

Genetski algoritam za rešavanje problema optimalnog planiranja kretanja u grafu

Petrović Ana, Spasojević Đorđe

Matematički fakultet

pana.petrovic@gmail.com, djordje.spasojevic1996@gmail.com

5. septembar 2019

Pregled

1 Opis problema

2 Algoritam

- Reprezentacija jedinke
- Funkcija prilagođenosti
- Selekcija
- Kreiranje nove generacije
 - Ukrštanje
 - Mutacija

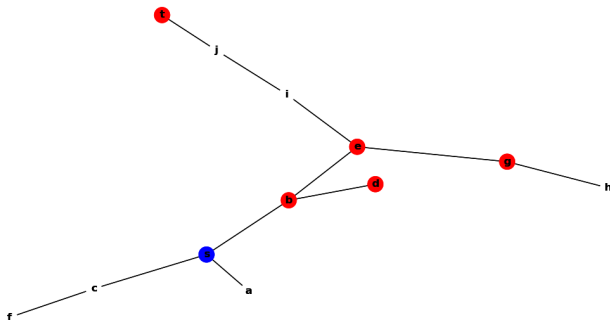
3 Rezultati

- Poređenje sa BF pristupom
- Histogrami

4 Zaključak

5 Literatura

Opis problema



- **Cilj:** Naći sekvencu poteza **robot** i **prepreka** tako da robot stigne iz početnog čvora **s** u ciljni čvor **t** u što manjem broju poteza.
- U jednom potezu može da se pomeri ili robot ili prepreka u susedni čvor koji nije zauzet

Genetski algoritam

- Ulaz: Prepreke, robot, graf, čvor t , putanja(s , t)
- Izlaz: Sekvenca poteza koja dovodi robota u čvor t

Inicijalizacija populacije;

Računanje fitnes funkcije za svaku jedinku

while *Nije dostignut maksimalan broj generacija* **do**

 Selekcija;

 Generisanje nove populacije primenom ukrštanja i mutacije

 Pronalaženje najbolje prilagođene jedinke u populaciji

end

return Trenutno najbolje prilagođena jedinka

Reprezentacija jedinke

- **Lista validnih poteza robota i prepreka počevši od inicijalnog stanja grafa**

- Jedan potez:

- (čvor A, čvor B, težina putanje(A, B), putanja(A, B))

- Jedinke mogu biti različite dužine

- Maksimalna dužina jedinke:

$$\text{brojPreprekaNaPutanji}(s,t) * \text{dužinaPutanje}(s,t)$$

- Primer jedinke dužine 3:

$$[('t', 'i', 2, ['t', 'j', 'i']), ('g', 'h', 1, ['g', 'h']), ('s', 'a', 1, ['s', 'a'])]$$

Funkcija prilagođenosti

- Svakoj jedinki dodeljuje se **skor**
- Određen eksperimentalno, dodeljivanjem nagrada i kazni

Skor = -težinaPutanje;

if *Robot* == *ciljniČvor* **then**

 | Skor += SOLVED_AWARD

end

for *prepeka iz liste prepeka* **do**

 | Skor -= OBSTACLE_PENALTY

end

if *ciljniČvor* *prazan* **and** *brojPreprekaNaPutu* == 0 **then**

 | Skor += DIST_FACTOR*(težinaPutanje(s, t) - težinaPutanje(r,t))

end

...

