Simuliranje jednoposlužiteljskih podvorbenih sustava

Prpić, Klaudija Marija

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:119:313688

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-18**

Repository / Repozitorij:

Faculty of Transport and Traffic Sciences - Institutional Repository

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

Klaudija Marija Prpić

SIMULIRANJE JEDNOPOSLUŽITELJSKIH PODVORBENIH SUSTAVA

DIPLOMSKI RAD

Zagreb, rujan 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI
POVJERENSTVO ZA DIPLOMSKI ISPIT

Zagreb, 2. travnja 2020.

Zavod: **Zavod za informacijsko komunikacijski promet**
Predmet: **Podvorbeni sustavi**

DIPLOMSKI ZADATAK br. 5912

Pristupnik: **Klaudija Marija Prpić (0135242291)**
Studij: Promet
Smjer: Informacijsko-komunikacijski promet

Zadatak: **Simuliranje jednoposlužiteljskih podvorbenih sustava**

Opis zadatka:

Dati detaljan matematički opis podvorbenih sustava. Analizirati značajke jednoposlužiteljskih podvorbenih sustava i prikazati modele kojima isti mogu biti opisani.
Izraditi simulator sustava posluživanja, u C# programskom jeziku, za potrebe analize performansi jednoposlužiteljskih sustava kao što su: prosječno vrijeme čekanja, prosječno vrijeme boravka u sustavu, prosječan broj korisnika u repu i prosječan broja korisnika u sustavu.
Usporediti performanse Poissonovskih i ne-Poissonovskih sustava posluživanja.

Mentor:

Š. Mrvelj

prof. dr. sc. Štefica Mrvelj

Predsjednik povjerenstva za diplomski ispit:

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

DIPLOMSKI RAD

SIMULIRANJE JEDNOPOSLUŽITELJSKIH PODVORBENIH SUSTAVA

SIMULATING A SINGLE-SERVER QUEUEING SYSTEM

Mentor: izv. prof. dr. sc. Štefica Mrvelj

Student: Klaudija Marija Prpić

JMBAG: 0135242291

Zagreb, rujan 2020.

SAŽETAK

U radu su prikazane značajke jednoposlužiteljskih podvorbenih sustava, što uključuje razdiobe međudolaznih vremena i vremena posluživanja te njihov matematički opis istih. S obzirom na razdiobe kojima se opisuje ulazni tok i proces posluživanja razlikuju se poissonovski i nepoissonovski modeli, a u radu su prezentirane značajke sljedećih modela: $M/M/1$, $M/G/1$ i $G/M/1$. Analiziran je utjecaj razdioba kojima se opisuju navedeni procesi na performanse sustava.

Nadalje, za potrebe rada kreiran je i simulator jednoposlužiteljskih modela pomoću C# programskog jezika. Također, prikazan je i kod napisan za potrebe simulatora.

Pomoću generiranih vrijednosti ulaznih veličina analizirani su i uspoređeni rezultati pojedine simulacije poissonovskih i nepoissonovskih podvorbenih modela.

KLJUČNE RIJEČI: podvorbeni sustavi; simulacija; C# razvojni jezik; simulator; jednoposlužiteljski podvorbeni modeli

SUMMARY

The paper presents the characteristics of single-server queueing systems, which include distributions of inter-arrival and serving times as well as their mathematical descriptions. Depending on the distributions with which the inputs and the serving processes are described, Poisson and the non-Poisson models are differentiated. The paper presents the characteristics of the following models: $M/M/1$, $M/G/1$ and $G/M/1$. The impact of the distributions with which the before mentioned processes are described, are analyzed.

Furthermore, a single-server queueing systems simulator written in C# is created for the purposes of this paper.

With the help of the generated inputs, every simulated Poisson and non-Poisson queueing system is analyzed and compared.

KEY WORDS: queueing systems; simulation; C# programming language; simulator; single-server queueing models

SADRŽAJ

1. UVOD 1
2. OPIS RAZDIOBI I MATEMATIČKI OPIS PODVORBENOG SUSTAVA 3
   2.1 Općenito o podvorbenim sustavima 3
   2.2 Matematički opis podvorbenog sustava za potrebe simulacije 5
      2.2.1 Ulazne veličine 5
         2.2.1.1 Ulazni proces 6
         2.2.1.2 Proces posluživanja 7
      2.2.2 Izlazne veličine 7
      2.2.3 Prometna mjerila 9
   2.3 Opis ulazno - izlaznih procesa 11
      2.3.1 Slučajni proces 12
      2.3.2 Opis funkcija razdiobi 14
         2.3.2.1 Eksponencijalna razdioba 14
         2.3.2.2 Poissonova razdioba 15
         2.3.2.3 Normalna razdioba 16
3. JEDNOPOSLUŽITELJSKI PODVORBENI MODELI 18
   3.1 Modeliranje jednoposlužiteljskih podvorbenih sustava 18
   3.2 Poissonovski jednoposlužiteljski modeli 19
   3.3 Nepoissonovski jednoposlužiteljski modeli 20
      3.3.1 $\boldsymbol{M/G/1}$ 21
      3.3.2 $\boldsymbol{G/M/1}$ 22
      3.3.3 $\boldsymbol{G/G/1}$ 23
4. OPIS RAZVOJNE OKOLINE C# 25
5. PRIKAZ RADA SIMULATORA 29
   5.1. Opis kôda u simulacijama 29
   5.2. Prikaz rada simulatora 34
   5.3 Izlazne vrijednosti simulacije 38
      5.3.1 $\boldsymbol{M/M/1}$ simulacija 38

5.3.2 $\boldsymbol{M/G/1}$ simulacija 43
5.3.3. $\boldsymbol{G/M/1}$ simulacija 49
6. ANALIZA I USPOREDBA IZLAZNIH VELIČINA SIMULACIJE 55
7. ZAKLJUČAK 58
POPIS KRATICA 59
POPIS LITERATURE 60
POPIS KORIŠTENIH SLIKA 62
POPIS KORIŠTENIH GRAFIKONA 62
POPIS KORIŠTENIH TABLICA 63

1. UVOD

Prizori u svakodnevnom životu gdje je moguće primijetiti redove, čekanje za nekom vrstom usluge i pružanje istih veoma su bitan segment, pojava ili ustrojstvo nekog sustava. Pod pojmom sustav mogu se naći različite ustanove poput bolnice, banke, pošte pa čak i dijelovi mreže poput usmjernika u informacijsko – komunikacijskoj mreži.

Prateći ponašanje takvog sustava kroz daljnju analizu i koristeći simulacije modela takvog sustava, moguće je dati rješenja primjenjiva u realnom okruženju. Takva rješenja moguće je primijeniti poznavanjem teorije podvorbenih sustava.

Podvorbeni sustavi obilježeni su raznim parametrima koji utječu na sam proces posluživanja. Ograničenja vezana uz vremena čekanja, broj korisnika u redu ili sustavu i dr., utječu na izvedbu podvorbe, koja može biti realizirana na različite načine. Ti načini se odnose na organizaciju i broj poslužitelja, vremenski redoslijed posluživanja korisnika, odnosno podvorbenu stegu. Za opis različitih podvorbenih sustava razvijeni su modeli kojima se mogu odrediti performanse sustava.

Svrha ovoga rada je opisati značajke jednoposlužiteljskih podvorbenih sustava i modela kojima oni mogu biti opisani. kao i njihove distribucije, te dati kratak uvid u C# programski jezik.

Cilj rada je kreirati simulator jednoposlužiteljskih podvorbenih modela, prikazati izlazne veličine i analizirati iste kroz grafičke prikaze $M/M/1$, $M/G/1$ i $G/M/1$ modela.

Diplomski rad sastavljen je od sljedećih povezanih dijelova:

1. Uvod
2. Opis razdiobi i matematički opis podvorbenog sustava
3. Jednoposlužiteljski podvorbeni modeli
4. Opis razvojne okoline C#
5. Prikaz rada simulatora
6. Analiza i usporedba izlaznih veličina simulacije
7. Zaključak

Drugo poglavlje opisuje razdiobe i matematički opis podvorbenog sustava. Opisuju slučajni procesi, teoretski opis podvorbenog sustava kao i izlazne veličine istih.

Treće poglavlje pruža uvid u jednoposlužiteljske podvorbene sustava razliku između poissonovskih i nepoissonovskih podvorbenih sustava, njihove karakteristične izlazne veličine dobivene Pollaczek-Hinčinovim formulama te kako modelirati takve sustave.

Četvrto poglavlje, *Opis razvojne okoline C#*, opisuje navedeni programski jezik kao i njegove karakteristike glede funkcija, tipova varijabli, radnog okruženja i tipova klasa i objekata.

Petim poglavljem prikazan je rad simulatora izrađenog za potrebe diplomskog rada. Prikazani su isječci kôda napisanog za potrebe rada simulatora te objašnjenja pojedinih dijelova istog. Također, prikazano je sučelje simulatora te su generirane simulacije za pojedine jednoposlužiteljske modele.

Šesto poglavlje daje analizu i usporedbu izlaznih veličina simulacija generiranih u prethodnom poglavlju.

2. OPIS RAZDIOBI I MATEMATIČKI OPIS PODVORBENOG SUSTAVA

Postoje različiti oblici podvorbenih sustava koji primjere nalaze u svakodnevnim ljudskim aktivnostima od čekanja na posluživanje u banci do prijenosa paketa između elemenata u komunikacijskoj mreži. Potreba za posluživanjem ili podvorbom postoji od ranih početaka ljudske civilizacije u trgovačkoj razmjeni pa sve do početaka modernizacije procesa, potrebe za organizacijom i optimizacijom podvorbenih sustava današnjice. Različiti sustavi u sebi sadrže dijelove podvorbenih sustava poput komunikacijske mreže, poštanskog ureda, studentske referade, medicinskih ustanova i mnogih drugih.

Kako bi se jednostavnije razumio podvorebni sustav koriste se određeni matematički modeli za njegov opisi, kako bi se razumjeli kompleksni procesi koji se odvijaju u sustavu. Pomoću matematičkih opisa podvorbenog sustava mogu se lakše razumjeti ulazno-izlazne veličine u samom procesu dolazaka i odlazaka korisnika na podvorbu. Teorija podvorbenih sustava zasniva se na vjerojatnosnim zakonima i vjerojatnosti ispunjenja događaja u sustavu u trenutku t poput vjerojatnosti pojave određenog broja na kocki, [1].

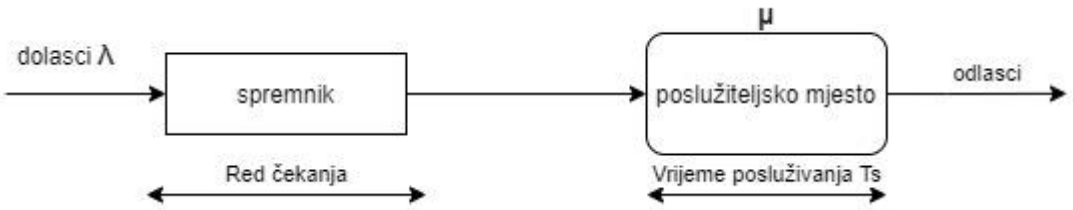
2.1 Općenito o podvorbenim sustavima

U realnom okruženju, svakodnevni događaji i kretanja ljudi odvijaju se nesmetano i slučajno. Takvi procesi zovu se stohastičnima jer su po svojoj prirodi nekontrolirani i slučajni. Upravo stohastički procesi prethode shvaćanju pojma podvorbenih sustava. Podvorbeni sustav cjelina je koja se sastoji od nekolicine elemenata, odnosno korisnika, repa, poslužitelja i pravila posluživanja ili podvorbe nadolazećih korisnika. Naime, podvorbeni sustav opisuje procese dolazaka korisnika i njihova posluživanja, dok se teorija dolazaka korisnika na posluživanje temelji upravo na njihovoj stohastičnoj prirodi. Primjeri primjene podvorbenog sustava u svakodnevnom životu mogu se pronaći u svakodnevnim aktivnostima. Takve aktivnosti ljudi i elementa podvorbeni su procesi koji se mogu usporediti s primjerima podvorbenih sustava u trgovini, banci, studentskoj referadi, medicinskoj ustanovi i mnogim drugima, a njihova problematika počiva na stohastičkim ulaznim i izlaznim procesima, [2].

Uvodno spomenuta problematika podvorbenih sustava, od stohastičkih pojava do procesa posluživanja i odlazaka korisnika iz podvorbenog sustava (PS) opisana je kroz teoriju prometa ili teoriju podvorbenih sustava. Problematika podvorbenih sustava opisuje se matematički vjerojatnosnim zakonima kako bi se omogućilo modeliranje istih. U poštanskom prometu, kada je riječ o PS mogu se razlučiti elementi poput korisnika, reda, poslužiteljskog mjesta dok su s druge strane u telekomunikacijskoj mreži ekvivalenti navedenom paket, sabirnica, usmjerivač. Time se, navedeni elementi iz različitih grana, uz teoriju podvorbenih sustava, mogu statistički proučavati

kroz modele i njihovu simulaciju. Simulacija i modeliranje podvorbenih sustava primjenjuje se kako bi se optimizirali ili planirali novi „slučajni“ procesi, na primjeru telekomunikacijskih mreža, planirali kapaciteti računalne opreme, infrastrukture i mreže, [1], [2].

Osnovni prikaz podvorbenog sustava s prethodno spomenutim elementima, kao i procesi koji se odvijaju u PS, prikazani su na slici 1.



Slika 1. Prikaz jednoposlužiteljskog podvorbenog sustava
Izvor: [3]

Korisnik je osoba, stvar ili paket u podvorbenom sustavu koji zahtijeva posluživanje. Kada se radi o stohastičkom procesu dolazaka korisnika u podvorbeni sustav, izvorište dolazaka može biti konačno ili beskonačno. Dakle, navedeno znači da velik broj korisnika može doći i zahtijevati na podvorbu te kako bi se matematički opisao ulazni proces kod beskonačnog izvorišta bitno je poznavanje međudolaznih vremena, a kod konačnog izvorišta potrebno je odrediti razdiobu međudolaznih vremena $k-1$ i korisnika k koji dolaze u PS. Skupina korisnika koji čekaju na posluživanje u podvorenom sustavu čine rep. Takva skupina predstavlja određena očekivanja nad podvorbenim sustavom, odnosno zahtjeve koje on mora biti spreman ispuniti. Značajka repa je podvorbena stega što bi predstavljalo skup pravila kako korisnici dolaze u rep, kako se ponašaju u repu i kako izlaze iz repa. Pa tako pravilo ulaska u rep određuje o kom sustavu je riječ s obzirom na to može li korisnik čekati u redu na posluživanje ili ne. Razlikuju se sljedeći sustavi:sustav s gubicima, čekanjem i odustajanjem od čekanja. Pravilo boravka u repu bazira se na varijantama čekanja, odustajanja od čekanja i formiranja jednog ili više repova. Izlazak iz repa zapravo predstavlja početak posluživanja kod poslužitelja i moguća su pravila pojedinačnog ili skupnog odlaska na posluživanje po nekom od poredaka. Poretci izlaženja iz repa su:

- „prvi došao, prvi poslužen“ (engl. *First Come First Served* – FCFS)
- „zadnji došao, prvi poslužen“ (engl. *Last Come First Served* – LCFS)
- „posluživanje nasumičnim poretkom“ (engl. *Service in Random Order* – SIRO)
- „najkraće vrijeme procesiranja“ (engl. *Shortest Processing Time* – SPT)
- „round robin“
- podvorba prema prioritetu (engl. *Service according to Priority* – PRI), [4].

Poslužitelj je dio poslužiteljskog mjesta podvorbenog sustava koji obavlja podvorbu, a može biti osoba ili stvar. Organizacija posluživanja razlikuje se po načinu rada tako da broj poslužitelja može biti od jedan do beskonačno, nepromjenjiv ili promjenjiv u vremenu, poslužitelji mogu biti poredani serijski ili paralelno te korisnici mogu imati različite razdiobe posluživanja. Posluživanje je vremenski odsječak tijekom kojeg korisnik boravi kod poslužitelja, a sam proces može imati razne oblike:

- konstantno vrijeme posluživanja
- vrijeme posluživanja ovisi o korisniku ili vrsti korisnika
- skupne podvorbe korisnika
- vrijeme posluživanja ovisi o dobu dana
- pristup korisnika poslužitelju upravlja se s pomoću brane ili vrata.

Osim elemenata koji ga čine, podvorbeni sustav opisiv je pomoću Kendall-ove oznake. Kendall-ovo zapisivanje uvedeno je 1953. godine i od tada predstavlja standard opisivanja glavnih obilježja podvorbenih sustava. Također, pomoću iste, omogućena je i klasifikacija podvorbenih modela koji se javljaju u podvorbenom sustavu. Formulacija Kendall-ove oznake je sljedeća: $A/B/C/K_q/K_i/P.S,$ [1], [5]. Slova u oznaci imaju značenje:

- A – razdioba međudolaznih vremena
- B – razdioba trajanja posluživanja
- C – broj poslužitelja
- K_q – kapacitet sustava
- K_i – kapacitet izvora
- $P.S$ – podvorbena stega.

2.2 Matematički opis podvorbenog sustava za potrebe simulacije

2.2.1 Ulazne veličine

Ulazni proces, proces posluživanja i podvorbena stega čine ulazne veličine, dovoljne kako bi se u potpunosti matematički opisao podvorbeni sustav, odnosno za potrebe rada, model jednoposlužiteljskog podvorbenog sustava.

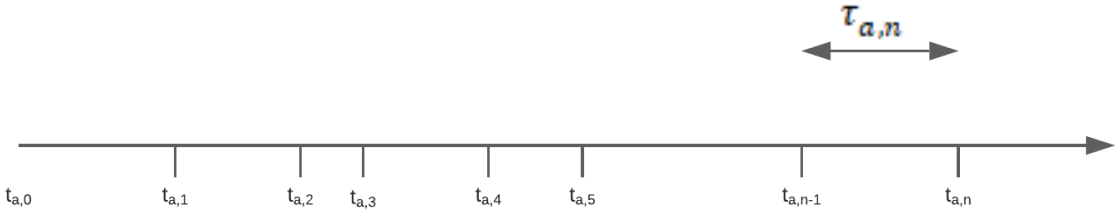
Karakteristike kojima se opisuju ulazni i izlazni procesi u podvorbenim modelima bazirani su na pojmovima iz teorije vjerojatnosti. Teorija vjerojatnosti spomenuta je u poglavlju 2.3, a odnosi se na teoriju podvorbenih sustava koja je zasnovana na teoriji vjerojatnosti. U podvorbenim sustavima ulazni procesi i procesi posluživanja slučajne su varijable koje se mogu opisati funkcijama razdiobe, [1].

S druge strane, izlazne veličine bitne su kao pokazatelji kvalitete i izvedbe samog podvorbenog sustava. Definiraju svrhu te koliko uspješno podvorbeni sustav obavlja svoje funkcije. Njihova uloga u simulaciji jednoposlužiteljskih modela biti će spomenuta u poglavlju 3.1.1.

Prometna mjerila povezuju ulazne i izlazne veličine, a u podvorbenom sustavu ona su definirana prometnim veličinama. Izlazni procesi bitni su za određivanje kvalitete usluge zajedno sa prometnim mjerilima te će biti prikazani detaljnije u poglavlju 2.2.3.

2.2.1.1 Ulazni proces

Ulazni proces čine dolasci korisnika u podvorbeni sustav. Korisnici mogu pristizati pojedinačno ili u skupinama u memoriju usmjerivača. Za potrebe simulacije, pošto se radi o jednoposlužiteljskom modelu, dolasci korisnika u podvorbeni sustav promatrat će se pojedinačno. Korisnici u podvorbeni sustav pristižu u trenucima: $t_{a,0};\ t_{a,1}; \dots t_{a,n}; \dots$, a vremenski razmaci između dolazaka korisnika u podvorbeni sustav τ_a nazivaju se međudolaznim vremenima: $\tau_{a,n} \equiv \tau_{a,n} - \tau_{a,n-1}; n = 1,2,..;\ \tau_{a,0} \equiv 0$. Prikaz ulaznog procesa prikazan je na slici 2.



Slika 2. Prikaz ulaznog procesa, [1]

Za potrebe simulacije pretpostaviti će se da dolasci korisnika u PS imaju međudolazna vremena τ_a koja imaju istu funkciju razdiobe $A(t)$ i funkciju gustoće $a(t)$ te za njih vrijedi izraz:

$$A(t) \triangleq [\tau_a \leq t]; A(t) = 0\ za\ t\ \leq 0; a(t) = \frac{dA(t)}{dt}; T_a = \frac{1}{\lambda} = \int_0^\infty t \cdot a(t)dt \quad (1)$$

Prosječno međudolazno vrijeme korisnika T_a u PS izrazom je jednako recipročnoj vrijednosti prosječne brzine dolazaka korisnika u sustav λ, [5].

2.2.1.2 Proces posluživanja

Proces posluživanja svodi se na vrijeme koje korisnik provodi kod poslužitelja. Vrijeme koje navedeni korisnik provede kod poslužitelja na posluživanju naziva se vrijeme ili trajanje posluživanja τ_s. Vrijeme posluživanja karakteriziraju sljedeći izrazi:

$$\tau_{s,1} = \tau_{s,2} = \tau_{s,n} = \cdots = \tau_s \Rightarrow SV\ \tau_s = „trajanje\ posluživanja" \quad (2)$$

$$B(t) = [\tau_s \leq t]; t \leq 0; b(t) = \frac{dB(t)}{dt};\ T_s = \frac{1}{\mu} = \int_0^\infty t \cdot b(t)dt \quad (3)$$

Bitan segment vremena posluživanja je matematičko očekivanje T_s, navedeno u prethodnom izrazu za funkciju razdiobe vremena posluživanja $B(t)$ i funkciju gustoće vjerojatnosti vremena posluživanja $b(t)$. Ono predstavlja prosječnu vrijednost koju korisnik provede kod poslužitelja na posluživanju. Izražava se kao obrnuto proporcionalna vrijednost brzini kojom poslužitelj poslužuje iste μ, odnosno kao integral funkcije gustoće vjerojatnosti $b(t)$ i vremena t.

Uz proces posluživanja, podvorbena stega bitna je kako bi se podvorbeni sustav u potpunosti definirao, a označava način na koji korisnik izlazi s posluživanja. Razlikuju se podvorbene stege koje su spomenute u poglavlju 2.1, [1].

2.2.2 Izlazne veličine

Pokazatelji uspješnosti podvorbenog modela mogu se prikazati kvantitativno pomoću izlaznih veličina. Prema definiciji, izlazne veličine „su mjerila izvedbe podvorbenog sustava". Orijentirane su prema značajkama kvalitete i same izvedbe, odnosno koliko uspješno dani podvorbeni model ostvaruje svoju svrhu, [1].

