

SVEUČILIŠTE JURJA DOBRILE U PULI
FAKULTET INFORMATIKE

Neven Nižić

**”Optimizacija raspodjele projektnih aktivnosti primjenom genetskih
algoritama i Monte Carlo simulacije”**

DIPLOMSKI RAD

Pula, rujan, 2025. godine

SVEUČILIŠTE JURJA DOBRILE U PULI
FAKULTET INFORMATIKE

Neven Nižić

**”Optimizacija raspodjele projektnih aktivnosti primjenom genetskih
algoritama i Monte Carlo simulacije”**

DIPLOMSKI RAD

**JMBAG: 0303118917, izvanredni student
Studijski smjer: Informatika**

**Kolegij: Modeliranje i simulacije
Znanstveno područje : Društvene znanosti
Znanstveno polje : Informacijske i komunikacijske znanosti
Znanstvena grana : Informacijski sustavi i informatologija**

Mentor: doc.dr.sc. Darko Etinger

Pula, rujan, 2025. godine

Sažetak

Upravljanje projektima često uključuje složene odluke vezane uz raspodjelu aktivnosti i resursa, osobito u uvjetima nesigurnosti i vremenskih ograničenja. Tradicionalne metode kao što su PERT i CPM često ne uspijevaju obuhvatiti stohastičku prirodu stvarnih projekata. U ovom radu predlaže se model koji kombinira genetske algoritme i Monte Carlo simulaciju s ciljem postizanja robusne optimizacije raspodjele projektnih aktivnosti. Genetski algoritam koristi se za pretraživanje prostora mogućih rješenja, dok Monte Carlo simulacija omogućava procjenu utjecaja varijabilnih trajanja aktivnosti na ukupno trajanje projekta. Eksperimentalna evaluacija modela provodi se na simuliranim projektnim podacima, a rezultati pokazuju poboljšanu robusnost i učinkovitost u odnosu na determinističke pristupe.

Ključne riječi : projektno upravljanje, genetski algoritam, Monte Carlo simulacija, optimizacija rasporeda, raspodjela aktivnosti

Abstract

Project management often involves complex decisions related to the allocation of activities and resources, especially under uncertainty and time constraints. Traditional methods such as PERT and CPM frequently fail to capture the stochastic nature of real-world projects. This thesis proposes a model that combines genetic algorithms and Monte Carlo simulation to achieve robust optimization of project activity allocation. The genetic algorithm is used to explore the space of possible solutions, while Monte Carlo simulation estimates the impact of variable activity durations on the overall project timeline. The model is experimentally evaluated using simulated project data, and results indicate improved robustness and efficiency compared to deterministic approaches.

Keywords : project management, genetic algorithm, Monte Carlo simulation, schedule optimization, activity allocation

Sadržaj

1	Uvod	1
1.1	Uvodni dio	1
1.2	Motivacija	1
1.3	Rizici i nesigurnosti u projektnom upravljanju	1
1.4	Ciljevi i istraživačka pitanja	2
2	Teorijska podloga	4
2.1	Knapsack problem	4
2.2	Genetski algoritmi	4
2.3	Monte Carlo simulacija	4
2.4	PERT metoda i trokutasta distribucija	5
3	Model problema	6
3.1	Ograničenja	6
3.2	Cilj optimizacije	6
3.3	Grafički prikaz modela	6
3.4	Zaključak poglavlja	7
4	Implementacija	8
4.1	Korištene biblioteke	8
4.2	Struktura implementacije	8
4.2.1	Monte Carlo modul	10
4.2.2	GA modul	10
4.2.3	Vizualizacija rezultata	10
5	Eksperimenti	12
5.1	Testni podaci	12
5.2	Scenariji testiranja	12
5.3	Metodologija	12
5.4	Plan prezentacije rezultata	13
6	Zaključak	14
	Literatura	15
	Popis slika	17
	Popis tablica	18

1 Uvod

1.1 Uvodni dio

Upravljanje projektnim aktivnostima predstavlja složen izazov zbog međusobne povezanosti zadataka, ograničenih resursa i neizvjesnosti u procjeni trajanja i troškova [1]. U kontekstu modernog upravljanja projektima, pravovremena i optimalna raspodjela resursa od ključne je važnosti za uspjeh projekta. Pritom se suočavamo s problemom koji je po svojoj prirodi *NP-težak* [2], što znači da optimalno rješenje nije moguće pronaći u razumnom vremenu za veće probleme primjenom klasičnih algoritama.

