Optimizacija - Minimum Edge Dominating Set

Nikola Stojanović

18. Jun 2024.

Sadržaj

1 Uvod

U ovom radu predstavljena je implementacija i evaluacija genetskog algoritma za rešavanje problema minimalnog skupa dominantnih ivica (MEDS).

Genetski algoritmi (GA) su metaheuristički pristupi inspirisani procesima prirodne selekcije i genetike. Osnovni princip GA je simulacija evolucije populacije rešenja tokom više generacija. Svaka jedinka u populaciji predstavlja potencijalno rešenje problema, a evolucija se odvija kroz operacije selekcije, ukrštanja i mutacije. Cilj je pronaći optimalno ili blizu optimalno rešenje kroz iterativni proces poboljšanja.

U daljem delu rada diskutovaćemo dizajn algoritma, podešavanje parametara pomoću grid pretrage, kao i rezultate dobijene na različitim veličinama grafova u poredjenju sa drugim algoritmima, poput pohlepnog i algoritma grube sile.

2 Opis problema

Problem minimalnog skupa dominantnih ivica (MEDS) u teoriji grafova zahteva pronalaženje najmanjeg mogućeg skupa ivica u grafu takvih da svaka ivica u grafu ili pripada tom skupu ili je susedna nekoj ivici iz tog skupa.
Formalno, dat je neusmeren graf $G = (V, E)$ gde je V skup čvorova, a E skup ivica. Potrebno je naći skup ivica $E' \subseteq E$ takav da za svaku ivicu $e1 \in E - E'$ postoji $e2 \in E$ gde su $e1$ i $e2$ susedne ivice.
Jedan od poznatih rezultata za problem MEDS je da se može aproksimirati faktorom 2 koristeći maksimalno uparivanje (maximal matching). Ako optimalno rešenje za MEDS ima veličinu OPT, tada će veličina skupa dominantnih ivica izvedenog iz maksimalnog uparivanja biti najviše $2 \times OPT$. Ovo svojstvo čini maksimalno uparivanje korisnim alatom za procenu kvaliteta rešenja dobijenih drugim algoritmima.

MEDS je dobro poznat NP težak problem u teoriji grafova, koji ima brojne praktične primene, uključujući optimizaciju mreža, raspodelu resursa, i dizajn komunikacionih sistema.

3 Implementacija algoritama

U ovom delu opisani su algoritmi koji su implementirani za rešavanje problema, uključujući genetski algoritam, algoritam grube sile i pohlepni algoritam.

3.1 Genetski Algoritam

Genetski algoritam za problem MEDS sastoji se od sledećih koraka:

1. **Inicijalizacija:** Generisati početnu populaciju potencijalnih rešenja (individua). U našem slučaju, svaka jedinka predstavlja binarni hromozom gde svaki gen označava da li je odgovarajuća ivica uključena u skup.
2. **Selekcija:** Izabrati jedinke na osnovu njihovih fitnes vrednosti za stvaranje parova.

Korišćena je turnirska selekcija, za velicinu turnira uzeto je 10% populacije. Prilikom selekcije korišćen je i elitizam. Za novu populaciju se odvaja odredjen broj najboljih jedinki iz prethodne populacije, čime se brže teži optimalnom rešenju (intenzifikacija).

3. **Ukrštanje:** Kombinovati parove jedinki za stvaranje potomaka. Korišćeno je jednopoziciono ukrštanje.
4. **Mutacija:** Uvesti nasumične promene za održavanje diverziteta. Pored fiksne stope mutacije implementirana je i postepena mutaciju gde se stopa mutacije linearno smanjuje od početne do krajnje vrednosti tokom generacija. Ovim se omogućuje algoritmu da u početnim generacijama više istražuje (diverzifikacija), a kako generacije prolaze teži ka optimalnom rešenju (takodje intenzifikacija).
5. **Zamena:** Zameniti staru populaciju novom.
6. **Zaustavljanje:** Ponavljati korake 2-5 za fiksni broj generacija.

