# Uporaba metoda operacijskih istraživanja u uslužnom sektoru

**Protić, Tanja**

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

# DIPLOMSKI RAD

**Tanja Protić**

Zagreb, 2015. godina.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

# DIPLOMSKI RAD

Mentor: Doc. dr. sc. Hrvoje Cajner, dipl. ing.

Student: Tanja Protić

Zagreb, 2015. godina.

Izjavljujem da sam ovaj rad izradila samostalno koristeći stečena znanja tijekom studija i navedenu literaturu.

Zahvaljujem se svom mentoru doc. dr. sc. Hrvoju Cajneru koji mi je jako puno pomogao, bez njegove pomoći ovaj se rad nikada ne bi priveo kraju. Također se zahvaljujem svima koji su mi omogućili studiranje naročito mami, baki i sestri, i svima koji su me podupirali tokom studija.

Tanja Protić

SVEUČILIŠTE U ZAGREBU
**FAKULTET STROJARSTVA I BRODOGRADNJE**
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala i mehatronika i robotika

| Sveučilište u Zagrebu<br>Fakultet strojarstva i brodogradnje | |
|---|---|
| Datum | Prilog |
| Klasa: | |
| Ur.broj: | |

# DIPLOMSKI ZADATAK

Student: **TANJA PROTIĆ** Mat. br.:0035172968

Naslov rada na hrvatskom jeziku: **UPORABA METODA OPERACIJSKIH ISTRAŽIVANJA U USLUŽNOM SEKTORU**

Naslov rada na engleskom jeziku: **THE USE OF OPERATIONS RESEARCH METHODS IN SERVICE INDUSTRIES**

Opis zadatka:

Operacijska istraživanja su skup znanstvenih metoda koji se koristi pri donošenju odluka i implementaciji optimalnih rješenja u različitim područjima proizvodnje, javne uprave, vojske i uslužnih djelatnosti. Uporabom metoda matematičkog modeliranja, statističke analize, stohastičkih simulacija i matematičkog optimiranja, u okviru operacijskih istraživanja postižu se optimalna rješenja u kompleksnom postupku odlučivanja. S obzirom da postoji sve veća potreba kontinuiranog poboljšanja procesa, i to ne samo u proizvodnom sektoru, sve veća je zastupljenost navedenih metoda i u djelatnostima uslužnog sektora. Posebice se to očituje u području zdravstvenih usluga gdje se primjenom metoda operacijskih istraživanja postižu optimalna rješenja koja rezultiraju u boljoj usluzi, smanjenju troškova i povećanju konkurentnosti. Uzimajući spomenute činjenice u obzir u ovom radu potrebno je:

1. Dati presjek najčešćih metoda operacijskih istraživanja koje se javljaju pri optimizaciji procesa u uslužnim djelatnostima, a s posebnim naglaskom na zdravstveni sustav.
2. Detaljno objasniti teorijske postavke spomenutih metoda s konkretnim primjerima iz prakse (posebice obraditi problem optimizacije bolničkih sustava u svijetu).
3. Uzimajući u obzir specifične uvjete u RH obrazložiti mogućnosti primjene metoda u uslužnom sektoru RH.
4. Simulirajući različite uvjete i ograničenja, uz korištenje realnih podataka, primijeniti neke od metoda operacijskih istraživanja za pronalazak optimalnog rješenja.

| Zadatak zadan: | Rok predaje rada: | Predviđeni datum obrane: |
|---|---|---|
| 24. rujna 2015. | 26. studenog 2015. | 2., 3. i 4. prosinca 2015. |

Zadatak zadao:

Doc. dr. sc. Hrvoje Cajner

Predsjednik Povjerenstva:

Prof. dr. sc. Franjo Cajner

# SADRŽAJ

SADRŽAJ ..... I
POPIS SLIKA ..... III
POPIS TABLICA ..... IV
POPIS OZNAKA ..... **Error! Bookmark not defined.**
SAŽETAK ..... V
SUMMARY ..... VI
1. UVOD ..... 1
1.1. Opis problema ..... 1
1.2. Struktura rada ..... 2
2. PRESJEK NAJČEŠĆIH METODA OPERACIJSKIH ISTRAŽIVANJA U USLUŽNIM DJELATNOSTIMA ..... 3
2.1. Primjena OI u različitim uslužnim djelatnostima ..... 7
2.1.1. Djelatnosti informacija i komunikacija. ..... 7
2.1.1.1. Proizvodnja i distribucija informacija ..... 7
2.1.1.2. Prijenos informacija i komunikacija ..... 8
2.1.2. Transport i skladištenje ..... 11
2.1.3. Djelatnosti zdravstvene zaštite i socijalne skrbi ..... 12
3. OPERACIJKA ISTRAŽIVANJA ..... 13
4. METODE OPERACIJSKIH ISTRAŽIVANJA U USLUŽNIM DJELATNOSTIMA ..... 17
4.1. Nelinearno programiranje [2] ..... 17
4.2. Linearno programiranje [5] ..... 18
4.2.1. Opći oblik problema linearnog programiranja ..... 19
4.2.2. Simpleks metoda [5] ..... 23
4.2.2.1. Jednofazni postupak ..... 23
4.2.2.2. Dvofazni postupak ..... 26
4.2.3. Dual [5] ..... 29
4.2.4. Postoptimalna analiza[5] ..... 30
4.2.4.1. Analiza osjetljivosti[4] ..... 30
4.2.4.2. Parametarsko programiranje ..... 31
4.3. Višekriterijalno programiranje ..... 31
4.4. Dinamičko programiranje [1] ..... 32
4.5. Cjelobrojno programiranje [1] ..... 33
4.6. Transportni problem ..... 34
4.6.1. Metoda sjeverozapadnog kuta ..... 36
4.6.1.1. Primjer metode sjeverozapadnog kuta [1] ..... 37
4.6.2. Vogleova metoda [2] ..... 38
4.6.2.1. Primjer Vogelove metode ..... 39
4.6.3. Stepping-Stonova metoda [2] ..... 41
4.7. Teorija igara [4] ..... 42
4.7.1.1. Elementi strategijske igre ..... 43

4.7.1.2. Igre između dvije osobe sa sumom nula ........ 44
4.7.1.3. Igra sa sedlom ........ 44
4.7.1.4. Igre bez sedla ........ 45
4.7.1.5. Dominacija ........ 46
4.7.1.6. Općeniti poučci za matrične igre ........ 46
4.7.1.7. Igre protiv prirode ........ 48
4.8. Mrežno programiranje ........ 49
4.9. Teorija grafova ........ 49
4.10. Problem zaliha ........ 50
4.11. Modeli repova čekanja ........ 50
4.11.1. Proces rađanja i umiranja [3] ........ 52
4.11.1.1. Veza procesa rađanja – umiranja i eksponencijalne distribucije ........ 53
4.11.1.2. Izvođenje vjerojatnosti stabilnog stanja za proces rađanja i umiranja ........ 55
4.11.2. Poissonov proces [4] ........ 55
4.11.3. Kombinirani dolasci i usluge [4] ........ 57
4.11.4. Opći Poissonov model [4] ........ 58
4.11.4.1. Parametri stabilnog stanja ........ 60
4.11.5. Specijalni Poissonovi modeli repova [4] ........ 64
4.11.5.1. Model (M/M/1) : (GD/∞/∞) repa čekanja ........ 64
4.11.5.2. Model (M/M/1) : (GD/N/∞) repa čekanja ........ 66
4.11.5.3. Model (M/M/S) : (GD/∞/∞) repa čekanja ........ 67
4.11.5.4. Model (M/M/S) : (GD/N/∞) repa čekanja ........ 68
4.11.5.5. Model (M/M/∞) : (GD/∞/∞) repa čekanja ........ 69
4.11.5.6. Model (M/M/R) : (GD/K/K) repa čekanja ........ 70
4.12. Optimiranje rojem čestica ........ 71
4.13. Simulacije [3] ........ 73
4.13.1. Simulacija diskretnog događaja ........ 77
4.13.1.1. Razumijevanje osnovnih koncepata simulacije ........ 77
4.13.2. Slučajni brojevi i Monte Carlo simulacija ........ 83
5. PRIMJENA METODA OPERACIJSKIH ISTRAŽIVANJA ........ 85
5.1. Agencija za ugovaranje police osiguranja ........ 85
5.1.1. Opis zadatka ........ 85
5.1.2. Opis varijabli ........ 86
5.1.3. Opis rješenja zadatka ........ 86
5.1.4. Interpretacija rezultata ........ 88
5.2. Simulacija sustava ambulante ........ 90
5.2.1. Podatci o dolascima i odlascima ........ 90
6. ZAKLJUČAK ........ 95
LITERATURA ........ 96
PRILOZI ........ 97

# POPIS SLIKA

Slika 24. Analiza objavljenih rezultata OI prema kategorijama djelatnosti [Tablica 22] .......... 5
Slika 25. Odnos između objavljenih istraživanja u odnosu na uslužne djelatnosti (crveno) ..... 6
Slika 1. Proces modeliranja [2] ..................... 14
Slika 2. Podjela matematičkih modela ..................... 15
Slika 3. Metode operacijskih istraživanja s obzirom na prirodu problema ..................... 16
Slika 4. Podjela metoda nelinearnog programiranja [2] ..................... 18
Slika 5. Shema Gauss – Jordanove eliminacije[5] ..................... 25
Slika 6. Dijagram toka simpleks postupka [5] ..................... 26
Slika 7. Dijagram toka dvofaznog simpleks postupka [5] ..................... 28
Slika 9. Shema stopa rađanja i umiranja za sustav repa [3] ..................... 54
Slika 10. Prikaz sustava u stanju $k$ ..................... 56
Slika 11. Prikaz intervala $\sigma$ ..................... 57
Slika 12. Vjerojatnost stabilnog stanja povećanjem broja entiteta ..................... 63
Slika 17. Opća procedura heurističkog modela ..................... 72
Slika 18. Koraci prilikom izrade simulacije[3] ..................... 75
Slika 19. Osnovna terminologija u simulacijama i veze ..................... 76
Slika 20. Shema sustava [3] ..................... 79
Slika 21. Shema dolazaka [3] ..................... 79
Slika 22. Shema odlazaka [3] ..................... 80
Slika 23. Dijagram toka za simulacijski model reda čekanja s jednim uslužnim mjestom [3] 82
Slika 13. Shema slijeda događaja ..................... 90
Slika 14. Model ambulante ..................... 91
Slika 15. Broj pacijenata koji čekaju u redu ..................... 92
Slika 16. Prosječno vrijeme čekanja i vrijeme trajanja usluge ..................... 93

# POPIS TABLICA

Tablica 1. Podjela uslužnih djelatnosti prema nacionalnoj klasifikaciji djelatnosti [6] ............ 4
Tablica 2. Primjena OI u poslovanju ........ 9
Tablica 3. Primjena OI u radu informacijskih tehnologija ........ 10
Tablica 4. Primjena OI u transportu i skladištenju ........ 11
Tablica 5. Nastavak prethodne tablice ........ 12
Tablica 6. Operacijska istraživanja provedena u zdravstvu ........ 12
Tablica 7. zadani transportni problem ........ 37
Tablica 8. Početno rješenje TP-a riješeno metodom sjeverozapadnog kuta ........ 37
Tablica 9. Zadani transportni problem, ........ 39
Tablica 10. Prvi korak rješavanja zadanog transportnog problema ........ 39
Tablica 11. Drugi korak rješavanja zadanog transportnog problema ........ 39
Tablica 12. Treći korak rješavanja zadanog transportnog problema ........ 40
Tablica 13. Četvrti korak rješavanja zadanog transportnog problema ........ 40
Tablica 14. Početno rješenje TP-a riješeno Vogelovom metodom ........ 40
Tablica 15. zadani transportni problem ........ 41
Tablica 16. Prvo osnovno rješenje dobiveno metodom sjeverozapadnog kuta ........ 41
Tablica 17. Prvi korak rješavanja ........ 41
Tablica 18. Drugi korak rješavanja ........ 42
Tablica 19. Matrica plaćanja za igrača A ........ 44
Tablica 20. Matrica plaćanja igrača A ........ 45
Tablica 21. Matrica plaćanja ........ 46
Tablica 22. Primjeri sustava čekanja [4] ........ 50
Tablica 23. Izračun vjerojatnosti da u vremenu $t + \Delta t$ stanje $j$ *[3]* ........ 55
Tablica 24. Vrijednosti $p_n$ za pripadajući broj jedinica $n$ ........ 62
Tablica 25. Distribucija vremena dolazaka i pružanja usluge ........ 78
Tablica 26. Varijable koje opisuju atribute ........ 86
Tablica 27. Statistički podatci broja pacijenata u redu čekanja ........ 93
Tablica 28. Statistički podatci vremena čekanja ........ 94

# SAŽETAK

Tema ovog rada je "Uporaba metoda operacijskih istraživanja u uslužnom sektoru". Operacijska istraživanja su složeno interdisciplinarno područje koje se bavi problematikom odlučivanja u realnim uvjetima uzimajući u obzir sve faktore koji na problem djeluju direktno ili indirektno, kako bi se pronašlo najbolje odnosno, optimalno rješenje. S obzirom da postoji sve veća potreba kontinuiranog poboljšanja procesa, i to ne samo u proizvodnom sektoru, sve veća je zastupljenost metoda operacijskih istraživanja i u djelatnostima uslužnog sektora. Posebice se to očituje u području zdravstvenih usluga gdje se na taj način postižu optimalna rješenja koja rezultiraju u boljoj usluzi, smanjenju troškova i povećanju konkurentnosti.

Osnovna ideja rada bila je istražiti i obrazložiti mogućnosti primjene metoda operacijskih istraživanja u uslužnom sektoru Republike Hrvatske, uzimajući u obzir specifične uvjete našeg podneblja. U teorijskom dijelu rada su definirani osnovni pojmovi, definicije i izrazi u operacijskim istraživanjima. Zatim, detaljno su objašnjene sve metode operacijskih istraživanja kako bi se kasnije mogla odabrati metoda koja je najprikladnija za rješavanje problema. Teorijski dio potkrijepljen je s dva konkretna primjera iz prakse. Istraživački dio rada bazira se na pronalaženju optimalnog rješenja za realne podatke zadanog problema, simulirajući različite uvjete i ograničenja. Na osnovu dobivenih rezultata i njihove analize, doneseni su odgovarajući zaključci o tome da li je problem uspješno riješen.

*Ključne riječi:*

operacijska istraživanja, metode, optimalno rješenje, uslužni sektor

## SUMMARY

This paper is about the "The use of operations research methods in service industries". Operations research are complex interdisciplinary field that deals with the issue of decisions in real terms, taking account all the factors that affect the issue, directly or indirectly, in order to find the best or optimal solution. There is an increasing need of continuous improve processes, and not just in the manufacturing sector, but also increasing the use of operations research methods in service industries. Especially, it is reflected in the health service where thus achieve optimal solutions result with a better service, reduce costs and increase competitiveness.

The basic idea of the study was to explore and explain the use of operations research methods in service industries considering the specific conditions in Croatia. The theoretical part defines the basic concepts, definitions and terms used in operations research. Next, there are detailed explanations of all operations research methods in order to choose the most appropriate method to solve the problem. The theoretical part is supported by two concrete examples from practice. The research part of the work is based on the optimal solution for a given problem with real data, simulating various conditions and restrictions. Based on the results and their analysis, there have been adopted appropriate conclusions on whether the problem is successfully solved.

*Keywords:*

operations research, methods, optimal solution, service sector

# 1. UVOD

Operacijska istraživanja su složeno interdisciplinarno područje koje se bavi problematikom odlučivanja u realnim uvjetima uzimajući u obzir sve faktore, koji na problem djeluju direktno ili indirektno kako bi se pronašlo optimalno rješenje. Dakle, radi se o skupu znanstvenih metoda koje se koriste pri donošenju odluka i implementaciji optimalnih rješenja u različitim područjima proizvodnje, javne uprave, vojske i uslužnih djelatnosti.

Zbog velikog broja varijabli koji utječu na neki problem nužna je računalna podrška za rješavanje problema. Interdisciplinarnost i korištenje programskih rješenja glavni je razlog odabira ove teme.

Problem koji će se riješiti u ovom radu odnosi se na primjenu operacijskih istraživanja u uslužnom sektoru, s posebnim naglaskom na zdravstveni sustav.

S obzirom da postoji sve veća potreba kontinuiranog poboljšanja procesa, i to ne samo u proizvodnom sektoru, sve veća je zastupljenost metoda operacijskih istraživanja i u djelatnostima uslužnog sektora. Posebice se to očituje u području zdravstvenih usluga gdje se na taj način postižu optimalna rješenja koja rezultiraju u boljoj usluzi, smanjenju troškova i povećanju konkurentnosti.

## 1.1. Opis problema

Uvijek kada je riječ o uslužnom sektoru, može se naslutiti da će na modeliranje rješenja veliki utjecaj imati ljudi budući da igraju glavnu ulogu, kao ulazni i izlazni faktori. Čekanje je najčešći problem u uslužnim djelatnostima. Tako se i u zdravstvenom sustavu javlja čekanje, problemi sa smještanjem pacijenata, problem komunikacije između radnika unutar zdravstvenog sustava, problem troškova. Glavni faktori unutar zdravstvenog sustava su pacijenti (primatelji usluge), osoblje u zdravstvenom sustavu (davatelji usluge), davatelji financijske potpore i javno zdravstvo. Sve druge veze i svi drugi faktori proizlaze iz njihove interakcije. Kako bismo mogli riješiti probleme koji se javljaju u zdravstvenom sustavu potrebno je razumjeti interakciju između gore navedena četiri faktora.

Stanje pacijenata, budućih ili sadašnjih proizlazi iz različitih faktora, pa tako npr. socijalni faktor može imati veliki utjecaj na zdravlje čovjeka. Fizička i socijalna okolina u kojoj ljudi žive i rade također imaju veliki utjecaj na njihovo zdravlje (uvjeti življenja, mogućnost školovanja, pristup prijevozu). Zatim, uvjeti unutar zajednice poput statusa izbjeglice, strah od deportiranja, diskriminacija. Sve to utječe na zdravlje pojedinca i na mogućnost adekvatne zdravstvene skrbi, a to je samo socijalni aspekt problema. Iz ovoga već možemo vidjeti dio interakcija koje proizlaze iz faktora pacijenta.

Osoblje unutar zdravstvenog sustava, u ovu skupinu spadaju medicinske sestre, farmaceuti, liječnici, terapeuti, psiholozi, socijalni radnici itd., svi oni igraju bitnu ulogu unutar zdravstvenog sustava. Trebamo se pitati radi li osoblje ono što bi moglo raditi da ima zadovoljene optimalne uvjete rada. Iz navedenog proizlaze sljedeća pitanja:

1. Imamo li dovoljno kvalificirane radne snage unutar zdravstvenog sustava?
2. Nalaze li se u područjima u kojima su populaciji potrebni?
3. Jesu li adekvatno plaćeni za usluge koje pružaju?

Financijska potpora osigurava plaćanje usluge, amortizaciju i unaprjeđenje zdravstvenog sustava novim instrumentima i tehnologijama kako bi se sustav unaprijedio pogotovo unutar gradova u kojima živi velik broj ljudi. Javno zdravstvo kao takvo pruža osnovno zdravstveno osiguranje.

Za rješavanje opisanog problema, potrebno je najprije napraviti presjek najčešćih metoda operacijskih istraživanja koje se javljaju pri optimiranju procesa u uslužnim djelatnostima, s posebnim naglaskom na zdravstveni sustav.

## 1.2. Struktura rada

U prvom dijelu rada dat će se teorijska podloga najčešćih metoda operacijskih istraživanja. Detaljno će biti objašnjene sve metode kako bi se kasnije mogla odabrati metoda koja je najprikladnija za rješavanje problema. Teorijske pretpostavke će se dati na nekim primjerima iz prakse s posebnim naglaskom na optimizaciju bolničkih sustava.

Drugi dio će se sastojati od analize problema koji se javljaju u Hrvatskoj i mogućnosti primjene nekih od rješenja. U zadnjem dijelu simulirat će se različiti uvjeti i ograničenja koristeći realne podatke te primijeniti neke od metoda operacijskih istraživanja za pronalazak optimalnog rješenja. Na kraju će se dati zaključak i interpretacija rezultata iz kojih će se vidjeti da li je problem uspješno riješen.

# 2. PRESJEK NAJČEŠĆIH METODA OPERACIJSKIH ISTRAŽIVANJA U USLUŽNIM DJELATNOSTIMA

U ovom poglavlju dat će se presjek najčešćih metoda OI u uslužnom sektoru i uzimajući u obzir specifične uvjete u RH i probleme koji se javljaju predložiti gdje bi se OI mogla primijeniti.

Udio BDP-a od uslužnih djelatnosti odražava se na konkurentnost države. Uslužne djelatnosti čine oko dvije trećine BDP-a u Hrvatskoj, a od toga najvažnija je turizam koji zauzima 15% udjela BDP-a.[1] Stalne inovacije i nove tehnologije od ključne su važnosti za opstanak i napredak uslužnih djelatnosti.

Za primjenu operacijskih istraživanja u uslužnim djelatnostima potrebno je proučiti već postojeće modele koji se koriste u proizvodnim djelatnostima i pronaći moguće primjene u uslužnim djelatnostima. S obzirom na razlike između uslužnih i proizvodnih djelatnosti trebaju se uzeti u obzir i ograničenja već postojećih metoda operacijskih istraživanja.

Budući da je sektor uslužnih djelatnosti jako kompleksno područje pitanje je mogu li operacijska istraživanja pomoći u rješavanju realnih problema u uslužnim djelatnostima, odnosno može li se povećati njihova učinkovitost kao što je to učinjeno u proizvodnim djelatnostima.

Postoje sličnosti između proizvodnih i uslužnih djelatnosti na primjer u planiranju projekata i organizaciji. Što znači da se modeli i teorije razvijene za proizvodnju mogu primijeniti u uslužnim djelatnostima. Generalno, proizvodne i uslužne djelatnosti čine dva slična sustava koja korištenjem raznih sredstva u nekim procesima proizvode opipljive proizvode ili nematerijalne usluge za korisnike. A razlike između tih sustava povezane su s različitim obilježjima u aktivnostima koje se provode kako bi se zadovoljio krajnji korisnik.

Postoje tri glavne razlike između sustava proizvodnih i uslužnih djelatnosti koje su bitne za primjenu OI u uslužnim djelatnostima:

1. Resursi nisu konstantni
2. Neposredan proces pružanja
3. Logistički procesi

[1] Podatak preuzet sa stranice: http://www.croatia.eu/article.php?lang=1&id=32

Resursi kod uslužnih djelatnosti nisu konstantni. Za razliku od proizvodnih djelatnosti gdje je broj resursa uglavnom određen (npr. broj strojeva i alata), dok kod uslužnih djelatnosti, ne samo da taj broj varira nego i potreba za određenim resursima varira tijekom iste aktivnosti.

Proces „proizvodnje“ je kod uslužnih djelatnosti neposredan. Kod proizvodnih djelatnosti postoji određen vremenski razmak između trenutka završetka proizvoda i trenutka dostavljanja proizvoda korisniku. Kod uslužnih djelatnosti „proizvodnja“ usluge se događa u istom vremenskom periodu u kojem se i dostavlja korisniku. Što dovodi do kompleksnijih međuodnosa u uslužnim djelatnostima, interakcijom više različitih modela dolazi se do objekata istraživanja.

Treća glavna razlika su logistički procesi, u proizvodnim djelatnostima postoji limitiran broj kategorija, a usluge su nematerijalne i smamim time različite prirode. Zbog velikog broja različitih oblika usluga proces dostavljanja odnosno interakcije s korisnicima značajno varira s obzirom na vrstu usluge.

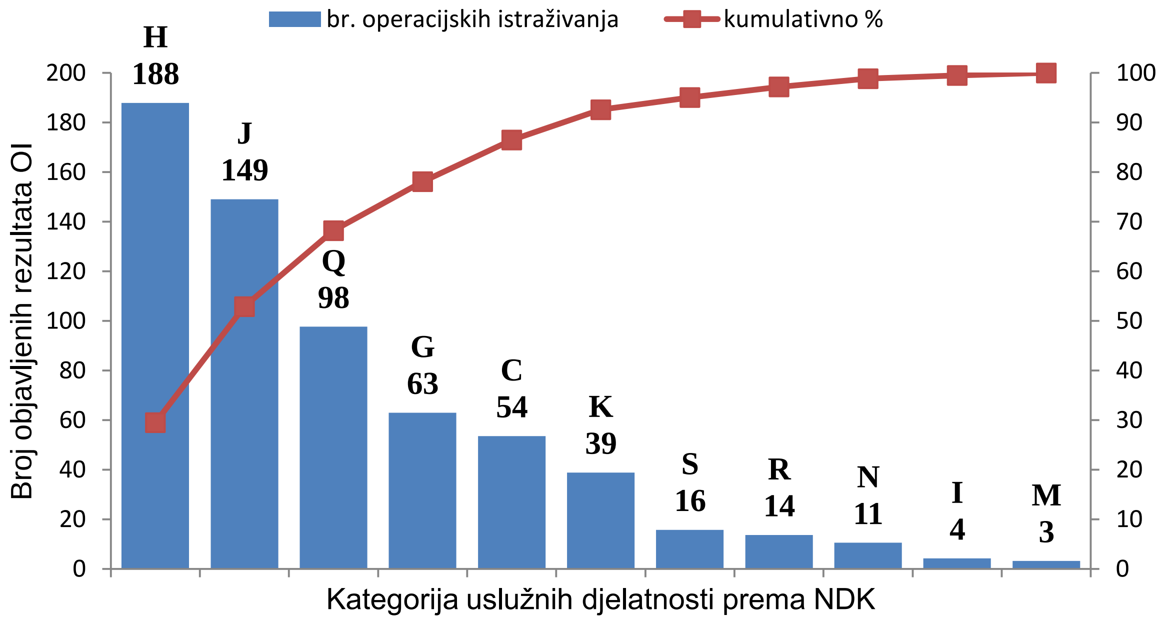
Raznolikost i kompleksnost glavne su karakteristike procesa uslužnih djelatnosti. Zbog toga je teško univerzalno kategorizirati metode OI. Budući da se fokus nalazi na primjeni metoda, napravit će se podjela prema područjima primjene. Sektor uslužnih djelatnosti prema državnom zavodu za statistiku obuhvaća 21 djelatnost, podjela je dana u tablici [Tablica 1].

Tablica 1. Podjela uslužnih djelatnosti prema nacionalnoj klasifikaciji djelatnosti [6]

| Oznake i nazivi područja NKD-a 2007. | | Broj hijerarhijskih razina NKD-a 2007. | | |
|---|---|---|---|---|
| | | odjeljci | skupine | razredi |
| UKUPNO (A-U) | | 88 | 272 | 615 |
| A | Poljoprivreda, šumarstvo i ribarstvo | 3 | 13 | 39 |
| B | Rudarstvo i vađenje | 5 | 10 | 15 |
| C | Prerađivačka industrija | 24 | 95 | 230 |
| D | Opskrba električnom energijom, plinom, parom i klimatizacija | 1 | 3 | 8 |
| E | Opskrba vodom; uklanjanje otpadnih voda, gospodarenje otpadom te djelatnosti sanacije okoliša | 4 | 6 | 9 |
| F | Građevinarstvo | 3 | 9 | 22 |
| G | Trgovina na veliko i na malo; popravak motornih vozila i motocikla | 3 | 21 | 91 |
| H | Prijevoz i skladištenje | 5 | 15 | 23 |
| I | Djelatnosti pružanja smještaja te pripreme i usluživanja hrane | 2 | 7 | 8 |

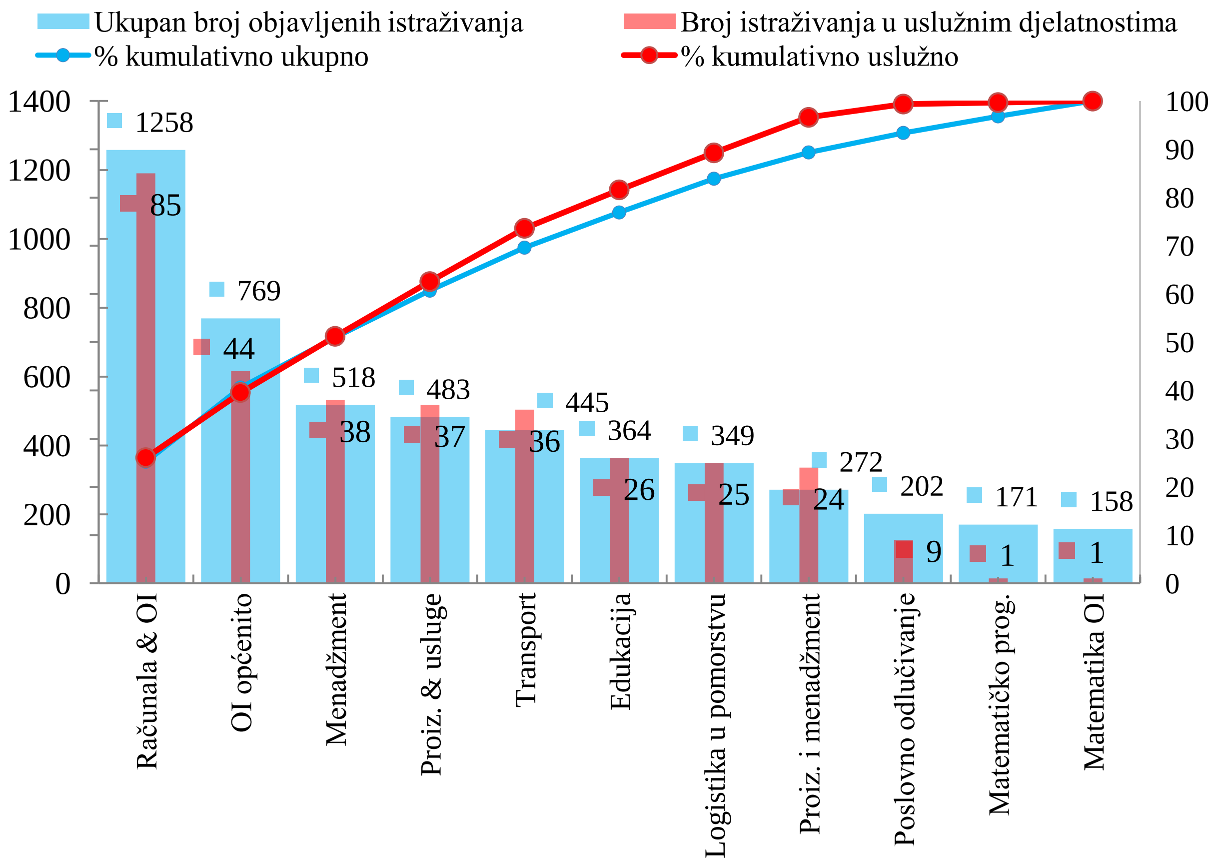
| Oznake i nazivi područja NKD-a 2007. | | Broj hijerarhijskih razina NKD-a 2007. | | |
|---|---|---|---|---|
| J | Informacije i komunikacije | 6 | 13 | 26 |
| K | Financijske djelatnosti i djelatnosti osiguranja | 3 | 10 | 18 |
| L | Poslovanje nekretninama | 1 | 3 | 4 |
| M | Stručne, znanstvene i tehničke djelatnosti | 7 | 15 | 19 |
| N | Administrativne i pomoćne uslužne djelatnosti | 6 | 19 | 33 |
| O | Javna uprava i obrana; obvezno socijalno osiguranje | 1 | 3 | 9 |
| P | Obrazovanje | 1 | 6 | 11 |
| Q | Djelatnosti zdravstvene zaštite i socijalne skrbi | 3 | 9 | 12 |
| R | Umjetnost, zabava i rekreacija | 4 | 5 | 15 |
| S | Ostale uslužne djelatnosti | 3 | 6 | 19 |
| T | Djelatnosti kućanstava kao poslodavaca; djelatnosti kućanstava koja proizvode različitu robu i pružaju različite usluge za vlastite potrebe | 2 | 3 | 3 |
| U | Djelatnosti izvanteritorijalnih organizacija i tijela | 1 | 1 | 1 |

U dijagramu [Slika 1] može se vidjeti da najveći broj istraživanja, 50% ukupnih istraživanja odnosi se na prijevoz i skladištenje te informacije i komunikacije.




Slika 1. Analiza objavljenih rezultata OI prema kategorijama djelatnosti [Tablica 1]

Kada se pogleda odnos broja istraživanja u uslužnim djelatnostima u usporedbi s drugim područjima [Slika 2] jasno je vidljivo koliko je sektor uslužnih djelatnosti neistražen. A s obzirom da uslužni sektor ima ogroman utjecaj na porast BDP-a, bilo bi dobro neke od metoda OI primijeniti u uslužnom sektoru kako bi se osigurala konkurentnost.




Slika 2. Odnos između objavljenih istraživanja u odnosu na uslužne djelatnosti (crveno)

Na temelju tih podataka od nabrojanih djelatnosti [Tablica 1] OI su se u prošlosti najčešće koristila u sljedećim kategorijama:

1. transport i skladištenje;
2. informacije i komunikacije;
3. djelatnosti zdravstvene zaštite i socijalne skrbi;
4. trgovini na veliko i na malo, popravak motornih vozila i motocikla;
5. financijske djelatnosti i djelatnosti osiguranja.

Unutar svake od pet gore navedenih kategorija, operacijsko istraživanje će se razmotriti sa stanovišta predmeta istraživanja, modela te metoda i tehnika OI.

## 2.1. Primjena OI u različitim uslužnim djelatnostima

Fokus rada odnosi se na aspekt primjene OI u uslužnom sektoru. U ovom dijelu razmatraju se predmeti istraživanja i metodologije koje su se pokazale kao korisne u raznim područjima uslužnog sektora, u trećem se poglavlju teorijske postavke nekih metoda detaljno objasniti.

