SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 465

RJEŠAVANJE PROBLEMA USMJERAVANJA I DODJELJIVANJA VALNIH DULJINA U WDM OPTIČKIM MREŽAMA PRIMJENOM METAHEURISTIKA

Roman Vazdar

Zagreb, svibanj 2012.

Ovom prilikom zahvaljujem se svom mentoru prof. dr. sc. Domagoju Jakoboviću na susretljivosti, razumijevanju i korisnim savjetima tijekom izrade ovog rada, ali i pristupačnosti te savjetima tokom cijelog studija. Također zahvaljujem se doc. dr. sc. Nini Skorin-Kapov na vrijednim savjetima te pojašnjenjima u počecima i tijekom izrade diplomskog rada.

Naposljetku, zahvaljujem se svima koji su mi pružili podršku i pomoć tijekom studija, posebno svojim roditeljima.

Sažetak

U ovom diplomskom radu proučen je i opisan problem usmjeravanja i dodjeljivanja valnih duljina u WDM optičkim mrežama korištenjem metaheuristika. Na standardni problem dodana su ograničenja za raspoređene svjetlosne zahtjeve te smetnje na putevima svjetlosnih zahtjeva. Za potrebe rješavanja navedenih problema detaljno je istražena problematika te su ispitana dva moguća algoritma: hibridni evolucijski algoritam i varijabilna pretraga susjedstva. Ideja hibridnog evolucijskog algoritma sastoji se od ugradnje metode lokalne pretrage u evolucijski algoritam s populacijom jedinki od koje svaka predstavlja potencijalno rješenje. Algoritam varijabilne pretrage susjedstva kao svoj zadatak ima pretragu susjedstava nekog početnog suboptimalnog rješenja u svrhu njegovog rješavanja. Učinkovitost algoritama ispitana je na stohastički stvorenim grafovima. Na kraju dane su smjernice za daljnje istraživanje.

Abstract

In this diploma thesis, the problem of routing and wavelength assignment in wavelength-divison multiplexing optical networks using metaheuristics is studied and described. New constraints have been added on standard problem: scheduled lightpath demands and imapairments on lightpaths. Solutions needed for solving the problems have been explored in detail and two algorithms have been studied: hybrid evolutionary algorithm and variable negihborhood search.

The idea behind the hybrid evolutionary algorithms consits of embedding local search into the framework of population based evolutionary algorithms where each individual represents a potential solution. Variable nieghborhood search algorithm searches multiple neighborhoods in an effort to improve initially created suboptimal solution. The efficiency of algorithms has been tested on stohastically generated graphs. Finally guidelines are given for further research.

Sadržaj

1. Uvod

Optičke komunikacijske mreže su mreže budućnosti. Kako sve više korisnika počinje korisiti podatkovne mreže, i kako njihovi uzorci uporabe počinju sve više naginjati prema aplikacijama zahtjevnijima za brzinu prijenosa kao naprimjer pregledavanje sadržaja na Internetu, Java aplikacije, video konferencije, itd., nazire se potreba za transportnim mrežama s velikom širinom pojasa. Takve mreže (kao na primjer današnji Internet) tu potrebu sve teže zadovoljavaju. Također jedna svjetlovodna nit u jednomodnom načinu rada ima potencijalnu širinu pojasa od 50 terabita po sekundi što je mnogo više od elektroničke brzine prijenosa podataka (od nekoliko desetaka gigabita po sekundi), stoga bi tu veliku razliku bilo poželjno iskoristiti. Multipleksiranje po valnoj podjeli može pomoći u iskorištavanju te razlike u širini pojasa zahtijevajući da oprema svakog krajnjeg korisnika

Radi na elektroničkim brzinama ali je moguće multipleksirati više WDM kanala od više krajnjih korisnika na istu svjetlovodnu nit. Kako bi optimalno mogli upravljati i koristiti optičke mreže potrebno je razvijati dobar software za njih, poglavito kod iskorištavanja mogućnosti multipleksiranja po valnoj podjeli. No time se rađaju novi problemi. Ponajprije problem usmjeravanja i dodjeljivanja valnih duljina. U ovom radu se ponajprije opisuje taj problem i načini njegova rješavanja zajedno s algoritmima koji ga rješavaju.

Rad je organiziran po poglavljima kako slijedi. U poglavlju 2 opisane su optičke telekomunikacijske mreže, njihove prednosti te načini multipleksiranja. U poglavlju 3 opisane su optičke mreže s valnim multipleksiranjem, zašto se valno multipleksiranje vrši te komponente u optičkim mrežama. Poglavlje 4 pobliže objašnjava optičke WDM mreže s valnim usmjeravanjem, pojam svjetlosnog puta i način na koji funkcioniraju. U poglavlju 5 opisuje se problem usmjeravanja i dodjeljivanja valnih duljina (RWA), njegova podjela na potprobleme, vrste svjetlosnih zahtjeva te pojašnjenje smetnji u optičkoj mreži. U poglavlju 6 se opisuju algoritmi korišteni za rješavanje RWA problema, opis termina potrebnih za njihovo razumijevanje te implementacija ograničenja prisutnih u sklopu ovoga rada unutar razvijenog programa RWA Solver. U poglavlju 7 slijedi analiza rezultata eksperimenta.

Zadnja dva poglavlja 8 i 9 sadrže zaključak i popis korištene literature.

2. Optičke telekomunikacijske mreže

Komunikacijsku mrežu čine međusobno povezani komunikacijski sustavi na koje se spaja korisnička oprema (komunikacijska, računalna) i druga oprema potrebna za pružanje informacijskih i komunikacijskih usluga te potporu aplikacija korisnicima (poslužiteljska računala i drugi sustavi). Optičke telekomunikacijske mreže se zasnivaju na uporabi svjetla kako bi se prenijela informacija od jedne točke do druge. Prijenosni medij koji se koristi je staklena nit smještena unutar ovojnice napravljene isto tako od stakla i obavijena zaštitnim slojem.

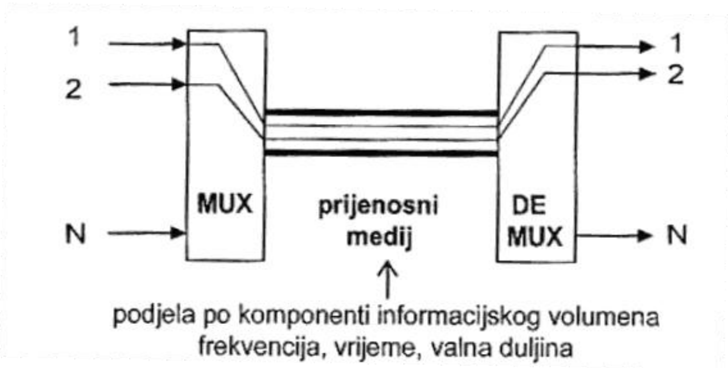
Svjetlovodnom niti prenosi se svjetlosni signal. Predajnik može biti dioda LED (*Light Emitting Diode*) ili poluvodički laser, a prijamnik je fotodioda. Optički kabel (engl. *fibre optic cabel*) sadrži jednu ili više svjetlovodnih niti. Tehnologija svjetlovodnih niti može zadovoljiti potrebu za povećanom širinom pojasa (engl. *bandwidth*), zbog sljedećeg velikog potencijala i mogućnosti:

1. velika širina pojasa (preko 50 Terabita po sekundi)
2. nisko prigušenje (engl. *attenuation*) signala (počevši od 0.2 dB/km), stoga je moguća upotreba vrlo malog broja obnavljača za transmisiju preko velike udaljenosti
3. neosjetljivost na elektromagnetske utjecaje i koroziju
4. teško izvedivo prisluškivanje
5. mogućnost multipleksiranja valnih duljina
6. nema preklapanja (engl. *crosstalk*) i smetnji između svjetlovodnih niti unutar jednog kabela
7. niska potreba za snagom
8. niti su nezapaljive, lagane i tanke

Nepovoljne strane gore spomenute tehnologije su isključivo jednosmjerni prijenos preko jedne svjetlovodne niti (stoga unutar jednog kabela imamo više niti), skuplja i teža za uporabu optička sučelja naspram električkih te složeniji i skuplji instrumentarij za izvedbu i održavanje. Unatoč tim nedostatcima kada je riječ o velikim mrežama velikog kapaciteta te velikim udaljenostima koje treba premostiti koriste se svjetlovodne niti. Prednosti uvelike nadmašuju nedostatke. Uz navedenu veliku širinu prijenosa svjetlovodnih niti, na krajnjim točkama često dolazi do pretvorbe iz optičkih signala u elektroničke. Elektronički signali imaju ograničenje širine pojasa na nekoliko gigabita po sekundi. Stoga je potrebno uvesti istodobnost između transmisija više korisnika unutar arhitekture mreže i njenih protokola, kako bi se iskoristila velika širina pojasa svjetlovodne niti. U optičkim telekomunikacijskim mrežama, ta istodobnost (engl. *concurrency*) se može realizirati putem različitih načina multipleksiranja.

Učinkovitost i kapacitet fizičkog sloja pospješuje višestruka uporaba prostorne komponente informacijskog volumena odnosno prijenosnog medija, tj. istodobni prijenos informacija koje pripadaju različitim vezama istim prijenosnim medijem.

Osim prostorne komponente moguća je i višestruka uporaba i drugih komponenti informacijskog volumena: frekvencijske, vremenske i valnih duljina kod optičkih svjetlovodnih niti.



Slika 1. Multipleksiranje po različitim komponentama infomracijskog volumena (1)

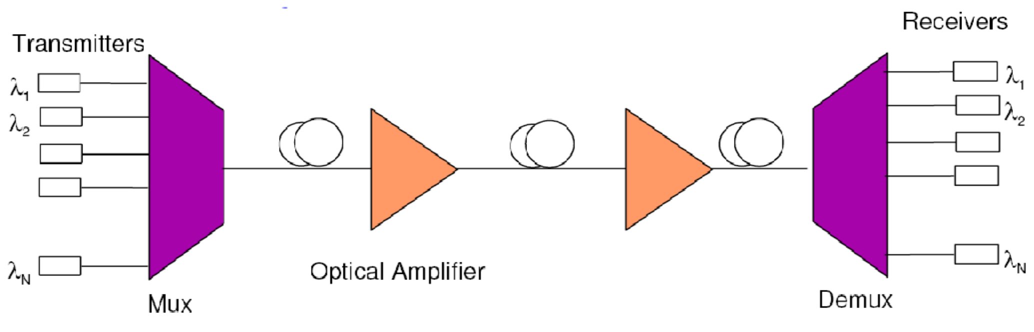
To znači da se svaka veza između jednog para odredište – izvor odvija na drugoj frekvenciji, u drugom vremenskom trenutku ili na drugoj valnoj duljini. Uz frekvencijsku i vremensku podjelu može se primjeniti i kodna podjela.

Podjela po nekoj komponenti informacijskog volumena provodi se tako da se na izvoru pojedine veze multipleksiraju na zajednički prijenosni medij,a na odredištu se provodi suprotan postupak demultipleksiranja.

Postoje različiti načini multipleksiranja s obzirom na komponente prijenosnog medija, kao što su: multipleksiranje u frekvencijskoj podjeli (engl. *Frequency Division Multiplexing*, FDM), multipleksiranje u vremenskoj podjeli (engl. *Time Divison Multiplexing*, TDM), multipleksiranje u valnoj podjeli (engl. *Wavelength Divison Multiplexing*, WDM) te multipleksiranje u kodnoj podjeli (engl. *Code Divison Multiplexing*, CDM). FDM i WDM su bazirani na istom konceptu uz činjenicu da se FDM koristi kod radijskog prijenosa dok se WDM koristi kod podjele valnih duljina u optičkom prijenosu putem svjetlovodnih niti. U ovom radu detaljnije će se opisati multipleksiranje u valnoj podjeli (WDM).

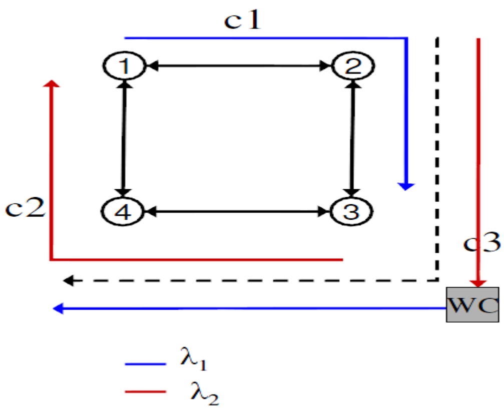
3. Optičke mreže s valnim multipleksiranjem

Multipleksiranje u valnoj podjeli (WDM) jest tehnologija pomoću koje se više signala simultano prenosi na raznim valnim duljinama. Time se iskorištava razlika između optičke i elektroničke širine pojasa. Odnosno oprema krajnjeg korisnika koristi širinu pojasa elektroničkog signala, dok se signali više krajnjih korisnika multipleksiraju na svjetlovodnu nit. Optički transmisijski spektar se razdvaja u više nepreklapajućih pojasa valnih duljina, gdje svaka valna duljina predstavlja jedan komunikacijski kanal koji ima proizvoljnu brzinu prijenosa podataka, npr. vršna brzina prijenosa u elektroničkom prijenosnom mediju.
Ovim multipleksiranjem dobivamo na značajnom povećanju kapaciteta bez goleme investicije za postavljanje novih vlakana. To je i jedan od razloga zbog čega su WDM tehnologija i mreže zasnovane na njoj popularne kod telekomunikacijskih kompanija.



