

Travelling salesman problem

Vladimir Jovanovic 42/19 C-smjer

2021

Prikazujem tabelu u kojoj se nalaze podaci o najboljem fitnessu, kao i o prosječnom fitnessu, za svaki crossover i mutaciju koju sam implementirao. Mjerenje sam vršio na datasetu dantzig42_d, koji sadrži matricu 42x42. Svaka kombinacija algoritama je testirana 5 puta po 1000 iteracija, na populaciji veličine 168. Najmanji fitness seta je 699.

Svi korišteni algoritmi se zasnivaju na "path" reprezentaciji rute, tojest svaka ruta je torka oblika $(1, x, \dots, x)$. Svaka ruta kreće iz početnog grada 1.

PMX - partially mapped crossover. Nasumično se biraju dvije pozicije na rutama parenta. Segment između te dvije tačke u prvom parentu se upisuje u drugog childa, a drugog roditelja u prvi child. Zatim se posmatraju dva segmenta iz oba roditelja, i vrši se mapiranje, tako što se mapiraju međusobno čvorovi iz oba roditelja na istim pozicijama u segmentu. Zatim se vrši upisivanje ostalih čvorova u prvi child na sledeći način: kopiraju se čvorovi iz prvog roditelja ako se oni ne nalaze u segmentu između dvije pozicije, a ako se nalaze, onda se posmatra na koji čvor drugog parenta su oni mapirani, i taj čvor se upisuje na mesto gdje bi išao čvor koji je u segmentu. Analogno za drugi child.

OX1 - order crossover. Na sličan način kao PMX se bira podruta glavne rute, tako što se nasumično odaberu dvije pozicije na ruti i uzima podruta između njih.. Potomci se stvaraju tako što se prekopira podruta jednog roditelja, a čuva se relativni raspored čvorova drugog roditelja, tako što se, počevši od druge nasumično odabrane pozicije, kopiraju čvorovi drugog roditelja, a kada se dođe do kraja rute nastavlja se sa početka tog istog roditelja.

OX2 - order based crossover. Nasumično se bira određeni broj pozicija u ruti drugog roditelja. Jedan potomak se dobija tako što se kopira sadržaj prvog roditelja, osim na pozicijama koje se nalaze na pozicijama koje su nasumično odabrane. (npr. ako je odabrana pozicija 2 drugog roditelja, i na toj poziciji se nalazi grad 4, onda se ne kopira čvor na poziciji 4 u prvom roditelju). Gradovi koji nedostaju se dodaju po redu po kom se pojavljuju u drugom roditelju. Analogno za drugog potomka.

POS - position based crossover. Takođe se nasumično bira određeni broj pozicija u ruti drugog roditelja, ali ovaj crossover postavlja čvorove na tim pozicijama u potomka, a ostatak čvorova se kopira iz prvog roditelja po redu po kom se javljaju. Analogno za drugog potomka.

EM - exchange mutation. Nasumično bira dva grada na ruti, i mijenja im mjesta.

DM - displacement mutation. Nasumično se biraju dvije tačke na ruti. Zatim se podruta između te dvije tačke uklanja iz rute, i dodaje se na nasumično odabranom mjestu.

IM - insertion mutation. Nasumično se bira jedan grad na ruti, izbacuje iz rute, i dodaje nazad na nasumično odabranu poziciju.

SIM - simple inversion mutation. Nasumično se biraju dvije tačke na ruti. Zatim se redosljed gradova podrute između te dvije tačke invertuje.

IVM - inversion mutation. Nasumično se biraju dvije tačke. Zatim se podruta između te dvije tačke uklanja iz rute, invertuje, i vraća na nasumično odabranu poziciju.

	PMX	OX1	OX2	POS
EM best	757	782	699	731
EM average	789	799	732	782
DM best	731	731	732	735
DM average	748	776	760	762
IM best	719	716	726	713
IM average	736	783	761	762
SIM best	704	764	727	704
SIM average	737	788	737	730
IVM best	708	797	733	709
IVM average	732	810	765	723
Best	704	716	699	704
Average	748	791	751	751

Sledeća tabela je nastala na sličan način kao i prva, uz hibridizaciju sa 2-opt algoritmom nakon svake 100-te generacije. Svaki algoritam je pušten na 500 iteracija, takođe po 5 puta.

	PMX	OX1	OX2	POS
EM best	709	707	699	712
EM average	718	710	710	725
DM best	709	708	699	699
DM average	715	721	714	713
IM best	699	707	704	705
IM average	708	716	708	710
SIM best	699	714	699	704
SIM average	705	715	723	715
IVM best	704	708	704	699
IVM average	708	742	715	712
Best	699	716	699	699
Average	711	721	714	715

NAPOMENA: u obje tabele, Best se odnosi na najbolji fitness po algoritmu za svaku mutaciju, a Average se odnosi na prosječnu distancu koju dati algoritam

izračuna. Najbolji rezultati po redovima (mutacijama) su boldovani.