Jedan od modela koji dobro opisuje izazove alokacije resursa je *problem ruksaka* (*Knapsack Problem*), gdje se ograničeni resursi moraju raspodijeliti na način koji maksimizira ukupnu korist [3]. U kontekstu projektnih aktivnosti, to znači odabrati skup zadataka koji će donijeti najveći povrat ulaganja (ROI) uz poštivanje vremenskih i resursnih ograničenja. Tradicionalne metode poput PERT-a (*Program Evaluation and Review Technique*) i CPM-a (*Critical Path Method*) [1, 4] pružaju okvir za planiranje i praćenje projekata, no često zanemaruju složene kombinacije zadataka i inherentnu neizvjesnost u procjenama trajanja i troškova.

Pregled literature pokazuje da se genski algoritmi (GA) uspješno primjenjuju na NP-teške optimizacijske probleme [5, 6], dok Monte Carlo simulacije (MC) omogućuju modeliranje i kvantifikaciju neizvjesnosti [7, 8]. Međutim, u literaturi postoji ograničen broj radova koji integriraju ove dvije metode u jedinstven hibridni pristup optimizaciji projektnih aktivnosti [9, 10]. Praznina se očituje u nedostatku sustavne analize kako kombinacija GA i MC može povećati robusnost rješenja u uvjetima visoke neizvjesnosti i složenih resursnih ograničenja.

1.2 Motivacija

Motivacija za ovaj rad proizlazi iz činjenice da se u stvarnim projektnim uvjetima planovi rijetko odvijaju točno onako kako su prvotno predviđeni [1]. Promjene u dostupnosti resursa, nepredviđeni rizici, promjene zahtjeva te pogrešne procjene trajanja zadataka česti su razlozi odstupanja od plana. Klasične metode planiranja projekata, iako korisne, često ne uzimaju u obzir dinamičke promjene i stohastičku prirodu projektnih parametara [11].

Upravo zbog toga potrebne su metode koje ne samo da optimiziraju raspodjelu resursa, već i uvažavaju nesigurnosti te omogućuju procjenu rizika povezanih s odabranim rješenjima. Genski algoritmi pružaju snažan alat za globalnu optimizaciju složenih problema pretrage [6], dok Monte Carlo simulacije omogućuju procjenu varijabilnosti i rizika rješenja [8]. Kombiniranjem ovih metoda moguće je razviti pristup koji generira kvalitetna rješenja, a pritom osigurava njihovu robusnost u uvjetima nesigurnosti [10].

1.3 Rizici i nesigurnosti u projektnom upravljanju

Rizici u projektnom upravljanju obuhvaćaju sve događaje ili uvjete koji, ako se pojave, mogu negativno utjecati na ciljeve projekta [12]. Oni mogu biti tehničke, or-

ganizacijske, financijske ili tržišne prirode, a često su međusobno povezani [11]. Nesigurnosti proizlaze iz nepotpunih informacija, promjenjivih uvjeta i nemogućnosti preciznog predviđanja budućih događaja [13]. Upravljanje rizicima uključuje njihovu identifikaciju, procjenu i razvoj strategija odgovora [12].

Monte Carlo simulacije predstavljaju jedan od najučinkovitijih pristupa kvantitativnoj procjeni rizika [14]. Kroz generiranje velikog broja scenarija temeljenih na slučajnim varijacijama ulaznih parametara, moguće je procijeniti distribuciju mogućih ishoda i vjerojatnosti ostvarenja ciljeva projekta [8].

1.4 Ciljevi i istraživačka pitanja

Ovaj diplomski rad ima za cilj razviti i evaluirati hibridni GA-MC model za optimizaciju raspodjele projektnih aktivnosti. Doprinos rada je trojak:

- **Teorijski doprinos:** povezivanje optimizacijskih tehnika s metodama procjene neizvjesnosti u kontekstu projektnih aktivnosti.
- **Metodološki doprinos:** razvoj integriranog modela koji povezuje evolucijsku optimizaciju i statističku simulaciju.
- **Praktični doprinos:** demonstracija primjenjivosti pristupa na sintetičkim projektnim podacima i analiza performansi modela.

