

Diskretni Time-Cost Trade-Off problem

Ana Marković
mi16127@alas.math.rs

Matematički fakultet, Univerzitet u Beogradu
<https://github.com/erlierine/Minimum-time-cost-tradeoff>

24. septembar 2022.

Sadržaj

- 1 Opis problema
- 2 Algoritam grube sile
- 3 S metaheuristike
- 4 Genetski Algoritam

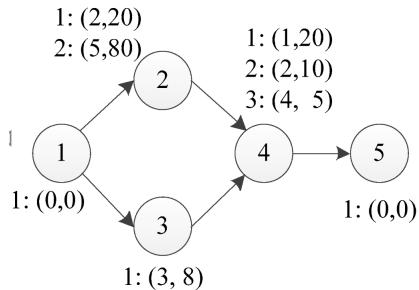
Opis problema

- Dat je projekat P , budžet B za izvršavanje projekta i skup zadataka J u okviru projekta P takav da je nad njim zadato parcijalno uređenje $(J, <)$.
- Ako u parcijalnom uređenju važi za zadatke j i k da je $j < k$ onda to znači da zadatak j mora da se izvrši pre zadatka k .
- Razmatramo diskretnu varijantu problema.
- Za svaki od datih zadataka dat je i konačan broj kombinacija (*vreme izvršavanja, cena*) tako da što je duže vreme izvršavanja zadatka cena je manja.

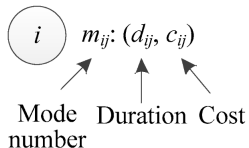
Opis problema

- Zavisnost zadataka je predstavljena acikličnim usmerenim grafom, tako da postoji grana od zadataka j do zadatka k ako u parcijalnom uređenju j mora da se izvrši pre zadatka k .
- Cilj nam je da pronađemo konfiguraciju izvršavanja koja će za dati budžet B da minimizuje vreme izvršavanja projekta.

Opis problema



Legned:

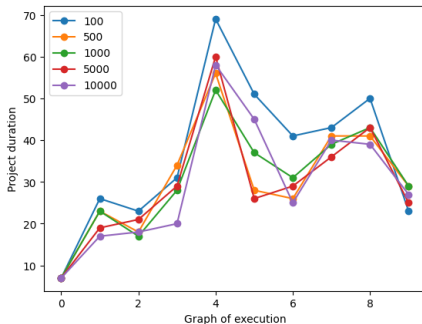


Algoritam grube sile

- Pretražuje sve kombinacije (*vreme izvršavanja, cena*) tako da se minimizuje vreme izvršavanja projekta za dati budžet.
- Pretragom svih kombinacija dobijamo eksponencijalno vreme izvršavanja.
- Ukoliko uspe da se završi algoritam rešenje će biti egzaktno.

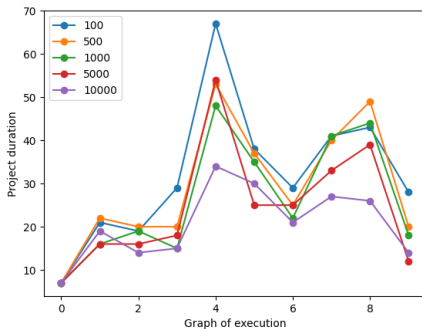
Lokalna pretraga

- Radi optimizaciju jednog slučajno generisanog rešenja.
- Mala promena u rešenju se vrši promenom vremena izvršavanja za **jedan** zadatak.
- Uslov zaustavljanja je broj iteracija.



Metoda promenljivih okolina

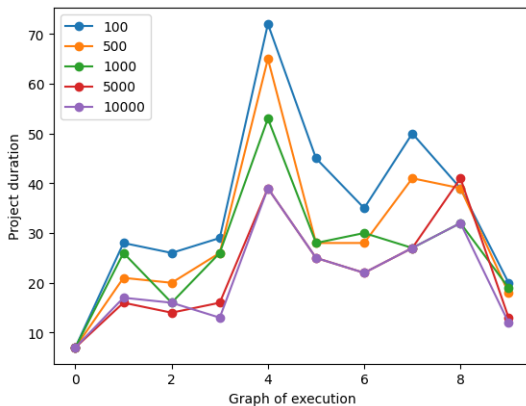
- Za razliku od lokalne pretrage mala promena u rešenju se vrši promenom vremena izvršavanja za **više** zadatak.
- Uslov zaustavljanja je broj iteracija.



Simulirano kaljenje

- Susedna rešenja se pretražuju na sličan način kao kod lokalne pretrage s tim što postoji verovatnoća da uzmemo rešenje koje je lošije od trenutnog radi bolje pretrage prostora rešenja.
- Stepem prihvatanja lošijeg rešenja **p** zavisi od broja iteracija **i**, uzeto je $p = 1/\sqrt{i}$ tako da će lošije rešenje biti prihvaćeno ređe kako izvršavanje odmiče.

Simulirano kaljenje



Genetski Algoritam

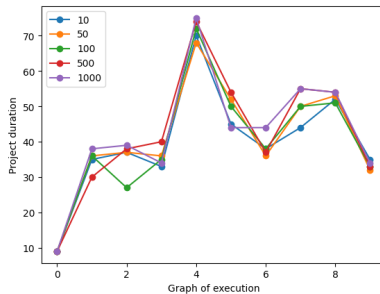
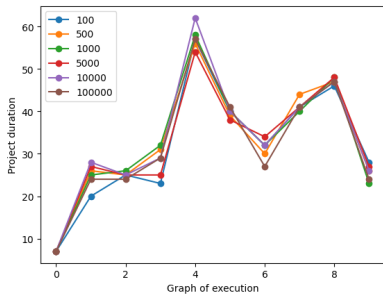
- Za razliku od prethodnih algoritama ovaj pripada grupi **p metaheuristika**, dodatno pripada grupi evolutivnih algoritama.
- Preživljavanje najboljih jedinki u populaciji.
- Jedinka predstavlja jedno rešenje iz prostora dopustivih rešenja, sadrži hromozom koji kodira to rešenje i na osnovu kog određujemo prilagođenost jedinke.
- Jedno kodiranje rešenja:

1	2	1	2	1
---	---	---	---	---

Genetski Algoritam

- U konkretnoj implementaciji:
 - za operator selekcije izabrana je **turnirska selekcija**
 - za operator ukrštanja **jednopoloziciono ukrštanje**
 - **mutacija** je vršena sa malom verovatnoćom na svakom genu, promenom vremena izvršavanja za slučajno odabran zadatak
 - **mutacija**
 - korišćen je **elitistički** pristup, tako da se u svaku sledeću generaciju prenose najbolje jedinke
- Implementiran je i algoritam kombinacije genetskog algoritma i simuliranog kaljenja tako da u svakoj generaciji pokušavamo da popravimo rešenje primenom simuliranog kaljenja na najbolju jedinku.

Genetski Algoritam



HVALA NA PAŽNJI!