

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Matevž Ogrinc

**Model prometa v vzorčnem fiksni
računalniškem omrežju**

DIPLOMSKO DELO
UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

prof. dr. Miha Mraz
MENTOR

Ljubljana, 2018

© 2018, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavlanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.

Univerza
v Ljubljani

Fakulteta za računalništvo
in informatiko



Tematika naloge:

Kandidat naj v svojem delu vzpostavi model vzorčnega fiksnega računalniškega omrežja v programskem okolju OMNeT++, pri čemer naj karakteristike uporabniškega bremena povzame po diplomskem delu Nives Bricman. Na osnovi postavljenega modela naj kandidat vpelje nedeterministično pogojeno generiranje paketov v odvisnosti od ure dneva in izvede optimizacijo vzorčnega omrežja z vidika potrebnih resursov. Cilj optimizacije naj bo ustrezna prepustnost omrežja, ki bo zadoščala potrebam uporabnikov.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani izjavljam, da sem avtor dela, da slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali drugem visokošolskem zavodu, razen v primerih kjer so navedeni viri.

S svojim podpisom zagotavljam, da:

- sem delo izdelal samostojno pod mentorstvom prof. dr. Mihe Mraza,
- so elektronska oblika dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko in
- soglašam z javno objavo elektronske oblike dela v zbirki "Dela FRI".

— Matevž Ogrinc, Ljubljana, september 2018.

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Matevž Ogrinc

Model prometa v vzorčnem fiksnem računalniškem omrežju

POVZETEK

V diplomskem delu podrobno analiziramo spletni promet v vzorčnem modelu namišljenega fiksnega omrežja na območju Upravne enote Slovenj Gradec. Na začetku opišemo načine porajanja prometa in klasifikacijo sestavnih delov, ki delujejo kot osnova za izdelavo našega modela omrežja.

Nato povzamemo vhodne podatke, ki zajemajo domače in službene uporabnike našega modela omrežja v naseljih Upravne enote Slovenj Gradec. V nadaljevanju razdelimo naselja v območja, kjer ima vsako območje svoj usmerjevalnik, ki usmerja generiran promet iz naselij proti zunanjemu svetu. Razdelitvi naselij sledi vzpostavitev in opis gradnikov našega modela omrežja v simulacijskem okolju OMNeT++ ter razlaga načina generiranja spletnega prometa.

Nadaljujemo s podrobno analizo treh sklopov simulacij delovanja omrežja pod različnimi obremenitvami in časi v dnevu. Pred simulacijami opišemo breme službenih in domačih uporabnikov na našem omrežju, robne pogoje, ki so fiksni skozi vse simulacije ter vhodne podatke za posamezne sklope simulacij. Po vsakem sklopu simulacij opišemo parametre, kot so čas strežbe na strežnikih in zmogljivost pasovnih širin povezav ter pogoje, ki vplivajo na zgostitve in slabo prepustnost v simulacijskem omrežju, ter izpostavimo možne izboljšave za naslednji sklop simulacij. Na koncu podrobneje predstavimo rezultate naših sklopov in vpliv vsake izboljšave na prepustnost ter delovanje omrežja.

Ključne besede: model omrežja, generiranje prometa, simulacija pod obremenitvami, ozka grla v omrežju, optimizacija omrežja

University of Ljubljana
Faculty of Computer and Information Science

Matevž Ogrinc

Case study of a fixed network traffic model

ABSTRACT

In this thesis, we analyse network traffic using a fixed virtual network model for the Administrative Office of Slovenj Gradec. First, we describe different types of traffic distribution and classify components to create a baseline for the model.

Second, we summarise the input data regarding home and work users. Third, we divide the settlements into areas, where each area contains a network router and apply the open source simulation tool OMNeT++ in order to characterize every component of the model. Finally, we describe in detail the manner in which we generate network traffic.

Furthermore, we analyse three sets of network simulations under different traffic loads according to time of day. Prior to conducting simulations, we described for each set of simulations, the fixed parameters, input data and how home and work users influence on traffic network load. After every set of simulations, we analysed key parameters, such as router processing time, connection bandwidth efficiency and other conditions that effect packet permeability and traffic congestion. We then devise possible solutions and test them in the next set of simulations. Finally, we present and summarise the results for each set of simulations and describe how each solution influences packet permeability and traffic congestion.

Key words: network model, traffic generation, simulation under traffic loads, network congestions, network optimization

ZAHVALA

Najprej bi se zahvalil mentorju prof. dr. Mihi Mrazu za strokovno pomoč, podporo in vodenje pri izdelavi diplomske naloge. Posebna zahvala gre moji družini, ki me je podpirala in mi skozi študij stala ob strani. Hvala tudi Tei in vsem mojim prijateljem, ki ste verjeli vame in me spodbujali, ko sem to potreboval.

— Matevž Ogrinc, Ljubljana, september 2018.

KAZALO

| | |
|---|------------|
| Povzetek | i |
| Abstract | iii |
| Zahvala | v |
| 1 Uvod | 1 |
| 1.1 Opredelitev problema | 1 |
| 1.2 Struktura naloge | 2 |
| 2 Opis problema | 3 |
| 2.1 Načini porajanja prometa | 3 |
| 2.1.1 Klasifikacija sestavnih delov omrežja | 4 |
| 2.1.2 Deterministično generiranje prometa | 5 |
| 2.1.3 Nedeterministično generiranje prometa | 5 |
| 2.2 Robni podatki za postavitve modela | 6 |
| 2.2.1 Prebivalstvo in zaposlena populacija | 6 |
| 2.2.2 Razdelitev naselij na območja | 10 |
| 3 Postavitev modela | 13 |
| 3.1 Model celotnega omrežja | 13 |
| 3.2 Sestavljeni moduli | 14 |
| 3.2.1 Modul Naselje | 15 |
| 3.2.2 Modul Usmerjevalnik | 15 |
| 3.2.3 Modul Območje | 16 |
| 3.2.4 Modul Internet | 16 |
| 3.2.5 Modul VhodIzhod | 17 |

| | | |
|----------|---|-----------|
| 3.3 | Enostavni moduli | 18 |
| 3.3.1 | Modul Streznik | 18 |
| 3.3.2 | Modul Generator | 18 |
| 3.3.3 | Modul Vhod | 18 |
| 3.3.4 | Modul Izhod | 19 |
| 3.3.5 | Modul Routing | 19 |
| 3.4 | Način generiranja paketov | 19 |
| 4 | Analiza prometa v modelu omrežja | 23 |
| 4.1 | Parametri simulacije | 23 |
| 4.2 | Rezultati prvega sklopa simulacij | 25 |
| 4.3 | Rezultati drugega sklopa simulacij | 29 |
| 4.4 | Rezultati tretjega sklopa simulacij | 32 |
| 4.5 | Ugotovitve analize | 34 |
| 5 | Zaključek | 37 |

1 Uvod

V diplomskem delu Nives Bricman [1] je izpostavljen problem gradnje modela novega fiksnega omrežja za namišljenega ponudnika internetnih storitev na območju upravne enote Slovenj Gradec, torej Mestne občine Slovenj Gradec in občine Mislinja, ki ju skupno sestavlja 33 manjših naselij na površini 286km^2 . Ponudnik želi v svoje omrežje privabiti in priključiti čim večje število zadovoljnih uporabnikov. Rezultat njenega dela je posplošen model omrežja, ki temelji na grobih ocenah količine omrežnega prometa in poenostavljeni implementaciji omrežja, saj ne odraža v dovoljšnji meri delovanja realnega omrežja. Ponudnik bi rešitev lahko uporabil za postavitev modela omrežja, vendar ne bi imel verodostojnega vpogleda v njegovo delovanje. V pričujoči diplomski nalogi se bomo lotili opisane problematike z bolj realističnim načinom generiranja prometa in nedeterminističnim izvajanjem posameznih opravil v modelu omrežja.

1.1 Opredelitev problema

Način generiranja omrežnega prometa v modelu omrežja je pomemben za pridobivanje bolj točnih simulacijskih rezultatov modela, saj je v realnem svetu promet na žičnih

povezavah in strežnikih zelo razgiban. Velika količina prometa vpliva na uporabniško izkušnjo, saj zaradi nje pride do ozkih grl, izgube paketov in upočasnitev delovanja povezav. V diplomskem delu bomo simulirali promet čimbolj realistično z uporabo podatkov o potencialnih uporabnikih ter njihovih navadah uporabe internetnih storitev povzetih po diplomski nalogi Nives Bricman. Poizkusili bomo ugotoviti, kako vplivajo na model fizičnega omrežja ponudnika internetnih storitev ob različnih časih v dnevu. Dogodki kot so strežni časi, medprihodni časi paketov itd. se morajo oblikovati nedeterministično, saj s tem simuliramo naključnost delovanja omrežja in obnašanje uporabnikov v omrežju.

Ne smemo pa pozabiti na uporabnike internetnih storitev, ki jih v večini tehnične podrobnosti izvedbe ne zanimajo. Zanje sta pomembni le zmogljivost in zanesljivost internetne povezave, kjer je z vidika uporabnika zmogljivost izražena s prihodno (angl. *download*) in odhodno (angl. *upload*) hitrostjo prenosa podatkov. Delovanje omrežja enačijo z delovanjem storitve, kar se v primeru odpovedi storitve pogostokrat izkaže za problematiko motenj na samih storitvah, ne pa za problem omrežja.

Ponudnik internetnih storitev bo lahko na podlagi naših rezultatov pridobil realnejši vpogled na pogoje in probleme pri postavitvi fiksnega omrežja na območju upravne enote Slovenj Gradec ter omogočil boljšo in optimizirano uporabniško izkušnjo svojim novim uporabnikom.

1.2 Struktura naloge

Diplomsko nalogo sestavlja pet poglavij, ki se med seboj nadgrajujejo in dopolnjujejo. Prvo poglavje je namenjeno uvodu, opredelitvi problema in opisu strukture diplomskega dela. Drugo poglavje diplomske naloge namenimo problematiki, s katero se bomo soočali v nadaljevanju pričujočega dela, hkrati pa opišemo teoretični pogled na porajanje prometa in sestavnih delov omrežja. V njem povzamemo poznane karakteristike opazovanega območja Upravne enote Slovenj Gradec, s katerimi pridobimo podlago za določitev obremenitve našega omrežja. V tretjem poglavju opišemo vzpostavitev gradnikov našega modela omrežja v simulacijskem okolju OMNeT++ in podrobneje predstavimo izbrani način generiranja prometa. Četrto poglavje vsebuje opis fiksnih vhodnih parametrov simulacij ter rezultate simulacij in izboljšave treh predlaganih modelov omrežnega prometa v vzpostavljenem osnovnem modelu omrežja. Peto poglavje je namenjeno povzetku in zaključkom dela.