Implementirane su dve metode izračunavanja fitnesa:

- **Infinity Fitness:** Za validna rešenja, fitnes se računa kao razlika broja svih ivica i broja ivica individue. Za nevalidna rešenja, dodeljuje fitnes vrednost negativne beskonačnosti. Ova fitnes funkcija penalizuje rešenja koja ne pokrivaju sve ivice u grafu, što kažnjava rešenja koja su bliska optimalnom i izbacuje iz evolucije zbog nepokrivenih u najgorem slučaju jedan ili par ivica grafa. Ovaj problem rešavamo seldećom fitnes funkcijom.
- **Kazneni Fitnes:** Računa fitnes funkciju kao razliku pokrivenih ivica, nepokrivenih ivica i broja ivica pojedinca, pri čemu ukoliko nisu pokrivene sve ivice grafa, umesto negativne beskonačnosti fitnes će samo biti umanjen za broj nepokrivenih ivica.

3.2 Algoritam Grube Sile

Algoritam grube sile pretražuje sve moguće podskupove ivica kako bi pronašao minimalni skup dominantnih ivica. Iako garantuje pronalaženje optimalnog rešenja, njegova vremenska složenost je eksponencijalna, što ga čini praktično neupotrbljivim za veće grafove.

3.3 Pohlepni Algoritam

Pohlepni algoritam iterativno bira ivice koje pokrivaju najveći broj nepokrivenih ivica dok svi ne budu pokriveni. Iako ne garantuje optimalno rešenje, pohlepni algoritam je efikasan i često daje dovoljno dobro rešenja za praktične primene.

4 Rezultati

Ispravnost algoritma prvenstveno je testirana na manjim grafovima, kako nasumičnim, tako i na specifičnim grafovima, čije optimalno rešenje možemo izračunati, poput kompletnog, putovnog (linijskog), cikličnog i zvezdanog grafa.

Graph type	Opt. solution	V	E	Brute Force	Greedy	Genetic Algorithm
random	N/A	7	15	2	2	2
random	N/A	9	17	3	4	3
complete	[V/2]	6	15	3	3	3
path	[(V-2)/3] + 1	10	9	3	4	3
cycle	[(V-1)/3] + 1	12	12	4	4	4
star	1	16	15	1	1	1

Tabela 1: Rezultati sa manjim grafovima

Kod genetskog algoritma, korišćeni su kao podrazumevani parametri populacija = 100, broj generacija = 10, fiksna stopa mutacije = 5%, elitizam = 5% i infiniti fitnes funkcija. Možemo videti na ovim grafovima da genetski algoritam pronalazi optimalno rešenje, dok pohlepni algoritam iako veoma efikasan, čak i na malim grafovima se može zaglaviti u lokalnom minimumu, ali dovoljno blisko optimalnom rešenju.

4.1 Grid Pretraga

Za testiranje na velikim grafovima, primenjena je grid pretraga. Cilj grid pretrage je da identifikuje optimalne parametre genetskog algoritma. Za svaki skup parametara, pored izračunatog rešenja, izračunat je i aproksimacioni faktor u poređenju sa maksimalnim uparivanjem, koji nam služi za proveru validnosti rešenja. Pored toga uporedićemo i sa pohlepnim rešenjem.
Korišćeni su nasumični grafovi od redom 10, 20, 30 i 40 čvorova, pritom je na svaki od grafova primenjena grid pretraga za 4 modifikacije genetskog algoritma (kombinacije algoritma sa dve fitnes funkcije i da li je korišćeno linearno smanjenje stope mutacije ili ne).

Na primeru grafa (V:40, E:388) ćemo pokazati primenu grid pretrage. Parametri korišćeni u grid pretrazi su:

- Veličina populacije: 200, 400, 600
- Broj generacija: 50, 100, 200
- Stopa mutacije: 0.01, 0.05 [1]
- Procenat elitizma: 0.05, 0.1

Vrednosti za poredjenje:

- Maksimalno uparivanje: 20
- Pohlepni algoritam: 18

pop.	gener.	mut.	elit.	reš	aproks.
200	50	0.05	0.05	36	0.56
200	50	0.05	0.1	40	0.5
200	100	0.05	0.05	31	0.65
200	100	0.05	0.1	31	0.65
200	200	0.05	0.05	30	0.67
200	200	0.05	0.1	29	0.69
400	50	0.05	0.05	29	0.69
400	50	0.05	0.1	29	0.69
400	100	0.05	0.05	24	0.83
400	100	0.05	0.1	23	0.87
400	200	0.05	0.05	21	0.95
400	200	0.05	0.1	21	0.95
600	50	0.05	0.05	24	0.83
600	50	0.05	0.1	24	0.83
600	100	0.05	0.05	21	0.95
600	100	0.05	0.1	22	0.91
600	200	0.05	0.05	22	0.91
600	200	0.05	0.1	20	1.0

Tabela 2: Sa postepenom mutacijom, Infinity fitnes funkcija.

pop.	gener.	mut.	elit.	reš	aproks.
200	50	0.05	0.05	36	0.56
200	50	0.05	0.1	29	0.69
200	100	0.05	0.05	31	0.65
200	100	0.05	0.1	29	0.69
200	200	0.05	0.05	24	0.83
200	200	0.05	0.1	26	0.77
400	50	0.05	0.05	27	0.74
400	50	0.05	0.1	26	0.77
400	100	0.05	0.05	26	0.77
400	100	0.05	0.1	23	0.87
400	200	0.05	0.05	20	1.0
400	200	0.05	0.1	21	0.95
600	50	0.05	0.05	23	0.87
600	50	0.05	0.1	25	0.8
600	100	0.05	0.05	22	0.91
600	100	0.05	0.1	19	1.05
600	200	0.05	0.05	21	0.95
600	200	0.05	0.1	19	1.05

Tabela 3: Sa postepenom mutacijom, Kazneni fitnes funkcija

[1] za stopu mutacije kod linearnog smanjenja, je prosledjivan jedan parametar, koji predstavlja inicijalnu stopu, dok je finalna u našem slučaju peti deo inicijalne stope.

pop.	gener.	mut.	elit.	reš	aproks.
200	50	0.01	0.05	23	0.87
200	50	0.01	0.1	25	0.8
200	50	0.05	0.05	83	0.24
200	50	0.05	0.1	93	0.22
200	100	0.01	0.05	26	0.77
200	100	0.01	0.1	21	0.95
200	100	0.05	0.05	88	0.23
200	100	0.05	0.1	92	0.22
200	200	0.01	0.05	21	0.95
200	200	0.01	0.1	23	0.87
200	200	0.05	0.05	93	0.22
200	200	0.05	0.1	92	0.22
400	50	0.01	0.05	21	0.95
400	50	0.01	0.1	20	1.0
400	50	0.05	0.05	74	0.27
400	50	0.05	0.1	81	0.25
400	100	0.01	0.05	20	1.0
400	100	0.01	0.1	18	1.11
400	100	0.05	0.05	82	0.24
400	100	0.05	0.1	78	0.26
400	200	0.01	0.05	19	1.05
400	200	0.01	0.1	18	1.11
400	200	0.05	0.05	76	0.26
400	200	0.05	0.1	81	0.25
600	50	0.01	0.05	19	1.05
600	50	0.01	0.1	20	1.0
600	50	0.05	0.05	68	0.29
600	50	0.05	0.1	75	0.27
600	100	0.01	0.05	19	1.05
600	100	0.01	0.1	20	1.0
600	100	0.05	0.05	75	0.27
600	100	0.05	0.1	72	0.28
600	200	0.01	0.05	19	1.05
600	200	0.01	0.1	17	1.18
600	200	0.05	0.05	72	0.28
600	200	0.05	0.1	80	0.25