Slika 2. Multipleksiranje u valnoj podjeli

Slika 2. Prikazuje način upotrebe multipleksera i demultipleksera u optičkim telekomunikacijskim mrežama. Prije je navedeno da se više podatkovnih signala multipleksiraju na jednu svjetlovodnu nit i njome se prenose, nakon što stignu na predajnu stranu oni se ponovno odvajaju a svaki predajnik selektivno izvuče odgovarajući signal korištenjem podesivih optičkih filtera. Za ostvarivanje WDM tehnologije potreban je veći broj komponenti: optički filteri, multiplekseri/demultiplekseri, WDM predajnici i prijemnici, optička pojačala, zvjezdasti raspréžnici, WDM usmjeritelji (engl. *wavelength router optical cross-connect*), valni pretvornici. Opis komponenti i njihove funkcije nisu predmet istraživanja unutar diplomskog rada, ali u slučaju kasnijeg referenciranja na neke od tih komponenti uslijediti će i njihovo detaljnije pojašnjenje. U slučaju da se od izvorišnog do odredišnog optičkog čvora koristi jedinstvena valna duljina, radi se o svjetlosnom putu (engl. *lightpath*, LP). Ukoliko nema valnih pretvornika u mreži, cijeloj konekciji (putu od odredišnog do izvorišnog čvora) mora biti dodijeljena ista valna duljina što je ograničenje jedinstvenosti valnih duljina (engl. *wavelength continuity constraint)* i biti će podrazumijevano u rješavanju problema dodjeljivanja valnih duljina.

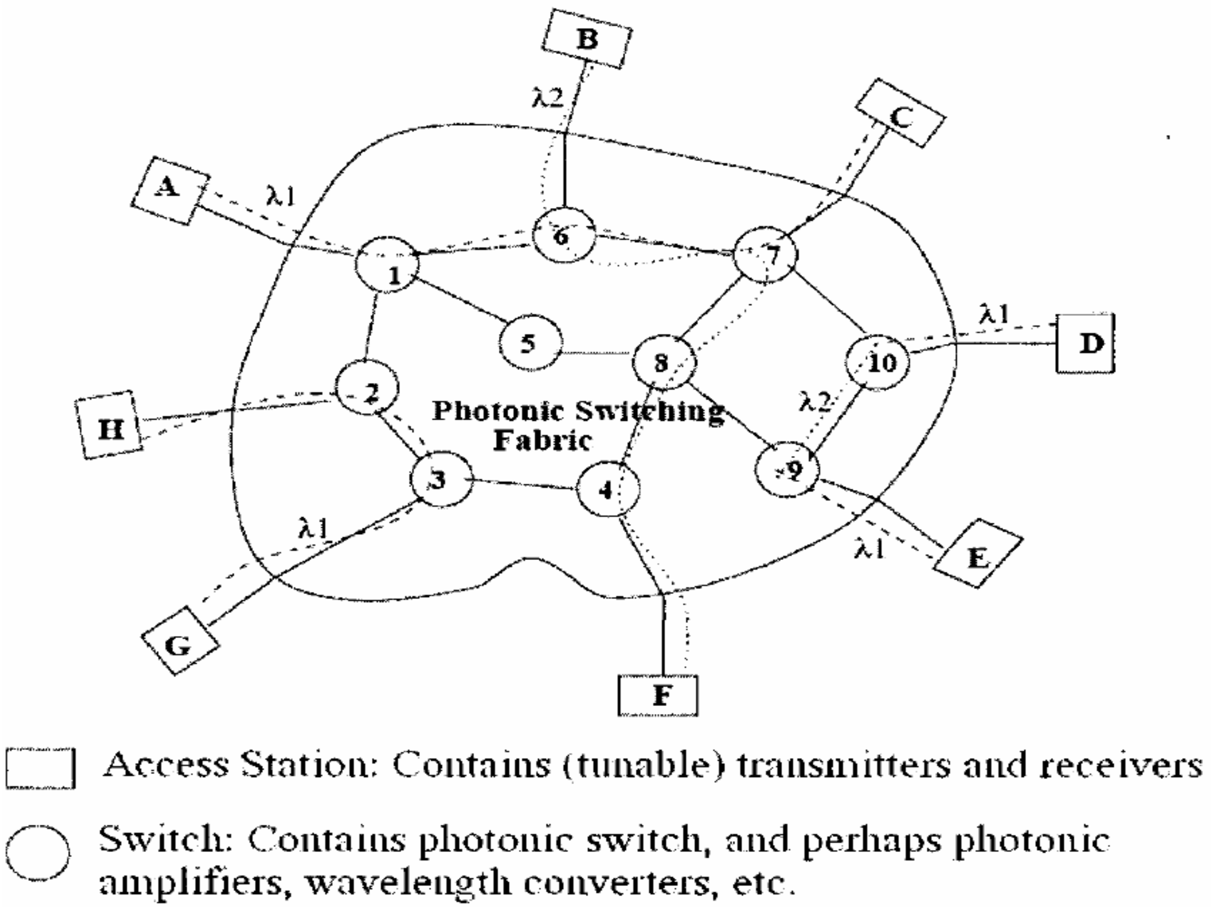


Slika 3. Virtualna topologija WDM optičke telekomunikacijske mreže sa valnim pretvornicima

4. Optičke WDM mreže s valnim usmjeravanjem

Problem usmjeravanja i dodjeljivanja valnih duljina (engl. *routing and wavelength assignment*, RWA) se vrši unutar optičkih mreža s valnim usmjeravanjem. Sastoje se od valnih usmjeritelja povezanih svjetlovodnim nitima. Svaki krajnji korisnik je spojen na valne usmjeritelje putem svjetlovodne niti. Kombinacija krajnjeg korisnika i valnog usmjeritelja se naziva čvor. Krajnji čvorovi su spojeni na valne usmjeritelje i svaki sadrži podesive (po valnoj duljini) prijamnike i predajnike.

Osnovni mehanizam komunikacije u mreži s valnim usmjeravanjem jest svjetlosni put ili optički WDM komunikacijski kanal. Svjetlosni put je optički komunikacijski kanal između dva čvora unutar mreže, i može obuhvaćati jednu ili više svjetlovodnih niti. Čvorovi između para čvora izvorište-odredište usmjeravaju svjetlosni put korištenjem valnih usmjeritelja. Krajnji čvorovi svjetlosnog puta su podešeni na valnu duljinu na kojoj svjetlosni put funkcionira putem predajnika i prijemnika.



Slika 4. Mreža s valnim usmjeravanjem

Slika 4. prikazuje primjer mreže s valnim usmjeravanjem gdje su vidljivi već postavljeni svjetlosni putevi. Svjetlosni putevi su postavljeni između krajnjih čvorova (na slici nazvanih *access station*) A i C (AC) na komunikacijskom kanalu valne duljine λ_1, između krajnjih čvorova B i F (BF) na komunikacijskom kanalu valne duljine λ_2 te između krajnjih čvorova H i G (HG) na komunikacijskom kanalu valne duljine λ_1. Svjetlosni put između čvorova A i C se usmjerava putem valnih usmjeritelja (na slici nazvani *switch*) 1, 6 i 7. Valna duljina λ_1 se upotrebljava više puta jer ne dolazi do preklapanja svjetlosnih puteva AC i HG. U prošlom poglavlju je spomenuto ograničenje jedinstvenosti valnih duljina koje je na slici 4 vidljivo kad su u pitanju prijašnje navedeni svjetlosni putevi AC, HG i BF. Valni usmjeritelji mogu sadržavati i optička pojačala, valne pretvornike te druge komponente. Valni usmjeritelji koji sadržavaju valne pretvornike ne moraju se pridržavati ograničenja jedinstvenosti valnih duljina putem nekog svjetlosnog puta što je i prikazano između krajnjih čvorova D i E koji između valnih usmjeritelja 9 i 10 koriste valnu duljinu λ_2 dok od krajnjih čvorova D i E do valnih usmjeritelja je u upotrebi valna duljina λ_1.

U optičkim mrežama s valnim usmjeravanjem od iznimne važnosti je ograničenje preklapanja valnih duljina (engl. *wavelength clash constraint*) koje zahtijeva da svjetlosni putevi koji prolaze istom svjetlovodnom niti imaju različite valne duljine kako ne bi smetali jedan drugom.

5. Problem usmjeravanja i dodjeljivanja valnih duljina

Kod mreža s valnim usmjeravanjem važno je spojiti dvije krajnje točke. Krajnje točke koje su prikazane i na slici 4. mogu predstavljati skup razne završne opreme (u obliku pristupnih točaka, komutatora, drugih usmjeritelja itd.). Ali u današnjim optičkim mrežama krajnje točke predstavljaju skup prometa od velikog broja terminala, uključujući i promet onih koji se spajaju od strane drugih regionalnih ili lokalnih podmreža. Stoga je ukupan promet tih krajnjih čvorova kojega šalje predajnik sumjerljiv i slične veličine kao i vršna elektronička brzina na komunikacijskom kanalu valne duljine.

Krajnji čvorovi komuniciraju jedni s drugima putem svjetlosnih puteva, te između dva čvora koji predstavljaju par izvorište-odredište uspostavljamo svjetlosni put ili konekciju. Broj svjetlosnih puteva koje je moguće uspostaviti je ograničen brojem WDM kanala (koji se kreće od 64 do 160) te zbog visoke cijene predajnika. Problem oko ograničenja zbog broja predajnika i prijemnika se neće razmatrati jer u ovom diplomskom radu nije bilo pristupa pravoj fizičkoj topologiji neke optičke mreže s njenim parametrima i detaljnim informacijama o komponentama. Umjesto toga predmet razmatranja će se osvrnuti na virtualnu topologiju, dok će fizička topologija biti predstavljena stohastičkim generiranjem grafa koji predstavlja neku WDM optičku mrežu.

Nakon što se skup svjetlosnih puteva odabere ili odredi u obliku skupa svjetlosnih zahtjeva, potrebno je usmjeriti svaki svjetlosni put kroz valne usmjeritelje i dodijeliti mu valnu duljinu, što se još naziva problemom usmjeravanja i dodjeljivanja valnih duljina (engl. *routing and wavelength assignment problem*), nadalje u tekstu biti će korištena kratica RWA. Problem usmjeravanja za dane zahtjeve svjetlosnih puteva traži najkraću rutu između čvora izvorišta i odredišta na fizičkoj topologiji. Ponekad najkraća ruta nije moguća te se u obzir uzimaju i dulje, dok se ponekad rute uspostavljaju s obzirom na najmanje smetnje ili vrijeme. Svjetlosni putevi koji se ne mogu uspostaviti zbog ograničenja na potencijalnim rutama ili valnim duljinama se blokiraju. Uvijek su prisutna prije spomenuta dva ograničenja: jedinstvenosti i preklapanja valnih duljina.

Cilj rješavanja RWA problema jest minimizacija broja valnih duljina za postavljanje svjetlosnih puteva u fizičkoj topologiji. U slučaju da ima valnih pretvornika (nema ograničenja jedinstvenosti valnih duljina) RWA problem se može postaviti kao problem s protokom gdje je protok prisutan na svakoj niti. To odgovara cjelobrojnom linearnom programiranju gdje je cilj funkcije minimizirati protok u svakoj svjetlovodnoj niti, što zauzvrat minimizira broj svjetlosnih puteva koji prolaze nekom svjetlovodnom niti. Problem pripada NP razredu.

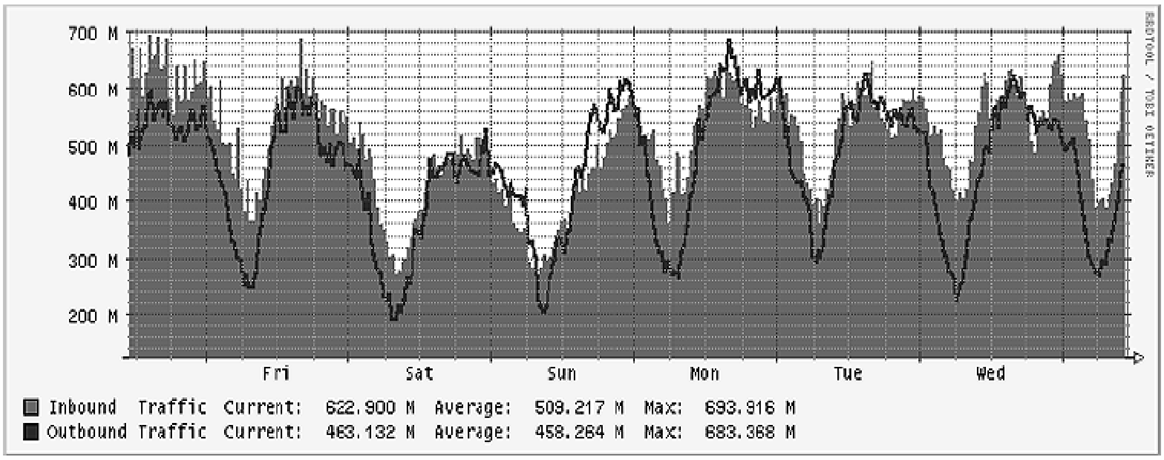
Kao što je već i prije rečeno ovaj diplomski rad rješava RWA problem s obzirom na ograničenje jedinstvenosti valnih duljina. Dijeli se na dva potproblema: usmjeravanje svjetlosnih puteva nad fizičkom topologijom i dodjeljivanje valne duljine putevima.

Sami RWA dijeli se na tri različite podvrste s obzirom na svjetlosne zahtjeve.

5.1. Vrste svjetlosnih zahtjeva

Statički zahtjevi (engl. *Static Lightpath Demands*) dolaze u obliku uređenih dvojki (s, d) gdje ***s*** označava izvorište a ***d*** označava odredište svjetlosnog puta. Koristi se termin svjetlosni zahtjev jer prije uspostavljanja svjetlosnog puta između dva krajnja čvora u zahtjevu treba doći do usmjeravanja i ne smije biti blokiran tj. prekršiti jedno od ograničenja. Skup zahtjeva svih statičkih zahtjeva je poznat prije uspostave neke mreže, a skup svjetlosnih puteva se uspostavlja na dulji vremenski rok.