2 Opis problema

Uporabniki so v današnjem času navajeni hitrega in zanesljivega delovanja interneta. Ob zgotovitvi prometa pride do izgubljanja paketov in daljših čakalnih dob, kar za uporabnike pomeni počasnejše in s tem slabše delovanje internetne povezave. Zato potrebujejo ponudniki internetnih storitev zanesljiv model, ki simulira delovanje omrežja ob različnih obremenitvah in težavah. Da zagotovimo verodostojen model, moramo simulacijsko omrežje čimbolj približati realnemu delovanju. Za izhodišče naše diplomske naloge bomo povzeli simulacijsko omrežje diplomskega dela Nives Bricman in ga ustrezno nadgradili in spremenili. Posledično bomo pridobili nov simulacijski model, ki bo našemu namišljenemu ponudniku internetnih storitev prikazal realnejše delovanje fiksnega omrežja.

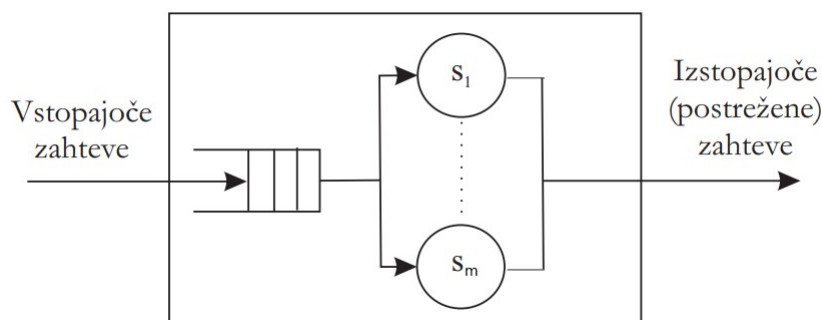
2.1 Načini porajanja prometa

Generiranje prometa je faktor, ki najbolj vpliva na zmogljivost in verodostojnost modela omrežja. V teoriji strežbe imamo način porajanja prometa ponavadi opisan s Kendallovo notacijo. Kendallova notacija je ena od najbolj uveljavljenih klasifikacij strežnih enot, ki jo je postavil britanski statistik D. G. Kendall [2]. Petorček Kendallove notacije je

sestavljen iz petih parametrov, ki opisujejo poljubno strežno enoto. Strežna enota je model strežnega sistema, ki ponuja le en tip strežbe, ima m paralelno vezanih strežnikov ($m \geq 1$) in pred strežniki eno čakalno vrsto. V našem primeru se bomo osredotočili na prve tri parametre Kendalllove notacije, to so $A/B/m$, ki zadovoljijo našo potrebo po opisu strežne enote, kjer A predstavlja verjetnostno porazdelitev medprihodnih časov vstopajočih zahtev, B verjetnostno porazdelitev strežnih časov zahtev in m število paralelno vezanih strežnikov. Vsak sestavni del našega modela omrežja bo v abstraktnem smislu predstavljal strežno enoto.

2.1.1 Klasifikacija sestavnih delov omrežja

V našem simulacijskem omrežju so sestavni deli opisani kot strežne enote, katerih osnove so povzete po [3]. Slika 2.1 predstavlja osnovno strežno enoto, kjer imamo eno čakalno vrsto in m paralelno vezanih strežnikov (S_1, \dots, S_m). Sestavni deli modela bodo imeli isto osnovo, vendar le z enim strežnikom, ki streže paketom. Strežnik lahko obdeluje naenkrat



Slika 2.1 Shema strežne enote [3].

le eno zahtevo ali v kontekstu pričujočega dela le en paket. Njegova značilnost je intenzivnost obdelave paketov, ki jo merimo s številom obdelanih zahtev na časovno enoto. Strežni časi so lahko konstantni ali pa določeni z neko porazdelitvijo. V našem primeru uporabljamo konstanten strežni čas $5 \mu s$ povzet po viru [4]. Kapaciteta strežne enote je enaka maksimalnemu številu zahtev, ki se lahko zadržujejo v njej. Lahko je končna ali neskončna. V našem primeru bomo uporabili obe. Na koncu imamo še vhodni in izhodni proces. Proces opisujeta vstopanje in izstopanje zahtev v strežno enoto. Vhodni proces označujemo z λ in predstavlja intenzivnost porajanja paketov na časovno enoto. Medprihodni časi paketov so lahko skozi čas konstantni ali pa sledijo eni izmed porazdelitev. Na

drugi strani strežne enote imamo izhodni proces, ki je odvisen od vhodnega procesa in intenzivnosti strežbe paketov na strežniku. V našem modelu omrežja uporabljamo strežne enote z enim strežnikom, kar pomeni, da je intenzivnost obdelave strežne enote enaka intenzivnosti strežbe paketov na strežniku. V primeru, da potrebujemo več paralelno vezanih strežnikov, se intenzivnost obdelave strežne enote poveča za tolikokrat, kolikor je število strežnikov. Predhodno omenjene zahteve v našem diplomskem delu enačimo s paketi in jih bomo v nadaljevanju tudi tako imenovali.

2.1.2 Deterministično generiranje prometa

Kot smo omenili v prejšnjem poglavju, je eden izmed glavnih vzrokov nenatančnosti simulacijskega omrežja uporaba determinističnega generiranja omrežnih paketov. Kendallova notacija za tak primer strežne enote je $D/D/1$, kjer D predstavlja deterministično porazdelitev medprihodnih časov vstopajočih zahtev in strežnih časov zahtev. Število paralelno vezanih strežnikov je v tem primeru 1. Deterministična porazdelitev medprihodnih časov pomeni, da se paketi pošiljajo v konstantnih časovnih intervalih. Tak način generiranja paketov nam ne pokaže ranljivosti omrežja na ozka grla, kar za novega ponudnika internetnih storitev lahko pomeni veliko izgubo in za uporabnike slabšo izkušnjo.

2.1.3 Nedeterministično generiranje prometa

Za rešitev problema determinističnega generiranja prometa v našem omrežju smo si ogledali različne vrste verjetnostnih porazdelitev, ki bi porajanje paketov čimbolje približale realnosti. Odločili smo se za uporabo Poissonovega modela povzetega po viru [5], ki nam napove verjetnost, da se pojavi določeno število dogodkov, v našem primeru rojstev paketov, v nekem časovnem obdobju, če poznamo pogostost pojavljanja rojstev neodvisno od časa, ki je potekel od zadnjega dogodka. Verjetnost Poissonove porazdelitve pridobimo po izrazu

$$P(X(t) = k) = \frac{(\lambda t)^k}{k!} e^{-\lambda t}, \quad (2.1)$$

kjer je e osnova naravnih logaritmov, k število porajanj paketov in λ intenzivnost porajanja paketov v časovnem intervalu $(0, t)$. Poissonova porazdelitev je hkrati tudi del Poissonovega procesa, ki je tesno povezan z eksponentno verjetnostno porazdelitvijo. Za lažje razumevanje relacije med njima si najprej oglejmo Poissonov proces. To je proces štetja naključno porajajočih se k dogodkov v časovnem intervalu. Poissonova verjetnostna porazdelitev se v procesu pojavi kot porazdelitev med seboj neodvisnih porajajočih

se dogodkov, kar označujemo z $X(t)$. Eksponentna porazdelitev je zvezna verjetnostna porazdelitev, ki opisuje časovne intervale med posameznimi dogodki. Njena zbirna funkcija verjetnosti je podana z izrazom

$$F(x; \lambda) = \begin{cases} 1 - e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0. \end{cases}, \quad (2.2)$$

kjer je λ parameter stopnje.

Kot smo predhodno omenili, je lastnost Poissonovega procesa in eksponentne verjetnostne porazdelitve njuna tesna povezava. Če so medprihodni časi zahtev v strežni enoti porazdeljeni po eksponentni porazdelitvi, potem je število zahtev v časovnem intervalu porazdeljeno s Poissonovo verjetnostno porazdelitvijo in proces imenujemo za Poissonov proces. V primeru, kadar je število porajanih zahtev v nekem intervalu Poissonova naključna spremenljivka označena z λt , so medprihodni časi eksponentno porazdeljeni in proces porajanja tudi poimenujemo za Poissonov proces. Posledično lahko opišemo strežno enoto s Kendallov notacijo $M/M/1$, kjer je M eksponentna porazdelitev Poissonovega procesa. S pomočjo Poissonovega procesa, eksponentne in Poissonove verjetnostne porazdelitve lahko zagotovimo bolj realno porajanje prometa v našem simulacijskem omrežju. Vendar, kot je omenjeno v viru [5], je dandanes omrežni promet bolj kompleksen in posledično postaja proces določanja medprihodnih časov vse bolj v neskladju z Poissonovo distribucijo.

2.2 Robni podatki za postavitev modela

V pričujočem razdelku bomo na kratko obnovili statistične in ostale karakteristične podatke upravne enote Slovenj Gradec opisane v delu Nives Bricman. Zanimajo nas podatki o prebivalcih naselij in zaposlenost populacije, saj prispevajo k oceni količine prenosa do uporabnika in količine prenosa od uporabnika. Obravnavana naselja in ocena prenosa sestavljajo območja, v katerih se promet v našem simulacijskem omrežju generira in sprejema.