Tabela 4: Bez postepene mutacije, Infinity fitnes funkcija

pop.	gener.	mut.	elit.	reš	aproks.
200	50	0.01	0.05	23	0.87
200	50	0.01	0.1	24	0.83
200	50	0.05	0.05	93	0.22
200	50	0.05	0.1	95	0.21
200	100	0.01	0.05	22	0.91
200	100	0.01	0.1	24	0.83
200	100	0.05	0.05	94	0.21
200	100	0.05	0.1	94	0.21
200	200	0.01	0.05	25	0.8
200	200	0.01	0.1	21	0.95
200	200	0.05	0.05	91	0.22
200	200	0.05	0.1	85	0.24
400	50	0.01	0.05	20	1.0
400	50	0.01	0.1	20	1.0
400	50	0.05	0.05	76	0.26
400	50	0.05	0.1	81	0.25
400	100	0.01	0.05	20	1.0
400	100	0.01	0.1	21	0.95
400	100	0.05	0.05	77	0.26
400	100	0.05	0.1	81	0.25
400	200	0.01	0.05	19	1.05
400	200	0.01	0.1	18	1.11
400	200	0.05	0.05	75	0.27
400	200	0.05	0.1	77	0.26
600	50	0.01	0.05	20	1.0
600	50	0.01	0.1	19	1.05
600	50	0.05	0.05	73	0.27
600	50	0.05	0.1	75	0.27
600	100	0.01	0.05	19	1.05
600	100	0.01	0.1	20	1.0
600	100	0.05	0.05	66	0.3
600	100	0.05	0.1	71	0.28
600	200	0.01	0.05	19	1.05
600	200	0.01	0.1	19	1.05
600	200	0.05	0.05	77	0.26
600	200	0.05	0.1	68	0.29

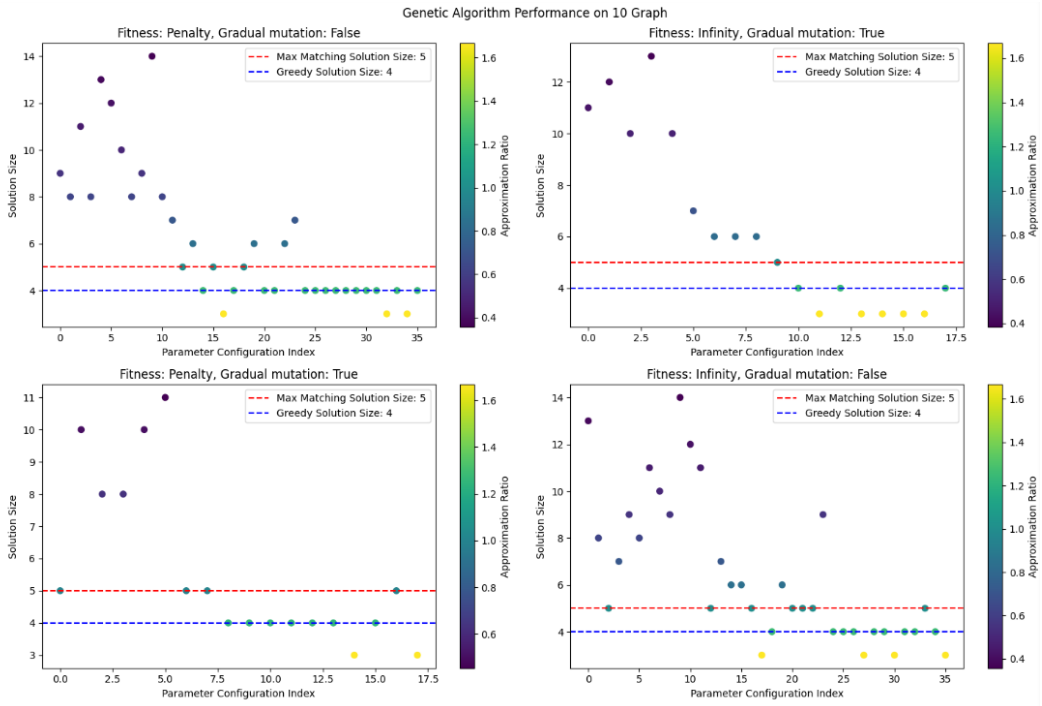
Tabela 5: Bez postepene mutacije, Kazneni fitnes funkcija