Izlazne veličine proizlaze iz Littleovog zakona, odnosno formule. Littleova formula korisna je jer povezuje količinske (K) i vremenske (T) značajke PS. Littleova formula glasi: $L = \lambda T$ te iz nje proizlaze formule koje opisuju rep, poslužiteljsko mjesto i cijeli PS. Karakteristične oznake koje vrijede kod izlaznih veličina, a obuhvaćene su Littleovom formulom su sljedeće:

- λ je prosječna vrijednost brzine ulazaka korisnika u PS
- T je prosječna vrijednost duljine boravka u repu, kod poslužitelja ili u PS
- L je prosječna vrijednost boravka korisnika u repu, kod poslužitelja ili u PS.

Za potrebe modeliranja sustava i simulacije nad istim, izlazne veličine dobar su pokazatelj kvalitete sustava s obzirom na odabrani jednoposlužiteljski model. Najčešće korištene izlazne veličine, zajedno sa njihovim značenjem, koje vrijede neovisno o kojem se podvorbenom modelu radi, prikazane su u tablici 1.

Tablica 1. Prikaz izlaznih veličina

Izlazna veličina	**Značenje**	**Formula**
Promet – a	Promet je jednak prosječnom intenzitetu dolazaka korisnika pomnoženim sa prosječnim vremenom posluživanja. Za vrijeme posluživanja vrijedi recipročan izraz $T_s = \frac{1}{\mu}$, stoga se promet može zapisati kao omjer intenziteta dolazaka i intenziteta posluživanja korisnika. Također, vrijedi izraz za promet jednak omjeru sume vremena posluživanja sa sumom međudolaznih vremena.	$a = \lambda \cdot T_s = \frac{\lambda}{\mu}$ $a = \frac{\sum_{i=1}^{N} t_{s_i}}{\sum_{i=1}^{N} t_{a_i}}$
Prometno opterećenje ili jakost prometa – ρ	Prometno opterećenje je jednako prometu podijeljenim sa brojem poslužitelja	$\rho = \frac{a}{c}$

Prosječan broj korisnika u sustavu - L_q	Prosječan broj korisnika u sustavu jednak je sumi prosječnog broja korisnika u redu – L_w i prosječnog broja korisnika koji su kod poslužitelja – L_s.	$L_q = L_w + L_s$
Prosječno vrijeme boravka u sustavu – T_q	Prosječno vrijeme boravka u sustavu, poznato kao vrijeme odaziva, jednako je sumi prosječne vrijednosti čekanja – T_w i prosječne vrijednosti trajanja posluživanja - T_s. Također, T_q je jednak omjeru prosječnog broja korisnika u sustavu i brzine dolazaka korisnika.	$T_q = T_w + T_s$ $T_q = \frac{L_q}{\lambda}$
Prosječno vrijeme čekanja - T_w	Prosječno vrijeme čekanja jednako je omjeru prosječne duljine reda i brzine dolazaka korisnika – λ.	$T_w = \frac{L_w}{\lambda}$

Izvor: [1]

2.2.3 Prometna mjerila

Izvedba podvorbenog sustava očituje se kroz prometna mjerila. Karakteristika PS ostvarena je između ulaznih i izlaznih veličina. Shema, prikazana na slici 1, prikazuje rad podvorbenog sustava čija je izvedba mjerljiva pomoću sljedećih veličina:

- a – ponuđeni promet
- ρ – prometno opterećenje
- a' – obavljeni promet
- ρ' – iskoristivost poslužitelja
- Q_n – razina posla
- L_q- prosječni broj korisnika u sustavu
- L_w- prosječan broj korisnika u repu

Veličine objašnjene u poglavlju 2.2.2 najčešće se koriste kao prometna mjerila kada se radi o izvedbi sustava izraženoj brojem korisnika i vremenom čekanja korisnika u podvorbenom

sustavu ili redu. Navedene su veličine dobar pokazatelj kvalitete sustava, odnosno kvantitativno izražavaju rad sustava, a izvedene su iz Littleove formule koja je detaljnije opisana u prethodnom potpoglavlju. Od bitnijih prometnih mjerila smatraju se ponuđeni promet, prometno opterećenje, obavljeni promet i iskoristivost poslužitelja, [1], [5].

Pod pojmom ponuđeni promet a se, prema [1], definira da „ponuđeni promet je jednak prosječnom broju korisnika ponuđenih podvorbenom sustavu λ tijekom razdoblja koje je jednako prosječnoj vrijednosti trajanja posluživanja jednog korisnika T_s." Potrebno je naglasiti da se pod ponuđenim korisnicima smatraju oni koji dolaze pred podvorbeni sustav. Izraz kojim se prikazuje ponuđeni promet je:

$$a = \frac{\lambda}{\mu} = \lambda \cdot T_s = \frac{T_s}{T_a} \tag{4}$$

Nadalje, prometno mjerilo prometnog opterećenja ρ se definira kao „jakost prometa je ponuđeni promet po jednom poslužitelju". S druge strane, ako poslužiteljsko mjesto ima c paralelnih poslužitelja koji rade brzinom μ onda je jakost prometa:

$$\rho = \frac{a}{c} = \frac{\lambda}{c \cdot \mu} \tag{5}$$

Potrebno je naglasiti da ako poslužiteljsko mjesto ima c poslužitelja, svaki može primiti prosječno λ/c broj korisnika, što dovodi do toga da sustav može odraditi najveći mogući promet od c *Erlanga*, jer je maksimalni promet koji može obaviti jedan poslužitelj 1 *Erlang*. Iz razloga što jakost prometa ne može biti veća od jedinica onda se izraz za jakost prometa piše kao:

$$\rho = min\left\{\frac{\lambda}{c \cdot \mu}; 1\right\} \tag{6}$$

U smislu obavljenog prometa a' dolazi se do definicije, prema [1], da je „obavljeni promet dio ponuđenog koji se ne gubi na ulazu u podvorbeni sustav, nego prolazi kroz njega." Obavljeni promet je mjera broja korisnika koji su posluženi u podvorbenom sustavu. Obavljeni promet jednak je ponuđenome ako sustav omogućuje stvaranje neograničenog repa, a korisnici ne napuste sustav prije nego budu posluženi. Definira se vjerojatnost $B(a)$ koja označava vjerojatnost da će nadolazeći korisnik pri određenoj veličini ponuđenog prometoma biti spriječen pri ulasku te neće biti poslužen. To se još naziva i vjerojatnost blokiranja, te se onda obavljeni promet može prikazati kao:

$$a' = a[1 - B(a)] \tag{7}$$

Kao jedno od prometnih mjerila je i iskoristivost poslužitelja ρ' koje se definira kao „obavljeni promet na jednoga poslužitelja u stacionarnom stanju sustava, odnosno to je dio ukupnog vremena promatranja unutar kojega je poslužitelj zaposlen. Za c paralelnih poslužitelja iskoristivost je:

$$\rho' = \frac{a'}{c} \tag{8}$$

U slučaju podvorbenog sustava bez gubitaka, odnosno da niti jedan korisnik ne može biti odbijen i ako svi poslužitelji imaju istu brzinu posluživanja, onda je iskoristivost poslužitelja jednaka jakosti prometa, [1], [5].

2.3 Opis ulazno - izlaznih procesa

Pojam koji se veže za ulazne i izlazne procese u podovrbenom sustavu je već spomenuta teorija podvorbenih sustava koja govori o slučajnom dolasku korisnika u podvorbeni sustav na posluživanje. Za opisivanje teorije podvorbenih procesa nužno je opisati vjerojatnosne zakonitosti koje vrijede kada je riječ o problematici ulaznih i izlaznih procesa i razdiobama kojima se isti opisuju. Također, kada je riječ o razdiobama vrijede i mjerila statističkih varijabli pomoću kojih se ulazni i izlazni procesi analiziraju.

Za dolaske korisnika, čekanje u redu i posluživanje istih, vrijedi pojam slučajnog procesa (SP) i slučajnih varijabli (SV). SV mogu biti diskretne i kontinuirane (neprekinute), prema tome slučajne procese definira pojava slučajnih varijabli, opisivih u skupu elementarnih događaja pomoću funkcija razdiobe. Skup elementarnih događaja može biti konačan i prebrojiv, beskonačan i neprebrojiv kao i beskonačan i prebrojiv. Diskretne SV opisuju se funkcijama vjerojatnosti i kumulativnim funkcijama vjerojatnosti u slučaju kada im je zadani skup $R(X) = X(\Omega)$ konačan ili prebrojivo beskonačan. Kada je skup kontinuiran i beskonačan, ne može se definirati funkcija razdiobe vjerojatnosti nad kontinuiranim SV dok se kumulativna funkcija razdiobe može, i kada je riječ o kontinuiranim i diskretnim SV. Kontinuirane SV opisuju se funkcijama gustoće vjerojatnosti i već navedenom kumulativnom funkcijom razdiobe u slučaju kada im je skup $R(X)$ neprebrojiv. Primjer za kontinuiranu SV, kada je riječ o procesu posluživanja, je vrijeme boravka korisnika kod poslužitelja, [6].

Određivanjem skupa slučajnih varijabli dobivene su moguće funkcije koje skup istih određuje. Sukladno tome, mjerila statističkih varijabli brojevi su koji se nazivaju momenti od kojih najvažniji, koje je moguće izdvojiti, su sljedeći:

- matematičko očekivanje
- varijanca
- standardna devijacija
- aritmetička sredina
- medijan

- i dr, [1].

2.3.1 Slučajni proces

Definicija slučajnog procesa je sljedeća: „SP je vremenski ovisna pojava koja se od trenutka do trenutka ponaša barem donekle nepredvidivo“. Također, jedna od definicija govori da je SP „zbirka slučajnih varijabli (SV) X koje su definirane na istom vjerojatnosnom prostoru i obilježene uz pomoć realnih brojeva“. Bitan segment na koji se odnose slučajni procesi ključni su za razumijevanje bitnih ulazno-izlaznih procesa i pripadnih razdiobi koje se koriste za opisivanje podvorbenih sustava. Kada se slučajni proces promatra kao vremensku pojavu dobiva se njegova vremenska ovisnost, a kada se isti promatra kroz matematički model dobiva se vremenska funkcija. Dobiveni rezultati slučajnog procesa kao matematičkog modela nekog vjerojatnosnog eksperimenta koji se mijenja u vremenu, su numerički. To znači da se svaka numerička vrijednost u nizu promatra kao slučajna varijabla čiji niz može biti konačan ili beskonačan. Modeliranje slučajnih procesa daje vjerojatnost pojave događaja, odnosno dobivena je učestalost ili frekvencija pojave istih. Primjer navedenog je frekvencija dolazaka paketa kojom će memorija usmjerivača postati puna, [1], [6].

Krajnja definicija slučajnog procesa glasi: „SP je zbirka SV koje su definirane na istom vjerojatnosnom prostoru i obilježene s pomoću realnih brojeva $N(t), t \in T$, gdje je T skup trenutaka na koje se odnosi SP, a tada je $X(t)$ iznos procesa u trenutku t. Trenutak t se naziva parametrom SP, a T skupom parametara SP. Parametar t je uvijek skalar, a T je skup skalara. Ako je T konačno ili prebrojivo beskonačno, naziva ga diskretnim. Inače T nazivamo kontinuiranim. Skup svih mogućih vrijednosti koje može imati $X(t)$ naziva se skupom stanja S. U tom smislu kao T, S može biti diskretan ili kontinuiran“. Ona govori o slučajnom procesu kao nepredvidivoj pojavi koja se odvija u vremenu. Rezultati tih nepredvidivih događaja u vremenu su diskretni ili kontinuirani:

$T = \{0, 1, 2, \dots\} \triangleq I$: diskretni T

$T = [0, \infty)$: kontinuirani, [6], [7].

Slučajni procesi klasificirani su prema prostoru ili skupu stanja S, prostoru ili skupu parametara T i međuovisnosti slučajnih varijabli $X(t)$. Također, postoji više različitih slučajnih procesa. Jedni od tih su Markovljevi slučajni procesi koje je prvi opisao ruski matematičar Markov, a koriste se u opisivanju podvorbenih modela. Markovljev proces je SP kod kojega razdiobe budućeg razvoja procesa ovise samo o sadašnjem stanju, a ne ovise o tome kako je proces došao u sadašnje stanje (neovisne su o tome kada se zadnji dolazak dogodio). Markovljev proces u potpunosti je određen ako:

- početna funkcija razdiobe vjerojatnosti: $P\left[X_{t_0} \leq x_0\right]$

- funkcija razdiobe prijelaznih vjerojatnosti ili funkcije vjerojatnosti prijelaza za sve $0 \leq t_m < t_n; m, n = 0,1,2, \ldots : P\left[X_{t_n} \leq x_n \middle| X_{t_m} \leq x_m\right]$, ako su funkcije razdiobe prijelaznih vjerojatnosti neovisne o t_n, a ovise samo o razlikama $t_n - t_m$, govori se o homogenim ili jednolikim razdiobama prijelaznih vjerojatnosti, [6].

Markovljevi procesi klasificiraju se na prema skupu stanja S i skupu parametara T od kojih oba mogu biti diskretni i kontinuirani. Prema tome, Markovljev lanac s diskretnim skupom parametara može (MLDV) imati diskretni skup stanja S i diskretni skup parametara T, Markovljev lanac s kontinuiranim skupom parametara (MLKV) ima diskretni skup stanja S i kontinuirani skup parametara T, Markovljev proces s diskretnim skupom parametara ima kontinuirani skup stanja S i diskretni skup parametara T te Markovljev proces s kontinuiranim skupom parametara ima kontinuirani skup stanja S i kontinuirani skup parametara T, [1], [8].

Drugi slučajni proces koji karakterizira ulazno-izlazne procese jednoposlužiteljskih modela je Poissonov SP. Kada je vrijeme kontinuirana SV i međudolazna vremena se ravnaju po eksponencijalnoj razdiobi riječ je o Poissonovom SP. Poissonov slučajni proces razmatra ulazni proces dolaska korisnika u sustav na dva načina:

- uz pomoć međudolaznih vremena
- s pomoću broja dolazaka unutar vremenskog odsječka duljine T_i.

Promatrajući dolaske korisnika unutar vremenskog odsječka promatra se funkcija $N(T_i)$ koja se naziva brojidbenom funkcijom, a broj dolazaka korisnika u vremenskom intervalu predstavlja broj događaja koji se dogodio do vremena t od početka promatranja ili ishodišta funkcije $t = 0$ i naziva se brojidbenim procesom $\{N(t), t \geq 0\}$. Definicija: „$\{N(t), t \geq 0\}$ gdje t označava razdoblje $(0, t]$ je brojidbeni proces ako su zadovoljeni sljedeći uvjeti:

- $N(0) = 0$
- $N(t)$ ima samo ne-negativne cjelobrojne iznose
- $S < t$ znači da je $N(s) < N(t)$
- $[N(t) - N(s)]$ je broj događaja koji se zbio nakon s, ali ne kasnije od t, tj. u razdoblju $(s, t]$, [1].

Poissonov SP može se smatrati brojidbenim procesom s brzinom odvijanja λ gdje vrijede određena pravila. Poissonov SP koristi se u opisivanju podvorbenih modela i modeliranju $M/M/1$ modela posluživanja, a njegova svojstva navode se u obliku poučaka:

1. poučak: neka je $N(t), t \geq 0$ Poissonov SP s brzinom odvijanja λ. Tada $SV\ Y$=“broj događaja u bilo kojem razdoblju duljine $\tau < 0$ “ ima Poissonovu razdiobu s parametrom

$\lambda\tau$:

$$P[Y = k] = \frac{(\lambda\tau)^k}{k!} e^{-\lambda\tau} \; ; \; k = 0,1,2, \ldots \tag{9}$$

2. poučak: Neka je $N(t), t \geq 0$ Poissonov SP s brzinom odvijanja λ i neka su $0 < t_1 < t_2 < \cdots < t_n - 1 < t_n < \ldots$ susjedni trenutci u kojima se pojavljuju događaji. Ako se međudolazna vremena definiraju na sljedeći način: $\tau_1 = t_1 - 0, \tau_2 = t_2 - t_2, \ldots, \tau_k = t_k - t_{k-1}, \ldots$, tada ovaj poučak izriče sljedeće: „Međudolazna vremena su međusobno neovisne identično raspodijeljene eksponencijalne slučajne varijable s prosječnom vrijednošću $\frac{1}{\lambda}$“

3. poučak: Ako je $N(t), t \geq 0$ brojidbeni proces kojemu su međudolazna vremena međusobno neovisne identično raspodijeljene eksponencijalne SV s prosječnom vrijednošću $\frac{1}{\lambda}$, tada je proces $\{N(t), t \geq 0\}$ Poissonov SP s brzinom odvijanja λ

4. poučak: Neka je $N(t); t \geq 0$ Poissonov SP i neka se u razdoblju $(0, t)$ pojavi samo jedan događaj. Ako se promatra $SV\ Y$ = „veličina (duljina) razdoblja do trenutka pojave (jednog) događaja“, Tada $SV\ Y$ ima kontinuiranu jednoliku razdiobu unutar razdoblja $\tau_1 = (0, t]$, [1], [6].

Također, još jedna od karakteristika Poissonovog SP je slučajni proces rađanja i umiranja $SP\ RIU$. Promatrajući slučajni proces kao ulazno-izlazni proces koji obuhvaća dolaske i odlaske korisnika kontinuirano u vremenu, prema teoriji podvorbenih sustava, koji se pojavljuju brzinom λ riječ je procesu rođenja kao pojavi koju se promatra s aspekta biologije, medicine ili astronomije, ali i u podvorbenim sustavima. Analogija procesu rođenja je proces dolazaka korisnika u sustav, a analogija procesu umiranje je proces odlazaka korisnika iz sustava. Navedeni procesi smatraju se proširenjem Poissonovog SP, ako se uvjetuje da procesi dolazaka i odlazaka korisnika ovise o stanju PS. Kada se istodobno promatraju navedeni procesi dolazaka i odlazaka, vjerojatnost pojave novog korisnika ili procesa rađanja unutar kratkog razdoblja jednaka je $\left(\mu_n h + 0(h)\right)$, a vjerojatnost odlaska korisnika ili procesa umiranja jednaka $\left(\mu_n h + 0(h)\right)$ gdje je n veličina populacije, a brzina rađanja i umiranja ovise o istoj, [1], [9].

2.3.2 Opis funkcija razdiobi

2.3.2.1 Eksponencijalna razdioba

Eksponencijalna razdioba s parametrom $\lambda > 0$, koji u PS označava prosječna vrijednost brzine ulazaka korisnika, koristi se kao funkcija gustoće vjerojatnosti $f(x)$ i funkcija razdiobe

$F(x)$. Primjeri gdje se javlja eksponencijalna razdioba su vrijeme do pojave prvog poziva, vrijeme između dva uzastopna poziva, vrijeme između dolaska dva uzastopna paketa u memoriju ili vrijeme do pojave prvog korisnika u sustavu i sl, [7], [10].

Eksponencijalna razdioba koristi se za neprekinutu slučajnu varijablu X s funkcijom gustoće vjerojatnosti $f(x)$ izraza:

$$f(x) = \begin{cases} 0, & x \leq 0 \\ \lambda e^{-\lambda x}, & x > 0 \end{cases} \tag{10}$$

Za funkciju razdiobe $F(x)$ kada je $x > 0$ vrijedi izraz:

$$F(x) = \int_{-\infty}^{x} f(t)dt = \int_{-\infty}^{0} f(t)dt + \int_{0}^{x} f(t)dt = 0 + \lambda \int_{0}^{x} e^{-\lambda t}dt = 1 - e^{-\lambda x} \tag{11}$$

te je stoga:

$$F(x) = \begin{cases} 0, & x \leq 0 \\ 1 - e^{-\lambda x}, & x > 0 \end{cases} \tag{12}$$

Očekivanje $E(X)$ za eksponencijalnu funkciju računa se parcijalnom integracijom prema formuli:

$$E(X) = \int_{-\infty}^{\infty} xf(x)dx = \lambda \int_{0}^{\infty} xe^{-\lambda x}dx = \ldots = \frac{1}{\lambda} \tag{13}$$

Očekivanje eksponencijalne funkcije obrnuto je proporcionalno parametru λ, odnosno prosječnoj vrijednosti brzine ulazaka korisnika u PS.

Varijanca $V(X)$ iznosi:

$$V(X) = \int_{-\infty}^{\infty} x^2 f(x)dx - (E(X))^2 = \lambda \int_{-\infty}^{\infty} x^2 \, e^{-\lambda x}dx - \frac{1}{\lambda^2} = \ldots = \frac{1}{\lambda^2} \tag{14}$$

Iznos varijance za eksponencijalnu funkciju moguće je odrediti integracijom razlike umnoška kvadrata varijable x^2 i funkcije distribucije vjerojatnosti $f(x)$ sa kvadratom očekivanja $(E(X))^2$. Ujedno, varijanca eksponencijalne funkcije obrnuto je proporcionalna kvadratu parametra λ.

2.3.2.2 Poissonova razdioba

Poissonova razdioba koristi se u graničnim slučajevima Binomne razdiobe kada je broj pokusa n dovoljno velik i vjerojatnost pojavljivanja p dovoljno mala. Očekivanje je tada $\lambda = np$, a idealne vrijednosti u kojima se koristi Poissonova razdioba su kada je $np \leq 10$, a $n > 50$.

Poissonovu razdiobu imaju slučajne varijable X čije vrijednosti bivaju jednake broju događaja u pojedinom vremenskom intervali. Primjerice, broj poziva, broj pristiglih paketa, broj korisnika u redu u nekom vremenskom intervalu, [1], [10].