2.2.1 Prebivalstvo in zaposlena populacija

Uporabnike v našem simulacijskem omrežju razdelimo na dve skupini. To so prebivalci poimenovani kot *domači uporabniki* in zaposlena populacija poimenovana *službeni uporabniki*. Domači uporabniki so tisti, ki stanujejo v mestni občini Slovenj Gradec in občini

Mislinja, medtem ko službeni uporabniki predstavljajo uporabnike, ki so zaposleni v eni ali drugi občini. Razlika med skupinama je v njihovih uporabniških navadah uporabe internetnih storitev. Domači uporabniki internetne storitve uporabljajo preko celega dneva z vzponi zvečer in padci po polnoči. Promet službenih uporabnikov doživi vzpone ob jutranjih urah in padce proti večeru. V upravni enoti Slovenj Gradec imamo 21.378 prebivalcev. Pri tem predpostavljamo, da jih 14.219 uporablja internetne storitve našega navideznega ponudnika [1]. Tabela 2.1 predstavlja število ljudi po posameznih naseljih in oceno števila domačih uporabnikov glede na posamezno občino. Želimo pa opomniti, da so za nas podatki o domačih in službenih uporabnikov le številke, ki jih lahko v modelu enostavno zamenjamo in je njihova verodostojnost nepomembna.

| Mestna občina Slovenj Gradec | | |
|--------------------------------|---------------------|-----------------------------------|
| Ime naselja | število prebivalcev | Ocena števila domačih uporabnikov |
| 1. Brda | 291 | 195 |
| 2. Gmajna | 466 | 334 |
| 3. Golavabuka | 176 | 124 |
| 4. Gradišče | 312 | 221 |
| 5. Graška Gora | 110 | 77 |
| 6. Legen | 1.061 | 703 |
| 7. Mislinjska Dobrava | 745 | 501 |
| 8. Pameče | 1.314 | 944 |
| 9. Podgorje | 977 | 674 |
| 10. Raduše | 228 | 153 |
| 11. Sele | 238 | 161 |
| 12. Slovenj Gradec | 7.477 | 4.837 |
| 13. Spodnji Razbor | 192 | 132 |
| 14. Stari trg | 642 | 420 |
| 15. Šmartno pri Slovenj Gradec | 1.237 | 810 |

| | | |
|-------------------------------|-------|-------|
| 16. Šmiklavž | 142 | 95 |
| 17. Tomaška vas | 129 | 74 |
| 18. Troblje | 325 | 214 |
| 19. Turiška vas | 143 | 90 |
| 20. Vodriž | 111 | 70 |
| 21. Vrhe | 329 | 223 |
| 22. Zgornji Razbor | 140 | 96 |
| Mestna občina Mislinja | | |
| 23. Dovže | 330 | 231 |
| 24. Gornji Dolič | 508 | 362 |
| 25. Kozjak | 408 | 273 |
| 26. Mala Mislinja | 192 | 140 |
| 27. Mislinja | 1.869 | 1.209 |
| 28. Paka - del | 48 | 29 |
| 29. Razborca | 104 | 71 |
| 30. Srednji Dolič | 249 | 163 |
| 31. Šentilj pod Turjakom | 305 | 201 |
| 32. Tolsti Vrh pri Mislinji | 156 | 97 |
| 33. Završe | 424 | 295 |

Tabela 2.1: Število vseh prebivalcev in okvirno število domačih uporabnikov po naseljih v mestnih občinah Slovenj Gradec in Mislinja [1].

Poleg domačih uporabnikov imamo opravka tudi s službenimi uporabniki. Službeni uporabniki predstavljajo ljudi, ki so v upravni enoti Slovenj Gradec zaposleni in preko zaposlitve uporabljajo internetne storitve našega namišljenega ponudnika. Tabeli 2.2 in 2.3 predstavljata predpostavljeno število službenih uporabnikov na območjih mestnih občin Slovenj gradec in Mislinja.

| Mestna občina Slovenj Gradec | | |
|--------------------------------|--------------------|--------------------------------------|
| Ime naselja | število zaposlenih | Ocena števila zaposlenih uporabnikov |
| 1. Brda | 31 | 16 |
| 2. Gmajna | 197 | 99 |
| 3. Golavabuka | 14 | 7 |
| 4. Gradišče | 29 | 15 |
| 5. Graška Gora | / | 0 |
| 6. Legen | 145 | 73 |
| 7. Mislinjska Dobrava | 256 | 128 |
| 8. Pameče | 822 | 411 |
| 9. Podgorje | 198 | 99 |
| 10. Raduše | 18 | 9 |
| 11. Sele | 40 | 20 |
| 12. Slovenj Gradec | 5.285 | 2.643 |
| 13. Spodnji Razbor | / | 0 |
| 14. Stari trg | 68 | 34 |
| 15. Šmartno pri Slovenj Gradec | 179 | 90 |
| 16. Šmiklavž | / | 0 |
| 17. Tomaška vas | 7 | 4 |
| 18. Troblje | 69 | 35 |
| 19. Turiška vas | 10 | 5 |
| 20. Vodriž | / | 0 |
| 21. Vrhe | 30 | 15 |
| 22. Zgornji Razbor | / | 0 |

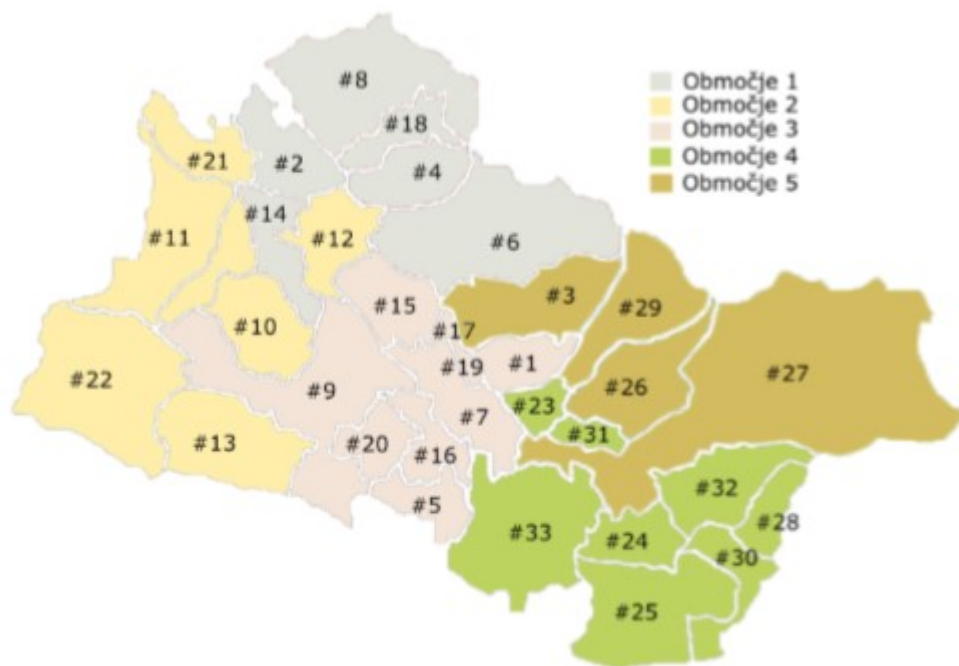
Tabela 2.2: Število vseh zaposlenih in okvirno število službenih uporabnikov po naseljih v mestni občini Slovenj Gradec [1].

| Mestna občina Mislinja | | |
|-----------------------------|--------------------|--------------------------------------|
| Ime naselja | število zaposlenih | Ocena števila zaposlenih uporabnikov |
| 23. Dovže | 23 | 12 |
| 24. Gornji Dolič | 62 | 31 |
| 25. Kozjak | 41 | 21 |
| 26. Mala Mislinja | 23 | 12 |
| 27. Mislinija | 234 | 117 |
| 28. Paka - del | / | 0 |
| 29. Razborca | 31 | 16 |
| 30. Srednji Dolič | / | 0 |
| 31. Šentilj pod Turjakom | 89 | 45 |
| 32. Tolsti Vrh pri Mislinji | 33 | 17 |
| 33. Završe | 47 | 24 |

Tabela 2.3: Število vseh zaposlenih in okvirno število službenih uporabnikov po naseljih v mestni občini Mislinja [1].

2.2.2 Razdelitev naselij na območja

Za lažjo predstavitev in postavitev modela fiksnega omrežja smo naselja obeh občin združili skupaj v območja po vzoru opisanem v viru [1]. Vsako območje ima svoj usmerjevalnik za posredovanje paketov iz naselij do zunanjega sveta in obratno. Naselja smo razdelili na 5 območij glede na njihovo lokacijo in število prebivalstva ter območja povezali z dvema glavnima usmerjevalnikoma. Območja imajo glede na število domačih in službenih uporabnikov različne pasovne širine na povezavah z zunanjim svetom, ki naj bi zadostovale za nemoteno komunikacijo vsakega naselja. Slika 2.2 ponazarja razdelitev prej omenjenih in oštevilčenih naselij na območja.



Slika 2.2 Razdelitev naselij na območja [1].

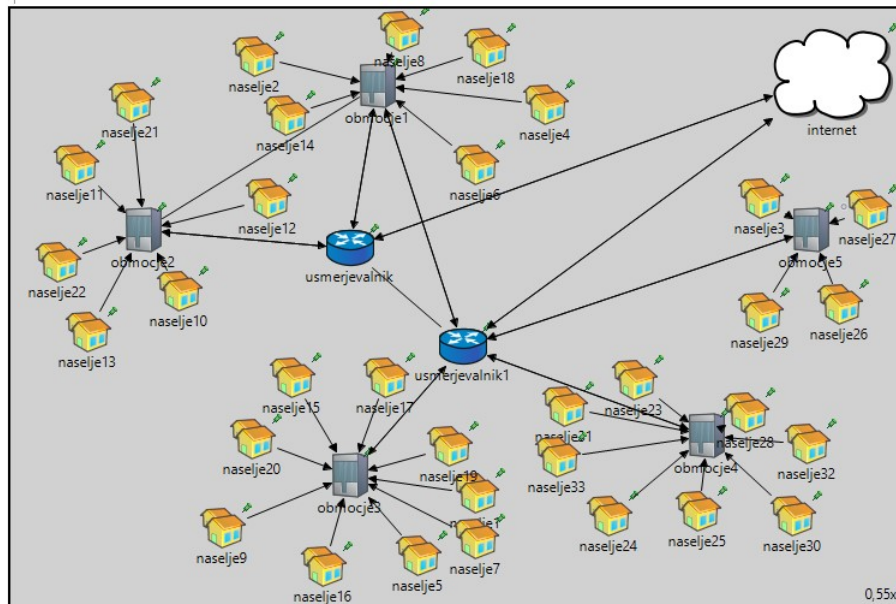
3 Postavitev modela

Za postavitev modela smo uporabili programsko okolje OMNeT++ [6], ki je eno izmed bolj razširjenih brezplačnih simulacijskih okolij. Gradniki OMNeT++, opisani podrobneje v viru [7], so enostavni in sestavljeni moduli, kjer so *enostavni moduli* (angl. *simple module*) opisani s programskim jezikom C++, *sestavljene moduli* (angl. *compound module*) pa z NED opisnim jezikom. Poleg modulov smo uporabili tudi razred cTopology, ki nam pomaga pri definiciji vsakega gradnika in pri sestavljanju usmerjevalnih tabel.