4.2 Analiza Rezultata

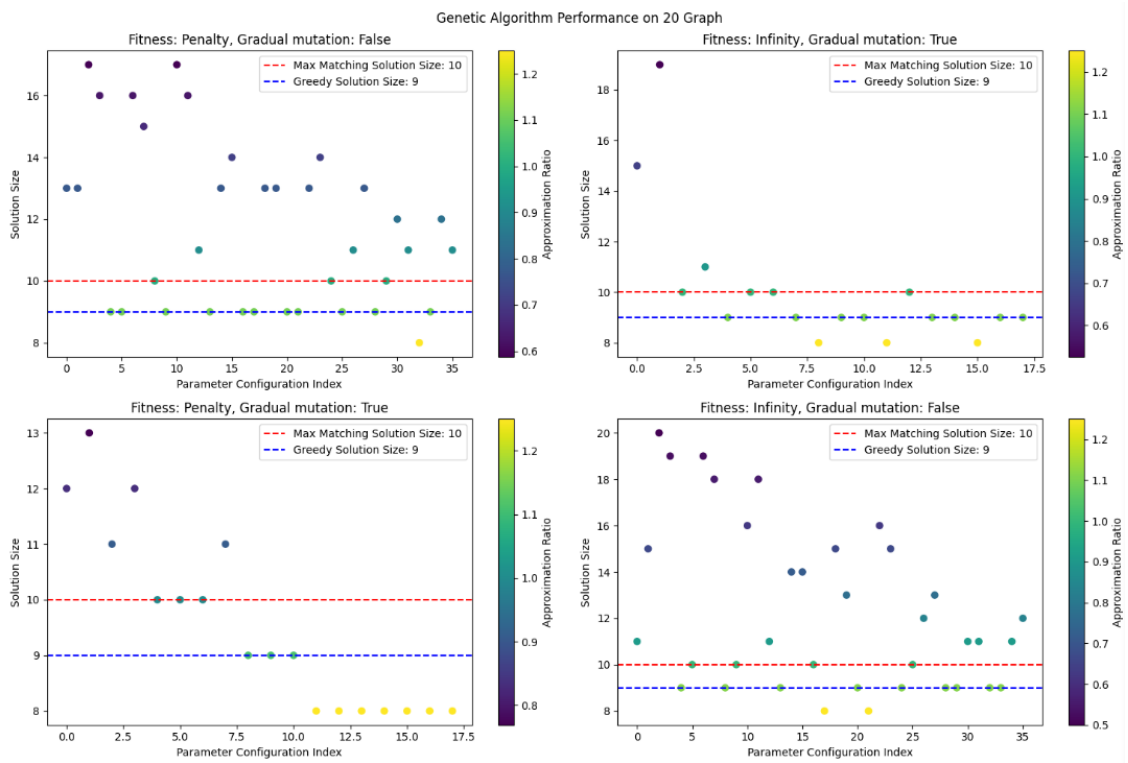
Analiza rezultata pokazuje da:

- Veličina populacije ima značajan uticaj na kvalitet rešenja. Veće populacije omogućavaju algoritmu da istraži veći deo prostora rešenja, što može dovesti do boljih rešenja. Možemo primetiti iz tabela iznad, da za vrednost populacije manjoj od ukupnog broja grana ne uspevamo da dobijemo zadovoljavajuće rešenje.
- Broj generacija takođe utiče na performanse algoritma. Veći broj generacija omogućava algoritmu da konvergira ka boljim rešenjima, ali povećava vreme izvršavanja.
- Stopa mutacije i procenat elitizma moraju biti pažljivo podešeni. Visoka stopa mutacije može ometati konvergenciju, dok nizak procenat elitizma može dovesti do gubitka dobrih rešenja. Možemo videti iz tabela, da za velike vrednosti i populacije i generacija, dobijamo daleko lošija rešenja za veću vrednost stope mutacije, dok elitizam ne utiče u velikoj meri.
- Postepena mutacija (linearno smanjivanje stope mutacije tokom generacija) pomaže u istraživanju prostora rešenja u početnim generacijama, a zatim se koncentriše na intenzifikaciju u kasnijim generacijama.

