systems, No. 3, pp. 165-181, 1994.
- [8] Kos M., Mikac M., Mikac D., "Different Approaches for Solving Ring-dimensioning Problem", Proceedings of the 10th ICTS, Vol.2, pp.609-618, 2002.
- [9] ITU-T SDH Standards – G.774, G.783, G.841, G.842
- [10] Morley G.D., Grower W.D., "A Comparative Survey of Methods for Automated Design of Ring-based Transport Networks", TRLabs Technical Report TR-97-04, Issue 1.0., 1998.
- [11] Wu T.H., "Fiber Network Service Survability", Artech House, 1992.
- [12] SDH Telecommunications Standard Primer, Tektronix 2001.
- [13] International Telecommunications, Vol. 37, No.4, 2003.
- [14] IEEE Communications Magazine, Vol. 37, No.8, 1999.
- [15] Kennington J.L., Nair V.S.S., Rahman M.H., "Optimization Based Algorithms for Finding Minimal Cost Ring Covers in Survivable Networks", Computational Optimization and Applications, 14, pp. 219-230, 1999.
- [16] Grover W.D., Slevinsky J.B., MacGregor M.H., "Optimized Design of Ring-based Survivable Networks", Canadian J. Elect. & Comp. Eng., Vol. 20, No.3., pp.139-149, 1995.
- [17] Cox L.A., Sanchez J., "Cost Savings from Optimized Packing and Grooming of Optical Circuits: Mesh vs. Ring Comparisons", NetAdvantage, 2000.

Dodatna literatura:

- [18] SONET Telecommunication Standard Primer, Tektronix 2001.
- [19] Henley E.J., Williams R.A., "Graph Theory in Modern Engineering", Academic Press, 1973.
- [20] Balakrishnan V.K., "Shaum's Outline of Graph Theory", McGraw-Hill, 1997.
- [21] Bronson R., Naadimuthu G., "Shaum's Outline of Operations Research", McGraw-Hill, 2nd Edition, 1997.
- [22] International Telecommunications, Vol. 37, No. 7, 2003.

Životopis – Matija Mikac

Matija Mikac rođen je 18.6.1977. godine u Varaždinu. Tijekom dodiplomskog studija primio je Rektorovu nagradu (1998), i Nagradu Ericsson Nikola Tesla (1999). Diplomirao je u rujnu 2000. godine na Zavodu za telekomunikacije Fakulteta elektrotehnike i računarstva. Po završetku studija zapošljava se na Zavodu za telekomunikacije Fakulteta elektrotehnike i računarstva kao znanstveni suradnik. Tijekom rada na Zavodu, sudjelovao je u nekoliko istraživačkih projekata. Područje njegovog stručnog i znanstvenog interesa su problemi planiranja telekomunikacijskih mreža. Član je laboratorija za napredna telekomunikacijska istraživanja LATS.

Popis objavljenih radova

1. M.Kos, M.Mikac, D.Mikac : "*Topological planning of communication networks*", IIS 2002, Varaždin, Hrvatska, 25-27. September 2002.
2. M.Kos, M.Mikac, D.Mikac : "*Different approaches for solving ring-dimensioning problem*", ICTSM 10, Monterey, CA, USA, 3-6. October 2002.
3. J.Gracin, M.Kos, M.Mikac, Ž.Popović : "*TCP ECN in ATM network*", ICTSM 2001, Dallas, Texas, USA

Tema magistarskog rada:

**HEURISTIČKI ALGORITMI ZA PLANIRANJE PRSTENASTIH
TELEKOMUNIKACIJSKIH MREŽA**

Sažetak:

Prstenaste telekomunikacijske mreže posjeduju svojstvo da mogu jednostavnim mehanizmima osigurati zaštitu prometa u slučaju kvarova. To ih je svojstvo učinilo vrlo čestim izborom u temeljnim mrežama u kojima je sprečavanje gubitka informacija u slučaju kvara od osobite važnosti. U postupku planiranja prstenastih mreža postoji niz specifičnih problema koje je potrebno riješiti. Ovim magistarskim radom daje se detaljan pregled postupka planiranja prstenastih mreža, te se ukazuje na osnovne probleme koji se javljaju u postupku. Problemi koji su obradivani radom uključuju problem dimenzioniranja jednostavnih prstenova, problem određivanja prstenova na zadanim topologijama mreža, te problem prekrivanja mrežne topologije prstenastim strukturama. Problemi su sagledani analitički, te su naznačene neke mogućnosti rješavanja. Poseban naglasak stavljen je na primjenjivost rješenja u stvarnim mrežama – iako se model prstenastih mreža može primijeniti korištenjem različitih tehnoloških rješenja, u radu se stavlja naglasak na korištenje u SDH prijenosnim sustavima.

Kao praktični dio rada implementirane su heurističke metode koje rješavaju spomenute probleme, te su analizirani dobiveni rezultati. Problemi dimenzioniranja prstenova i prekrivanja mrežne topologije prstenovima su također matematički formulirani, te je provedena optimizacija korištenjem matematičkog programiranja. Analizom rezultata zaključeno je da se iznesene metode mogu primijeniti u postupku planiranja prstenastih mreža.

Praktični dio rada uključuje nekoliko razvijenih programskih alata koji se mogu koristiti u postupku planiranja prstenastih mreža – alati su korišteni u izvedbi nekih istraživačkih projekata te je time njihova funkcionalnost potvrđena na primjerima stvarnih mreža.

Master thesis:

***HEURISTIC ALGORITHMS FOR RING-BASED
TELECOMMUNICATION NETWORKS DESIGN***

Abstract

Ring-based telecommunication networks can provide simple and effective traffic protection mechanisms in case of network failure. Because traffic protection is one of the most important issues in network design, ring-based networks are often used as parts of backbone networks. Several specific problems can arise within ring-based network design process. This work includes detailed overview of such planning process. Specifications of following problems are given: ring dimensioning problem, ring enumeration problem and problem of covering network with a subset of rings. All the problems are analyzed and heuristic algorithms that could solve them efficiently are proposed. Aspects of practical usage of proposed algorithms were studied, with special treatment for, in praxis, most often used, SDH transport systems.

All algorithms were implemented within software tools developed as practical part of this work. Software tools were already used in some exploration projects and have shown their functionality on real-world network examples.

Ključne riječi:

prstenasta mreža, jednostavni prsten, zaštitni prsten, samooporavljeni prsten, optimizacija kapaciteta prstena, problem dimenzioniranja prstena, problem određivanja (enumeracija) prstenova, problem prekrivanja mrežne topologije zatvorenim strukturama, zaštita prometa, SDH, planiranje prstenastih mreža

Keywords:

ring network, cycle, ring, protection ring, self-healing ring, ring capacity optimization, ring dimensioning problem, ring enumeration, network covering, ring covering, traffic protection, SDH, ring-based network planning process