3.1 Model celotnega omrežja

Slika 3.1 prikazuje model celotnega omrežja našega navideznega ponudnika internetnih storitev na območju upravne enote Slovenj Gradec, torej Mestne občine Slovenj Gradec in občine Mislinja. Omrežje vsebuje sestavljene module *Obmocje*, ki ponazarjajo specifična območja pod katera spadajo določena naselja opisana v poglavju 2. Poleg *Obmocij* so prikazani še sestavljeni moduli *Naselje*, *Usmerjevalnik* in *Internet*. Prikazane gradnike opišemo podrobneje v nadaljevanju poglavja. Hkrati moramo omeniti, da model vsebuje le module do ponudnikove zadnje aktivne naprave, ne pa uporabnikovega usmerjevalnika,

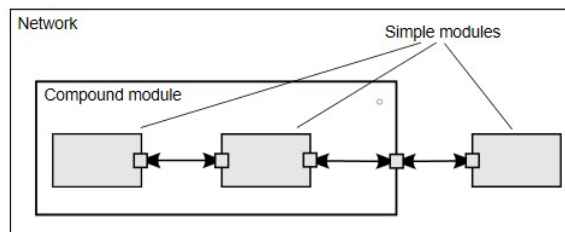
saj imamo več izvorov paketov modelirano v samem modulu `Naselje`.



Slika 3.1 Slika celotnega omrežja.

3.2 Sestavljeni moduli

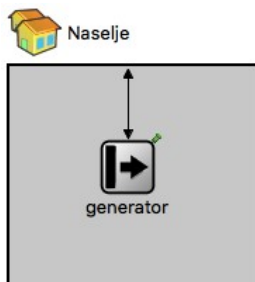
V našem simulacijskem modelu omrežja so glavni gradniki sestavljeni moduli (glej sliko 3.2). Sestavljeni so iz več enostavnih modulov, ki sestavljenemu modulu dajejo funkcionalnosti. Enostavni moduli imajo definirana tudi vrata (angl. *port*) preko katerih teče komunikacija in prenos paketov med enostavnimi moduli. Prav tako imajo vrata tudi sestavljeni moduli preko katerih teče komunikacija in prenos med zunanjim svetom in notranjo vsebino modula. Med sestavljene module spadajo `Naselje`, `Obmocje`, `Internet`, `Usmerjevalnik` in `VhodIzhod`.



Slika 3.2 Shema sestavljenega modula.

3.2.1 Modul Naselje

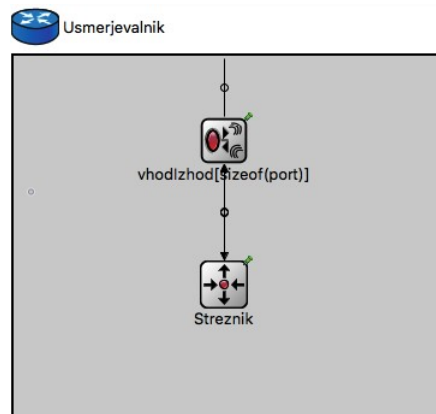
Naselje je v našem omrežju hkrati izvor in ponor paketov. Ponazarja posamezno naselje in njihove prebivalce. Kot je prikazano na sliki 3.3, vsebuje le en enostavni modul in sicer generator paketov s katerim generiramo odhodni promet uporabnika. V notranjosti generatorja izračunamo povprečno porabo internetnih storitev na prebivalca ter s pomočjo Poissonove porazdelitve nedeterministično generiramo pakete. Kot končno vozlišče se istočasno predstavlja zunanjemu svetu in sprejema promet generiran iz sestavljenega modula **Internet**.



Slika 3.3 Komponente sestavljenega modula Naselje.

3.2.2 Modul Usmerjevalnik

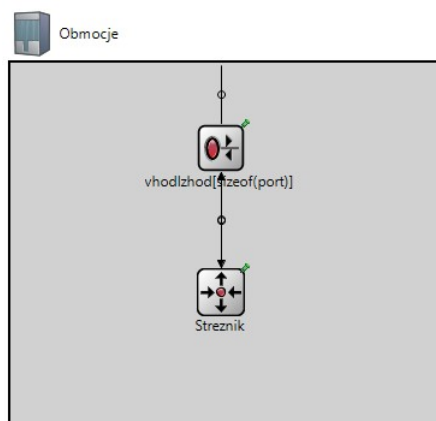
V našem simulacijskem omrežju je sestavljeni modul **Usmerjevalnik** (glej sliko 3.4) najpomembnejši. Vsebuje enostavni modul **Strežnik** in sestavljeni modul **VhodIzhod**. **VhodIzhod** izvaja nalogi sprejemanja in oddajanja paketov. Pakete iz zunanjega sveta sprejme in nato posreduje proti strežniku, medtem ko pakete poslana iz strežnika odda proti zunanjemu svetu. Modul **Strežnik** skrbi za obdelavo in posredovanje paketov na ustrezna vrata preko katerih **VhodIzhod** ugotovi kam v zunanji svet nadalje poslata pakete.



Slika 3.4 Komponente sestavljenega modula Usmerjevalnik.

3.2.3 Modul Obmocje

Sestavljeni modul **Obmocje** ponazarja cilj prvega skoka od skupine naselij. Na sliki 3.5 vidimo, da je **Obmocje** sestavljeno iz istih komponent kot sestavljeni modul **Usmerjevalnik**, saj opravlja isto nalogo z razliko v tem, da ima več izhodnih povezav. Enostavni modul **Streznik** v tem primeru hrani usmerjevalno tabelo, s katero lahko posreduje pakete do končnega naselja in uporabnika.

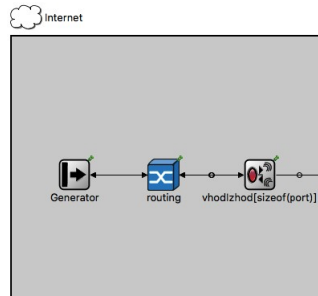


Slika 3.5 Komponente sestavljenega modula Obmocje.

3.2.4 Modul Internet

Za cilj generiranih paketov iz modulov **Obmocje** smo ustvarili sestavljeni modul **Internet**, ki predstavlja zunanji svet. Na sliki 3.6 vidimo, da je sestavljen enako kakor **Obmocje**,

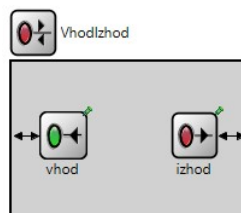
sprememba delovanja pa se izkaže pri generiranju paketov, saj skrbi za generiranje paketov proti Obmocju in predstavlja vir prometa proti uporabniku omrežja.



Slika 3.6 Komponente sestavljenega modula Internet.

3.2.5 Modul VhodIzhod

VhodIzhod (glej sliko 3.7) opravlja nalogo mediatorja med zunanjim svetom in notranjimi komponentami sestavljenega modula. Prisluškuje izhodnim vratom preko katerih prihajajo paketi generirani od zunaj, hkrati pa prisluškuje notranjim vratom preko katerih prihajajo paketi generirani znotraj sestavljenega modula. VhodIzhod je sestavljen iz dveh enostavnih modulov. To sta Vhod in Izhod. Oba vsebujeta čakalno vrsto, ki sprejme pakete in jih obdrži, dokler se povezava med njima in naslednjim enostavnim modulom ne sprosti. Delovanje čakalne vrste se najbolje odraža v sestavljenem modulu Usmerjevalnik, saj morajo paketi počakati, da je Streznik prost za obdelavo novega paketa. Zaradi uporabe nedeterminističnega načina generiranja in velikega števila paketov smo morali modulu Izhod dodati čakalno vrsto, saj vsak paket potrebuje nekaj časa, da se po zunanji povezavi prenese.



Slika 3.7 Komponente sestavljenega modula VhodIzhod.

3.3 Enostavni moduli

Enostavni moduli so osnovni gradniki sestavljenih modulov. Delovanje modulov je sprogramirano s programskim jezikom C++. V našem omrežju imamo enostavne module **Streznik**, **Generator**, **Vhod**, **Izhod** in **Routing**. Vsak od naštetih modulov opravlja svoje naloge, ki so v nadaljevanju bolj podrobno opisane.

3.3.1 Modul Streznik

Glavna naloga enostavnega modula **Streznik** je pravilno posredovanje prihajajočih paketov v sestavljenem modulu **Usmerjevalnik**. Razlika delovanja med moduloma **Routing** in **Streznik** je ta, da **Streznik** posreduje pakete naprej po simulacijskem omrežju medtem, ko **Routing** posreduje pakete do modula tipa **Generator** znotraj sestavljenega modula **Internet**. Ker **Streznik** potrebuje nekaj časa, da postreže pakete, smo implementirali čakalne vrste. Kadar **Streznik** ne obdeluje paketa, počaka na klic modula **Vhod**. Ko ga prejme, pošlje sporočilo samemu sebi. Ob prihodu sporočila začne po krožni disciplini (angl. *round robin*) zahtevati pakete iz nepraznih čakalnih vrst. Vsako zahtevo opravi s klicom funkcije `StartTransmitting()`, ki je vsebovana v modulu **Vhod**. Po koncu obdelave paketa ga posreduje na pravilno izhodno povezavo do modula **Izhod** in zopet samemu sebi pošlje sporočilo za zahtevo naslednjega paketa. Ko so vse čakalne vrste prazne, preide nazaj v originalno stanje poslušanja.

3.3.2 Modul Generator

Enostavni modul **Generator** izvaja nedeterministično generiranje dejanskih paketov v našem omrežju. **Generator** je ključni del v dveh sestavljenih modulih našega omrežja. To sta **Internet** in **Naselje**. **Internet** skrbi za sprejem vseh paketov poslanih iz modulov **Naselje** (angl. *upload*). Za vsak paket, ki ga sprejme, ustvari 10 novih paketov po viru [8], s katerimi ponazarja promet proti uporabnikom (angl. *download*). Generatorji v modulu **Naselje** opravljajo podobno nalogo z razliko v tem, da so njihovi generirani paketi usmerjeni proti modulu **Internet** in ne odgovarjajo na prejete pakete.

