

SOFTVER ZA ANALIZU EFEKATA DINAMIČKOG RUTIRANJA U MREŽAMA SA INTERKONEKCIJOM

Snežana Mladenović, Aleksandra Kostić-Ljubisavljević
Univerzitet u Beogradu - Saobraćajni fakultet

Sadržaj: *U tržišnim uslovima neohodno je upravljati telekomunikacionom mrežom na način da se isbalansiraju zahtevi korisnika za što manjom tarifom i zahtevi operatora za što većim prihodom. Da bi se donele ispravne odluke kako u projektovanju tako i u eksploataciji telekomunikacione mreže, neophodno je izvršiti poređenje različitih metoda dinamičkog rutiranja pri različitim načinima naplate interkonekcije. Za potrebe ovakvog istraživanja razvijen je originalni softver čiji je osnovni zadatak simulacija saobraćaja u telekomunikacionoj mreži uz mogućnost izbora metode dinamičkog rutiranja i načina naplate interkonekcije.*

Ključne reči: *softver, interkonekcija mreža, dinamičko rutiranje*

1. Uvod

Veći broj operatora na nekom telekomunikacionom tržištu podrazumeva postojanje njihove međusobne interkonekcije. Način određivanja interkonekcijskih zaduženja mora biti jasan i transparentan da bi se zadovoljili interesi i operatora i regulatora. S druge strane, dinamičko rutiranje saobraćaja daje mogućnost efikasnijeg i racionalnijeg korišćenja postojećih resursa telekomunikacione mreže. Drugim rečima, način naplate interkonekcije i primenjena metoda dinamičkog rutiranja mogu značajno uticati na prihod operatora.

Cilj našeg istraživanja je analiza efekata različitih metoda dinamičkog rutiranja pri različitim načinima naplate interkonekcije. Za potrebe ovakvog istraživanja razvijen je originalni softver čiji je osnovni zadatak simulacija saobraćaja u telekomunikacionoj mreži korišćenjem različitih metoda dinamičkog rutiranja saobraćaja pri različitim načinima naplate interkonekcije. Dodatni zadatak softvera je statistička obrada relevantnih parametara rezultata simulacije.

Rad je koncipiran na sledeći način: Drugi deo se odnosi na prikaz nekih od metoda koje se koriste za tarifiranje interkonekcije. U sekciji 3. dat je kratak pregled karakteristika dinamičkog rutiranja. U delu 4. predstavljen je dizajn realizovanog softvera a peti prikazuje neke aspekte njegove implementacije. Softver je testiran na standardizovan način i neki rezultati tog testiranja su predstavljeni u sekciji 6. Sedma

sekcija daje primer primene realizovanog softvera na jednu telekomunikacionu mrežu sa tri operatora.

2. Pojam interkonekcije

Evropska komisija definiše interkonekciju kao "fizičku i logičku vezu telekomunikacionih mreža korišćenu od strane istog ili različitih operatora, na zahtev pretplatnika jednog operatora da komuniciraju sa pretplatnicima istog ili drugog operatora. Interkonekcija, isto tako, podrazumeva da se pretplatnicima daje mogućnost da koriste usluge koje pružaju drugi operatori. Usluge mogu da pružaju delovi mreža koje su obuhvaćene interkonekcijom ili drugi provajderi koji imaju pristup mreži." [1].

Postoje brojni koncepti koji se primenjuju u telekomunikacionim mrežama širom sveta u cilju određivanja adekvatne i prihvatljive cene interkonekcije sa aspekta operatora i regulatora [2]. Najčešće su u upotrebi koncepti u kojima su cene bazirane na troškovima, na maloprodaji, "*price cap*" koncept, koncept podele profita ili "*bill and keep*" koncept [3]. U dosadašnjem istraživanju smo se bavili sa dva najčešće korišćena koncepta za naplatu interkonekcije:

- "*bill and keep*" koncept (Metoda *i-0*) i
- "*cost based*" koncept (Metoda *i-1*).