Prema izvoru [1], diskretna slučajna varijabla X ima Poissonovu razdiobu s parametrom $\lambda > 0$ ako X poprima vrijednost iz skupa $0,1,2,$ ... s vjerojatnostima:

$$f(x) = P(X = x) = \frac{\lambda^x}{x!} e^{-\lambda}, x = 0,1,2,\ldots. \tag{15}$$

Funkcija distribucije $F(x)$ za Poissonovu razdiobu glasi:

$$F(x) = \sum_{x_i=0}^{x} \frac{\lambda^{x_i}}{x_i!} e^{-\lambda} \tag{16}$$

Za $X \sim \mathcal{P}(\lambda)$ matematičko očekivanje $E(X)$ i varijanca $V(x)$ imaju istu vrijednost:

$$\begin{aligned} E(X) &= \lambda \\ V(X) &= \lambda \end{aligned} \tag{17}$$

2.3.2.3 Normalna razdioba

Definicija normalne razdiobe glasi: za svaku neprekinutu slučajnu varijablu X s funkcijom gustoće vjerojatnosti $f(x)$:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, x \in \mathbb{R} \tag{18}$$

kaže se da ima normalnu ili Gaussovu razdiobu s parametrima μ, $\sigma \in \mathbb{R}$ i piše se $X \sim \mathcal{N}(\mu, \sigma^2)$. Parametar μ u PS označava prosječnu vrijednost brzine posluživanja korisnika. S druge strane, parametar σ označava standardno odstupanje ili devijaciju. Normalna razdioba je dobar model za većinu fizikalnih mjerenja te je dobar model za usporedbu simetričnosti i spljoštenosti ostalih razdioba. Pomoću normalne razdiobe izvodi se mnoštvo drugih distribucija. Karakteristična krivulja normalne razdiobe je takozvana „zvonasta“ krivulja, koja se također naziva i Gaussovom krivuljom, [7].

Za normalnu razdiobu funkcija distribucije $F(x)$ glasi:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}} dt \tag{19}$$

Izrazi za očekivanje $E(x)$ i varijancu $V(x)$ normalne razdiobe glase:

$$\begin{aligned} E(x) &= \mu \\ V(x) &= \sigma^2 \end{aligned} \tag{20}$$

Izraz očekivanja jednak je parametru μ, odnosno prosječnoj vrijednosti brzine posluživanja korisnika u PS. Varijanca nomalne razdiobe jednaka je kvadratu standardne devijacije σ^2.

3. JEDNOPOSLUŽITELJSKI PODVORBENI MODELI

3.1 Modeliranje jednoposlužiteljskih podvorbenih sustava

Sustav je kompleksna cjelina koju čini više segmenata. Iste segmente potrebno je sagledati s više gledišta. Modeliranjem istog potrebno je identificirati i obuhvatiti sve navedene segmente kako bi se model približio stvarnom sustavu. Prema definiciji, model je reprezentacija nekog sustava kojom se prikazuju relevantna svojstva, dok je modeliranje proces izrade istog. Prvi segment koji je opisan u ovom radu biti će modeliranje podvorbenog sustava s jednim poslužiteljskim mjestom kojeg sačinjava simuliranje slučajnih dolazaka korisnika u red na posluživanje, odnosno ulaznih veličina, kako bi se dobila odgovarajuća reprezentacija izlaznih veličina poput vremena boravka u sustavu, vremena posluživanja i dr. Korisnici u ovom slučaju mogu biti paketi, ćelije, okviri, fizičke osobe ili neki od elementa. U ovome radu analiza podvorbenih sustava načinjena je za podvorbene sustave karakteristične za komunikacijske mreže. Dolazni i odlazni korisnici su podatkovni paketi koje poslužuju usmjerivači komunikacijske mreže, [11], [12].

Prije svega, bitni koraci u izradi modela su:

- identifikacija modela – kod PS-a poznavanje sustava i identifikacija već spomenutih elemenata istog poput repa, poslužiteljskog mjesta, korisnika i poslužitelja
- formuliranje zadaće – modeliranje PS-a tako da se model promatra kao „crna kutija“ u koju dolaze i odlaze elementi, time ne postoji znanje o istom kao ni o procesima
- analiza zadaće – prikupljanje informacija poput slučajnog procesa dolazaka i odlazaka korisnika, koja su pravila ulaska i izlaska u rep, tko je poslužitelj i sl.
- gradnja modela – u PS-u je bitno identificirati modele. matematički ih opisati te ih izraditi prema istom matematičkom opisu
- rješavanje modela – rješavanje matematičkih formulacija modela na kojeg se nadovezuje postojeći sustav. Isto je definirano u poglavlju 2
- ispitivanje i provjera modela – usporedba realnog sustava s modelom koristeći statističke metode
- opis rješenja – opis izrađenog modela

- ostvarenje rješenja - cilj modeliranja je model sličan realnom primjeru sustava
- nadzor sustava – promatranje parametara modela u odnosu na realan sustav, odnosno postoje li promjene u mjerilima izvedbe, [8], [12].

Podvorbene modele karakteriziraju procesi stohastičke ili slučajne prirode. Samim time teže je modelirati problem koji uključuje slučajnu komponentu ulazno-izlaznih varijabli. Prema tome, bitno je razumijevanje pojma „slučajan“ koji se odnosi na slučajne procese dolazaka i odlazaka korisnika i elemenata u podvorbenom sustavu kako bi model bilo moguće simulirati. Rezultate simulacije modela moguće je analizirati i analitički prikazati kako bi se dobiveni rezultati mogli usporediti i analizirati s istovjetnim jednoposlužiteljskim modelima. Rezultati stohastičkog modela daju rezultate za distribuciju scenarija gdje postoji varijacija rezultata slučajne prirode. Procijenjeni skupovi izlaznih vrijednosti proučavaju se temeljem danih ulaznih vrijednosti pomoću kvantitativnih mjerila. Ovisno o metodi simulacije, ako je dovoljno velik skup ulaznih varijabli ispitan simulacijom moguće je determinirati izlazne vrijednosti i ostvariti željene rezultate, [1], [8].

S jedne strane cilj modela je prikazati vjerno značajke sustava koji se modelira, dok s druge strane vjernost prikaza u realnom okruženju dobiva se simulacijom istog. Simulacija je način prikazivanja stvarnih sustava ili procesa pomoću nekih određenih modela. Pošto se radi o podvorbenom sustavu, podvorbeni modeli koji se mogu koristiti unutar simulacije su modeli s općim i Poissonovim razdiobama međudolaznih vremena i vremena posluživanja. Kao što je spomenuto, ulazne veličine predstavljaju stohastički skup događaja dok mjerila izvedbe definiraju kvalitetu usluge pojedinog jednoposlužiteljskog modela, ovisno radi li se o ne-Poissonovim ili Poissonovim ulaznim parametrima.

3.2 Poissonovski jednoposlužiteljski modeli

Poissonovske jednoposlužiteljske prometne modele karakteriziraju prethodno opisani poissonovski slučajni procesi dolazaka i posluživanja korisnika. Prometne modele je moguće modelirati prema teoriji vjerojatnosti te vjerojatnosti pojave određenog broja korisnika u sustavu pri tome gledajaći da je slučajni proces vjerojatnosni eksperiment koji se mijenja u vremenu, na njega se ne može utjecati, a stvara numeričke vrijednosti korisne u samom modeliranju. Ako se na slučajni proces ne može utjecati, kao što je navedeno, značajke se svode na vjerojatnosni prostor i slučajne varijable $X(t)$ kao trenutak t, opisane numeričkim vrijednostima koje su skalarne, u ovom slučaju vrijeme, onda je skup trenutaka t skupina slučajnih procesa koji mogu biti:

- $T = \{0, 1, 2, \dots\} \triangleq I : diskretni\ T$
- $T = [0, \infty) : kontinuirani\ T.$

Slučajni procesi, detaljnije objašnjeni u 3.1, ključni za jednoposlužiteljske modele su Markovljevi, označeni simbolom „M“ prema Kendallovim oznakama. Isti su bitni za prikaz stanja jednoposlužiteljskog modela, odnosno slučajni procesi dolazaka i odlazaka korisnika i njihov razvoj u buduće stanje ne ovisi o prošlim stanjima, već samo o sadašnjem ili trenutnom stanju. To znači da prijelazi iz trenutnog u sljedeće, buduće stanje ne ovise o prošlim dolascima u sustav već samo o trenutnom stanju (dolasku ili odlasku korisnika iz sustava u sadašnjem trenutku). Također, slučajni procesi su brojidbeni procesi što znači da se „brojanje događaja“ unutar istog odvija do vremena t ili nekog trenutnog stanja. Poissonovski procesi dolazaka korisnika u sustav također se smatraju brojidbenim procesom kada se broj korisnika pristiglih u sustav kao skupa varijabli slučajnog procesa promatra do trenutnog stanja t, [1].

Tipični jednoposlužiteljski model navedenih karakteristika je $M/M/1/\infty/\infty/FCFS$. Simboli modela $M/M/1$, predstavljeni slovom „M“, odnose se redom na Poissonove dolaske i Poissonovo vrijeme posluživanje, a svedeni su na eksponencijalne razdiobe međudolaznih vremena i trajanja posluživanja. Također, model se može detaljnije objasniti pomoću Kendallove oznake na sljedeći način:

- M – predstavlja eksponencijalnu funkciju razdiobe međudolaznih vremena
- M – predstavlja eksponencijalnu funkciju razdiobe trajanja posluživanja
- C – Kendallovom oznakom predstavlja kapacitet, u ovom slučaju jednog poslužiteljskog mjesta $C = 1$
- K_q –znači da u sustavu ima mjesta za beskonačan broj korisnika, odnosno kapacitet podvorbenog sustava je beskonačan $K_q = \infty$. $M/M/1$ je model sustava bez gubitaka gdje će svaki korisnik koji dolazi pred podvorbeni sustav moći stati u red
- K_i – predstavlja beskonačno izvorište $K_i = \infty$
- $FCFS$ – predstavlja podvorbenu stegu ili pravilo izlaska iz repa poslužitelju pravilom prvi došao prvi poslužen, [1], [11].

Stanja $M/M/1$ opisuju se slučajnim procesom rađanja i umiranja. Dolasci korisnika u podvorbeni sustav alegorija su procesu „rađanja“ ili stanja dolaska, dok su odlasci korisnika nakon završetka posluživanja alegorija su procesu „umiranja“ ili stanja odlaska. Način prikaza procesa rađanja i umiranja odvija se dijagramom stanja i prijelaza kojim se prikazuje način statističkog mjerenja trenutnog broja korisnika u sustavu.

3.3 Nepoissonovski jednoposlužiteljski modeli

Nepoissonovski podvorbeni sustavi karakteristični su po svojstvima koja su različita, odnosno suprotna poissonovskim svojstvima podvorbenih modela. Zajedničko svojstvo modela $M/G/1$ i $G/M/1$, je svojstvo zaboravljivosti eksponencijalne razdiobe koje se rabi u rješavanju poissonovskih modela. Također, prijelazi između stanja ne mogu se opisati Markovljevim procesom jer ne ovise samo o sadašnjem, već i prethodnim stanjima.

Međutim, novo svojstvo navedenih modela je svojstvo obnavljanja trenutaka koje govori o trenucima neovisnim o prošlim stanjima već je riječ o trenucima u kojima se proces obnavlja ili regenerira te njegovo prošlo stanja nije potrebno za predviđanje budućeg. Model $M/G/1$ trenutke obnavljanja koristi za odlaske korisnika dok model $G/M/1$ za dolaske korisnika, [1].

3.3.1 $M/G/1$

Najjednostavniji nepoissonovski podvorbeni model je $M/G/1$. Dolasci korisnika u sustav su Markovljevi, a vrijeme posluživanja ima neku opću razdiobu. Navedeni model često se koristi u područjima telekomunikacija i računalstva. Njegova svojstva se mogu prikazati uz pomoć Kendallovih oznaka, a ona su sljedeća:

- M – označava da je proces dolazaka Poissonov slučajni proces s parametrom λ. Ako je proces dolazaka Poissonov to znači da su međudolazna vremena eksponencijalne slučajne varijable: $A(f) = 1 - e^{-\lambda t}$
- G – označava da je proces posluživanja neki opći s razdiobom bilo kojeg oblika, označeno simbolom G. Bitna svojstva za opisivanja razdiobe su matematičko očekivanje $E\{\tau s\} = T_s = \int_{-\infty}^{+\infty} t \cdot f_s(t)dt$, i varijanca $Var\{\tau_s\} = {\sigma_{\tau_s}}^2$
- 1 – na poslužiteljskom mjestu se nalazi jedan poslužitelj
- ostala svojsta: duljina repa nema ograničenja, izvor je beskonačan i pravilo izlaska iz repa je $FCFS$, [1].

Kod navedenog modela, za dobivanje izlaznih veličina koriste se Pollaczek-Hinčinove jednadžbe. Pollaczek-Hinčinove jednadžbe predstavljaju razdiobu posluživanja uz pomoć koeficijenta varijacije V_{τ_s} gdje je isti mjera „eksponencijalnosti“ modela.

$$V_{\tau_s} = \frac{\sigma_{\tau_s}}{T_s} \tag{21}$$

Prema prethodnoj jednadžbi, koeficijent varijacije V_{τ_s} računa se iz omjera varijance σ_{τ_s} i prosječnog vremena posluživanja T_s.

Prvi slučaj Pollazcek-Hinčinovih jednadžbi promatra „idealno slučajno posluživanje“ $M/M/1$ modela gdje su:

$$Var\{\tau_s\} = \frac{1}{\mu^2} \rightarrow \sigma_{\tau_s} = \frac{1}{\mu} = T_s \tag{22}$$

Varijanca $Var\{\tau_s\}$ obrnutno je proporcionalna kvadratu parametra μ koji u PS označava prosječnu vrijednost brzine posluživanja. Također drugi korijen od varijance daje standardno odstupanje σ_{τ_s}. Također, standardno odstupanje jednako je prosječnom iznosu vremena posluživanja korisnika u PS.

Prosječan broj korisnika u sustavu Lq jednak je izrazu:

$$\begin{aligned} L_q = E\{n_q\} &= a + \frac{\lambda^2 Var\{\tau_s\} + a^2}{2(1-a)} = a + \frac{a^2}{2(1-a)}\left[1 + \left(\frac{\sigma_{\tau_s}}{T_s}\right)^2\right] = \\ &= a + \frac{(E\{n_a\})^2 \cdot E\{\tau_s^2\}}{2(1-a)} \end{aligned} \tag{23}$$

Prosječno vrijeme zadržavanja T_q u sustavu se može dobiti iz sljedećeg izraza:

$$\tau_q = \tau_q + \tau_s \rightarrow T_q = T_w + T_s = T_w + \frac{1}{\mu} \tag{24}$$

Nadalje, prosječno vrijeme čekanja T_w dobiva se:

$$T_w = \frac{L_q}{\lambda} - \frac{1}{\mu} = \frac{\lambda^2 Var\{\tau_s\} + a^2}{2\lambda(1-a)} = \frac{aT_s}{2(1-a)}\left[1 + \left(\frac{\sigma_{\tau_s}}{T_s}\right)^2\right] = \frac{E\{n_a\} \cdot E\{\tau_s^2\}}{2(1-a)} \tag{25}$$

Prosječan broj korisnika u repu L_w prema Pollazek – Hinčinu dobiva se prema formuli:

$$L_w = L_q - a = \frac{a^2}{2(1-a)}[1 + V_{\tau_s}^2] \tag{26}$$

3.3.2 $G/M/1$

Kod podvorbenog modela $G/M/1$ trenutak obnavljanja je svaki trenutak dolazaka korisnika u podvorbeni sustav, te takve trenutke određuje trajanje posluživanja koji ima eksponencijalnu razdiobu.

$G/M/1$ model ima opću distribuciju međudolaznih vremena te eksponencijalno distribuirano vrijeme posluživanja. Sve izlazne veličine u $G/M/1$ se mogu dobiti iz $M/M/1$

modela, s time da se radi pretvorba $a \rightarrow r_0$, gdje je r_0 jedini korijen iznosa između nula i jedan jednadžbe:

$$z = A[\mu(1 - z)] \tag{27}$$

Gdje je $A(p)$ jednak:

$$A(p) = \mathcal{L}\{a(t)\} = \int_0^\infty e^{-pt} \cdot a(t)dt \tag{28}$$

Odnosno to je Laplaceov transformat funkcije gustoće vjerojatnosti međudolaznih vremena, [1], [13]. Time se dobije izraz za prosječan broj korisnika u sustavu L_q:

$$L_q = \frac{r_0}{1 - r_0} \tag{29}$$

Shodno tome, prosječan broj korisnika u repu L_w se dobije kao:

$$L_w = r_0 L_q \tag{30}$$

Prosječno vrijeme posluživanja T_s zadaje se kao:

$$T_s = \frac{1}{\mu} \tag{31}$$

Prosječno vrijeme zadržavanja u sustavu T_q zadaje se:

$$T_q = \frac{1}{\mu - \mu r_0} = \frac{\mu}{1 - r_0} = \frac{T_s}{1 - r_0} \tag{32}$$

Na koncu, prosječno vrijeme čekanja T_w:

$$T_w = \frac{r_0}{\mu - \mu r_0} = \frac{r_0}{1 - r_0} \cdot T_s = L_q \cdot T_s \tag{33}$$

3.3.3 $G/G/1$

$G/G/1$ model podvorbenog sustava je jednoposlužiteljski model gdje su međudolazna vremena prikazana nekom općom distribucijom i gdje su vremena posluživanja također opće distribuirana. $G/G/1$ je kompleksan model čija se stacionarna distribucija vremena može riješiti koristeći Wiener – Hopf metodom, ali su Kingman (1962.), Marshall (1968.) i Marchal (1978.) za stacionarno stanje dobili sljedeće:

$$\frac{\lambda^2 \cdot \sigma_s^2 + a(a-2)}{2\lambda(1-a)} \leq T_w \leq \frac{\lambda(\sigma_a^2 + \sigma_s^2)}{2(1-a)} \tag{34}$$

Vrijednosti pomoću kojih se dobiva prosječno vrijeme čekanja u sustavu T_w su sljedeće:

- σ_a^2 = varijanca međudolaznih vremena
- σ_s^2 = varijanca trajanja posluživanja
- a = ponuđeni promet.

Time je dobivena aproksimacija prosječnog vremena čekanja u sustavu $G/G/1$, [1], [14].

4. OPIS RAZVOJNE OKOLINE C#

Tehnologija današnjice kreće se u više smjerova, od unaprjeđenja mobilnih komunikacijskih mreža četvrte generacije ka petoj, razvoja novih, pametnijih uređaja, umjetne inteligencije i algoritama do stvaranja novih aplikacija. Premda, osim platformi za stvaranje aplikacija, one mogu i simulirati postojeće sustave. Programski jezik C# objektno je orijentirani programski jezik kojim je moguće elemente nekog realnog sustava gledati kao objekte te ih tako implementirati unutar simulacije. Na primjeru komunikacijske mreže koja sadrži elemente poslužitelja, na primjer usmjernika, repa odnosno memorije, paketa ili entiteti koji se poslužuju i šalju mrežom do odredišta, moguće je opisati već postojeći, realni sustav kroz model ili simulaciju pomoću spomenutih objekata.

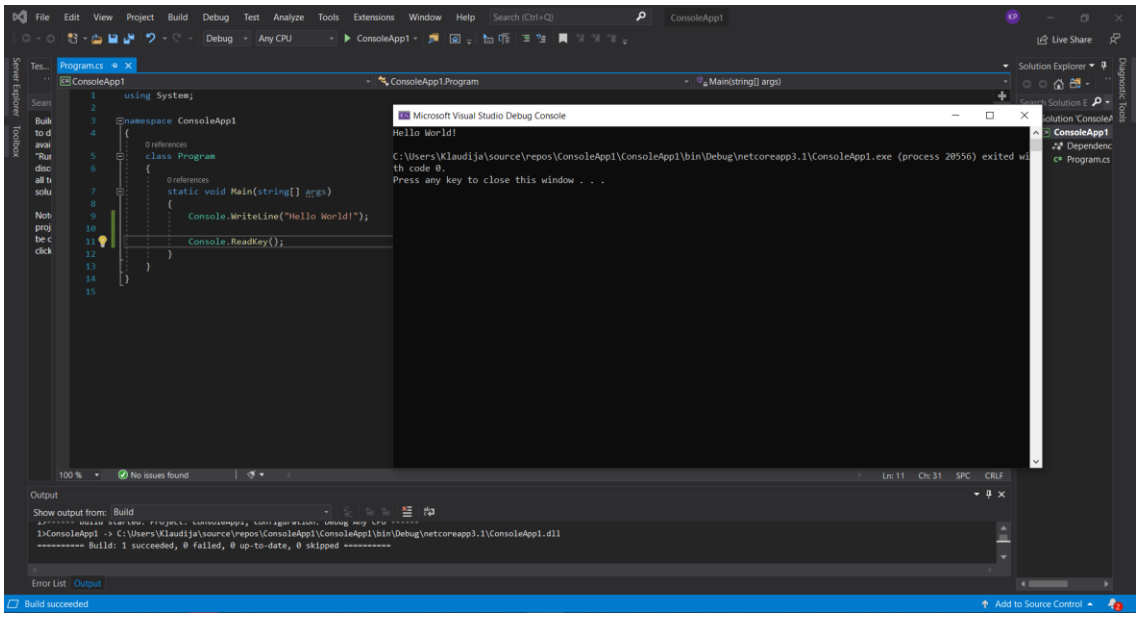
C# je programski jezik koji proizlazi iz skupine C programskih jezika kao što su C, C++, Java, JavaScript i dr. On radi na *.NET frameworku* za razvoj različitih aplikacija koje mogu biti od *Web* do *Windows* aplikacija. C# programski jezik počiva na integriranoj razvojnoj okolini (engl. *Integrated Development Environment* – IDE). Jedna od najpoznatijih, isto tako korištena za potrebe ovog rada je Visual Studio. Kao razvojna okolina popularna je zbog svojih mogućnosti uređivanja, dizajna, analize i mnogih drugih funkcionalnosti. Osim objektno orijentirane strukture, C# programski jezik posjeduje značajke koje ga čine jednostavnim za korištenje. Jedna od tih značajki je orijentiranost prema softverskim komponentama što znači da je pomoću C# programskog jezika moguće uvesti komponente drugog softvera u razvojnu okolinu programskog jezika. Sljedeća od značajki je automatsko upravljanje memorijom kako bi se oslobodili resursi memorije od objekata koji se više ne koriste. Kod pisanja programskog kôda u C# korisne su značajke detekcije pogrešaka pomoću „*try-catch*“ bloka i ispravljanje pogrešnog unosa varijabli. Uz dosad navedene, još jedna od prednosti primjene C# programskog jezika ostvarena je kroz značajku podrške prema starim i novim verzijama biblioteka koje se koriste u kôdu što znači kada se u verziji aplikacije koristi stara biblioteka, njenom nadogradnjom nadograđuje se i sama aplikacija u kojoj je ista primijenjena, [15], [16].

Kada se radi o izradi aplikacije, bitno je otvoriti novu skriptu i instalirati *.NET Core SDK* (engl. *Software Development Kit - SDK*) ili razvojni alat. U radnoj skripti je potrebno inicijalizirati varijablu i dodijeliti joj vrijednost. Najčešće se radi o tekstu jednostavnog programa ili ispisa u konzoli „Hello World“. Aplikacija se zatim izvozi u programu ekstenzije .cs. Radna skripta prati hijerarhijsku strukturu biblioteka i programa. Jedna od tih je sistemska biblioteka *System* koja sadrži klase poput *Console*, klasi u kojoj je učitan program, zatim *Collections*, *IO* i dr. Pojedinoj sistemskoj biblioteci pristupa se pomoću direktive *using* što znači da se u skripti mogu koristiti dani elementi iz pojedinog imenskog prostora, [16].