Najjednostavnije je napravit podjelu prema predmetu istraživanja, odnosno prema djelatnostima koje su nabrojane u uvodu poglavlja [Tablica 1]. Proučit će se koncept modeliranja i rješavanja različitih problema kako bi se pružio uvid u obilježja koja utječu na razvijanje područja OI u uslužnom sektoru.

### *2.1.1. Djelatnosti informacija i komunikacija.*

Razvojem kapaciteta prerade informacija i širenjem interneta jasni su nam pojmovi informacijskog doba i informacijske ekonomije. Brzo su se pojavile razne vrste informacijskih djelatnosti, koje između ostalog uključuju pružanje internet usluge, usluge obrade podataka, usluge ciljanog oglašavanja na temelju informacijskih podataka iz prošlosti. Kao što se može pretpostaviti ova kategorija djelatnosti je najbrže rastuća u gotovo svim državama svijeta, pa tako i u Hrvatskoj. Logično je da se zbog brzog rasta ove grane zahtjeva najbolji mogući rad sustava kako bi se osigurala konkurentnost.

Sustav djelatnosti informacija i komunikacija predstavlja jako široko područje koje se može podijeliti na djelatnosti koje uključuju proces:

1. proizvodnje i distribucije informacija i kulturnih proizvoda,
2. pružanja sredstava za prijenos ili distribuciju prethodno navedenih proizvoda, podataka ili komunikacije,
3. obrade podataka.

Od navedene tri kategorije najčešća primjena OI nalazi se u prijenosu informacija i komunikaciji, no i druga područja mogu poslužiti za istraživanje budući da se i tu pronalaze slučajevi primjene OI.

#### 2.1.1.1. *Proizvodnja i distribucija informacija*

Primjena OI u djelatnostima proizvodnje i distribucije informacija:

- optimalan pristup donošenju odluka u razvoju softverskih paketa,

- teorija igara za istraživanje kratkoročne i dugoročne konkurentnost jednokratne prodaje softvera i prodaje softvera kao usluge,
- analiza omjera sive za izradu modela procjene napora razvoja softvera,
- modeli za optimiziranje vremena izdavanja softvera uz pretpostavku nesavršenog procesa otklanjanja grešaka softvera i kazne za zakašnjelo izdavanje softvera,
- ispitivanje optimalnih kontrolnih odluka što se tiče cijene, veličine mreže i strategije zapošljavanja u kontekstu razvoja besplatnog softvera,
- određivanje maksimalnog profita s obzirom na tipove korisnika, te broj i cijenu svake verzije softvera,
- minimiziranje troška računalnog piratstva u ovisnosti o novim verzijama softvera,
- teorija igara između isporučitelja softvera i poduzeća koje je koristilo softver, za analizu efekta različitih radnji, poduzeće želi minimizirati ukupne troškove odabirom javno dostupnog ažuriranja ili nove verzije softvera, isporučitelj analizira što će se dogoditi i pokušava izvući maksimalan profit,
- određivanje cijene informacijskih paketa nelinearnim programiranjem mješovitim brojevima uz maksimiziranje profita,
- maksimiziranjem profita odabirom najbolje dugoročne strategije cijene distribucije informacijskih proizvoda, proizvedenih uz jako male troškove kopiranjem originala.

#### 2.1.1.2. *Prijenos informacija i komunikacija*

Informacije se prenose različitim kanalima, na primjer radio prijenos, TV emitiranje, bežičnom i žičanom mrežom, satelitskom mrežom. Na temelju toga ova se kategorija djelatnosti može podijeliti na potkategorije djelatnosti:

1. TV i radio emitiranje,
2. žičane telekomunikacije,
3. bežične telekomunikacije,
4. telekomunikacija u općem smislu.

Istraživanja u tim djelatnostima mogu podijeliti s obzirom na probleme:

1. u poslovanju [Tablica 2],
2. u radu informacijskih tehnologija [Tablica 3].

Tablica 2. Primjena OI u poslovanju

| | **Primjena** | **Metode i matematički modeli** |
|---|---|---|
| 1. | Raspored emitiranja promidžbenih programa | Cjelobrojno programiranje i heuristički algoritam |
| 2. | Određivanje cijene naknade emitiranja | LP, maksimiziranje profita |
| | | Matematički modeli |
| | | Teorija igara |
| | | Stohastički modeli optimizacije |
| | | Iterativne gradijentne metode |
| | Odabir vremena emitiranja određenog sadržaja | Dinamičko programiranje |
| | | Markovljevi modeli |
| 3. | Više - pristupne arhitekture Internet poslužitelja | Gradijentne metode minimiziranje funkcije troška |
| | Kooperacije internet poslužitelja i poslužitelja glavnih podatkovnih linija | Teorije repova čekanja |
| | | Teorije igara |
| | | Optimizacije troškova |
| | Naplate usluge | Teorija igara (maksimiziranje profita) |
| | | LP (klasično optimiranje) |
| | | Simpleks |
| | | Aproksimativne gradijentne metode, |
| | | M/M/∞ model repova za optimiranje cijene |
| | Upravljanja prihodima | Markovljevi modeli |
| | Planiranja oglasa | Cjelobrojno programiranje |
| | | Hibridni genetski algoritmi (GA) |
| | Ugovora outsourceing-a pozivnih centara | Teorija igara unutar repova čekanja |
| | Sustava usmjeravanja poziva | M/M/N model teorija repova |
| | | Markovljevi modeli donošenja odluka |
| | Zapošljavanja pozivnih agenata | M/M/N modeli repova |
| | | Asimptotska aproksimacija |
| | | Iterativni algoritmi za nehomogena Poissonova pristizanja |
| | | Markovljevi lanci |
| | | M/M/N modeli repova čekanja |
| | Analize konstrukcije modela pozivnog centra | Markovljevi lanci u 2D repovima čekanja |
| | | Modeli repova čekanja |
| 4. | Optimalnog planiranja radne snage | Iterativne metode procjene |
| | | Algoritmi učenja klasifikacije |
| | | Klasifikacija |

Tablica 3. Primjena OI u radu informacijskih tehnologija

| | Primjena | Metode i matematički modeli |
|---|---|---|
| 1. | Optimiranje mrežnih paketa pri zemaljskom emitiranju | NLP-teški problemi, iterativno rješavanje |
| | | Heuristička metoda (2 stupnja rješavanja) |
| | Dodjeljivanje frekvencije s obzirom na ograničen raspon i interferenciju s drugim izvorima | Heuristička metoda problem velikih razmjera |
| | | Genetski algoritmi (GA) i analitička optimiranja |
| | Lokacija i konfiguracija bazne postaje<br><br>cilj: maksimalna pokrivenost, uz minimalne troškove instalacije | Mješovito cjelobrojno programiranje (MCP) |
| | | Optimiranje lančanim LP |
| | | Heuristički algoritmi |
| | | Stohastički algoritmi optimiranja |
| | | Matematički modeli simulacije |
| | | Cjelobrojno programiranje (CP) |
| | | Mrežno programiranje (MP), pohlepni algoritmi |
| | | Deterministički modeli |
| | | Iterativne gradijentne metode |
| | Razni mrežnih problema (npr. izdržljivost, brzina, kapacitet) | Metode traženja maksimuma |
| | | LP unutar modela aproksimacije |
| | | M/M/N model teorije repova čekanja |
| | Raspodjele sredstava među korisnicima | MCLP unutar heurističkog algoritma |
| | | CNLP – problem duala |
| | | LP unutar modela aproksimacije |
| | | Aproksimacija dinamičkim programiranjem (DN) |
| | | (M/M/PP) : (/G/1/K) model repa čekanja |
| | Sheme rada mreže | PERT, CPM |
| | | Markovljevi redovi (konačan broj spremnika) |
| | Jakosti prijenosa | Pohlepni algoritmi |
| | | Optimiranje rojem čestica (Depth-First-Search) |
| | Planiranja paketa podataka | MCNLP minimalan trošak |
| | | Genetski algoritmi optimiranja topologije mreže |
| | | Samoorganizirajuće neuronske mreže (SOM) |
| 2. | Mrežnog pristupa telefonske tvrtke na velikim udaljenostima | Cjelobrojno programiranje, optimiranje troška |
| | Konstruiranja specifičnih mehanizama internet usluga | Klasifikacija (Sigmond funkcija) |
| | | NLP, područje povjerenja, *n*-D problemi |
| | | Klasifikacija iterativnim srednjim vrijednostima |
| | | GI/M/1 i M/G/1 modeli repova čekanja |
| | | Metoda Lagrangeovih multiplikatora |
| | | Neuronske mreže, generički algoritmi |
| | | Transportni problem, PERT |
| 4. | Konstruiranja mreže (optimalni ulaz) | Jednostavno LP (postoji optimalno rješenje) |

### 2.1.2. Transport i skladištenje

U ovoj kategoriji razmotrit će se operacijska istraživanja primijenjena u transportu i skladištenju. Ovdje spadaju tipične aktivnosti poput transporta putnika i tereta, pružanje usluga parkiranja, usluge terminala, kargo, upravljanje i skladištenje teretom, poštanske i kurirske usluge. Transport se odvija zračnim, kopnenim, pomorskim i željezničkim prometom. Operacijska istraživanja u ovom području se klasificiraju na putnički prijevoz i prijevoz tereta. Za prijevoz putnika najviše pozornosti privlače zrakoplovna industrija i javni prijevoz. Prijevoz tereta se razlaže na terminalne transportne usluge brodovima te poštanske/kurirske usluge.

Dakle, transport se može podijeliti u 3 glavne kategorije:

1. zrakoplovna industrija,
2. javni prijevoz,
3. djelatnosti logistike i transporta.

U sljedećoj tablici nalaze se primjeri primjene operacijskih istraživanja za svaku od navedenih kategorija [Tablica 4].

Tablica 4. Primjena OI u transportu i skladištenju

| | **Primjena** | **Metode i matematički modeli** |
|---|---|---|
| 1. | Raspored letova | Maksimiziranje |
| | | Metode donošenja odluka |
| | | Generativni algoritmi |
| | | Mješovito cjelobrojno NLP |
| | Planiranje piste | Heuristički pristup rješavanju problema |
| | Planiranje flote | Kombinacija programiranja s ograničenjima |
| | | Deterministički modeli |
| | | Heuristički pristup rješavanju problema |
| | | Stohastički algoritmi optimiranja |
| | Planiranje posade | Neuronske mreže, generički algoritmi |
| | | Matematički modeli simulacije |
| | | Stohastički algoritmi optimiranja |
| | Izvedbe terminala | Mješovito cjelobrojno linearno programiranje |
| | | Pareto simuliranje žarenja |
| | | Razni matematički modeli |
| | Upravljanje prihodima | Markovljevi redovi (konačan broj obroka) |
| | | Data Envelopment Analysis (DEA). |

Tablica 5. Primjena OI u transportu i skladištenju

| | **Primjena** | **Metode i matematički modeli** |
|---|---|---|
| 2. | Raspoređivanje vozila | Linearni model mješovitih/cijelih brojeva |
| | | Simulacija formulacije linearnog programa |
| | | Hibridni genetski algoritam |
| | | D/C/C model teorije repova čekanja |
| | | Tehnike matematičkog programiranja |
| | Raspoređivanje vozača | Cjelobrojno linearno programiranje |
| | | Iterativni heuristički pristup rješavanju problema |
| | | Metoda kaznenih funkcija |
| | Usmjeravanje – izrada ruta za vozila | Genetski algoritam |
| | | Višekriterijalno optimiranje |
| | | Evolucijski algoritam s lokalnim ekstremima |
| | Konstrukcija mreže | Metoda uzastopnih presjeka |
| | | Algoritam za mrežu javnog prijevoza |
| | | Mješovito cjelobrojno linearno programiranje |
| | | Multikriterijalna analiza odluka |
| | | Model protoka |
| | | Simulacijski model diskretnih događaja |

### *2.1.3. Djelatnosti zdravstvene zaštite i socijalne skrbi*

Gotovo u svim zemljama na svijetu najveći problem kod djelatnosti zdravstvene zaštite je nedostatak resursa zbog velike potražnje od strane građana. Stoga je naglasak OI u ovom sustavu optimalno iskorištenje resursa u svrhu poboljšanja učinkovitosti i učinka sustava. Generalno gledajući OI su vitalni alat u planiranju i donošenju odluka raznih aktivnosti, planiranje i organiziranje na višoj i nižoj razini.

Tablica 6. Operacijska istraživanja provedena u zdravstvu

| Primjena | Metode i matematički modeli |
|---|---|
| Cijene zdravstvenog osiguranja<br>optimalna njega<br>optimalna procjena<br>Raspodjela resursa (vozila, ambulante, bolnice)<br>Organiziranje naručivanja pacijenata<br>Planiranje operacije<br>Planiranje vozila<br>Planiranje kapaciteta<br>Menadžment troškova<br>Menadžment zaliha<br>Organiziranje naručivanja pacijenata<br>Planiranje operacije<br>Planiranje vozila | Neparametarske metode<br>Stohastički algoritmi<br>Metode traženja maksimuma i minimuma<br>Metode repova čekanja<br>Metode procjene na temelju učenja<br>Matematički modeli zaliha<br>Dinamičko programiranje lokacije<br>Heuristički algoritmi i Lagrangeovi multiplikatori<br>Multivarijatne metode<br>Binarno cjelobrojno programiranje<br>Metode repova čekanja<br>Metode zaliha |

# 3. OPERACIJKA ISTRAŽIVANJA

Operacijska istraživanja (OI) jesu stručna i znanstvena disciplina koja se bavi pomaganjem kod donošenja odluka na bazi egzaktnih metoda. [1]

Kako je ranije bilo spomenuto, OI nastoje odrediti najbolji, odnosno optimalni smjer aktivnosti u problemu odlučivanja u okviru danih restrikcija i ograničenih kapaciteta.

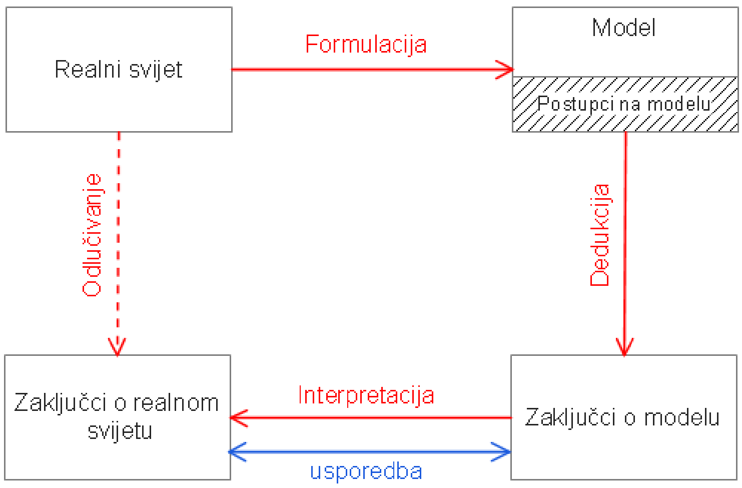
Da bi se mogao analizirati sustav iz realnog svijeta, točnije da bi se mogao riješiti neki problem, nužno je izraditi matematički model sustava. Pri izradi modela na umu treba imati da su veze u realnom svijetu nelinearne i nepredvidive te da podatci nisu apsolutno točni. Prije izrade modela potrebno je procijeniti ili provjeriti točnost podataka kako bi se mogle pojednostavniti veze među njima.

Kada se operacijska istraživanja koriste za rješavanje problema u uslužnim i proizvodnim djelatnostima, treba slijediti određenu proceduru. Prvi korak, definiranje problema obuhvaća određivanje ciljeva i područja koja je potrebno proučiti prije nego li se problem može riješiti. Kada je cilj jasno definiran, počinje promatranje sustava, definiranje parametara koji otežavaju ostvarenje cilja, odnosno prikupljanje podataka za procjenu vrijednosti parametara koji utječu na probleme organizacije. Procijenjeni parametri se koriste za razvoj i vrednovanje matematičkog modela. Formulacija modela obuhvaća razvoj novog modela ili primjenu postojećih modela na definirani problem. Potrebno je ustanoviti da u prethodnom koraku razvijeni matematički model daje točan prikaz stvarnosti. Kako bi provjerili valjanost našeg modela trebamo provjeriti optimalnost za vrijednosti koje nisu korištene za procjenu. Treba imati na umu da trenutna prihvatljivost matematičkog modela ne podrazumijeva njegovu buduću prihvatljivost, model se mora aktualizirati i prilagoditi budućim ograničenjima. Odabire se alternativa koja najbolje odgovara zadanim ciljevima. Odabrana alternativa prezentira se donositeljima odluke. Ukoliko se preporuke prihvate, dolazi do implementacije. Sustav se mora konstantno nadgledati i ažurirati kako bi se osiguralo ostvarenje ciljeva. U zadnjem koraku potrebno je provesti korekcije i adaptacije modela s obzirom na dinamično okruženje.

Na shematskom prikazu [Slika 3] prikazan je opći postupak prilikom modeliranja. Analizom stanja u realnom svijetu prikupljaju se informacije za formulaciju modela. Odabranim postupcima nad modelom nastaje odziv, zaključci o ponašanju modela. Zaključci o modelu se interpretiraju kako bi bili upotrebljivi u realnom svijetu. Na kraju se zaključci o modelu uspoređuju sa zaključcima do kojih se dolazi analizom realnog svijeta, ali bez korištenja modela. Tek se tada mogu formulirati elementi za donošenje stvarne odluke. [2]

Procedura izrade modela [3]:

5. definiranje problema,
6. promatranje sustava,
7. formulacija matematičkog modela (uključujući i statistički model) problema,
8. provjera modela i upotreba modela za predviđanje,
9. odabir odgovarajuće alternative,
10. prezentiranje rezultata i zaključaka,
11. primjena i analiza učinkovitosti preporuka.



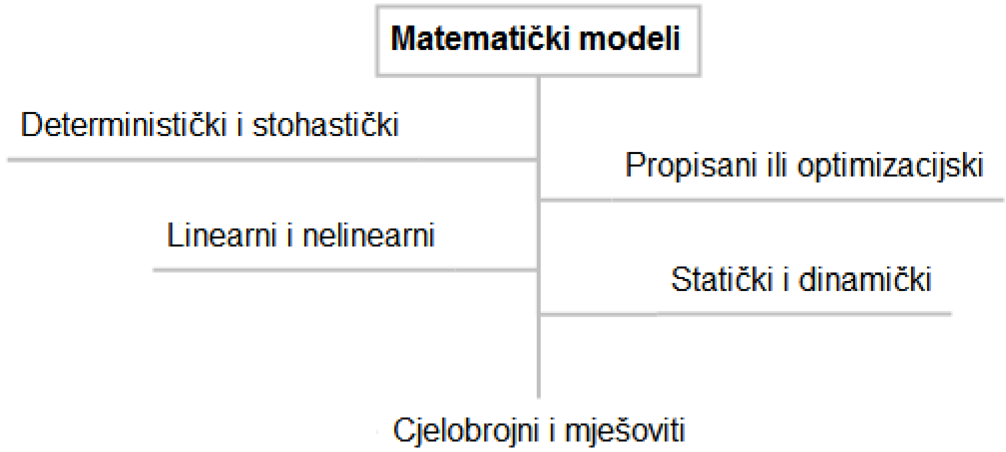

Slika 3. Proces modeliranja [2]

U modele operacijskih istraživanja i njihovo rješavanje, mogu se ubrojiti [4]:

- matematički modeli optimiranja,
- modeli transporta,
- teorija grafova,
- modeli mrežnih tijekova,
- teorija igara,
- modeli zaliha,
- modeli repova čekanja,
- modeli odlučivanja,
- simulacija.

Matematički modeli optimiranja mogu biti [Slika 4]:

12. Propisani ili optimizacijski,
13. Statički i dinamički,
14. Linearni i nelinearni,
15. Cjelobrojni i mješovit cjelobrojni („ne cjelobrojni“),
16. Deterministički i stohastički.



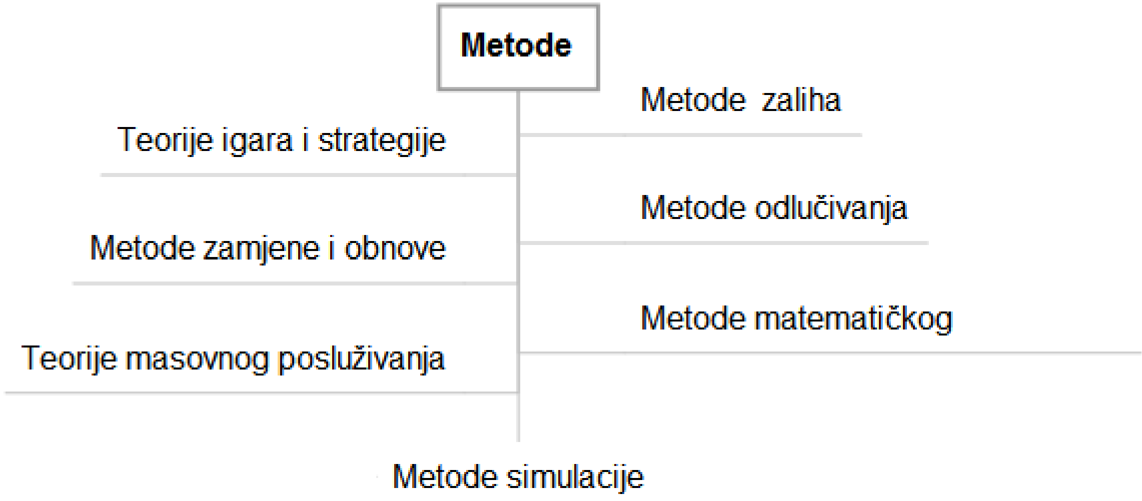

Slika 4. Podjela matematičkih modela

Potreba da se problemi iz prakse obuhvate i riješe, dovode do nastanka dva pravca u Operacijskim istraživanjima [4]:

1. teorijski pravac,
2. primijenjeni pravac.

Nemoguće je povući jasnu granicu između ta dva pravca. Teorijski pravac bavi se izradom matematičkih teorija za pojedina pitanja. Tako se kod cjelobrojnog programiranja istražuju svojstva diskretnih skupova, nova područja u teoriji grafova; u matematičkom programiranju istražuju se svojstva pojedinih funkcija modela, izrađuju se metode za rješavanje teorijskih modela. U primijenjenom pravcu razvijaju se metode za konkretne praktične probleme, programi i programski paketi. [4]

Izbor metode ovisi o modelu [Slika 5]. Nekada treba izraditi novu metodu za model, a u nekim slučajevima se oslanjamo na već poznate metode . Na primjer, kod postavljenog konkretnog linearnog modela za rješavanje se koristi simpleks metoda. [4]




Slika 5. Metode operacijskih istraživanja s obzirom na prirodu problema

# 4. METODE OPERACIJSKIH ISTRAŽIVANJA U USLUŽNIM DJELATNOSTIMA

S obzirom da postoji velik broj metoda OI koje se koriste u uslužnim djelatnostima, u daljnjem tekstu bit će opisane samo one metode bitne za rješavanje problema bolničkih sustava. Ostale metode će biti opisane isključivo u nekoliko rečenica.

## 4.1. Nelinearno programiranje [2]

Nelinearno programiranje (NP) se bavi optimiranjem nelinearnih (linearnih) funkcija s linearnim ili nelinearnim ograničenjima.

Matematički model NP-a se može napisati u obliku (1):

$$\max F(x) \tag{1}$$

Uz ograničenja (2) i (3) za slučaj maksimuma dobiva se *x* (4):

$$g_i(x) \leq 0, \qquad i = 1, 2, 3 \dots m \tag{2}$$

$$h_j(x) = 0, \qquad i = 1, 2, 3 \dots l \tag{3}$$

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{4}$$

gdje je:

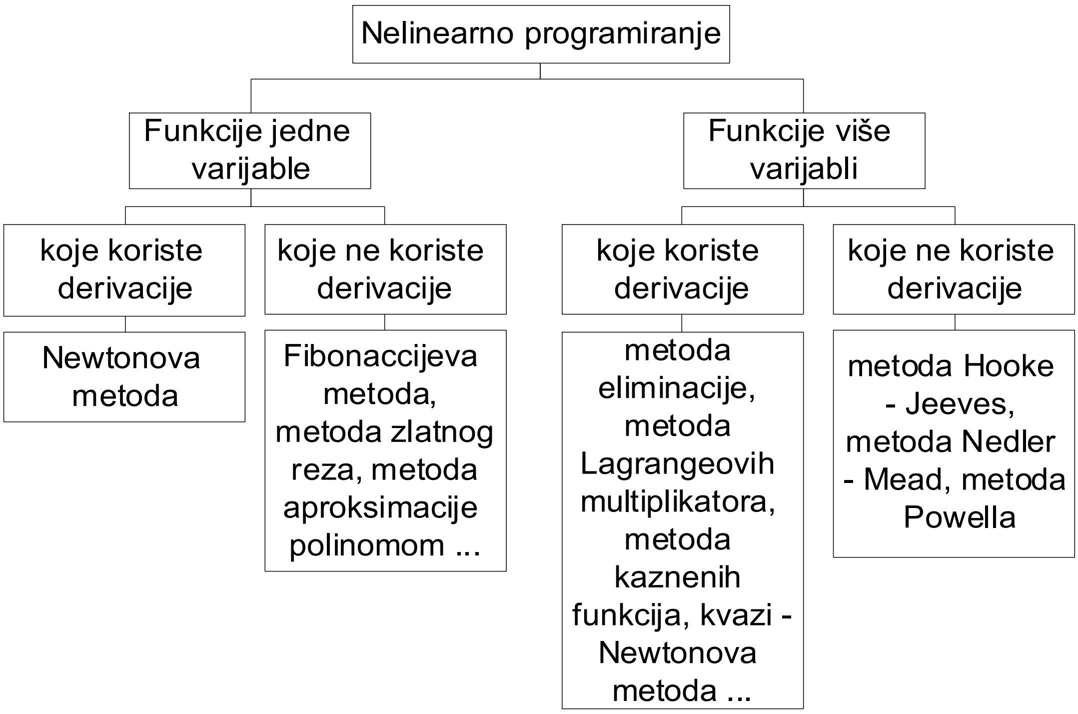
- $F(x)$ – funkcija cilja,
- $g_1 \dots g_m$ – ograničenje u obliku nejednadžbe,
- $h_1 \dots h_l$ – ograničenje tipa jednadžbe.

Pritom je $\boldsymbol{x} \in \boldsymbol{X}$ (višedimenzionalno), odnosno $\boldsymbol{x}$ je vektor koji mora zadovoljiti ograničenja. Svi vektori u rješenju čine skup mogućih rješenja. [2]

Dakle, problem NP-a svodi se na pronalaženje optimalne vrijednosti $\boldsymbol{x}_{1,2 \dots n}$ koje će zadovoljiti postavljena ograničenja, a u optimalnom rješenju nelinearna funkcija cilja će dosegnuti maksimalnu vrijednost.

Metode optimiranja koje se koriste kod programiranja su [2]:

1. analitičke metode – koriste se klasičnim tehnikama diferencijalnog i varijacijskog računa (parcijalne derivacije, ekstremi…); metoda se upotrebljava za jednostavnije oblike nelinearnih funkcija,
2. numeričke metode – koriste informacije iz prethodnog koraka za dobivanje boljih rješenja; ove metode se koriste elementima vektorske analize,
3. grafičke metode – crtanje funkcije jedne ili više varijabli pri čemu se traži minimum ili maksimum,
4. eksperimentalne metode – optimum funkcije cilja dobije se eksperimentalnim radom.




Slika 6. Podjela metoda nelinearnog programiranja [2]

## 4.2. Linearno programiranje [5]

Metode linearnog programiranja (LP) su najvažniji instrument operacijskih istraživanja. U linearno programiranje (LP) spadaju problemi gdje su funkcija cilja i funkcije ograničenja zapisane u obliku linearnih jednadžbi i nejednadžbi. Pod linearnim programiranjem podrazumijeva se rješenje nekog matematičkog zadatka koji se sastoji u optimiranju neke linearne funkcije čije varijable zadovoljavaju neki sustav linearnih jednadžbi i nejednadžbi.

U praksi ima veliko značenje jer se može primijeniti u rješavanju različitih problema. Tipični problemi koji spadaju u ovu kategoriju su problemi optimalnog razmještanja resursa, transportni problemi, mrežni problemi i drugi slični problemi.

Najčešće problemi optimizacije za funkciju cilja imaju maksimum profita (kada se radi o dobiti) ili minimum troškova, a za ograničenja granične vrijednosti (limite) resursa (npr. raspoloživ broj radnih sati, fizička ograničenja prostora ili transportnih sredstava). Minimum troškova se pri tome u općem slučaju ne može uzeti u uskom smislu kao minimum operativnih troškova. Potrebno je realnije modelirati pripadne troškove te obuhvatiti operativni utrošak materijala, rada (direktni i indirektni), angažman ili trošenje prostora, strojeva i opreme iskazano putem amortizacije, popratne troškove. Sve ove elemente potrebno je prema potrebi na odgovarajući način podijeliti jedinici proizvoda odgovarajućim normativom.

Neki postupci nelinearnog programiranja (NPL) provode uzastopnu linearizaciju problema u pojedinim točkama te onda koriste varijante LP za rješavanje tih lineariziranih potproblema. Uzastopna linearizacija nelinearnih problema s ograničenjima provodi se kod niza metoda NPL-a iz tog razloga potrebni su učinkoviti algoritmi nalaženja optimuma LP-a.

LP se najviše koristi u vojsci i industriji: proizvodni programi, izbor lokacija tvornica, optimalno planiranje investicijskih ulaganja, razmještaj strojeva, izbor optimalnih tehnoloških postupaka, sastavljanje optimalnih planova prehrane, transporta, izbor i razmještaj sredstava naoružanja. Uspješnost optimizacije kod linearnih problema ovisi o znanju i sposobnosti inženjera koji modelira promatrani problem.

### *4.2.1. Opći oblik problema linearnog programiranja*

Opći problem kod LP-a definiran je sa:

- funkcijom cilja (5) i
- ograničenjima (6),(7) i (8).

$$\min f(x) = \sum_{i=1}^{n_0} c_i x_i \qquad za\ i = 1, n_0 \tag{5}$$

$$\sum_{i=1}^{n_0} a_{ij} x_i = b_j \qquad za\ j = 1, m_0 \tag{6}$$

$$\sum_{i=1}^{n_0} a_{ij} x_i \leq b_j \tag{7}$$

$$\sum_{i=1}^{n_0} a_{ij} x_i \geq b_j \tag{8}$$

Gdje su:

- $n_0$ zadani broj varijabli,
- $m_0$ izvorno zadani broj ograničenja,
- $c_i$ koeficijenti funkcije cilja (konstante),
- $a_{ij}$ konstantni koeficijenti,
- $b_j$ vrijednosti granica

Svi linearni problemi mogu se svesti na standardni oblik. Kod standardnog oblika LP vrijede sljedeći uvjeti:

1. određuje se minimum funkcije cilja,
2. sve varijable su nenegativne,
3. slobodni koeficijenti desnih strana jednadžbi ograničenja su nenegativni,
4. sva ograničenja su svedena na ograničenja jednakosti, odnosno (9):

$$\begin{aligned} &\min f(x) = \boldsymbol{c}^T \boldsymbol{x} \\ &\text{uz } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ &\boldsymbol{x} \geq 0 \end{aligned} \tag{9}$$

Da bi se opći problem sveo na standardni oblik, potrebno je provesti sljedeće intervencije u modelu optimiranja za konkretni LP:

1. Potrebno je da su članovi desne strane ograničenja ($b_j$) nenegativni, što se postiže eventualnim množenjem ograničenja sa (-1), kada se mijenja i karakter znaka nejednakosti u ograničenju, iz ≤ u ≥ i obrnuto.
2. Varijable $x_i$ moraju biti nenegativne, pa ako je neka varijabla slobodnog predznaka treba je zamijeniti s razlikom dvaju novouvedenih nenegativnih varijabli (10):

$$x_i \to x_{i,1} - x_{i,2} \tag{10}$$

Izraz (10) ima nepovoljnu posljedicu da uvodi niz dodatnih varijabli u problem. Kao alternativa ovom izrazu može se ići na dodavanje velike pozitivne konstante $K_i$ (ili konstante $K$) varijabli $x_i$ na način (11):

$$x_i \to {x_i}^* - K_i \tag{11}$$

no to ima za moguću posljedicu numerički slabo uvjetovan problem, uz nužnost da se nakon rješavanja LP provjeri jesu li sve konstante dovoljno velike da su varijable pozitivne. U suprotnom problem treba riješiti iznova s većom vrijednosti $K_i$.

3. Pretpostavlja se traženje minimuma funkcije $f(x)$, pa ako je u zadatku potrebno odrediti maksimum, može se npr. primijeniti transformacija (12):

$$\max(f) \to \min(-f) \tag{12}$$

4. U standardnom obliku LP, prihvatljiva su samo ograničenja jednakosti, pa se ograničenja u obliku nejednadžbi svode na ograničenja u obliku jednadžbi uvođenjem (nenegativnih) dopunskih varijabli (eng. *slack variables* i *surplus variables* $s_i$) na način (13):

$$\begin{aligned} &\text{umjesto:} \quad a_{i1}x_1 + a_{i2}x_2 + \ \dots + a_{in}x_n \leq \ ili \geq \ b_i \\ &\text{slijedi:} \quad a_{i1}x_1 + a_{i2}x_2 + \ \dots + a_{in}x_n \mp s_i = \ b_i \qquad za\ s_i \geq 0 \end{aligned} \tag{13}$$

Time se LP u standardnom obliku može zapisati kao (14):

$$\min f = \boldsymbol{c}^T \boldsymbol{x} \quad ili \quad \min f = \sum c_i x_i \qquad \text{za } i = 1, n \tag{14}$$

uz ograničenja (15) i (16):

$$\boldsymbol{Ax} = \boldsymbol{b} \quad ili \quad \sum_{j=1}^{n} a_{ij} x_j = b_i \qquad \text{za } i = 1, m \tag{15}$$

$$\boldsymbol{x} \geq 0 \tag{16}$$

gdje je:

- $n$ ukupan broj varijabli (zadane varijable plus dopunske varijable nastale kod svođenja u standardni oblik)
- $m$ ukupan broj ograničenja.