3.3.3 Modul Vhod

Vhod je enostavni modul, ki sprejema pakete iz zunanosti usmerjevalnika in jih vstavlja v čakalno vrsto. V sestavljenem modulu **Usmerjevalnik** je potrebno pakete zadržati, dokler je **Streznik** prost in pripravljen za obdelavo naslednjega paketa. To zagotovimo z

dodatno metodo `VrniStreznikovoStanje()`, ki modulu `Streznik` pove, da je v čakanju nov paket. Poleg opisane metode vsebuje tudi metodo `StartTransmitting()`, ki je zadolžena za pošiljanje naslednjega paketa v vrsti (angl. *first in first out* - FIFO). Čakalna vrsta ima omejeno dolžino in v primeru zgostitve dodatne pakete zavrže.

3.3.4 Modul Izhod

Podobno kot modul `Vhod`, je `Izhod` zadolžen za zadrževanje paketov v čakalni vrsti dokler ni zunanja povezava med `Usmerjevalnikom` in naslednjim sestavljenim modulom prosta. Ko se povezava sprosti, `Izhod` začne s pošiljanjem naslednjega paketa.

3.3.5 Modul Routing

V sestavljenem modulu, v katerem se nahaja `Routing`, smo predpostavili ničelni strežni čas. Ob vsakem prispelem paketu preveri njegovo vsebino, prebere naslov prejelnika paketa in na podlagi naslova ugotovi, kam je paket namenjen. Nato paket primerno posreduje. `Routing` poleg usmerjanja paketov skrbi tudi za usmerjevalno tabelo. Na začetku simulacije preveri in zapiše s pomočjo funkcije `calculateUnweightedSingleShortestPathsTo(targetNode)` v modulu `Streznik` najkrajše poti do ostalih modulov v omrežju. Z ustvarjeno usmerjevalno tabelo pravilno in optimalno posreduje in sprejema generirane pakete. Modul `Routing` potrebujemo za sestavljeni modul `Internet`, da lahko v ničelnem strežnem času preusmeri paket na pravilno povezavo proti naseljem.

3.4 Način generiranja paketov

Generirano breme s strani prebivalca na uro smo pridobili iz diplomske naloge Nives Brimman, kjer smo za vsako uro izračunali maksimalno obremenitev in minimalno obremenitev in vzeli povprečje. Poleg domačih uporabnikov imamo tudi službene uporabnike, katerih obremenitev se doda, kadar so aktivni v službi. Temu času pravimo poslovna ura. Poslovne ure veljajo od 5:30 do 15:30. Slika 3.8 prikazuje omenjene obremenitve na posameznega domačega uporabnika skupaj s službenimi uporabniki ob poslovnih urah.



Slika 3.8 Generirano breme posameznika po urah v dnevu.

Obremenitev na sliki 3.8 uporabimo pri generiranju prometa v vseh naseljih, kjer jo vstavimo v OMNeT++ funkcijo `poisson`, ki nam vrne po Poissonu distribuirano število s katero izračunamo število paketov, ki jih generator mora zgenerirati v določenem časovnem intervalu. Nato generator z fiksnim medprihodnim časom generira pakete. Ko mine 0,01s na novo zgeneriramo s pomočjo prej omenjene `poisson` funkcije distribuirano število za ponovni izračun števila paketov. Ta proces ponavljamo skozi celotni čas simulacije in nam prinaša različne intervale intenzivnosti generiranja paketov. Ker imamo fiksni medprihodni čas paketov je potrebno en paket generirati s seštevkom velikosti paketov, ki bi se morali v tem času zgenerirati. Če želimo pridobiti točno število paketov, ki so prispeli do cilja, moramo na končnih vozliščih deliti velikost pridobljenega paketa z velikostjo enega paketa. Še enkrat omenimo, da je to posledica prilagoditve za OMNeT++, saj boljšega približka generiranju prometa programsko okolje ne omogoča. Končna formula za izračun število paketov je podana z izrazom

$$\text{steviloPaketov} = \left(\frac{\text{obremenitev} * \text{steviloUporabnikov}}{\text{velikostPaketov}} \right) * 0,01, \quad (3.1)$$

kjer je `obremenitev` število vrnjeno od funkcije `poisson`, `steviloUporabnikov` število uporabnikov v določenem mestu in v primeru poslovnih ur tudi službenih uporabnikov, `velikostPaketov` predstavlja fiksno velikost enega paketa v naši simulaciji in število 0,01, ki predstavlja čas v katerem se mora končna količina paketov v fiksnem medprihodnem

času zgenerirati. V primerjavi s Poissonovo distribucijo medprijemnih časov paketov naš način generiranja v boljši meri simulira delovanje omrežja.

4 Analiza prometa v modelu omrežja

V pričujočem poglavju opišemo analizo simulacij, ozka grla in druge rezultate, ki nam omogočajo vpogled v delovanje modela fiksnega omrežja glede na njegovo nastavitve in uro v dnevno. Na začetku smo opisali parametre in potek simulacij, kasneje pa primerjali naše rezultate in ugotovitve z rezultati iz diplomske naloge Nives Bricman ter navedli možne izboljšave. Vnaprej pa moramo izpostaviti, da ima program OMNeT++ omejitve glede delovanja, saj pri visokih obremenitvah simulacije ne more dokončati. Posledično smo morali obremenitev in navade uporabnikov prilagoditi programskemu okolju.

4.1 Parametri simulacije

Robni pogoji so statični parametri v našem omrežju, ki se jih ne spreminja. V našem primeru so robni pogoji *čas strežbe* na strežni enoti, *število odzivnih paketov* iz zunanjega sveta, *zakasnitve* in *maksimalni čas simulacije*. Omenjene zakasnitve predstavljajo zakasnitve, ki se pojavijo po strežbi na strežniku in na povezavah med sestavljenimi moduli. Zapisujemo jih v .ini datoteki in veljajo za celotno omrežje pri vsaki simulaciji. Na sliki 4.1 so definirani robni pogoji za vse elemente, kjer *sim-time-limit* predstavlja maksimalni

čas simulacije, `delay` je zakasnitev, `casStrezbe` predstavlja čas strežbe, odzivni paketi pa predstavljajo število paketov generiranih kot odgovor iz zunanjega sveta na poslani paket. Za 10 odzivnih paketov smo se odločili zaradi predpostavke, da ponudniki internetnih storitev ponujajo do 10-krat večjo *download* hitrost od *upload* hitrosti povzeto po viru [8].

```
[General]
debug-on-errors = true
sim-time-limit = 1h
**.Generator.odzivniPaketi = 10
**.casStrezbe = 5us
**.delay = 0.1us
```

Slika 4.1 Definirani robni pogoji v .ini datoteki.

Poleg robnih pogojev imamo definirane tudi pasovne širine posameznih povezav in velikost čakalnih vrst. Pasovne širine, ki jih uporabljamo v naši simulaciji, smo originalno povzeli po Nives Bricman, vendar smo kasneje tudi ugotovili, da ne bodo vzdržale, saj je obremenitev omrežja prevelika. Tabela 4.1 prikazuje originalne vrednosti na povezavah med območji in usmerjevalniki ter usmerjevalniki in zunanjim svetom.

| Pasovne širine povezav | | |
|-------------------------|-----------------|---------------|
| Modul | Download [Mbps] | Upload [Mbps] |
| obmocje1-usmerjevalnik1 | 1.368 | 579 |
| obmocje2-usmerjevalnik1 | 2.709 | 1.146 |
| usmerjevalnik1-internet | 4.077 | 1.725 |
| obmocje3-usmerjevalnik2 | 1.246 | 528 |
| obmocje4-usmerjevalnik2 | 795 | 337 |
| obmocje5-usmerjevalnik2 | 744 | 315 |
| usmerjevalnik2-internet | 2.785 | 1.179 |

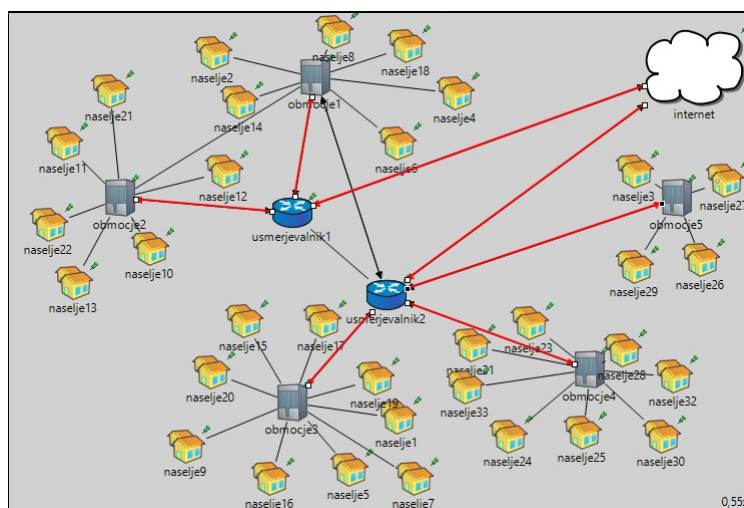
Tabela 4.1: Povzete pasovne širine povezav [1].

Zadnji parameter, ki ga moramo opisati, so velikosti čakalnih vrst. Čakalne vrste so pred posameznim strežnikom in pred vsako povezavo v sestavljenih moduli *Obmocje* in *Usmerjevalnik*. Velikost čakalne vrste pred povezavami je neomejena, medtem ko je velikost čakalne vrste pred strežnikom fiksne velikosti in je zmožna hrambe največ 10.000 paketov.

Za naše simulacijsko omrežje smo opravili trikrat po 24 simulacij, saj smo simulirali po urah v dnevu. Po končanem ciklu smo model omrežja izboljšali in na novo zagnali simulacije. Za vsak sklop simulacij bomo opisali njihove vhodne podatke in prikazali rezultate, ozka grla in predlagane rešitve. Vsak paket ne glede na to, ali je generiran s strani območij, ali s strani zunanjega sveta, ima od začetka do konca tri skoke in dva postanka, kjer se postreže na strežniku. Na vsakem postanku izmerimo čas, ki ga paket preživi v čakalni vrsti in v strežbi. Podatki, ki nas v simulacijah zanimajo, so maksimalni, minimalni in povprečni čas čakanja enega paketa v čakalnih vrstah, minimalni in maksimalni čas, ki ga en paket preživi v celotnem modelu omrežja in število generiranih ter število izgubljenih paketov. Analiza omenjenih podatkov nam bo prikazala lokacije in razloge za ozka grla ter nas z rešitvami približala k željenemu funkcionalnemu fiksному omrežju za našega navideznega ponudnika internetnih storitev.