Istraživačka pitanja na koja ovaj rad želi odgovoriti su:

1. Može li hibridni GA-MC pristup postići bolje rezultate optimizacije u odnosu na samostalnu primjenu GA ili MC?
2. U kojoj mjeri Monte Carlo simulacija može poboljšati robusnost rješenja dobivenih genskim algoritmom?
3. Kakav je utjecaj kombiniranog pristupa na stabilnost i pouzdanost optimizacije u uvjetima neizvjesnosti?

Ciljevi istraživanja su:

- Dizajnirati i implementirati integrirani GA-MC model.
- Evaluirati performanse modela na različitim scenarijima.
- Usporediti rezultate hibridnog pristupa s rezultatima pojedinačnih metoda.

Sinergija između genskih algoritama i Monte Carlo simulacija proizlazi iz njihove komplementarne prirode: GA učinkovito pretražuje veliki prostor mogućih rješenja i pronalazi visokokvalitetne kandidate [6], dok MC kvantificira rizik i procjenjuje varijabilnost tih rješenja kroz stohastičko modeliranje [8]. Kombinacija ovih metoda rezultira robusnim pristupom koji ne samo da optimizira projektne aktivnosti, već i uzima u obzir realne nesigurnosti u planiranju.

Struktura rada organizirana je na sljedeći način:

- **Poglavlje 2** daje pregled relevantne literature i teorijskih osnova problema.

- **Poglavlje 3** opisuje korištenu metodologiju, uključujući principe rada GA i MC.
- **Poglavlje 4** prikazuje implementaciju hibridnog modela.
- **Poglavlje 5** opisuje plan i rezultate eksperimentalnog vrednovanja modela.
- **Poglavlje 6** sadrži zaključak i preporuke za budući rad.

2 Teorijska podloga

2.1 Knapsack problem

Problem ruksaka (engl. *Knapsack Problem*) jedan je od klasičnih problema kombinatorne optimizacije i pripada klasi NP-teških problema. U osnovnoj verziji, cilj je odabrati podskup objekata s pripadajućim težinama i vrijednostima, tako da ukupna težina ne prelazi određeni kapacitet, a ukupna vrijednost je maksimalna.

Formalno, za skup n objekata s težinama w_i i vrijednostima v_i , te kapacitet W , cilj je maksimizirati:

$$\max \sum_{i=1}^n v_i x_i \quad \text{uz ograničenje} \quad \sum_{i=1}^n w_i x_i \leq W, \quad x_i \in \{0, 1\}$$

U kontekstu upravljanja projektima, aktivnosti se mogu interpretirati kao objekti s određenim trajanjem i vrijednošću (npr. važnost, doprinos cilju), a resursi (vremenski ili financijski) kao kapacitet. Knapsack model se tada koristi za optimalnu raspodjelu resursa među projektnim aktivnostima.

2.2 Genetski algoritmi

Genetski algoritmi (GA) su metaheurističke metode inspirirane prirodnim procesom evolucije. Rješenja problema predstavljaju se kao kromosomi (obično binarni nizovi), a operacije evolucije uključuju selekciju, križanje (crossover) i mutaciju. Tipični koraci uključuju:

1. Inicijalizacija populacije
2. Evaluacija funkcije cilja (fitness)
3. Selekcija roditelja (npr. turnirska selekcija, rulet)
4. Primjena operatora križanja i mutacije
5. Zamjena stare populacije novom generacijom

Kodiranje rješenja ovisi o problemu — za raspodjelu aktivnosti često se koristi binarno ili cijelobrojno kodiranje. GA su prikladni za probleme s velikim i nepreglednim prostorom rješenja jer ne zahtijevaju informacije o gradijentima.

2.3 Monte Carlo simulacija

Monte Carlo simulacija koristi nasumično generirane ulazne podatke kako bi procijenila ponašanje složenog sustava. Ključni elementi uključuju:

- Definiranje slučajnih varijabli (npr. trajanje aktivnosti)
- Odabir prikladne distribucije (npr. uniformna, normalna, trokutasta)
- Velik broj iteracija ($n > 10^3$) za stabilne rezultate

Simulacijom velikog broja scenarija može se procijeniti rizik projekta, očekivano trajanje, kao i distribucija mogućih ishoda. U ovom radu, MC simulacija koristi se za uvođenje nesigurnosti u trajanje aktivnosti.