Jednadžbe ograničenja moraju biti linearno nezavisne uz uvjet $m < n$ , jer inače ne postoje stupnjevi slobode problema potrebni za optimizaciju.

Obzirom da je gradijent kod linearnih problema konstantan, optimum smjerom pada funkcije teži u beskonačnost ako područje nije ograničeno. Dopušteno područje kod LP mora biti omeđeno da bi postojao minimum. Kod LP optimum uvijek leži na nekom ograničenju odnosno na rubu dopustivog područja, što znači da je u točki optimuma aktivno neko ograničenje. Ovo slijedi iz prirode linearne funkcije cilja, ali i iz postavljanja nužnog uvjeta ekstrema $f(\boldsymbol{x})$, s obzirom da nisu svi koeficijenti funkcije cilja $c_i$ jednaki nuli. Stoga se postupak traženja rješenja LP svodi na traženje dopustivog rješenja (koje zadovoljava ograničenja) uz istovremeno smanjivanje vrijednosti funkcije cilja. Kako optimum kod linearnih problema leži na nekom od ograničenja, algoritam traženja optimuma može se svesti na pretraživanje granica dopustivog područja.

U *nD* prostoru (problem sa *n* varijabli), svako ograničenje u obliku jednadžbi geometrijski predstavlja (*n*-l) dimenzionalnu hiperplohu. Funkcija cilja $f = \boldsymbol{c}^T\boldsymbol{x}$ je hiperravnina u *n*D prostoru čiji je gradijent vektor koeficijenata $\boldsymbol{c}$. Dopustivo područje (eng. *feasible region*) je onaj dio *n*D prostora (domena) gdje su zadovoljena sva ograničenja.

Kod 2D problema domena je dio ravnine $O_{x1x2}$, a ograničenja u obliku jednadžbi su pravci u toj ravnini. Pravcima u toj ravnini mogu se prikazati i nivo-linije funkcije cilja. Funkcija cilja je ravnina iznad domene odnosno ravnine $O_{x1x2}$ koja se unutar domene može prikazati svojim nivo-pravcima (izo-linije). Dopustivo područje je zatvoreni poligon. Kod 3D problema domena je u prostoru $O_{x1x2x3}$, a ograničenja jednakosti su ravnine u tom prostoru.

Aktivna ograničenja su ona gdje su uvjeti ograničenja zadovoljeni sa znakom jednakosti, što znači da u aktivna ograničenja spadaju sva ograničenja jednakosti i ona ograničenja nejednakosti gdje je ograničenje nejednakosti ($\leq$ , $\geq$) zadovoljeno sa znakom jednakosti (ograničenja na kojima leži trenutna točka), odnosno gdje su vrijednosti dodatnih varijabli $s_i$ jednake nuli.

Mogući slučajevi rješenja LP su:

- jedinstveno rješenje,
- nejedinstveno rješenje, rješenje brid poligona (aktivno ograničenje),
- neograničen problem, rješenje je u beskonačnosti,
- nemoguć problem, dopustivo područje ne postoji.

Kod posljednja dva slučaja, ako je očito da rješenje za zadani problem mora postojati, grešku treba pronaći u postavljanju modela LP.

Pretpostavka je da je LP sveden na standardni oblik te da ima $m$ jednadžbi ograničenja jednakosti i ukupno $n$ varijabli. Bazično rješenje problema je (dopustivo) rješenje sustava jednadžbi ograničenja (15) i (16) kada se višak varijabli postavi jednakim nuli, dok se ostalih $m$ varijabli nazivaju bazičnim varijablama i mogu se odrediti iz $m$ jednadžbi ograničenja. Višak varijabli nakon zadovoljavanja zadanih ograničenja, odnosno ($n$ – $m$) varijabli, nazivaju se nebazičnim varijablama.

Traženje optimuma se svodi na traženje optimuma među baznim dopustivim rješenjima odnosno na kretanje po poligonu ograničenja uz istovremeno smanjivanje vrijednosti funkcije cilja.

Kako je broj baznih rješenja velik, za rješavanje stvarnih problema je potreban učinkovit algoritam. Kombinacije za $n$ varijabli i $m$ ograničenja, za stupanj slobode problema koji odgovara višku od ($n$ – $m$) varijabli daje najveći broj baznih rješenja (17):

$$\binom{n}{m} = \frac{n!}{m!\,(n-m)!} \tag{17}$$

### *4.2.2. Simpleks metoda [5]*

Simpleks postupak je specijalizirani algoritam za učinkovito rješavanje LP zadanih u standardnom obliku. Postupak se svodi na kretanje po skupu bazičnih dopustivih točaka uz istovremeno smanjenje vrijednosti funkcije cilja. Postupak daje rješenje ili informaciju postoji li rješenje (jednostruko ili višestruko), te radi li se o neograničenom problemu ili problemu koji nema rješenje.

Simpleks u *n*D prostoru je konveksni skup od (n + 1) točke koje ne leže na istoj hiperravnini, odnosno u 2D (ravnini) skup od tri točke koje ne leže na istom pravcu.

#### 4.2.2.1. *Jednofazni postupak*

Jednofazni postupak predstavlja problem s ograničenjima ≤ . Sustav jednadžbi ograničenja (15) sa *m* jednadžbi i *n* varijabli svodi se na kanonski oblik, odnosno takav oblik gdje u svakoj jednadžbi postoji jedna varijabla s jediničnim koeficijentom koja se ne pojavljuje ni u jednoj drugoj jednadžbi (18):

$$\boldsymbol{I}_{(m\times m)} \cdot \boldsymbol{x}_m + \boldsymbol{Q}_{(m\times(n-1))} \cdot \boldsymbol{x}_{(n-m)} = \boldsymbol{b} \tag{18}$$

gdje je:

- $\boldsymbol{I}$ jedinična matrica reda m,
- $\boldsymbol{x}$ vektor varijabli,
- $\boldsymbol{Q}$ matrica koeficijenata.

Sustav jednadžbi (15) ima $m$ jednadžbi pa se njegovom primjenom može odrediti vrijednost $m$ varijabli. Ostale varijable, njih $(n - m)$, mogu poprimiti proizvoljne vrijednosti te predstavljaju slobodu u definiciji problema, a problem simbolički rečeno ima beskonačno mnogo kombinacja rješenja.

Kada se u sustavu (18) nebazične varijable $\boldsymbol{x}_{(n - m)}$ postave na nulu, rješenje za bazične varijable je $\boldsymbol{x}_m = \boldsymbol{b}$. Riječ je naravno o jednom od dopustivih rješenja jer zadovoljava sustav ograničenja (15).

Kod problema gdje su sva zadana ograničenja oblika $\sum a_{ij}x_j \leq b_i$ sustav jednadžbi ograničenja u obliku (18) se dobiva izravno. Dopunske varijable prema (13) koje ograničenje nejednadžbe transformiraju u ograničenje u obliku jednadžbe postaju u ovom slučaju bazične varijable s obzirom da se pojavljuju samo u jednoj jednadžbi pripadnog ograničenja.

Sada slijedi korak prijelaza iz jednog bazičnog dopustivog rješenja u drugo, geometrijski susjedno bazično dopustivo rješenje, uz smanjenje vrijednosti funkcije cilja. Ovo se realizira primjenom Gauss – Jordanove eliminacije (GJ) [Slika 7]. Pri tome $p$-ta bazična varijabla postaje nebazičnom i zamjenjuje mjesto sa $q$-tom nebazičnom koja postaje bazična varijabla (uz uvjet $a_{pq} \neq 0$). Tako se mijenja skup bazičnih varijabli.

To implicira da $x_q$ mora biti eliminiran iz svih redaka osim $p$-tog, a cijeli $q$-ti stupac dobiva koeficijente nula osim $a_{pq} = 1$, što se postiže eliminacijom članova $q$-tog stupca u svim recima (18) pomoću $p$-tog retka (pivot redak). Ovaj postupak odgovara koraku eliminacije kod Gaussove metode. Dijeljenje pivot retka s pivot elementom slijedi prema (19):

$$a_{pj}' = \frac{a_{pj}}{a_{pq}} \qquad \text{za } j = 1, n$$

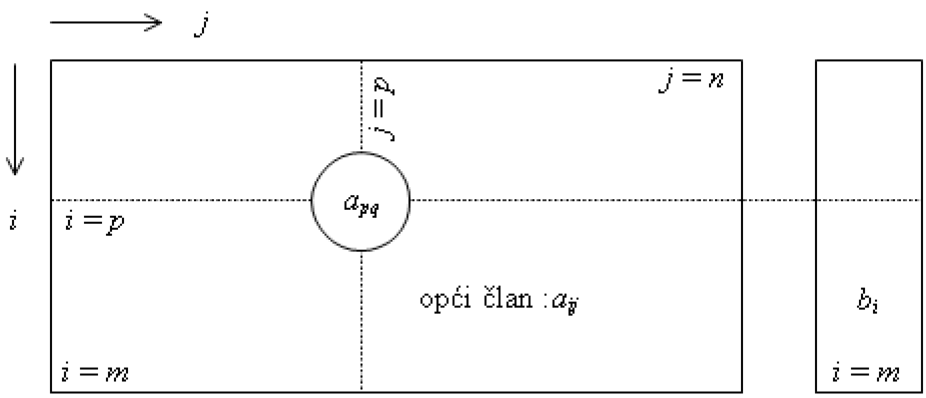
$$b_p' = \frac{b_p}{a_{pq}} \tag{19}$$

(19)[2]

[2] Nove vrijednosti koeficijenta označuju se sa crticom ( ' )

Eliminacija [Slika 7] u $q$-tom stupcu (20):

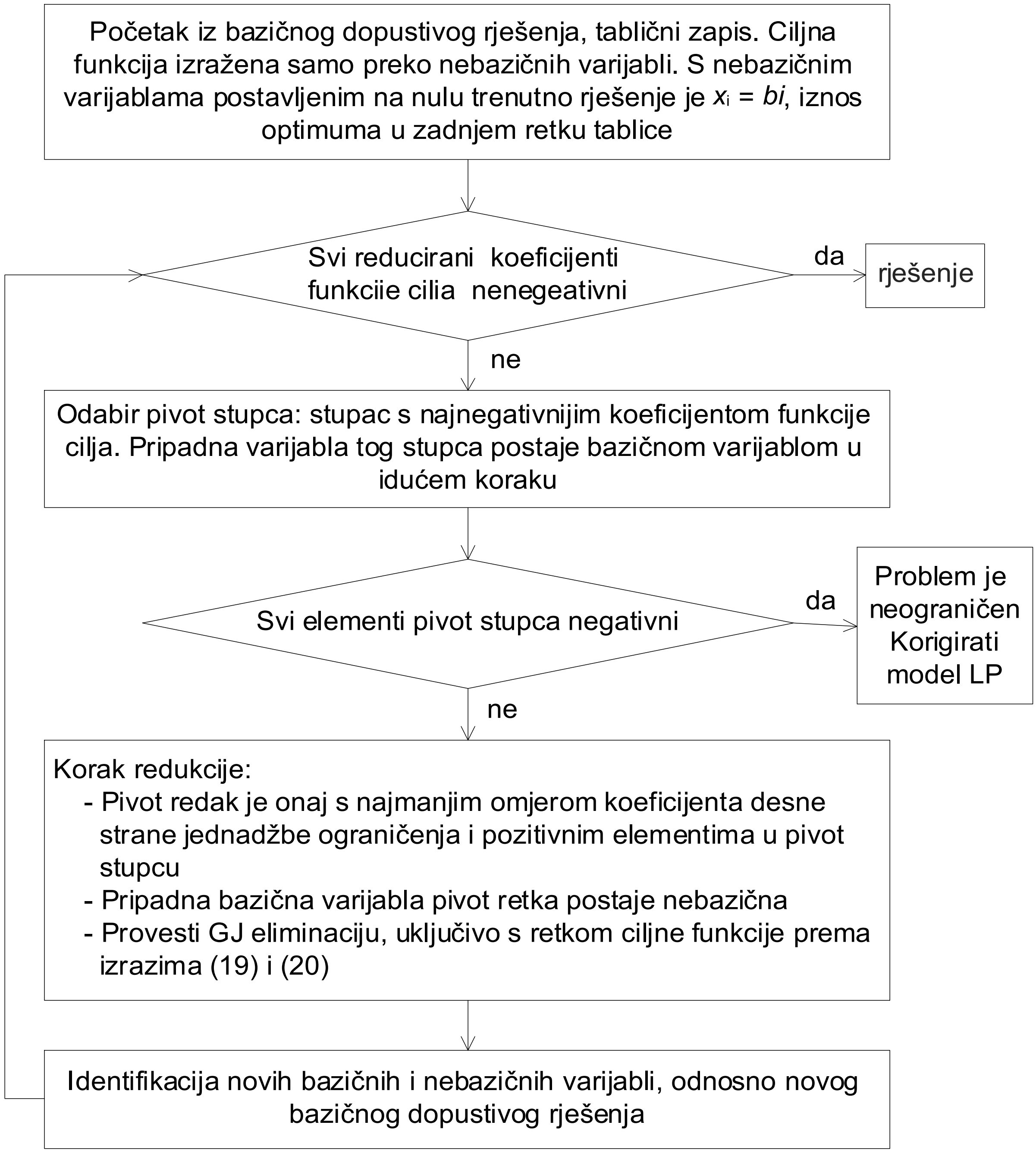
$$a_{ij}' = a_{ij} - a_{pj} \cdot \frac{a_{iq}}{a_{pq}} \qquad \text{za } j = 1, n$$
$$b_i' = b_i - b_p \cdot \frac{a_{iq}}{a_{pq}} \qquad j = 1, m, i \neq p \tag{20}$$




Slika 7. Shema Gauss – Jordanove eliminacije[5]

Za pivot stupac $q$ odabire se (nebazični) stupac s najnegativnijim koeficijentom koeficijentom u retku funkcije cilja. Razlog ovome je to što je vrijednost nebazičnih varijabli postavljena na nulu, pa se najveće smanjenje funkcije cilja postiže promjenom vrijednosti iz nula one nebazične varijable čiji je pripadni koeficijent funkcije cilja najnegativniji. Ako su svi koeficijenti nenegativni, postavljanje bilo koje nebazične varijable u baziču ne smanjuje funkciju cilja. Za pivot redak odabire se onaj gdje je omjer koeficijenta desne strane $b$ s pozitivnim članom u $q$-tom stupcu najmanji, pa bazična varijabla koja pripada tom retku u idućem koraku postaje nebazična. Ako su svi koeficijenti u pivot stupcu $q$ negativni, problem je neomeđen (neograničen), pa je rješenje $f$ = - ∞. Ako se ovo pravilo ne poštuje, iduća točka može biti nedopustiva. Zamjena varijabli znači korak prema idućoj (susjednoj) baznoj dopustivoj točki uz smanjenje vrijednosti funkcije, što se ponavlja do dolaska u ekstrem. GJ eliminacija se provodi i u zadnjem retku tablice (funkcija cilja) koji ima koeficijente samo uz nebazične varijable. Kada svi koeficijenti retka funkcije cilja tijekom postupka postanu nenegativni, dobiveno je rješenje, a ako je neki od njih nula, moguća su višestruka rješenja. Obično se primjenjuje tablični zapis sustava jednadžbi ograničenja (18). Svakom retku (pojedinom ograničenju) pripada odgovarajuća bazična varijabla, čija se vrijednost (uz nebazične varijable postavljene na nulu) vidi u stupcu $b$.

Na dijagramu [Slika 8] je pregledno prikazan dijagram toka za postupak rješavanja simpleks problema.




Slika 8. Dijagram toka simpleks postupka [5]

#### 4.2.2.2. *Dvofazni postupak*

Dvofazni postupak predstavlja zadani problem s ograničenjima ≤, =, ≥ . U ovom slučaju potrebno je odgovarajućim postupkom doći u polazni oblik jednofaznog postupka Za svako ograničenje tipa = i ≥ nužno je uvesti po jednu umjetnu varijablu (eng. *artificial variable*) u odgovarajuće ograničenje kako bi se dobio polazni kanonski oblik jednofaznog postupka. Time umjetne varijable postaju bazične varijable dotičnih jednadžbi ograničenja, a sustav poprima početni kanonski oblik sustava jednadžbi u prostoru proširenom umjetnim varijablama (eng. *expanded space*). Pri tome se koristi umjetna funkcija cilja koja predstavlja sumu svih umjetnih varijabli, ali izraženu preko nebazičnih varijabli.

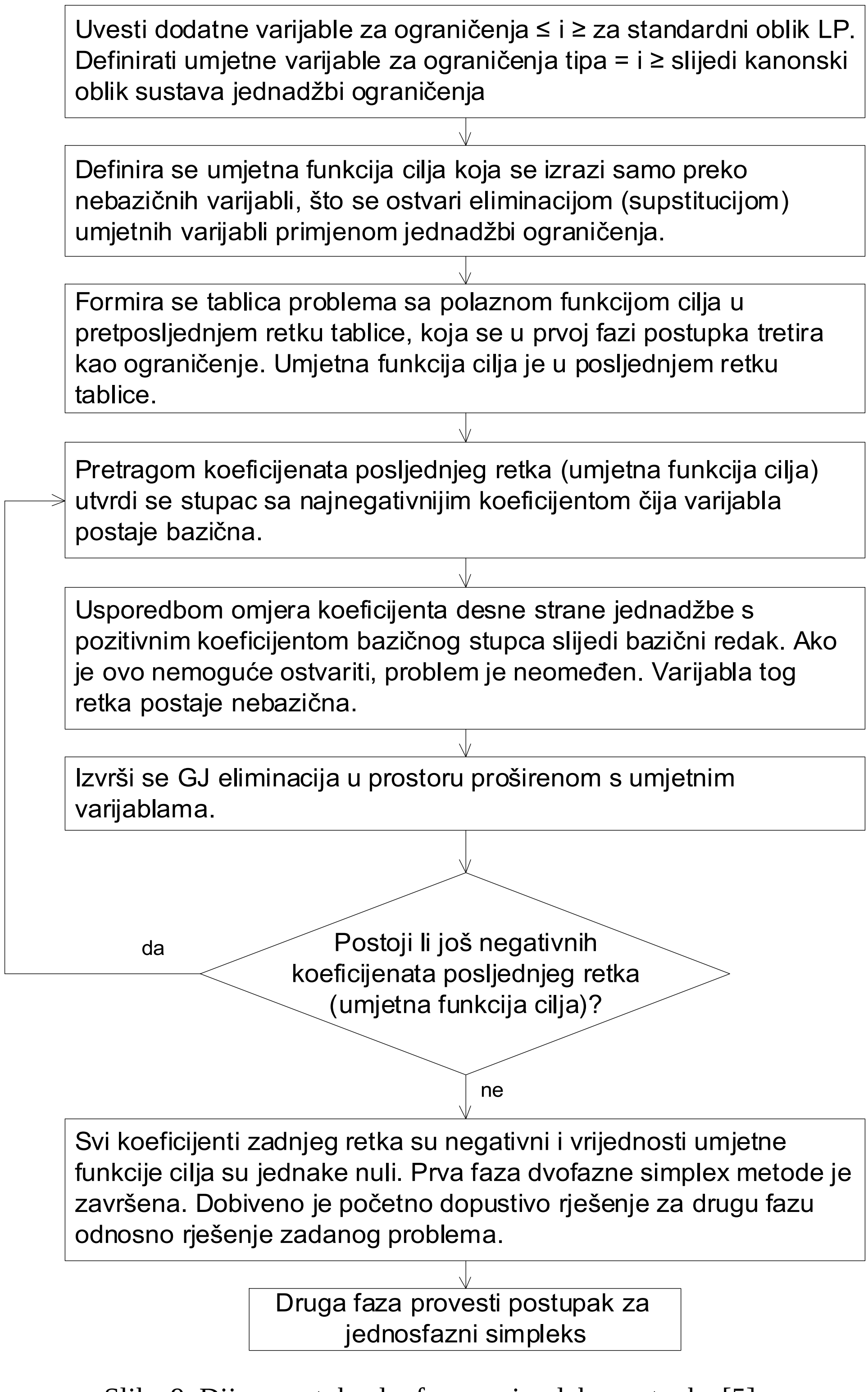
Redoslijed dvofaznog postupka isti je kao kod jednofaznog postupka, .ali u proširenom prostoru, i to u nizu koraka sve dok se ne dođe do točke u polaznom prostoru, kada se prošireni prostor povrati u prostor polaznih varijabli. Ovo se događa kada sve umjetne varijable postanu nebazične, odnosno s vrijednostima jednakim nuli, čime je i umjetna funkcija cilja jednaka nuli. Od ove točke dalje, Simpleks metodom se u izvornom prostoru korača do optimuma, a umjetna funkcija cilja i umjetne varijable se odbace.

Kod zapisa tablice problema, polazna funkcija cilja se pojavljuje u pretposljednjem retku, a umjetna funkcija cilja u posljednjem. Pri tome je prethodno potrebno umjetnu funkciju cilja izraziti isključivo preko nebazičnih varijabli, što se postiže eliminacijom (supstitucijom) bazičnih varijabli iz umjetne funkcije cilja putem jednadžbi ograničenja. U ovom dijelu postupka (prva faza) izvorna funkcija cilja se tretira kao ograničenje te se i nad njom provodi Gaussova eliminacija (kao u (19) i (20)) čime ona na kraju prve faze postupka ostaje funkcija trenutno nebazičnih varijabli.

Polazni problem se prvo mora svesti na standardni oblik. Prva faza postupka je završena kada je dostignut optimum prve faze, odnosno kada je vrijednost umjetne funkcije cilja i umjetnih varijabli nula. Tada se prošireni prostor reducira u polazni prostor odbacivanjem umjetnih varijabli i umjetne funkcije cilja. Ukoliko se prva faza ne uspije na ovaj način riješiti, problem je pogrešno zadan ili loše modeliran (npr. pogrešno zadana ograničenja).

Alternativno se umjesto dvofazne Simpleks metode može koristiti i BIG- M, metoda s kaznenim multiplikatorima. Kod ove metode se također uvode umjetne varijable, ali se ne uvodi umjetna funkcija cilja. Umjesto toga se umjetne varijable, pomnožene s velikim kaznenim multiplikatorima, pribroje zadanoj funkciji čime ih sam tijek minimizacije svodi na nulu logikom kaznene funkcije. Kazne funkcije obrađene su u nelinearnom programiranju. Prije provedbe Simpleks metode, umjetne varijable (koje su bazične) eliminiraju se u funkcije cilja primjenom odgovarajućih jednadžbi ograničenja.

Na slici [Slika 9] prikazan je dijagram toka za prvu fazu postupka rješavanja dvofaznoe metode. Nakon prve faze rješavanje se nastavlja s drugom fazom prema proceduri prikazanoj na dijagramu toka [Slika 8] za jednofaznu metodu.




Slika 9. Dijagram toka dvofaznog simpleks postupka [5]

#### 4.2.3. Dual [5]

Ovdje je prenesena samo osnovna informacija kao. Za svaki LP problem sa $n$ varijabli i $m$ ograničenja (primami problem) postoji dualni problem sa $m$ varijabli i $n$ ograničenja. Iz rješenja jednog može se dobiti rješenje drugog.

Svaka varijabla dualnog problema je pridružena ograničenju zadanog (primarnog) problema i obrnuto. Svaka varijabla primarnog problema pridružena je ograničenju dualnog problema. Primami problem ima ograničenja tipa ≤ , a dualni tipa ≥ .

Koeficijenti funkcije cilja („cijene“) primarnog problema postaju slobodni koeficijenti jednadžbi ograničenja („granice resursa“) dualnog problema. Slobodni koeficijenti jednadžbi ograničenja primarnog problema postaju koeficijenti funkcije cilja dualnog problema.

Matrica koeficijenata jednadžbi ograničenja primarnog problema se transponira za ograničenja dualnog problema. Traženje maksimuma| primarnog problema postaje traženje minimuma dualnog problema. Nenegativnost varijabli vrijedi za zadani i dualni problem. Ograničenja jednakosti se transformiraju u dvostruka ograničenja nejednakosti ≤ i ≥ .

Ako je primami problem definiran sa (21) sa ograničenjima oblika ≤ i slobodnim predznakom $\boldsymbol{b}$,

$$\begin{array}{ll} \text{max}imum & \boldsymbol{c^T} \cdot \boldsymbol{x} \\ \text{ograničenja:} & \boldsymbol{A}_{(m \times n)} \cdot \boldsymbol{x} \leq \boldsymbol{b} \end{array} \tag{21}$$

tada je dualni problem (22):

$$\begin{array}{ll} \text{max}imum & \boldsymbol{b^T} \cdot \boldsymbol{y} \\ \text{ograničenja:} & \boldsymbol{A}^T \cdot \boldsymbol{y} \geq \boldsymbol{c} \end{array} \tag{22}$$

gdje su:

- **y** dualne varijable $y_i$, $i$ = l, $m$, svaka pridružena po jednom ograničenju primarnog problema.

Standardni zapis dualnog problema slijedi iz zapisa LP u obliku (21), gdje se ograničenja svode na oblik ≤ množenjem ograničenja oblika sa -1 bez obzira na promjenu predznaka $\boldsymbol{b}$, te svođenja ograničenja tipa = na tip ≤ . Nakon rješenja dualnog LP, iz odgovarajuće konačne tablice (iz reduciranih koeficijenata funkcije cilja u stupcima dodatnih varijabli uz ograničenja) može se dobiti rješenje primarnog LP.

### *4.2.4. Postoptimalna analiza[5]*

Postoptimalna analiza se nadovezuje na optimalno rješenje. Posebno se to odnosi na analizu osjetljivosti. Osjetljivost problema na promjene konstanti u izrazima ograničenja izravno je ovisna o vrijednostima Lagrangeovih množitelja u točki rješenja. Uvidom u vrijednosti koeficijenata u završnoj tablici problema, moguće je do izvjesne mjere napraviti analizu osjetljivosti, odnosno odrediti promjenu optimalnog rješenja za određenu promjenu nekih ulaznih podataka problema.

Kod LP vrijednost Lagrangeovog množitelja za pojedino ograničenje jednaka je iznosu reduciranog koeficijenta funkcije cilja u stupcu dodatne ili umjetne varijable koja pripada tom ograničenju (u završnoj tablici u kojoj je dobiveno rješenje):

- kod ograničenja $\leq$ , to je reducirani koeficijent funkcije cilja uz pripadnu dodatnu varijablu u završnoj tablici, nenegativne vrijednosti
- kod ograničenja $=$ , to je reducirani koeficijent funkcije cilja uz pripadnu umjetnu varijablu u završnoj tablici, slobodnog predznaka
- kod ograničenja tipa $\geq$ , to je reducirani koeficijent funkcije cilja uz pripadnu umjetnu varijablu u završnoj tablici (nepozitivne vrijednosti) ili (ekvivalentno) reducirani koeficijent funkcije cilja uz pripadnu dodatnu varijablu u završnoj tablici, suprotnog predznaka

#### 4.2.4.1. *Analiza osjetljivosti[4]*

U analizi osjetljivosti se istražuje koliko se početni podaci mogu promijeniti uz uvjet da se optimalno rješenje ne mijenja. Rješenje više nije optimalno ili dopušteno kada se za primjer duala, na desnoj strani pojavi negativni element ili se on pojavi u funkciji cilja.

Pomoću analize osjetljivosti izračunavaju se, mogući pomaci pojedinih ograničenja između dviju točaka koje su susjedne optimumu. Formalna računica se svodi za izračunavanje najmanjeg pozitivnog i apsolutno najmanjeg koeficijenta od elemenata desne strane i koeficijenata, odnosnog stupca. Ti kvocijenti odgovaraju onima koji se izračunavaju u drugoj fazi za izbor pivot reda.

Dosada smo se pitali za promjene koje su se ispoljavale na optimumu ako se početne primame vrijednosti (elementi desne strane) nebazičnih varijabli optimuma promijene. Takva promjena može se interpretirati kao pomak pojedinih ograničenja dopuštenog rješenja. Na isti se način pita i za pomak optimuma ako se početne dualne vrijednosti (koeficijent funkcije cilja) baznih varijabli koje se nalaze u optimalnom rješenju promijene. Takva promjena može se predstaviti kao okretanje funkcije cilja.

Okretanje funkcije cilja u gornjem problemu postiže se npr. promjenom koeficijenta funkcije cilja. Kod analize osjetljivosti se traži apsolutno najmanja negativna vrijednost i najmanja pozitivna vrijednost za kod koje koeficijent funkcije cilja postaje negativan. Te vrijednosti označavaju granice do kojih se može mijenjati prvotni koeficijent iz funkcije cilja odgovarajuće varijable prije nego optimum skoči na neku drugu ugaonu točku

U sažetom obliku pravila računanja glase: kod analize osjetljivosti nebazičnih varijabli apsolutno najmanji negativan i najmanji pozitivan kvocijent iz primarnih vrijednosti i odgovarajućeg stupca optimalnog rješenja pokazuje širinu odstupanja oko odgovarajuće početnog rješenja iz apsolutno najmanjeg negativnog i najmanjeg pozitivnog kvocijenta iz dualnih vrijednosti i koeficijenta odnosnog reda optimalnog rješenja.

#### 4.2.4.2. *Parametarsko programiranje*

Kod ovog programiranja ide se korak dalje u odnosu na analizu osjetljivosti. Ne pita se samo koliko je moguće promijeniti pojedine početne koeficijente nego ih se i mijenja. Promjene se mogu odnositi na elemente desne strane (primame vrijednosti), na koeficijente funkcije cilja (dualne vrijednosti) ili pak na preostale koeficijente. Istovremeno je moguće promijeniti više koeficijenata, što znači da se promjena izvrši u jednom čvrsto zadanom odnosu. [3]

## 4.3. Višekriterijalno programiranje

U procesu odlučivanja često nije moguće definirati samo jednu funkciju cilja. Kriterija, koje traženo rješenje treba zadovoljavati, obično ima više. Kod izbora automobila, na primjer, bilo pri kupovini ili proizvodnji, želimo ostvariti što veću udobnost, brzinu i ubrzanje uz najmanju cijenu i utrošak goriva. Problem prehrane traži zadovoljenje dnevnih potreba organizma uz najmanje troškove i minimalnu konzumiranu količinu štetnih tvari. Kod planiranja proizvodnje želi se postići maksimalni doprinos za pokriće (razlika između prihoda i direktnih troškova), ali cilj može biti i zadovoljenje potreba tržišta u što kraćem roku. Izbor konačnog rješenja tada predstavlja kompromis između nekoliko zahtjeva, često suprotnih. Od donosioca odluke zahtijeva se da procijeni prihvatljivost ponuđenog rješenja i eventualno usmjeri proces optimizacije prema boljemu. Metoda višekriterijalnog programiranja ima mnogo, od kojih svaka ima svoje prednosti, ali i nedostatke. [1]

Višekriterijalno programiranje može se podijeliti u dvije osnovne kategorije [1]. U prvu kategoriju spadaju metode optimalnog izbora, koje se koriste u slučaju kada treba izabrati najbolju mogućnost između ograničenog broja prethodno definiranih mogućnosti. Svaka će mogućnost biti opisana s nekoliko atributa, koji mogu biti kvalitativni i kvantitativni. Kao optimalno rješenje odabire se mogućnost čiji su atributi po nekim kriterijima bolji od atributa ostalih mogućnosti. Te će se metode koristiti, primjerice, pri kupovini elektroničkog računala kad treba izabrati najpovoljniju između nekoliko ponuda raznih proizvođača, pri odabiranju poslovnih partnera i slično.

U drugu kategoriju spadaju metode optimizacije s više funkcija cilja koje nisu povezane s izborom iz skupa prethodno definiranih rješenja. Problemi kojima se te metode bave sastoje se od određenog broja funkcija cilja koje treba maksimizirati ili minimizirati i skupa ograničenja koja se postavljaju na varijable modela. Takvi problemi imaju beskonačno mnogo rješenja. Kao primjer za tu grupu metoda može se navesti problem prehrane ili oblikovanja novog proizvoda.

Svaka metoda višekriterijalnog programiranja zahtijeva prisutnost čovjeka koji postavlja zahtjeve, usmjerava proces optimizacije i na kraju prihvaća ili odbija ponuđeno rješenje. Za tu osobu, ili više njih, uobičajeni je naziv donosilac odluke. Dalja podjela obiju grupa vrši se prema učešću donosioca odluke u procesu optimizacije. [1]

Metode optimizacije s više funkcija cilja dijele se prema zahtjevima koji se postavljaju na donosioca odluke da usmjeri proces prema zadovoljavajućem rješenju,te prema tipu informacija koje treba dati u proces i prema trenutku u kojem su te informacije potrebne. Tako postoje metode koje ne zahtijevaju od donosioca odluke da se izjasni o preferencijama na funkciju cilja, odnosno metode kod kojih se preferencije zadaju prije procesa optimizacije, u toku njega, ili nakon njega.

Metode optimalnog izbora dijele se prema tome daje li donosilac odluke informaciju o svojim zahtjevima na atribute modela, ili se izjašnjava o preferenciji neke mogućnosti, ili ne daje nikakvu informaciju.

## 4.4. Dinamičko programiranje [1]

Dinamičko programiranje je metoda za modeliranje i rješavanje posebno strukturiranih zadataka vezanih uz više – etapne procese odlučivanja. Problem se rješava po fazama, te se u proračunu svake faze koriste optimalne vrijednosti dobivene u prethodnoj fazi. Taj je postupak poznat kao Bellmanov princip i njime se dobije slijed optimalnih odluka.

Dakle, proces se sastoji od niza etapa i u svakoj od njih treba donijeti odluku, tj. treba izabrati vrijednosti parametara odlučivanja pri čemu taj izbor ovisi o prethodnom ponašanju procesa i utječe na buduće ponašanje procesa.