4.2 Rezultati prvega sklopa simulacij

Za prvi sklop simulacij smo uporabili podatke pridobljene iz diplomske naloge Nives Bricman. Že pri prvi simulaciji, ki obsega obremenitev med 00:30 in 01:30, smo zaznali kar 9 ozkih grl. Vsako od območij je imelo polno čakalno vrsto za pakete iz naselij proti internetu. Hkrati se je zapolnila čakalna vrsta v modulu **Internet** proti naseljem. Slika 4.2 prikazuje povezave, ki so premalo zmogljive za količino prometa, ki se generira proti območjem in iz njih.



Slika 4.2 Povezave z ozkimi grli pri prvem sklopu simulacij.

Trend opazimo skozi vse simulacije ne glede na del dneva, s tem da se količina izgubljenih paketov in časi v čakalnih vrstah povečujejo in zmanjšujejo glede na uro v dnevu. V vsakem primeru pa pride do zgostitve na istih povezavah. Paketi, ki pridejo mimo zgostitev na prej omenjenih povezavah, nimajo nobenih ovir do končnega cilja. Tabela 4.2 prikazuje maksimalne, minimalne in povprečne čase v čakalnih vrstah za povezave, kjer je prišlo do zgostitve.

| časi paketov v čakalnih vrstah | | | |
|--------------------------------|-------------|---------------|--------------|
| Povezava | Minimum [s] | Povprečje [s] | Maksimum [s] |
| Obmocje1->Usmerjevalnik1 | 0.00007 | 49,0 | 95,08 |
| Obmocje2->Usmerjevalnik1 | 0,00341 | 212,6 | 431,17 |
| Obmocje3->Usmerjevalnik2 | 0,00201 | 76,2 | 142,46 |
| Obmocje4->Usmerjevalnik2 | 0.00041 | 60,7 | 125,38 |
| Obmocje5->Usmerjevalnik2 | 0,00111 | 32,0 | 66,09 |
| Usmerjevalnik1<-Internet | 0.00017 | 48,5 | 91,08 |
| Usmerjevalnik2<-Internet | 0.00005 | 10,6 | 23,72 |

Tabela 4.2: Časi v čakalnih vrstah na zgoščenih povezavah.

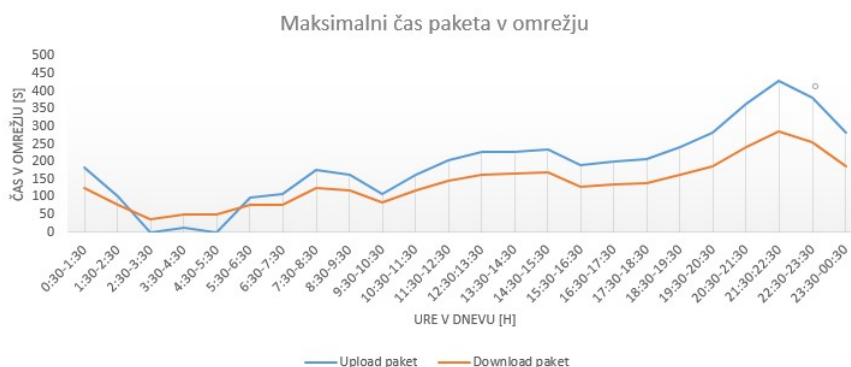
Poleg omenjenih zgostitev na povezavah smo opazili še dve dodatni ozki grli. Ti ozki grli sta v našem primeru posebnosti, saj pri drugih podobnih povezavah ne nastanejo. Natančneje si bomo ogledali povezavo od modula `Usmerjevalnik1` proti modulu `Obmocje2` in povezavo od modula `Usmerjevalnik2` proti modulu `Obmocje5`. Tabela 4.3 prikazuje povprečno čakalno dobo paketa v čakalni vrsti ter primerjavo tega časa s časom na ostalih podobnih povezavah. Kot opazimo, je čas čakanja pri povezavi `Usmerjevalnik1->Obmocje2` kar 5-krat večji, na povezavi `Usmerjevalnik2->Obmocje5` pa 2-krat večji od podobnih povezav. Moramo pa omeniti, da je na čakalnih vrstah za strežnik v modulih `obmocje` in `usmerjevalnik` čas čakanja enega paketa zanemarljiv.

| časi paketov v čakalnih vrstah | | |
|--------------------------------|---------------|--|
| Povezava | Povprečje [s] | Primerjava s podobnimi povezavami [Zgoščena : Ostale] |
| Usmerjevalnik1->Obmocje2 | 55,59 | 5:1 |
| Usmerjevalnik2->Obmocje5 | 38,07 | 3:1 |

Tabela 4.3: Časi v čakalnih vrstah na izstopajočih dveh povezavah.

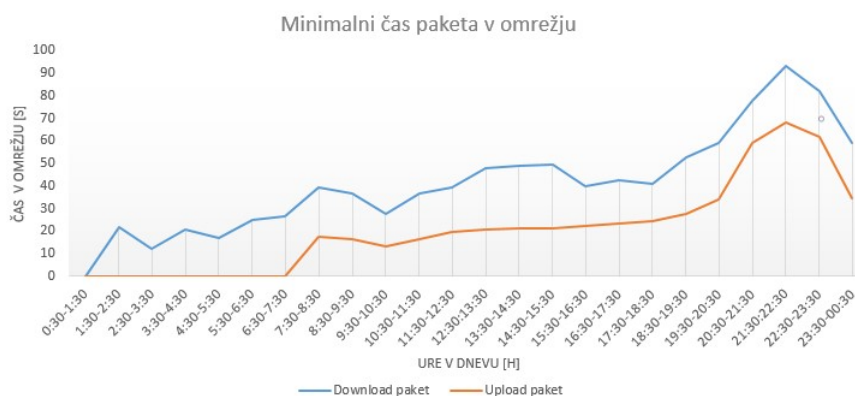
Glede na rezultate simulacij smo prišli do ugotovitve, da sta razloga za ozki grli večji pretok paketov in premalo zmogljive pasovne širine za število uporabnikov v njihovih naseljih. V primeru *Obmocja2* je skupno število uporabnikov kar dvakrat večje kot v ostalih območjih, kar doprinese do večjega prometa in posledično večjih čakalnih časov. V primeru *Obmocja5* smo opazili, da sta število uporabnikov in pasovna širina povezave primerljiva z *Obmocjem4*, vendar ima povezava do *Obmocja5* kar trikratni povprečni čas čakanja paketa v čakalnih vrstah. Zato sklepamo, da je razlog za tako razliko število naselij. *Obmocje5* ima samo štiri naselja, medtem ko jih ima *Obmocje4* osem. Zato se pri *Obmocju5* generira in sprejema več prometa na eno naselje v primerjavi z *Obmocjem4*.

Poleg časa bivanja paketov v čakalnih vrstah smo v simulacijah opazovali tudi čas, ki ga paket preživi v celotnem našem omrežju. Skozi simulacije opazimo podobno kot pri čakalnih časih vzpone in padce glede na obremenitev v določeni uri. Sliki 4.3 in 4.4 prikazujeta maksimalni in minimalni čas, ki ga paket generiran iz strani zunanjega sveta (angl. *download*) in paket generiran iz strani naselij (angl. *upload*) preživita v našem simulacijskem omrežju.



Slika 4.3 Maksimalni čas paketa v omrežju za upload in download.

Ob jutranjih urah med 1:00 in 5:00 uro zjutraj se čas prebivanja paketa v omrežju zniža. To je posledica manjše obremenitve na omrežju. Po peti uri zjutraj pa opazimo visok skok v časih paketov. To je posledica navad službenih uporabnikov, ki ob jutranjih urah na delovnem mestu začnejo uporabljati naše storitve. Ob večernih urah med 22:00 in 23:00 pa opazimo visoko uporabo naših storitev iz strani domačih uporabnikov, kar je sorazmerno z obremenitvijo na sliki 3.8. Tako visoki časi v omrežju so posledica ozkih grl v našem omrežju.



Slika 4.4 Minimalni čas paketa v omrežju za upload in download.

Če primerjamo sliki 4.3 in 4.4 opazimo, da se minimalni časi paketa v omrežju gibljejo podobno kot maksimalni časi glede na obremenitev. Poleg tega opazimo, da je minimalni čas paketa generiranega iz naselij proti zunanjemu svetu med urama 00:30 in 07:30 zelo nizek. Od 07:30 dalje pa opazimo visok narastek, kar je kot prej omenjeno zaradi službenih uporabnikov.

Na koncu smo še pogledali število izgubljenih, generiranih in sprejetih paketov v našem omrežju. Kot smo že prej omenili, je prišlo do več ozkih grl, kar posledično ustvari veliko količino izgubljenih paketov. Po statistikah, ki smo jih pridobili, je pri simulacijah prišlo do 1/3 izgube paketov zaradi ozkih grl v simulaciji, kar je za našega navideznega ponudnika internetnih storitev nesprejemljivo.

Glavni razlog za ozka grla v prvemu sklopu simulacij so prenizke pasovne širine med povezavami, zato bomo pasovne širine prilagodili zahtevam in jih zvišali, kjer je to potrebno, še posebej pa pri povezavah, kjer je prišlo do posebnosti. Zaradi izboljšav pričakujemo manjše čakalne čase in tudi manj izgubljenih paketov. Eksperiment opisan v pričujočem razdelku je le eden od treh, saj smo za točnost rezultatov sklop simulacij

ponovili še dvakrat z istimi vhodnimi in robnimi podatki ter pridobili podobne rezultate.