Uz pojedine značajke koje se koriste u C# programskom jeziku dolaze i elementi koje isti koristi. Jedni od spomenutih elemenata koji se koriste su objekti jer se radi o objektno orijentiranom programiranju, dok se ostali mogu podijeliti na:

- strukturu programskog jezika koja sadrži programe, imenski prostor, tipove
- varijable i tipove varijabli
- operatore koji polučuju vrijednost
- polja, izjave
- klase, objekte, atribute i metode
- sučelja, [16].

Prilikom pokretanja skripte u C#, navedeni elementi se mogu se vidjeti, odabrati i koristiti u izradi aplikacije ili, u ovom slučaju, simulacije. Detaljnije, izgled skripte započinje s metodom *Main.* U istoj metodi definirana je klasa u kojoj se mogu deklarirati varijable gdje se, na primjer, zapisuje već spomenuti jednostavni program „*Hello World*“ prikazanom na slici 3. Za ispis istog koristi se sistemska metoda *Console.WriteLine()* kojoj se pristupa poznatom direktivom *using*. Kôd se iz skripte izvozi u program ekstenzije *.cs* ili *.exe* pomoću kompajlera. Za dodjelu funkcionalnosti i svrhe kôdu, kako bi u konačnici isti postao aplikacijom, koriste se različite biblioteke. Primjenom biblioteka datoj aplikaciji mogu se dodati funkcionalnosti dizajna, funkcija, grafikona, povezivanja s drugim sučeljima i sl, odnosno nužne radnje kako bi se ostvarila krajnja funkcionalnost same aplikacije.



Slika 3. Prikaz Hello Word osnovnog programa

Primjena programskog jezika C# je široka zbog spomenute jednostavnosti i značajka koje olakšavaju sam proces razvoja do intuitivnosti i sličnosti s ostalim programskim jezicima iz skupine C jezika. Ekosustav razvoja aplikacija pomoću C# programskog jezika je velik. Omogućen je jednostavan razvoj više tipova aplikacija poput igara, Windows klijentskih aplikacija i servisa, poslovnih aplikacija, Cloud aplikacija, Web servisa i aplikacija, korporativnih aplikacija i usluga i mnogih drugih. Primjeri Windows aplikacija u Sype, Microsoft Office, PhotoShop, Paintbrush i sl. Web aplikacije razvijaju se na razvojnom okviru ASP.NET u kombinaciji s HTML ili JavaScript tehnologijom, ASP.NET MVC ako C# programski jezik treba povezati za SQL bazom podataka te ASP.NET server, [16].

Također, uzevši u obzir da je C# objektno orijentirani razvojni jezik, potrebno je naglasiti način na koji se objekti i klase kreiraju. S obzirom da se često pojam klasa i objekt koriste kao istoznačnice, to su zapravo dva potpuno drugačija termina. Klasa definira tip objekta, ali ona sama po sebi nije objekt. S druge strane, objekt je entitet koji se bazira na klasi, ali se definira kao instanca klase koja se kreira pomoću ključne riječi new, [17]. Stoga, instanciranje nekog objekta zvanog $automobil$ iz klase $Parkiraliste$ bi se odvio na način prikazan na slici 4.

C#

```
Parkiraliste automobil1 = new Parkiraliste();
```

Slika 4. Prikaz instanciranja objekta *automobil* iz klase *Parkiraliste*

Nadalje, bitna stavka koju C# posjeduje je nasljeđivanje klasa (engl. *Class inheritance*). U prijevodu, svaki puta kada se kreira neka nova klasa, ona je u mogućnosti naslijediti elemente neke druge klase koja dopušta nasljeđivanje elemenata poput atributa i metoda. Jedino nije moguće nasljeđivanje konstruktora. Način na koji se izvodi nasljeđivanje prikazan je na slici 5,

C#

```
1. public class tablicaUsmjeravanjaLite: tablicaUsmjeravanjaMaster
2. {
3. }
```

Slika 5. Primjer nasljeđivanja klasa u C#

Kao jedna od mogućih stavki, C# također ima opciju interakcije ili manipulacije s datotekama osobnog računala. Kako bi se omogućila navedena opcija, potrebno je dodati $System.IO$ u radnu okolinu koja sadrži klasu $File$ kojom se može utjecati na datoteke. Takav način manipulacijom datotekama je koristan u slučajevima kao što su čitanje seta podataka koji je iz vanjskih izvora, zapisivanje seta podataka za nekog vanjskog korisnika i mnogi drugi. Nadalje, bitno je napomenuti da se mogu koristiti relativne i apsolutne putanje prilikom zapisivanja i čitanja datoteka. Linije programskog kôda na slici 6 prikazuju generiranje pseudonasumičnih skupova brojeva, koji se zatim zapisuju u datoteku na lokalnom računalu, [18].

C#

```
1. using System;
2. using System.IO;
3. using System.Collections.Generic;
4.
5. namespace ConsoleApp1
6. {
```

```
7.      class Program
8.      {
9.          static void Main(string[] args)
10.         {
11.
12.             Random rand = new Random();
13.             var broj = 0;
14.
15.             TextWriter tw = new StreamWriter("brojevi.txt");
16.             List<int> numbs = new List<int>();
17.             for (int i = 0; i < 100; i++)
18.             {
19.                 broj = rand.Next(100);
20.                 numbs.Add(broj);
21.             }
22.             foreach (var br in numbs)
23.             {
24.                 tw.WriteLine(br);
25.
26.             }
27.             tw.Close();
28.         }
29.     }
30. }
```

Slika 6. Prikaz generiranja pseudonasumičnih skupova brojeva i zapisivanja u datoteku

C# programski jezik je programski jezik koji je intuitivan i jednostavan za započeti. Njegovi glavni atribut je razina upotrebe koja može varirati od početnika do već iskusnog razvojnog programera. Od jednostavnih konzolnih aplikacija pa sve do video igara, C# ima svrhu koja se svakim nadogradnjama sve više ostvaruje.

5. PRIKAZ RADA SIMULATORA

Simulator jednoposlužiteljskih modela izrađen je pomoću postupaka i tehnika opisanih u prethodnome poglavlju. Program je izrađen C# programskim jezikom, te *Windows Forms* grafičkog sučelja. Također, korištene su biblioteke za dodatne funkcionalnosti kao što su *Accord.Net* za matematičke funkcije, *ZedGraph* za kreiranje grafova i *BunifuUI* za grafičke elemente. Simulator se sastoji od tri posebna dijela, a to su $M/M/1$, $M/G/1$ i $G/M/1$ simulacije od kojih svaki ima svoja zasebna polja za unos, tablice za ulazne i izlazne veličine te grafove. Također, u simulatoru će se koristiti izraz paketi, a uz to će se vremenske jedinice izražavati u milisekundama.

5.1. Opis kôda u simulacijama

Modeli simulirani u ovom radu $M/M/1$, $M/G/1$ i $G/M/1$ gdje kod nepoissonovskih modela G označava normalnu razdiobu. U izradi samih jednoposlužiteljskih modela, ovisno radi li se o poissonovskim ili nepoissonovskim distribucijama bitno je prikazati stohastičnost ulaznih parametara. Problematiku ove vrste moguće je riješiti korištenjem svojevrsnog generatora slučajnog procesa, koji u razvoju ovog simulatora izveden preko *Accord.Net* biblioteke, odnosno prostora imena (engl. *Namespace*) *Accord.Statistics.Distributions.Univariate*. Unutar navedenog prostora imena nalaze se razne vrste distribucija, koje imaju svoje parametre pomoću kojih se iste instanciraju, [19]. Primjer instanciranja eksponencijalne distribucije za vremena dolazaka i posluživanja korisnika u $M/M/1$ modelu prikazan je na slici 7.

C#

```
1.  double sredVrijBTa = 1 / (Convert.ToDouble(textBox1.Text));
2.  double sredVrijBTs = 1 / (Convert.ToDouble(textBox2.Text));
3.
4.  ExponentialDistribution exp = new ExponentialDistribution(sredVrijBTa);
5.
6.  double[] uzorci = exp.Generate(Convert.ToInt32(txtUzorci.Text));
7.
8.  ExponentialDistribution expTs = new ExponentialDistribution(sredVrijBTs);
9.
10. double[] uzorciTs = expTs.Generate(Convert.ToInt32(txtUzorci.Text));
```

Slika 7. Programski kôd za instanciranje eksponencijalne distribucije vremena dolazaka i posluživanja korisnika u $M/M/1$ modelu

Varijable tipa double, *sredVrijBTa* i *sredVrijBTs* zapravo se odnose na parametar λ eksponencijalne distribucije koji je unutar simulacije izveden kao kvocijent broja 1 i unesene srednje vrijednosti. Navedene varijable su potrebne prilikom instanciranja objekata iz klase *ExponentialDistribution* jer se pomoću istih dodjeljuje parametar λ. Nakon što su distribucije instancirane, spremaju se u polja tipa double tako da se pomoću metode *Generate()* generira nasumičan skup vrijednosti odabrane distribucije. Kod modela koji koriste nepoissonovske

distribucije, a u ovom slučaju je to normalna distribucija, dodjeljuju se parametri srednje vrijednosti μ i standardnog odstupanja σ. Primjer generiranja nepoissonovskih ulaznih parametara prikazan je na slici 8.

C#

```
1.  double sredVrijBTa = 1 / (Convert.ToDouble(txtExpMean.Text));
2.
3.  ExponentialDistribution expTa = new ExponentialDistribution(sredVrijBTa);
4.
5.  double[] uzorci = expTa.Generate(Convert.ToInt32(txtUzorciMG1.Text));
6.
7.  NormalDistribution normTs = new NormalDistribution(Convert.ToDouble(txtNormMean
    .Text), Convert.ToDouble(txtNormStdDev.Text));
8.
9.  double[] uzorciTs = normTs.Generate(Convert.ToInt32(txtUzorciMG1.Text));
```

Slika 8. Linije kôda za generiranje nepoissonovskih ulaznih parametara

Kôd prikazan na slici 8 razlikuje se samo u pogledu instanciranje klase *NormalDistribution*, u koju se učitavaju vrijednosti μ i σ, dok ostala sintaksa ostaje nepromijenjena u odnosu na $M/M/1$.

Nadalje, potrebno je opisati način na koji se dobiju određene ulazne i izlazne vrijednosti. Ulazne vrijednosti su također zapisane u tablicu unutar simulatora. Vremena odlazaka i dolazaka dobiveni su linijama kôda opisanim na slici 9.

C#

```
1.  if (i == 0)
2.  {
3.      tdol = uzorci[0];
4.      todl = uzorci[0] + uzorciTs[0];
5.
6.  }
7.
8.  else
9.  {
10.     tdol = tdol + uzorci[i];
11.
12.     if (todl > tdol)
13.     {
14.         todl = todl + uzorciTs[i];
15.     }
16.     else
17.         todl = tdol + uzorciTs[i];
18. }
```

Slika 9. Linije kôda za izračun vremena dolazaka i odlazaka

Unutar petlje koja je dužine ulaznih uzoraka, postavljeni su uvjeti izvršavanja. Ako je petlja u prvoj iteraciji, varijabla *tdol* koja se odnosi na vrijeme dolaska poprimit će vrijednost prvog generiranog uzorka vremena dolaska. S druge strane, varijabla *todl* koja se odnosi na vrijeme odlaska, poprimit će zbroj prvog generiranog uzorka vremena dolaska i prvog generiranog uzorka vremena posluživanja. U svim ostalim iteracijama, vrijeme dolaska će se kumulativno zbrajati s generiranim uzorcima vremena dolazaka, a vremena odlaska će se kumulativno zbrajati ovisno o

gore navedenim uvjetima. U suštini predstavlja nasumičan skup vrijednosti dobivenih iz prethodnih linija kôda u kojima se instanciraju objekti klasa eksponencijalne ili normalne distribucije iz kojih se dalje mogu postaviti temelji za izlazni skup vrijednosti.

Što se izlaznih vrijednosti tiče, one su izvedene kao teorijske i empirijske vrijednosti. Teorijske vrijednosti dobivene su pomoću formula opisanih u poglavlju 3, dok su empirijske vrijednosti one koje su izračunate iz skupa ulaznih vrijednosti. Linijama kôda na slici 10, prikazane su teoretske izlazne vrijednosti za $G/M/1$ model.

C#

```
1.  double intenzitetDolazaka = Convert.ToDouble(txtUzorciGM1.Text) / ukupanZbrojta
    ;
2.
3.  double promet = ukupanZbrojts / ukupanZbrojta;
4.  double Lq = promet + ((Math.Pow(promet, 2)) / (2 * (1 - promet))) * (1 + (Math.
    Pow((normTa.StandardDeviation / normTa.Mean), 2)));
5.  double Lw = Lq - promet;
6.  double Ts = promet / intenzitetDolazaka;
7.  double Tw = (((promet * Ts) / (2 * (1 - promet))) * (1 + (Math.Pow((normTa.Stan
    dardDeviation / normTa.Mean), 2))));
8.  double Tq = Tw + Ts;
9.  lblMiGM1.Text = "μ = " + (intenzitetDolazaka / promet).ToString().Substring(0,
    6);
10. lblLambdaGM1.Text = "λ = " + (intenzitetDolazaka).ToString().Substring(0, 6);
11. lblPrometGM1.Text = "a = " + (promet).ToString().Substring(0, 6);
```

Slika 10. Linije kôda za izračun teoretskih izlaznih vrijednosti

Kôdom opisanim na slici 10, prikazane vrijednosti dobivene su koristeći formule za navedeni model. Za srednje vrijednosti vremena čekanja i duljine reda, korištene su metode *StandardDeviation* pomoću koje se iz instanciranog objekta iz klase *NormalDistribution* dobiva standardno odstupanje i *Mean* koji iz istog instanciranog objekta pronalazi srednju vrijednost.

S druge strane, empirijski podaci za gore navedene izlazne vrijednosti dobivene su iz skupa ulaznih podataka. Podaci su dobiveni uzimanjem srednje vrijednosti iz liste podataka. Na primjer, zbroj skupa podataka podijeljen je s brojem elemenata koje taj skup ima. Pomoću kôda, jedan od empirijskih izračuna (u ovom slučaju duljina reda), izvodi se na način prikazanom na slici 11.

C#

```
1.  double empTw = listaUzorakaTw.Average();
2.  double emTq = listaTq.Average();
3.  double emTs = uzorciTs.ToList().Average();
4.  double emLw = 0;
5.
6.
7.  var unijaDolOdl = listatdol.Union(listatodl).OrderBy(i => i).ToList();
8.  bool pomZastavica = true;
9.  double brojac = 0;
10.
11. for (int i = 0; i < unijaDolOdl.Count; i++)
12. {
13.     for (int j = 0; j < listatdol.Count; j++)
14.     {
15.         if (unijaDolOdl[i] == listatdol[j])
16.         {
```

```
            pomZastavica = true;
            break;

        }
        else if (unijaDolOdl[i] == listatodl[j])
        {
            pomZastavica = false;
            break;
        }
    }

    if (pomZastavica)
    {
        brojac++;
        listaEmLq.Add(brojac);
    }
    else if (!pomZastavica)
    {
        brojac--;
        listaEmLq.Add(brojac);
    }
}

double emLq = listaEmLq.Average();

for (int i = 0; i < listaEmLq.Count; i++)
{
    if (listaEmLq[i] == 0 || listaEmLq[i] == 1)
    {
        listaEmLw.Add(0);
    }
    else if (listaEmLq[i] > 1)
    {
        listaEmLw.Add(listaEmLq[i] - 1);
    }
}
emLw = listaEmLw.Average();
```

Slika 11. Kôd za izračun empirijske vrijednosti duljine reda

Kôd prikazan na slici 11, prikazuje općenitu logiku izvršavanja izračuna empirijskih vrijednosti. Vrijednosti se zapisuju u listu koja se zatim pomoću metode *Average()* zapisuje u odgovarajuće varijable. Jedan od kompleksnijih izračuna je duljina reda, također prikazan na slici 11. Kako bi se moglo saznati u kojem trenutku se korisnik nalazi u redu ili ne, bilo je potrebno koristiti dvije *for* petlje kako bi se moglo saznati u kojem trenutku red ima ili nema korisnika. Također su korištene i varijable tipa *bool* nazvane *pomZastavice* koje služe kao pomoćne zastavice ako uvjet je ili nije ispunjen. Ukoliko je uvjet za duljinu reda bio ispunjen, odnosno vrijednost iz skupa *i* je odgovarala vrijednosti iz skupa *j* onda se pomoćna zastavica postavlja u status istine, odnosno *true*, te se brojač inkrementira za jedan i upisuje u listu empirijskih vrijednosti duljine reda. U suprotnom slučaju, ako uvjet nije ispunjen, onda se brojač dekrementira za jedan i upisuje u listu empirijskih vrijednosti duljine reda.

Simulator raspolaže i mogućnošću kreiranja grafova funkcije gustoće vjerojatnosti, graf vremena čekanja i graf duljine reda. Svi grafovi koriste *ZedGraph* biblioteku za generiranje

grafova te koriste podatke opisane u prethodnim linijama koda opisanim na slikama 7 do 11. Kao primjer kreiranja grafa funkcije gustoće vjerojatnosti $M/G/1$ prikazan je sljedećim linijama kôda na slici 12.

C#

```
GraphPane fgvGM1 = zgFGV_GM1.GraphPane;
Histogram histFGV_GM1 = new Histogram();
double sirinaRasponaDistribucije = 0.5;
histFGV_GM1.Compute(vrijednosti, 18, sirinaRasponaDistribucije);

for (int i = 0; i < zgVrijednosti.Length; i++)
{
    if (i == 0)
    {
        zgListaRaspona.Add("0" + "-" + sirinaRasponaDistribucije.ToString());
    }
    else if (i == zgVrijednosti.Length)
    {
        zgListaRaspona.Add((sirinaRasponaDistribucije * i).ToString() + "-
" + ((sirinaRasponaDistribucije * i) + sirinaRasponaDistribucije).ToString());

        break;
    }
    else
    {
        zgListaRaspona.Add((sirinaRasponaDistribucije * i).ToString() + "-
" + ((sirinaRasponaDistribucije * (i + 1)).ToString()));
    }

}


BarItem myBar = fgvGM1.AddBar("Broj korisnika", null, zgVrijednosti,
                                                Color.Red);
myBar.Bar.Fill = new Fill(Color.DarkBlue, Color.CadetBlue,
                                                 Color.AliceBlue);

LineItem myCurve = fgvGM1.AddCurve("Max broj korisnika u rasponu",
      null, zgVrijednosti, Color.DarkRed, SymbolType.Circle);


fgvGM1.XAxis.MajorTic.IsBetweenLabels = true;
var poljeSirineRaspona = zgListaRaspona.ToArray();

fgvGM1.XAxis.Scale.TextLabels = poljeSirineRaspona;

zgFGV_GM1.AxisChange();
zgFGV_GM1.Refresh();
```

Slika 12. Kôd za kreiranje grafova unutar simulatora

Kako bi se kreirala distribucija i dodijelile vrijednosti skupovima distribucije, koristi se klasa *Histogram* iz *Accord.Net* biblioteke unutar koje se zadaju vrijednosti, širina raspona distribucije te intervali po kojima će grupirati podatke. Dobivene vrijednosti se dodaju u instancu klase *BarItem* te se onda graf generira pomoću metoda *AxisChange()* i *Refresh()*. Također, kako ZedGraph ne može generirati oznake na osi x ili osi y, potrebno je bilo kreirati algoritam koji radi iste.

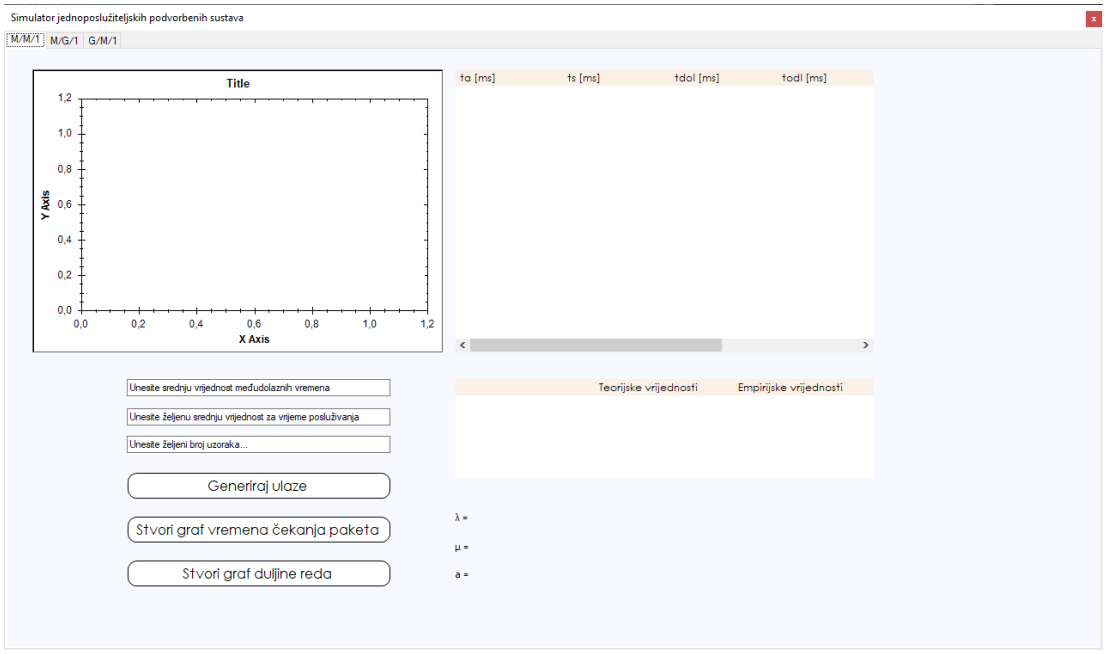
5.2. Prikaz rada simulatora

Kao što je već spomenuto u poglavlju 5, kao grafičko sučelje simulatora izabran je *Windows Forms*, odnosno skraćeno *WinForms* izgled sučelja. To je jednostavno razvojno okruženje za grafička sučelja te je iznimno popularno od edukativnih svrha pa sve do profesionalne upotrebe.