Kod dinamičkog programiranja ne može se postaviti striktna matematička formulacija. Nije jednostavno definirati ni klase problema koji se na taj način mogu riješiti, niti tipična područja primjene.

Dva najpoznatija problema dinamičkog programiranja su problem najbržeg prijenosa i problem raspodjele investicija.

## 4.5. Cjelobrojno programiranje [1]

Kod velikog broja problema u matematičkom programiranju postoji zahtjev da varijable imaju cjelobrojne vrijednosti. Cjelobrojnost se javlja kod različitih problema: investicijski problemi instaliranja većeg broja strojeva, razmještanje radnika na odgovarajuće strojeve, problem montaže, problem fiksnih troškova, itd.

Ponekad se cjelobrojni modeli koriste za rješavanje necjelobrojnih problema kako bi se ubrzao proces rješavanja problema (trgovački putnik, problem lokacije). Tako primjerice simpleks metoda uvijek daje necjelobrojne vrijednosti varijabli, a sporedna će ograničenja zahtijevati da sve strukturne varijable ili samo neke budu cijeli brojevi. Ukoliko su samo neke varijable cijeli brojevi, radi se o mješovito – cjelobrojnom programiranju. A ako su sve strukturne varijable cijeli brojevi radi se o diskretnom ili cjelobrojnom programiranju. [4]

Za određivanje optimalnih cjelobrojnih rješenja LP-a razvijeni su specijalni algoritmi rješavanja:

1. metode odsijecanja ravnina – uvođenjem hiperravnina skup mogućih rješenja se iterativno sužava dok ne padne na cijeli broj,
2. metode stabala odlučivanja – pokušava se od optimalnog Mješovito cjelobrojnog rješenja postepenim uvođenjem pojedinih cjelobrojnih vrijednosti za varijable, doći do optimalnog cjelobrojnog rješenja,
3. heurističke metode – vode do suboptimalnih rješenja (prednost im je u bržem vremenu računanja, a nedostatak što ne pronalaze uvijek optimalno rješenje).

## 4.6. Transportni problem

Transportni problem (TP) je specijalni slučaj linearnog optimiranja. Cilj transportnog problema je odrediti najbolji način transporta od više izvora sredstava tj. ishodišta (npr. skladišta sirovina, materijala, proizvodna poduzeća koja opskrbljuju potrošača) do korisnika izvora sredstava tj. odredišta. Najčešće se razmatra problem minimalnih transportnih troškova prijevoza neke robe. [2]

Osnovne značajke transportnoga problema:

- izvori sredstava, tj. ishodišta označavaju se sa $R_1$, $R_2$, ...., $R_i$ ($i$ = 1, 2, ..., $r$), a njihov kapacitet sa $x_{i0}$,
- korisnici izvora sredstava, tj. odredišta označavaju se sa $S_1$, $S_2$, ..., $S_j$ ($j$ = 1, 2, ..., $s$), a njihovi kapaciteti sa $x_{0j}$;
- sa $x_{ij}$ označava se količina materijala koja se preveze iz bilo kojeg $i$-tog ishodišta u bilo koje $j$-to odredište, a sa $c_{ij}$ trošak transporta jedinice tereta iz bilo kojeg $i$-tog ishodišta u bilo koje $j$-to odredište.

Kod ovakvih problema mogu nastupiti dvije situacije:

1. da je suma ponude jednaka sumi potražnje - u tom slučaju se govori o zatvorenom modelu transporta, tj.

$$\sum_{i=1}^{r} x_{io} = \sum_{j=1}^{s} x_{oj} \ , \qquad (23)$$

2. ili da suma ponude nije jednaka sumi potražnje - u tom se slučaju govori o otvorenom modelu transporta, tj.

$$\sum_{i=1}^{r} x_{io} \neq \sum_{j=1}^{s} x_{oj} \ . \qquad (24)$$

Za drugi slučaj je potrebno model zatvoriti uvođenjem fiktivnog ishodišta, odnosno odredišta (ovisno o tome nedostaje li ponude ili potražnje).

Kako bi se riješio zadani transportni problem potrebno je definirati funkciju cilja $F(x)$ i sistem ograničenja. Zatim je potrebno odrediti nenegativne vrijednosti varijable $x_{ij}$ ($i$ = 1, 2, ..., $r$; $j$ = 1, 2, ..., $s$) tako da se ostvare minimalni ukupni troškovi transporta iz $r$-tog ishodišta u $s$-to odredište, a da se pri tome zadovolje uvjeti ponude i potražnje.

Matematička formulacija problema glasi:

$$F(x) = \sum_{i=1}^{r} \sum_{j=1}^{s} c_{ij} \cdot x_{ij} \rightarrow min\,. \tag{25}$$

Uvjeti ponude su:

$$\begin{gathered} \sum_{j=1}^{s} x_{1j} = x_{10} \\ \sum_{j=1}^{s} x_{2j} = x_{20} \\ \ldots\ldots\ldots\ldots \\ \sum_{j=1}^{s} x_{rj} = x_{r0} \end{gathered} \tag{26}$$

Uvjeti potražnje su:

$$\begin{gathered} \sum_{i=1}^{r} x_{i1} = x_{01} \\ \sum_{i=1}^{r} x_{i2} = x_{02} \\ \ldots\ldots\ldots\ldots \\ \sum_{i=1}^{r} x_{is} = x_{0s} \end{gathered} \tag{27}$$

Pri tome vrijedi uvjet nenegativnosti varijabli:

$$x_{ij} \geq 0\,, \qquad i = 1, 2, \ldots, r\,; j = 1, 2, \ldots, s\,. \tag{28}$$

U općem slučaju transportnog problema može se postaviti ukupno ($r$ + $s$) jednadžbi:

- $r$ jednadžbi za ishodišta,
- $s$ jednadžbi za odredišta.

Zbog uvjeta jednakosti sume ponude i sume potražnje broj neovisnih jednadžbi bit će ($r$ + $s$ – 1). Broj neovisnih varijabli određuje broj varijabli $x_{ij}$ koje mogu poprimiti vrijednost veću od nule ($xij$ > 0) u nekom otvorenom nedegeneriranom rješenju. Varijable koje imaju vrijednost strogo veću od nule (>), zovu se bazne varijable. Ako je broj baznih varijabli manji od ($r + s$ – 1), tada će nastupiti degeneracija, a takvo se rješenje u teoriji linearnog programiranja naziva osnovno degenerirano rješenje.

Značajke metoda za rješavanje transportnog problema jesu u tome što sam problem rješavaju u dvije faze:

1. najprije se određuje početno rješenje,
2. zatim se vrši poboljšanje početnog rješenja (što je u većini slučajeva iterativni postupak pronalaženja konačnog, tj. optimalnog rješenja).

Za određivanje početnog rješenja koriste se dvije metode, metoda sjeverozapadnog kuta i Vogelova metoda.

Za određivanje optimalnog rješenja mogu se koristiti sljedeće metode:

- Stepping-Stonova metoda (metoda skakanja s kamena na kamen),
- MODI metoda,
- Ford-Fulkersonova metoda.

Ovim postupcima se provjerava je li početno osnovno rješenje optimalno i pronalazi se bolje rješenje.

### *4.6.1. Metoda sjeverozapadnog kuta*

Metoda sjeverozapadnog kuta (Northwest corner rule) kao polaznu tablicu uzima tablicu u kojoj su dane ponude ishodišta i potražnje odredišta. Polazi se od lijevog gornjeg kuta te se uspoređuju potrebe $x_{10}$ i $x_{01}$.

a) Ako je $x_{10} > x_{01} \rightarrow x_{11} = x_{01}$, razlika $x_{10}$ - $x_{01}$ stavlja se u sljedeći stupac u istom redu ($x_{12} = x_{10} - x_{01}$). Ako je potražnja odredišta $S_2$ manja od ove razlike: $x_{12} = x_{02}$, razlika $x_{10} - x_{01} - x_{02}$ prenosi se u treći stupac u istom redu (sve dok se ponuda ishodišta $R_1$ ne iskoristi).
b) Ako je $x_{10} = x_{01} \rightarrow x_{11} = x_{10} = x_{01}$, zatim se prelazi se na određivanje varijable $x_{22}$ (tj. ide se dijagonalno od lijevog gornjeg kuta do donjeg desnog kuta).
c) Ako je $x_{10} < x_{01} \rightarrow x_{11} = x_{10}$, razlika $x_{01}$ - $x_{10}$ stavlja se u drugi redak u istom stupcu (prelazi se na određivanje varijable $x_{21}$). Postupak dalje slijedi kao pod točkom a) samo se mora uzeti u obzir da je ovdje prelazak vertikalan.

#### 4.6.1.1. *Primjer metode sjeverozapadnog kuta [1]*

Sljedeći primjer pokazuje kako doći do početnog rješenja transportnog problema preko metode sjeverozapadnog kuta. U tablici je zadan jednostavni transportni problem sa svojim ishodištima i odredištima.

Tablica 7. zadani transportni problem

| **TRANSPORTNI PROBLEM** | | | | | | | |
|---|---|---|---|---|---|---|---|
| ***Ishodišni troškovi $x_{i,j}$*** | | ***Odredišta*** | | | | | ***Ponuda ishodišta $x_{io}$*** |
| | | ***$S_1$*** | ***$S_2$*** | ***$S_3$*** | ***$S_4$*** | ***$S_5$*** | |
| ***Ishodišta*** | ***$R_1$*** | *4* | *2* | 5 | 7 | 6 | 20 |
| | ***$R_2$*** | 7 | 8 | 3 | 4 | 5 | 110 |
| | ***$R_3$*** | 2 | 1 | 4 | 3 | 2 | 120 |
| ***Potražnja odredišta $x_{oj}$*** | | 70 | 40 | 30 | 60 | 50 | 250/250 |

Potrebno je postaviti uvjete potražnje i ponude te funkciju.

$$\left.\begin{aligned} x_{11}+x_{12}+x_{13}+x_{14}+x_{15}&=20\\ x_{21}+x_{22}+x_{23}+x_{24}+x_{25}&=110\\ x_{31}+x_{32}+x_{33}+x_{34}+x_{35}&=120 \end{aligned}\right\}\text{uvjeti ponude}$$

$$\left.\begin{aligned} x_{11}+x_{21}+x_{31}&=70\\ x_{12}+x_{22}+x_{32}&=40\\ x_{13}+x_{23}+x_{33}&=30\\ x_{14}+x_{24}+x_{34}&=60\\ x_{15}+x_{25}+x_{35}&=50 \end{aligned}\right\}\text{uvjeti potražnje}$$

$F(x) = 4 \cdot x_{11} + 2 \cdot x_{12} + 5 \cdot x_{13} + 7 \cdot x_{14} + 6 \cdot x_{15} + 7 \cdot x_{21} + 8 \cdot x_{22} + 3 \cdot x_{23} + 4 \cdot x_{24} + 5 \cdot x_{25} + 2 \cdot x_{31} + 1 \cdot x_{32} + 4 \cdot x_{33} + 3 \cdot x_{34} + 2 \cdot x_{35} \rightarrow \min$

Tablica 8. Početno rješenje TP-a riješeno metodom sjeverozapadnog kuta

| TRANSPORTNI PROBLEM | | | | | | |
|---|---|---|---|---|---|---|
| $x_{i,j}$ | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ | $x_{io}$ |
| $R_1$ | ***20*** | | | | | 20 |
| $R_2$ | **50** | **40** | **20** | | | 110 |
| $R_3$ | | | **10** | **60** | **50** | 120 |
| $x_{oj}$ | 70 | 40 | 30 | 60 | 50 | 250/250 |

### *4.6.2. Vogleova metoda [2]*

Vogelova metoda je vrlo efikasna u određivanju prvog osnovnog rješenja transportnog problema. Postupak rada može se prikazati u četiri faze:

1. Za svako ishodište i odredište potrebno je odrediti kaznu za nekorištenje najjeftinijeg puta:
   - kazna se računa kao razlika dviju najmanjih cijena za svaki redak i stupac,
   - ako su u nekom retku (stupcu) dvije cijene iste, kazna je za to odredište (ishodište) jednaka nuli.
1. Izabrati redak ili stupac sa najvećom kaznom i u njemu polje ($i$, $j$) s minimalnim troškom transporta. U to se polje matrice unosi vrijednost varijable $x_{ij}$ = min ($x_{io}$, $x_{oj}$). Time se iscrpljuje $i$-ti redak ili stupac pa se izostavlja iz daljnjeg razmatranja.
2. Ako je redak (stupac) bio izostavljen, ponovo se računa kazna za svaki redak (stupac).
3. Potrebno je ponoviti faze 2 i 3 dok se ne dobije cijelo početno osnovno rješenje.

U drugoj fazi postoji više redaka ili stupaca sa jednakim maksimalnim kaznama. Postupak rada ima tri faze:

1. Pogleda se je li minimalna cijena u jednom od tih redaka (stupaca) ujedno i minimalna u stupcu (retku) u kojem se nalazi. Ako je, bira se taj redak (stupac). Ako nije, računaju se sekundarne kazne za retke i stupce koji imaju istu kaznu.
2. Računa se sekundarna kazna tj. izračuna se razlika između drugog najmanjeg troška u retku i najmanjeg troška u stupcu što sadrži taj najmanji trošak. Sekundarna kazna za stupac računa se na isti način.
3. Odabere se redak ili stupac sa najvećom sekundarnom kaznom.

#### 4.6.2.1. *Primjer Vogelove metode*

Primjer u nastavku [Tablica 9] pokazuje kako doći do početnog rješenja transportnog problema preko Vogelove metode.

Tablica 9. Zadani transportni problem,

| **TRANSPORTNI PROBLEM** | | | | | | |
|---|---|---|---|---|---|---|
| Jedinični troškovi $c_{i,j}$ | | Odredišta | | | | Ponuda ishodišta $x_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | $S_4$ | |
| Ishodišta | $R_1$ | 8 | 18 | 9 | 10 | 60 |
| | $R_2$ | 10 | 12 | 3 | 15 | 80 |
| | $R_3$ | 12 | 15 | 16 | 4 | 60 |
| Potražnja odredišta $x_{oj}$ | | 40 | 60 | 30 | 70 | 200 |

Tablica 10. Prvi korak rješavanja zadanog transportnog problema

| **PRIMJENA VOGELOVE METODE** | | | | | | | |
|---|---|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | | $x_{io}$ | $c'_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | $S_4$ | | |
| Ishodišta | $R_1$ | 8 | 18 | 9 | 10 | 60 | 1 |
| | $R_2$ | 10 | 12 | 3 | 15 | 80 | 7 |
| | $R_3$ | 12 | 15 | 16 | 4 | 60 | 8 |
| $x_{oj}$ | | 40 | 60 | 30 | 70 | 200 | |
| $c'_{oj}$ | | 2 | 3 | 6 | 6 | | |

$x_{ij}$ = min ($x_{io}$, $x_{oj}$)

$x_{34}$ = min (60, 70) = 60

Tablica 11. Drugi korak rješavanja zadanog transportnog problema

| **SUPTABLICA 2x4** | | | | | | | |
|---|---|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | | $x_{io}$ | $c'_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | $S_4$ | | |
| Ishodišta | $R_1$ | 8 | 18 | 9 | 10 | 60 | 1 |
| | $R_2$ | 10 | 12 | 3 | 15 | 80 | 7 |
| $x_{oj}$ | | 40 | 60 | 30 | 10 | 140 | |
| $c'_{oj}$ | | 2 | 3 | 6 | 5 | | |

$x_{23}$ = min *(80, 30) = 30*

Tablica 12. Treći korak rješavanja zadanog transportnog problema

| SUPTABLICA 2x3 | | | | | | |
|---|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | $x_{io}$ | $c'_{io}$ |
| | | $S_1$ | $S_2$ | $S_4$ | | |
| Ishodišta | $R_1$ | 8 | 18 | 10 | 60 | 2 |
| | $R_2$ | 10 | 12 | 15 | 50 | 2 |
| $x_{oj}$ | | 40 | 60 | 10 | 110 | |
| $c'_{oj}$ | | 2 | 6 | 5 | | |

$x_{23}$ = min (50, 60) = 50

Tablica 13. Četvrti korak rješavanja zadanog transportnog problema

| SUPTABLICA 1x3 | | | | | |
|---|---|---|---|---|---|
| $c_{i,j}$ | Odredišta | | | $x_{io}$ | $c'_{io}$ |
| | $S_1$ | $S_2$ | $S_4$ | | |
| $R_1$ | 8 | 18 | 10 | 60 | 2 |
| $x_{oj}$ | 40 | 10 | 10 | 60 | |

$$[x_{ij}] = \begin{bmatrix} 40 & 10 & 0 & 10 \\ 0 & 50 & 30 & 0 \\ 0 & 0 & 0 & 60 \end{bmatrix}$$

Tablica 14. Početno rješenje TP-a riješeno Vogelovom metodom

| PRIMJENA VOGELOVE METODE | | | | | |
|---|---|---|---|---|---|
| $x_{i,j}$ | | Odredišta | | | |
| | | $S_1$ | $S_2$ | $S_3$ | $S_4$ |
| Ishodišta | $R_1$ | 40 | 10 | | 10 |
| | $R_2$ | | 50 | 30 | |
| | $R_3$ | | | | 60 |

Dakle rješenje za F(x) Vogleovom metodom iznosi 1530.

$$F(x) = 40 \cdot 8 + 10 \cdot 18 + 10 \cdot 10 + 50 \cdot 12 + 30 \cdot 3 + 60 \cdot 4 = 1530$$

Metodom sjeverozapadnog kuta F(x) iznosi 1640.

### *4.6.3. Stepping-Stonova metoda [2]*

Ovom metodom se ispituje optimalnost osnovnog rješenja. Najprije se vrši računanje relativnih troškova za nezauzeta polja matrice transporta. Relativni troškovi ($d_{ij}$) pokazuju za koliko bi se promijenili ukupni troškovi transporta, ako bi se jedinica robe transportirala preko nezauzetog polja. Razlikuju se pozitivni relativni troškovi (povećanje transportnih troškova) te negativni relativni troškovi (smanjenje transportnih troškova). Optimalni plan transporta nastaje onda kada u osnovnom rješenje ne postoji niti jedan negativni relativni trošak.

Degeneracija kod TP-a nastupa kad je broj bazičnih varijabli manji od ($r + s – 1$).

#### *4.6.3.1.1. Primjer rješavanja Stepping-Stone metodom*

Tablica 15. zadani transportni problem

| **DEGENERACIJA KOD TP-a** | | | | | |
|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | $x_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | |
| Ishodišta | $R_1$ | 8 | 10 | 4 | 30 |
| | $R_2$ | 9 | 3 | 6 | 20 |
| | $R_3$ | 6 | 8 | 5 | 40 |
| | $R_4$ | 7 | 9 | 3 | 30 |
| $x_{oj}$ | | 30 | 40 | 50 | 110/110 |
| Degeneracija: r + s – 1 = 4 + 3 – 1 = 6; broj bazičnih varijabli = 5 | | | | | |

Tablica 16. Prvo osnovno rješenje dobiveno metodom sjeverozapadnog kuta

| DEGENERACIJA KOD TP-a | | | | | |
|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | $x_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | |
| Ishodišta | $R_1$ | $8/30$ | $10/$ | $4/$ | 30 |
| | $R_2$ | $9/$ | $3/20$ | $6/$ | 20 |
| | $R_3$ | $6/$ | $8/20$ | $5/20$ | 40 |
| | $R_4$ | $7/$ | $9/$ | $3/30$ | 30 |
| $x_{oj}$ | | 30 | 40 | 50 | $110/110$ |
| *F(x)* = 650 | | | | | |

Tablica 17. Prvi korak rješavanja

| DEGENERACIJA KOD TP-a | | | | | |
|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | $x_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | |
| Ishodišta | $R_1$ | $8/30-x$ | $10/$ | $4/+x$ | 30 |
| | $R_2$ | $9/$ | $3/20$ | $6/$ | 20 |
| | $R_3$ | $6/\varepsilon+x$ | $8/20$ | $5/20-x$ | 40 |
| | $R_4$ | $7/$ | $9/$ | $3/30$ | 30 |
| $x_{oj}$ | | 30 | 40 | 50 | $110/110$ |
| min $(20 - x = 0, 30 - x = 0)$; $x = 20$<br>$x_{11} = 10$, $x_{33} = 0$<br>$x_{13} = 20$, $x_{31} = 20$ | | | | | |
| $d_{12} = 0$ ; $d_{13} = -3$ ; $d_{21} = 8$ ; $d_{23} = 6$ ; $d_{41} = 3$ ; $d_{42} = 3$ | | | | | |

Tablica 18. Drugi korak rješavanja

| DEGENERACIJA KOD TP-a | | | | | |
|---|---|---|---|---|---|
| $c_{i,j}$ | | Odredišta | | | $x_{io}$ |
| | | $S_1$ | $S_2$ | $S_3$ | |
| Ishodišta | $R_1$ | $8/10$ | $10/$ | $4/20$ | 30 |
| | $R_2$ | $9/$ | $3/20$ | $6/$ | 20 |
| | $R_3$ | $6/20$ | $8/20$ | $5/$ | 40 |
| | $R_4$ | $7/$ | $9/$ | $3/30$ | 30 |
| $x_{oj}$ | | 30 | 40 | 50 | $110/110$ |
| min $(20 - x = 0, 30 - x = 0)$; $x = 20$<br>$x_{11} = 10$, $x_{33} = 0$<br>$x_{13} = 20$, $x_{31} = 20$<br>$F(x)' = F(x) + d_{13}\, x = 590$ | | | | | |

## 4.7. Teorija igara [4]

Mnogi modeli odlučivanja postavljaju velike zahtjeve u pogledu znanja u određenim situacijama u kojima se mora donijeti neka odluka. Ti zahtjevi se posebno odnose na obujam i kvalitetu informacija. Deterministički modeli prešutno pretpostavljaju potpune i sigurne informacije. Ta pretpostavaka može u određenim situacijama biti točna, ali u mnogim situacijama, u kojima je donošenje odluke orijentirano na budućnost, takva pretpostavka nije dopuštena Nepoznavanje prirodnih procesa ili personalnih interakcija rezultira stohastičkim rizikom, a u modelskom pristupu zahtijeva tzv. modele neizvjesnosti.

Ako se neizvjesne situacije odlučivanja interpretiraju kao igre koje igra donosilac odluke s obzirom na svoje aktivnosti protiv neutralnog ili protiv neopredijeljene strane, tada se govori o igrama protiv prirode, odnosno o igrama protiv protivnika. Igre protiv prirode, protiv sreće, vremenskih okolnosti ili defekta masovnih proizvoda odvijaju se često uz pomoć posebnih pravila odlučivanja.

Osnovna premisa za primjenu teorije igara protiv protivnika (teorija strategijskih igara) je da se u svjesnim odlukama protivnog igrača izvedenim iz njegovog interesa kao reakcije na alternativni potencijal suigrača koji se može naslutiti.

Primijene li se pravila strategijskih igara na igre protiv prirode, dolazi se do optimalnog rezultata koji je obično bliži donjem pragu, jer se pretpostavlja da je inteligentan protivnik jači partner od indiferentne prirode. Neizvjesnost u igri protiv protivnika proizlazi iz nepoznavanja ponašanja protivnika i njegove orijentacije u pogledu cilja. Ovako postavljena teza akcija – reakcija prihvatljiva je ako reakcija u igri ima stvarnog odraza u igri, što praktički znači da u igri postoje zavisni odnosi između obaju igrača. Ta zavisnost definira se kao konflikt između, osoba.

#### 4.7.1.1. *Elementi strategijske igre*

Slika jedne konfliktne situacije kao igre protiv protivnika zahtijeva potpuni i precizni modelsko – teoretski opis i razgraničenje igrača koji sudjeluju, pravila igre, raspoložive strategije kao i konsekvence koje proizlaze iz sukoba strategija.

Igra je skup pravila i dogovora po kojima se igrači moraju ravnati. Kao igrač se može pojaviti pojedinac, grupa, država itd. Važno je samo da postoji jedinstveni i samostalni donosilac odluke. Igrač teži ostvariti postavljeni cilj ili sustav ciljeva uz dosljedno odabiranje strategijskih mogućnosti. Pri tome se igrač mora pridržavati pravila igre koja, između ostalog, utvrđuju s kojim informacijama o cilju, ponašanju i strategijskom prostoru protivnika može računati. Prostor odlučivanja sastoji se iz mnoštva strategijskih kombinacija. Strategija se uobičajeno definira kao skup uputa za igrača, koji sadrži sve poteze koji bi mogli doći u obzir u toku jedne partije. Pod partijom se podrazumijeva svaka pojedina realizacija jedne igre. Svako sučeljavanje strategije dovodi do posljedica koje se mogu izraziti u dimenzijama kao što su prestiž, novac, utjecaj itd.

#### 4.7.1.2. *Igre između dvije osobe sa sumom nula*

Najjednostavniji i najpoznatiji tip igara su igre između dvije osobe sa sumom nula. Dva igrača igraju jedan protiv drugog. Izraz „suma nula“ označava da je i zajednički dobitak obaju igrača jednak nuli. To znači da je dobitak jednog igrača jednak gubitku ili negativnom dobitku drugoga. Ako u takvoj igri jedan igrač dobije $k$ jedinica, njegov protivnik gubi $k$ jedinica, pa je $k + (-k) = 0$.

Najjednostavnije igre između dvaju igrača sa sumom nula imaju sljedeća pravila: U svakom potezu svaki igrač mora donijeti jednu jedinu odluku. Izbor prvog igrača ne smije biti poznat drugom igraču prije nego što se ovaj drugi odluči za vlastiti izbor. Dakle, svaki igrač mora izvršiti svoj izbor, a da pri tom ne zna za izbor drugog igrača, iako su mu poznate sve mogućnosti izbora protivnika. Visina dobitka, odnosno gubitka, dana je matricom plaćanja koja se odnosi na sve izbore u igri.

Cilj teorije igara je određivanje optimalne strategije. Razlikuju se čista i mješovita strategija. Kod čiste strategije svaki igrač vrši određeni izbor koji ostaje nepromijenjen u ponavljanju igre. Najčešće se odabire mješovita strategija. Ona se sastoji od fiksnih udjela različitih mogućnosti izbora. Npr. igrač se odlučuje da od tri mogućnosti izabere prvu u 30% poteza, drugu u 25% poteza, a treću u 45% poteza.

#### 4.7.1.3. *Igra sa sedlom*

Kod posebnih struktura matrica plaćanja u igri postoji sedlo ili sedlasta točka. Pretpostavi li se da su oba igrača inteligentna (u smislu teorije igara), nastojat će svaki odabrati onu strategiju koja mu donosi maksimalni dobitak, respektirajući pri tome strategiju protivnika. Budući da je protivnik inteligentan suparnik i nastoji postići što je moguće veći dobitak, igrač A mora uključiti rezoniranje suparnika B u izboru vlastite strategije. Svaki igrač odabire strategiju kojom maksimizira vlastiti minimalni dobitak i istovremeno minimizira maksimalni gubitak. To je tzv. minimaks kritetrij, poznat pod nazivom von Neumannov kriterij.

Rješenje problema može se pronaći na jednostavan način pronalazi se minimalni element svakog reda i upisuje se s desne strane tabele, a podno svakog stupca upisuje se maksimalni element svakog stupca. Zatim se pronalazi najveći minimalni element i najmanji maksimalni element to je sedlasta točka [Tablica 19]

Tablica 19. Matrica plaćanja za igrača A

| Strategije | $B_1$ | … | $B_n$ | | |
|---|---|---|---|---|---|
| $A_1$ | | | | $S_1$ | |
| ⋮ | | plaćanja | | ⋮ | ←max |
| $A_n$ | | | | $S_n$ | |
| | $S_1$ | … | $S_n$ | | |
| | | ↑<br>min | | | |

Ako postoji sedlo, igrači ne trebaju skrivati svoje strategije jedan od drugoga. Ako B zna da je A izabrao optimalnu strategiju, nikakvu prednost ne može iz toga izvući, jer ne može umanjiti dobitak protivnika. Takva svojstva ima svaka igra sa sedlom.

Matematički se takvi problemi označavaju analogno linearnom optimiranju kao minimax ili maxmin problemi, a odgovarajuće strategije se nazivaju minimaks strategije.

#### 4.7.1.4. *Igre bez sedla*

Mnogo je teže odrediti optimalnu strategija kod igara bez sedla. Dva igrača A i B igraju, u tablici [Tablica 20] je matrica plaćanja za igrača A. Matrica plaćanja za igrača B dobije se na taj način da se matrica plaćanja igrača A pomnoži s -1 i transponira.

Tablica 20. Matrica plaćanja igrača A

| A/B | $F_1$ | … | $F_n$ | | |
|---|---|---|---|---|---|
| $E_1$ | | | | $S_1$ | |
| ⋮ | | plaćanja | | ⋮ | max! |
| $E_n$ | | | | $S_n$ | |
| | $S_1$ | … | $S_n$ | | |
| | | min! | | | |

Svaki igrač želi, po mogućnosti, postići najveći dobitak. Za optimalno rješenje potrebne su mješovite strategije, za razliku od čistih strategija koje se odnose na odluke koje se ne mijenjaju. Problem se sastoji u tome da se odredi učestalost kojom će se naizmjence primjenjivati vlastita strategija. Mora se pretpostaviti da je moguće zadržati vlastitu strategiju u tajnosti tako dugo dok se protivnik ne odluči za svoju. Za izračunavanje optimalnih mješovitih strategija može se koristiti linearno programiranje. Od više tehnika predloženih za rješavanje ovog problema koristit ćemo onu koju je pokazao Mtiller-Merbach.

Najprije treba postaviti funkciju cilja i restrikcije. Recimo da se to provede za igrača B. Želi se maksimalni gubitak $V$ minimizirati za igrača B. S varijablama $y_1, ..., y_n$ označit ćemo relativnu učestalost izbora $F_1, .., F_n$. Ako igrač A odabere $E_n$, gubitak za igrača B ne smije biti veći od $V$. Za varijable $y_1, ..., y_n$ vrijedi zahtjev nenegativnosti. $V$ je slobodna varijabla. Treba minimizirati gubitak $V$ ili maksimirati negativan gubitak koji je u stvari jednak dobitku.

$$D = -V \max! \qquad (29)$$

Te restrikcije unose se u simpleks tabelu. Simpleks metoda daje rješenje za ovaj problem. Identitet ovih dvaju problema bazira se na teoremu dualnosti. Prema tome, ako se problem igrača B označi kao primami problem, a problem igrača A kao dualni, rješenje se pronalazi u optimalnom rješenju primarnog problema i to u zadnjem redu tabele ispod dopunskih varijabli.

#### 4.7.1.5. *Dominacija*

Pojedine strategije mogu dominirati nad drugima. To znači da je neka strategije uvijek lošija za igrača od neke druge, bez obzira na to kojom strategijom nastupa protivnik. Prije izračunavanja optimalne strategije vrši se redukcija po dominaciji. Na taj je način igra pojednostavljena bilo da se radi o igri sa sedlom ili bez sedla.

#### 4.7.1.6. *Općeniti poučci za matrične igre*

Matrica plaćanja [Tablica 21] može biti tipa $m$ x $n$. Igrač A ima na raspolaganju $m$ strategiju, a igrač B $n$ strategiju. Konačne igre s takvom matricom zovu se još i matrične igre.

Tablica 21. Matrica plaćanja

| | Strategija | I | II | $n$ |
|---|---|---|---|---|
| | I | $a_{11}$ | $a_{11}$ | $a_{1n}$ |
| | II | $a_{21}$ | $a_{22}$ | $a_{2n}$ |
| A | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| | $m$ | $a_{m1}$ | $a_{m2}$ | $a_{mn}$ |

Pomoću matematičkih metoda mogli bi se dokazati sljedeći poučci koje je naveo Ackoff:

1. Svaka matrična igra ima određenu vrijednost $v$. Ta je vrijednost jednoznačna.
2. Za igrača A postoji najbolja strategija, to znači postoje relativne učestalosti $x_1$, $x_2$, …, $x_m$ uz uvjet $x_1 + x_2 + \ldots + x_m = 1$, tako da odabiranjem strategije I s učestalošću $x_1$, strategije II s učestalošću $x_2$, strategiju $m$ s učestalošću $x_m$ igrač A osigurava minimalni dobitak $v$ pri čemu je $v$ vrijednost igre.
4. Analogno, postoji i za igrača B najbolja strategija (30) sa sljedećim svojstvima: odabere li igrač B strategiju I, II, …, $n$ s učestalošću $y_1$, $y_2$, …, $y_n$, može biti siguran da će izgubiti najvišeg iznosu $v$.

$$Y = (y_1, y_2, \ldots, y_n) \qquad \text{uz uvjet } y_j = 1 \tag{30}$$

Treba uočiti da za matričnu igru koja ima sedlo u polju ($i_o j_o$) vrijedi sljedeće rješenje (31):

$$X_i = \begin{cases} 0 & i \neq i_0 \\ 1 & i = i_0 \end{cases} \qquad Y_j = \begin{cases} 0 & j \neq j_0 \\ 1 & j = j_0 \end{cases} \qquad v = a_{i_0 j_0} \tag{31}$$

### 4.7.1.6.1. *Općenita metoda rješenja matrične igre*

Može se pokazati da se nepoznanice $x_1$, $x_2$, …, $x_m$, $y_1$, $y_2$, …, $y_n$, $v$ mogu izračunati iz sljedećeg sustava jednadžbi i nejednadžbi (32) - (35):

$$x_1 + x_2 + \ldots + x_m = 1 \qquad x_i \geq 0 \tag{32}$$

$$y_1 + y_2 + \ldots + y_n = 1 \qquad y_j \geq 0 \tag{33}$$

$$a_{1j}x_1 + a_{2j}x_2 + \ldots + a_{mj}x_m \geq v \qquad za\ j = 1,2, \ldots, n \tag{34}$$

$$a_{i1}y_1 + a_{i2}y_2 + \ldots + a_{in}y_n \leq v \qquad za\ i = 1,2, \ldots, m \tag{35}$$

Uvjet (34) sastoji se od $n$, a uvjet (35) od $m$ nejednadžbi. Iz čega proizlazi $m + n + 1$ nepoznanica i $m + n + 2$ uvjeta, s dodatnim ograničenjima $x_i \geq 0$ i $y_j \geq 0$. Zadnji i predzadnji uvjet mogu biti jednadžbe ili nejednadžbe. Igra može imati beskonačno mnogo rješenja $x_i$ i $y_j$. zadatci se mogu riješiti algebarskim načinom, grafičkom metodom, matričnim računom.