Raspoređeni zahtjevi (engl. *Scheduled Lightpath Demands*) se sastoje od uređene četvorke oblika (s, d, α, ω) ili petorke oblika $(s, d, n, \alpha, \omega)$. Oznake ***s*** i ***d*** označavaju izvorišni i odredišni čvor, $\boldsymbol{\alpha}$ označava vrijeme uspostave svjetlosnog puta, $\boldsymbol{\omega}$ označava vrijeme raskida svjetlosnog puta, dok $\boldsymbol{n}$ označava broj zahtijevanih svjetlosnih puteva koji je važan u slučaju da zahtijevana brzina prijenosa bude veća od nazivne brzine svjetlosnog puta (obično 2,5 Gb/s ili 10 Gb/s). Pretpostavka u ovom diplomskom radu jest da će $\boldsymbol{n}$ uvijek biti jedan jer njegovo povećanje utječe na povećanje broja valnih duljina ali putem jednostavnog pribrajanja što ne utječe na povećanje složenosti programa niti zahtijeva upotrebu korištenih algoritama. Kod raspoređenih zahtjeva je poznat raspored uspostave i raskida svjetlosnih putova no treba napomenuti ako dva svjetlosna puta dijele istu svjetlovodnu nit, ali u različitim vremenskim intervalima (α, ω), mogu koristiti istu valnu duljinu što će biti pojašnjeno kasnije. Unutar mreža promet je predvidljiv jer raste ili pada unutar određenih perioda tj. vremenskih intervala. Recimo unutar radnog vremena između dva ureda nekih velikih kompanija. Slika 5. to zornije prikazuje čime se može pretpostaviti da u budućnosti većim razvojem WDM optičkih mreža najzastupljeniji će biti raspoređeni zahtjevi.



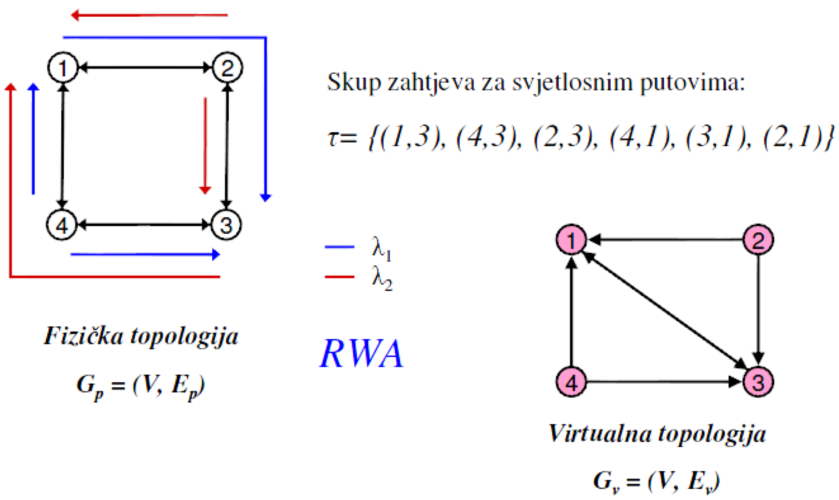
Slika 5. Promet na svjetlovodnim nitima između New Yorka i Washingtona Abilene mreže između 3. travnja 2003. i 10. travnja 2003. (2)

Dinamički zahtjevi (engl. *Dynamic Lightpath Demands*) su zahtjevi za svjetlosnim putovima koji pristižu dinamički sa slučajnim vremenima trajanja konekcije. Shodno tome potrebno je i dinamički raditi proces usmjeravanja svjetlosnih zahtjeva. Dinamički zahtjevi nisu predmet istraživanja ali moguće su detaljnije informacije u članku (3).

5.2. Potproblem usmjeravanja

Postoje četiri različite vrste usmjeravanja: određeno usmjeravanje (engl. *fixed routing*), određeno-izmjenično usmjeravanje (engl. *fixed-alternate routing*), adaptivno usmjeravanje (engl. *adaptive routing*) te usmjeravanje tolerantno na smetnje (engl. *fault-tolerant routing*). Određeno usmjeravanje traži najkraći put između dva čvora u nekoj optičkoj mreži. Kod određeno-izmjeničnog usmjeravanja svaki čvor u optičkoj mreži sadrži poredanu tablicu usmjeravanja s određenim putevima do svakog odredišnog čvora. Putevi u tablici mogu biti poredani s obzirom na najmanju moguću udaljenost, što znači da su putevi na primjer: najkraći put, drugi najkraći put, treći najkraći put, itd. Adaptivno usmjeravanje traži put između odredišnog i izvorišnog čvora dinamički, ovisno o stanju mreže. Stanje mreže je određeno skupom svih konekcija (ili svjetlosnih zahtjeva) koji su trenutno uspostavljeni i koriste se. Usmjeravanje tolerantno na smetnje traži puteve koji su neovisni jedan o drugome za jedan svjetlosni zahtjev. Neovisni svjetlosni putevi nekog svjetlosnog zahtjeva ne dijele niti jednu svjetlovodnu nit diljem svojeg puta.

U ovom diplomskom radu se koristi određeno usmjeravanje te će ono biti pobliže opisano, za ostale vrste usmjeravanja dodatne informacije su dostupne u (4). Kod određenog usmjeravanja za svaki svjetlosni zahtjev rješava se problem najkraćeg puta upotrebom Dijkstrinog ili Bellman-Fordovog algoritma. U programu upotrijebljenom za istraživanja u ovom diplomskom radu (RWA Solver) koristi se Bellman-Ford algoritam (detaljniji opis u kasnijem poglavlju). Svjetlosni putevi se uspostavljaju nad fizičkom topologijom provođenjem pronalaska najkraćih puteva za sve svjetlosne puteve. Iz samih svjetlosnih zahtjeva moguće je napraviti virtualnu topologiju kao što je prikazano na slici 5.



Slika 6. Dobivanje virtualne topologije putem svjetlosnih zahtjeva (5)

U samom programu RWA Solver stvara se novi graf istovjetan fizičkoj topologiji sa označenim svjetlosnim putevima.

5.3. Potproblem dodjeljivanja valnih duljina

Nakon odabira puta za svaki svjetlosni zahtjev, broj svjetlosnih puteva koji prolazi kroz svaku svjetlovodnu nit određuje zagušenje (engl. *congestion*) na toj svjetlovodnoj niti. Potrebno je dodijeliti valnu duljinu svakom svjetlosnom putu s obzirom na ograničenje preklapanja valnih duljina. Nedostatak valnih pretvornika iziskuje poštivanje ograničenja jedinstvenosti valnih duljina.

Dodjeljivanje valne duljine različitim svjetlosnim putevima, s obzirom na minimizaciju broja valnih duljina (odnosno boja) uz ograničenje jedinstvenosti valnih duljina svodi se na problem bojanja grafa.

Slijede matematičke definicije iz teorije grafova kako bi opis problema bojanja grafa bio razumljiviji. Definicije su preuzete iz (6) te su tamo i detaljnije opisane.

Definicija 1. Jednostavni graf G sastoji se od uređenog para $G = (V, E)$, skraćeno $G(V, E)$, sa sljedećim svojstvima:

- $V = \{v_0, \ldots, v_n\}$ je neprazan, konačan skup čije elemente zovemo vrhovi grafa G
- $E = \{(v_i, v_j)\}$ je konačan skup različitih dvočlanih (neuređenih) podskupova skupa V koje zovemo bridovi.

Definicija 2. Brid $e = \{v, u\}$ spaja vrhove v i u te se kraće piše vu. U toj situaciji kažemo da su vrhovi v i u grafa G susjedni. Također, vrhovi v i u su incidentni s bridom e.

Definicija 3. Skup vrhova koji su susjedni vrhu v zovemo susjedstvo vrha v i označavamo s oznakom $H(v)$.

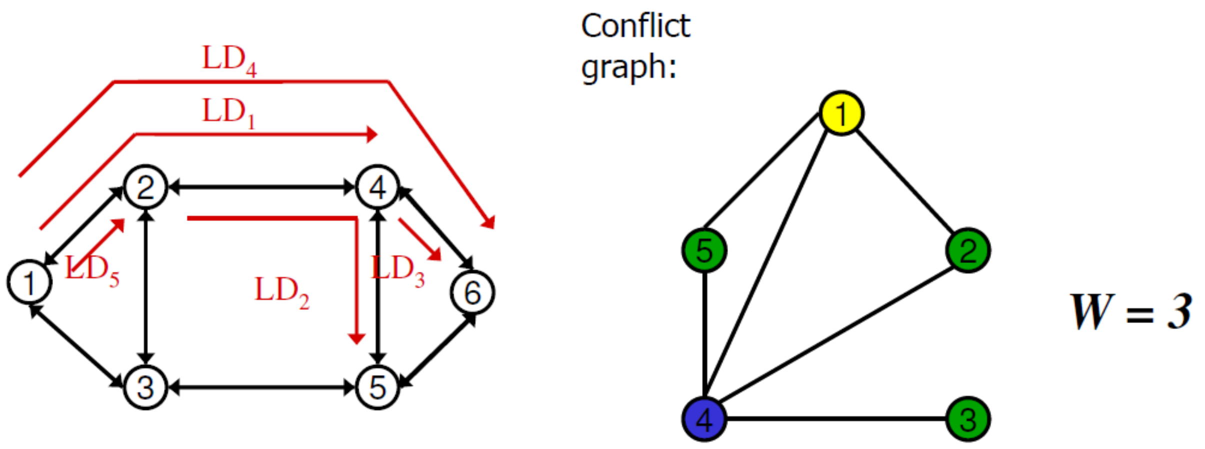
Definicija 2. Stupanj vrha v grafa G jednak je broju bridova koji su incidentni s v. Označavamo ga s $d(v)$. Najveći stupanj vrha u grafu označavamo s $\Delta(G)$.

Definicija 3. Za zadane disjunktne grafove $G_1 = (V(G_1), E(G_1))$ i $G_2 = (V(G_2), E(G_2))$ definiramo njihovu uniju $G_1 \cup G_2$, kao graf $G_1 \cup G_2 = \big(V(G_1) \cup V(G_2),\ E(G_1) \cup E(G_2)\big)$.

Definicija 4. Graf je **povezan** ako se ne može prikazati kao unija neka dva grafa. U suprotnom kažemo da je graf **nepovezan**. Svaki se nepovezani graf dakle može prikazati kao unija povezanih grafova.

Postupak dodjeljivanja valnih duljina je sljedeći:

1. Konstruira se graf $G(V, E)$ tako da svaki svjetlosni put u cijeloj fizičkoj topologiji je predstavljen vrhom u grafu G. Između dvaju vrhova postoji dvosmjerni brid ako svjetlosni putevi predstavljeni tim čvorovima prolaze istom svjetlovodnom niti. Taj graf naziva se konfliktnim grafom (engl. *conflict graph*).
2. Potrebno je obojiti vrhove grafa G, tako da vrhovi povezani bridom nisu iste boje.



Slika 7. Svjetlosni putevi te konfliktni graf nastao iz njih (5)

Problem je detaljnije opisan te riješen sekvencijalnim algoritmom za bojanje u (4).
Problem spada u NP-razred[1], stoga je teško odrediti minimalni broj boja potreban kako bi se obojio graf G (taj broj se još naziva kromatskim brojem $\chi(G)$ grafa).

Formalna definicija problema slijedi:

Definicija 5. Neka je definirana funkcija $c : V \rightarrow N$ koja pridružuje svakom vrhu grafa G prirodan broj $c(V)$ kojeg ćemo zvati boja vrha. Pridruživanje boja vrhovima grafa, odnosno funkciju c, zovemo bojanje vrhova grafa ili kraće bojanje grafa.

Definicija 6. Bojanje vrhova grafa G s najviše k boja zovemo ***k*-bojanje** grafa G.

Definicija 7. Legalno k-bojanje grafa G je bojanje grafa G s najviše k boja tako da su susjednim vrhovima pridružene različite boje.

U daljnjem tekstu problem legalnog k-bojanja grafa G će se radi jednostavnosti možda neki put pisati kao "problem bojanja" grafa G.

Definicija 8. Graf G je k-obojiv ako postoji legalno k-bojanje od G.

Definicija 9. Ako je graf G k-obojiv ali nije $(k-1)$ obojiv, kažemo da je G k−kromatski, odnosno kažemo da je kromatski broj od G jednak k i pišemo $\chi(G) = k$.

[1] NP-razred predstavlja probleme odluke koje je moguće riješiti nedeterminističkim strojem u polinomijalnom vremenu

Definicija 10. Particiju skupa $V(G)$ na k disjunktnih nepraznih podskupova $V_1, \ldots, V_k$ tako da vrijedi $V(G) = \bigcup_{i=1}^{k} V_i$ zovemo k-dioba grafa G. Ako su podskupovi $V_1, \ldots, V_k$ ujedno i nezavisni skupovi onda se radi o legalnoj k-diobi grafa G.

Podskupove $V_1, \ldots, V_k$ u daljnjem tekstu zovemo klase boja.

Sljedeći teoremi navode gornje ograde za kromatski broj proizvoljnog grafa.

Teorem 1. (Brooks) Za proizvoljan graf G vrijedi:

$$\chi(G) \leq \Delta(G) + 1 \tag{2.1}$$

Nadalje, ako je G povezan, tada u gornjem izrazu vrijedi jednakost u slučaju da je G potpun graf ili neparan ciklus.

Teorem 2. (Stacho) Kromatski broj proizvoljnog grafa G zadovoljava nejednakost:

$$\chi(G) \leq \Delta_2(G) + 1 \tag{2.2}$$

gdje je:

$$\Delta_2(G) = \max_{u \in V(G)} \max_{\substack{v \in H(u) \\ d(v) \leq d(u)}} d(v)$$

Dakle, $\Delta_2(G)$ je najveći stupanj vrha v u grafu G uz uvjet da je v susjedan barem jednom vrhu u čiji je stupanj veći ili jednak stupnju od v.