Sučelje simulatora zamišljeno je kao radni prostor u kojem se nalaze tri kartice koje označavaju pojedini jednoposlužiteljski model. Svaki model ima iste elemente, s pojedinim dodanim elementima ovisno o kojem se modelu radi. Elementi koji se nalaze na pojedinoj kartici su:

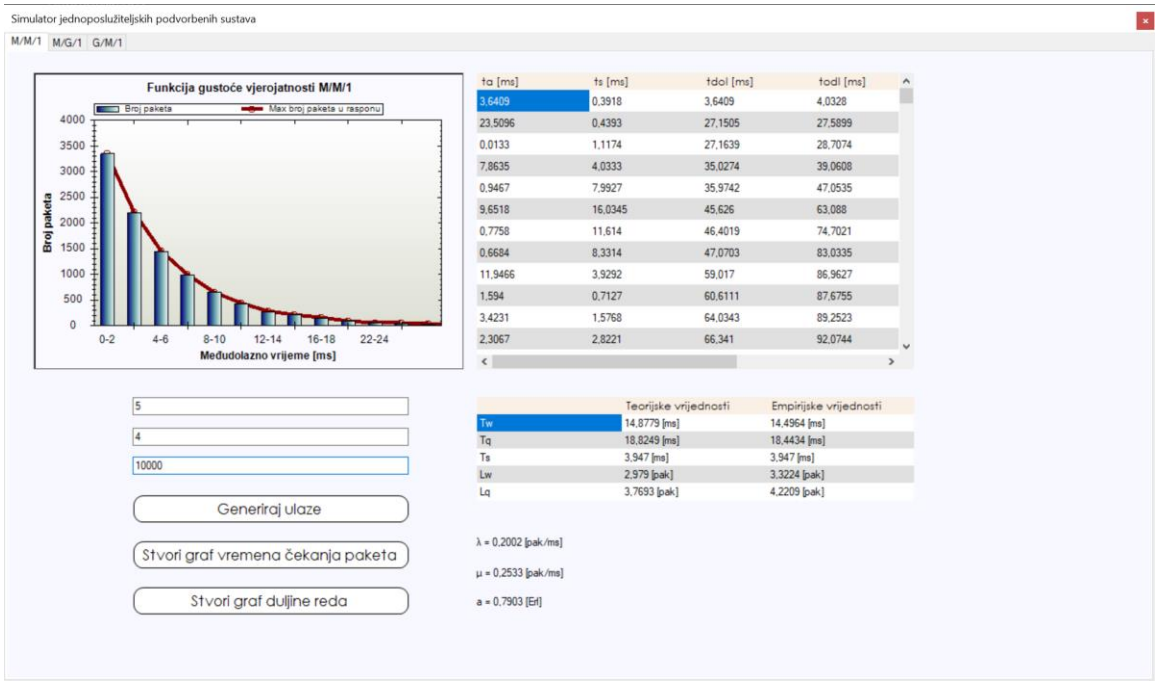
- tekstualna polja za unos traženih vrijednosti
- gumb za generiranje ulaznih i izlaznih vrijednosti te grafa funkcije gustoće vjerojatnosti
- gumb za generiranje grafa vremena čekanja
- gumb za generiranje grafa duljine reda
- *ZedGraph* radno okruženje u kojem se generiraju grafovi
- tablica ulaznih vrijednosti
- tablica izlaznih vrijednosti.

Simulator, odnosno njegovo sučelje prije nego što su unesene vrijednosti, prikazano je na slici 13.



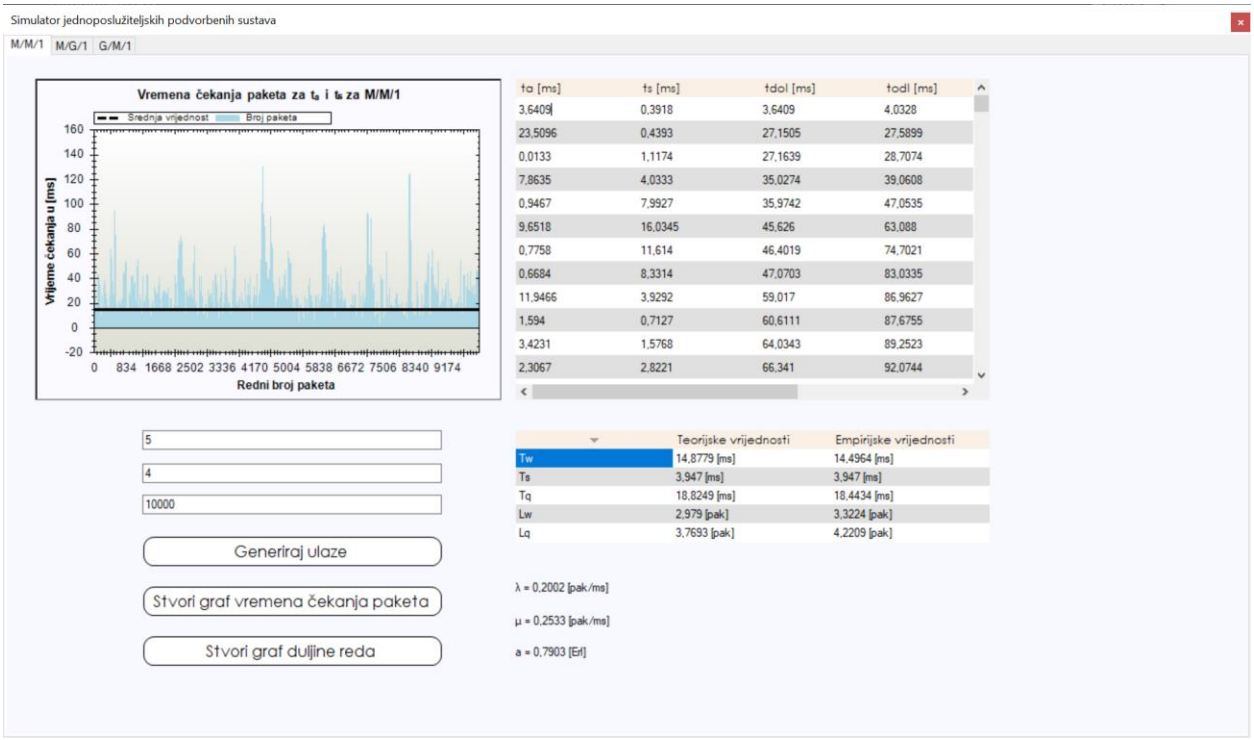
Slika 13. Sučelje simulatora jednoposlužiteljskih podvorbenih sustava

Unosom vrijednosti u tražena polja za unos teksta, simulator je spreman za generiranje ulaza pomoću gumba *Generiraj ulaze*. Pritiskom na gumb, generirat će skup ulaznih i izlaznih vrijednosti, kao i graf funkcije gustoće vjerojatnosti za željeni model jednoposlužiteljskog podvorbenog sustava. Prikaz generiranih ulaza i izračunatih izlaza zajedno s grafom funkcije gustoće vjerojatnosti za $M/M/1$ model prikazan je na slici 14.



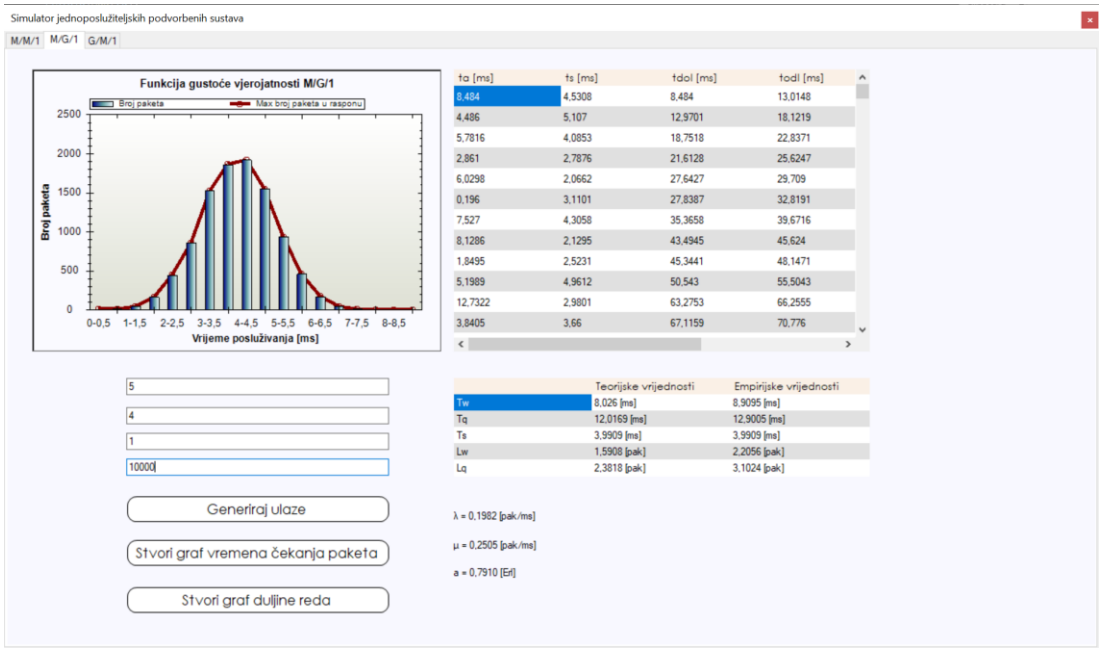
Slika 14. Prikaz generiranih ulaza i izračunatih izlaza simulatora za $M/M/1$ model

Prozorom za graf moguće se je koristiti tako da se uvećava i smanjuje željeno područje pretrage, vraćanje u izvornu veličinu te je moguće spremiti graf na računalo. Unutar tablica ulaznih i izlaznih veličina moguće je označavanje i kopiranje u neku drugu aplikaciju, kao na primjer, *Microsoft Excel*. Nadalje, bitno je spomenuti kako bi se grafovi vremena čekanja paketa i grafovi duljine reda mogli stvoriti, prvotno je potrebno generirati ulaze koji bi iste mogli omogućiti. Simulator s generiranim grafom vremena čekanja paketa prikazan je na slici 15.



Slika 15. Prikaz generiranih izlaznih veličina zajedno s grafom vremena čekanja paketa

U slučaju nepoissonovskih modela, $M/G/1$ i $G/M/1$, jedina razlika u sučelju je da se pojavljuju nova polja za unos. Nova polja su srednja vrijednost normalne distribucije μ i njeno standardno odstupanje σ. Prikaz simulatora s generiranim vrijednostima za $M/G/1$ nalazi se na slici 16.



Slika 16. Prikaz simulatora s generiranim vrijednostima za $M/G/1$

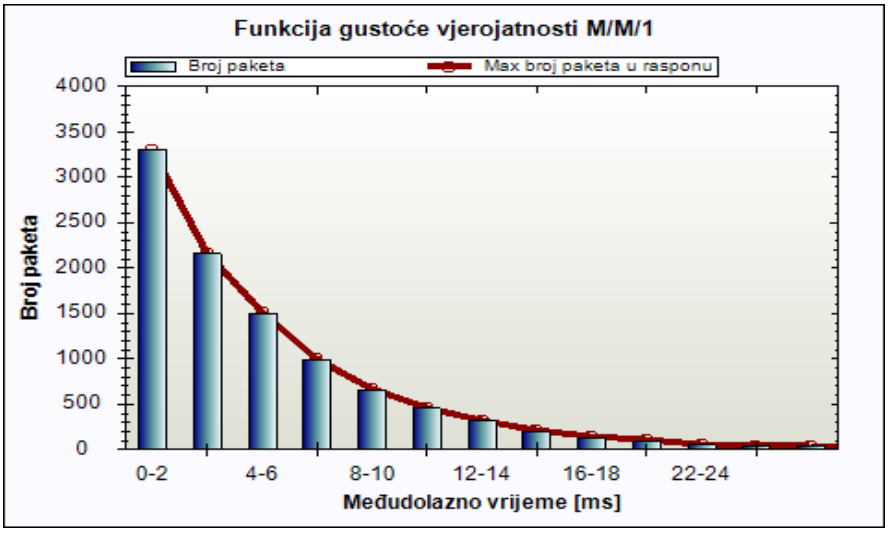
5.3 Izlazne vrijednosti simulacije

U ovom poglavlju opisan je praktični dio jednoposlužiteljskih modela. Jednoposlužiteljski modeli simulirani u radu su Poissonov $M/M/1/\infty/\infty/FCFS$ i nepoissonovski $M/G/1/\infty/\infty/FCFS$ i $G/M/1/\infty/\infty/FCFS$. Slučajni proces ulaznih vrijednosti dobiven simulacijom koristi srednje vrijednosti za eksponencijalnu i normalnu razdiobu te standardno odstupanje za normalnu razdiobu. Promatrani uzorak koji se koristi za dobivanje izlaznih vrijednosti je 10000 paketa te ga je moguće proizvoljno odrediti prilikom pokretanja, odnosno generiranja istih. Prometno opterećenje jednoposlužiteljskih modela jednako je dobivenom prometu $a = \rho$. Također, ulazni parametri međudolaznih vremena i vremena posluživanja kod $M/M/1$ modela ravnaju se po eksponencijalnoj razdiobi. Kod nepoissonovskih modela, $M/G/1$ model ima vremena posluživanja distribuirana po normalnoj razdiobi, a međudolazna vremena po eksponencijalnoj, dok $G/M/1$ model ima vremena posluživanja distribuirana po eksponencijalnoj razdiobi, a međudolazna vremena po normalnoj.

Izlazne vrijednosti za svaki od simuliranih modela, unutar sučelja simulatora, prikazane su tablicom izlaznih vrijednosti i podijeljene na empirijske i teorijske vrijednosti. Teorijske vrijednosti dobivene su prema standardnim formulama ovisno o promatranom jednoposlužiteljskom modelu, dok su empirijske vrijednosti dobivene iz prosjeka skupa ulaznih vrijednosti 10000 paketa sa pojedinačnim vrijednostima međudolaznih vremena t_a,vremena posluživanja t_s, vremena odlazaka paketa t_{odl}, vremena dolazaka paketa t_{dol}, vremena čekanja paketa t_w i vremena čekanja u sustavu t_q.

5.3.1 $M/M/1$ simulacija

Prva simulacija modela $M/M/1$ dobivena je iz unosa srednjih vrijednosti koje za međudolazno vrijeme iznose 5, a za vrijeme posluživanja 4. Već spomenuti uneseni uzorak simulacije iznosi 10000 paketa. S obzirom na to da se radi o eksponencijalnim ulaznim parametrima dobivena je funkcija gustoće vjerojatnosti prikazana na grafikonu 1. Prema grafikonu 1, može se vidjeti kako se najveći broj paketa, približno njih 3300 nalazi u rasponu između 0-2 [ms]. Zatim ga slijedi skup od približno 2100 paketa u rasponu od 2-4 [ms] te se daljnjim analiziranjem grafa funkcije gustoće vjerojatnosti za $M/M/1$ može ustanoviti da se međudolazna vremena paketa doista ravnaju prema eksponencijalnoj distribuciji.



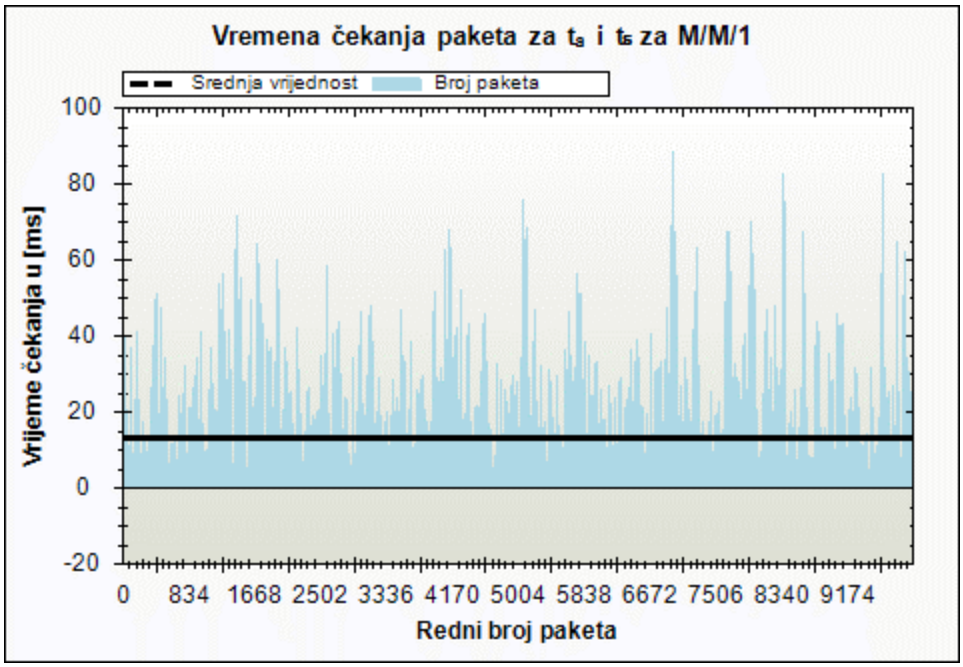
Grafikon 1. Graf funkcije gustoće vjerojatnosti za prvu simulaciju $M/M/1$

Izlazne vrijednosti za prvu simulaciju prikazane su na tablici 2. Kao što je već spomenuto, te vrijednosti su podijeljene na emprijske i teorijske. Teorijska vrijednost prosječnog vremena čekanja T_w iznosi 14,7372 $[ms]$ dok je empirijska vrijednost za istu 13,2031 $[ms]$. Teorijsko prosječno vrijeme čekanja T_s iznosi 3,9883 $[ms]$ i podudara se s empirijskom vrijednošću. Teorijsko prosječno vrijeme čekanja u sustavu T_q je 18,7255 $[ms]$ dok je empirijska vrijednost 17,1914 $[ms]$. Teorijska vrijenost prosječnog broja korisnika u redu L_w iznosi 2,908 $[pak]$, a emprijska vrijednost približno je slična teorijskoj i glasi 2,9851 $[pak]$. Teorijski prosječan broj korisnika u sustavu L_q glasi 3,695 $[pak]$ i također blizu je iznosu empirijske vrijednosti od 3,8799 $[pak]$. Iznos prometa navedene prve simulacije je $a = 0{,}7870$ $[Erl]$ dok brzina dolazaka λ iznosi 0,1973 $[pak/ms]$, a brzina posluživanja μ iznosi 0,2506 $[pak/ms]$.

Tablica 2. Izlazne vrijednosti za prvu simulaciju $M/M/1$

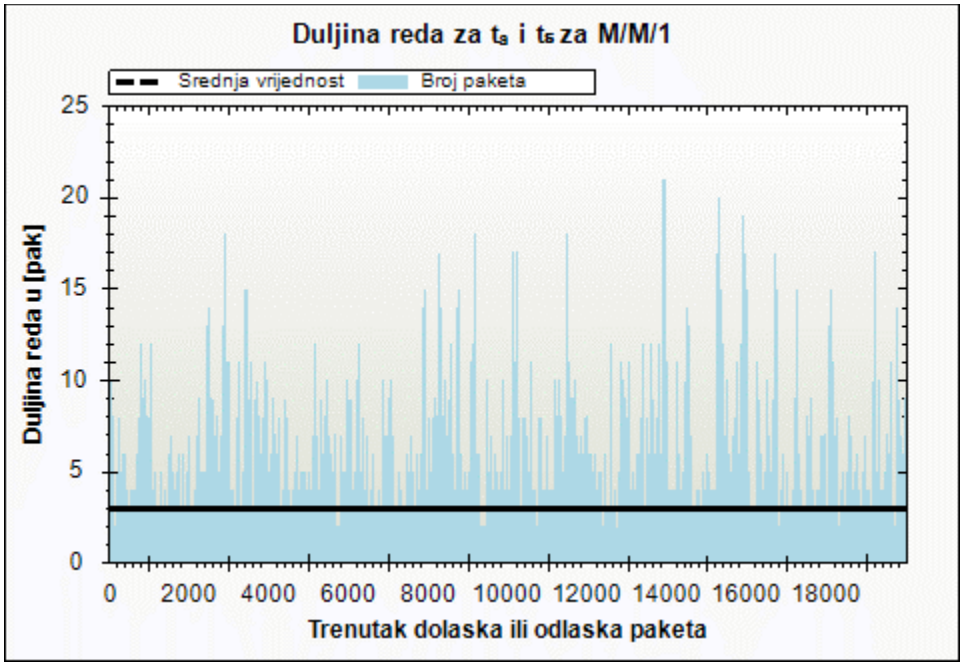
	Teorijske vrijednosti	Empirijske vrijednosti
Tw	14,7372 [ms]	13,2031 [ms]
Tq	18,7255 [ms]	17,1914 [ms]
Ts	3,9883 [ms]	3,9883 [ms]
Lw	2,908 [pak]	2,9851 [pak]
Lq	3,695 [pak]	3,8799 [pak]

Grafički prikaz vremena čekanja za t_a i t_s za $M/M/1$ model prikazan je na grafikonu 2. Također, vidljiva je srednja vrijednost vremena čekanja paketa T_w u iznosu od 13,2031 $[ms]$ označena crnom linijom. Iz grafikona 2 se može vidjeti vrijeme čekanja za svaki paket t_w te je najduže vrijeme čekanja iznosilo približno 85 $[ms]$.



Grafikon 2. Graf vremena čekanja za prvu simulaciju $M/M/1$

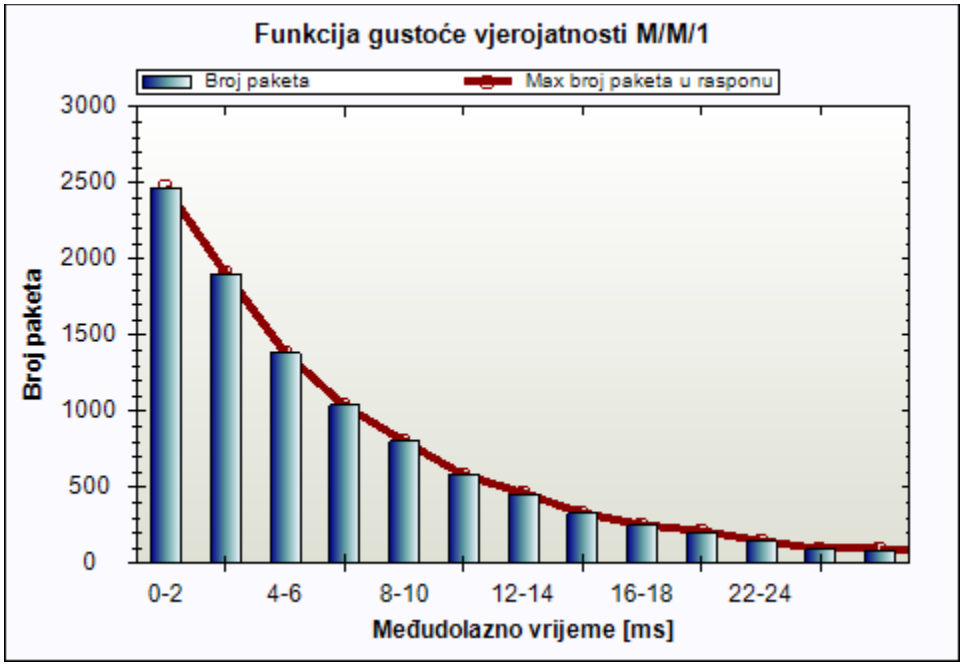
Grafički prikaz duljine reda za t_a i t_s za $M/M/1$ modela prve simulacije vidljiv je na grafikonu 3. Označeno crnom linijom je srednja vrijednost broja korisnika u redu L_w iznosi 2,9851 $[pak]$. Iz grafa se da isčitati da je najveća duljina reda bila 21 $[pak]$.