### 4.7.1.6.2. *Algebarsko rješenje*

Polazi se od činjenice da svaka matrična igra ima vrijednost $v$, koja uvijek postoji i jednoznačna je. Nepoznanice se pokušavaju pronaći rješavanjem gore navedenih uvjeta.

Pokušava se pronaći takva vrijednost za $v$ koja zadovoljava $n$ uvjeta. Prvi je korak u tome da se predzadnja i zadnja relacija prihvate kao jednadžbe.

Općeniti postupak sastoji se u tome da se jedan uvjet uzme u obliku nejednadžbe, a ostali u obliku jednadžbe. Ako se opet ne dođe do rješenja, uzimaju se neki drugi uvjeti kao nejednadžba, a ostali ponovo zadržavaju oblik jednadžbe. To se provodi tako dugo dok se ne dođe do rješenja. Poučak koji pojednostavljuje računanje (36) i (37):

$$a_{1j}x_1 + a_{2j}x_2 + \ \dots + a_{mj}x_m > v \qquad y_j = 0 \tag{36}$$

$$a_{i1}y_1 + a_{i2}y_2 + \ \dots + a_{in}y_n < v \qquad x_i = 0 \tag{37}$$

#### 4.7.1.7. *Igre protiv prirode*

Kod osnovnog kriterija teorije igara pretpostavlja se da je suigrač inteligentan (u smislu teorije igara), što znači da će svaki otklon od optimalne strategije suigrač pretvoriti u vlastitu prednost. Ako se igra protiv prirode (u smislu teorije igara) neinteligentnog igrača, ne može se prihvatiti. Neumannovo pravilo: mogući različiti pristupi utvrđivanju racionalnog pravila igre.

Kriteriji igre protiv prirode:

1. Laplaceov kriterij: vjerojatnosti različitih stanja prirode su nepoznati. Uzima se kao da su vjerojatnosti jednake i igrač izabire red za koji vrijedi (38):

$$\max_i \quad \frac{1}{n}[a_{i1} + a_{i2} + \ \dots + a_{in}] \tag{38}$$

   Odnosno onaj red za koji je očekivani dobitak maksimalan. Daljnja istraživanja usmjerena su na vjerojatnosti različitih mogućih stanja prirode. Ako su te vjerojatnosti poznate, uvrštavaju umjesto $1/n$ u gornji izraz.

5. Hurwiczov kriterij: uvažava se optimizam igrača brojem $\alpha$, $0 \leq \alpha \leq 1$. Neka je $A_i$ najveći, odnosno $a_i$ najmanji element jednog reda $i$. Tada se bira onaj red kojem odgovara (39):

$$\max_i \quad [\alpha A_i + (1-\alpha)a_i] \tag{39}$$

6. Savagov kriterij: ovaj kriterij zahtijeva da se definira nova matrica s elementima (40):

$$\boldsymbol{A}_{ij} = \max_k a_{kj} - a_{ij} \tag{40}$$

koji pokazuje raskorak između efektivno realiziranog dobitka $a_{ij}$ i onog koji bi se mogao postići, tj. max $a_{kj}$ kada bi stanje prirode bilo poznato. Ta se matrica naziva matrica žaljenja. Igrač A bira onaj red za koje je maksimalno "žaljenje" minimalno (41):

$$\min \left[ \max_{k} \boldsymbol{A}_{kj} \right] \qquad i = 1,2,\dots,m \qquad j = 1,2,\dots,n \tag{41}$$

## 4.8. Mrežno programiranje

Kod mrežnog programiranja (MP) sve se aktivnosti, potrebne za provođenje jednog projekta, prikazuju u grafičkom obliku pazeći pri tome da se usklade sve vremenske zavisnosti između pojedinih aktivnosti. Slika koja tako nastaje je mreža pa otuda naziv mrežno programiranje.

U primjeni MP razlikuju se tri faze provedbe koje su međusobno zavisne:

1. planiranje strukture,
2. analiza vremena,
3. kontrola i upravljanje.

Metode koje su danas poznate pod nazivom MP-a nazivaju se metoda kritičnog puta i PERT metoda (eng. *Project Evaluation and Review Technique*).

## 4.9. Teorija grafova

Teorija grafova je matematička disciplina koja se koristi u planiranju radnih organizacija. Graf je mreža sastavljena od čvorova koje povezuju bridovi. Bridovi mogu biti orijentirani u jednom smjeru (tada se nazivaju lukovi) i označavaju se strjelicom. U tom se slučaju govori o orijentiranim grafovima. Kada bridovi nisu orijentirani radi se o neorijentiranim grafovima. Ako su čvorovi međusobno direktno ili indirektno povezan nizom bridova, imamo povezani graf. Graf s naziva potpunim ako je svaki čvor povezan jednim bridom direktno sa svakim drugim čvorom. Grafovi mogu biti povezani u obliku stabla i u obliku lanca. Postoje još mnoge druge vrste grafova.

Važno područje primjene teorije grafova odnosi se na izračunavanje najkraćeg i najdužeg puta u grafu. Uz izračunavanje optimalnih putova u grafu važnu ulogu u praktičnoj primjeni teorije grafova ima određivanje maksimalnog tijeka.

## 4.10. Problem zaliha

Problem vođenja zaliha javlja se u svim onim djelatnostima gdje je potrebno donijeti odluku o količini sirovina i materijala, poluproizvoda, finalnih proizvoda, likvidnih financijskih sredstava i slično koji moraju biti odmah raspoloživi za neki proces koji ih konzumira.

Motivi za povećanje nivoa zaliha nalaze se u sigurnosti da će se proces moći odvijati nesmetano. Zaustavljanje procesa zbog nedostatnih zaliha izaziva troškove, gubitak dobiti ili neke druge nepovoljne posljedice. Osim toga, svaki postupak za obnavljanje zaliha kao što je izdavanje narudžbe, pokretanje proizvodnje, dostava i slično najčešće izaziva određene fiksne troškove. S druge strane, držanje zaliha stvara troškove skladištenja koji nastanu zbog gubitka vrijednosti robe na skladištu, troškova za skladišni prostor, i što je često najvažnije, zamrznutog kapitala koji ne donosi nikakvu dobit. Dakle, problem zaliha predstavlja značajno pitanje u ekonomiji radne organizacije, a politika zaliha je dio poslovne politike i važno područje djelovanja organa upravljanja i rukovođenja. [1] [4]

Postoji velik broj matematičkih modela koji opisuju različite slučajeve vođenja zaliha. Modeli mogu biti deterministički i stohastički s dopuštenim ili nedopuštenim manjkovima robe. U nekim se modelima manjkovi moraju nadoknaditi a u drugima ne itd.

## 4.11. Modeli repova čekanja

Dobro su poznati problemi čekanja usluživanja ili pružanja usluga. Javljaju se u društvu, transportu, komercijali, komunikacijama, poslovanju i industriji [Tablica 22]. Za opis sustava čekanja potrebno je definirati ulazni, izlazni proces i disciplinu usluge

Tablica 22. Primjeri sustava čekanja [4]

| Sustav repa | Jedinice (ulazni proces) | Poslužitelj usluge (izlazni proces) |
|---|---|---|
| Zdravstvo | Ljudi | Medicinsko osoblje, ambulante, kreveti |
| Transfuzija krvi | Pristignute doze krvi | Transfuzija krvi pacijentima |
| Sudstvo | Parnice | Suci |
| Zakonodavstvo | Objave | Pravnici |
| Banka | Ljudi | Službenici u bankama |
| Supermarket | Ljudi | Blagajnici |
| Benzinska postaja | Ljudi, automobili | Crpka, poslužitelj |
| Automat za prodaju | Ljudi | Automat |
| Autocesta | Automobili | Prometni signali |

| Sustav repa | Jedinice (ulazni proces) | Poslužitelj usluge (izlazni proces) |
|---|---|---|
| Dokovi | Kamioni, brodovi | Zaposleno osoblje |
| Zračna luka | Zrakoplovi | Pista |
| Parkiranje | Automobili | Parkirališta |
| Javni transport | Ljudi | Autobusi, tramvaji, vlakovi |
| Telefon | Ljudi | Oprema, linije, signal |
| Materijalno poslovanje | Sirovine, proizvodi | Konvejer |
| Računalna oprema | Program | Računalo |

Osnovni elementi modela repova ovise o [4]:

1. distribuciji ulaza,
2. distribuciji vremena usluge,
3. sustavu posluživanja,
4. disciplini usluge,
5. dužini repa,
6. izvorima pozivanja,
7. ljudskom ponašanju.

Ulazni proces je proces pristizanja, a pristizanja predstavljaju korisnike. U svim modelima pretpostavlja se da je moguće samo jedno pristizanje u danom trenutku. U nekim slučajevima ta pretpostavka je nerealna, ako dolazi do više od jednog pristizanja u danom trenutku govori se o grupnom pristizanju. Pretpostavlja se da na proces pristizanja ne utječe broj jedinica koje su već prisutne, odnosno bez obzira nalazi li se u nekom sustavu 100 ili 5 jedinica pretpostavlja se da proces reguliranja dolazaka ostaje isti.

Postoje dvije situacije u kojima ne vrijedi navedena pretpostavka:

1. pristizanja su izvučena iz malog skupa,
2. nova pristizanja ne uspijevaju ući u sustav.

Ako vrijedi pretpostavka da na proces pristizanja ne utječe broj jedinica koje su već prisutne, onda se proces pristizanja opisuje određivanjem distribucije vjerojatnosti koja regulira sukcesivna pristizanja.

Za opisivanje izlaznog procesa potrebno je definirati distribuciju vjerojatnosti koja osigurava reguliranje posluživanja. Često se pretpostavlja da distribucija vremena posluživanja ne ovisi o broju već prisutnih jedinica, što znači da poslužitelj neće raditi brže ako je prisutno više jedinica za posluživanje.

Proučit će se dva sustava posluživanja:

1. paralelno posluživanje,
2. serijsko posluživanje.

Paralelno posluživanje podrazumijeva da svi poslužitelji pružaju istu vrstu usluge, što znači da svaka jedinicu koja pristigne treba samo jednog poslužitelja za dovršetak usluge. Serijsko posluživanje podrazumijeva da svako pristizanje prolazi nekoliko poslužitelja za dovršetak usluge.

Disciplina usluge ili disciplina reda određuje redoslijed odabiranja jedinica iz reda čekanja za početak usluge. Najčešće discipline usluge su:

1. FCFS – prvi došao prvi uslužen, eng. *First come, first served*
2. LCFS – zadnji došao prvi uslužen, eng. *Last come, first served*
3. SIRO – uslužen po slučajnom redoslijedu, eng. *Service in random order*

Najčešća je FCFS disciplina u kojoj su jedinice uslužene po redoslijedu pristizanja, posebna kategorija ove discipline je prioritetno usluživanje. U tom slučaju se jedinice prvo klasificiraju u određene kategorije, svakoj kategoriji je pridodana razina prioriteta, daljnji redoslijed posluživanja određen je FCFS disciplinom. Kod LCFS discipline jedinice koje posljednje pristignu biti će prve uslužene, ukoliko redoslijed pristizanja nema utjecaja na redoslijed posluživanja govori se o SIRO disciplini.

Način ka koji jedinice odabiru red u koji će stati također igra ulogu u teoriji redova. Najčešće se odabire najkraći red, ali negdje vrsta usluge predodređuje odabir reda. Trebaju se razmotriti situacije kada se jedinice prebacuju iz jednog u drugi red.

### *4.11.1. Proces rađanja i umiranja* [3]

Proces rađanja i umiranja koristit će se za definiranje različitih sustava repova. Proces rađanja i umiranja je kontinuirani stohastički proces za koji je stanje sustava u bilo kojem trenutku cijeli pozitivni broj. Rađanja i umiranja predstavljaju dolaske i odlaske (kraj usluge) entiteta u ili iz sustava.

Neka je stanje sustava repova u vremenu $t$ definirano brojem entiteta prisutnih u sustavu repa u vremenu $t$. Za slučaj kada je $t = 0$ stanje sustava biti će jednako broju entiteta koji su već u početku bili prisutni u sustavu. $P_{ij}(t)$ predstavlja vjerojatnost da će u vremenu $t$ biti prisutno $j$ entiteta u sustavu i da će u vremenu $t = 0$ biti prisutno $i$ entiteta. Za velike vrijednosti $t$, $P_{ij}(t)$ se približava vjerojatnosti $p_j$ koja je neovisna o početnom stanju $i$. Vjerojatnost $p_j$ naziva se stabilno ili ravnotežno stanje sustava u stanju $j$. za sustave repova koji će se ovdje obraditi $p_j$ se može zamisliti kao vjerojatnost da će u nekom trenutku u dalekoj budućnosti $j$ entiteta biti prisutno. Može se reći da $p_j$ predstavlja za vrijeme u dalekoj budućnosti djelić vremena u kojem je $j$ entiteta prisutno.

U većini sustava repova vjerojatnost $P_{ij}(t)$ za male vrijednosti $t$ jako ovisi o $i$ (broju već prisutnih entiteta). Pa tako će se npr. za jako mali $t$, vrijednosti $P_{50,1}(t)$ i $P_{1,1}(t)$ znatno razlikovati. Ali ako se sustav nalazi u stabilnom stanju $P_{50,1}(t)$ i $P_{1,1}(t)$ će biti blizu vrijednosti $p_j$. Pitanje je koliko $t$ mora biti velik da bi se postiglo stabilno stanje. Stanje u kojem se $P_{ij}(t)$ nalazi prije nego li uđe stanje stabilnosti naziva se prijelazno ponašanje sustava repova. Za sada će se razmatrati samo stabilno stanje sustava i vrijednost $p_j$.

Proces rađanja i umiranja u potpunosti je definiran vrijednostima:

- $\lambda_j$ – stopa rađanja i
- $\mu_j$ – stopa umiranja

Ako se proces rađanja i umiranja nalazi u stanju $j$ u vremenu $t$, onda je kretanje procesa određeno sljedećim zakonitostima:

1. Vjerojatnost da će se dogoditi rađanje između vremena $t$ i $t + \Delta t$ iznosi $\lambda_j \Delta t + o(\Delta t)$[3], gdje je $\lambda_j$ je stopa rađanja u stanju $j$. Stanje sustava nakon rađanja se povećava za 1, sa $j$ na $j + 1$
2. Vjerojatnost da će se dogoditi umiranje između vremena $t$ i $t + \Delta t$ iznosi $\mu_j \Delta t + o(\Delta t)$, gdje je $\mu_j$ stopa umiranja u stanju $j$. stanje sustava nakon umiranja smanjuje se za 1, sa $j$ na $j - 1$. Jednakost $\mu_0 = 0$ mora uvijek vrijediti jer bi u suprotnom nastupio negativan događaj.
3. Rađanja i umiranja međusobno su neovisna

Navedene zakonitosti se koriste za izračun vjerojatnosti da će se u vremenu između $t$ i $t + \Delta t$ pojaviti događaj (rađanja ili umiranja).

#### 4.11.1.1. *Veza procesa rađanja – umiranja i eksponencijalne distribucije*

Većina sustava repova sa eksponencijalnom distribucijom vremena dolazaka i vremena usluživanja mogu se modelirati korištenjem procesa umiranja i rađanja.

[3] $\lim_{\Delta t \to 0} \frac{o(\Delta t)}{\Delta t} = 0$

Da bi se to ilustriralo pogledati će se sustav repa jednog poslužitelja i beskonačno mnogo dolazaka. Kod ovog sustava vremena međudolazaka eksponencijalno su distribuirana parametrom $\lambda$ i vremena trajanja usluživanja eksponencijalno su distribuirana parametrom $\mu$. Ako je broj entiteta u vremenu $t$ jednak $j$ onda vjerojatnost rađanja u intervalu od $t$ do $t + \Delta t$ neće ovisiti o dužini vremena koje se provede u stanju $j$. Stoga se vjerojatnost da će se dogoditi rađanje u intervalu između $t$ i $t + \Delta t$ može odrediti kao da se rađanje dogodilo upravo u trenutku $t$ (42):

$$\int_0^{\Delta t} \lambda e^{-\lambda t} dt = 1 - e^{-\lambda \Delta t} \tag{42}$$

Razvojem funkcije (42) u Taylorov red dobiva se (43):

$$e^{-\lambda \Delta t} = 1 - \lambda \Delta t + o(\Delta t) \tag{43}$$

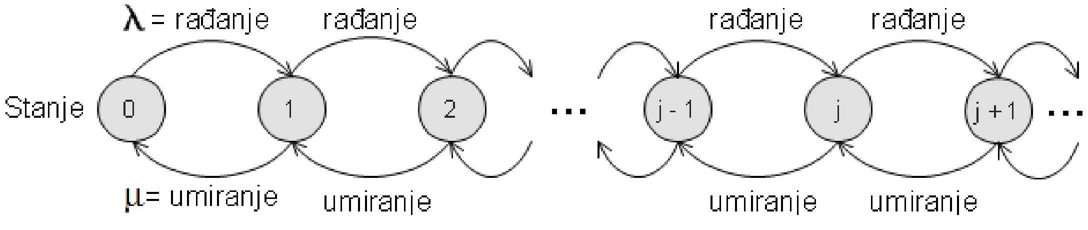
Uvrštavanjem jednadžbe (43) u (42) dobit će se $\lambda_j \Delta t + o(\Delta t)$, što predstavlja vjerojatnost da će se u vremenu između $t$ i $t + \Delta t$ dogoditi rađanje. Iz navedenog se može zaključiti da je stopa rađanja u stanju $j$ jednaka stopi dolazaka $\lambda$.

Kako bi se odredila stopa umiranja u vremenu $t$, treba uzeti u obzir da ukoliko je stanje nula u vremenu $t$ onda se ne vrši nikakva usluga, stoga ne može doći do završetka usluge u vremenu između $t$ i $t + \Delta t$. Iz toga slijedi da je $\mu_0 = 0$.

Ako je u vremenu $t$, $j \geq 1$, onda je jasno da se samo jedan entitet uslužuje (s obzirom da je prisutan samo jedan uslužitelj S = 1). Vjerojatnost da će entitet završiti s uslugom u vremenu između $t$ i $t + \Delta t$ iznosi (44):

$$\int_0^{\Delta t} \mu e^{-\mu t} dt = 1 - e^{-\mu \Delta t} = \mu \Delta t + o(\Delta t) \tag{44}$$

Iz toga slijedi da je za $j \geq 1$, $\mu_j = \mu$. Konačno, ako se pretpostavi da su dolasci i završetci usluživanja međusobno neovisni onda se može zaključiti sustav repa (*M/M/1*) : (*GN/∞/∞*) proces umiranja i rađanja [Slika 10].




Slika 10. Shema stopa rađanja i umiranja za sustav repa [3]

4.11.1.2. *Izvođenje vjerojatnosti stabilnog stanja za proces rađanja i umiranja*

Potrebno je povezati $P_{ij}(t + \Delta t)$ sa $P_{ij}(t)$, kako bi se to napravilo treba uočiti četiri načina da stanje bude $j$ u vremenu $t + \Delta t$ [Tablica 23].

Ako sustav nije u stanju $j - 1$, $j + 1$ ili $j$ koje vodi u stanje $j$ u vremenu $t + \Delta t$ to znači da će u vremenu između $t$ i $t + \Delta t$ nastupiti jedan događaj (rađanje ili umiranje), a vjerojatnost da se to desi je $o(\Delta t)$, stoga se može pisati (45):

$$P_{ij}(t+\Delta t) + (49) + (50) + (51) + (52) \tag{45}$$

Nakon uvrštavanja vrijednosti iz tablice [Tablica 23] u (45) dobiti će se (46):

$$P_{ij}{}'(t) = \lambda_{j-1}P_{i,j-1}(t) + \mu_{j+1}P_{i,j+1}(t) - P_{i,j}(t)\mu_j - P_{i,j}(t)\lambda_j \tag{46}$$

Stabilno stanje $p_j$ postiže se za (47):

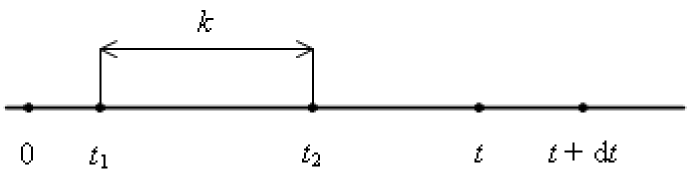
$$p_j = \lim_{t\to\infty} P_{ij}(t) \tag{47}$$

Tablica 23. Izračun vjerojatnosti da u vremenu $t + \Delta t$ stanje $j$ [3]

| **Stanje u $t$** | **Stanje u $t + \Delta t$** | **Vjerojatnost tog slijeda događaja** |
|---|---|---|
| $j - 1$ | $j$ | $P_{i,j-1}(t)\left(\lambda_{j-1}\Delta t + o(\Delta t)\right)$ (48) |
| $j + 1$ | $j$ | $P_{i,j+1}(t)\left(\mu_{j+1}\Delta t + o(\Delta t)\right)$ (49) |
| $j$ | $j$ | $P_{i,j}(t)\left(1 - \mu_j\Delta t - \lambda_{j-1}\Delta t - 2o(\Delta t)\right)$ (50) |
| drugo stanje | $j$ | $o(\Delta t)$ (51) |

### *4.11.2. Poissonov proces [4]*

Pretpostavlja se da je jedan sustav u određenom stanju $k$ [Slika 11]. To je stanje opisano time što se u vremenu između 0 i $t$ zbiva točno $k$ događaja. Vjerojatnost da će se u vremenu između $t$ i $t + \mathrm{d}t$ nastupiti daljnji događaj, to znači da će sustav prijeći u stanje u stanje $k + 1$, neka je $\lambda$ konstanta. Prijelaz u više stanje $k + 2$, $k + 3$ znači da nastup drugog i trećeg događaja unutar $t$ i $t + \mathrm{d}t$ treba zanemariti.




Slika 11. Prikaz sustava u stanju *k*

Uz te pretpostavke računaju se vjerojatnosti da se sustav u vremenu *t* + d*t* nalazi u stanju *n*. Ta se vjerojatnost sastoji:

a) iz vjerojatnosti da se sustav u vremenu *t* nalazi u stanju *n* – 1 i da u vremenu *t* + d*t* nastupa jedan događaj,

b) iz vjerojatnosti da je sustav u vremenu *t* bio u stanju *n* i da u vremenu između *t* + d*t* nije nastupio nijedan događaj.

Na osnovu toga može se pisati (52):

$$p_n(t + \mathrm{d}t) = p_{n-1}(t)\lambda \mathrm{d}t + p_n(t)(1 - \lambda)\mathrm{d}t \tag{52}$$

Ili preko granične vrijednosti d*t* → 0, (53):

$$p_n(t) = \lambda p_{n-1}(t) + \lambda p_n(t) \tag{53}$$

Rješenje diferencijalne jednadžbe (54):

$$p_n(t) = e^{-\lambda t}\frac{(\lambda t)^n}{n!} \tag{54}$$

Za *t* = 1 dobiva se izraz poznat kao Poissonov zakon (55):

$$p_n = e^{-\lambda}\frac{\lambda^n}{n!} \tag{55}$$

Po ovom zakonu (55) računa se vjerojatnost nastupa *n* događaja u jedinici vremena uzimajući u obzir gore navedene pretpostavke.

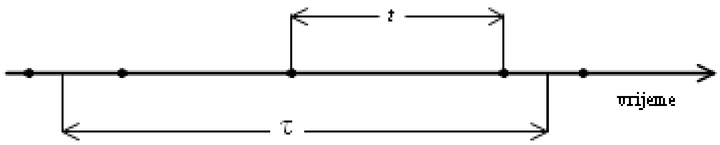
Srednja vrijednost (56) i varijanca (57) po jedinici vremena su:

$$E(n) = \bar{x} = \lambda \tag{56}$$

$$Var(n) = \sigma^2 = \lambda \tag{57}$$

Vjerojatnost da će interval vremena između dva događaja koji slijede jedan za drugim biti jednak ili veći od $\sigma$ [Slika 12], jednak je vjerojatnosti da unutar intervala $\sigma$ ne nastupi nijedan događaj (58):

$$p_0 = e^{-\lambda\sigma} \tag{58}$$




Slika 12. Prikaz intervala $\sigma$

Ako sa $F(\sigma)$ označimo funkciju od $\sigma$, možemo tu vjerojatnost napisati u obliku $1 - F(\sigma)$, iz čega je:

$$F(\sigma) = 1 - e^{-\lambda\sigma} \tag{59}$$

A funkcija gustoće vjerojatnosti je:

$$f(\sigma)\mathrm{d}\sigma = e^{-\lambda\sigma}\mathrm{d}(\lambda\sigma) \tag{60}$$

Na taj način su u okviru Poissonovog procesa intervali između dva događaja koji slijede jedan za drugim raspoređeni prema eksponencijalnom zakonu.

Srednja vrijednost (61) i varijanca (62) su:

$$E(\sigma) = \frac{1}{\lambda} \tag{61}$$

$$Var(\sigma) = \frac{1}{\lambda^2} \tag{62}$$

### *4.11.3. Kombinirani dolasci i usluge [4]*

Modeli repova čekanja klasificirani su kroz procese dolazaka, usluge, broj uslužnih mjesta, veličinu prostora za čekanje i pravila usluživanja. Obilježavanje glavnih karakteristika repova čekanja standardizirano je Kendallovom notacijom (63):

$$(a/b/c) : (d/e/f) \tag{63}$$

Gdje slova predstavljaju osnovne elemente modela repova čekanja:

- $a$ = distribucija dolazaka,
- $b$ = distribucija usluge,
- $c$ = broj paralelnih usluga,
- $d$ = disciplina u repu,
- $e$ = maksimalan broj dozvoljen u sustavu,
- $f$ = veličina izvora pozivanja.

Standardizirano obilježavanje dolazaka i usluga ($a$ i $b$) zamjenjuje se oznakama:

- M = Poissonova ili Markovljeva distribucija dolaska ili dolaska ili ekvivalentna eksponencijalna distribucija međudolazaka ili usluge
- D = konstantan ili deterministički međudolazak ili vrijeme usluge
- $E_k$ = Erlangova ili gama ($\gamma$) distribucija međudolazaka ili usluge s parametrom $k$
- GI = općenita nezavisna distribucija dolazaka
- G = općenita distribucija dolazaka ili vremena usluge

Razmotrimo primjer Kendallove notacije (64):

$$(M/D/10) : (GD/N/\infty) \tag{64}$$

Iz gornje oznake može se zaključiti da se radi o modelu repova čekanja u kojem postoji Poissonova distribucija dolazaka s konstantnim vremenom usluge i 10 paralelnih mjesta usluge. Disciplina repa općenito je GD u smislu da može biti FCFS, LCLS, SIRO ili neki drugi postupak koji poslužitelji koriste za odabir jedinica iz repa za početak usluge. Bez obzira koliko jedinica dolazi sustav može imati maksimalno N jedinica. Možemo vidjeti da oznaka beskonačnosti na kraju označuje neograničen kapacitet izvora.

### *4.11.4. Opći Poissonov model [4]*

Ovdje će se opisati osnove općeg modela repa koji se temelji na Poissonovom procesu s dolascima i uslugama ovisnim o stanju sustava. Stopa dolazaka i usluga ovisna je o stanju sustava. Pretpostaviti će se sustav s ukupno $N$ poslužitelja, stopa po kojoj poslužitelji izlaze iz rada funkcija je broja poslužitelja koji rade. Ako je $\lambda$ stopa otkazivanja poslužitelja, onda je stopa otkazivanja za čitavi sustav koja ima $n$ poslužitelja u radu jednaka $\lambda_n$ uzimajući u obzir da je $n \leq N$. Ako sustav ima $S$ paralelnih uslužnih mjesta (kanala) i ako je $\mu$ stopa usluge po poslužitelju, a $n$ broj jedinica u sustavu (sustav predstavlja rep i uslugu) tada je stopa usluge (odlazaka) iz sustava $n\mu$ ako je $n < S$. Ako je $n \geq S$ onda je stopa usluge jednaka $S\mu$.

Iz navedenog primjera vidi se da u općenitom modelu repa stope dolazaka i i odlazaka mogu biti funkcije stanja sustava predstavljenog brojem jedinica, *n*. Tako $\lambda_n$ i $\mu_n$ definiraju dolaske i odlaske kao funkciju od *n*. Cilj općenitog modela je izvesti izraz za $p_n$, odnosno vjerojatnost da će se u stabilnom stanju u sustavu naći *n* jedinica koje su funkcije $\lambda_n$ i $\mu_n$. Ako se pođe od promatranja malog vremenskog perioda $\Delta t$ u koji može doći najviše jedna jedinica, odnosno može biti uslužena najviše jedna jedinica. Vremenski period koji se promatra je toliko malen da je praktički nemoguće da u njega mogu prispjeti dvije ili više jedinica koje trebaju biti uslužene. Vjerojatnost da će u repu biti *n* jedinica izračunava se na temelju odgovarajućih odnosa (65).

Općenito se vrijednost $p_n$ sastoji od:

- vjerojatnosti da u rep dužine ($N$ – 1) jedinica dolazi jedna jedinica, ali ne izlazi nijedna,
- vjerojatnost da u rep dužine *n* jedinica ostaje nepromijenjen,
- vjerojatnost da iz repa dužine ($N$ + 1) odlazi jedna jedinica,

$$P_i = P_{i-1}(1-\mu\Delta t)\lambda\Delta \mathrm{t} + P_i(1-\mu t)(1-\lambda\Delta \mathrm{t}) + P_{i+1}\mu\Delta t(1-\lambda\Delta \mathrm{t}) \tag{65}$$

Gdje su:

- $p_n$ = vjerojatnost da se u repu čekanja nalazi *n* jedinica,
- $\lambda\Delta t$ = vjerojatnost da u rep dolazi jedna jedinica u periodu $\Delta t$,
- $\mu\Delta t$ = vjerojatnost da će se u u periodu $\Delta t$ uslužiti jedna jedinica.

Sređivanjem jednadžbe (65) dobiva se jednadžbu ravnoteže ulaza i izlaza iz sustava (66):

$$\lambda_{n-1}p_{n-1} + \mu_{n+1}p_{n+1} = p_n(\lambda_n + \mu_n) \qquad \text{za } n = 1,2,3,\ldots \tag{66}$$

Jednadžba (66) vrijedi samo za $n > 0$, za izračunavanje jednadžbe ravnoteže kada je $n = 0$ treba uzeti u obzir da stanje 0 korespondira jedino sa stanjem 1 (67):

$$\lambda_0 p_0 = \mu_1 p_1 \qquad \text{za } n = 0 \tag{67}$$

Jednadžbe ravnoteže računaju se rekurzivno polazeći od $p_1$, te se nastavlja induktivno prema $p_n$. Uz $n = 0$ iz dobiva se jednadžba (68):

$$p_1 = \frac{\lambda_0}{\mu_1}p_0 \tag{68}$$

Te uz $n = 1$, iz jednadžbe (69) dobivamo:

$$\lambda_0 p_0 + \mu_2 p_2 = p_1(\lambda_1 + \mu_1) \qquad \text{za } n = 1 \tag{69}$$

Uz supstituciju jednadžbe (68) u jednadžbu (69) i pojednostavljenje dobiva se (70):

$$p_2 = \frac{\lambda_1 \lambda_0}{\mu_1 \mu_2} p_0 \tag{70}$$

Općenito bi se moglo dokazati da vrijedi (71):

$$p_n = \frac{\lambda_{n-1}\lambda_{n-2} \dots \lambda_0}{\mu_n \mu_{n-1} \dots \mu_1} p_0 \qquad \text{za } n = 1, 2, 3, \dots \tag{71}$$

Vrijednost $p_0$ se računa iz jednadžbe (72):

$$\sum_{n=0}^{\infty} p_n = 1 \tag{72}$$

4.11.4.1. *Parametri stabilnog stanja*

Kada je izračunata vjerojatnost $p_n$ da će broj jedinica u sustavu biti $n$, mogu se računati i ostali parametri. Ti parametri poslužit će u analizi modela repova kao preporuka za konstrukciju sustava. Najpoznatiji parametri su očekivane vrijednosti jedinica koje čekaju, očekivano vrijeme čekanja i očekivana iskorištenost sustava.

Za opis parametara koristiti će se sljedeće oznake:

- $L_S$ = očekivani broj jedinica u sustavu,
- $L_q$ = očekivani broj jedinica u repu,
- $W_S$ = očekivano vrijeme čekanja u sustavu,
- $W_q$ = očekivano vrijeme čekanja u repu.