2.4 PERT metoda i trokutasta distribucija

PERT (Program Evaluation and Review Technique) je metoda upravljanja projektima koja uzima u obzir nesigurnost trajanja aktivnosti koristeći tri vremenske procjene:

$$T_E = \frac{T_o + 4T_m + T_p}{6}$$

gdje su:

- T_o – optimistična procjena trajanja,
- T_m – najvjerojatnija procjena,
- T_p – pesimistična procjena.

Za simulacije se često koristi trokutasta distribucija jer je jednostavna za implementaciju i dovoljno fleksibilna. Vrijednosti se biraju unutar intervala $[T_o, T_p]$ s maksimumom u T_m .

Monte Carlo simulacija s trokutastom distribucijom omogućuje realniji prikaz varijabilnosti trajanja aktivnosti u projektnom planiranju, osobito kada nedostaju povijesni podaci.

3 Model problema

Projekt optimizacije raspodjele projektnih aktivnosti može se formalno predstaviti kao skup od n zadataka, pri čemu svaki zadatak i ima definirane sljedeće karakteristike:

- **procijenjeno trajanje** d_i ,
- **trošak** c_i ,
- **vrijednost** odnosno povrat ulaganja (ROI) v_i ,
- **distribucija nesigurnosti** koja opisuje varijabilnost trajanja i/ili troška.

Ovakav formalizam omogućuje primjenu metoda kombinatorne optimizacije [15] i stohastičkih simulacija [16] za donošenje optimalnih odluka u uvjetima ograničenih resursa.

3.1 Ograničenja

Projekt je podložan realnim ograničenjima resursa, koja se u modelu izražavaju na sljedeći način:

$$\sum_{i=1}^n d_i \leq T_{\max} \quad (\text{ukupno vrijeme}) \quad (1)$$

$$\sum_{i=1}^n c_i \leq B_{\max} \quad (\text{budžet}) \quad (2)$$

$$\sum_{i=1}^n r_i \leq R_{\max} \quad (\text{maksimalan broj radnika}) \quad (3)$$

gdje T_{\max} označava raspoloživo ukupno vrijeme, B_{\max} raspoloživi budžet, a R_{\max} maksimalan broj raspoloživih radnika.

3.2 Cilj optimizacije

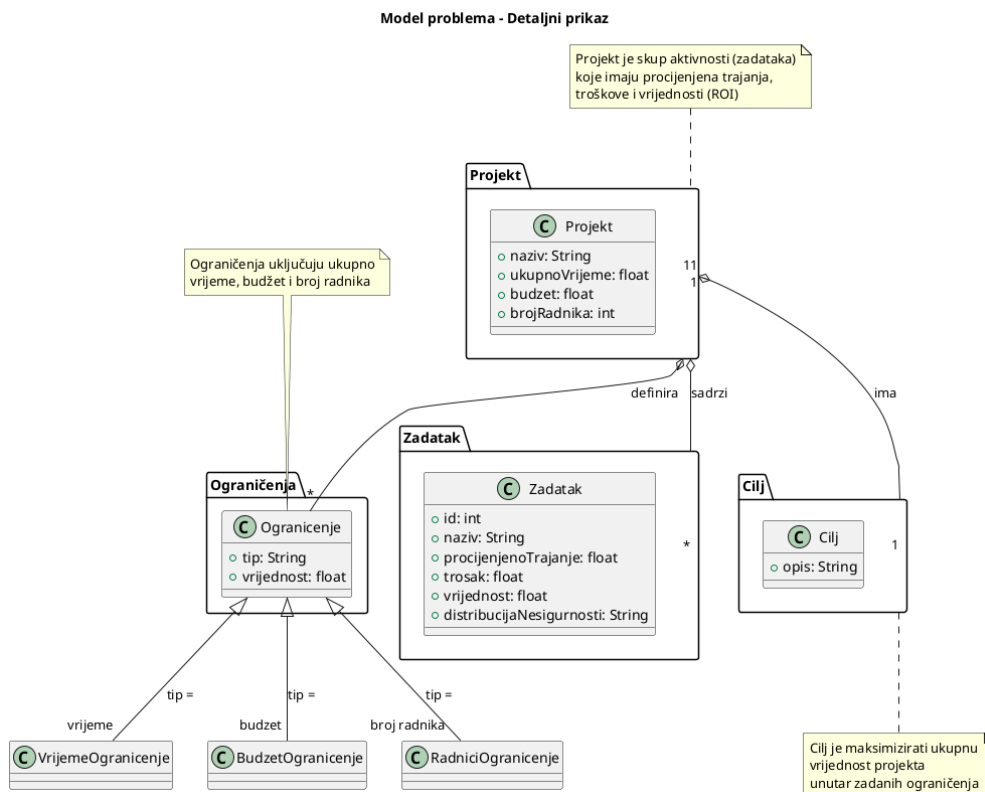
Cilj optimizacije je **maksimizirati ukupnu vrijednost projekta** ostajući unutar svih definiranih ograničenja:

$$\max \sum_{i=1}^n v_i \cdot x_i \quad (4)$$

pri čemu $x_i \in \{0, 1\}$ označava binarnu varijablu koja označava odabir zadatka i za izvršenje.

3.3 Grafički prikaz modela

Na slici 1 prikazan je konceptualni model problema optimizacije raspodjele projektnih aktivnosti, uključujući ulazne podatke, ograničenja i ciljnu funkciju.



Slika 1: Model problema optimizacije raspodjele projektnih aktivnosti

3.4 Zaključak poglavlja

Ovaj model problema omogućuje matematičku i vizualnu formalizaciju optimizacijskog zadatka. Jasna definicija ulaza, ograničenja i cilja ključna je za primjenu optimizacijskih metoda poput genetskih algoritama [6] u kombinaciji s Monte Carlo simulacijama kako bi se dobila rješenja visoke kvalitete u uvjetima nesigurnosti.

4 Implementacija

Implementacija razvijenog modela optimizacije provedena je u programskom jeziku Python, zbog njegove fleksibilnosti, bogatog ekosustava biblioteka te široke primjene u znanstvenom računarstvu i optimizacijskim metodama. Python omogućuje brz razvoj prototipa, jednostavnu integraciju različitih modula te učinkovitu vizualizaciju rezultata.

4.1 Korištene biblioteke

Za potrebe izrade sustava korištene su sljedeće biblioteke:

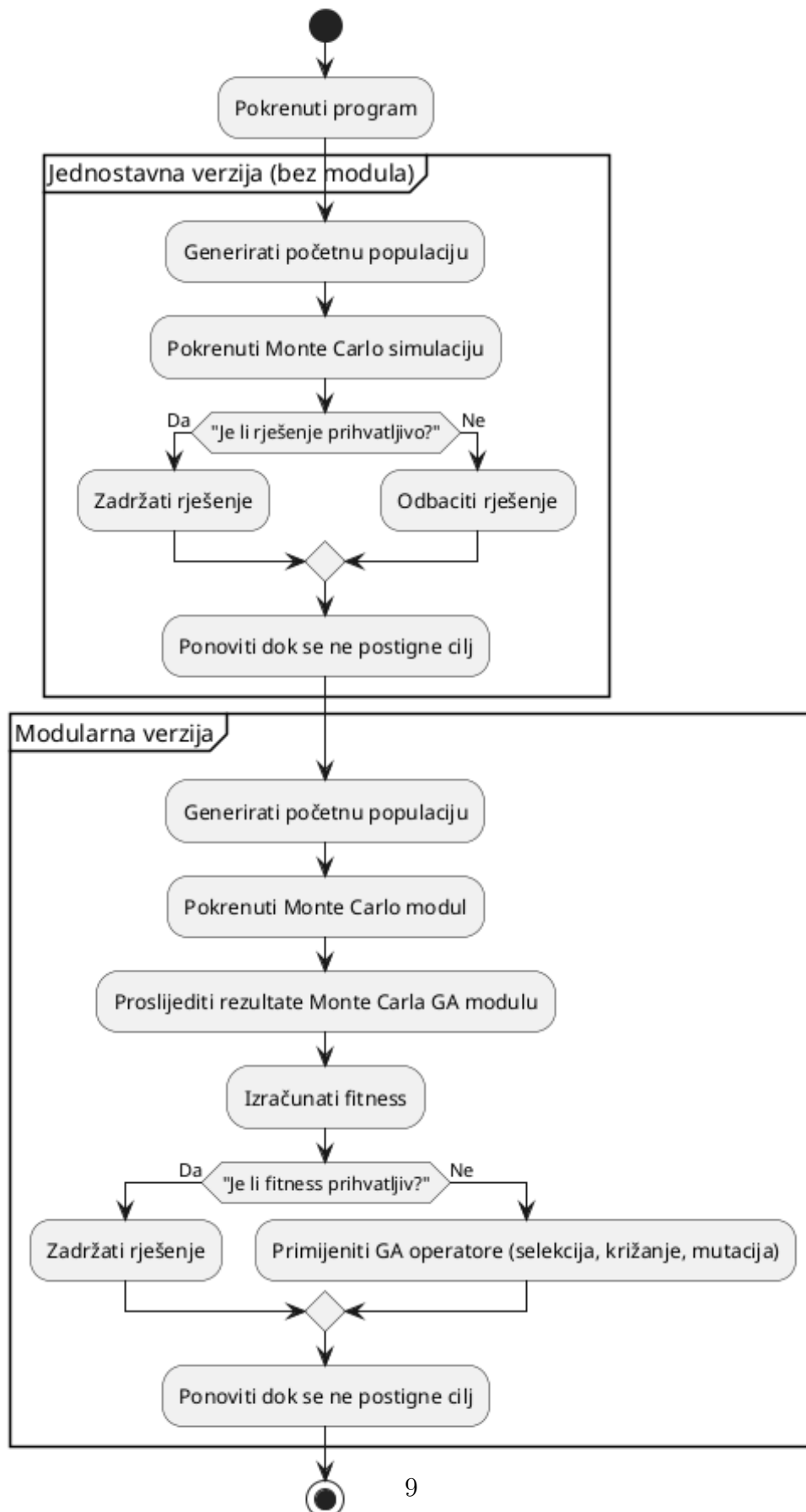
- **NumPy** – osnovna biblioteka za rad s višedimenzionalnim poljima i vektorima, pruža optimizirane matematičke funkcije te omogućuje učinkovito generiranje slučajnih brojeva potrebnih za Monte Carlo simulacije.
- **random** – ugrađeni Python modul za generiranje slučajnih vrijednosti, korišten u inicijalnim fazama simulacije i generiranja početnih populacija u genetskom algoritmu.
- **matplotlib** – biblioteka za vizualizaciju podataka, korištena za grafički prikaz distribucija trajanja zadataka, konvergencije genetskog algoritma i usporedbu dobivenih rješenja.
- **DEAP** (Distributed Evolutionary Algorithms in Python) – opcionalna biblioteka za implementaciju genetskih algoritama i drugih evolucijskih metoda. Omogućuje jednostavnu definiciju operatora selekcije, križanja i mutacije, kao i prilagodbu parametara evolucijskog procesa.

4.2 Struktura implementacije

Implementacija je podijeljena u tri glavna modula:

1. **Monte Carlo modul** – odgovoran za generiranje distribucija trajanja zadataka i procjenu nesigurnosti.
2. **Genetski algoritam (GA) modul** – provodi optimizaciju raspodjele projektnih aktivnosti koristeći rezultate Monte Carlo simulacije.
3. **Modul za vizualizaciju** – omogućuje analizu rezultata kroz grafičke prikaze i usporedbe.

Usporedba jednostavne i modularne verzije programa



Slika 2: Dijagram toka procesa optimizacije korištenjem Monte Carlo simulacije i genetskog algoritma

4.2.1 Monte Carlo modul

Monte Carlo simulacija implementirana je korištenjem biblioteka NumPy i random. Za svaki zadatak definirana je procijenjena vrijednost trajanja (optimistična, pesimistična i najvjerojatnija procjena). Generiranjem velikog broja simulacija (npr. 10^4 iteracija), dobivena je empirijska distribucija trajanja svakog zadatka.

Pseudokod implementacije Monte Carlo modula:

```
for svaki zadatak u projektu:
    generiraj nasumične vrijednosti trajanja prema odabranoj distribuciji
    spremi rezultate u niz
procijeni prosjek, medijan i standardnu devijaciju trajanja
```

Rezultat Monte Carlo simulacije je skup distribucija koji se koristi u evaluacijskoj funkciji genetskog algoritma.