Izraz (2.2) daje strožu gornju ogradu od izraza (2.1) ako nijedna dva vrha najvećeg stupnja nisu susjedna.

5.4. Impairment-Aware RWA

U sve-optičkim mrežama[2] transmisijske smetnje zbog neidealnih optičkih komponenti u fizičkom sloju mogu znatno utjecati na performanse mreže. U sve-optičkoj mreži optičko-elektroničko-optičke (engl. *optical-electrical-optical, OEO*) pretvorbe signala se ne upotrebljavaju u čvorovima između odredišta i izvorišta konekcije, čime se postiže potencijalno manja cijena mreže. Takve mreže se nazivaju transparentnima (engl. *transparent*) jer u njima signal s podacima ostaje u optičkoj domeni diljem cijelog svjetlosnog puta. Ali, kvaliteta optičkog signala degradira putujući kroz optičke komponente prisutne unutar svjetlosnog puta, te u stvarnom svijetu transparentne mreže su teže izvedive jer bez regeneracije signala može doći do prevelike degradacije signala. Postoje i djelomično transparentne (engl. *translucent*) optičke mreže koje imaju mogućnost regeneracije tj. OEO pretvorbe signala na nekim čvorovima. Fizička veličina transparentne mreže je većinom određena o smetnjama kao što su: prigušenje, šum (engl. *noise*), preslušavanje (engl. *crosstalk*), kromatičko/polarizacijski tip disperzije (engl. *chromatic/polarization-mode dispersion*, CD/PMD), nelinearne smetnje niti (engl. *fiber non-linearities*), konkatenacija filtera (engl. *filter concatenation*), polarizacijski ovisan gubitak/pojačanje (engl. *polarization-dependent loss/gain*, PDL/PDG), prijelazne pojave signala (engl. *signal transients*) i druge. Stoga se smetnje u transmisiji putem fizičkog sloja u sve-optičkom usmjeravanju svjetlosnih puteva trebaju uzeti u obziru slučaju kasnijeg predviđanja realistične mreže.

Pri planiranju i izradi modela potencijalnih mreža razlikujemo dvije vrste modela koji se tiču WDM mreža s valnim usmjeravanjem: idealne i realistične mreže. U idealnim se smatra da su sve komponente optičke mreže zajedno s nitima slobodne od mogućih grešaka pri prijenosu uzrokovanih smetnjama ili kvarovima. U slučaju realističnih mreža u obzir se uzimaju i smetnje. S obzirom na model mreže koji se upotrebljava koriste se zasebni tipovi RWA algoritma.

Svjetlosni putevi na svome putu sadrže više neidealnih optičkih komponenti koje uzrokuju smetnje i degradaciju signala, ovisno o transparentnosti mreže do regeneracije signala ne

[2] Optičke mreže s valnim usmjeravanjem gdje informacijski put između odredišta i izvorišta ostaje u optičkoj domeni. Takav sve-optički put pruža protokolarnu transparentnost mreži.

mora doći. Smetnje se dijele na linearne i nelinearne te uzrokuju šumove i distorziju signala što dovodi do njegove degradacije. Ako je kvaliteta signala na izvorišnom čvoru izrazito loša, broj krivo prenesenih bitova (engl. *bit-error rate,* BER) je velik. Stoga je svjetlosni put neupotrebljiv i blokira se. Ta se pojava naziva blokiranje u fizičkom sloju (engl. *physical-layer blocking*).

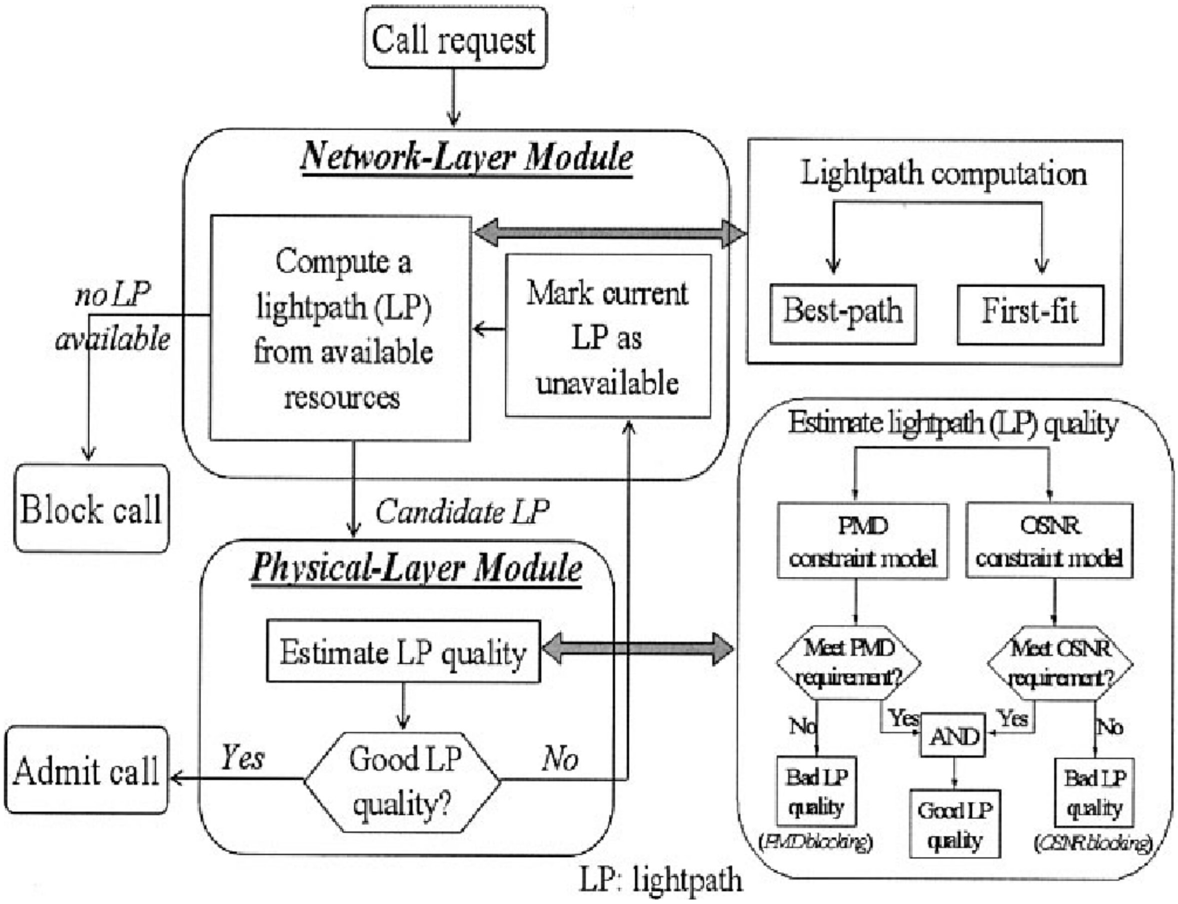
Neke od linearnih smetnji su: šum pojačala (engl. *amplifier noise*), polarizacijski tip disperzije niti (engl. *fiber polarization-mode dispersion,* PMD), disperzija grupne brzine (engl. *group velocity dispersion*, GVD), preslušavanje komponenti (engl. *component crosstalk*), itd.

Dok su neke nelinearne smetnje: četverovalno miješanje (engl. *four-wave mixing*, FWM), samofazna modulacija (engl. *self-phase modulation*, SPM), modulacija faze presjeka (engl. *cross-phase modulation*, XPM), raspršenje (engl. *scattering*).

Kada su u pitanju transmisijske smetnje , selekcija puta i valne duljine u odnosu na optički omjer signala i šuma (engl. *optical signal-to-noise-ratio,* OSNR) je nužna. Statički efekti i nelinearne smetnje mogu znatno utjecati na vjerojatnost blokiranja. Statički efekti ovise o fizičkoj konfiguraciji i moraju se uzeti u obzir za bilo koju količinu prometa u mreži. Nelinearne smetnje degradiraju kvalitetu transmisije kada je broj uspostavljenih svjetlosnih puteva velik, tj. kada je količina prometa velika. Smetnje u transmisiji mogu postati znatno izraženije pri velikim brzinama prijenosa, posebno iznad 40 Gb/s i više.

Prikladan parametar kao kriterij za evaluaciju kvalitete signala svjetlosnog puta je BER jer obuhvaća sve smetnje. BER nije dostupan prije nego što se svjetlosni put uspostavi. Stoga se BER računa unaprijed pomoću prije definiranih statističkih vrijednosti nekih smetnji. U obzir se najčešće uzimaju polarizacijski tip disperzije niti (PMD) te optički omjer signala i šuma (OSNR). Dodatne smetnje se mogu uzimati u obzir kada je potrebno. Na slici 8. vidljiv je postupak blokiranja u fizičkom sloju te je po tom načelu rađen i algoritam u ovom radu.

Daljnje informacije o osnovnim smetnjama u WDM optičkim mrežama dostupne su u (4).



Slika 8. Model RWA algoritma svjesnog smetnji (4)

Potrebno je napomenuti da prag za prihvatljivu vrijednost BER-a se može pretvoriti u prag za prihvatljivu vrijednost tzv. Personickovog Q faktora ili jednostavnije faktora kvalitete Q. Faktor kvalitete Q, nadalje Q faktor, se može evaluirati kao funkcija parametra sustava u kojem se vrši transmisija (tj. mreže kao sustav ispitivan u ovom radu) i smetnji nastalih pri transmisiji. U slučaju da na odredišnom čvoru nema mehanizama za
ispravljanje digitalnog signala, Q faktor od 16,9 dB približno odgovara BER-u od 10^{-12}.

Odnos između Q faktora i BER-a je određen formulom (7):

$$BER = \frac{1}{2} erfc\left(\frac{Q}{\sqrt{2}}\right) \tag{2.3}$$

U ovom diplomskom radu za izračun kvalitete signala i njegovu upotrebu u blokiranju u fizičkom sloju biti će korišten Q faktor. Upotreba i izračun Q faktora unutar aplikacije razvijene za rješavanje RWA problema biti će opisan u kasnijem poglavlju.
Više informacija i detaljniji opis BER-a i njegovog računanja u (8).

6. RWA algoritmi

Unutar ovog diplomskog rada za potrebe rješavanja RWA problema upotrijebljeno je više algoritama. Za potrebe rješavanja problema usmjeravanja koji se svodi na problem najkraćeg puta korišten je Bellman-Ford algoritam. Za problem dodjeljivanja valnih duljina korišteni su: hibridni evolucijski algoritam (HEA), te varijabilna pretraga susjedstva (engl. *variable neighborhood search*, VNS).

6.1. Bellman-Ford algoritam

Problem najkraćeg puta spada u P razred[3] složenosti. Složenost samog Bellman-Ford algoritma jest $O(|V| * |E|)$, gdje $|V|$ označava broj vrhova grafa, a $|E|$ označava broj bridova grafa. Za razliku od Dijkstrinog algoritma, Bellman-Ford algoritam ima mogućnost detekcije negativnih ciklusa u grafu.

Pseudokod Bellman-Ford algoritma je naveden u slici 9.

[3] problemi odluke koje je moguće riješit determinističkim strojem u polinomijalnom vremenu (tj. složenost najgoreg slučaja je ograničena polinomijalnom funkcijom)

```
procedure BellmanFord(list vertices, list edges, vertex source)
  //Ova implementacija nad grafom, koji se sastoji od liste vrhova i bridova, izmjeni vrhove
  // tako da njihove udaljenosti i atributi koji predstavljaju njihovog prethodnika
  // sadržavaju najkraći put

  // Korak 1: inicijalizacija grafa
  for each vertex v in vertices:
    if v is source then v.distance := 0
    else v.distance := infinity
    v.predecessor := null

  // Step 2: upotrba bridova za najkraći put
  for i from 1 to size(vertices)-1:
    for each edge uv in edges: // uv is the edge from u to v
      u := uv.source
      v := uv.destination
      if u.distance + uv.weight < v.distance:
        v.distance := u.distance + uv.weight
        v.predecessor := u

  // Step 3: provjera negativnih ciklusa
  for each edge uv in edges:
    u := uv.source
    v := uv.destination
    if u.distance + uv.weight < v.distance:
      error "Graf sadrži negativni ciklus"
```

Slika 9. Pseudokod Bellman-Ford algoritma (9)

6.2. Hibridni evolucijski algoritam

Upotrebom lokalne pretrage unutar populacijski orijentiranih evolucijskih algoritama dobivaju se hibridni evolucijski algoritmi (engl. hibrid evolutionary algorithms, HEA). Glavni dijelovi algoritma su lokalna pretraga (engl. *local search*, LS) i specijalizirani operator križanja. Nakon početne inicijalizacije neke populacije križanjem dviju jedinki dobiva se nova jedinka koja se operatorom lokalne pretrage dodatno poboljšava. Svaka jedinka predstavlja potencijalno rješenje. Daljnji dio teksta, pseudokodova te detaljnije informacije o hibridnom evolucijskom algoritmu moguće je pronaći u (10) te i u (6). Kako bi hibridni evolucijski algoritam bio lakše razumljiv slijedi objašnjenje genetskih algoritama i lokalne pretrage.

6.2.1. Genetski algoritmi

Genetski algoritmi služe kao tehnika pretraživanja u računarstvu pri pronalaženju točnih ili aproksimiranih rješenja u optimizaciji ili problemima pretraživanja. Inspirirani su Darwinovom teorijom evolucije.

Zajedničke karakteristike genetskih algoritama su biološki procesi koje simuliraju, kao: selekcija, mutacija, križanje i nasljeđivanje.