Pretpostaviti će se sustav sa $S$ paralelnih usluga. Iz definicije $p_n$ računa se očekivani broj jedinica u sustavu (73) te očekivani broj jedinica u repu (74):

$$L_s = \sum_{n=0}^{\infty} n p_n \tag{73}$$

$$L_q = \sum_{n=S+1}^{\infty} (n-S) p_n \tag{74}$$

Budući da postoji stroga veza između $L_S$ i $W_S$ također i između $L_q$ i $W_q$ parametri se automatski izračunavaju jedan iz drugoga. Ako je $\lambda_{ef}$ efektivna prosječna stopa dolazaka, neovisna o broju jedinica u sustavu *n*, onda je (75) i (76):

$$L_s = \lambda_{ef} W_S \tag{75}$$

$$L_q = \lambda_{ef} W_q \tag{76}$$

Vrijednost $\lambda_{ef}$ se računa iz $\lambda_n$ zavisnog stanja i vjerojatnosti $p_n$ (77):

$$\lambda_{ef} = \sum_{n=0}^{\infty} \lambda_n p_n \tag{77}$$

Očekivano vrijeme čekanja u sustavu ($W_S$) dobiva se zbrojem očekivanog vremena čekanja u repu ($W_q$) i očekivanog vremena usluge (78):

$$W_S = W_\mathrm{q} + 1/\mu \tag{78}$$

Gdje je:

- $\mu$ = stopa usluge po poslužitelju
- $1/\mu$ = očekivano vrijeme usluge

Ukoliko se jednadžba (78) pomnoži sa $\lambda_{ef}$ dobiva se (79):

$$L_S = L_\mathrm{q} + \lambda_{ef}/\mu \tag{79}$$

Očekivano iskorištenje sustava je definirano kao funkcija prosječnog broja zaposlenih poslužitelja. Budući da razlika $L_S$ i $L_q$ mora biti jednaka očekivanom broju zaposlenih poslužitelja računa se (80):

$$\text{očekivani broj zaposlenih poslužitelja} = \mathrm{S}^* = L_\mathrm{S} - L_\mathrm{q} = \lambda_{ef}/\mu \tag{80}$$

Postotak iskorištenja sustava sa S paralelnih uslužnih mjesta (kanala) računa se prema (81):

$$\text{postotak iskorištenja} = \frac{\mathrm{S}^\mathrm{x}}{\mathrm{S}} \cdot 100 \tag{81}$$

Na kraju se mogu sumirati rezultati parametara za poznati $p_n$ (82):

$$p_n \to L_s = \sum_{n=0}^{\infty} n p_n \to W_S = \frac{L_S}{\lambda_{ef}} \to W_q = W_\mathrm{S} - \frac{1}{\mu} \to L_q = \lambda_{ef} W_q \to \mathrm{S}^* = L_S - L_q \tag{82}$$

#### *4.11.4.1.1. Primjer računa iskoristivosti [4]*

Radi boljeg shvaćanja primijeniti ćemo gore navede formule. Ako se pretpostavi rep s jednim uslužnim mjestom (kanalom) kod kojeg su prosječne stope dolazaka $\lambda_n$=3 i usluge $\mu_n$=8 po satu za sve n $\geq$ 0.

Koraci rješavanja računa iskoristivosti za:

1. Izračun vjerojatnosti da se u repu čekanja nalazi *n* jedinica (83):

$$p_n = \left(\frac{\lambda}{\mu}\right)^n p_0 = \left(\frac{3}{8}\right)^n p_0 = (0.375)^n p_0 \qquad \text{za } n \geq 0 \tag{83}$$

2. Izračun vrijednosti $p_0$ iz jednadžbe (72) dobiva se jednadžba (85):

$$\sum_{n=0}^{\infty} (0.375)^n p_0 = 1 \tag{84}$$

$$p_0(1 + 0.375 + 0.375^2 + 0.375^3 + \ldots + 0.375^n) = 1 \tag{85}$$

3. Primjenom formule za sumu geometrijskog reda na lijevu stranu jednadžbe (85), dobiva se jednadžba (86):

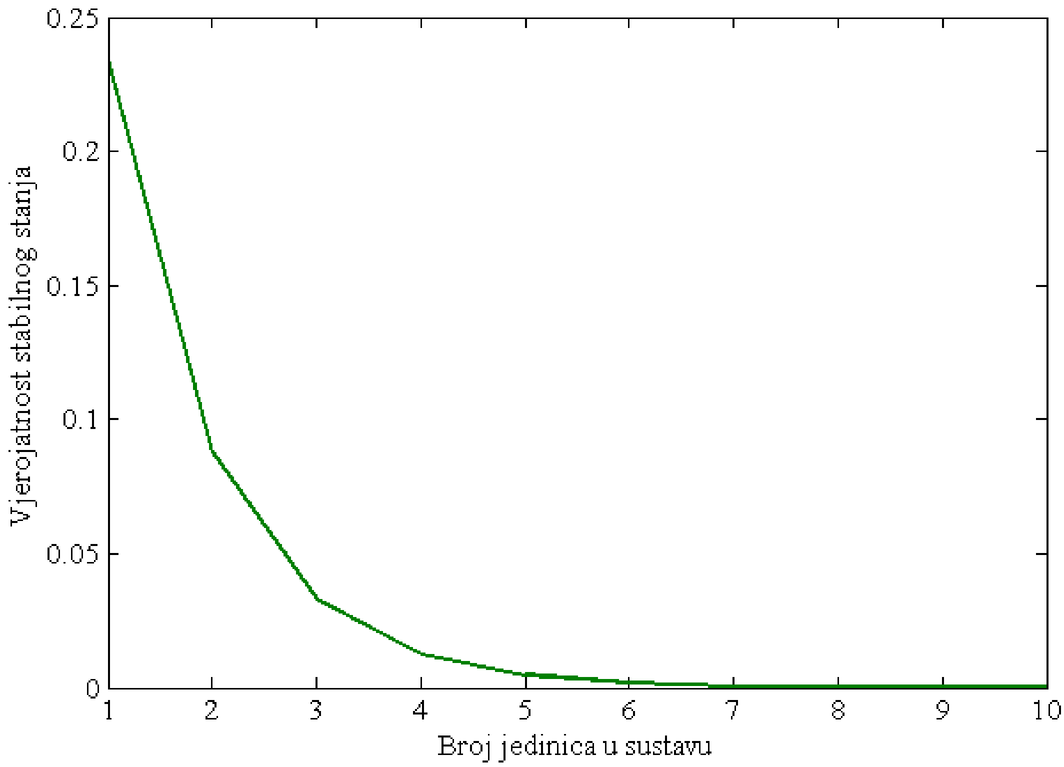
$$\frac{p_0}{1 - 0.375} = 1 \qquad \rightarrow \qquad p_0 = 0.0625 \tag{86}$$

Iz čega slijedi da je $p_0$ = 0.625, iz jednadžbe (83) se može izračunati vjerojatnost $p_n$ da se u repu čekanja nalazi *n* jedinica za pripadajući broj jedinica u sustavu [Tablica 24].

Tablica 24. Vrijednosti $p_n$ za pripadajući broj jedinica *n*

| ***n*** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| $\boldsymbol{p_n}$ | 0.625 | 0.2344 | 0.0879 | 0.0330 | 0.0124 | 0.0046 | 0.0017 | 0.0007 | 0.0002 |

Na dijagramu [Slika 13] se vidi da se povećanjem broja jedinica u sustavu, vjerojatnost da se u repu čekanja nalazi *n* jedinica smanjuje. Kako broj entiteta teži u beskonačnost tako vjerojatnost teži u nulu.




Slika 13. Vjerojatnost stabilnog stanja povećanjem broja entiteta

7. Prosječna stopa dolazaka računa se iz jednadžbe (77) i (72):

$$\lambda_{ef} = 3(p_0 + p_1 + p_2 + \ \dots + p_n) = 3 \tag{87}$$

8. Očekivani broj jedinica u sustavu (88):

$$L_s = \sum_{n=0}^{\infty} np_n = 0 \cdot 0.625 + 1 \cdot 0.234 + \ \dots \ + 8 \cdot 0.0002 = 0.6 \tag{88}$$

9. Iz formule (75) dobiva se vrijeme čekanja u sustavu (89):

$$W_s = \frac{L_s}{\lambda_{ef}} = \frac{0.6}{3} = 0.2 \tag{89}$$

10. Očekivano vrijeme čekanja u repu računa se iz (78), a rezultat je prikazan jednadžbom (90):

$$W_q = W_s - \frac{1}{\mu} = 0.2 - \frac{1}{8} = 0.075 \tag{90}$$

11. Iz (76) dobiva se očekivani broj jedinica u repu:

$$L_q = \lambda_{ef} \cdot W_q = 3 \cdot 0.075 = 0.225 \tag{91}$$

12. Konačno, se računa postotak iskorištenja iz (80) i (81), budući da sustav ima samo jedno mjesto pružanja usluge (S = 1) dobiva se (92):

$$\text{iskorištenje} = \frac{\mathrm{S}^{\mathrm{x}}}{\mathrm{S}} \cdot 100 = \frac{\mathrm{L_s} - \mathrm{L_q}}{1} \cdot 100 = \frac{0.6 - 0.225}{1} \cdot 100 = 37.5\% \tag{92}$$

#### *4.11.5. Specijalni Poissonovi modeli repova [4]*

Navest će se nekoliko specijalnih slučajeva prema Kendallovoj notaciji. Za svaki model repova čekanja dane su odgovarajuće formule koje vrijede za stanje ravnoteže (stacionarno stanje)

##### 4.11.5.1. *Model (M/M/1) : (GD/∞/∞) repa čekanja*

Ovo je najjednostavniji model repova čekanja koji se može primijeniti na velik broj sustava. Model polazi od toga da na neko mjesto usluge s prosječnim brojem usluživanja od $\mu_n = \mu$ mjesta usluživana u jedinici vremena dolazi $\lambda_n = \lambda$ entiteta po jedinici vremena. Vrijeme potrebno za uslugu i intervali pristizanja eksponencijalno su raspodijeljeni. To je isto kao da je broj usluga i pristizanja po vremenskoj jedinici raspodijeljen prema Poissonu.

Pretpostaviti će se da procjena dolazaka neovisna o broju entiteta u sustavu tako da vrijedi $\lambda_n = \lambda$ za svaki *n*. Uzimajući u obzir formule izvedene za opći Poissonov model [4.11.4] mogu se dobiti odgovarajuće formule za bilo koji drugi model. Neka je intenzitet prometa entiteta $\Psi$ omjer dolazaka i usluživanja (93):

$$\Psi = \frac{\lambda}{\mu} \tag{93}$$

Tada se izraz za vjerojatnost $p_n$, da se u repu čekanja nalazi *n* jedinica u modelu svodi na (94):

$$p_n = \Psi^n p_0 \qquad \text{za } n = 0, 1, 2, \dots \tag{94}$$

Vjerojatnost $p_n$, za slučaj $n = 0$ računa se iz činjenice da je suma svih vjerojatnosti $p_n$, jednaka jedan, odnosno vrijedi jednadžba (95):

$$p_0(1 + \Psi^1 + \Psi^2 + \dots + \Psi^n) = 1 \qquad \text{za } n = 0, 1, 2, \dots \tag{95}$$

Ako se pretpostavi da je $\Psi < 1$, vrijedi (96):

$$p_0\left(\frac{1}{1-\Psi}\right) = 1 \qquad ili \qquad p_0 = 1-\Psi \qquad \text{za}\, \Psi < 1 \tag{96}$$

Iz jednadžbe (96) dobiva se formula (97) koja predstavlja geometrijsku distribuciju:

$$p_n = (1-\Psi)\Psi^n \qquad ili \qquad p_0 = 1-\Psi \qquad \text{za}\, n = 0, 1, 2, \dots \tag{97}$$

Nužan uvjet je $\Psi < 1$, kako bi geometrijski red konvergirao. Činjenica da je $\Psi <$ , znači da brzina dolazaka mora biti manja od brzine usluge u danoj situaciji, odnosno $\lambda < \mu$.

Faktor $L_S$, očekivani broj jedinica u sustavu izvodi se na sljedeći način (98):

$$\begin{aligned} L_S = \sum_{n=0}^{\infty} n p_n &= \sum n(1-\Psi)\Psi^n \\ &= (1-\Psi)\Psi \frac{\mathrm{d}}{\mathrm{d}\Psi} \sum_{n=0}^{\infty} \Psi^n \\ &= (1-\Psi)\Psi \frac{\mathrm{d}}{\mathrm{d}\Psi}\left(\frac{1}{1-\Psi}\right) \\ &= \frac{\Psi}{1-\Psi} \end{aligned} \tag{98}$$

Korištenjem veza danih parametrima stabilnog stanja [4.11.4.1] mogu se dobiti svi ostali parametri sutava:

- $L_S$ = očekivani broj jedinica u sustavu (99),

$$L_S = \frac{\Psi}{1-\Psi} \tag{99}$$

- $L_q$ = očekivani broj jedinica u repu (100),

$$L_q = \frac{\Psi^2}{1-\Psi} \tag{100}$$

- $W_S$ = očekivano vrijeme čekanja u sustavu (101),

$$W_S = \frac{L_S}{\lambda} = \frac{1}{\mu(1-\Psi)} \tag{101}$$

– $W_q$ = očekivano vrijeme čekanja u repu (102).

$$W_q = \frac{L_q}{\lambda} = \frac{\Psi}{\mu(1-\Psi)} \tag{102}$$

4.11.5.2. *Model (M/M/1) : (GD/N/∞) repa čekanja*

Jedina razlika između ovog modela i prethodnog , je u maksimalnom broju jedinica u sustavu *N*, što znači da je maksimalna dužina repa *N* – 1. To znači da jednom kada se u sustavu nalazi *N* entiteta nije dozvoljen novi pristup novih entiteta u rep.

Stoga se može pisati (103) i (104):

$$\lambda_n = \begin{cases} \lambda & n = 0, 1, 2, 3, \dots, N-1 \\ 0 & n = N \end{cases} \tag{103}$$

$$\mu_n = \mu, \qquad n = 0,1,2, \dots, N \tag{104}$$

Ako se u obzir uzme jednadžba za intenzitet prometa (93), dobiva se jednadžba (105):

$$p_n = \begin{cases} \Psi^n p_0 & n \le N \\ 0 & n > N \end{cases} \tag{105}$$

Vrijednost $p_0$ izračunava se iz jednadžbe (106):

$$\sum_{n=0}^{N} p_n = 1 \quad \rightarrow \quad p_0(1 + \Psi + \Psi^2 + \ \dots + \Psi^N) = 1 \tag{106}$$

Primjenom formule za sumu geometrijskog reda dobiva se $p_0$ (107):

$$p_0 = \begin{cases} \dfrac{1-\Psi}{1-\Psi^{N+1}} & \Psi \neq 1 \\ \dfrac{1}{1+N} & \Psi = 1 \end{cases} \tag{107}$$

Ubacivanjem (107) u (105) može se pisati da je $p_n$ (108):

$$p_n = \begin{cases} \dfrac{1-\Psi}{1-\Psi^{N+1}} \Psi^n & \Psi \neq 1 \\ \dfrac{1}{1+N} & \Psi = 1 \end{cases} \qquad n = 0,1,2, \dots, N \tag{108}$$

Ψ ne treba biti manji od jedan kao u modelu Model (M/M/1) : (GD/∞/∞). Broj dozvoljen u sustavu kontroliran je dužinom repa (*N* – 1), ne relativnim odnosom dolazaka i usluga. Primjenjujući $p_n$ (109) dobiva se očekivani broj jedinica u sustavu (109):

$$L_s = \begin{cases} \dfrac{\Psi[1-(N+1)]\Psi^N + N\Psi^{N+1}}{(1-\Psi)(1-\Psi^{N+1})}\Psi^n & \Psi \neq 1 \\ \dfrac{N}{2} & \Psi = 1 \end{cases} \tag{109}$$

Parametri $L_q$, $W_s$ i $W_q$ mogu biti izvedeni iz $L_s$ kada je određena mjera dolazaka $\lambda_{ef}$ :

$$\lambda_{ef} = \lambda(p_0 + p_1 + \ ... + p_{N-1}) + \ 0p_N \qquad \rightarrow \qquad \lambda_{ef} = \lambda(1 - p_N) \tag{110}$$

Vjerojatnost da se kupci ne mogu pridružiti sustavu je $p_n$, prema tomeodnos kupaca koji se mogu pridružiti sustavu je $(1 - p_N)$. Primjenom $L_q$, i $\lambda_{ef}$ dolazi se do:

$$L_q = L_s - \frac{\lambda_{ef}}{\mu} = L_s - \frac{\lambda(1-p_N)}{\mu} = L_s - \Psi(1-p_N) \tag{111}$$

$$W_q = \frac{L_q}{\lambda_{ef}} = \frac{L_q}{\lambda(1-p_N)} \tag{112}$$

$$W_s = W_q + \frac{1}{\mu} = \frac{L_s}{\lambda(1-p_N)} \tag{113}$$

4.11.5.3. *Model (M/M/S) : (GD/∞/∞) repa čekanja*

U ovom modelu dolasci slijede Poissonovu distribuciju s prosječnim brojem dolazaka u jedinici vremena $\lambda$ i maksimumom od $S$ jedinica koje mogu biti istovremeno uslužene. Sva mjesta usluživanja imaju istu prosječnu konstantnu vrijednost $\mu$. Ako je broj dolazaka u sustav $n$ jednak ili prelazi $S$, tada je $\mu_n = S\mu$. Ako je $n$ manji od $S$ mjera usluge je $nS$.

$$\lambda_n = \lambda \qquad\qquad n \geq 0 \tag{114}$$

$$\mu_n = \begin{cases} n\mu & n \leq S \\ S\mu & n \geq S \end{cases} \tag{115}$$

U nastavku se računa $p_n$ i $p_0$ :

$$p_n = \begin{cases} \dfrac{\Psi^n}{n!}p_0 & 0 \leq n \leq S \\ \dfrac{\Psi^n}{S^{n-S}S!}p_0 & n > S \end{cases} \tag{116}$$

$$p_0 = \left( \sum_{n=0}^{S-1} \frac{\Psi^n}{n!} + \frac{\Psi^S}{S!\left(1 - \frac{\Psi}{S}\right)} \right)^{-1} \text{ gdje je } \frac{\Psi}{S} = \frac{\lambda}{\mu S} < 1 \tag{117}$$

Očekivane vrijednosti računaju se iz:

$$L_q = \frac{\Psi^{S+1} p_0}{(S-1)!\,(S-\Psi)^2} = \frac{S\Psi}{(S-\Psi)^2} p_0 \tag{118}$$

$$L_s = L_q + \Psi \tag{119}$$

$$W_q = \frac{L_q}{\lambda} \tag{120}$$

$$W_s = W_q + \frac{1}{\mu} \tag{121}$$

Pronalaženje optimalnog broja uslužnih mjesta:

1. Pronalazi se najmanji cijeli broj $K > \lambda / S$. za izabrani broj poslužitelja $K$ pronalazi se vrijednost $L_s$ i izračunava vrijednost troškova $T_K$ za taj $K$.
2. $K$ se povećava za jedan, pronalazi se vrijednost $L_s$ za novi broj poslužitelja izračunavaju se ukupni troškovi $T_{K+1}$.
3. Uspoređuju se vrijednosti $T_K$ i $T_{K+1}$ ako je $T_K > T_{K+1}$ treba se vratiti na korak 2. Postupak se nastavlja sve dok se $T$ smanjuje. Ako se $T$ povećava ili ostaje isti procedura se prekida, optimalno rješenje je pronađeno.

### 4.11.5.4. *Model (M/M/S) : (GD/N/∞) repa čekanja*

Ovaj model se od prethodnog razlikuje po tome što je dana granica kapaciteta sustava $N$, pa je maksimalna dužina repa $N - S$. pomoću općih izraza dolazi se od parametara:

$$\lambda_n = \begin{cases} \lambda & 0 \le n \le N \\ 0 & n > N \end{cases} \tag{122}$$

$$\mu_n = \begin{cases} n\mu & 0 \le n \le S \\ S\mu & S \le n \le N \end{cases} \tag{123}$$

Uz supstitiuciju $\lambda_n$ i $\mu_n$ u općem izrazu za $p_n$ i uključujući $\Psi = \lambda / \mu$ dobiva se:

$$p_n = \begin{cases} \frac{\Psi^n}{n!} p_0 & 0 \leq n \leq S \\ \frac{\Psi^n}{S^{n-S} S!} p_0 & S \leq n \leq N \end{cases} \tag{124}$$

Gdje je:

$$p_0 = \begin{cases} \left( \sum_{n=0}^{S-1} \frac{\Psi^n}{n!} + \frac{\Psi^S \left(1 - \frac{\Psi}{S}\right)^{N-S}}{S!} \right)^{-1} & \frac{\Psi}{S} \neq 1 \\ \left( \sum_{n=0}^{S-1} \frac{\Psi^n}{n!} + \frac{\Psi^S (N - S + 1)}{S!} \right)^{-1} & \frac{\Psi}{S} = 1 \end{cases}$$

Očekivane vrijednosti računaju se iz:

$$L_q = \begin{cases} p_0 \frac{\Psi^{S+1}}{(S-1)!\,(S-\Psi)^2} \left[ 1 - \left(\frac{\Psi}{S}\right)^{N-S} - (N-S)\frac{\Psi}{S}\left(1 - \frac{\Psi}{S}\right) \right], & \frac{\Psi}{S} \neq 1 \\ p_0 \frac{\Psi^S (N-S)(N-S+1)}{2S!} & \frac{\Psi}{S} = 1 \end{cases} \tag{125}$$

$$L_s = L_q + (S - \bar{S}) = L_q + \frac{\lambda_{ef}}{\mu} \tag{126}$$

Gdje je $\bar{S}$ prosječan broj iskorištenih poslužitelja:

$$\bar{\mathrm{S}} = \sum_{n=0}^{\mathrm{S}} (\mathrm{S} - n)\, p_n \tag{127}$$

4.11.5.5. *Model (M/M/∞) : (GD/∞/∞) repa čekanja*

Ovaj model se zove model samoposluživanja budući da je broj poslužitelja neograničen jer su jedinice ujedno i poslužitelji.

Iz općenitog modela imamo:

$$\lambda_n = \lambda \qquad n \geq 0 \tag{128}$$

$$\mu_n = n\mu \qquad n \geq 0 \tag{129}$$

$$p_n = \frac{\lambda^n}{n!\,\mu^n} p_0 = \frac{\Psi^n}{n!} p_0 \tag{130}$$

$$p_0 = e^{-\Psi} \tag{131}$$

Uzimajući u obzir gore navedene jednadžbe i opći slučaj modela repova čekanja može se pisati:

$$p_n = \frac{\Psi^n}{n!\,e^{\Psi}} \qquad za\ n = 0,1,2,\ldots \tag{132}$$

$$L_s = \Psi \tag{133}$$

$$W_s = \frac{1}{\mu} \tag{134}$$

$$L_q = W_q = 0 \tag{135}$$

$W_q = 0$ zbog toga jer se radi o modelu samoposluživanja. Rezultati ovog modela mogu se koristiti za aproksimaciju Model (M/M/S) : (GD/∞/∞) modela ako se *S* dovoljno poveća, puno je praktičnije koristiti aproksimaciju budući da je računanje puno jednostavnije.

### 4.11.5.6. *Model (M/M/R) : (GD/K/K) repa čekanja*

Ovaj model naziva se strojni model usluge. Pretpostavlja *R* servisera koji su na raspolaganju da izvrše servisiranje svih *K* strojeva. Ovaj model primjer je ograničenog izvora pozivanja budući da pokvareni stroj ne može izazvati nove pozive dok je na servisu. Neka je $\lambda$ stopa kvara po stroju, a $\mu$ stopa usluge, onda vrijedi:

$$\lambda_n = \begin{cases} (K-n)\lambda & 0 \le n \le K \\ 0 & n \ge K \end{cases} \tag{136}$$

$$\lambda_n = \begin{cases} n\mu & 0 \le n \le R \\ R\mu & \mathrm{R} \le n \le K \\ 0 & n > K \end{cases} \tag{137}$$

Ubacivanjem (136) i (137) u dobiva se :

$$p_n = \begin{cases} \binom{K}{n} \Psi^n p_0 & 0 \le n \le R \\ \binom{K}{n} \dfrac{n!\,\Psi^n}{R^{n-R} R!} p_0 & R \le n \le K \end{cases} \tag{138}$$

$$p_0 = \left( \sum_{n=0}^{R} \binom{K}{n} \Psi^n + \sum_{n=R+1}^{K} \frac{n!\,\Psi^n}{R^{n-R} R!} \right)^{-1} \tag{139}$$

Očekivane vrijednosti uz $R \neq 1$ prikazane su jednadžbama :

$$L_q = \sum_{n=R+1}^{R} (n-R) p_n \tag{140}$$

$$L_s = L_q (R - \bar{R}) = L_q + \frac{\lambda_{ef}}{\mu} \tag{141}$$

Gdje $\bar{R}$ predstavlja prosječan broj servisera koji nemaju što raditi:

$$\bar{R} = \sum_{n=r+1}^{R} (R-n) p_n \tag{142}$$

$$\lambda_{ef} = \mu (R - \bar{R}) = \lambda (K - L_s)$$

U slučaju $R$=1, samo jedan serviser:

$$L_q = K - \left(1 + \frac{1}{\Psi}\right)(1 - p_0) \tag{143}$$

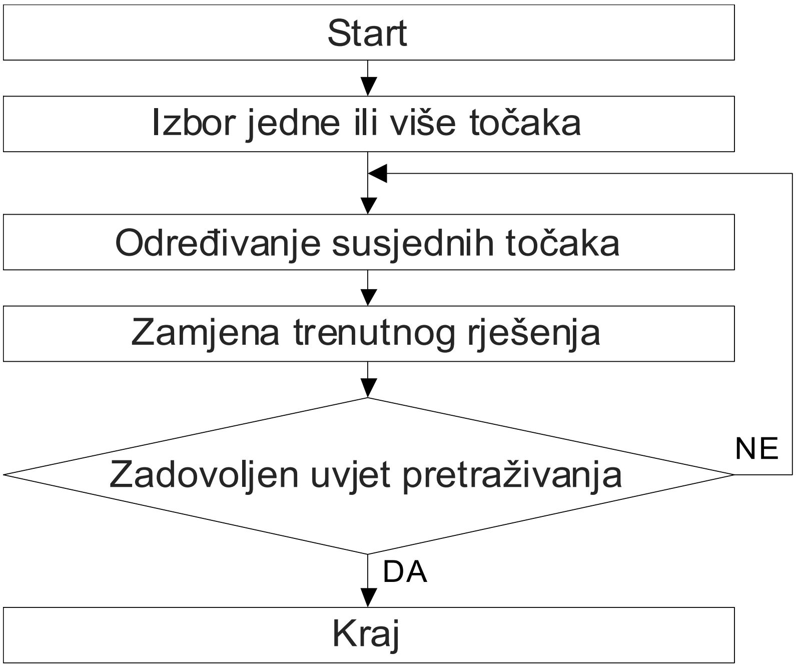
$$L_s = K - \frac{1 - p_0}{\Psi} \tag{144}$$

### 4.12. Optimiranje rojem čestica [1], [2]

Postoji još jedan način podjele metoda za optimiranje, na egzaktne, gdje spadaju linearno programiranje, nelinearno i dinamičko programiranje, te na heurističke gdje spada optimiranje rojem čestica. Heuristika jest pravilo temeljeno na prethodnom iskustvu pomoću kojeg tražimo rješenje. Heuristički algoritmi se temelje na heuristici i koriste se za rješavanje složenih problema. Shema opće procedure heurističkog modela prikazana je na sljedećoj slici.

Dakle, najvažniji elementi heurističkog algoritma jesu:

1. memorija potencijalnih rješenja (čestice koje generirano pamte pozicije i uspoređuju pozicije),
2. mehanizam transformacije postojećih rješenja,
3. mehanizam razmatranja susjednih točaka,
4. mehanizam smjene potencijalnih rješenja,
5. uvjeti zaustavljanja.




Slika 14. Opća procedura heurističkog modela

Uvjeti pri kojima dolazi do zaustavljanja:

1. maksimalan broj iteracija – prije samog pretraživanja odredi se broj iteracija,
2. mjere poboljšanja vrijednosti kriterija između dvije ili više opcija,
3. mjere napredovanja u prostoru pretraživanja (pomak počne stagnirati),
4. ograničenje vezano uz resurse.

Postupak pri rješavanju problema ovim algoritmom vrši se sljedećim koracima:

1. najprije se odredi veličina roja (roj je populacija čestica, a čestica je jedinka populacije), odnosno broj čestica $N$,
2. generira se početna populacija slučajnim odabirom $x_i(t)$, pri čemu $t$ označava iteraciju, a $i$ brojeve 1, 2, …, $N$,
   a) izračunati iteraciju 1: $x_1(1)$, $x_2(1)$, …, $x_N(1)$,
   b) evaluirati funkciju cilja s obzirom na čestice: $F(x_1(1))$, $F(x_2(1))$, …, $F(x_N(1))$,
3. računa se brzina čestice,
4. u iteraciji $t + 1$ računaju se parametri,
   a) najbolju poziciju do sada čestice $x_i(t)$ označiti s $p_i(t)$, što je vrijednost s najvećom vrijednošću funkcije cilja - najbolje lokalno rješenje ($P$ označava memoriju),
   b) najbolju poziciju među svim česticama (najbolja pozicija cijelog roja) označiti s $p_g(t)$, što je najbolje globalno rješenje,

c) izračunati $v_i(t + 1)$,

d) odrediti novu poziciju čestice $x_i(t +1)$,

5. na kraju se provjerava konvergencija, odnosno uvjet zaustavljanja - ako ne postoji, vraćamo se na četvrti korak, u suprotnom je kraj.

Jednadžba glasi:

$$x_i(t+1) = x_i(t) + v_i(t+1)\,, \tag{145}$$

$$v_i(t+1) = v_i(t) + c_1 R_1\big(p_i(t) - x_i(t)\big) + c_2 R_2\left(p_g(t) - x_i(t)\right), \tag{146}$$

pri čemu je:

$c_1$ ... kognitivni ili spoznajni parametar,

$c_2$ ... socijalni ili društveni parametar,

za:

$$1{,}8 < c_1 = c_2 \leq 2{,}2\ , \tag{147}$$

$$R_1, R_2\ \langle 0; 1]\ , \tag{148}$$

$$p_i(t+1) = \begin{cases} x_i(t), & ako\ je\ F\big(x_i(t+1)\big) \leq p_i(t) \\ p_i(t), & za\ sve\ ostalo \end{cases}\ . \tag{149}$$

## 4.13. Simulacije [3]

Simulacije se koriste za analizu i proučavanje kompleksnih sustava. U prethodnom poglavlju smo se fokusirali na formuliranje modela koji se mogu riješiti analitički. U gotovo svim modelima cilj je bio odrediti optimalno rješenje. Međutim, zbog složenosti, stohastičkih odnosa i drugih faktora realni problemi se ne mogu adekvatno reprezentirati u obliku analitičkih modela obrađenih u prethodnom poglavlju. Pokušaji korištenja analitičkih modela često zahtijevaju jako puno pojednostavljenih pretpostavki. Pa je takav rezultat puno lošiji i često je neadekvatan za implementaciju. Zbog toga je donositelju odluke jedini alternativni oblik modeliranja i analize simulacija. Najčešće se primjenjuje u situacijama dolaska putnika u redovima za posluživanje, usluživanja kupaca, posluživanja radnika alatom, testiranja pilota.

Postoje mnogi razlozi za korištenje simulacija, ali najčešći su:

a) kompleksni problemi, nelinearne varijable, velik broj varijabli i ograničenja,

b) simulacijski se model koristi za eksperimentiranje bez da se ometa realni sustav,

c) simulacijom se postiže kompresija ili skraćenje vremena (za prikupljanje podataka se utroši manje vremena, nego da se to radi iz realnog sustava),

d) korištenjem simulacijskog modela menadžeri se vježbaju pri donošenju odluka, dakle manja šteta će biti ukoliko se pogriješi na simulacijskom modelu nego u realnom sustavu.

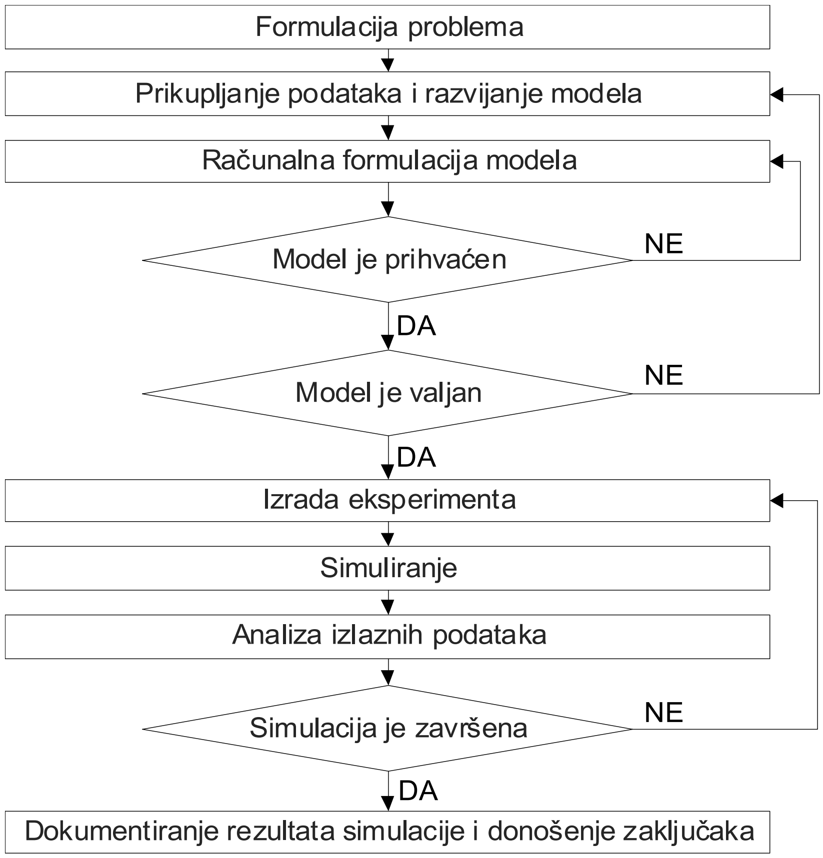
e) Vrlo rijetko se simulacijski model koristi za određivanje optimalne vrijednosti funkcije cilja,

Simulacija se može definirati kao tehnika koja imitira rad stvarnog sustava koji se mijenja s vremenom. Simulacijski model predstavlja skup pretpostavki o radu sustava, koje su izražene kao matematičke i logičke veze među objektima od interesa. Za razliku od egzaktnih matematičkih rješenja simulacijski proces uključuje izvođenje i rad modela kroz vrijeme kako bi se prikupili reprezentativni uzorci specifičnih mjera. U tom smislu se na proces simulacije može gledati kao na eksperiment uzorkovanja realnog sustava gdje rezultati predstavljaju uzorke. Faktori kao što su početni uvjeti, trajanje i točnost modela simulacije utječu na konačnu kvalitetu procjene. Prednost simulacije u odnosnu na analitičke metode jest jednostavnost primjene i fleksibilnost u pogledu reprezentiranja sustava realnog svijeta.