Stoga će u daljem tekstu biti objašnjena ova dva načina naplate interkonekcije.

"*Bill and keep*" pristup podrazumeva da se operatorima interkonekcija uopšte ne naplaćuje. Svaki operator "naplati" svojim krajnjim korisnicima izlazni saobraćaj koji oni generišu ka drugoj mreži, i "zadržava" sav prihod koji tako proistekne. Model "naplati-i-zadrži" pretpostavlja sledeće: da postoje nadoknade za interkonekciju one bi se ionako međusobno poništile, što ne bi rezultiralo ni gubitkom ni dobitkom ni za jednog operatora. Dalje, time što propuštaju te nadoknade, operatori izbegavaju administrativni teret naplate jedne drugoj za razmenjen saobraćaj.

Ovaj koncept se primenjuje u situacijama kada je saobraćaj između operatora u ravnoteži. Da bi se ustanovilo da ta ravnoteža postoji potrebno je kontinualno merenje saobraćaja i troškova koji iz njega proističu.

"*Cost based*" koncept je zasnovan na troškovima i podrazumeva da se korišćenjem studija troškova i principa ekonomske efikasnosti, cene interkonekcije formiraju tako da se postigne povraćaj troškova operatora. Fiksni troškovi se mogu nadoknaditi primenom proporcionalnih fiksnih ili *flat* tarifa. Na primer, jednokratni trošak uspostavljanja kanala veze se može nadoknaditi nepovratnom tarifom u adekvatnom iznosu. Varijabilni troškovi bi se trebali nadoknaditi primenom varijabilnih tarifa. To jest, troškove zavisne od saobraćaja bi trebalo nadoknaditi naplatom interkonekcije po minutu.

Ovo su naizgled jednostavni koncepti, ali oni su se na mnogim tržištima praktikovali samo povremeno. Regulatori radije izaberu da uključe veliku količinu troškova u minutne tarife, nego da razdvoje troškove za različite komponente mreže i servise. Dominantni operatori se radije odlučuju za tarife koje su bazirane na upotrebi, jer takve tarife osiguravaju rastući profit od interkonekcije kad god se konkurent proširi i poveća saobraćaj u okviru svoje mreže. Međutim, oslanjanje isključivo na tarife bazirane na upotrebi i nije baš ekonomski najprikladnije.

U svakom slučaju, veoma je teško ustanoviti sve sadržane troškove. Još je teže kada kada se cena mora ustanoviti za neobjedinjeni pristup mreži. Ipak, ovaj koncept je

jedan od najprihvaćenijih načina utvrđivanja cene interkonekcije širom sveta i stoga je našao mesto i u našem istraživanju.

3. Dinamičko rutiranje

Za razliku od metoda sa fiksnim pravilima rutiranja, u kojima se kapaciteti u mreži dodeljuju saglasno prognoziranim saobraćajnim zahtevima između parova komutacionih čvorova, dinamičko rutiranje se zasniva na drugačijem pristupu: saobraćaj se dodeljuje onim putanjama na kojima trenutno postoje slobodni kapaciteti. Primenom metoda dinamičkog rutiranja saobraćaja mogu se ostvariti značajne uštede u troškovima mreže jer je akcenat na što je moguće boljem iskorišćenju raspoloživih a ne na angažovanju dodatnih resursa. Pored uštede, postoje i druge prednosti dinamičkog nad fiksnim rutiranjem koje su razmatrane u [4] i [5]. U spomenutoj literaturi su klasifikovane i opisane brojne metode dinamičkog rutiranja, a naše dosadašnje istraživanje se bavilo sa sledeće četiri:

- rutiranje po najkraćoj putanji (Metoda 1-*j*),
- rutiranje po putanjama koje u sebi sadrže maksimalno 4 čvora – ruta sa najviše 3 linka (Metoda 2-*j*),
- rutiranje po putanjama koje su na slučajan način izabrane iz seta mogućih putanja (Metoda 3-*j*), i
- rutiranje po poslednjoj uspešnoj putanji (Metoda 4-*j*).