Implementiraju se u računalnim simulacijama u kojima je populacija apstraktna reprezentacija (kromosomi ili genotipovi nekog genoma) potencijalnih rješenja (jedinke) nekom problemu optimizacije koji evoluira prema boljim rješenjima. Uobičajeno, rješenja se prikazuju pomoću nizova nula i jedinica, ali i druga kodiranja su moguća. Sama evolucija započinje od neke slučajno odabrane, početne populacije i odvija se u generacijama.

U svakoj generaciji uzima se dobrota (*engl. fitness*) svake jedinke (*engl. individual*) te se evaluira. Slijedi odabir tj. selekcija više jedinki po vrijednosti njihove dobrote iz trenutne populacije. Odabrane jedinke se križaju i po mogućnosti mutiraju kako bi se formirala nova populacija. Novoformirana populacija se koristi u sljedećoj iteraciji samog algoritma. Algoritam prestaje s radom nakon što dobrota neke populacije dosegne zadovoljavajuću vrijednost, ili nakon određenog broja generacija. Ako se dosegne broj generacija koje su određene za prestanak rada algoritma moguće je i dobivanje i nedobivanje zadovoljavajućeg rješenja. Navedeni opisani postupak rada genetskog algoritama je vrlo sličan postupku rada evolucijskog algoritma pošto je genetski algoritam podskup evolucijskog algoritma.

Potpoglavlje je preuzeto iz (11). Detaljnije o genetskim algoritmima moguće je naći u (12).

6.2.2. Lokalna pretraga

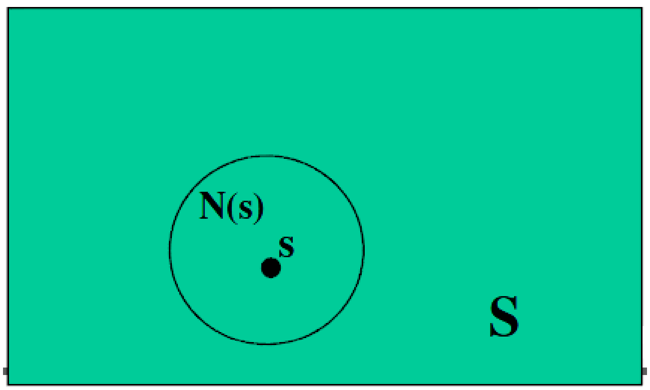
Lokalna pretraga je metaheuristička metoda za rješavanje računski složenih optimizacijskih problema. Često se koristi kao dio kompleksnijih metaheuristika zasnovanih na pojedinačnom rješenju. Sastavni elementi lokalne pretrage su: susjedstvo, početno rješenje i odabir susjeda. Pseudokod lokalne pretrage vidljiv je na slici 10.

Započni lokalno pretraživanje
Generiraj početno rješenje (trenutno rješenje)
Radi
    Generiraj skup kandidata rješenja susjednih trenutnom rješenju
    Izaberi boljeg susjeda kao novo trenutno rješenje, ako postoji. Ako ne, stani.
Dok je pronađeno bolje rješenje
Kraj

Slika 10. Pseudokod lokalne pretrage

Vidljivo je da lokalna pretraga spada u iterativni algoritam koji kroz iteracije stalno pokušava pronaći bolje rješenje. Potezi koji dovode do boljeg rješenja spadaju u susjedstvo trenutnog rješenja.

Definicija 10. Susjedstvo $N(s)$ rješenja s u S je skup rješenja $N(s) \subset S$, gdje se svako rješenje $s' \in N(s)$ može dobiti iz s primjenom neke elementarne transformacije/perturbacije, koja se zove **pomak**, na s.



Slika 11. Susjedstvo N(s)

Odabir susjednog rješenja određuje mogućnost kvalitete rezultata, a načini na koji se susjedna rješenja biraju je velik. Pojam susjedstva i tehnike odabira susjeda tj. susjednog rješenja biti će detaljnije obrađene u varijabilnoj pretrazi susjedstva.

6.2.3. Tabu pretraga

Tabu pretraga (TP) je matematička optimizacijska metoda koja pripada klasi lokalnih pretraga. Tabu pretraga iterativnim postupkom pretražuje susjedstva u prostoru rješenja, nakon pronalaska potencijalnog rješenja, ono se označuje kao tabu (nedostupno ili zabranjeno) kako se pretraga ne bi vratila na to rješenje. Rješenja koja su zabranjena se pamte u tabu listi najčešće na određeno „vrijeme“ ili određeni broj poteza.

Tabu lista (TL) je memorijska struktura pomoću koje se određuje koja su rješenja iz prostora rješenja trenutno zabranjena (imaju tabu status). Tabu vrijeme određuje broj iteracija za vrijeme kojih potencijalno rješenje ne može biti ponovno odabrano. Cilj tabu liste je izbjegavanje zapinjanja u lokalnim optimumima i kruženja između susjednih rješenja.

Tabu pretraga koristi lokalnu pretragu, ali u slučaju da naiđe na lokalni optimum dopušta i poteze koji u obzir uzimaju rješenja koja se ne poboljšavaju. Pronađeni lokalni optimum (rješenje) pamti se u tabu listi, a povratak na njega spriječen je korištenjem tabu liste.

Uporabom tabu lista postoji opasnost da se brani potencijalno rješenje koje je ujedno i optimalno tj. prihvatljivo rješenje. Stoga je uvijek dobro definirati kriterij prihvaćanja, koji privremeno ukida tabu status poteza. Vrlo često korišten kriterij prihvaćanja omogućuje prihvaćanje potencijalnog rješenja ukoliko je ono bolje od trenutno najbolje pronađenog.

Kao i kod lokalne pretrage, tehnike odabira susjednog rješenja poboljšavaju učinkovitost pretrage potencijalnih rješenja, zajedno uz prostor pretrage.

6.2.4. Opis hibridnog evolucijskog algoritma za bojanje

Korišteni hibridni evolucijski algoritam se najbolje može opisati na pseudokodu:

***Hibridni Algoritam za Bojanje (HAB)*:**

Ulaz: graf $G = (V, E)$, cijeli broj $k > 0$

Izlaz: najbolje pronađeno rješenje x^*

Početak:

P = *InicijalizirajPopulaciju*(|*P*|);

dok (!*UvjetZaustavljanja*()) radi

{

(x_1, x_2) = *OdaberiRoditelje*(*P*);

x = *OperatorKrižanja*(x_1, x_2);

x = *LokalnaPretraga*(*x*, *L*);

P = *AžurirajPopulaciju*(*P*, *x*)

}

Kraj.

Slika 12. HEA pseudokod (6)

U početku algoritma potrebno je inicijalizirati populaciju predeterminirane veličine. Za svaku jedinku u populaciji potrebno je ispuniti klase boja neke jedinke vrhovima koji tu boju onda i poprimaju. Klasa boje predstavlja skup

Broj klasa boja određuje se pronalaskom nekog legalnog k-bojanja za neki (dovoljno velik) broj boja $k = k_0$. Zatim, kada je legalno k-bojanje pronađeno isti se algoritam uzastopno koristi za pronalazak k-bojanja sa sve manjim brojem boja $(k = k_0 - 1, k_0 - 2, \ldots)$. Na ovaj način problem bojanja grafa sveden je na uzastopno rješavanje sve težih problema k-bojanja. Ispunjavanje jedinke vrši se provjerom nepovezanosti vrhova (kasnije u tekstu definiranih kao konfliktni vrhovi). Ako neki vrhovi ne pripadaju istom bridu dodjeljuju se istoj klasi boje. Naravno pošto želimo minimizirati broj boja, te je broj klasa uvijek manji od broja vrhova, pri malom broju klasa boja neki vrhovi će ostati nedodijeljeni te se oni dodjeljuju nasumično.

Nakon inicijalizacije slijedi iterativna pretraga prostora rješenja unutar petlje čiji se uvjet zaustavljanja odredi po volji, u slučaju za ovaj diplomski rad oba algoritma korištena u programu iteriraju u petlji tražeći legalno k-bojanje te ako nakon određenog broja iteracija ne dolazi do poboljšanja rješenja, algoritam prekida s radom. Nakon svake iteracije rješenje se evaluira i određuje broj konfliktnih bridova u njemu, te ako je broj konfliktnih bridova manji u trenutnom rješenju od rješenja iz prošle iteracije, došlo je do poboljšanja rješenja. Ako postoji barem jedan konfliktni brid, rješenje nije legalno k-bojanje.

Unutar petlje čija svaka iteracija predstavlja jednu generaciju jedinki, prvo se radi odabir roditelja, u pseudokodu označeno metodom OdabirRoditelja(P), gdje se roditelji biraju nasumično unutar populacije. Roditelji moraju biti različite jedinke.

Iz odabranih roditelja križanjem se dobiva dijete. Operator križanja jest pohlepno križanje particija (engl. *greedy partition crossover*), a postupak algoritma vidljiv je u pseudokodu[4]:

Operator križanja (PKP):

Ulaz: rješenja $x_1 = \{V_1^1, \ldots, V_k^1\}$ i $x_2 = \{V_1^2, \ldots, V_k^2\}$

Izlaz: rješenje $x = \{V_1, \ldots, V_k\}$

Početak:

za j ($1 \le j \le k$)
{
ako (j neparan), onda $A := 1$, inače $A := 2$;
odaberi i takav da je klasa V_i^A maksimalnog kardinaliteta;
$V_j := V_i^A$;
odstrani sve vrhove iz klase V_j iz rješenja x_1 i x_2;
}
Nasumce dodijeli klasama preostale vrhove iz $V - (V_1 \cup \ldots \cup V_k)$;

Kraj.

Slika 13. Pseudokod operatora križanja PKP

Dijete koje se dobilo križanjem prolazi kroz lokalnu pretragu. Za lokalnu pretragu u korištenom programu upotrijebljen je Tabucol algoritam(17)(18). Tabucol je implementacija tabu pretrage za problem k-bojanja grafova. Algoritam iterativno mijenja klasu boje jednog (ne uvijek istog) vrha, a cilj je postepeno smanjivati ukupan broj konfliktnih bridova dok legalno bojanje nije uspostavljeno. Tabu lista se koristi u svrhu izbjegavanja lokalnih optimuma te kako bi se izbjeglo kratkoročno kruženje (u kratkim ciklusima). Tabucol korišten u ovom programu radi jednu vrstu transformacija jedinki tako da iz susjedstva odabere jednog susjeda. Ta transformacija označava promjenu klase boje jednog vrha unutar jedne jedinke i još se naziva *1-potez*. Susjed se razlikuje od trenutnog rješenja različitim razmještajem jednog vrha u klasi boje, znači razlikuje se za jednu transformaciju (1-potez). Susjedstvo $N(x)$ rješenja $x \in S$ je definirano kao skup k-bojanja koja se mogu dobiti iz x

[4] Kardinalitet klase se odnosi na broj čvorova koje sadrži.

primjenom jednog *1-poteza*. Rješenje x može biti neprihvatljivo ako sadrži konfliktne vrhove. Konfliktni vrhovi pripadaju istom bridu unutar neke klase boje. Transformacija koja premješta konfliktni vrh iz jedne klase boje u drugu naziva se *kritični 1-potez*. Broj konfliktnih bridova određuje dobrotu neke klase boje, dok zbroj dobrota klasa boja određuje dobrotu jedinke.

Radi učinkovitosti, Tabucol izvodi samo kritične 1-poteze. Primjenom *1-poteza* na neko rješenje x, u tabu listu se zapisuje taj potez kao par vrha i klase boje (v, V_i), čime se zabranjuje dodjela te boje tom vrhu na određeni broj iteracija. Trajanje zabrane tom 1-potezu je određen brojem konfliktnih vrhova u rješenju x, i o parametrima A i b. Parametar A se bira slučajno iz intervala $[0,9]$ dok parametar b poprima vrijednost 0,6. U slučaju da je 1-potez kritičan i nije označen kao tabu onda se on smatra potencijalnim 1-potezom. Tabucol funkcionira birajući najprihvatljiviji potencijalni 1-potez ako ih postoji više i time osigurava pronalazak legalnog k-bojanja. Potencijalni 1-potez je najprihvatljiviji ako je njegova dobrota $\delta(v, V_i)$ najveća od svih mogućih potencijalnih 1-poteza. Dobrota poteza definirana je kao $\delta(v, V_i) = f(x) - f(x + (v, V_i))$, gdje je $f(x) = \sum_{i=1}^{k} |E_i|$. Detaljnije to znači da je dobrota jedinke $f(x)$ određena brojem bridova čija su oba vrha unutar iste klase boje (konfliktni bridovi), $|E_i|$. Parametar k unutar sumacije označava broj klasa boja. Najbolji potencijalni 1-potezi su oni koji premještajem nekog konfliktnog vrha iz jedne klase boje u drugu smanjuju broj postojećih konfliktnih bridova ne stvarajući nove konfliktne bridove. Ako više potencijalnih 1-poteza ima istu dobrotu onda se jedan od njih bira nasumično. Algoritam se zaustavlja kada za neko rješenje dobije dobrotu $f(x) = 0$ te samim time to rješenje predstavlja legalno k-bojanje.

Slijedi pseudokod Tabucol algoritma:

Algoritam Tabucol:

Ulaz: graf $G = (V, E)$, cijeli broj $k > 0$

Parametri: *MaxIter*, *A* i *b*

Izlaz: rješenje x^*

Inicijalizacija:

postavi: $x^* := x$, $iter = 0$, $TL := \emptyset$;

Pretraga:

```
dok (f(x) > 0 AND iter <= MaxIter) radi
{
    postavi  iter := iter + 1;
    odaberi najbolji potencijalni 1-potez (v, i);
    stavi 1-potez (v, c(v)) u TL na A + b*F(x) iteracija;
    postavi  x := x + (v, i);
    ako je f(x) < f(x*), onda postavi  x* := x;
}
vrati rješenje x*;
```

Slika 14. Pseudokod Tabucol algoritma (6)

MaxIter označava najveći mogući broj iteracija dok $F(x)$ označava broj konfliktnih vrhova u rješenju x.