...

return Skor

Selekcija

- **Turnirska selekcija**

- Bira se k nasumičnih jedinki, od kojih se bira najbolje prilagođena za pobednika
- Lista selektovanih jedinki se popunjava do neke određene veličine *reproduction_size*
- Korišćena je strategija elitizma, pa u turnire ulaze samo one jedinke koje ne pripadaju elitnim
- Na selektovane jedinke se kasnije primenjuje operator ukrštanja

Kreiranje nove generacije

```
Popunjavanje nove generacije elitnim jedinkama;  
Odabiranje nasumične jedinke iz elitnih za mutaciju  
while Veličina nove generacije manja od population_size do  
    | Odabiranje 2 nasumične jedinke iz liste selektovanih  
    | Ukrštanje roditelja  
    | Primena mutacije na nasumičnu elitnu jedinku  
    | Primena mutacije na dete dobijeno ukrštanjem  
end  
return Nova generacija
```

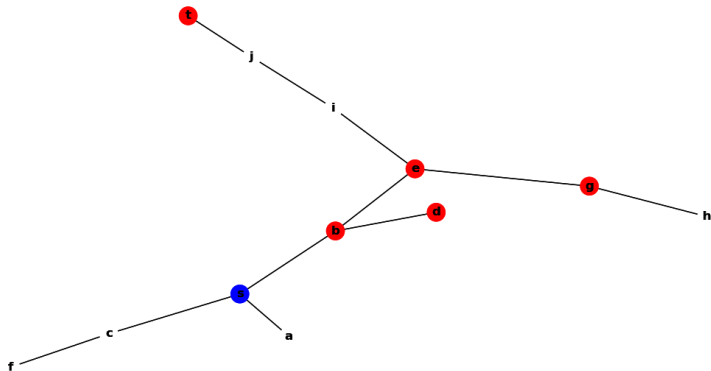

Ukrštanje

- Odabere se nasumično pozicija i iz roditelja koji je kraće dužine
- Taj roditelj se iseče do te pozicije i taj deo se dodeli detetu kao početak
- Za svaku poziciju od i -te pa do dužine drugog roditelja se proverava da li je moguć potez na i -toj poziciji u drugom roditelju, i ako jeste, izvršava se
- Dete se popunjava dokle god je moguće
- Moguće dobijanje još nasumičnijih poteza u detetu
- Moguće dobijanje deteta kraće dužine

Mutacija

- **Dodavanje jednog nasumičnog legitimnog poteza na trenutnu sekvencu**
- Cilj:
 - Produžavanje skraćenih jedinki
 - Rešavanje slučajeva zaglavljivanja robota
- Dešava se sa verovatnoćom koja je malo veća od uobičajene
- Primenjuje se i nad elitnim i nad ostalim jedinkama
- Najpre se proverava da li postoji moguć potez koji izvršava robot i ako postoji dodaje se
 - U suprotnom, dodaje se validan potez neke od prepreka

Rezultati



Rezultati

	M = 5%	M = 10%	M = 20%
Generacija	190	89	17
Broj poteza	41	26	40
Vreme izvršavanja	15.704	24.09	23.999

	maxIt = 50	maxIt = 200	maxIt = 500
Generacija	nije rešen	89	73
Broj poteza	nije rešen	26	40
Vreme izvršavanja	8.10	24.09	49.87

	popSize = 50	popSize = 200	popSize = 1000
Generacija	33	4	6
Broj poteza	40	38	20
Vreme izvršavanja	6.18	32.41	93.064

Poređenje sa BF pristupom

● Algoritam grube sile

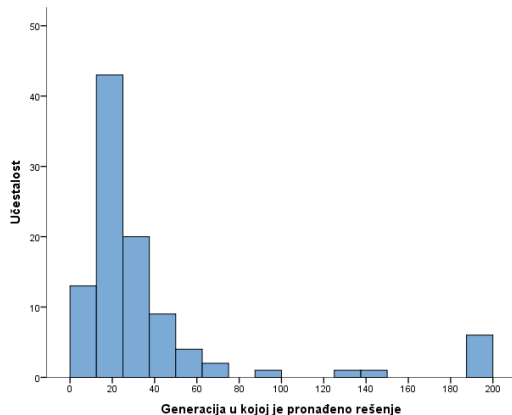
- Pronalazi najoptimalnije rešenje: **18 poteza**
- U ovom slučaju za **0.53 sekunde**
- U slučaju komplikovanijih grafova, značajno duže vreme izvršavanja