S obzirom da su nazivi metoda rutiranja sami po sebi jasni, dalja objašnjenja će na ovom mestu biti izostavljena.

4. Dizajn softvera

Kako je već istaknuto, osnovni zadatak softvera je poređenje različitih metoda naplate interkonekcije pri različitim metodama rutiranja saobraćaja u mreži. Ovo poređenje treba obaviti uz variranje brojnih ulaznih veličina: topologije mreže, kapaciteta čvorova, troškova čvorova, kapaciteta linkova, troškova linkova, broja konekcija, ... Takođe, zaključci se mogu smatrati validnim samo ako su izvedeni na osnovu "dovoljnog" velikog broja eksperimenata. Realizovani softver simulira opsluživanje konekcija u telekomunikacionoj mreži i vrši statističku obradu relevantnih parametara.

U strukturi softvera, koja je šematski prikazana na Slici 1, izdvajaju se sledeće logičke celine:

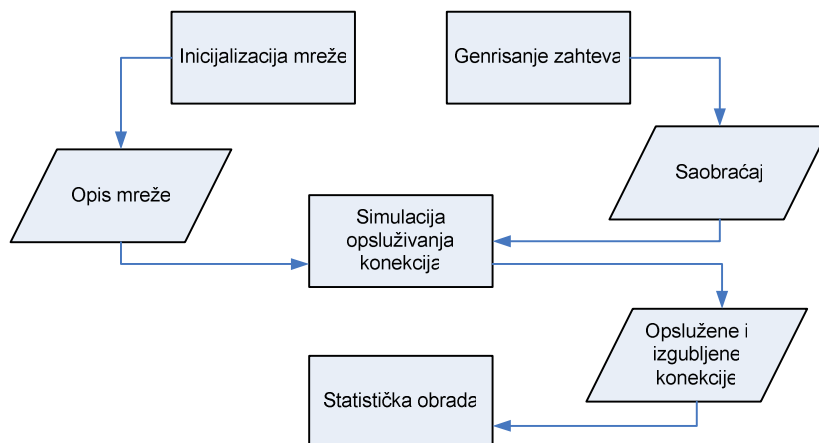
INICIJALIZACIJA MREŽE. Zadatak podsistema za inicijalizaciju mreže je da konstitutivne elemente grafa konkretne mreže reprezentuje strukturama podataka podesnim za proces simulacije. Konstitutivni elementi mreže su čvorovi i linkovi. Atributi čvora su kapacitet i trošak čvora a atributi linka su početni i krajnji čvor, kapacitet linka i trošak linka.

Pod pojmom **kapacitet čvora** ovde se smatra srednji broj zahteva za konekcijom koji se može generisati u svakom čvoru. Zbog jednostavnijeg rada, a i kasnije lakše obrade dobijenih rezultata, kapaciteti čvorova su normalizovani. Pretpostavljeno je da svaki čvor može da prosledi sav tranzitni saobraćaj koji kroz njega prolazi, pod uslovom da su linkovi koji ulaze i izlaze iz njega slobodni.

Trošak čvora je parametar koji predstavlja trošak prosleđivanja konekcije kroz posmatrani čvor. U zavisnosti od toga koja je funkcija datog čvora u procesu opsluživanja saobraćaja, trošak može da se odnosi na generisanje, tranzitiranje i terminiranje konekcije. Za sve čvorove smo unapred definisale da su troškovi generisanja i terminiranja konekcije jednaki i duplo veći od troškova tranzitiranja.

Kapacitet linka je broj kanala za opsluživanje i predstavlja ulaznu veličinu. Jednostavnosti radi, pretpostavile smo da svi linkovi u mreži imaju isti kapacitet za jedno izvršenje softvera.