Detaljnije o operatoru križanja, Tabucol algoritmu i ažuriranju populacije može se naći u (6) (10).

Nakon Tabucol algoritma populaciju je potrebno ažurirati. Ažuriranje obuhvaća zamjenu jednog od dvaju roditelja njegovim djetetom. Zamjenjuje se lošiji roditelj, a u slučaju da su roditelji jednake dobrote, odabire se nasumični roditelj za zamjenu. Ovim postupkom najbolja jedinka u populaciji ostaje očuvana. To se svojstvo još naziva i elitizam i svojstvo je genetskih algoritama.

6.3. Varijabilna pretraga susjedstva

Neka N^t , $t = (1, \dots, t_{max})$ označava zatvoreni skup susjedstva gdje je $N^t(s)$ skup rješenja (susjedi od s) u t susjedstvu od s. Većina metoda lokalne pretrage u obzir uzimaju samo jedan tip susjedstva, tj. $t_{max} = 1$. Osnovni tip metode varijabilne pretrage susjedstva (engl. variable neighborhood search, VNS) korištenjem više od jednog susjedstva izbjegava zaglavljivanje u lokalnom optimumu. Algoritam VNS spada u metaheuristiku koja istraživanjem više susjedstva pronalazi bolja rješenja. Rješenje u drugom susjedstvu se uzima u obzir samo ako je bolje od trenutno najboljeg rješenja.

1. **Inicijalizacija**

   1.a. Nasumično stvori k-bojanje grafa G

   1.b. Postavi $I_{VNS} = 0$ i $t = 1$

   1.c. Nasumično stvori permutaciju π iz skupa $\{1,2,3,4,5,6\}$, koja označava susjedstva koja će se upotrebljavati

2. **Dok je $I_{VNS} <> |V|$**

   2.a. Postavi $I_{VNS} = I_{VNS} + 1$

   2.b. *Ispitivanje susjedstva*. Nasumično stvori rješenje s' u susjedstvu $N^{(\pi(t))}$

   2.c. *Lokalna pretraga*. Primjeni Tabucol algoritam nad rješenjem s' koji se zaustavlja nakon izvršavanja $5 * |V|$ 1-poteza bez poboljšanja najboljeg poznatog rješenja. Nakon primjene Tabucola s'' je najbolje poznato rješenje.

   2.d. *Izmijeni ili ne*. Ako je s'' bolji od rješenja s, onda postavi $s = s''$, $t = 1$, $I_{VNS} = 0$; Inače postavi $t = t + 1$ ako je I_{VNS} djeljiv sa $\left\lceil \frac{|V|}{6} \right\rceil$

Slika 15. Pseudokod VNS algoritma za problem bojanja grafa (13)

Osnovni VNS je detaljnije opisan u (14). U ovom radu VNS algoritam se koristi za problem bojanja grafa te je i prilagođen tom problemu. Pseudokod na slici 15. detaljnije prikazuje VNS algoritam. Kao i HEA u prijašnjem poglavlju VNS algoritam se sastoji od particija tj. klasa boja koje sadrže vrhove grafa. Razlika između HEA i VNS algoritma leži u nepostojanju populacije i bioloških procesa kod VNS-a. Za potrebe VNS algoritma u obzir će se uzimati isti način evaluacije dobrote jedinke ($f(x) = \sum_{i=1}^{k}|E_i|$) te Tabucol algoritam korišten kao metoda lokalne pretrage radi na način sličan onome u HEA-u. Pri inicijalizaciji VNS algoritma nasumično se stvori neko rješenje koje predstavlja bojanje grafa G. Nakon toga stvara se permutacija susjedstva u koraku 1.c., permutacija se sastoji od jedinstvenih šest brojeva u intervalu $[1,6]$ gdje svaki prirodni broj predstavlja jedno od susjedstva opisano kasnije u ovom poglavlju (broj 1 označava ulančano susjedstvo, broj 2 označava granatno susjedstvo, itd.) te se stvara nasumično. Susjedstva koja su korištena u VNS algoritmu odabrana su zbog najboljih rezultata od svih susjedstva koja su ispitana u (13). Autori članka su odabrali šest susjedstva, od kojih tri pripadaju susjedstvima vrhova, a druga tri pripadaju susjedstvima klasa boja. Glavna petlja algoritma se izvršava dok I_{VNS} nije jednak broju čvorova grafa. Ispitivanje susjedstva nad početno stvorenim rješenjem izvršava transformacije s obzirom na koje susjedstvo se bira iz $\pi(t)$. Nakon toga slijedi primjena Tabucol algoritma kako bi se dobio lokalni optimum s''. Na kraju ako je s'' bolji od prijašnjeg najboljeg rješenja s onda ono postaje trenutno najbolje rješenje. Detaljnije o algoritmu i ostalim susjedstvima koja se u njemu ne primjenjuju moguće je naći u (13).

Korištena susjedstva su:

- **Ulančano susjedstvo** (engl. *chain neighborhood*) – iz rješenja s stvara se susjedno rješenje (susjed). Odabire se nasumično konfliktni vrh x (izvorni vrh) te se prebacuje u drugu najbolju moguću klasu boje V_j. Pošto je s lokalni optimum, taj kritični 1-potez će stvoriti nove konfliktne vrhove u V_j. Tada se slučajnim odabirom bira novi konfliktni vrh $y \in V_j$ te se pridjeljuje drugoj najboljoj mogućoj boji l. Time opet postoji mogućnost stvaranja novih konfliktnih vrhova u V_l. Ovime imamo niz promjena (jedna vrsta ulančanih poteza) gdje se mora paziti da se boja nekog vrha mijenja samo jednom. Taj proces ulančanih 1-poteza se izvršava više puta, što je određeno s brojem odabranih izvornih vrhova i, koji se bira nasumično unutar $\left[1, i_{max}^{(1)}\right]$. Gornja ograda $i_{max}^{(1)}$ se postepeno smanjuje od 20 prema 5 kako se I_{VNS} povećava od 0 prema $|V|$.

- **Granatno susjedstvo** (engl. *grenade neighborhood*) – Odabire se nasumično konfliktni vrh x (granata) te se prebacuje u drugu najbolju moguću klasu boje V_j. Tada se redom svaki konfliktni vrh iz klase boje V_j prebacuje u drugu najbolju moguću klasu boje. Ovaj se proces ponavlja sa i različitih granata gdje se i bira nasumično unutar $\left[1, i_{max}^{(2)}\right]$. Gornja ograda $i_{max}^{(2)}$ se postepeno smanjuje od 40 prema 1 kako se I_{VNS} povećava od 0 prema $|V|$.
- **Vatrometno susjedstvo** (engl. *firework neighborhood*) – Odabire se nasumično konfliktni vrh x (vatromet) te se prebacuje u drugu najbolju moguću klasu boje V_j. Tada se svaki konfliktni vrh smatra granatom i time se slijedi postupak iz granatnog susjedstva. Ovaj se proces ponavlja sa i različitih vatrometa gdje se i bira nasumično unutar $\left[1, i_{max}^{(3)}\right]$. Gornja ograda $i_{max}^{(3)}$ se postepeno smanjuje od 30 prema 1 kako se I_{VNS} povećava od 0 prema $|V|$.
- **Ispunjavajuće susjedstvo** (engl. *empty–refill neighborhood*) - prvo se ispražnjava klasa boje V^* postepenim micanjem svakog vrha u drugu najbolju moguću klasu boje. Tada ispunimo klasu boje V^* postepenim odabirom p drugih vrhova (po mogućnosti konfliktnih vrhova) koje sve mičemo u klasu boje V^*.
- **Susjedstvo stabilnih skupova** (engl. *stable set neighborhood*) - prvo se nasumično odabere konfliktni vrh $x \in V^*$, te se radi poredana lista $L = \left(v_1, \dots, v_{|V|}\right)$ svih vrhova u skupu vrhova V grafa G, gdje je $v_1 = x$, $p-1$ zadnjih vrhova liste L su vrhovi

$V^* - \{x\}$, tj. svi vrhovi klase boje V^* osim vrha x. Elementi liste $(v_2, \dots, v_{|V|-p})$ su preostali vrhovi iz skupa V u nasumičnom redoslijedu. Tada se konstruira maksimalni stabilni skup W na pohlepni način, pregledavajući poredanu listu L. Na kraju se premještava svaki vrh koji je u V^*, a nije u W u najbolju moguću klasu boje. Dok se u V^* premještava svaki vrh koji je u W a nije u V^*.

- **Susjedstvo prazne klase** (engl. *empty class neighborhood*) - prvo se redom svaki vrh iz klase boje V^* prebacuje u drugu najbolju moguću klasu boje. Time se dobiva particija svih čvorova V u $k-1$ klasa boja. Tada se primjeni *Tabucol* algoritam (zaustavlja se nakon izvršavanja $|V|$ *1-poteza* bez poboljšanja najboljeg poznatog rješenja) kako bi se smanjio broj konflikta u tih $k-1$ klasa boja (dodavanje vrha u klasu boje V^* se smatra tabu potezom).

6.4. Implementacija ograničenja WDM mreža s valnim usmjeravanjem

Unutar ovoga rada razmatrana su dva ograničenja na RWA problem koji mogu učiniti sam problem zahtjevnijim za riješiti. To su raspoređeni svjetlosni zahtjevi te smetnje.

Raspoređeni svjetlosni zahtjevi proširuju postojeće svjetlosne zahtjeve s vremenom uspostave i vremenom raskida svjetlosnog puta. Prilikom rješavanja RWA problema s raspoređenim svjetlosnim zahtjevima nakon usmjeravanja stvara se konfliktni graf. Postupak je sljedeći:

1. Konstruira se graf $G(V, E)$ tako da je svaki svjetlosni put u cijeloj fizičkoj topologiji predstavljen vrhom u grafu G. Između dvaju vrhova postoji dvosmjerni brid ako svjetlosni putevi predstavljeni tim čvorovima prolaze istom svjetlovodnom niti **u istom vremenu**.
2. Potrebno je obojiti vrhove grafa G, tako da vrhovi povezani bridom nisu iste boje.

Kod razmatranja smetnji u WDM optičkim mrežama upotrebljava se procjena degradacije optičkog signala na odredišnom čvoru. Izraz koji određuje Q factor na odredišnom čvoru jest:

$$Q_{end} = a_0 + a_1 * OSNR_{end} + a_2 * N_{SPAN} + a_3 * (P_0 * N_{SPAN})^B \qquad (2.4)$$

Faktor kvalitete Q se sastoji od linearnih i nelinearnih smetnji. $OSNR_{end}$ se izražava u decibelima (dB) i označava optički omjer signala i šuma na odredišnom čvoru. Produkti $a_2 * N_{SPAN}$ i $a_3 * (P_0 * N_{SPAN})^B$ u obzir uzimaju nelinearne smetnje, s obzirom na sva pojačala diljem puta (dodatni pojačivač na početku veze, te pojačala veze na čvorovima između izvorišnog i odredišnog) za djelomično transparentene mreže. Parametar N_{SPAN} je broj odjeljaka (engl. *span*) transparentnog puta (odjeljci se odnose na dijelove svjetlovodne niti između dva pojačala), P_0 [dBm] je razina snage pri puštanju signala (najčešće 3 dBm). Koeficijenti a_0, a_1, a_2, a_3 i B ovise o tipu sustava za vezu i najčešće se podešavaju pri testiranju tog sustava kada njemu postoji pristup. Neke standardne vrijednosti su preporučene od strane FP6 Europskog istraživačkog projekta NOBEL I (15) $a_0 \cong 0{,}4$, $a_1 \cong 0{,}96$, $a_2 \cong -0{,}041$, $a_3 \cong 0{,}02$, $B \cong 0{,}2$.

Krajnji, izlazni OSNR svjetlosnog puta (ili njegovog dijela) može se izračunati zbrajanjem vrijednosti OSNR-a nad svim osnovnim komponentama. U osnovne komponente spadaju svi odjeljci i čvorovi diljem svjetlosnog puta. Neka i označava jednu od komponenti, $OSNR_i$ kao vrijednost OSNR-a u toj komponenti koja je dio sveukupnog OSNR-a se računa:

$$OSNR_i[dB] = P_0[dBm] - QN - |T_i|[dB] - |F_i|[dB] \tag{2.5}$$

QN je kvantni šum (tipična vrijednost -58 dB), F_i je vrijednost šuma optičkog pojačala koje pripada komponenti, a T_i je ukupno prigušenje elementa u decibelima $[dB]$. Kada se i odnosi na odjeljak, F_i je vrijednost šuma pretpojačala (pretpostavlja se da ima ista svojstva kao i pojačalo), a kada se i odnosi na čvor F_i je vrijednost šuma dodatnog pojačala na izlazu čvora. Slično je i T_i prigušenje svjetlovodne niti odjeljka, te i prigušenje za komutacijski materijal (engl. *switching fabric*) čvora. Optički omjer signala i šuma $OSNR$ se na kraju djelomično transparentnog puta izražava u linearnim jedinicama pomoću R_{total}, a računa se pomoću izraza:

$$OSNR_{end} = 10 * \log_{10}(R_{total}) \text{ gdje je } R_{total} = \sum_i \frac{1}{R_i} \tag{2.6}$$

Pretpostavka je da su svi odjeljci neke veze iste duljine, tj. da je razmak između pojačala unutar veze (dio svjetlosne niti podijeljene pojačalima uz pretpojačalo) iste duljine. U stvarnosti odjeljci ne moraju biti uniformne duljine. Također pretpostavka je da su svi gubici kompenzirani u svakom elementu pojačalom tj. došlo je do potpune regeneracije signala. Kada se faktor kvalitete Q mjeri na putu u kojem je n čvorova (uključujući izvorišni i odredišni), samo $(n-1)$ čvorova doprinosi smetnjama, jer svjetlosni put ne prelazi dodatno pojačalo u odredišnom čvoru. Za potrebe računanja smetnji unutar fizičke topologije predstavljene grafom korišteni su parametri i njihove vrijednosti preuzete iz (15) i navedene su u tablici 1.