Reprezentativni model realnog sustava služit će za analiziranje ponašanja sustava u različitim uvjetima. Pa će tako mijenjanjem parametara i pravila unutar sustava biti vidljivi različiti ishodi uz iste početne uvjete. Treba naglasiti da simulacija ne predstavlja tehniku optimiranja (iako je optimiranje simulacijom moguće). Koristi se najčešće za analizu sustava mijenjanjem njegovih parametara. Odnosno za pitanja tipa „što će se desiti ako…?“ (npr. „Što će se desiti ako se poveća broj radnika?“, „Što će se desiti ako se skrati radno vrijeme?“ i slično). Optimiranje sustava može se izvesti kombinacijom simulacije i drugih modela optimiranja.

Proces svake simulacije se sastoji od sljedećih koraka [Slika 15]:

1. prikupljanje podataka
2. dodjeljivanje slučajnih brojeva,
3. formulacija simulacijskog model,
4. analiza rezultata.




Slika 15. Koraci prilikom izrade simulacije[3]

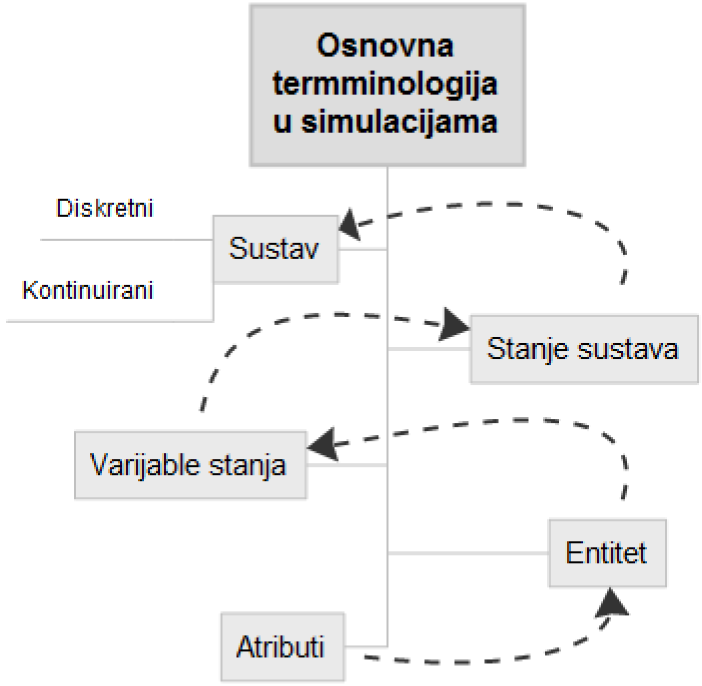
Prikupljanje podataka se vrši na dva načina:

1. statičkim uzorkovanjem – iz osnovnog skupa se odabire uzorak i na njemu se radi (ovaj način se primjenjuje kada je prikupljanje podataka skupo ili kada podaci nisu raspoloživi iz javnih izvora);
2. povijesnim podacima – uzimaju se iz podataka poduzeća, novina, znanstvenih baza, vladinih izvora, statističkih baza.

Osnovna terminologija u simulacijama [Slika 16]:

- sustav[4] – skup entiteta koji agira i interagira radi postizanja logičkog svršetka;
- stanje sustava – skup varijabli potreban za opisivanje statusa sustava u bilo kojem trenutku
- varijable stanja – varijable koje opisuju moguće promjene stanja unutar sustava
- entitet – predmet od interesa
- atributi – karakteristike koje opisuju entitet

[4] Definicija prema Schmidtu i Tayloru iz 1970.g.




Slika 16. Osnovna terminologija u simulacijama i veze

Točan opis sustava ovisi o ciljevima koji se žele postići simulacijom. Iz tog razloga ono što predstavlja sustav u jednoj simulaciji u drugoj može biti samo podskup cjelokupnog sustava. Budući da su sustavi u suštini dinamični (mijenjaju se vremenom) potrebno je definirati stanje sustava.

Za objašnjavanje terminologije [Slika 16] razmotrit će se pojednostavljeni sustav banke. Sustav banke se sastoji se od bankara i korisnika koji čekaju u redu ili im se pruža usluga. Odlaskom i dolaskom korisnika sustav se mijenja. Kako bi se opisale promijene u sustavu potreban nam je skup varijabli. Na primjer broj zauzetih bankara, broj korisnika u banci, vrijeme pristizanja sljedećeg korisnika i vrijeme odlaska korisnika kojima se usluga pružila. To su varijable stanja, one zajedno opisuju svaku moguću promjenu stanja unutar banke. Korisnici su entiteti, a karakteristike korisnika npr. zanimanje korisnika je atribut tog entiteta.

Razlikuju se tri vrste varijabli:

- varijabla odlučivanja – kontrolira ih donositelj odluka i mijenjaju se od događaja do događaja u procesu simulacije,
- nekontrolirane varijable – slučajni događaji koji se ne mogu kontrolirati,
- zavisne varijable – varijable na koje utječu prve dvije varijable.

Sustavi se mogu klasificirati kao diskretni i kontinuirani:

1. diskretan je sustav onaj kod kojeg se varijable stanja mijenjaju u diskretnim tj prebrojivim vremenskim trenutcima (npr. banka).
2. Kontinuirani sustav je onaj kod kojeg se varijable mijenjaju kontinuirano tijekom vremena (npr. kemijski proces)

Tipovi simulacijskih modela:

1. statički – reprezentacija sustava u točno određenom vremenskom trenutku,
2. dinamički - reprezentacija sustava koji se tijekom vremena razvija (mijenja).

Unutar ta dva tipa simulacijski model može biti:

1. deterministički –ne sadrži slučajne varijable,
2. stohastička – sadrži jednu ili više slučajnih varijabli,

Tipovi simulacija s obzirom na trajanje:

1. Terminirana simulacija – zaustavlja se u definiranom periodu u vremenu,
2. Simulacija stabilnog stanja – zaustavlja se u nedefiniranom periodu u vremenu (traje „beskonačno" dugo)

### *4.13.1. Simulacija diskretnog događaja*

Prvo će se razmotriti diskretni stohastički modeli. Takvi modeli se zovu simulacijski modeli diskretnih događaja. Simulacija diskretnog događaja uključuje modeliranje stohastičkog sustava koji se razvija vremenom reprezentacijom sustava u kojem se varijable stanja mijenjaju samo u diskretnim vremenskim trenutcima.

U prethodnom poglavlju pretpostavilo se da su dolasci i odlasci eksponencijalno distribuirani. Kod stvarnih problema potrebno je koristiti empiričku distribuciju dolazaka. U simulacijama se može koristiti bilo koja distribucija vremena dolazaka i vremena trajanja usluge, ali takva da što bolje opisuje dani sustav.

#### 4.13.1.1. *Razumijevanje osnovnih koncepata simulacije*

Kako bi se objasnili osnovni koncepti simulacije proći će se kroz pojednostavljeni primjer. Pri tome će se obratiti pozornost na terminologiju u simulacijama [Slika 16].

Prvo je potrebno definirati sustav. Neka je dan sustav repova čekanja s jednim uslužnim mjestom ($S$ = 1). Korisnici u sustav pristižu iz beskonačnog izvora entiteta, nakon toga im se pruži usluga ako je pružatelj usluge slobodan ili čekaju u redu ukoliko je pružatelj usluge zauzet. Kapacitet prostora čekanja je beskonačan i korisnici se uslužuju prema redoslijedu dolaska. Pretpostavlja se da u jednoj vremenskoj jedinici stiže jedan korisnik, s distribucijom dolazaka prema tablici [Tablica 25]. Korisnicima se usluga pruža s distribucijom vremena pružanja usluge prikazanoj u tablici [Tablica 25]. Vremena pružanja usluge su slučajna. Nakon pružanja usluge svi se korisnici vraćaju u izvor pristizanja iz kojeg su došli.

Tablica 25. Distribucija vremena dolazaka i pružanja usluge

| Distribucija vremena dolazaka | | Distribucija vremena pružanja usluge | |
|---|---|---|---|
| Vrijeme između dolazaka | Vjerojatnost | Trajanje pružanja usluge | Vjerojatnost |
| 1 | 0.20 | 1 | 0.35 |
| 2 | 0.30 | 2 | 0.40 |
| 3 | 0.35 | 3 | 0.25 |
| 4 | 0.15 | | |

Prije nego li se krene u detalje simuliranja potrebno je definirati stanje sustava, razumjeti koncepte događaja i vrijeme unutar simulacije.

Varijable stanja:

- broj korisnika u sustavu,
- status pružatelja usluge (zauzet ili slobodan),
- vrijeme sljedećeg dolaska.

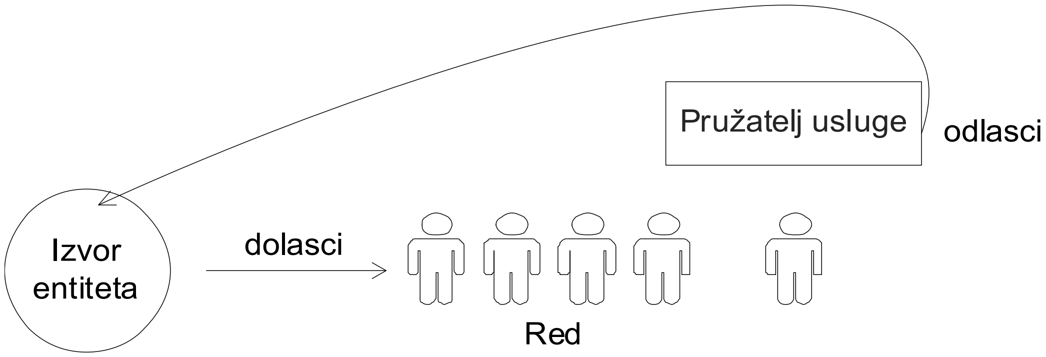
Koncept događaja usko je vezan sa stanjem sustava. Događaj je definiran kao situacija koja izaziva trenutačnu promjenu stanja sustava. U sustavu s jednim pružateljem usluge postoje dva događaja koja mogu promijeniti stanje sustava:

1. dolazak u sustav i
2. odlazak iz sustava nakon završetka usluge.

U simulaciji će se ti događaji planirati u određenom vremenskom trenutku.

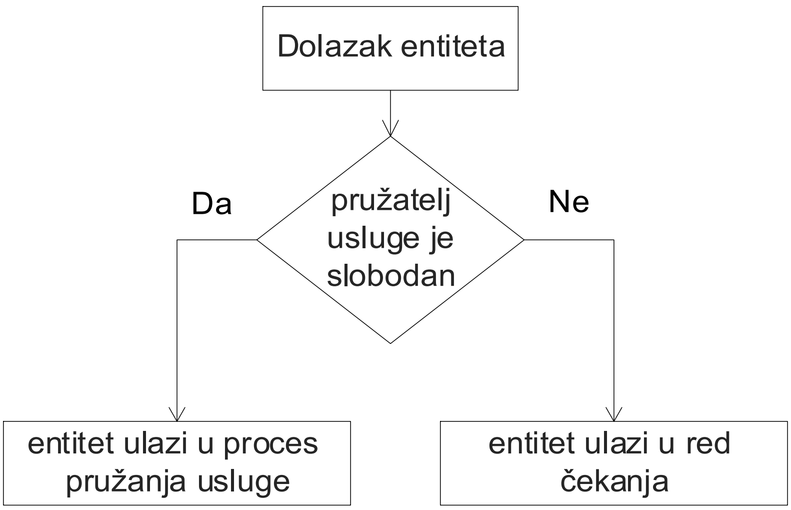
Informacije o događajima se spremaju u popis događaja. Unutar tog popisa vodi se evidencija o tipu događaja koji se planira i vremenu u kojem je će se događaj dogoditi. Vrijeme se u simulaciji sačuvano u varijabli vrijeme (na satu).

Simulacija započinje s praznim sustavom i pretpostavkom da se prvi događaj događa u vremenu nula [Slika 17]. Prvom entitetu se usluga odmah pruža budući da je pružatelj usluge slobodan.




Slika 17. Shema sustava [3]

Entiteti koji stižu nakon prvog entiteta, pružatelja usluge mogu zateći kao slobodnog ili zauzetog. Ako je pružatelj usluge slobodan entitet ulazi u proces pružanja usluge. Ako je pružatelj usluge zauzet entitet će stati u red [Slika 18].

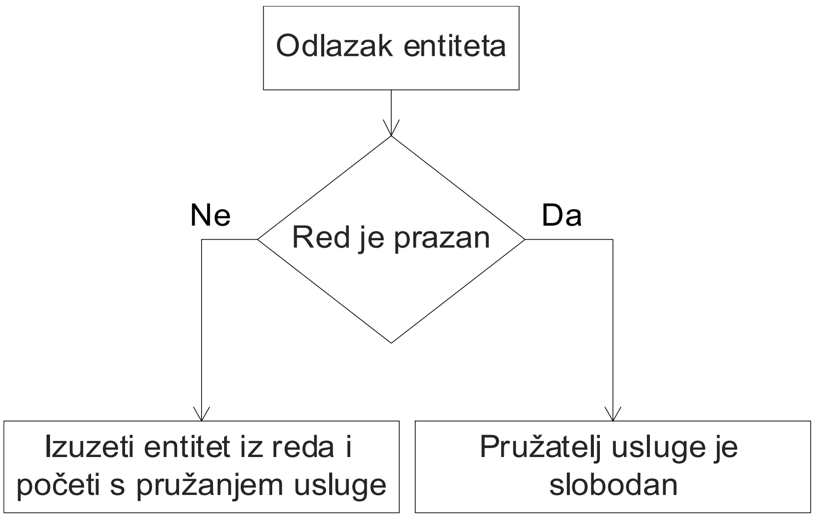



Slika 18. Shema dolazaka [3]

Odlazak prvog korisnika planira se generiranjem slučajnih vremena pružanja usluge prema distribuciji vremena pružanja usluge [Tablica 25]. Vremena odlazaka definiraju se zbrojem trenutnog i generiranog vremena. Sljedeći dolazak u sustav se planira generiranjem slučajnih dolazaka iz distribucije vremena dolazaka [Tablica 25]. Vremena dolazaka definirat će se kao zbroj trenutnog i generiranog vremena.

Na primjer generiralo se vrijeme pružanja usluge od 2 minute stoga je trenutno vrijeme odlaska prvog korisnika biti 2 minute. Slično ako smo generirali dolaske od 1 minute sljedeći dolazak će biti planiran za trenutno vrijeme 1 minute.

Oba događaja i njihova vremena planiranja se vode u popisu događaja. Jednom kada se završe sve potrebne akcije prvog dolaska, lista događaja se skenira za određivanje sljedećeg planiranog događaja i pripadajućeg vremena. Ako je sljedeći događaj dolazak, trenutno vrijeme se postavlja na planirano vrijeme dolaska i prolazi se kroz proces dolaska. Ako je sljedeći događaj odlazak, vrijeme se postavlja na planirano vrijeme odlaska i prolazi se kroz proces odlaska. Pri odlasku entiteta provjerava je li dužina reda čekanja veća od nule. Ukoliko je, iz reda čekanja izuzima se jedan korisnik koji počinje s procesom usluživanja postavljanjem vremena odlaska kao zbroj trenutnog i generiranog vremena. Ako nema nikoga u redu čekanja pružatelj usluge je slobodan [Slika 19].




Slika 19. Shema odlazaka [3]

Ovaj pristupu simulacije naziva se mehanizam sljedećeg događaja pomakom vremena, zbog načina na koji se trenutno vrijeme računa. Vrijeme simulacije se pomiče na vrijeme sljedećeg događaja, a to je prvi događaj u popisu događaja. Budući da se varijable stanja mijenjaju samo u vremenu događaja, preskaču se periodi vremena neaktivnosti između vremena. To se izvodi skakanjem s događaja na događaj, veličina skokova može varirati. Taj proces zahtjeva da u bilo kojem trenu simulacije imamo planirani budući dolazak i odlazak. Budući dolazak se planira kada je novi dolazak u sustavu. A vrijeme odlaska se može jedino planirati kada korisnik dođe do procesa usluživanja. Što znači da se odlasci ne mogu planirati ukoliko je pružatelj usluge slobodan. U tom slučaju kreira se demo odlazak na način da se vrijeme odlaska postavi na neku veliku vrijednost u odnosu na ono što bi se u sustavu očekivalo.

Kod metode fiksiranog prirasta pomakom vremena, vrijeme simulacije se pomiče u koracima $\Delta t$ vremenskih jedinica, gdje je $\Delta t$ jedna vremenska jedinica. Nakon svakog ažuriranja vremena provjerava se postoji li planirani događaj u toj trenutnoj vremenskoj jedinici. Ukoliko je neki događaj planiran provode se prikladne radnje vezane uz taj događaj. Ukoliko nema planiranih događaja, ili su završene sve potrebne radnje trenutnog vremena vrijeme simulacije se ažurira s $\Delta t$ jedinica i proces se ponavlja. Ovaj se postupak ponavlja sve dok se ne postigne unaprijed definirano stanje zaustavljanja.

Dijagram toka procesa simulacije sustava reda čekanja s jednim uslužnim mjestom prikazan je na slici [Slika 20]. Radi jednostavnosti pretpostavlja se da su vremena pristizanja i pružanja usluge generirana za prvih nekoliko dolazaka prema zadanim distribucijama vjerojatnosti [Tablica 25].

Varijable u sustavu:

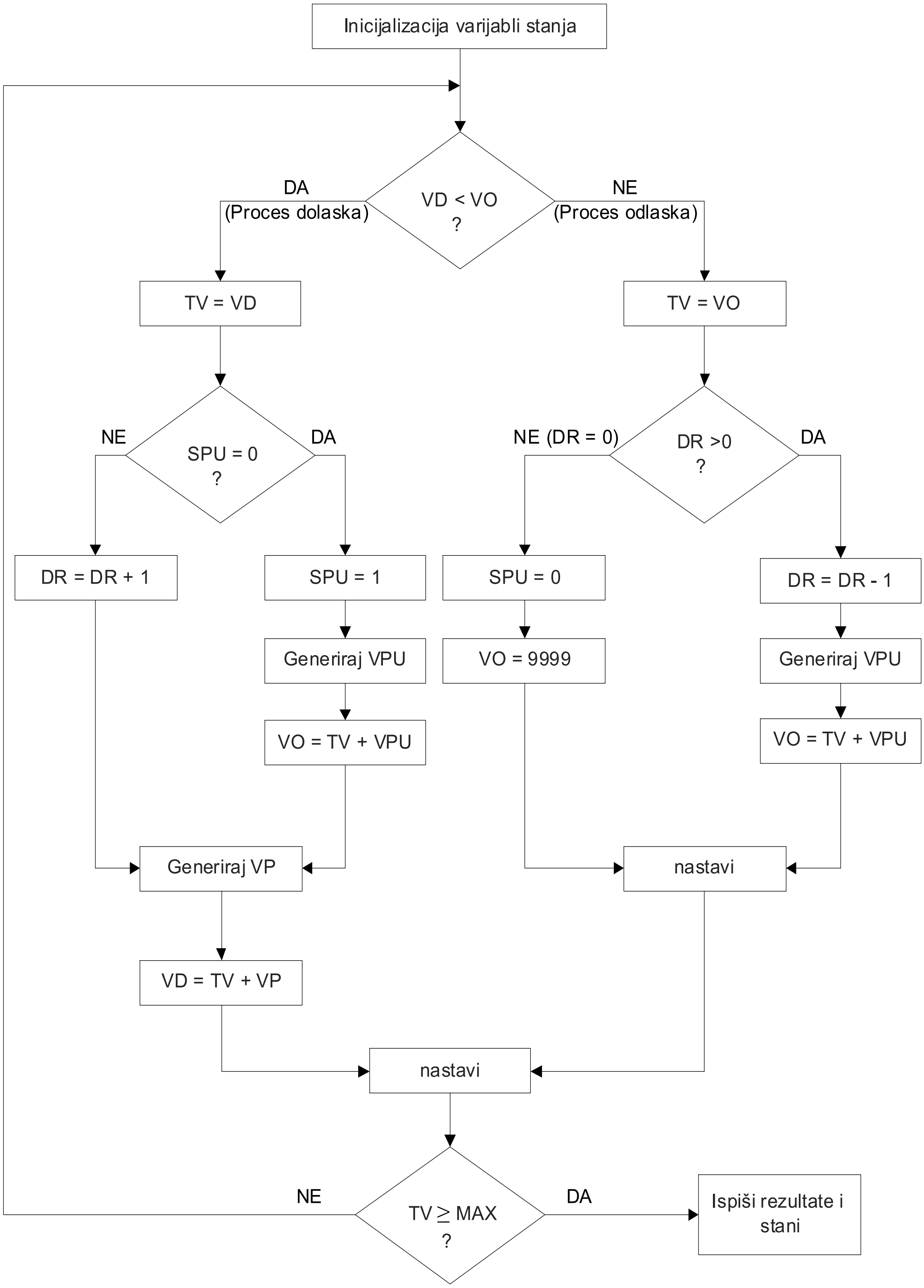
- vremena pristizanja, VP;
- vremena pružanja usluge, VPU;
- trenutno vrijeme, TV;
- vrijeme planiranog sljedećeg dolaska, VD;
- vrijeme planiranog sljedećeg odlaska, VO;
- status pružatelja usluge (1 = zauzet, 0 = slobodan), SPU;
- dužina reda čekanja, DR;
- trajanje simulacije (u vremenskim jedinicama), MAX.

Simulacija započinje inicijalizacijom svih varijabli uz pretpostavke:

- VD=0, prvi dolazak održati će se u vremenu 0;
- SPU=0, jer je u vremenu 0 sustav prazan;
- DR=0, jer je u vremenu 0 sustav prazan;
- VO=9999, postavljanje demo odlaska s obzirom na SPU=0, pri čemu VO > MAX.

Sada će se dati kratak opis logike dijagrama. Budući da simulacija sadrži samo dva događaja treba se usporedbom tih događaja odrediti događaj koji slijedi. Nakon inicijalizacije VD = 0 i manji je od VO = 9999 što znači da će uslijediti događaj dolaska. Taj događaj je sada događaj 1, trenutno vrijeme se ažurira na vrijeme događaja 1 odnosno TV = 0. Dolazak u vremenu 0 nailazi na prazan sustav odnosno SPU = 0, korisnik neposredno kreće u proces usluživanja. Tada se postavlja SPU = 1 kako bi se naznačila zauzetost pružatelja usluge. Nakon toga slijedi generiranje vremena usluživanja i postavljanje vremena odlaska tog korisnika. Budući da je

TV = 0, slijedi VO = VPU, za dovršetak procesa dolaska generira se vrijeme pristizanja VP nakon čega se postavlja vrijeme VD = VP zbog TV=0 s ovime je završen događaj 1. Neka bude TV < MAX, slijedi događaj 2 i tako dalje sve dok TV ≥ MAX.




Slika 20. Dijagram toka za simulacijski model reda čekanja s jednim uslužnim mjestom [3]

### 4.13.2. Slučajni brojevi i Monte Carlo simulacija

U primjeru simulacije reda čekanja kretanje kroz vrijeme postignuto je generiranjem vremena pristizanja i usluživanja zadanim distribucijama vjerojatnosti. Može se reći da su sva vremena događaja direktno ili indirektno određena generiranim vremenima pristizanja i usluživanja. Proces generiranja tih vremena iz date distribucije vjerojatnosti poznat je pod nazivom uzorkovanje iz vjerojatnosti, generiranje slučajnih varijabli ili Monte Carlo uzorkovanje. Slučajni broj definiran je kao nezavisni slučajni uzorak izvučen iz kontinuirane uniformne distribucije. Za neprekidnu slučajnu varijablu $X$ kažemo da ima uniformnu distribuciju na intervalu $(a, b)$, $a < b$ ako joj je funkcija gustoće dana izrazom (150):

$$f(x) = \begin{cases} \frac{1}{b-a} & x \in (a, b) \\ 0 & x \in (a, b) \end{cases} \tag{150}$$

Monte Carlo simulacija ili simulacija vjerojatnosti je metoda koja se koristi za razumijevanje utjecaja rizika i neizvjesnosti u različitim modelima predviđanja. Monte Carlo simulacije se mogu činiti kompliciranima, ali najjednostavnije objašnjenje proizlazi iz pitanja: „Zašto se radi Monte Carlo simulacija?“. Postoje brojni razlozi korištenja Monte Carlo simulacije, ali svi se svode na jednu stvar, a to je predviđanje učinkovitosti bez provođenja stotine eksperimenata i sakupljanja uzoraka.

Pri izradi modela predviđanja, odnosno bilo kakvog modela kojim se razmatraju događanja u budućnosti postavljaju se određene pretpostavke. Na primjer predviđanje završetka određenog događaja ili predviđanja troška nekog projekta. Budući da su to projekcije u budućnosti najbolje što se može napraviti je procijeniti očekivanu vrijednost. Ne može se sa sigurnošću znati stvarna vrijednost, ali s obzirom na stare podatke, stručna znanja u određenom području ili iskustva iz prošlosti može se napraviti procjena. Ta procjena je korisna pri izradi modela predviđanja, ali isto tako sadrži nesigurnost i rizik jer se radi o procjeni nepoznate vrijednosti.

U nekim slučajevima moguće je procijeniti raspon vrijednosti. Na primjer može se procijeniti vrijeme završetka nekog projekta ili usluge. Na temelju iskustva može se procijeniti maksimalno trajanje nekog zadatka u najgorem slučaju i minimalno trajanje tog istog zadatka u najboljem slučaju. Korištenjem raspona mogućih vrijednosti, umjesto pogađanja svake vrijednosti zasebno, može se stvoriti puno realnija slika budućih događanja. Kada se model temelji na rasponima procjena, dobivene vrijednosti također će biti u obliku raspona. Odnosno dobit će se procjena ukupne minimalne i maksimalne vrijednosti. Kada se na raspolaganju ima raspon vrijednosti mogu se razumjeti rizici i nesigurnosti modela. Monte Carlo simulacija daje informaciju o vjerojatnosti rezultata ishoda, ovisno o tome kako su se kreirali rasponi procjene. Cijela stvar funkcionira tako što se kreira neka slučajna vrijednost koja ovisi o rasponu procjena. Model se onda računa pomoću tih slučajnih vrijednosti. Rezultat modela se sprema i cijeli proces se ponavlja. Proces se ponavlja nekoliko stotina ili tisuća puta, ali se svaki puta koriste druge slučajne varijable. Treba obratiti pažnju da se tokom ponavljanja modela ne desi ponavljanje slučajnih varijabli po nekom uzorku. Kada je simulacija gotova imamo velik broj različitih vrijednosti koje su nastale na temelju slučajnih vrijednosti procijenjenog raspona. Ti se rezultati onda koriste za opis mogućnosti ili vjerojatnosti dostizanja različitih rezultata modela. Kao kod svakog modela predviđanja simulacija će biti onoliko dobra koliko je dobra procjena onoga koji kreira model. Bitno je zapamtiti da simulacija daje samo vjerojatnosti nekog događaja, ne i točnost te vjerojatnosti. Kada se na kraju dobiju točne vrijednosti one se mogu iskoristiti za adaptaciju modela simulacije, odnosno iza izračun greške. Ako se čitava situacija dovoljno dobro organizira, buduće procjene bi mogle puno bolje reprezentirati buduće događaje. Ako se za procjenu koriste različite varijable s različitim atributima i ako se vremenom promatra ponašanje varijabli i atributa i na temelju toga stvori dovoljno dobra procjena, vrlo je vjerojatno da će i rezultati simulacije biti puno pouzdaniji.

Monte Carlo simulacije ne napreduju u vremenu, neovisne su o vremenu u tom smislu riječi za razliku od drugih simulacija. Kod Monte Carlo simulacija uzima se određeni dio neke procedure, određen vremenski period, koji je bitan i stalno se ponavlja isti taj dio sa različitim brojevima. U tom smislu broj koraka odnosno broj ponavljanja mora biti dovoljno velik kako bi se simulacija mogla stabilizirati. Jer se može desiti da slučajna vrijednost u prvom drugom i trećem koraku bude neki jako velik broj, zbog čega će i vrijednost na kraju simulacije biti velika, ako se dakle uzme premali broj ponavljanja rezultati će vrlo vjerojatno biti statistički neprihvatljivi.

# 5. PRIMJENA METODA OPERACIJSKIH ISTRAŽIVANJA

U ovom poglavlju će se na temelju dva primjera pokazati primjena OI u uslužnim djelatnostima, primjenom programskog paketa MATLAB pokušat će se doći do rješenja ukoliko je to moguće.

Prvi primjer odnosi se na ugovaranje police osiguranja. Na raspolaganju imamo velik broj realnih podataka o korisnicima raznih polica osiguranja, zadatak je s obzirom na određene karakteristike korisnika procijeniti ugovaranje nove police osiguranja.

U drugom primjeru na raspolaganju imamo podatke o dnevnim dolascima i odlascima pacijenata u domu zdravlja. Simulirati će se sustav ambulante, u kojem 8 liječnika opće medicine pruža usluge u gradu s otprilike 22 tisuće stanovnika. Primijeniti će se Monte Carlo simulacija, podatke potrebne za simulaciju pružili su nam liječnici opće prakse u Dugom Selu.

## 5.1. Agencija za ugovaranje police osiguranja

Prema izvoru podataka [7] dani su podatci za rudarenje: opis zadatka, opis varijabli, podatci za testiranje sa rezultatom, podatci za testiranje bez rezultata, stvarni rezultati.

### *5.1.1. Opis zadatka*

Agencija koja ugovara police osiguranja želi predvidjeti koje su osobe zainteresirane za kupnju police osiguranja mobilne kućice. Cilj agencije je opisati potencijale klijente u svrhu predviđanja zainteresiranosti korisnika. Boljim poznavanjem korisnika postiže se optimalna učinkovitosti zaposlenika (manje nepotrebnih sastanaka, e – mail poruka, ciljane promotivne ponude). Na kraju se pomoću zaključaka o korisnicima mogu formirati modeli maksimalnog profita.

Na temelju 5822 korisnika, raspoređenih unutar 86 kategorija sastavlja se tablica podataka.

Agencija na raspolaganju ima:

1. sociološko – demografske podatke o korisnicima,
2. ugovorene police osiguranja korisnika.

Potrebno je predvidjeti koji su korisnici zainteresirani za ugovaranje police osiguranja mobilne kućice te opisati trenutne ili potencijalne korisnike. Treba objasniti zašto baš ti korisnici ugovaraju policu osiguranja mobilne kućice.

### *5.1.2. Opis varijabli*

Atributi su podijeljeni u dvije jednako velike grupe, kao što je navedeno u zadatku. Sociološko – demografski podatci o korisniku predstavljaju 43 atributa, ti podatci izvedeni su iz poštanskog broja korisnika. Odnosno svi korisnici s istim poštanskim brojem imaju iste sociološko – demografske atribute.

Ugovorene police osiguranja korisnika predstavljaju 42 atributa. Prvi dio se odnosi na kategoriju doprinosa za pojedine vrste polica osiguranja, a drugi dio na broj ugovorenih polica korisnika.

Stoga je jasno da varijable koje opisuju atribute predstavljaju pripadnost određenoj kategoriji ili količinu [Tablica 26]. Količina iskazana u obliku cijelog broja, kategorija također ima cjelobrojne vrijednosti. Zadnji atribut ima svojstvo da može predstavljati i količinu i kategoriju, jer je maksimalan broj ugovorenih polica 1.

Tablica 26. Varijable koje opisuju atribute

| Opis atributa | | Opis varijable | |
|---|---|---|---|
| | | količina | kategorija |
| 1 | Sporedni tip korisnika | / | 1 – 41 |
| 2 | Prosječna starost | / | 1 – 6 |
| 3 | Broj kuća | Cijeli broj | / |
| 4 | Prosječna veličina kućanstva | / | 1 – 6 |
| 5 | Glavni tip korisnika | / | 1 – 10 |
| 6 – 43 | Vjerska pripadnost, bračno stanje, prihodi, edukacija | / | 0 – 9 |
| 44 – 64 | Doprinos police različitih osiguranja | / | 0 – 9 |
| 65 – 85 | Broj ugovorenih polica različitih osiguranja | Cijeli broj | / |
| 86 | Broj ugovorenih polica osiguranja mobilnih kućica | Cijeli broj | 0 – 1 |

### *5.1.3. Opis rješenja zadatka*

Podatci se stavljaju u tablicu koja mora imati oblik (151), takva matrica podataka sadrži varijable koje utječu na funkciju cilja i podatke koji sadrže specifične vrijednosti funkcije cilja u ovisnosti o pripadajućim varijablama.

$$\begin{pmatrix} x_1 & x_2 & \dots & x_n & y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_m & x_m & \dots & x_m & y_m \end{pmatrix} \tag{151}$$

Nakon toga se počinje s obradom podataka koja će za rezultat imati atribute (6) sadržane unutar $m$ x ($n$ + 1) matrice **X**, te rezultate (7) sadržane unutar $m$D vektora $\boldsymbol{y}$.

$$\boldsymbol{X} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & \vdots & \vdots \\ x_m & x_m & \dots & x_m \end{bmatrix}_{m \times n} \tag{152}$$

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} \tag{153}$$

Prije nego li se krene s ovim korakom provjerava se broj setova podataka i broj atributa, ne smije se desiti da je broj atributa veći od broja podataka.

Podatci se moraju dovesti u stanje u kojem je moguće napraviti inverz umnoška $\mathbf{X}^T\mathbf{X}$. Ukoliko to nije moguće podatci se provjeravaju. Broj atributa ne smije biti veći od broja podataka, odnosno $m > n$ i ne smije postojati linearna ovisnost između atributa Atributi se moraju razumno odabrati, treba znati kako pojedini atribut utječe na rezultat i treba li smanjiti broj atributa ili prikupiti veći broj podataka.