Trošak linka se definiše kao onaj trošak koji snosi operator da bi opslužio zahtev za konekcijom koja je generisana u njegovoj mreži, ili tranzitira kroz njegovu mrežu, ili terminira u njegovoj mreži. Iako ovi troškovi u praksi nisu na isti način definisani, niti se na isti način računaju, u ovom radu će oni biti posmatrani pod zajedničkim nazivom trošak linka. Mi smo uvele „faktor troška interkonekcije“ kojim se množi trošak linka koji se nalazi u izabranoj putanji kojom se opslužuje konekcija, pod uslovom da se taj link nalazi u drugoj mreži.



Slika. 1. Struktura realizovanog softvera

GENERISANJE ZAHTEVA. U svakom čvoru se generiše saobraćaj koji je pretstavljen Poasonovim potokom događaja, sa srednjom brojem zahteva koja predstavlja umnožak kapaciteta čvora. Umnožak i kapacitet čvora su ulazne veličine. Umnožak se unosi sa ekranskog obrasca (veličina *saob*) a kapaciteti čvorova iz ulazne datoteke. Atributi konekcije su: izvorište, odredište, trenutak generisanja i trajanje. Trajanje je slučajna promenljiva koja ima eksponencijalnu raspodelu sa srednjim vremenom od 3 minuta. Kako odabrani implementacioni jezik Visual Basic nema ugrađenu mogućnost generisanja slučajnih brojeva po Poissonovoj i eksponencijalnoj raspodeli, korišćeni su iz literature poznati algoritmi [6] za prelazak sa uniformne na Poasonovu, tj. eksponencijalnu raspodelu.

SIMULACIJA OPSLUŽIVANJA KONEKCIJA. Za zadatu mrežu i generisane zahteve, korisnik bira jednu od ponuđenih metoda rutiranja, jedan od ponuđenih načina naplate interkonekcije i broj kanala po linku. Osim ovih podataka korisnik unosi i *tarifu* (cenu) koja se naplaćuje od krajnjeg korisnika paketa servisa.

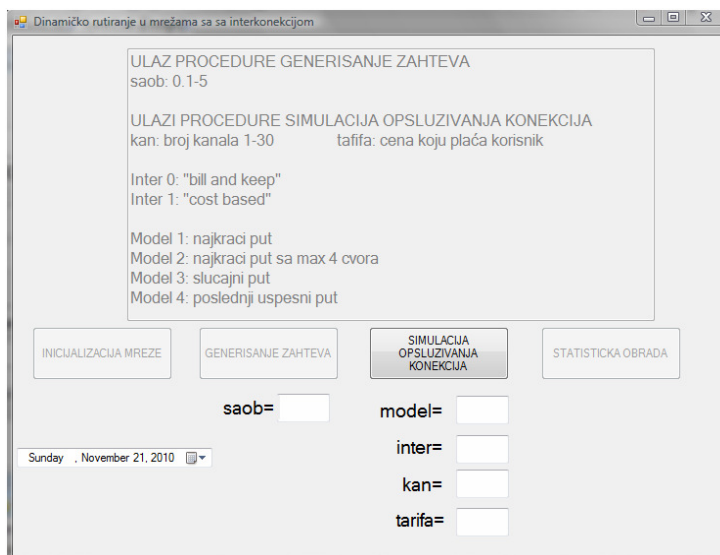
Nakon toga, aktivira se proces simulacije opsluživanja konekcija. Konekcija može biti opslužena ili izgubljena. "Trag" o svakoj opsluženoj ili izgubljenoj konekciji beleži se u jednu od izlaznih datoteka.

STATISTIČKA OBRADA. Izlazi iz prethodne faze daju mogućnost formiranja brojnih statistika vezanih za zahteve, čvorove, linkove, putanje, troškove, ...