Tablica 1. Vrijednosti parametara u djelomično transparentnoj optičkoj mreži

s_{ampl}: maksimalna duljina odjeljka [km]	85
α: prigušenje kabela [dB/km]	0,23
CM: margina kabela[dB]	3
QN: kvantni šum[dB]	-58
F_L: vrijednost šuma za pojačalo u vezi i predpojačalo	5
F_B: vrijednost šuma za dodatno pojačalo	6
P_0: razina snage pri puštanju signala	3
T_{node}: prigušenje komutacijskog materijala	13,0
a_0	0,4
a_1	0,96
a_2	-0,041
$a_3 = B/10$	0,02

Formule su navedene u tablici 2. i 3.

$OSNR_{node}[dB]$	$P_0 - QN - T_{node} - F_b$
$\frac{1}{R_{node}}$ (u linearnim jedinicama)	$10^{-OSNR_{node}(dB)/10}$

Broj odjeljaka	$N_{span} = \left\lceil \frac{L}{s_{ampl}} \right\rceil$
Duljina odjeljaka	$L_{span} = L/N_{span}$
T_{span}	$L_{span} * \alpha + CM$
T_{span}(u linearnim jedinicama)	$10^{T_{span}(dB)/10}$
$OSNR_{span}$	$P_0 - QN - T_{span} - F_L$
$1/R_{span}$	$10^{-OSNR_{span}/10}$
$1/R_{link}$	$N_{span} \frac{1}{R_{span}}$
R_{link}	$\left(1/R_{link}\right)^{-1}$
$1/R_{total}$	$\sum_{links} \frac{1}{R_{link}} + \frac{(n-1)}{R_{node}}$
$OSNR_{end}$	$10 * \log_{10} R_{total}$
$Q[dB]$	$Q_{end} = a_0 + a_1 * OSNR_{end} + a_2 * N_{SPAN} + a_3 * (P_0 * N_{SPAN})^B$

Valja napomenuti n označava broj čvorova u mreži, dok je sveukupni broj odjeljaka sumacija svih N_{span}.

Nakon izračuna faktora kvalitete nekog svjetlosnog puta te ukoliko je on manji od 17 dB blokira se. Blokirani svjetlosni putevi se odbacuju te se ne razmatraju pri rješavanju potproblema dodjeljivanja valnih duljina.

7. Rezultati eksperimenta

U ovom poglavlju navedeni su rezultati eksperimenta dobiveni rješavanjem RWA problema sa statičkim i raspoređenim svjetlosnim zahtjevima te s obzirom na ili bez fizičkih smetnji. Grafovi nad kojima su se ispitivanja izvršila stvoreni su slučajno. Postupak je sljedeći:
Prvo se stvara graf s n vrhova uz vjerojatnost p_1 postojanja brida između dva vrha. Nad njime se rješava problem usmjeravanja uz određeni k broj svjetlosnih zahtjeva. Svjetlosni zahtjevi se također stvaraju slučajnim procesom, odabirom dva vrha na grafu uz vjerojatnost p_2 stvaranja svjetlosnog zahtjeva. Za vrijeme usmjeravanja provjeravaju se smetnje, te zadovoljava li neki svjetlosni put uvjet kvalitete optičkog signala kojeg će primiti na odredišnom čvoru. S obzirom na vrstu zahtjeva stvara se konfliktni graf nad kojim se rješava problem dodjeljivanja valnih duljina s dva razvijena algoritma: HEA i VNS.
Složenost Bellman-Ford algoritma je $O(|V| * |E|)$, odnosno umnožak broja bridova i vrhova.

Složenost HEA algoritma jest O(L*MaxIter*k*$|V_i| * (|V_i| - 1)$), gdje je L broj generacija metaheuristike bez poboljšanja trenutnog rješenja (nakon što broj generacija poprimi vrijednost L algoritam prekida s radom), MaxIter označava maksimalni broj iteracija Tabucol algoritma koji služi kao lokalna pretraga. Ostali parametri pripadaju funkciji evaluacije gdje je potrebno za svaku klasu boje izračunati broj konfliktnih bridova. Gdje je k broj klasa boja, a $|V_i|$ označava broj vrhova u klasi boje s najviše vrhova.

Složenost VNS algoritma jest O($|V| * |V|$ *k*$|V_{konf}| * (|V_i|)$). Složenost je slična HEA algoritmu osim parametra $|V|$ koji predstavlja broj vrhova u konfliktnom grafu. Složenost $O(|V_{konf}| * (|V_i|))$ predstavlja složenost evaluacijske funkcije koja zahtijeva najviše računskih operacija unutar lokalne pretrage. Gdje je $|V_{konf}|$ broj konfliktnih vrhova grafa G, a $|V_i|$ broj vrhova u klasi boje pojedinog konfliktnog vrha. Parametar k označava faktor umnoška broja vrhova grafa kojim se dobiva broj iteracija *Tabucol* algoritma.
Parametri algoritama se razlikuju. Za HEA algoritam uzima se veličina populacije 10, a za broj generacija nakon kojih se prekida rad algoritma (gore naveden kao L) se uzima vrijednost

200. Parametri lokalne pretrage su $|V| * 3$ kao broj iteracija *Tabucol* algoritma. Trajanje tabu statusa nekog 1-poteza (v, i) određuje se izrazom $A + b * F(x)$ gdje se A bira unutar intervala $[0,9]$, b je vrijednosti 0,6, a $F(x)$ označava broj konfliktnih vrhova.

Parametri VNS algoritma su broj iteracija koji je određen brojem vrhova grafa, te broj iteracija Tabucol algoritma koji je određen kao $|V| * 5$. Trajanje tabu statusa nekog *1-poteza* (v, i) jest 10.

Kako bi se funkcija evaluacije valjanosti pojedinog rješenja u bilo kojem od dva algoritma izvršavala brzo, stvaraju se zasebna asocijativna polja [5]gdje se dohvaćanje neke vrijednosti po ključu izvodi uz složenost O(1). Asocijativna polja sadržavaju trenutne boje svih vrhova, a i stvaraju se zasebna specifična za poneko susjedstvo unutar VNS algoritma. Postoji i gama (γ) matrica koja olakšava izračun 1-poteza. Naime za svaki par $\delta(v, i)$ koji predstavlja vrh i boju u gama matrici je naveden broj konfliktnih vrhova te se time mogu odrediti *kritični 1-potezi*. Tijekom izvođenja algoritama računski najzahtjevnija je lokalna pretraga u obliku Tabucol algoritma kojeg sadrže oba algoritma. Asocijativno polje koje predstavlja trenutne boje svih vrhova koristi se kako bi se ubrzalo izvođenje evaluacije rješenja, vrhovi koji pripadaju istoj klasi boje te imaju istu boju su konfliktni vrhovi i označavaju postojanje barem jednog konfliktnog brida. Drugo asocijativno polje sadrži parove vrhova koji označavaju postojanje brida u samom grafu, tj. ako u tom asocijativnom polju postoji par dvaju vrhova onda između njih postoji i brid. Koristi se i zasebna tabu matrica kojom se označava koliko iteracija je neki potez tabu. Ako je neki *kritični 1-potez* označen kao par (v, i) postavljen kao tabu to znači da je za određen broj iteracija nemoguće izvesti premještaj vrha v u boju i (odnosno u klasu boje V_i). Za neki *1-potez* u tabu matrici postavi se vrijednost koja je jednaka zbroju trenutne iteracije Tabucol algoritma kojoj je pribrojen broj iteracija stanja tabu nekog *1-poteza.*

[5] Asocijativna polja predstavljaju skup parova (ključ,vrijednost) gdje je ključ jedinstvena vrijednost.

Unatoč svim strukturama podataka, te korištenjem funkcija najmanje složenosti za potragu potencijalnih *1-poteza* i evaluacije, Tabucol algoritam i dalje može biti vremenski zahtjevan za velike grafove i veliki broj iteracija.

Slijedi nekoliko inačica grafova koji su korišteni za rješavanje problema usmjeravanja i dodjeljivanja valnih duljina te nakon njih i rezultati za rješavanje RWA problema nad njima.

7.1. Ispitni primjeri grafova

Kao ispitni primjeri generirani su grafovi koji predstavljaju neku fizičku topologiju. Fizičke topologije su predstavljene kao grafovi s manjim brojem vrhova te manjim brojem bridova. Obično su to grafovi s manje od 50 vrha. Iz tih grafova generiranjem svjetlosnih zahtjeva stvaraju se konfliktni grafovi. Na isti način rješava se problem RWA u člancima (19)(20) gdje se za svjetlosne zahtjeve rješava potproblem dodjeljivanja valnih duljina. Broj svjetlosnih zahtjeva obično se odvija u iznosu od par stotina. Teško je uzimati gotove grafove za probleme bojanja grafa jer su potproblemi usmjeravanja i dodjeljivanja valnih duljina povezani te je nemoguće usmjeravanjem dobiti graf istovjetan nekom teško rješivom specifičnom grafu.

Robusnost predstavlja broj pokretanja algoritma za koje je dobiven kromatski broj. Većina grafova imaju visoku uspješnost pošto su grafovi manje složenosti, te se pronalazi legalno $k-bojanje$.

Prvi ispitni graf stvoren je s 234 vrha te 445 bridova. Stvoren je prvotno graf fizičke topologije od 50 vrhova sa 0,2 vjerojatnošću za postojanjem brida između dva vrha, te uz vjerojatnost od 0,19 za stvaranjem svjetlosnog zahtjeva između dva vrha.

Tablica 2. Prvi ispitni graf, idealna mreža (bez smetnji)

Prvi ispitni graf, bez smetnji	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)	Raspoređeni zahtjevi broj bridova
Statički zahtjevi - HEA	5/5	8	0,638	-
Raspoređeni zahtjevi - HEA	5/5	5	0,899	220
Statički zahtjevi - VNS	5/5	8	0,288	-
Raspoređeni zahtjevi - VNS	5/5	5	0,232	220

U tablici 2. vidljivi su rezultati izvršavanja algoritama na prvom ispitnom grafu u slučaju kada nema smetnji. Nakon prelaska sa statičkih na raspoređene zahtjeve broj bridova se smanjuje jer neki bridovi mogu koristiti istu valnu duljinu zbog različitog vremena zauzeća svjetlosnog puta. Kromatski broj ostaje isti za oba algoritma uz kraće vrijeme izvođenja VNS-a.
Za prvi ispitni graf sa smetnjama broj blokiranih svjetlosnih puteva je bio 179 te nije bilo potrebe raditi dodjeljivanje valnih duljina jer je uz 55 vrhova broj bridova spao na vrijednost 0, što znači da zbog velikog blokiranja su preostali svjetlosni zahtjevi koji su neovisni jedan o drugome i rješenje potproblema dodjeljivanja valnih duljina je upotreba jedne valne duljine.

Drugi ispitni graf stvoren je s 206 vrha te 556 bridova. Prvotno je stvoren graf fizičke topologije od 50 vrhova sa 0,17 vjerojatnošću za postojanjem brida između dva vrha, uz vjerojatnost od 0,17 za stvaranjem svjetlosnog zahtjeva između dva vrha. Broj bridova kod raspoređenih zahtjeva je 270.

Tablica 3. Drugi ispitni graf, idealna mreža (bez smetnji)

Drugi ispitni graf, bez smetnji	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)
Statički zahtjevi - HEA	5/5	9	0,533
Raspoređeni zahtjevi - HEA	5/5	6	0,752
Statički zahtjevi - VNS	5/5	9	0,235
Raspoređeni zahtjevi - VNS	5/5	6	0,137

U ispitivanju bez smetnji vidljiva je razlika u vremenu izvođenja između VNS i HEA algoritma. Vidljivo je i povećanje vremena izvođenja HEA algoritma kada su u pitanju raspoređeni zahtjevi unatoč tome što se smanjuje broj bridova.

Tablica 4. Drugi ispitni graf, realistična mreža

Drugi ispitni graf sa smetnjama	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)	Broj bridova sa smetnjama
Statički zahtjevi - HEA	5/5	8	0,345	491
Raspoređeni zahtjevi - HEA	5/5	5	1,324	238
Statički zahtjevi - VNS	5/5	8	0,178	491
Raspoređeni zahtjevi - VNS	5/5	5	0,245	238

U ispitivanju sa smetnjama došlo je do blokiranja određenih svjetlosnih puteva zbog manjka u kvaliteti signala (Q faktor). To dovodi do smanjenja broja vrhova i bridova u konfliktnom grafu nad kojim se vrši rješavanje problema dodjeljivanja valnih duljina. Kod raspoređenih zahtjeva dolazi do dodatnog smanjenja broja bridova ali broj vrhova ostaje nepromijenjen. Broj vrhova sa smetnjama je 198 dok je broj blokiranih svjetlosnih puteva 8.

Oba algoritma daju iste kromatske brojeve kao svoje rješenje, ali vrijeme izvođenja je i dalje veće u slučaju raspoređenih zahtjeva.

Treći ispitni graf stvoren je s 514 vrhova te 2090 bridova. Prvotno je stvoren graf fizičke topologije od 50 vrhova sa 0,5 vjerojatnošću za postojanjem brida i vjerojatnošću od 0,4 za stvaranjem svjetlosnog zahtjeva između dva vrha. Broj bridova kod raspoređenih zahtjeva je 1093.