● Predloženi genetski algoritam

- popSize = 100, elite = 50, maxIter = 200 repSize = 30, tournamentSize = 10, mutationRate = 20
- Ne pronalazi najbolje rešenje, ali nalazi dovoljno dobro
- Vreme izvršavanja: **1.73 sekunde**
- Generacija u kojoj je pronađeno rešenje: **51**
- Broj poteza u rešenju: **28**

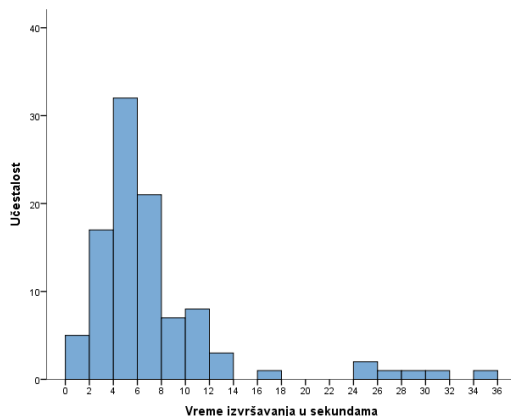
Histogram - broj generacije

- $popSize = 200$, $elite = 50$, $maxIter = 200$, $repSize = 30$, $tournamentSize = 10$, $mutationRate = 20$
- 100 pokretanja programa



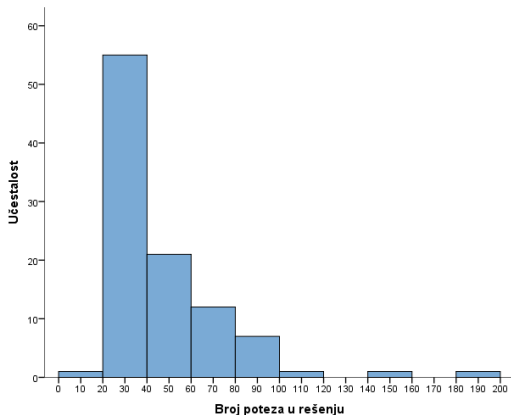
Histogram - vreme izvršavanja

- $popSize = 200$, $elite = 50$, $maxIter = 200$, $repSize = 30$,
 $tournamentSize = 10$, $mutationRate = 20$
- 100 pokretanja programa



Histogram - broj poteza

- *popSize* = 200, *elite* = 50, *maxIter* = 200, *repSize* = 30, *tournamentSize* = 10, *mutationRate* = 20
- 100 pokretanja programa



Zaključak

- Algoritam se pokazao kao relativno uspešan
- U komplikovanijim primerima, značajno unapređenje u vremenu izvršavanja u odnosu na algoritam grube sile
- U većini slučajeva rešava problem u ranijim iteracijama
- Prosečno vreme do pronalaženja dovoljno dobrog rešenja je oko 5 sekundi
- Ne pronalazi optimalan broj poteza, ali najčešće sasvim dovoljno dobar

Literatura

- Ismail AL-Taharwa, Alaa Sheta, and Mohammed Al-Weshah. A mobile robot path planning using genetic algorithm in static environment. 2008.
- Sarah Alnasser and Hachemi Bennaceur. An efficient genetic algorithm for the global robot path planning problem. Digital Information and Communication Technology and its Applications (DICTAP) 2016 Sixth International Conference, 2016.
- Christos H. Papadimitriou, Prabhakar Raghavan, Madhu Sudan, and Hisao Tamaki. Motion planning on a graph. Proc. 35th IEEE Symposium on Foundations of Computer Science (FOCS), 09 2001.
- Lydia E. Kavraki and Steven M. LaValle. Motion Planning. 2008.
- Chaymaa Lamini, Said Benhlilima, and Ali Elbekri. Genetic algorithm based approach for autonomous mobile robot path planning. 2018.