Varijable koje su opisane u prethodnom dijelu se prvo moraju normalizirati. Budući da Moramo smanjiti raspon $0 \le x_i \le 41$,ovaj raspon je ovako velik jer prvi atribut sadrži velik broj kategorija. Stoga ćemo mi smanjiti taj raspon na $1 \le x_1 \le 5$, nakon čega je raspon naših podataka unutar razumnih granica $0 \le x_i \le 10$.

Nakon normalizacije dodaje se stupac u matrici **X**, koji se sastoji od $m$ – dimenzionalnog vektora popunjenog jedinicama. Svaki sljedeći stupac predstavlja jedan atribut, a svaki red matrice **X** je jedan set podataka iz kojih se uči.

Za učenje se koristi samo dio podataka, a ostali podatci služit će za provjeru algoritma. Najčešće se 2/3 podataka koristi za učenje, a ostatak 1/3 podataka je testni skup.

Recimo da imamo neku funkciju $h$ koja ovisi varijablama **X**, i parametrima $\boldsymbol{\theta}$. Funkcija $h$ služi za procjenu vrijednosti rezultata $y$. Parametri $\boldsymbol{\theta}$ odabrati će se tako da se funkcija troška minimizira. Funkcija troška predstavlja prosjek sume kvadratne razlike između rezultata funkcije $h$ *i* rezultata $y$ (iz skupa za učenje), tj srednje kvadratno odstupanje. Funkcija $h$ ovisi vrijednosti $x$, a funkcija troška o parametru $\theta$. Svaka vrijednost $\theta$ odgovara jednoj drugoj hipotezi, i za svaku novu hipotezu dobije se nova vrijednost funkcije troška. Dakle može se reći da je cilj pronaći optimalni $\boldsymbol{\theta}$ koji minimizira funkciju troška. Zato ćemo postaviti neki početni $\boldsymbol{\theta}$ i malo po malo ga smanjivati kako bi dobili što manju funkciju troška.

Cilj je pronalaženje optimalne vrijednosti nekog vektora $\boldsymbol{\theta}$ koji će pomoći u predviđanju funkcije cilja. Vektor $\boldsymbol{\theta}$ opisuje kako se funkcija cilja ponaša ovisno o zadanim varijablama. Opisuje ponašanje funkcije cilja u odnosu na rast ili pad vrijednosti određene varijable. Naučeni obrasci ponašanja služe za predviđanje funkcije cilja u novim, nepoznatim okolnostima. Varijable nove okolnosti su opisno iste varijable različitih vrijednosti, a funkcija cilja je nepoznata.

Problem spada *n*D probleme, riješen je NLP, iterativnom gradijentnom metodom. Gradijentna metoda spada u grupu najčešće korištenih metoda, pripada numeričkim metodama optimiranja. Koristi se za linearne i nelinearne funkcije cilja, koje mogu, ali ne moraju imati ograničenja. Unutar nelinearnog programiranja [Slika 4] može se svrstati u funkcije više varijabli koje koriste derivacije. Cilj metode je iterativno približavanje optimumu zadane funkcije po gradijentnoj trajektoriji. Trajektorija je u 2D problemima, okomita na jednako udaljene nivo linije u konturnom dijagramu zadane funkcije. U njenim točkama postiže se najveća promjena, najveći prirast ili pad funkcije cilja.

Rješenje je dobiveno programskim paketom MATLAB, pripadajući kod nalazi se u prilogu[I]. Gradijentna metoda NLP odabrana je iz razloga što za razliku od klasifikacije u budućnosti može dati bolje rezultate, ako se odluči konkretnije sakupljati podatke o korisnicima. Agencija nije sakupila relevantne sociološko – ekonomske atribute koji opisuju korisnika, podatci su jako grubo klasificirani. Iz tog razloga rezultati koji se postižu zanemarivanjem sociološko – demografskih podataka isti su kao i kada ih uzmemo u obzir. Jedini atribut iz ove skupine koji utječe na promjenu je atribut 43 koji opisuje kupovnu moć korisnika. Ostali atributi iz te grupe su irelevantni.

### *5.1.4. Interpretacija rezultata*

Najjači faktor predviđanja je doprinos police osiguranja automobila i broj police osiguranja automobila. Odnosno najjači pojedinačni faktor predviđanja je maksimalna razina (kategorija 6) doprinosa jedne police osiguranja automobila ili dvije police osiguranja.

Drugi statistički relevantni faktori predviđanja:

- *Visoka razina kupovne moći* (najviše kategorija 7, kategorija 5 ili više)
- *Posjedovanje police privatnog osiguranja od odgovornosti*
- *Posjedovanje police osiguranja broda*
- *Posjedovanje police osiguranja socijalnog osiguranja*
- *Posjedovanje police osiguranja od požara* (razina 4)

Intuitivno možemo zaključiti ovo osiguranje koriste klijenti koji su bogatiji od prosjeka i koji se općenito osiguravaju više od prosjeka.

Vjerojatnost da korisnik s nižom premijom osiguranja automobila (razina5) ima osiguranu mobilnu kamp kućicu manja je od prosjeka. Vjerojatno je ova skupina korisnika siromašnija pa stoga postoji manja vjerojatnost posjedovanja kamp prikolice.

S obzirom da od ukupnog broja podataka (5800) za učenje samo 389 korisnika posjeduje kamp prikolicu podatci za učenje nisu baš dobri, potreban je set sa većim brojem korisnika koji imaju kamp prikolicu.

## 5.2. Simulacija sustava ambulante

Na raspolaganju imamo podatke o dnevnim dolascima i odlascima pacijenata u domu zdravlja. Simulirat će se sustav ambulante, u kojem 8 liječnika opće medicine pruža usluge u gradu s otprilike 22 tisuće stanovnika. Primijeniti će se Monte Carlo simulacija, podatke potrebne za simulaciju pružili su nam liječnici opće prakse u Dugom Selu.

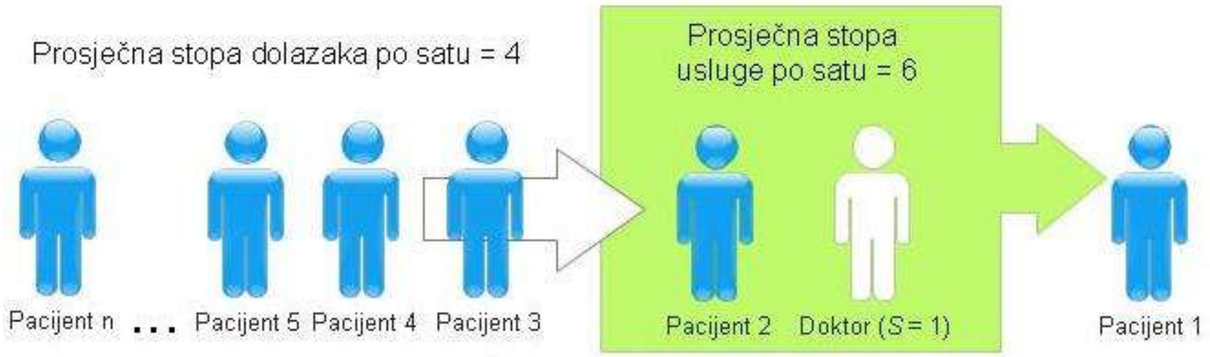
### *5.2.1. Podatci o dolascima i odlascima*

Prema evidenciji liječnika svaki dan se pojavi između 130 i 140 pacijenata koji traže usluge ambulante. Oko 60% pacijenata se za pregled naručuje, a prema liječnicima oko 14 % naručenih pacijenata se ne pojavi na pregledu, 46% pacijenata dođe na vrijeme, a ostalih 40% koji se pojave nisu imali zakazan sastanak. Kod nekih liječnika nije moguće naručivanje.

U ambulanti se radi u smjenama odnosno u svakoj smjeni su 4 dežurna liječnika. Liječnici u prosijeku mogu primiti maksimalno 33 pacijenta dnevno. Odnosno u prosjeku liječnik na pacijenta potroši oko 15 minuta. Liječnici Dnevno naručuju maksimalno 15 pacijenata. Dnevna pauza je 30 minuta.

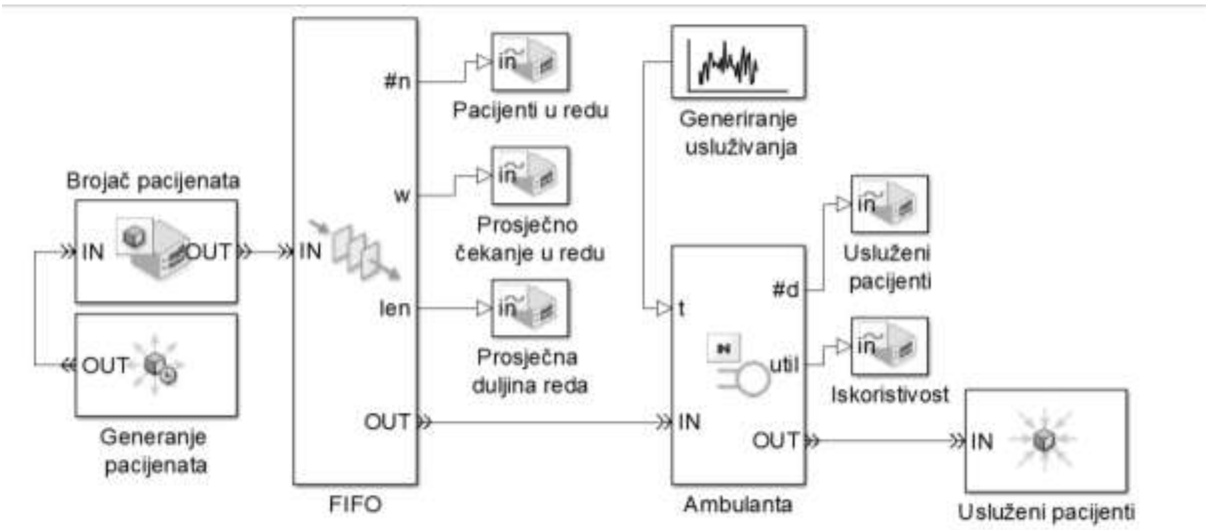
U danom trenutku kapacitet ambulante omogućava pregled 4 pacijenta, odnosno svaki liječnik može primiti samo jednog pacijenta u danom trenutku. Što znači da se radi o jednostrukom repu čekanja kada gledamo na sustav pojedinog liječnika i višestrukom repu čekanja kada gledamo na cjelokupni sustav.

Na shemi [Slika 21] se može vidjeti slijed događaja. Prikazan je beskonačan broj dolazaka odnosno beskonačan kapacitet reda uz beskonačan izvor pozivanja.




Slika 21. Shema slijeda događaja

Za rješavanje ovog primjera koristit će se programski paket MATLAB. Potrebno je izraditi sustav naveden u primjeru. Na slici [Slika 22] možemo vidjeti pojedine dijelove gotovog sustava.




Slika 22. Model ambulante

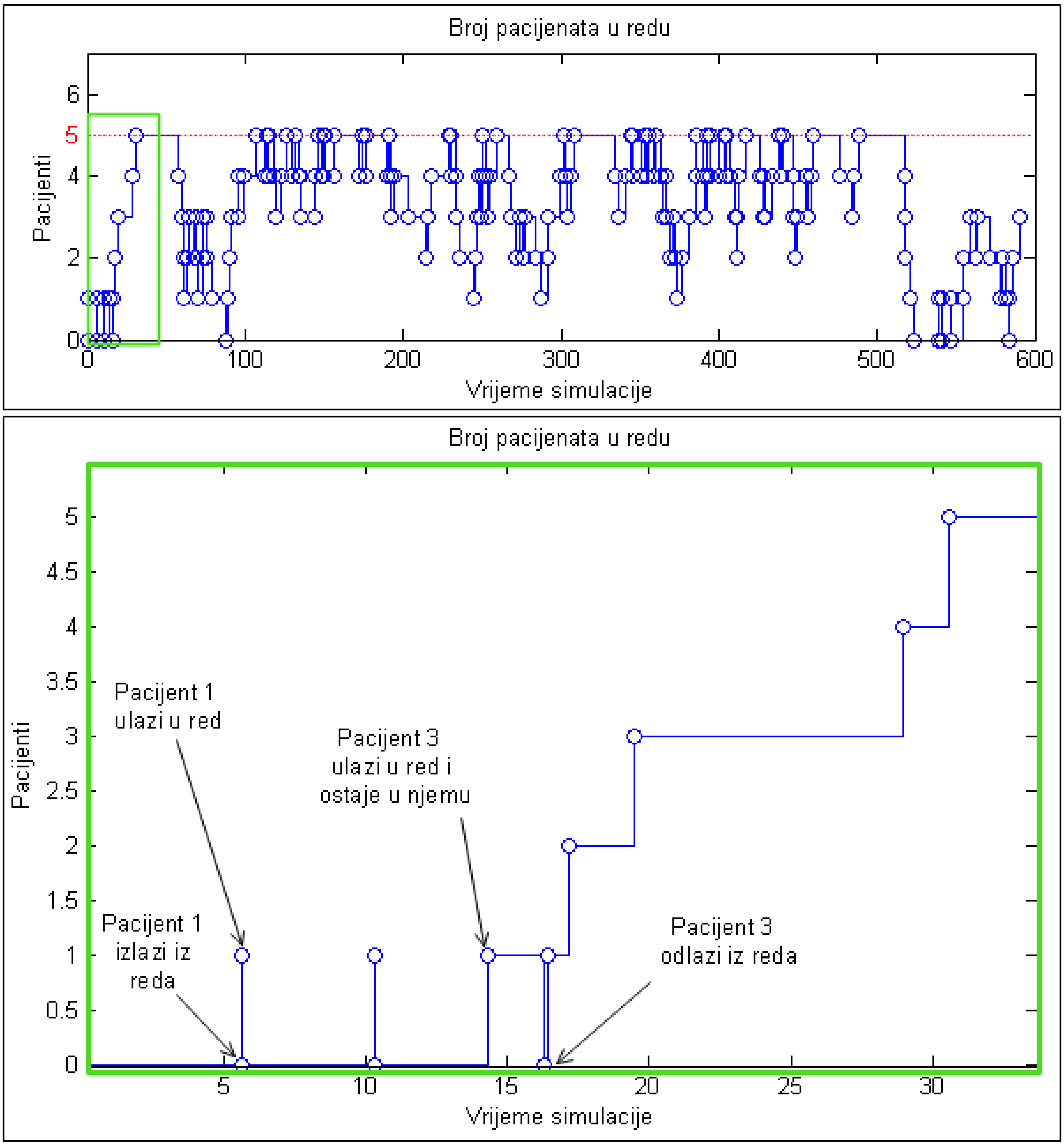
Prvo je potrebno generirati pacijente po eksponencijalnoj razdiobi, a to se postiže blokom *Generiranje pacijenata*. Blok *Brojač pacijenata* služi da bi uvijek imali informaciju o broju pacijenata koji pristižu u sustav. Nakon toga postavljamo disciplinu reda, u ovom slučaju koristit će se FIFO disciplina reda. U bloku *FIFO* mogu se vidjeti izlazi *#n*, *w*, *len*, *OUT*. Oni daju informacije o prosječnom vremenu čekanja (izlaz *w*), broju pacijenata u redu (izlaz *#n*) i prosječnoj duljini reda (izlaz *len*). Izlaz *OUT* služi za transfer entiteta kroz sustav. Kada pacijenti dođu na red ulaze u ambulantu gdje se vrši usluga. U bloku *Ambulanta* se mogu vidjeti izlazi *#d* i *util* - oni daju informaciju o broju usluženih pacijenata (izlaz *#d*) i iskoristivosti poslužitelja (izlaz *util*). Zadano je da se u ambulanti nalazi samo jedan davatelj usluge. Vrijeme trajanja usluge u ambulanti generira se prema eksponencijalnoj distribuciji korištenjem bloka *Generiranje usluživanja*. Svi izlazi koji daju informacije spojeni su na tzv., *signal scope* blokove, koji nakon završetka simulacije izbacuju grafički prikaz rezultata.

Očekivane vrijednosti mogu se izračunati jednostavnom *for* petljom [PRILOG I]. Kao rezultat će se dobiti sljedeći rezultati:

- očekivani broj jedinica u sustavu $Ls = 2$,
- očekivani broj jedinica u repu $Lq = 1.333$,
- očekivano vrijeme čekanja u sustavu $Ws = 30$ minuta,
- očekivano vrijeme čekanja u repu $Wq = 20$ minuta,

– postotak iskorištenosti $S$ = 66.67 %.

Gore navedeni rezultati ukazuju na očekivane vrijednosti, one ne moraju odgovarati stvarnim vrijednostima. Može se očekivati da će se stvarne vrijednosti nalaziti iznad i ispod očekivane. Na dijagramu [Slika 23] se može vidjeti broj pacijenata koji se nalaze u redu. Krugovi predstavljaju pacijente koji ulaze ili izlaze iz reda, a horizontalne linije predstavljaju vrijeme koje pacijent provede u redu. Može se zaključiti da nikada neće viti više od pet pacijenata u redu.



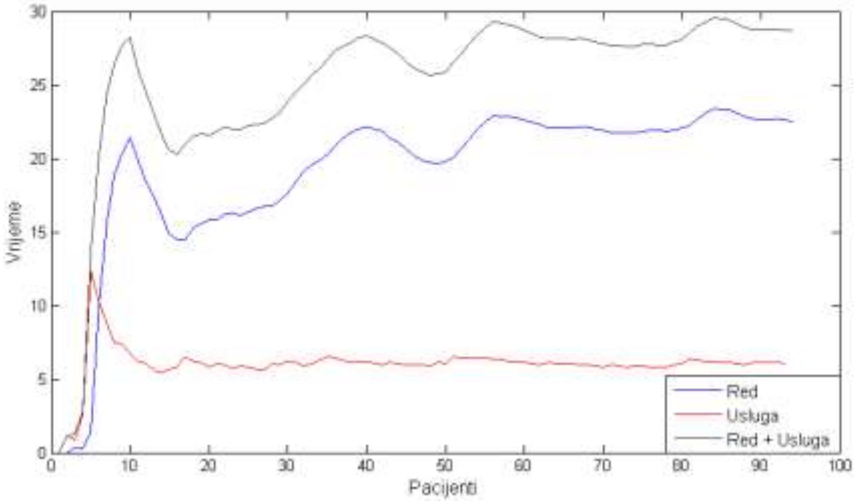

Slika 23. Broj pacijenata koji čekaju u redu

U tablici [Tablica 27] se nalaze statistički podatci koji ukazuju na ponašanje entiteta u repu čekanja. Vidljivo je da je aritmetička sredina broja pacijenata 3.293, a očekivana vrijednost je bila *Lq*=1.333. Standardna devijacija iznosi 1.457, ako se ona usporedi sa srednjom vrijednošću može se zaključiti da aritmetička sredina ne reprezentira uzorak dovoljno dobro. Prosječan broj pacijenata u redu je 3.293, a devijacija je 1.457 iz toga slijedi da je prosječno odstupanje od prosječnog broja pacijenata u redu 44%. Iz tog razloga treba uzeti u obzir vrijednost najčešće frekvencije i medijana, koji u ovom slučaju iznose 4 i daju realan podatak o broju pacijenata u redu čekanja.

Tablica 27. Statistički podatci broja pacijenata u redu čekanja

| | |
|---|---|
| min | 0 |
| max | 5 |
| mean | 3.292 |
| median | 4 |
| mode | 4 |
| std | 1.456 |
| range | 5 |

Na dijagramu [Slika 24] prikazana je usporedba različitih vremena čekanja, te ukupno prosječno vrijeme koje jedan pacijent provede u sustavu. Vidi se da je prosječno vrijeme čekanja u redu skoro duplo veće od vremena trajanja usluge. Skok na početku dijagrama ukazuje na mali broj simulacija u tom trenutku, to se desilo jer u prvom uzorku simulacije vrijeme između dolazaka jako kratko i naglo. Vidi se da se nakon nekog vremena taj dio stabilizira.




Slika 24. Prosječno vrijeme čekanja i vrijeme trajanja usluge

Iz podataka [Tablica 28] je vidljivo da se prosječno vrijeme čekanja u redu (19 minuta) i medijan prosjeka (21 minuta) podudara s očekivanim vremenom čekanja u redu ($Wq$ = 20 minuta). Očekivano vrijeme čekanja u sustavu je $W_S$ = 30 minuta, a medijan 27 minuta. Što znači da iz očekivanih vrijednosti možemo dobiti dovoljno dobru predodžbu o vremenima u sustavu i repu.

Tablica 28. Statistički podatci vremena čekanja

| | Red | Usluga | Red + Usluga |
|---|---|---|---|
| min | 0 | 0 | 0 |
| max | 23.39 | 12.32 | 29.58 |
| mean | 19.24 | 6.033 | 25.27 |
| median | 21.43 | 6.065 | 27.66 |
| mode | 0 | 0 | 0 |
| std | 5.237 | 1.363 | 5.854 |
| range | 23.39 | 12.32 | 29.58 |

# 6. ZAKLJUČAK

Operacijska istraživanja su se u uslužnim djelatnostima počela sve više razvijati. Najčešće se koriste u djelatnostima transporta i skladištenja, informacija i komunikacija, djelatnosti zdravstvene zaštite i socijalne skrbi, trgovini na veliko i na malo, financijske djelatnosti i djelatnosti osiguranja.

Operacijska istraživanja su jako kompleksno područje. Kako bi se uspješno riješio problem nekog sustava potrebno je surađivati s raznim stručnjacima iz raznih područja. Ovo se osobito odnosi na uslužne djelatnosti. Uspješno rješavanje problema ne ovisi samo o godinama iskustva stručnjaka operacijskih istraživanja, nego o sposobnosti da se uz pomoć stručnjaka koji godinama rade unutar tog sustava pronađu problemi i razlozi zbog kojih se oni javljaju.

Generalno gledajući stručnjak operacijskih istraživanja teško će riješiti problem iz nekog drugog sustava, ako tematika kojom se sustav bavi nije područje njegove struke. Kako bi se metode operacijskih istraživanja adekvatno mogle primijeniti na bilo koji problem iz bilo kojeg područja nije dovoljno biti samo stručnjak na području operacijskih istraživanja, već je potrebno biti stručnjak i u području sustava promatranog problema.

Za razlučivanje i sistematiziranje, a naročito rješavanje problema nužno je poznavanje teorije. Modificiranjem metoda se može kreirati rješenje, kod praktičnih problema. Skoro nikada nije dovoljna samo jedna metoda za rješavanje problema. Metode i matematički modeli međusobno se kombiniraju i tvore nove metode.

Obradom tematike ovog rada došlo se do zaključka da je jako komplicirano svrstati metodologiju nekog problema promatrajući sektor uslužnih djelatnosti. Djelatnosti u ovom sektoru se same po sebi jako razlikuju i uspješna implementacija neke metode u jednoj djelatnosti neće osigurati uspješnu implementaciju te iste metode u drugoj djelatnosti. Promatrajući sustav zdravstva primjećuje se da je jako teško pronaći granicu između problema, tj. teško je izvući jedan dio iz konteksta i onda očekivati adekvatna rješenja.

U zdanjem poglavlju samostalno su riješena dva zadatka, došlo se do zaključka da podatci koji su prikupljeni, odnosno podatci koji su odabrani kao relevantni u biti i nisu bili u potpunosti relevantni. Podatci u bili suviše generalizirani, iz čega se zaključuje da je nužno unutar sustava imati stručnjaka za operacijska istraživanja, koji će sugerirati moguće razloge i moguća međudjelovanja unutar sustava.

# LITERATURA


[1] Klapić D., Mornar V.: Operacijska istraživanja, Zagreb, 1996.

[2] Štefanić, N.: Operacijska istraživanja I, Bilješke sa predavanja, Zagreb, 2013.

[3] Wayne L. Winston: Operations Research applications and algorithms, Indiana, 2004.

[4] Barković D.: Operacijska istraživanja, Osijek, 1997.

[5] Vučina D.: Metode inženjerske numeričke optimizacije, Split, 2005

[6] Narodne novine, metodologija za statističku primjenu nacionalne klasifikacije djelatnosti 2007. – NKD 2007[1]; 16.11.2015., http://narodne-novine.nn.hr/default.aspx

[7] P. van der Putten, M. van Someren (eds), CoIL Challenge 2000: The Insurance Company Case, Sentient Machine Research, Amsterdam; Leiden Institute of Advanced Computer Science Technical Report; http://www.liacs.nl/~putten/library/cc2000/

[8] Yiting Xing, Ling Li, Zhuming Bi, Marzena Wilamowska-Korsak and Li Zhang: Operations Research (OR) in Service Industries: A Comprehensive Review

[9] Machine learning: http://cs229.stanford.edu/materials.html

# PRILOZI

I. Programski kod NLP, gradijentna metoda

II. Programski kod za primjer

III. CD rješenja primjera

## Programski kod NLP, gradijentna metoda

```
%% Machine Learning Online Class
%  class.coursera.org
%  Instructions by: Andrew Ng
%
%% ----------------------\Učitavanje podataka/----------------------------%
%Početak...
clear all; close all; clc;
%
fprintf('Učitavanje podataka ''za_ucenje.txt''...\n')
fprintf('Podatci o korisnicima: \n');
fprintf('Socioekonomski status(1:43)\n');
fprintf('i ugovorene police osiguranja(44:86)\n');
%
data = load('za_ucenje.txt');
%
%% ----------------------\Normalizacija podataka/------------------------%
% Kakvi su podatci?
% 1.Normalizacija kategorija
% 2.Normalizacija brojeva
% !!! Oprez   ono što predvišamo se ne normalizira !!!
fprintf('Normalizacija varijabli... \n')
%-----------------------\Normalizacija kategorija/-----------------------%
% Budući da su podatci kategorije 1:41, 1:6, 1:9, 1:12 itd.
% Kategorizacija će se prilagoiti (preveliki rasponi)
% Biramo kategorije 1, 1.1, 1.2, ... ,4.8, 4.9, 5
% Raspon je sada puno bolji! Prije(1:41), sada(1:5)
%% Machine Learning Online Class
%  class.coursera.org
%  Instructions by: Andrew Ng
%
%% ----------------------\Učitavanje podataka/----------------------------%
%Početak...
clear all; close all; clc;
%
fprintf('Učitavanje podataka ''za_ucenje.txt''...\n')
fprintf('Podatci o korisnicima: \n');
fprintf('Socioekonomski status(1:43)\n');
fprintf('i ugovorene police osiguranja(44:86)\n');
%
data = load('za_ucenje.txt');
%
%% ----------------------\Normalizacija podataka/------------------------%
% Kakvi su podatci?
% 1.Normalizacija kategorija
% 2.Normalizacija brojeva
% !!! Oprez   ono što predvišamo se ne normalizira !!!
fprintf('Normalizacija varijabli... \n')
%
data=[changem(data(:,2:end),[1:0.1:5]',[1:41]'),data(:,1)]; % zamjena
X = data(:, 1:end-1); y =data(:,end);
% definiranje dimenzija matrice
m = size(X,1); n = size(X,2);
%--------------------------\Normalizacija brojeva/-----------------------%
% % na srednju vrijednost 0
% % [X mu sigma] = featureNormalize(X)
% mu = mean(X); sigma = std(X); t = ones(size(X,1), 1);
% X_norm = (X - (t * mu)) ./ (t * sigma); X=X_norm;
```

```
% % Dodavanje dodatnog atributa Xo
X = [ones(m, 1) X];

%% ----------------------\Gradijentno približavanje/----------------------%
fprintf('Gradijentno približavanje ...\n');
% Odabir koraka učenja (0.1 : 0.0001) i broja iteracija
alpha = 0.0257; num_iters = 3000;
% Odabir početne vrijednosti theta
theta = zeros(n+1, 1);
m = size(X,1);J_history = zeros(num_iters, 1);
for iter = 1:num_iters
    theta = theta - alpha * (1/m) * (((X*theta) - y)' * X)';
    J_history(iter)=computeCostMulti(X, y, theta);
    J = computeCostMulti(X, y, theta);
    J = 0;
    J = (1/(2*m)) * (X*theta-y)' * (X*theta-y);
end
% Prikaz grafa konvergencije
hold on; plot(1:numel(J_history),J_history,'-r','LineWidth',1);
xlabel('Broj iteracija'); ylabel('Funkcija "troška"');
% Rezultat:
fprintf('Theta izračunat gradijentnim približavanjem: \n');
fprintf(' %f \n', theta);
fprintf('\n');
fprintf('Pritisnite enter za nastavak.\n');
pause;
%
%% ----------------------------\Procjena/------------------------------%
theta1 = (X'*X)^(-1)*(X'*y);
y1=X*theta1;
y2=X*theta;
data_test = [load('testiranje.txt'), load('pravi.txt')];
y_test = data_test(:, 1);
nk=[1:0.1:5]';sk=[1:41]';data_test=[changem(data_test(:,2:end),nk,sk)];
X_test = data_test(:, 1:end); X_test = [ones(size(X_test,1), 1) X_test];
m = size(X_test,1); n = size(X_test,2);
y_procjena=X_test*theta1;
Y=round(y_procjena);
fprintf('Procjena cilja: %f \n',Y);
tocno=ismember(round(y_procjena),y_test, 'rows');
tocno=size(tocno(tocno==1),1);
fprintf('Dobro procjenjeni: %f \n', tocno);

%
data=[changem(data(:,2:end),[1:0.1:5]',[1:41]'),data(:,1)]; % zamjena
X = data(:, 1:end-1); y =data(:,end);
% definiranje dimenzija matrice
m = size(X,1); n = size(X,2);
%-------------------------\Normalizacija brojeva/-----------------------%
% % na srednju vrijednost 0
% % [X mu sigma] = featureNormalize(X)
% mu = mean(X); sigma = std(X); t = ones(size(X,1), 1);
% X_norm = (X - (t * mu)) ./ (t * sigma); X=X_norm;
% % Dodavanje dodatnog atributa Xo
X = [ones(m, 1) X];

%% ----------------------\Gradijentno približavanje/----------------------%
fprintf('Gradijentno približavanje ...\n');
% Odabir koraka učenja (0.1 : 0.0001) i broja iteracija
```

```
alpha = 0.0257; num_iters = 3000;
% Odabir početne vrijednosti theta
theta = zeros(n+1, 1);
m = size(X,1);J_history = zeros(num_iters, 1);
for iter = 1:num_iters
    theta = theta - alpha * (1/m) * (((X*theta) - y)' * X)';
    J_history(iter)=computeCostMulti(X, y, theta);
    J = computeCostMulti(X, y, theta);
    J = 0;
    J = (1/(2*m)) * (X*theta-y)' * (X*theta-y);
end
% Prikaz grafa konvergencije
hold on; plot(1:numel(J_history),J_history,'-r','LineWidth',1);
xlabel('Broj iteracija'); ylabel('Funkcija "troška"');
% Rezultat:
fprintf('Theta izračunat gradijentnim približavanjem: \n');
fprintf(' %f \n', theta);
fprintf('\n');
fprintf('Pritisnite enter za nastavak.\n');
pause;
%
%% ------------------------------\Procjena/------------------------------%
theta1 = (X'*X)^(-1)*(X'*y);
y1=X*theta1;
y2=X*theta;
data_test = [load('testiranje.txt'), load('pravi.txt')];
y_test = data_test(:, 1);
nk=[1:0.1:5]';sk=[1:41]';data_test=[changem(data_test(:,2:end),nk,sk)];
X_test = data_test(:, 1:end); X_test = [ones(size(X_test,1), 1) X_test];
m = size(X_test,1); n = size(X_test,2);
y_procjena=X_test*theta1;
Y=round(y_procjena);
fprintf('Procjena cilja: %f \n',Y);
tocno=ismember(round(y_procjena),y_test, 'rows');
tocno=size(tocno(tocno==1),1);
fprintf('Dobro procjenjeni: %f \n', tocno);

%
data=[changem(data(:,2:end),[1:0.1:5]',[1:41]'),data(:,1)]; % zamjena
X = data(:, 1:end-1); y =data(:,end);
```

# Programski kod teorija repova čekanja

```
clear all
clc
% Teorija repova (M/M/1):(GN/inf/inf)
%
% Izračun očekivanih vrijednosti,
% potrebne varijable: lambda, mu, n
%
% Simulacija repova čekanja,
% potrebne varijable: vsim
%% Tanja Protić
%%
%
fprintf('Teorija repova (M/M/1):(GN/inf/inf) \r\n')
%
%----------------------------\Unos podataka/------------------------------%
%
lambda = input('Upišite prosječnu stopu dolazaka po satu: ');
mu = input('Upišite prosječnu stopu usluge po satu: ');
m = input('Upišite broj jedinica u sustavu: '); % = beskonačno
s=1;     %broj uslužnih mjesta
r=lambda/mu;    % intenzitet "prometa" u repu čekanja

%---------------------\Izračun očekivanih vrijednosti/--------------------%
%
% pn - vjerojatnost stabilnog stanja
%  n - zadani broj jedninica u sustavu
% Ls - očekivani broj jedinica u sustavu
% Lq - očekivani broj jedinica u repu
% Ws - očekivano vrijeme čekanja u sustavu
% Wq - očekivano vrijeme čekanja u repu
%  S - postotak iskorištenosti
%  s - broj uslužnih mjesta
%
%-------------------------------------------------------------------------%

p0=1-r;     %suma geometrijskog reda
for n=1:m;
    pn=r^(n-1)*p0;
    Ls=(n-1)*pn;
    pn1(n)=pn;      %kreira vektor pn1
    Ls1(n)=Ls;      %kreira vektor Ls1
end
Ls=sum(Ls1);
Ws=Ls/lambda;
Ws1=Ws*60;      % iz sata u u minute
Wq=Ws-1/mu;
Wq1=Wq*60;      % iz sata u u minute
```