4.3 Rezultati drugega sklopa simulacij

Za začetek bomo opisali spremembe parametrov za drugi sklop simulacij. Tabela 4.4 prikazuje nove izboljšane pasovne širine za povezave v primerjavi s pasovnimi širinami v tabeli 4.1. Za takšne pasovne širine smo se odločili zato, ker nižje ne bi morale prenesti obremenitve uporabnikov med 21:00 in 00:00.

| Nove pasovne širine povezav | | |
|-----------------------------|-----------------|---------------|
| Modul | Download [Mbps] | Upload [Mbps] |
| obmocje1-usmerjevalnik1 | 13.680 | 5.790 |
| obmocje2-usmerjevalnik1 | 27.090 | 11.460 |
| usmerjevalnik1-internet | 40.770 | 17.250 |
| obmocje3-usmerjevalnik2 | 12.460 | 5.280 |
| obmocje4-usmerjevalnik2 | 7.950 | 3.370 |
| obmocje5-usmerjevalnik2 | 7.440 | 3.150 |
| usmerjevalnik2-internet | 27.850 | 11.790 |

Tabela 4.4: Ocenjene pasovne širine povezav.

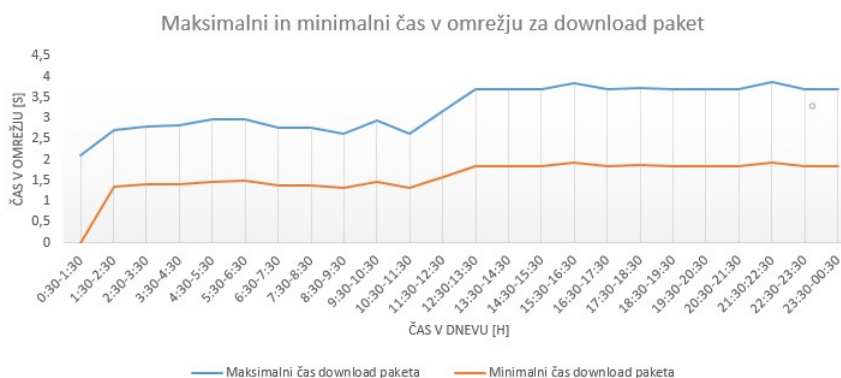
Kakor smo predvidevali, so se časi v čakalnih vrstah zmanjšali in tudi več paketov je prišlo do končnega cilja. Smo pa naleteli na nov problem in to je prevelik strežni čas na usmerjevalnikih. Število izgubljenih paketov v čakalnih vrstah v usmerjevalniku pred strežnikom je drastično narastlo. To je posledica večjega pretoka paketov, katerih strežnik ne more v dovolj hitrem času postreči. Povprečni čakalni čas pred strežnikom je v tabeli 4.5 označen z *Usmerjevalnik1* in *Usmerjevalnik2*.

| časi paketov v čakalnih vrstah | | | |
|--------------------------------|-------------|---------------|--------------|
| Povezava | Minimum [s] | Povprečje [s] | Maksimum [s] |
| Usmerjevalnik1 | 0,0000023 | 1,7 | 3,57 |
| Usmerjevalnik2 | 0,0000074 | 1,3 | 2,58 |

Tabela 4.5: Časi v čakalnih vrstah na usmerjevalnikih.

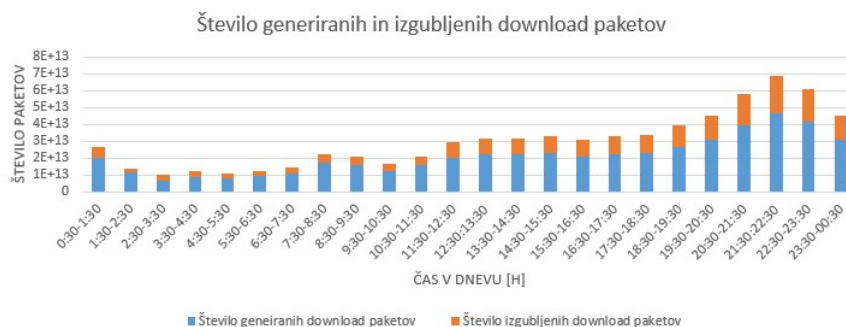
Čas v čakalnih vrstah je vplival tudi na minimalni in maksimalni čas paketa v omrežju.

Slika 4.5 nam pokaže maksimalni in minimalni čas paketov iz zunanjega sveta proti naselju. Kot opazimo sta maksimalna in minimalna časa zelo tesno povezana. Takrat, kadar pride do narastka v maksimalnem, pride tudi do narastka pri minimalnem. Kljub temu, da smo v primerjavi s prvim sklopom simulacij izboljšali čas, je še vedno prevelik za potrebe naših uporabnikov, saj ob veliki obremenitvi časi narastejo do 4 sekund v omrežju. Visok narastek minimalnega časa od 00:30 do 02:30 ure je posledica zgojitve na prej omenjenih usmerjevalnikih.



Slika 4.5 Maksimalni in minimalni čas paketa v omrežju za download paket.

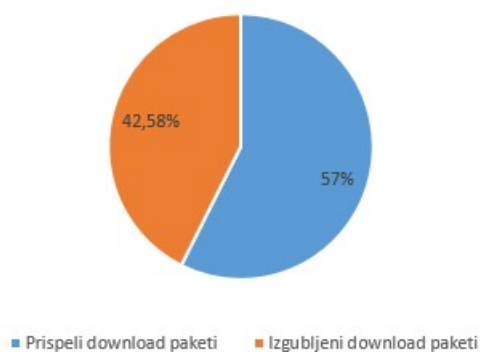
Čas paketa v omrežju iz naselij proti internetu je po drugi strani zelo konsistenten in ima povprečje 0,002 sekundi, kar je za našega navideznega ponudnika internetnih storitev zelo ugodno. Glede na primerjavo na sliki 4.6 med generiranimi paketi iz interneta in izgubljenimi paketi na čakalni vrsti pred strežnikom v modulih Usmerjevalnik1 in Usmerjevalnik2 opazimo, da se število generiranih in izgubljenih paketov sorazmerno večja z obremenitvijo našega simulacijskega omrežja, kjer pride do zelo visokega narastka v urah od 19:30 do 00:30, v jutranjih urah pa ponovno upade. Hkrati pa tudi opazimo kratek narastek v urah med 07:30 in 09:30, kar je posledica prometa službenih uporabnikov.



Slika 4.6 Število generiranih paketov z številom izgubljenih.

Na prvi pogled se zaradi številke zdi razmerje med generacijo in izgubo relativno dobro. Če pogledamo število generiranih in izgubljenih paketov skozi celoten dan natančneje (glej sliko 4.7) ugotovimo, da samo 57 procentov generiranih paketov pride od modula `Internet` do modulov tipa `NaseLje`. Tako smo ugotovili, da kljub izboljšanim pasovnim širinam pride do dveh novih ozkih grlih v usmerjevalnikih, zaradi katerih pride do še večjih izgub paketov, kakor v prvem sklopu simulacij. Tovrstni problem je bil pričakovan, saj smo z izboljšavami prišli do večjega pretoka paketov skozi omrežje, katere strežnik prepočasi obdeluje in pošilja naprej, kar povzroči slabo uporabniško izkušnjo našim uporabnikom in posledično izgube za našega navideznega ponudnika internetnih storitev.

Procent prispelih in izgubljenih paketov



Slika 4.7 Procent prispelih in izgubljenih download paketov.

Kot smo omenili, je glavni razlog za ozka grla v drugem sklopu simulacij previsok

strežni čas, vendar strežne čase težko izboljšamo, zato bomo za tretji sklop simulacij povečali število strežnikov na usmerjevalnikih 1 in 2, kjer pride do zgojitve. S povečanim številom pričakujemo še izboljšane čase v čakalnih vrstah in veliko manj izgubljenih paketov, saj si bodo delo strežniki razporedili. Tako se bo naša Kendallova notacija omenjena v poglavju 2 spremenila iz $D/D/1$ v $D/D/m$, kjer je m število paralelno vezanih strežnikov v naši strežni enoti. Prav tako kot pri prvemu sklopu simulacij smo tudi pri drugemu ponovili eksperiment še dvakrat, da smo preverili točnost rezultatov in ugotovili da so podobni skozi vse tri eksperimente.

4.4 Rezultati tretjega sklopa simulacij

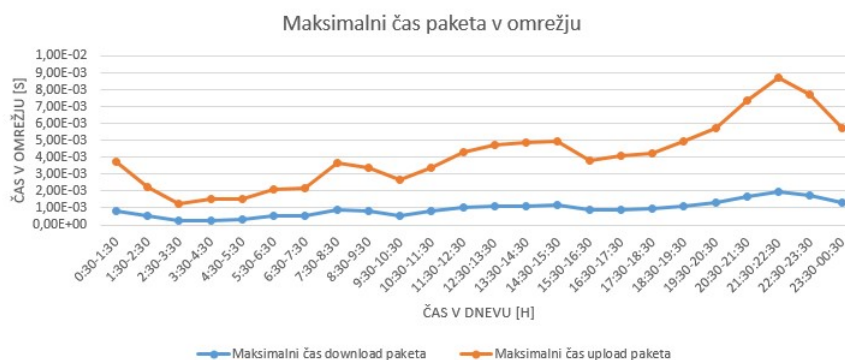
Kot smo naredili za drugi sklop simulacij, bomo tudi sedaj opisali izboljšave na omrežju. V tretjem sklopu simulacij bomo analizirali delovanje omrežja z izboljšanimi pasovnimi širinami (glej tabelo 4.4) in povečanim številom strežnikov na usmerjevalnikih. Najprej moramo omeniti, da je strežni čas vsakega strežnika približno $5 \mu s$ povzeto po viru [4]. Ker tega časa ne moremo spremeniti, nam je cenovno najugodnejše povečati njihovo število. Tako se bo Kendallova notacija spremenila iz $D/D/1$ v $D/D/6$, saj za optimalno delovanje strežne enote potrebujemo kar 5 dodatnih strežnikov. Z dodatkom novih strežnikov smo povečali intenzivnost obdelave paketov in posledično sprostili čakalne vrste.