Tablica 5. Treći ispitni graf u idealnoj mreži

Treći ispitni graf, bez smetnji	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)
Statički zahtjevi - HEA	5/5	13	1,817
Raspoređeni zahtjevi - HEA	5/5	9	10,0668
Statički zahtjevi - VNS	5/5	13	4,315
Raspoređeni zahtjevi - VNS	5/5	9	2,198

U trećem ispitnom grafu dolazi do veće razlike pri vremenu izvođenja. Kod statičkih zahtjeva veću brzinu izvođenja ima HEA algoritam naspram VNA algoritma. Kromatski broj za oba algoritma ostaje isti. Kod raspoređenih zahtjeva, vrijeme izvođenja pada za VNA algoritam dok za HEA algoritam raste. Kromatski broj je također isti za oba algoritma.

Četvrti ispitni graf stvoren je sa 371 vrhom te 2207 bridova. Prvotno je stvoren graf fizičke topologije od 40 vrhova sa 0,25 vjerojatnošću za postojanjem brida i vjerojatnošću od 0,45 za stvaranjem svjetlosnog zahtjeva između dva vrha. Broj bridova kod raspoređenih zahtjeva je 1187.

Tablica 6. Četvti ispitni graf bez smetnji

Četvrti ispitni graf, bez smetnji	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)
Statički zahtjevi - HEA	5/5	17	2,0485
Raspoređeni zahtjevi - HEA	5/5	9	5,9508
Statički zahtjevi - VNS	5/5	17	1,216
Raspoređeni zahtjevi - VNS	5/5	9	3,402

U četvrtom ispitnom grafu bez smetnji oba dva algoritma uspješno rješavaju graf sa istim kromatskim brojem. Brže vrijeme izvođenja ima HEA algoritam ali i dalje dolazi do porasta vremena izvođenja pri rješavanju raspoređenih zahtjeva.

Tablica 7. Četvrti ispitni graf sa smetnjama

Četvrti ispitni graf sa smetnjama	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)	Broj bridova sa smetnjama
Statički zahtjevi - HEA	5/5	17	2,275	2062
Raspoređeni zahtjevi - HEA	5/5	9	4,653	1125
Statički zahtjevi - VNS	5/5	17	1,169	2062
Raspoređeni zahtjevi - VNS	5/5	9	5,259	1125

Četvrti ispitni graf sa smetnjama označava brže izvođenje osim za raspoređene zahtjeve kod algoritama. Broj vrhova sa smetnjama je 360. Dok je broj blokiranih svjetlosnih puteva 11. Uz jedanaest blokiranih svjetlosnih puteva smanjuje se broj vrhova grafa nad kojim se vrši dodjeljivanje valnih duljina. Ujedno se smanjuje i broj bridova, ali vrijeme izvođenja se povećava.

Peti ispitni graf stvoren je sa 373 vrha te 3555 bridova. Prvotno je stvoren graf fizičke topologije od 35 vrhova sa 0,20 vjerojatnošću za postojanjem brida i vjerojatnošću od 0,60 za stvaranjem svjetlosnog zahtjeva između dva vrha. Broj bridova za raspoređene zahtjeve kod grafa bez smetnji jest 1851.

Tablica 8. Peti ispitni graf bez smetnji

Peti ispitni graf, bez smetnji	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)
Statički zahtjevi - HEA	5/5	24	2,139
Raspoređeni zahtjevi - HEA	5/5	14	2,875
Statički zahtjevi - VNS	5/5	24	1,221
Raspoređeni zahtjevi - VNS	5/5	14	1,158

Kao i za prošle grafove kromatski brojevi za oba dva algoritma su isti uz nisko vrijeme izvođenja. Manji porast je zabilježen kod raspoređenih zahtjeva.

Tablica 9. Peti ispitni graf sa smetnjama

Peti ispitni graf sa smetnjama	Robusnost	Najmanji pronađeni broj boja	Vrijeme izvođenja (minute)	Broj bridova sa smetnjama
Statički zahtjevi - HEA	5/5	15	1,023	1342
Raspoređeni zahtjevi - HEA	5/5	9	1,364	727
Statički zahtjevi - VNS	5/5	15	0,388	1342
Raspoređeni zahtjevi - VNS	5/5	9	0,490	727

Peti ispitni graf predstavlja primjer topologije gdje smetnje uzrokuju veliki broj blokiranih svjetlosnih puteva zbog nezadovoljavanja granice kvalitete Q faktora. To dovodi do smanjenja broja vrhova i bridova u grafu nužnih za rješavanje bojanja grafa. Broj vrhova sa smetnjama je 257. Dok je broj blokiranih svjetlosnih puteva 116. Vrijeme izvođenja je nisko a oba algoritma daju istu vrijednost kromatskog broja. I dalje dolazi do porasta vremena izvođenja kod raspoređenih zahtjeva.

Šesti ispitni graf je poslužio za usporedbu HEA, VNS i tabucol algoritma. Sadrži 473 vrhova i 11,625 bridova.

Tablica 10. Šesti ispitni graf (usporedba algoritama)

Šesti ispitni graf	Tabucol	HEA	VNS
Robusnost	5/5	5/5	5/5
Legalno k-bojanje (broj boja)	52	51	51
Prosječno vrijeme izvođenja	0,24	11,851	8,6506

Izvođenjem algoritama nad tim ispitnim grafom vidljivo je da HEA i VNS daju bolje legalno k-bojanje nego Tabucol algoritam. Iako je vrijeme izvođenja Tabucola brže, za potrebe dobivanja što manjeg legalnog k-bojanja potrebno je razmotriti HEA i VNS algoritam koji za trenutne fizičke topologije i trenutni broj zahtjeva mogu pronaći bolje legalno k-bojanje.

Sedmi ispitni graf je također poslužio za usporedbu HEA, VNS i Tabucol algoritama.
Sadrži 249 vrhova i 5398 bridova.

Tablica 11. Sedmi ispitni graf (usporedba algoritama)

Sedmi ispitni graf	Tabucol	HEA	VNS
Robusnost	5/5	3/5	5/5
Legalno k-bojanje (broj boja)	42	40	41
Prosječno vrijeme izvođenja	0,058	3,267	8,6506

Izvođenjem algoritama sedmog ispitnog grafa vidljivo je da HEA algoritam daje najbolje rezultate ali uz određeno rasipanje. U nekim slučajevima dano je legalno k-bojanje od 41 boje. Algoritam VNS je u svim slučajevima dao 41 boju kao rezultat. Algoritmi HEA i VNS daju bolje rezultate od Tabucol algoritma uz dulje vrijeme izvođenja.

7.2. Postavke eksperimenta

Za programsko ostvarenje algoritma korišten je razvojni alat Microsoft Visual Studio 2010 te Microsoft .NET Framework 4.0. Programski jezik Microsoft Visual C# korišten je za ostvariavanje aplikacije za rješvanje problema nazvane RWA_Solver. Ispitivanje za različite primjere grafova izvršavalo se na prijenosnom računalu Acer Extensa 5635G s 2.2 GHz Intel Core 2 Duo T6600 procesorom te 3.0 GB RAM. Pri tome je korišten Microsoft Windows 7 SP1 operativni sustav. Ispitivanje se izvršavalo u pozadini, u dretvi s normalnim prioritetom izvršavanja, na jednoj jezgri.

8. Zaključak

U ovom radu predstavljen je problem usmjeravanja i dodjeljivanja valnih duljina (RWA) u WDM optičkim mrežama. Također su predstavljeni algoritmi koji taj problem rješavaju. Uz osnovna ograničenja razmatrala su se ograničenja s obzirom na smetnje te s obzirom na vrstu zahtjeva. U ovom radu razmatrani su statički i raspoređeni zahtjevi. Svi algoritmi su obećavajući za rješavanje problema, pogotovo kad su u pitanju statički zahtjevi. Rješavanjem RWA problema za statičke zahtjeve možemo razmotriti i rješavanje raspoređenih zahtjeva, naime kod raspoređenih zahtjeva do preklapanja svjetlosnih puteva dolazi samo ako oni prolaze istim putem u istom vremenu, stoga raspoređeni zahtjevi predstavljaju lakšu inačicu grafa nad kojim treba riješiti problem dodjeljivanja valnih duljina. Smetnje je moguće izmjeriti samo u stvarnom vremenu na stvarnom sustavu, ali u slučaju da RWA problem želimo riješiti uz postojanje smetnji, to je također moguće pretpostavljanjem njihovih vrijednosti s obzirom na prethodno obavljena mjerenja. Algoritmi korišteni u radu mogu u prihvatljivom vremenu polučiti dobra rješenja za RWA problem, a HEA i VNS algoritmi mogu riješiti potproblem dodjeljivanja valnih duljina bez obzira na smetnje te korištenje raspoređenih ili statičkih zahtjeva. Kada se primjene na veće grafove, koji su više nego dovoljni da prikažu fizičke topologije na nekom većem regionalnom području, polučuju dobre rezultate.

Ideje za daljnje istraživanje i razvoj na ovom području leže u istraživanju RWA problema s obzirom na druga ograničenja (npr. dinamičke svjetlosne zahtjeve, izgradnja virtualne topologije, svjetlosna stabla, itd.). Što se tiče potproblema dodjeljivanja valnih duljina, moguće je iskoristiti i druge metaheuristike za njegovo rješavanje ili u HEA algoritmu umjesto lokalne pretrage postaviti VNS algoritam.

Literatura

1. **I. Lovrek, M. Matijašević, G. Ježić, D. Jevtić.** *Komunikacijske mreže (radna inačica udžbenika v0.1, dio 1).* Zagreb : Fakultet elektrotehnike i računarstva, 2010.

2. *Routing and Wavelength Assignment of Scheduled Lightpath Demands.* **Josué Kuri, Nicolas Puech, Maurice Gagnaire, Emmanuel Dotaro and Richard Douville.** 8, s.l. : IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, OCTOBER 2003, Vol. 21.

3. **Naohiro Sakiyama, Akira Nagata, Takahiro Matsuda, and Miki Yamamoto.** Dynamic Routing and Wavelength Assignment with Power Consideration. *Citeseerx.* [Online] 2003. [Cited: June 5, 2012.] http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.12.5269&rep=rep1&type=pdf.

4. **Mukherjee, Biswanath.** *Optical WDM Networks.* Davis, CA 95616 U.S.A. : Springer Science+Business Media, Inc., 2006. 0-387-29055-9.

5. **Skorin-Kapov, Nina.** *Optička mreža s valnim multipleksiranjem (WDM); Fotoničke telekomunikacijske tehnologije .* Zagreb : an., 2008/2009.

6. *DIPLOMSKI RAD br. 1754, Rješavanje problema bojanja grafa primjenom hibridnog evolucijskog algoritma.* **Kindl, Hrvoje.** Zagreb : s.n., rujan 2008.

7. *On The Offline Physical Layer Impairment Aware RWA Algorithms in Transparent Optical Networks: State-of-the-Art and Beyond.* **Siamak Azodolmolky, Yvan Pointurier, Mirosław Klinkowski, Eva Marin, Davide Careglio, Josep Sol´e-Pareta, Marianna Angelou and Ioannis Tomkos.** s.l. : ONDM'09 Proceedings of the 13th international conference on Optical Network Design and Modeling, 2009. 978-1-4244-4187-7.

8. *Performance of translucent optical networks under dynamic traffic and uncertain physical-layer information.* **M. Yannuzzi, M. Quagliotti, G. Maier, E. Marín-Tordera, X. Masip-Bruin, S. Sánchez-López, J. Solé-Pareta, W. Erangoli, G. Tamiri.** s.l. : IEEE Press Piscataway, NJ, USA, 2009. 978-1-4244-4187-7.

9. Bellman–Ford algorithm. *Wikipedia, the free encyclopedia.* [Online] [Cited: June 12, 2012.] http://en.wikipedia.org/wiki/Bellman–Ford_algorithm.

10. *Hybrid Evolutionary Algorithms for Graph Coloring.* **Philippe Galinier, Jin-kao Hao.** 4, s.l. : Journal of Combinatorial Optimization, Springer Netherlands, 2002, Vol. 3. 1573-2886.

11. **Vazdar, Roman.** *ZAVRŠNI RAD br. 1361, Višedretveni modeli evolucijskih algoritama.* 2010.

12. **Golub, Marin.** *Genetski algoritam, Prvi dio.* 2004.

13. *A variable neighborhood search for graph coloring.* **Cedric Avanthay, Alain Hertz, Nicolas Zufferey.** s.l. : European Journal of Operational Research 151, 2003.

14. *Variable Neighborhood Search: Principles and applications.* **Pierre Hansen, Nenad Mladenović.** s.l. : European Journal of Operational Research 130, 2001.

15. Next Generation Optical Networks for Broadband European Leadership. [Online] www.ist-nobel.org/Nobel/servlet/Nobel.Main..

16. **Skorin-Kapov, Nina.** *Optička mreža s valnim multipleksiranjem (WDM); Fotoničke telekomunikacijsske tehnologije 2008/2009.* Zagreb : an., 2008/2009.

17. *A survey of local search methods for graph coloring.* **Galinier, Philippe and Hertz, Alain.** 9, s.l. : Computers and Operations Research - Anniversary focused issue of computers & operations research on tabu search, 2006, Vol. 33. 0305-0548.

18. *Using tabu search techniques for graph coloring.* **Hertz, A. and Werra, D.** 4, s.l. : Computing, 1987, Vol. 39. 1436-5057 (Online).19. *Solving the RWA Problem in WDM Optical Networks Using the BCO Meta-Heuristic.* **Marković, Goran Z. and Aćimović-Raspopović, Vladanka S.** 1, s.l. : Telfor Journal, 2010, Vol. 2.

20. *A Genetic Algorithm Approach for Solving the Routing and Wavelength Assignment Problem in WDM Networks.* **Banerjee, Nilanjan, Mehta, Vaibhav and Pandey, Sugam.**