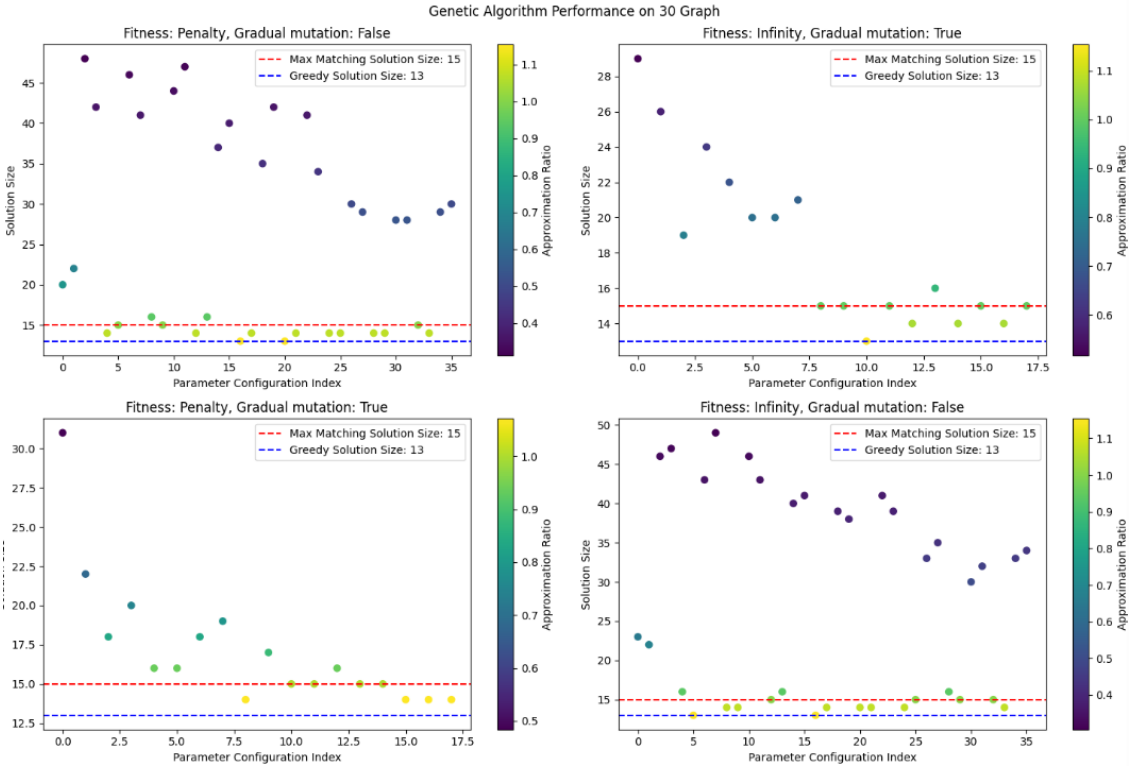
Rezultati ostalih testiranja:



Slika 1: Graf (V:10, E:24)



Slika 2: Graf (V:20, E:90)



Slika 3: Graf (V:30, E:223)

5 Zaključak

Genetski algoritam pokazao se kao efikasan pristup za rešavanje problema minimalnog skupa dominantnih ivica (MEDS), posebno kada se poredi sa algoritmima grube sile i pohlepnim algoritmom. Kroz implementaciju i evaluaciju različitih pristupa, ustanovili smo da genetski algoritam može generisati približno optimalna rešenja čak i za veće grafove gde drugi algoritmi nisu praktični.

Grid pretraga omogućila nam je da identifikujemo ključne parametre koji utiču na performanse genetskog algoritma. Pokazalo se da veličina populacije i broj generacija značajno utiču na kvalitet rešenja, dok pravilno podešavanje stope mutacije i procenata elitizma može dodatno poboljšati rezultate.

Uvođenje linearno smanjujuće stope mutacije doprinosi ravnoteži između diverzifikacije i intenzifikacije tokom evolucije populacije. Poboljšanje samog algoritma se može nastaviti dalje u pravcu hibridizacije genetskog algoritma sa drugim optimizacionim tehnikama i testiranje na još složenijim grafovima.

Genetski algoritmi, sa svojim sposobnostima adaptivnog pretraživanja velikih prostora rešenja, pokazali su se kao vredan alat za optimizacione probleme u teoriji grafova i potencijalno šire.

Literatura

[1] https://en.wikipedia.org/wiki/Edge_dominating_set

[2] https://en.wikipedia.org/wiki/Matching_(graph_theory)

[3] https://www.csc.kth.se/~viggo/wwwcompendium/node13.html#3118p

[4] https://github.com/MATF-RI/Materijali-sa-vezbi