Po končanem sklopu simulacij smo prišli do ugotovitve, da smo z izboljšavami prišli do rešitve, ki bi zadovoljila potrebe uporabnikov in navideznega ponudnika internetnih storitev. V tabeli 4.6 so predstavljeni maksimalni, minimalni in povprečni časi paketov v čakalnih vrstah na povezavah, ki so imele največ prometa. Kot opazimo, so časi dovolj nizki za normalno obratovanje.

| časi paketov v čakalnih vrstah | | | |
|--------------------------------|-------------|---------------|--------------|
| Povezava | Minimum [s] | Povprečje [s] | Maksimum [s] |
| Obmocje1->Usmerjevalnik1 | 0.000065 | 0,0025 | 0,0046 |
| Obmocje2->Usmerjevalnik1 | 0,000072 | 0,0029 | 0,0053 |
| Obmocje3->Usmerjevalnik2 | 0,000066 | 0,0028 | 0,0051 |
| Obmocje4->Usmerjevalnik2 | 0.000075 | 0,0024 | 0,0043 |
| Obmocje5->Usmerjevalnik2 | 0,000056 | 0,0025 | 0,0043 |

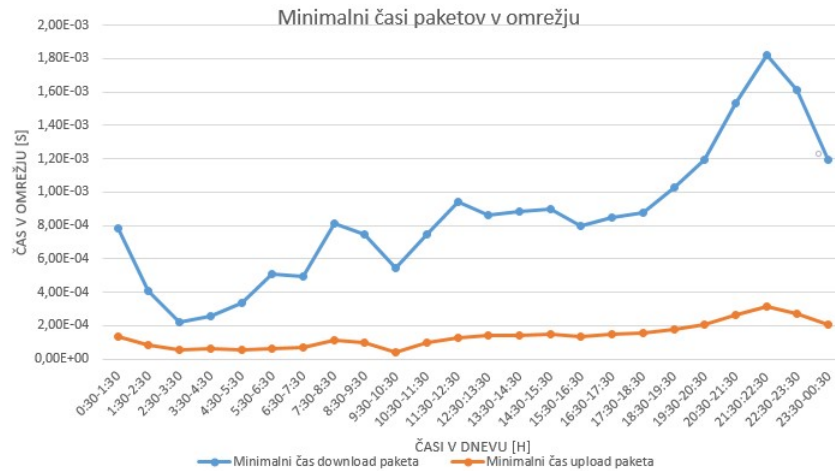
Tabela 4.6: Največji časi v čakalnih vrstah.

Poleg časov paketov v čakalnih vrstah smo opazili veliko izboljšavo časa, ki ga preživi paket v omrežju. Kot vidimo na sliki 4.8 je zanimivost to, da je čas, ki ga paket preživi v omrežju iz naselja proti internetu skozi celotni sklop simulacij večji kakor čas, ki ga paket preživi v omrežju iz interneta proti naseljem. Razlog za ta pojav je čas v čakalnih vrstah, ki smo ga opisali v prejšnji tabeli. Kljub temu se trend sorazmernosti z obremenitvijo ohranja in ponovno vidimo narastek med 20:30 in 22:30 ter med 07:30 in 08:30, ko se začnejo poslovne ure.



Slika 4.8 Maksimalni časi upload in download paketov v omrežju.

Če pogledamo minimalne čase na sliki 4.9 opazimo, da so minimalni časi paketov generiranih iz strani interneta proti naseljem večji, kakor paketi iz naselij proti internetu. Hkrati pa opazimo, da se minimalni in maksimalni časi med seboj ne razlikujejo za veliko, kar je za uporabniško izkušnjo zelo ugodno. Naraščanje in padanje pa je kakor pri prejšnjih grafih tudi sorazmerno z obremenitvijo v določeni uri.



Slika 4.9 Minimalni časi upload in download paketov v omrežju.

Na koncu še opazimo, da v tretjem sklopu simulacij ni nikjer v našem simulacijskem omrežju prišlo do izgube paketov. Hkrati pa moramo ponovno omeniti, da so opisani rezultati pridobljeni iz enega eksperimenta. Kljub temu, da smo eksperiment ponovili dvakrat z istimi vhodnimi in robnimi podatki, smo pri vseh treh eksperimentih pridobili podobne rezultate, kar potrjuje naše ugotovitve v razdelku.

4.5 Ugotovitve analize

Če povzamemo rezultate, do katerih smo prišli z analizo vseh treh sklopov simulacij pridemo do ugotovitve, da na kvaliteto omrežja vplivajo skoraj vsi elementi v njem, ki pa vplivajo tudi drug na drugega. Da zagotovimo optimalno delovanje fiksnega omrežja, moram vzpostaviti optimalne pasovne širine povezav med usmerjevalniki in območji, saj se po njih posreduje generiran promet iz interneta proti naselju in obratno. Hkrati pa moramo zagotoviti dovolj zmogljivo pasovno širino med usmerjevalniki in internetom, saj sta obe povezavi skozi vse tri sklope simulacij imeli največ prometa. V zakup pa moramo vzeti tudi naselja vezana na območja, saj smo ugotovili, da število naselij v enem območju pripomore k večjemu ali nižjemu prometu v primerjavi z drugimi območji s primerljivim številom prebivalstva in različnim številom naselij. Na koncu smo ugotovili, da če izboljšamo pretok paketov proti usmerjevalnikom, moramo povečati število strežnikov v usmerjevalnikih, saj v nasprotnem primeru ne morejo v dovolj hitrem času postreči vseh paketov in posledično pride do visokih čakalnih časov in izgub. Zadnji

sklop simulacij nam je pokazal, da je rezultat prilagoditev vseh elementov omrežja, ki bi vplivali na kvaliteto naših storitev, omrežje, ki zadovolji potrebe naših uporabnikov iz upravne enote Slovenj Gradec in potrebe našega navideznega ponudnika internetnih storitev.

5 Zaključek

V pričujočem diplomskem delu smo na osnovi postavljenega modela vzorčnega računalniškega omrežja izvedli nedeterministično generiranje paketov odvisno od ure dneva in potrebno optimizacijo za zagotovitev potreb uporabnika. Najprej smo vzpostavili po zgledu diplomskega dela Nives Bricman model fiksnega računalniškega omrežja v programskem okolju OMNeT++ in opisali vsak gradnik v njem. Poleg gradnikov smo s pomočjo *Poissonovega procesa* vpeljali tudi nedeterministični način generiranja paketov. Nato smo izpostavili pomembne robne pogoje in parametre kot so čas strežbe in zmogljivost pasovnih širin, s katerimi smo lahko opravili potrebne analize. V analizah smo si podrobneje ogledali čakalne čase v čakalnih vrstah, število generiranih in izgubljenih paketov in čas, ki ga en paket preživi od izvora do ponora. Izvedli smo tri sklope po 24 simulacij, kjer je vsaka simulacija zavzemala eno uro v dnevu in imela svojo obremenitev službenih in domačih uporabnikov določeno po zgledu diplomske naloge Nives Bricman. Prišli smo do ugotovitve, da na zmogljivost celotnega omrežja vplivajo elementi kot so velikost pasovnih širin na povezavah, število strežnikov v strežnih enotah in tudi količina naselij v določenem območju. Prav tako smo prišli do ključnih izboljšav, s katerimi smo

lahko zmogljivost omrežja izboljšali in posledično zagotovili boljšo prepustnost. Naše delo se od diplomske naloge Nives Bricman razlikuje v treh elementih. Ti so tehnična izvedba, koncept in eksperimenti. V tehnični izvedbi smo razširili generatorje v naselja in v njih vzpostavili nedeterministični način generiranja paketov, hkrati pa smo povečali število strežnikov v strežnih enotah in izboljšali zmogljivost pasovnih širin za potrebe uporabnikov. Koncept diplomske naloge Nives Bricman je bil osredotočen na vzpostavitvi fiksnega računalniškega omrežja za navideznega ponudnika internetnih storitev, medtem ko se naša diplomska naloga osredotoča na podrobno analizo in izboljšavo vzpostavljenega omrežja za potrebe uporabnikov. Zaradi razlike v konceptih in tehnični izvedbi se deli posledično razlikujeta tudi po opravljenih eksperimentih, saj smo z njimi skušali čimbolje približati delovanje modela fiksnega omrežja k dejanskemu delovanju realnih omrežij. Rezultat našega dela je podrobnejši vpogled v delovanje vzpostavljenega omrežja in optimizacija, ki skuša zadostiti potrebam uporabnikov.

Prav tako smo pri naši implementaciji vzorčnega računalniškega omrežja zanemarili usmerjevalne protokole, ki v realnosti vplivajo na delovanje omrežja. Primer usmerjevalnega protokola je OSPF (angl. *Open Shortest Path First*), ki ob zgostitvi na naslednjem usmerjevalniku pakete preusmeri na drug manj zgoščen usmerjevalnik. Protokol hrani usmerjevalno tabelo, ki pa se tekom simulacije večkrat posodobi z informacijskimi paketi o stanjih sosednjih vozlišč. Za optimalno delovanje z uporabo tega protokola bi morali model fiksnega vzorčnega računalniškega omrežja nadgraditi z večjim številom usmerjevalnikov in poti od izvora do ponora. Ker v našem primeru omrežja tega nimamo, bi posledično implementacija usmerjevalnega protokola bila nesmiselna ter ne bi vplivala na rezultate simulacij.

LITERATURA

- [1] Nives Bricman. Zasnova fizičnega omrežja za potrebe novega ponudnika internetnih storitev. Diplomaska naloga, Univerza v Ljubljani, 2018.
- [2] Chee Hock Ng. Queuing modelling fundamentals. John Wiley & Sons, New York, 1997.
- [3] Miha Mraz and Miha Moškon. *Modeliranje računalniških omrežij*. Založba FE in FRI, 2012.
- [4] Leopoldo Angrisani, Giorgio Ventre, Lorenzo Peluso, and Annarita Tedesco. Measurement of processing and queuing delays introduced by an open-source router in a single-hop network. *IEEE transactions on instrumentation and measurement*, 55(4):1065–1076, 2006.
- [5] Junhui Zhang, Jiqiang Tang, Xu Zhang, Wen Ouyang, and Dongbin Wang. A survey of network traffic generation. In *Third International Conference on Cyberspace Technology (CCT 2015)*. IET, 2015.
- [6] Omnet++ discrete event simulator - home. [Online; Dostopano 10.06.2018].
- [7] Omnet++ simulation manual, omnet++ version 5.4.1. [Online; Dostopano 15.06.2018].
- [8] Maria Kihl, Per Ödling, Christina Lagerstedt, and Andreas Aurelius. Traffic analysis and characterization of internet user behavior. In *2010 International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 224–231. IEEE, 2010.