Grafikon 3. Graf duljine reda za prvu simulaciju $M/M/1$

Za potrebe usporedbe simulacije, provedena je i druga simulacija s približno jednakim prometnim opterećenjem. Druga simulacija modela $M/M/1$ dobivena je iz unosa srednjih

vrijednosti koje za međudolazno vrijeme iznose 7, a za vrijeme posluživanja 5,5. Također kao i kod prve simulacije, uzorak iznosi 10000 paketa. Također se radi o eksponencijalnim ulaznim parametrima, te se dobije graf funkcije gustoće vjerojatnosti prikazan na grafikonu 4. Prema grafikonu 4, može se vidjeti kako se najveći broj paketa, približno njih 2500 nalazi u rasponu između 0-2 $[ms]$. Zatim ga slijedi skup od približno 1800 paketa u rasponu od 2-4 $[ms]$ te se daljnjim analiziranjem grafa funkcije gustoće vjerojatnosti za $M/M/1$ može ustanoviti da se međudolazna vremena paketa doista ravnaju prema eksponencijalnoj funkciji, ali u odnosu na prvu simulaciju, veća su međudolazna vremena.



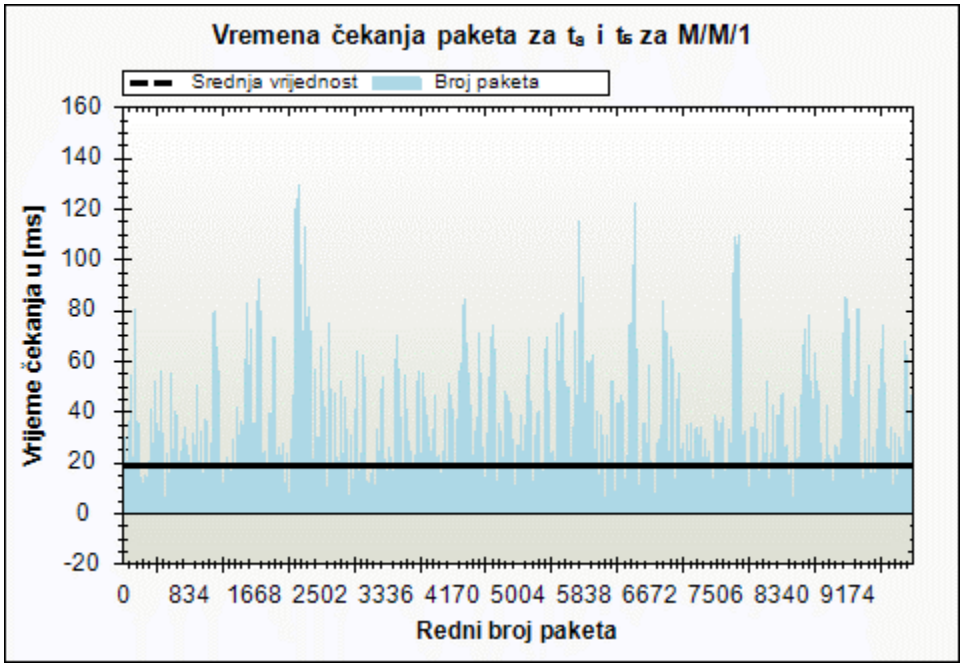
Grafikon 4. Graf funkcije gustoće vjerojatnosti za drugu simulaciju $M/M/1$

Izlazne vrijednosti za drugu simulaciju prikazane su na tablici 3. Kao što je već spomenuto, te vrijednosti su podijeljene na empirijske i teorijske. Teorijska vrijednost prosječnog vremena čekanja T_w iznosi 18,5017 $[ms]$ dok je empirijska vrijednost za istu 18,3006 $[ms]$. Teorijsko prosječno vrijeme čekanja T_s iznosi 5,4319 $[ms]$ i podudara se s empirijskom vrijednošću. Teorijsko prosječno vrijeme čekanja u sustavu T_q je 23,9337 $[ms]$ dok je empirijska vrijednost 23,7326 $[ms]$. Teorijska vrijednost prosječnog broja korisnika u redu L_w iznosi 2,633 $[pak]$, a empirijska vrijednost približno je slična teorijskoj i glasi 2,964 $[pak]$. Teorijski prosječan broj korisnika u sustavu L_q glasi 3,406 $[pak]$ i također blizu je iznosu empirijske vrijednosti od 3,8515 $[pak]$. Iznos prometa navedene prve simulacije je $a = 0{,}7730$ $[Erl]$ dok brzina dolazaka λ iznosi 0,1423 $[pak/ms]$, a brzina posluživanja μ iznosi 0,1840 $[pak/ms]$.

Tablica 3. Izlazne vrijednosti za drugu simulaciju $M/M/1$

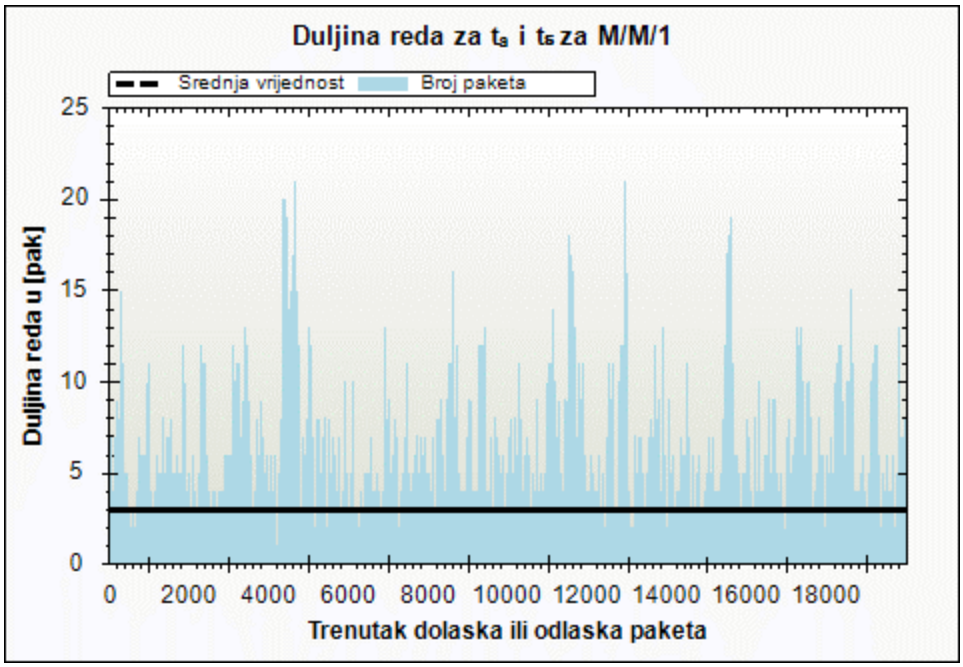
	Teorijske vrijednosti	Empirijske vrijednosti
Tw	18,5017 [ms]	18,3006 [ms]
Tq	23,9337 [ms]	23,7326 [ms]
Ts	5,4319 [ms]	5,4319 [ms]
Lw	2,633 [pak]	2,964 [pak]
Lq	3,406 [pak]	3,8515 [pak]

Grafički prikaz vremena čekanja za t_a i t_s za $M/M/1$ model prikazan je na grafikonu 5. Također, vidljiva je srednja vrijednost vremena čekanja paketa T_w u iznosu od 18,3006 $[ms]$ označena crnom linijom. Iz grafikona 5 se može vidjeti vrijeme čekanja za svaki paket t_w te je najduže vrijeme čekanja iznosilo približno 140 $[ms]$.



Grafikon 5. Graf vremena čekanja za drugu simulaciju $M/M/1$

Grafički prikaz duljine reda za t_a i t_s $M/M/1$ modela druge simulacije vidljiv je na grafikonu 6. Označeno crnom linijom je srednja vrijednost broja korisnika u redu L_w iznosi 3,8515 $[pak]$. Iz grafa se može vidjeti da je najveća duljina reda bila 22 $[pak]$.



Grafikon 6. Graf duljine reda za drugu simulaciju $M/M/1$

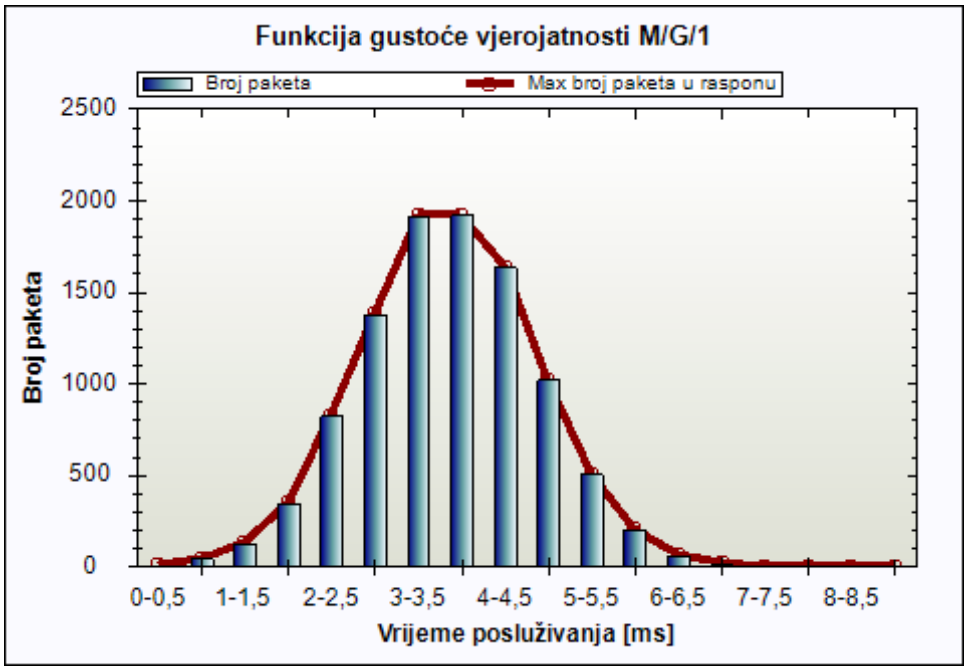
Usporede li se tablice 2 i 3, odnosno izlazne veličine prve i druge simulacije, dolazi se do zaključka da se te dvije simulacije razlikuju u iznosima prosječnih vremena čekanja T_w $\left(T_w = \frac{\rho \cdot T_s}{1-\rho}\right)$, a onda posljedično u iznosima prosječnog vremena zadržavanja u sustavu T_q ($T_q = T_w + T_s$). Iako su oba sustava opterećena istom količinom prometa prosječno vrijeme čekanja je veće za simulirani sustav s duljim vremenom posluživanja.

Prosječan broj korisnika u sustavu L_q i prosječna vrijednost broja korisnika koji čekaju L_w su približno jednake, što je također očekivano s obzirom na to da te dvije veličine ovise isključivo o prometnom opterećenju za analizirani model ($L_q = \frac{\rho}{1-\rho}, L_w = \frac{\rho^2}{1-\rho}$). Razlika se pojavljuje zbog male razlike u prometnom opterećenju.

5.3.2 $M/G/1$ simulacija

Prva simulacija modela $M/G/1$ ima ulazne vrijednosti dolazaka paketa eksponencijalno distribuirane stoga međudolazno vrijeme iznosi 5, a za vrijeme posluživanja koristi se normalna razdioba s parametrima srednje vrijednosti μ koji u ovom slučaju iznosi 4 i standardne devijacije σ iznosa 1. Već spomenuti uneseni uzorak simulacije iznosi 10000 paketa. Prema grafikonu 7, prikazana je funkcija gustoće vjerojatnosti vremena posluživanja. Iz navedene funkcije može se vidjeti kako se najveći broj paketa, približno njih 2000 nalazi u rasponu između 3,5-4 [ms] te iza njega slijedi nešto manji broj paketa, otprilike 1900, u rasponu od 3-3,5 [ms]. Sljedeći najveći broj paketa iznosi približno 1700 paketa, razdijeljen u razred od 4-4,5 [ms] te nešto manji broj paketa,

otprilike 1400, nalazi se u razredu od 3-3,5 $[ms]$ prema čemu se može zaključiti da se radi o normalnoj distribuciji. Razredi normalne distribucije grupirani su prema razmacima vremena posluživanja od 0,5 $[ms]$.



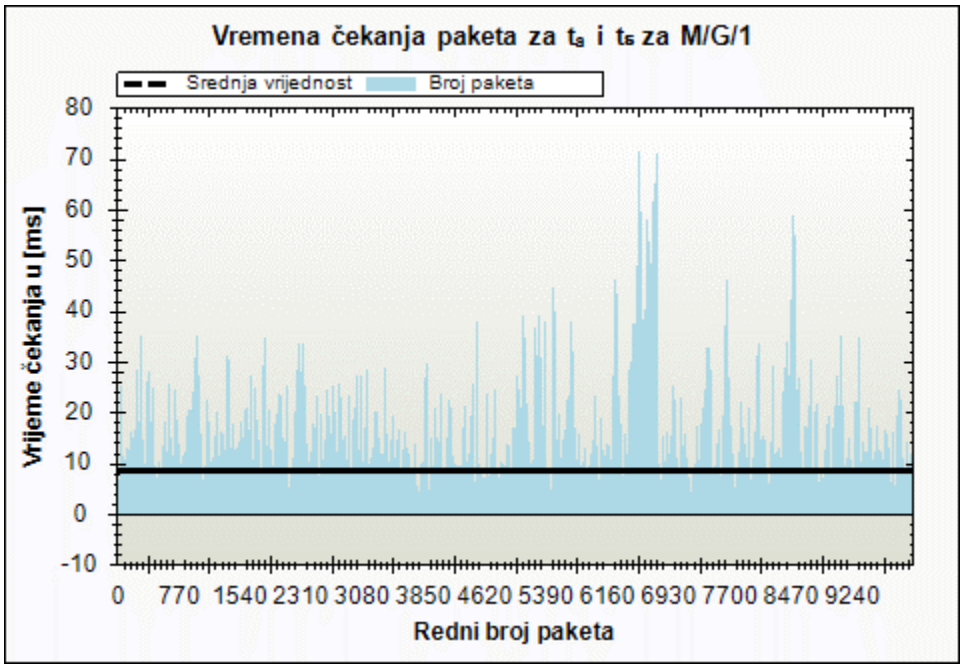
Grafikon 7. Graf funkcije gustoće vjerojatnosti za prvu simulaciju $M/G/1$

Izlazne vrijednosti za prvu simulaciju prikazane su na tablici 4. Teorijska vrijednost prosječnog vremena čekanja T_w iznosi 8,0678 $[ms]$ dok je empirijska vrijednost za istu 8,315 $[ms]$. Teorijsko prosječno vrijeme čekanja T_s iznosi 3,9862 $[ms]$ i podudara se s empirijskom vrijednošću. Teorijsko prosječno vrijeme čekanja u sustavu T_q je 12,0541 $[ms]$ dok je empirijska vrijednost 12,3012 $[ms]$. Teorijska vrijednost prosječnog broja korisnika u redu L_w iznosi 1,6031 $[pak]$, a empirijska vrijednost glasi 2,0534 $[pak]$. Teorijski prosječan broj korisnika u sustavu L_q glasi 2,3952 $[pak]$, a empirijska vrijednost iznosi 2,9477 $[pak]$. Iznos prometa navedene prve simulacije je $a = 0{,}7920[Erl]$ dok brzina dolazaka λ iznosi 0,1987 $[pak/ms]$, a brzina posluživanja μ iznosi 0,2508 $[pak/ms]$.

Tablica 4. Izlazne veličine prve simulacije $M/G/1$

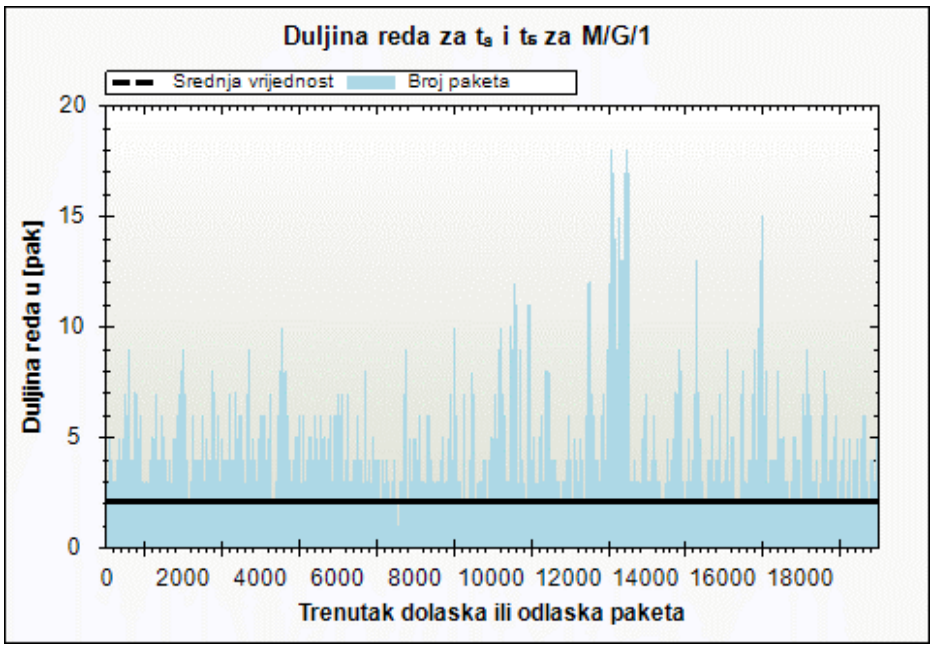
	Teorijske vrijednosti	Empirijske vrijednosti
Tw	8,0678 [ms]	8,315 [ms]
Tq	12,0541 [ms]	12,3012 [ms]
Ts	3,9862 [ms]	3,9862 [ms]
Lw	1,6031 [pak]	2,0534 [pak]
Lq	2,3952 [pak]	2,9477 [pak]

Grafički prikaz vremena čekanja za t_a i t_s $M/G/1$ model prikazan je na slici 11. Također, vidljiva je srednja vrijednost vremena čekanja paketa T_w u iznosu od 8,315 $[ms]$ označena crnom linijom. Iz grafikona 8 se može vidjeti vrijeme čekanja za svaki paket t_w te je najduže vrijeme čekanja iznosilo približno 70 $[ms]$.



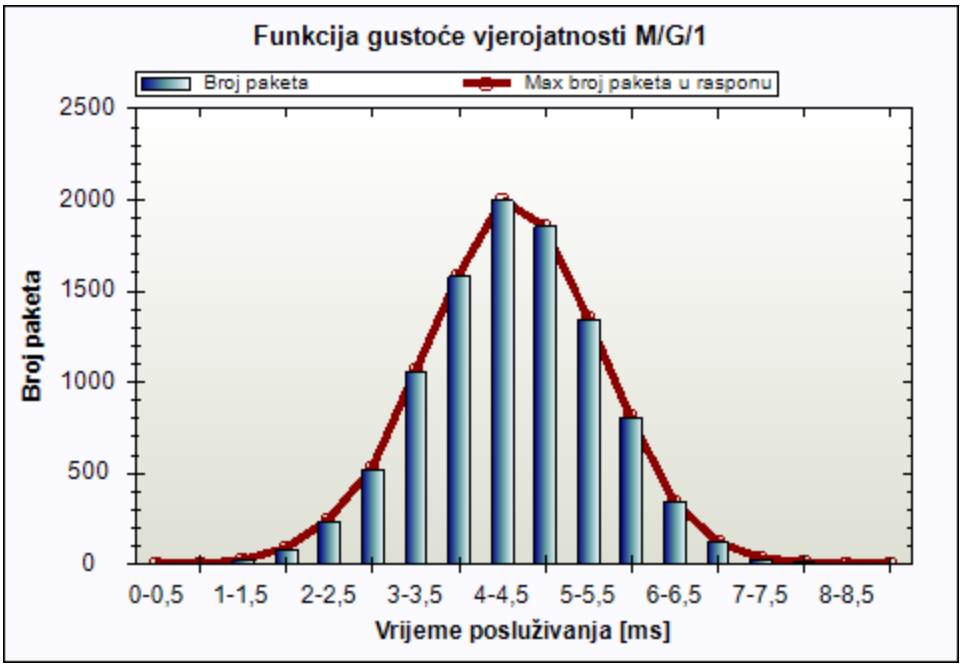
Grafikon 8. Graf vremena čekanja za prvu simulaciju $M/G/1$

Grafički prikaz duljine reda za t_ai t_s $M/G/1$ modela druge simulacije vidljiv je na grafikonu 9. Crna linija označava srednju vrijednost broja korisnika u redu L_w i iznosi 2,0534 $[pak]$. Iz grafa se može vidjeti da je najveća duljina reda bila 18 $[pak]$.



Grafikon 9. Graf duljine reda prve simulacije $M/G/1$

Kod druge simulacije modela $M/G/1$, međudolazno vrijeme iznosi 7, a vrijeme posluživanja, također normalne razdiobe s parametrima srednje vrijednosti μ koji u ovom slučaju iznosi 5,5 i standardne devijacije σ iznosa 1. Grafikon 10 prikazuje funkciju gustoće vjerojatnosti vremena posluživanja koja ima najveći broj paketa, približno njih 2000, u rasponu između 4-4,5 $[ms]$. Nešto manji broj paketa, otprilike 1800, nalazi se u rasponu od 4,5-5 $[ms]$. Sljedeći najveći broj paketa iznosi približno 1600 paketa, pripada razredu od 3,5-4 $[ms]$. Iza njega slijedi nešto manji broj paketa, otprilike 1300, u razredu vremena posluživanja od 5-5,5 $[ms]$ što je pokazatelj da se radi o normalnoj distribuciji.