4.2.2 GA modul

Genetski algoritam implementiran je pomoću biblioteke DEAP, a u slučaju jednostavnije implementacije korišteni su osnovni Python moduli random i NumPy. Ključne komponente GA modula su:

- **Inicijalizacija populacije** – generira se početna populacija rješenja koja predstavlja različite načine raspodjele projektnih aktivnosti.
- **Funkcija prilagodbe (fitness)** – izračunava kvalitetu rješenja na temelju trajanja, troškova i vrijednosti dobivenih iz Monte Carlo simulacije.
- **Selekcija** – odabire najbolja rješenja prema definiranim kriterijima (npr. turnirska selekcija).
- **Križanje (crossover)** – kombinira dijelove rješenja dvaju roditelja kako bi se stvorila nova, potencijalno bolja rješenja.
- **Mutacija** – nasumično mijenja dio rješenja radi očuvanja raznolikosti populacije.
- **Zaustavni uvjet** – algoritam se zaustavlja nakon određenog broja generacija ili kada se konvergencija dostigne.

4.2.3 Vizualizacija rezultata

Za analizu rezultata korištena je biblioteka matplotlib, a rezultati se prikazuju u tri glavne forme:

1. **Distribucije trajanja zadataka** – histogrami i krivulje gustoće dobivene iz Monte Carlo simulacija.
2. **Konvergencija genetskog algoritma** – graf promjene najbolje fitness vrijednosti kroz generacije.

3. **Usporedba rješenja** – prikaz najboljeg pronađenog rješenja naspram prosječnih i početnih rješenja.

Ova struktura implementacije omogućuje fleksibilnu nadogradnju sustava, integraciju dodatnih optimizacijskih metoda i prilagodbu različitim vrstama projekata.

5 Eksperimenti

Cilj ovog poglavlja je provesti eksperimentalnu evaluaciju razvijenog modela optimizacije raspodjele projektnih aktivnosti koji kombinira genetske algoritme (GA) i Monte Carlo simulaciju (MC). U ovoj fazi rada eksperimenti će biti okvirno opisani, dok će se stvarni rezultati i detaljne analize dodati nakon provedbe testiranja.

5.1 Testni podaci

Za potrebe eksperimenta koristit će se sintetički skupovi podataka koji će uključivati:

- različite vrijednosti projektnih aktivnosti,
- različite troškove resursa,
- varijabilna trajanja aktivnosti (optimistično, realno, pesimistično).

Takav pristup omogućava kontrolu nad parametrima te usporedivost rezultata između različitih scenarija.

5.2 Scenariji testiranja

Eksperimenti će biti podijeljeni u tri osnovna scenarija:

1. **Samo GA** — Genetski algoritam optimizira raspodjelu bez uključivanja Monte Carlo simulacije (baseline pristup).
2. **Samo MC** — Monte Carlo simulacija koristi se za procjenu trajanja i uspješnosti bez GA optimizacije.
3. **Kombinacija GA + MC** — Predloženi hibridni pristup gdje se GA koristi za optimizaciju, a MC za evaluaciju potencijalnih rješenja (finalni model).

5.3 Metodologija

Za svaki scenarij provodit će se više ponavljanja kako bi se smanjio utjecaj slučajnih varijacija. Planirane metrike koje će se prikupljati:

- **Broj uspješnih projekata** — koliko projekata je završilo unutar planiranog roka i budžeta.
- **Ukupna ROI vrijednost** — ukupni povrat investicije iz optimizirane raspodjele.
- **Stabilnost rješenja** — varijacija rezultata kroz ponavljanja eksperimenta.

5.4 Plan prezentacije rezultata

Nakon provedbe eksperimenata, rezultati će biti prikazani:

- tablicama (kvantitativni rezultati i usporedbe),
- grafovima (vizualizacija trendova i distribucija),
- opisnim analizama (interpretacija dobivenih rezultata).

Sljedeći grafikon prikazat će usporedbu prosječne uspješnosti između triju scenarija:



Slika 3: Primjer usporedbe prosječne uspješnosti za tri scenarija (primjer, podaci privremeni).

6 Zaključak

U ovom diplomskom radu predstavili smo kompleksan pristup optimizaciji raspodjele projektnih aktivnosti koristeći kombinaciju genetskih algoritama i Monte Carlo simulacije. Cilj je bio razviti model koji uzima u obzir nesigurnost u trajanju, troškovima i vrijednosti zadataka, te u okviru zadanih ograničenja vremena, budžeta i raspoloživih resursa maksimizira ukupnu vrijednost projekta.