5. Implementacija softvera

Prvi korak u implementaciji – kodiranju je izbor odgovarajućeg programskog jezika ili softverskog alata. S obzirom da je ključni zadatak softvera simulacija opsluživanja zahteva za konekcijama, prve ideje su bile da se odabere neki od komercijalno raspoloživih simulacionih jezika ili softverskih alata. Međutim, autorima poznati simulacioni jezici i softverski alati nisu davali mogućnost da korisnik posredstvom jednostavnog korisničkog interfejsa prati saobraćajnu situaciju u kojoj je svaka konkretna konekcija opslužena ili izgubljena. Osnovna ideja je bila da se pamćenjem "traga" o svakoj konekciji korisnik lako može uveriti da je opslužena konekcija prosleđena optimalno prema zadatom kriterijumu u tekućoj saobraćajnoj situaciji, odnosno da za izgubljenu konekciju nije bilo alternative.

Stoga je odabrano da se programski sistem razvija "od nule", u programskom jeziku opšte namene. Sve opisane logičke celine su implementirane kao programske celine u Microsoft Visual Basic okruženju i objedinjene u Windows aplikaciju nazvanu *Dinamičko rutiranje u mrežama sa interkonekcijom* [7]. Glavni prozor aplikacije predstavljen je na Slici 2.



Slika 2. Glavni prozor aplikacije

Jedan od izlaza procesa simulacije je datoteka predstavljena na Slici 3. koja obezbeđuje memorisanje "traga" o svakoj konekciji. Kolone ove datoteke predstavljaju redom:

- 1) redni broj (šifru) izvorišnog čvora,
- 2) redni broj (šifru) odredišnog čvora,
- 3) trenutak generisanja konekcije,
- 4) trajanje konekcije u sekundama,
- 5) putanja po kojoj je konekcija realizovana ili naznaka (----) da nije realizovana,
- 6) cena opslužene konekcije,
- 7) trenutak okončanja konekcije,
- 8) rang putanje po kojoj je konekcija realizovana.

Source Node	Destination Node	Time	Duration	Path	Cost	Rank
10	9	778	28	-----		
6	6	778	181	nema putanje	0	959
4	4	778	411	nema putanje	0	1189
2	1	778	78	2 -> 1	1	856
8	4	778	691	-----		
8	10	778	133	-----		
3	5	778	407	-----		
8	8	778	81	nema putanje	0	859
6	6	778	256	nema putanje	0	1034
4	9	779	63	-----		
1	4	779	170	1 -> 2 -> 4	2	949
7	7	779	156	nema putanje	0	935
7	8	779	87	7 -> 8	7	866
9	7	779	419	9 -> 8 -> 7	10	1198
8	7	779	92	-----		
5	3	779	4	-----		
4	10	779	444	-----		
7	7	779	109	nema putanje	0	888
8	10	779	159	-----		
4	4	780	147	nema putanje	0	927
4	10	780	451	-----		
7	4	780	307	-----		
4	5	780	264	4 -> 5	3	1044
6	2	780	302	-----		
9	9	780	221	nema putanje	0	1001
10	5	780	30	-----		
5	4	780	552	5 -> 3 -> 2 -> 1 -> 4	6	1332
10	6	780	5	-----		
6	7	780	460	-----		
10	7	780	119	-----		
10	10	781	56	nema putanje	0	837
7	10	781	243	7 -> 10	8	1024
9	10	781	75	9 -> 10	4	856
4	3	781	39	-----		

Sika 3. Fragmet datoteke *tragovi konekcija*

Modul za statističku obradu formira statistike koje se odnose na:

- procentualnu zauzetost svakog od linkova,
- troškove konekcija (ukupne i raspodeljene po mrežama),
- broj konekcija koji je generisan, tranzitirao i terminirao u svakom od čvorova,
- prihode nastale opsluživanjem konekcija (ukupne i raspodeljene po mrežama),
- ukupan broj konekcija koji je generisan, tranzitirao ili terminirao u svakoj od mreža,
- za svaku putanju koja je korišćena memorišu se podaci o linkovima koji je sačinjavaju, broju konekcija koje su tu putanju koristile, kao i o vremenu zauzeća,

- broj realizovanih konekcija između parova čvorova,
- kumulativne troškove realizovanih konekcija između parova čvorova, kao i prosečne troškove realizovanih konekcija između parova čvorova,
- ...