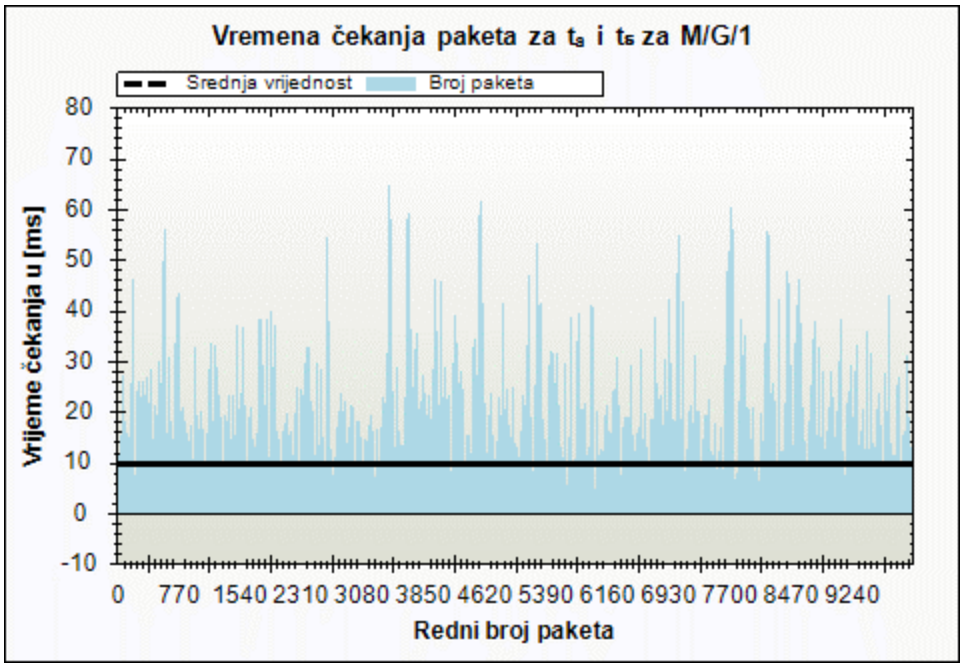
Grafikon 10. Funkcija gustoće vjerojatnosti druge simulacije $M/G/1$

Izlazne veličine druge simulacije prikazane su na tablici 5. Prosječno vrijeme čekanja T_w teorijske vrijednosti iznosi 10,4816 $[ms]$ dok je empirijska vrijednost za isto 9,7932 $[ms]$. Teorijsko i empirijsko prosječno vrijeme čekanja T_s podudaraju se iznosima te ono glasi 5,4948 $[ms]$. Teorijsko prosječno vrijeme čekanja u sustavu T_q je 15,9764 $[ms]$ dok je empirijska vrijednost 15,288 $[ms]$. Teorijska vrijednost prosječnog broja korisnika u redu L_w iznosi 1,501 $[pak]$, a empirijska vrijednost glasi 1,67746 $[pak]$. Teorijski prosječan broj korisnika u sustavu L_q glasi 2,288 $[pak]$, a empirijska vrijednost iznosi 2,6676 $[pak]$. Iznos prometa navedene druge simulacije je $a = 0{,}7869\ [Erl]$ dok brzina dolazaka λ iznosi 0,1432 $[pak/ms]$, a brzina posluživanja μ iznosi 0,1819 $[pak/ms]$.

Tablica 5. Izlazne veličine druge simulacije $M/G/1$

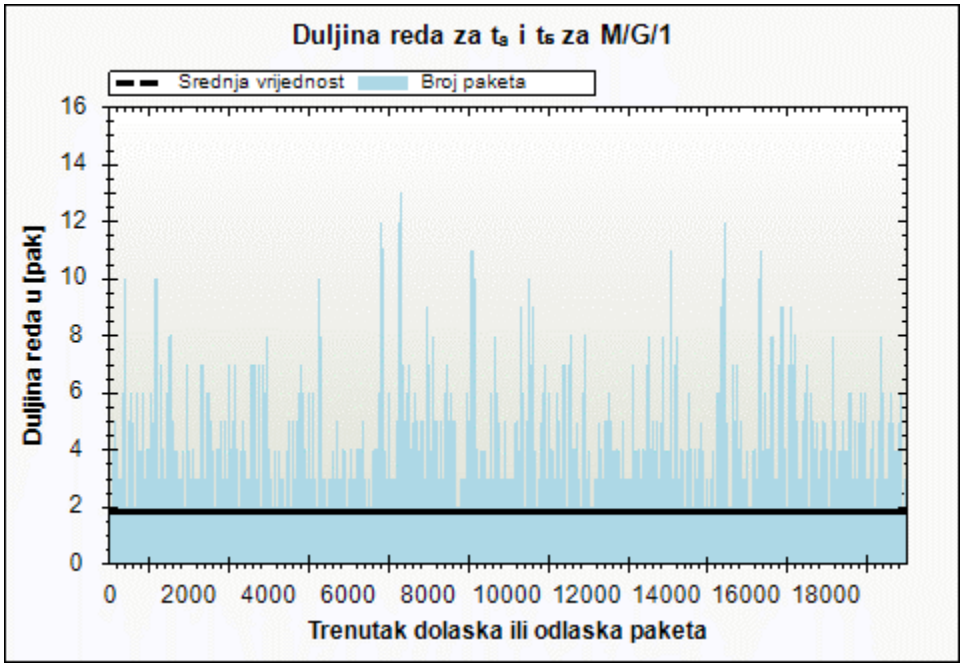
	Teorijske vrijednosti	Empirijske vrijednosti
Tw	10,4816 [ms]	9,7932 [ms]
Tq	15,9764 [ms]	15,288 [ms]
Ts	5,4948 [ms]	5,4948 [ms]
Lw	1,501 [pak]	1,7746 [pak]
Lq	2,288 [pak]	2,6676 [pak]

Grafički prikaz vremena čekanja za t_a i t_s za $M/G/1$ model prikazan je na grafikonu 11. Srednja vrijednost vremena čekanja paketa T_w vidljiva na grafikonu kao crna linija, iznosi 9,7932 $[ms]$. Svako pojedino vrijeme čekanja paketa t_w označen je grafički kao stupac u uzorku od 1000 paketa te je najduže vrijeme čekanja bio približno 65 $[ms]$.



Grafikon 11. Vremena čekanja druge simulacije $M/G/1$

Grafički prikaz duljine reda za t_a i t_s za $M/G/1$ modela druge simulacije vidljiv je na grafikonu 12. Kao što je navedeno, crna linija označava srednju vrijednost broja korisnika u redu L_w i ona iznosi 1,7746 $[pak]$. Također, iščitavanjem sa grafikona može se vidjeti da je najveća duljina reda iznosila 13 $[pak]$.



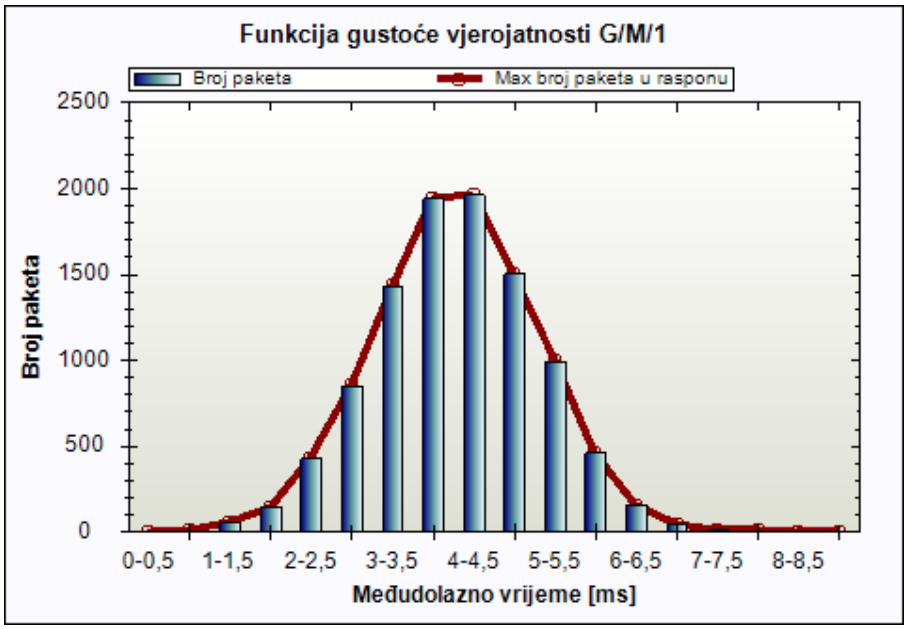
Grafikon 12. Duljina reda za drugu simulaciju $M/G/1$

Usporedivši tablice 4 i 5, odnosno izlazne veličine prve i druge simulacije modela $M/G/1$, dolazi se do zaključka da se te dvije simulacije razlikuju u iznosima prosječnih vremena čekanja T_w $\left(\frac{aT_s}{2(1-a)}\left[1+\left(\frac{\sigma_{\tau_s}}{T_s}\right)^2\right]\right)$, a zatim kao posljedično u iznosima prosječnog vremena zadržavanja u sustavu T_q ($T_q = T_w + T_s$).

Prosječan broj korisnika u sustavu L_q i prosječna vrijednost broja korisnika koji čekaju L_w su približno slične između dvije simulacije, a to je i očekivano s obzirom na to da te dvije veličine dominantno ovise o prometnom opterećenju za analizirani model $\left(L_q = \rho + \frac{a^2}{2(1-\rho)}\left[1+\left(\frac{\sigma_{\tau_s}}{T_s}\right)^2\right], L_w = L_q - \rho\right)$. Razlika se pojavljuje zbog male razlike u prometnom opterećenju.

5.3.3. $G/M/1$ simulacija

Prva simulacija modela $G/M/1$ ima ulazne vrijednosti dolazaka paketa normalno distribuirane sa srednjom vrijednošću μ koja iznosi 5 i standardnom devijacijom σ koja iznosi 1. Vrijeme posluživanja je eksponencijalno distribuirano sa srednjom vrijednošću 4. Prema grafikonu 13, prikazana je funkcija gustoće vjerojatnosti vremena dolazaka paketa. Iz navedene funkcije može se vidjeti kako se najveći broj paketa, približno njih 2000 nalazi u rasponu između 4-4,5 [ms]. Otprilike 1950 paketa u rasponu od 3,5-4 [ms] slijedi iza najvećeg broja paketa. Sljedeći najveći broj paketa iznosi približno 1500 paketa i nalazi se u razred od 4,5-5 [ms] te nešto manji broj paketa, otprilike 1400, nalazi se u razredu od 3-3,5 [ms]. Prema grafičkom prikazu, također se radi o normalnoj distribuciji vremena dolazaka. Razredi normalne distibucije grupirani su prema razmacima vremena dolazaka od 0,5 [ms].



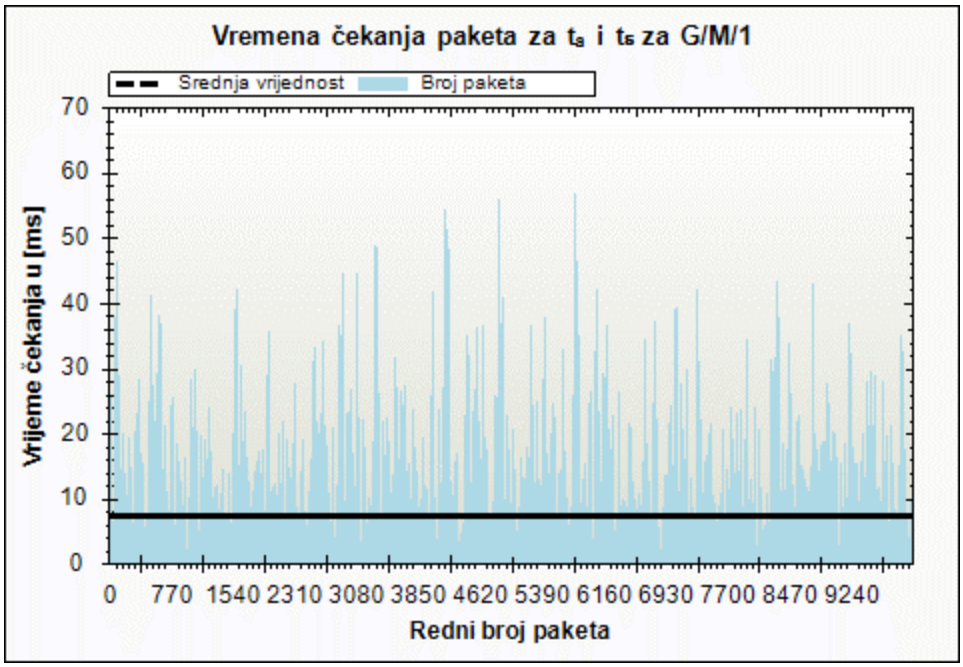
Grafikon 13. Funkcija gustoće vjerojatnosti prve simulacije za $G/M/1$

Izlazne vrijednosti za simulaciju prikazane su na tablici 6. Teorijska vrijednost prosječnog vremena čekanja T_w iznosi 8,894 $[ms]$ dok je empirijska vrijednost za istu 7,3374 $[ms]$. Teorijsko prosječno vrijeme čekanja T_s iznosi 4,0362 $[ms]$ i podudara se s empirijskom vrijednošću. Teorijsko prosječno vrijeme čekanja u sustavu T_q je 12,9302 $[ms]$ dok je empirijska vrijednost 11,3737 $[ms]$. Teorijska vrijednost prosječnog broja korisnika u redu L_w iznosi 1,7828 $[pak]$, a empirijska vrijednost glasi 1,5037 $[pak]$. Teorijski prosječan broj korisnika u sustavu L_q glasi 2,5918 $[pak]$, a empirijska vrijednost iznosi 2,3363 $[pak]$. Iznos prometa navedene prve simulacije je $a = 0{,}8090\ [Erl]$ dok brzina dolazaka λ iznosi 0,2004 $[pak/ms]$, a brzina posluživanja μ iznosi 0,2477 $[pak/ms]$.

Tablica 6. Izlazne veličine prve simulacije $G/M/1$

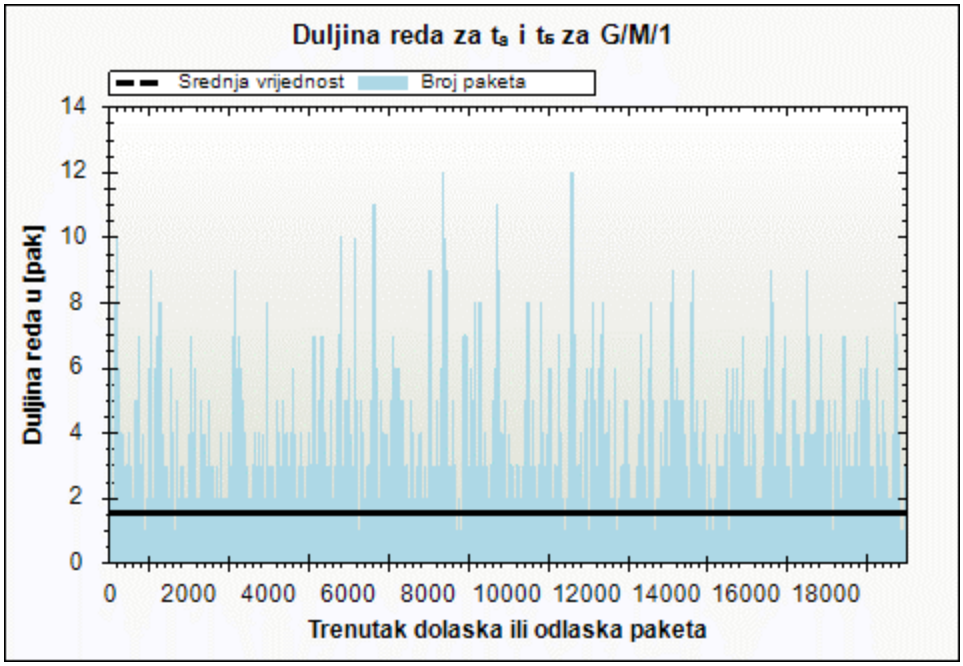
	Teorijske vrijednosti	Empirijske vrijednosti
Tw	8,894 [ms]	7,3374 [ms]
Tq	12,9302 [ms]	11,3737 [ms]
Ts	4,0362 [ms]	4,0362 [ms]
Lw	1,7828 [pak]	1,5073 [pak]
Lq	2,5918 [pak]	2,3363 [pak]

Grafički prikaz vremena čekanja za t_a i t_s za $G/M/1$ model prikazan je na grafikonu 14. Također, vidljiva je srednja vrijednost vremena čekanja paketa T_w u iznosu od 7,3374 $[ms]$. Nadalje, svako pojedino vrijeme čekanja paketa t_w vidljiv je na grafikonu 10 te najduže vrijeme čekanja iznosi približno 59 $[ms]$.



Grafikon 14. Vremena čekanja za prvu simulaciju $G/M/1$

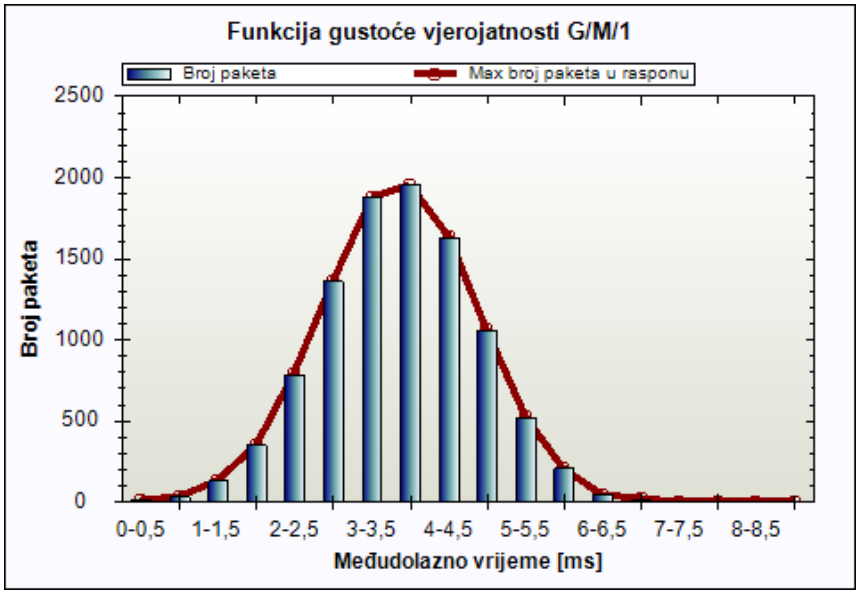
Sljedeći grafički prikaz vidljiv na grafikonu 15 prikazuje duljine reda za t_a i t_s za $G/M/1$ model. Crna linija ili srednja vrijednost broja korisnika u redu L_w iznosi 1,5073 $[pak]$. Iz grafa se može vidjeti da je najdulja duljina reda bila 12 $[pak]$.



Grafikon 15. Duljina reda za prvu $G/M/1$ simulaciju

Sljedeća, druga simulacija za model $G/M/1$ ima ulazne vrijednosti dolazaka paketa normalne distribucije srednje vrijednosti μ koji iznosi 7 i standardne devijacije σ iznosa 1 stoga, a

za vrijeme dolazaka koristi se eksponencijalna distribucija sa srednjom vrijednosti 5,5. Na grafikonu 16, prikazana je funkcija gustoće vjerojatnosti vremena dolazaka prema kojoj se može vidjeti najveći broj paketa, približno njih 2000, smještenih u rasponu između 3,5-4 $[ms]$. Odmah do njega nalazi se 1800 paketa, u rasponu od 3-3,5 $[ms]$. Treći supac najvećeg broja paketa sadrži njih približno 1600 i smješten je u razred od 4-4,5 $[ms]$ te iza njega slijedi četvrti stupac po broju paketa, njih otprilike 1300, i nalazi se u razredu od 2,5-3 $[ms]$. Razredi normalne distibucije grupirani su prema razmacima vremena dolazaka od 0,5 $[ms]$.



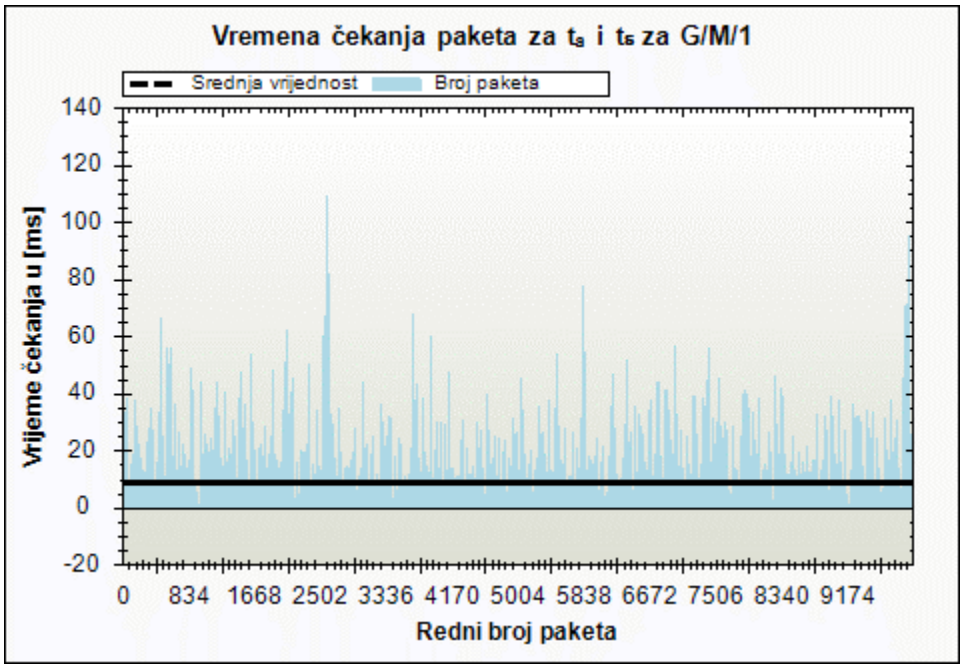
Grafikon 16. Graf funkcije gustoće vjerojatnosti za drugu simulaciju $G/M/1$

Izlazne veličine za simulaciju prikazane su na tablici 7. Teorijska vrijednost prosječnog vremena čekanja T_w iznosi 10,1019 $[ms]$ dok je empirijska vrijednost za istu 8,613 $[ms]$. Teorijsko prosječno vrijeme čekanja T_s iznosi 5,4802 $[ms]$ i isto je kao i ono empirijske vrijednosti. Teorijsko prosječno vrijeme čekanja u sustavu T_q je 15,5821 $[ms]$ dok je empirijska vrijednost 14,0933 $[ms]$. Teorijska vrijednost prosječnog broja korisnika u redu L_w iznosi 1,4437 $[pak]$, a emprijska vrijednost glasi 1,2679 $[pak]$. Teorijski prosječan broj korisnika u sustavu L_q glasi 2,2269 $[pak]$, a empirijska vrijednost iznosi 2,0731 $[pak]$. Iznos prometa navedene prve simulacije je $a = 0{,}7832$ $[Erl]$ dok brzina dolazaka λ iznosi 0,1429 $[pak/ms]$, a brzina posluživanja μ iznosi 0,1429 $[pak/ms]$.