Kroz detaljnu analizu problematike i pregled postojeće literature, identificirali smo ključne izazove u upravljanju projektima, posebice u segmentu neizvjesnosti i složenosti optimizacije. Implementacijom metaheurističkih metoda, u ovom slučaju genetskih algoritama [6, 17], omogućili smo efikasno pretraživanje velikog prostora rješenja, dok je Monte Carlo simulacija služila za kvantitativno modeliranje rizika i nesigurnosti [18, 19], dajući time realističniju procjenu performansi optimizacijskog rješenja.

Praktična implementacija rezultirala je modelom koji omogućuje donošenje informiranih odluka u planiranju i upravljanju projektima, pružajući projektnim menadžerima alate za bolje usklađivanje ciljeva i ograničenja. Pokazalo se da je kombinacija ovih metoda učinkovita u pronalasku balansiranih rješenja koja maksimiziraju povrat ulaganja, uz minimizaciju rizika od prekoračenja budžeta ili rokova.

Iako su postignuti rezultati zadovoljavajući, postoje brojna područja za buduća istraživanja i unaprjeđenja, među kojima izdvajamo:

- Proširenje modela na dinamičke uvjete projekata koji se mijenjaju tijekom vremena, uključujući nepredvidive vanjske utjecaje.
- Integracija dodatnih metaheurističkih i hibridnih algoritama, poput algoritama rojčaste inteligencije ili simuliranog kaljenja, radi poboljšanja kvalitete rješenja [20].
- Primjena tehnika strojnog učenja za preciznije predviđanje distribucija nesigurnosti i automatsku adaptaciju parametara optimizacije.
- Razvoj softverskih alata s intuitivnim korisničkim sučeljem za praktičnu primjenu predloženih metoda u realnim projektnim okruženjima.

Zaključno, ovaj rad potvrđuje važnost primjene naprednih algoritamskih rješenja u upravljanju projektima, posebno u uvjetima nesigurnosti, te doprinosi boljem razumijevanju i praktičnoj primjeni optimizacijskih i simulacijskih metoda u području projektne ekonomike i menadžmenta. Kao što ističe Kerzner [1], učinkovito upravljanje projektima u suvremenom okruženju zahtijeva kombinaciju tradicionalnih i naprednih pristupa, a naš model pruža značajan doprinos u tom smjeru.

Literatura

- [1] Harold Kerzner. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. Wiley, 12th edition, 2017.
- [2] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [3] Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack Problems*. Springer, 2004.
- [4] D. G. Malcolm, J. H. Roseboom, C. E. Clark, and W. Fazar. Application of a technique for research and development program evaluation. *Operations Research*, 7(5):646–669, 1959.
- [5] John H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, 1st edition, 1975.
- [6] David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- [7] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1949.
- [8] Reuven Y. Rubinstein and Dirk P. Kroese. *Simulation and the Monte Carlo Method*. Wiley, 3rd edition, 2016.
- [9] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [10] Wei Zhang and Ming Li. An efficient algorithm for project scheduling under uncertainty. *International Journal of Project Management*, 29(5):567–574, 2011.
- [11] Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. PMI, 7th edition, 2021.
- [12] David Hillson. *Managing Risk in Projects*. Routledge, 2009.
- [13] Pete Smith. *Project Uncertainty: Managing Risk in New Ventures*. Routledge, 2014.
- [14] David Vose. *Risk Analysis: A Quantitative Guide*. John Wiley & Sons, 2008.
- [15] Fred Glover. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5):533–549, 1986.
- [16] Averill M. Law. *Simulation Modeling and Analysis*. McGraw-Hill Education, 5th edition, 2015.

- [17] Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, 1998.
- [18] Roger Miller and Donald Lessard. Project risk management using monte carlo simulation. *Project Management Journal*, 40(3):23–33, 2009.
- [19] Gordana Avlijaš, Krzysztof Grabiński, and Dragana Milinković. Using monte carlo simulation in project management. *Management - Journal of Sustainable Business and Management Solutions in Emerging Economies*, 13(47):33–38, 2008.
- [20] Amir H. Gandomi, Xin-She Yang, Siamak Talatahari, and Amir H. Alavi. Metaheuristic algorithms in modeling and optimization. *Applied Soft Computing*, 13(1):1–11, 2013.

Popis slika

1	Model problema optimizacije raspodjele projektnih aktivnosti . . .	7
2	Dijagram toka procesa optimizacije korištenjem Monte Carlo simulacije i genetskog algoritma	9
3	Primjer usporedbe prosječne uspješnosti za tri scenarija (primjer, podaci privremeni).	13

Popis tablica