Pošto je softver modularno koncipiran moguće ga je proširiti dodatnim funkcijama (npr. novim metodama rutiranja, novim načinima naplate interkonekcije, novim raspodelima koje se tiču generisanja zahteva, novim tipovima statistika) ili bi se neke od postojećih funkcija mogle usavršiti. Stoga je opredeljenje da se ovaj softverski sistem razvija inkrementalno, razvojem serije prototipova, pri čemu će svaki sledeći prototip uzimati u obzir nove ili dopunjene zahteve korisnika (tzv. spiralni model razvoja softvera). Možemo smatrati da je u ovoj fazi već razvijen drugi prototip i eksperimenti koji će biti opisani sprovedeni su na njemu. On predstavlja nadogradnju prvog prototipa, ranije opisanog u [8].

6. Vrednovanje softvera

Svaki softverski sistem mora biti testiran je na standardizovan način, kroz verifikaciju i validaciju. Verifikacijom se proverava da li smo sistem korektno razvili (interni kvaliteti softvera) a validacija proverava da li je razvijeni sistem zadovoljava potrebe korisnika (eksterni kvaliteti softvera). Testiranje ovog softvera je obavljeno prema standardnim kriterijumima kvaliteta [9, 10], a posebna pažnja je, uzimajući u obzir prirodu problema, posvećena korektnosti, efikasnosti i prilagodljivosti.

Korektnost je osobina softvera da tačno izvršava funkciju radi koje je konstruisan. Projektovani softver mora apsolutno verno oponašati saobraćaj u telekomunikacionoj mreži. Uvidom u obimne izlazne datoteke koje čuvaju trag o svakoj opsluženoj ili izgubljenoj konekciji, može se sa velikom verovatnoćom tvrditi da je softver korektan, tj. da implementira sva važeća pravila opsluživanja zahteva. Za nekorektan softver nema smisla razmatrati ostale kriterijume kvaliteta.

Simulacioni softver, po pravilu mora biti **efikasan** (sa što je moguće kraćim vremenom izvršavanja) kako bi korisnik bio u prilici da zaključke izvodi na osnovu velikog broja sprovedenih eksperimenata. U našem slučaju, za mrežu prikazanu na Slici 4. i period posmatranja od 21 minuta registrovano je CPU vreme simulacije na računaru Intel Core 2 Duo CPU E8400 3 GHz, 3.49GB RAM. za intezitete saobraćaja od 0,5 do 5, sve posmatrane metode rutiranja i metode naplate interkonekcije i fiksni broj kanala koji iznosi 30. Rezultati merenja trajanja simulacije prikazani su u Tabeli 1. i pokazuju da je realizovani simulacioni softver efikasan. Inicijalizacija mreže, generisanje zahteva i stastička obrada imaju zanemarljivo trajanje u poređenju sa simulacijom, a učestalost poziva ovih procedura je neuporedivo manja.

Posmatrani softverski sistem je apsolutno **prilagodljiv** - neznatnim modifikacijama koda se može proširiti i prilagoditi novim korisničkim zahtevima što se svakodnevno i čini. Ako je softver korektan, njegova efikasnost i prilagodljivost se mogu popravljati (efikasniji algoritmi, brži hardver, bolja interna struktura modula, ...), dok obratno naravno ne važi.