Tablica 7. Izlazne veličine druge $G/M/1$ simulacije

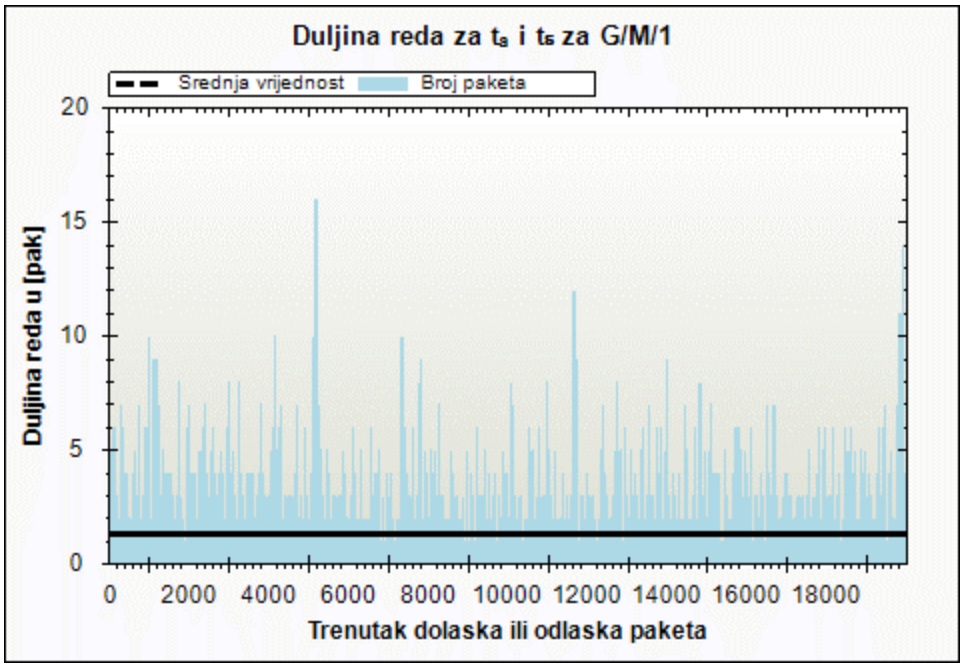
	Teorijske vrijednosti	Empirijske vrijednosti
Tw	10,1019 [ms]	8,613 [ms]
Tq	15,5821 [ms]	14,0933 [ms]
Ts	5,4802 [ms]	5,4802 [ms]
Lw	1,4437 [pak]	1,2679 [pak]
Lq	2,2269 [pak]	2,0731 [pak]

Na grafikonu 17 prikazan je grafički prikaz vremena čekanja za t_a i t_s za $G/M/1$ model. Iz istog prikaza vidljiva je srednja vrijednost vremena čekanja paketa T_w u iznosu od 8,613 $[ms]$ označena crnom linijom. Na grafikonu 10 može se vidjeti vrijeme čekanja za svaki paket t_w. Najduže vrijeme čekanja približno iznosi 110 $[ms]$.



Grafikon 17. Vremena čekanja druge simulacije $G/M/1$

Grafički prikaz duljine reda za t_a i t_s za $G/M/1$ modela druge simulacije vidljiv je na grafikonu 18. Srednja vrijednost broja korisnika u redu L_w iznosi 1,2679 $[pak]$. Prema grafikonu, može se vidjeti da je najdulja duljina reda bila 16 $[pak]$.



Grafikon 18. Prikaz duljine reda za drugu simulaciju $G/M/1$

Usporedivši tablice 6 i 7, odnosno izlazne veličine prve i druge simulacije modela $G/M/1$, dolazi se do zaključka da se te dvije simulacije razlikuju u iznosima prosječnih vremena čekanja T_w $\left(T_w = \frac{r_0}{1-r_0} \cdot T_s\right)$, a zatim kao posljedično u iznosima prosječnog vremena zadržavanja u sustavu T_q $(T_q = \frac{\mu}{1-r_0})$.

Prosječan broj korisnika u sustavu L_q i prosječna vrijednost broja korisnika koji čekaju L_w su približno slične između dvije simulacije, a to je i očekivano s obzirom na to da te dvije veličine pretežito ovise o prometnom opterećenju za analizirani model $\left(L_q = \frac{r_0}{1-r_0}, L_w = r_0 L_q\right)$. Razlika se pojavljuje zbog male razlike u prometnom opterećenju.

6. ANALIZA I USPOREDBA IZLAZNIH VELIČINA SIMULACIJE

Izvedbama simulacije dobiveni su različiti podaci izlaznih veličina. Simulacije navedene u prethodnom poglavlju prikazuju statistički različite skupove podataka koje je moguće dobiti simulirajući pojedine modele. Izlazni podaci dobiveni simulacijama jednoposlužiteljskih modela mogu se usporediti kako bi se utvrdila iskoristivost pojedinog modela. Također, usporedni prikaz rezultata simulacija kao izlaznih veličina razlikuje se ovisno radi li se o empirijskim ili teorijskim vrijednostima te će se iste koristiti kao pokazatelj odstupanja rezultata pojedinog modela. Kao što je spomenuto, teorijske vrijednosti dobivene su prema formulama za rješavanje podvorbenih modela dok su empirijske vrijednosti izlaznih veličina temeljene na analizi ulaznog skupa vrijednosti. Veličina ulaznog skupa podataka ovisi o točnosti izlaznih rezultata.

U tablici 8 prikazane su empirijske vrijednosti izlaznih veličina prve simulacije poissonovskih i nepoissonovskih modela. Također, zbog usporedbe rezultata, zadržano je približno prometno opterećenje između simuliranih modela. Vidljivo je da su T_w i T_q najkraća kod nepoissonovih modela, a naročito kod $G/M/1$ modela s normalnom razdiobom međudolaznih vremena. Prosječno vrijeme posluživanja nakon generiranja vremena posluživanja približno su jednaka za svaki od simuliranih modela, $T_s \approx 4\ [ms]$. Prosječan broj korisnika u redu L_w je najmanji kod $G/M/1$ simuliranog modela, a isti je slučaj kao i kod prosječnog broja korisnika u sustavu L_q. Razlog tomu je što kod prvog modela na te veličine utječe samo prometno opterećenje, a kod druga dva modela i ostali ulazni parametri.

Tablica 8. Izlazne veličine prve simulacije

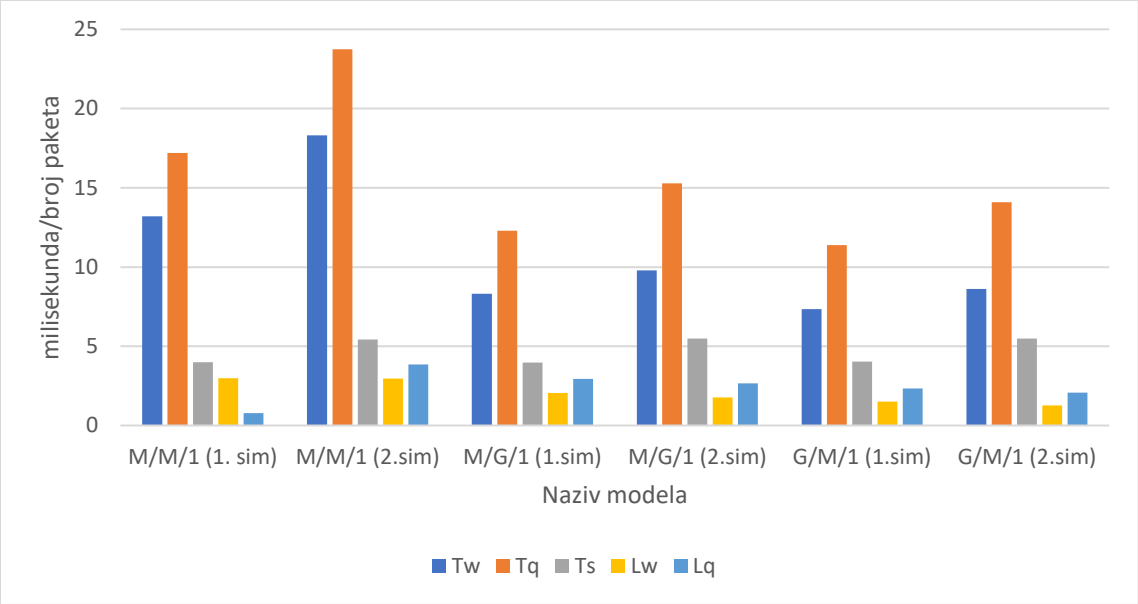
	$M/M/1$	$M/G/1$	$G/M/1$
$T_w[ms]$	13,2031	8,315	7,3374
$T_q[ms]$	17,1914	12,3012	11,3737
$T_s\ [ms]$	3,9883	3,9862	4,0362
$L_w[pak]$	2,9851	2,0534	1,5073
$L_q\ [pak]$	3,8799	2,9477	2,3363
$a\ [Erl]$	0,7870	0,7920	0,8090

Kao i kod tablice 8, u tablici 9 prikazane su empirijske vrijednosti izlaznih veličina druge simulacije. U drugoj simulaciji vrijeme posluživanja veće je nego u prvoj simulaciji što utječe na izlazne veličine te iznosi približno $T_s \approx 5{,}5\ [ms]$. Ponovno je vidljivo da su T_w i T_q najkraća kod nepoissonovih modela, a naročito kod $G/M/1$ modela s normalnom razdiobom međudolaznih vremena. Prosječan broj korisnika u redu L_w ponovno je najmanji kod $G/M/1$ simuliranog modela, a isti je slučaj kao i kod prosječnog broja korisnika u sustavu L_q. Također je vidljivo da su izlazne veličine poissonovskog modela upola veće od onih nepoissonovskih.

Tablica 9. Izlazne veličine druge simulacije

	$\boldsymbol{M/M/1}$	$\boldsymbol{M/G/1}$	$\boldsymbol{G/M/1}$
$\boldsymbol{T_w[ms]}$	18,3006	9,7932	8,613
$\boldsymbol{T_q[ms]}$	23,7326	15,288	14,0933
$\boldsymbol{T_s\ [ms]}$	5,4319	5,4948	5,4802
$\boldsymbol{L_w[pak]}$	2,964	1,7746	1,2679
$\boldsymbol{L_q\ [pak]}$	3,8515	2,6676	2,0731
$\boldsymbol{a\ [Erl]}$	0,7730	0,7869	0,7832

Također, napravljen je grafički prikaz izlaznih veličina radi usporedbe nepoissonovih i poissonovskih modela te kako bi se prikazale razlike među istima. Nastavno, prikazane su i razlike između prve i druge simulacije iz kojih se vidljivo određeno odstupanje u izlaznim veličinama kada se promijene ulazni parametri.



Grafikon 19. Usporedba ulaznih i izlaznih veličina prve i druge simulacije za analizirane modele

Može se uočiti kako su najsličniji rezultati oni između prve nepoissonove $M/G/1$ simulacije i druge $G/M/1$ simulacije Iz provedene analize može se zaključiti, da podvorbeni sustav u kome je proces dolazaka, odnosno međudolaznih vremena, normalno distribuiran, a vremena posluživanja eksponencijalno distribuirana ima najbolje performanse, posebice ako je vrijeme posluživanja kraće. Ovakve analize pomažu kod donošenja odluka o upravljanju ulaznim tokom i procesom posluživanja.

7. ZAKLJUČAK

Slučajni proces, poput dolaska korisnika na posluživanje u banci, teško je prikazati u nekom okruženju, a da pri tome ne izgubi prefiks „slučajan“. Upravo je takva priroda događaja opisana podvorbenim sustavima. Podvorbeni sustavi samo su mali segment onoga što se događa u stvarnosti. Dolazak korisnika u red, dolazak paketa na posluživanje u memoriji usmjerivača i sl. podvorbeni su procesi koji se svakodnevno odvijaju u pozadini realnog okruženja. Prema tome, podvorbeni sustavi primjenjuju se u svakodnevnim djelatnostima te se, kako bi se dobile matematičke izlazne veličine i odgovarajuće performanse realnog sustava, isti reproduciraju kroz odgovarajuće modele.

Ulazne veličine koje karakterizira slučajni proces simulirane su u ovom radu. Također uz pomoć tih simuliranih ulaznih veličina može se stvoriti okolina koja je potrebna za izračun izlaznih veličina. Izlazne veličine su mjerila izvedbe nekog podvorbenog sustava te daju predodžbu izvedbe nekog prometnog sustava.

Uz potrebno znanje o podvorbenim sustavima, kao i znanje nekog razvojnog programskog jezika, može se razmatrati kreiranje simulatora za isti. Kao razvojni jezik C# se ispostavio kao jednostavan, ali uz dublje korištenje se ispostavio kao iznimno napredno sredstvo za kreiranje simulatora. Također, biblioteke *ZedGraph* i *Accord.Net* su sa svojim mogućnostima kreiranja grafa i naprednih matematičkih funkcija pokazali kao odlično rješenje dijela kompleksnosti simulatora.

Simulator kreiran za potrebe rada služi iznimno za edukacijske svrhe te nije namijenjen za profesionalni rad. Međutim, daje jasnu sliku razlike između podvorbenih modela, od kojih se $G/M/1$ ispostavio kao optimalno rješenje za jednoposlužiteljske sustave. Također, veoma je primjetna razlika između poissonovskih i nepoissonovskih podvorbenih modela, gdje je razlika izlaznih veličina čak upola manja kod nepoissonovskih. Primjetno je kako su se povećanjem vremena posluživanja povećali iznosi izlaznih veličina.

POPIS KRATICA

FCFS – First Come – First Served

IDE - Integrated Development Environment

LCFS – Last Come – First Served

MLDV - Markovljev lanac s diskretnim skupom parametara

MLKV - Markovljev lanac s kontinuiranim skupom parametara

PRI – Service according to Priority

SDK - Software Development Kit

SIRO – Service In Random Order

SP RIU – Slučajni proces rađanja i umiranja

SP - Slučajni proces

SPT – Shortest Processing Time

SV - Slučajna varijabla

POPIS LITERATURE

[1] Begović M. *Podvorbeni sustavi.* Zagreb: Fakultet Prometnih Znanosti Sveučilišta u Zagrebu; 2006.

[2] Mrvelj Š. *Podsjetnik iz teorije vjerojatnosti.* [Prezentacija], Podvorbeni sustavi. Fakultet Prometnih Znanosti Sveučilišta u Zagrebu, 2019.

[3] Tutorials Point. *Discrete System Simulation*. Dostupno na: https://www.tutorialspoint.com/modelling_and_simulation/modelling_and_simulation_discrete_system_simulation.htm. [Pristupljeno: srpanj, 2020]

[4] Bucknell University. *Queue Behavior and Queue Discipline*. Dostupno na: https://www.eg.bucknell.edu/~xmeng/Course/CS6337/Note/master/node31.html. [Pristupljeno: srpanj, 2020]

[5] Mrvelj Š. *Matematički opis podvorbenog sustava.* [Prezentacija], Podvorbeni sustavi. Fakultet Prometnih Znanosti Sveučilišta u Zagrebu, 2019.

[6] Mrvelj Š. *Pojam slučajnog procesa.* [Prezentacija], Podvorbeni sustavi. Fakultet Prometnih Znanosti Sveučilišta u Zagrebu, 2019.

[7] Huzak M. *Vjerojatnost i matematička statistika.* Zagreb: Prirodoslovno Matematički Fakultet - Matematički odjel Sveučilišta u Zagreb; 2006.

[8] Buffet M, Caslin J, Patrick A, Boyle P, Porteus B, Tindale J, et al. *Stohastičko modeliranje.* Dostupno na: http://aktuari.math.pmf.unizg.hr/docs/sm.pdf. [Pristupljeno: srpanj 2020]

[9] Mrvelj Š. *Poissonov slučajni proces i slučajni proces rađanja i umiranja.* [Prezentacija], Podvorbeni sustavi. Fakultet Prometnih Znanosti Sveučilišta u Zagrebu, 2019.

[10] Striko K, Fratrović T, Ivanković B. *Vjerojatnost i statistika s primjerima iz tehnologije prometa.* Zagreb: Fakultet Prometnih Znanosti Sveučilišta u Zagrebu; 2008.

[11] Mrvelj Š. *Jednoposlužiteljski sustav.* [Prezentacija], Podvorbeni sustavi. Fakultet Prometnih Znanosti Sveučilišta u Zagrebu, 2019.

[12] Peraković D, Periša M. *Projektiranje IS-a.* [Prezentacija], Informacijski sustavi mrežnih operatora. Fakultet Prometnih Znanosti Sveučilišta u Zagrebu, 2018.

[13] Moltchanov DA. *G/M/1 Queueing system.* Dostupno na: http://www.cs.tut.fi/~moltchan/ELT-53607/lecture08.pdf. [Pristupljeno: lipanj 2020]

[14] Bhat UN. *An Introduction to Queueing Theory: Modeling and Analysis in Applications.* Boston: Birkhauser; 2008.

[15] Microsoft. *Introduction to the C# language and the.NET Framework*. Dostupno na: https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework. [Pristupljeno: kolovoz, 2020]

[16] Microsoft. *A tour of the C# language*. Dostupno na: https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/. [Pristupljeno: kolovoz, 2020]

[17] Microsoft. *Classes (C# Programming Guide)*. Dostupno na: https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/classes-and-structs/classes. [Pristupljeno: kolovoz, 2020]

[18] Microsoft. *StreamWriter Class*. Dostupno na: https://docs.microsoft.com/en-us/dotnet/api/system.io.streamwriter?view=net-5.0. [Pristupljeno: kolovoz, 2020]

[19] Accord.Net Framework. *Accord.Net*. Dostupno na: http://accord-framework.net/. [Pristupljeno: kolovoz, 2020]

POPIS KORIŠTENIH SLIKA

Slika 1. Prikaz jednoposlužiteljskog podvorbenog sustava *Izvor:* [3] 4
Slika 2. Prikaz ulaznog procesa, [1] 6
Slika 3. Prikaz Hello Word osnovnog programa 26
Slika 4. Prikaz instanciranja objekta *automobil* iz klase *Parkiraliste* 27
Slika 5. Primjer nasljeđivanja klasa u C# 27
Slika 6. Prikaz generiranja pseudonasumičnih skupova brojeva i zapisivanja u datoteku 28
Slika 7. Programski kôd za instanciranje eksponencijalne distribucije vremena dolazaka i posluživanja korisnika u $M/M/1$ modelu 29
Slika 8. Linije kôda za generiranje nepoissonovskih ulaznih parametara 30
Slika 9. Linije kôda za izračun vremena dolazaka i odlazaka 30
Slika 10. Linije kôda za izračun teoretskih izlaznih vrijednosti 31
Slika 11. Kôd za izračun empirijske vrijednosti duljine reda 32
Slika 12. Kôd za kreiranje grafova unutar simulatora 33
Slika 13. Sučelje simulatora jednoposlužiteljskih podvorbenih sustava 35
Slika 14. Prikaz generiranih ulaza i izračunatih izlaza simulatora za $M/M/1$ model 35
Slika 15. Prikaz generiranih izlaznih veličina zajedno s grafom vremena čekanja paketa 36
Slika 16. Prikaz simulatora s generiranim vrijednostima za $M/G/1$ 37

POPIS KORIŠTENIH GRAFIKONA

Grafikon 1. Graf funkcije gustoće vjerojatnosti za prvu simulaciju $M/M/1$ 39
Grafikon 2. Graf vremena čekanja za prvu simulaciju $M/M/1$ 40
Grafikon 3. Graf duljine reda za prvu simulaciju $M/M/1$ 40
Grafikon 4. Graf funkcije gustoće vjerojatnosti za drugu simulaciju $M/M/1$ 41
Grafikon 5. Graf vremena čekanja za drugu simulaciju $M/M/1$ 42
Grafikon 6. Graf duljine reda za drugu simulaciju $M/M/1$ 43
Grafikon 7. Graf funkcije gustoće vjerojatnosti za prvu simulaciju $M/G/1$ 44
Grafikon 8. Graf vremena čekanja za prvu simulaciju $M/G/1$ 45
Grafikon 9. Graf duljine reda prve simulacije $M/G/1$ 46
Grafikon 10. Funkcija gustoće vjerojatnosti druge simulacije $M/G/1$ 47
Grafikon 11. Vremena čekanja druge simulacije $M/G/1$ 48
Grafikon 12. Duljina reda za drugu simulaciju $M/G/1$ 48
Grafikon 13. Funkcija gustoće vjerojatnosti prve simulacije za $G/M/1$ 50
Grafikon 14. Vremena čekanja za prvu simulaciju $G/M/1$ 51

Grafikon 15. Duljina reda za prvu $G/M/1$ simulaciju 51
Grafikon 16. Graf funkcije gustoće vjerojatnosti za drugu simulaciju $G/M/1$ 52
Grafikon 17. Vremena čekanja druge simulacije $G/M/1$ 53
Grafikon 18. Prikaz duljine reda za drugu simulaciju $G/M/1$ 54
Grafikon 19. Usporedba ulaznih i izlaznih veličina prve i druge simulacije za analizirane modele 57

POPIS KORIŠTENIH TABLICA

Tablica 1. Prikaz izlaznih veličina 8
Tablica 2. Izlazne vrijednosti za prvu simulaciju $M/M/1$ 39
Tablica 3. Izlazne vrijednosti za drugu simulaciju $M/M/1$ 42
Tablica 4. Izlazne veličine prve simulacije $M/G/1$ 45
Tablica 5. Izlazne veličine druge simulacije $M/G/1$ 47
Tablica 6. Izlazne veličine prve simulacije $G/M/1$ 50
Tablica 7. Izlazne veličine druge $G/M/1$ simulacije 53
Tablica 8. Izlazne veličine prve simulacije 55
Tablica 9. Izlazne veličine druge simulacije 56

Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj diplomski rad isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu diplomskog rada pod naslovom **SIMULIRANJE JEDNOPOSLUŽITELJSKIH PODVORBENIH SUSTAVA**

na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

U Zagrebu, 15.9.2020

Student/ica:

Klaudija Marija Popić
(potpis)