Tabela 1. *Efikasnost simulacionog softvera za različite metode rutiranja, različite načine naplate interkonekcije i različite intezitete saobraćaja pri fiksnom broju kanala*

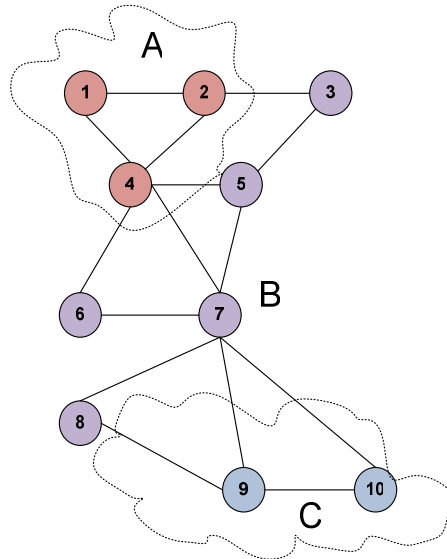
saobraćaj	Metoda	Metoda	Metoda	Metoda	Metoda	Metoda	Metoda	Metoda
	1-0	2-0	3-0	4-0	1-1	2-1	3-1	4-1
	srednje CPU vreme u sekundama							
0,5	2	1	2	2	2	1	3	3
1	4	2	5	6	5	1	5	5
1,5	7	2	7	7	7	3	7	7
2	10	3	9	1 0	1 0	3	1 0	1 0
2,5	13	3	1 1	1 2	1 3	4	1 2	1 1
5	16	5	1 4	1 3	1 5	5	1 4	1 3
3,5	18	6	1 6	1 5	1 8	5	1 6	1 5
4	21	6	1 9	1 8	2 0	6	1 9	1 9
4,5	23	7	2 1	2 1	2 4	7	2 0	2 1
5	24	7	2 1	2 3	2 4	7	2 1	2 3

7. Primer primene softvera

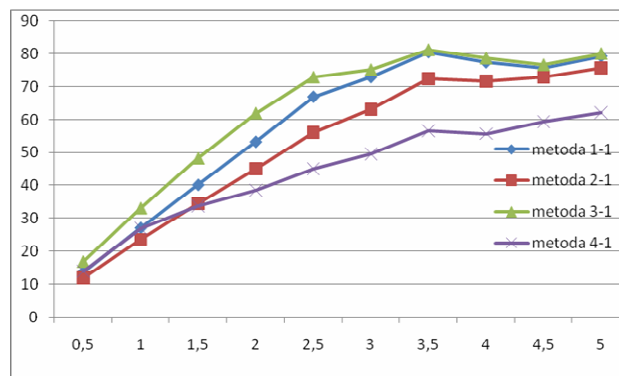
Softver je do sada primenjen na mreže sa različitom topologijom. U ovom radu, posmatraćemo telekomunikacionu mrežu, koja se sastoji od ukupno 10 čvorova, koji su povezani na način prikazan na Slici 4. Mreža se sastoji od tri međusobno povezana segmenta koji odgovaraju operatorima A, B i C.

Softverski sistem dopušta da korisnik definiše period koji se simulira. Za eksperimentisanje je odabran period od 21 minuta rada mreže [4], pri čemu su iz statističke obrade izostavljena prva 3 i poslednja 3 minuta ovog perioda. Takođe, bira se neki od načina naplate interkonekcije opisan u sekciji 2 i neka metoda rutiranja opisana u sekciji 3.

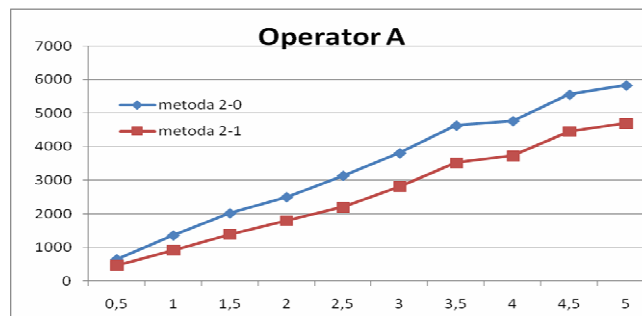
Realizovani softver daje mogućnost da se za kratko vreme obavi veliki broj eksperimenata, da se dobiju brojni izlazni podaci i prati međuzavisnost brojnih veličina. Prikaz samo dve od velikog broja takvih zavisnosti dat je na Slikama 5. i 6.



Slika 4. Topologija mreže u studiji slučaja



Slika 5. Srednja vrednost procenta zauzetosti svih linkova u posmatarnoj mreži pri "cost based" načinu naplate interkonekcije



Slika 6. Poređenje prihoda operatora za A za "bill and keep" i "cost based" način interkonekcije za rute sa maksimalno 3 linka

8. Zaključak

Globalni cilj našeg istraživanja je analiza različitih kombinacija metode rutiranja i metode naplate interkonekcije u cilju optimizacije rada mreže, kao i postizanja željenog balansa između zahteva korisnika i zahteva operatora. Za ove potrebe razvijen je originalni softver koji simulira saobraćaj u telekomunikacionoj mreži korišćenjem različitih načina naplate interkonekcije pri različitim metodama rutiranja saobraćaja. Dodatni zadatak softvera je statistička obrada relevantnih parametara rezultata simulacije.

Akcentat pri projektovanju i implementaciji ovog softvera je stavljen na efikasnost i prilagodljivost. Efikasnost sistema omogućava da se za kratko vreme obavi veliki broj eksperimenata, a prilagodljivost da se neznatnim modifikacijama on proširi (nove metode rutiranja, nove raspodele koje se tiču generisanja zahteva, novi načini naplate interkonekcije, nove statistike, itd.) na čemu autori već rade.

Literatura

- [1] *Directive 97/33/EC* of the European Parliament and of the Council on interconnection in telecommunications, dostupno na ec.europa.eu
- [2] M. Armstrong, "Network Interconnections in Telecommunications", *Economic Journal*, Vol. 108, pp. 545-564, 1998.
- [3] A. Kostić-Ljubisavljević, S. Mladenović, V. Aćimović-Raspopović, "Poređenje nekih metoda naplate interkonekcije", *TELFOR 2009, Zbornik radova na CD-u*, Beograd, 2009.
- [4] *ITU-T Recommendations E series*
- [5] M. Pioro and D. Medhi, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, Elsevier, 2004.
- [6] D. E. Knuth, *The art of computer programming, Volume 2: Seminumerical algorithms*, Addison Wesley, 1969.
- [7] <http://msdn.microsoft.com/library>
- [8] S. Mladenović, A. Kostić-Ljubisavljević, V. Aćimović-Raspopović, "Softver za analizu efekata interkonekcije telekomunikacionih mreža ", *ETLAN 2010, 54. Konferencija za ETRAN, Zbornik radova na CD-u*, Donji Milanovac, 2010.
- [9] IEEE Standard 1012-2004 for Software Verification and Validation
- [10] R. J. Leach, *Introduction to Software Engineering*, Computer Science Programming, 2000.

Abstract: *Under market conditions, it is necessary to manage telecommunication network in a way to balance user's demands for less charging rate and operator's demands for higher revenue. In order to make valid decisions both in design and in exploitation of telecommunication network it is necessary to perform a comparison between different interconnection methods under different routing methods. For the purpose of this research, the original software has been developed whose main task is the traffic simulation in telecommunication network with the possibility to chose dynamic routing method and charging method.*

Keywords: *software, networks interconnection, dynamic routing*

SOFTWARE FOR ANALYZING EFFECTS OF DYNAMIC ROUTING IN NETWORKS WITH INTERCONNECTION

Snežana Mladenović, Aleksandra Kostić-Ljubisavljević