

Heurističke metode optimizacija

Projekt: Parkiranje vozila u spremi štu

IZVJEŠTAJ

2018./19.

Juraj Pejnović

Andrej Jertec

Opis problema

Problem projektnog zadatka je parkiranje vozila u spremištima kompanije za javni prijevoz. Potrebno je parkirati konačni broj vozila u konačni broj traka ispunjavajući određena zadana ograničenja koji problem približuju realnim uvjetima. Za svako vozilo potrebno je pronaći jedinstvenu poziciju u jednoj od traka. Traka treba ispunjavati ograničenja vezana za specifičnosti vozila. Dodatno, vozila po duljini trebaju stati u traku te biti iste serije vozila. Raspored u traci određen je vremenom polaska vozila te se vozila sa manjim vremenom polaska postavljaju ispred drugih vozila. Traka može biti blokirana od drugih traka ili i sama blokirati pojedine trake. U tom slučaju vremena polaska vozila u blokiranim trakama mora biti nakon vremena polaska vozila blokirajućih traka. Vozila uz seriju imaju i specifičan raspored koji nije ograničenje u zadanome problemu, već se koristi kao jedan od parametara cilja.

Problem ima dva globalna cilj. Prvi je minimizacija broja različitih serija vozila u susjednim trakama, broja korištenih traka i neiskorišteni prostor u korištenim trakama. Drugi globalni cilj je maksimizacija broja vozila s istim tipom rasporeda unutar trake, broja vozila s istim tipom rasporeda u susjednim trakama i optimizirati vremenski razmak između izlaska vozila iz trake spremišta.

Opis primijenjenog algoritma

Pokrivanje prostora rješenja postignuto je formiranjem stabla čiji čvorovi predstavljaju stanja traka sa vozilima u nekom datom trenutku. Vozila su na početku sortirana po vremenima polaska i tim redoslijedom ih se i evaluira. Koristi se python list za spremanje sljedećih čvorova za obilazak i to kao stog. Stablo retražuje depth first bez primjene nekakve kvalitetnije heuristike jer je prostor pretraživanja golem i iziskivao bi stalno i zahtjevno sortiranje čvorova u listi. Probali smo greedy algoritam prema funkcijama evaluacije koje su zadane u uputama, ali takav algoritam nam nije davao nikakva rješenja.

Prikaz rješenje je istovjetno formatu rješenja za predaju. Vektor čiji indeks elementa predstavlja traku ispunjen sa vektorom koji sadrži indekse vozila koja se nalaze u toj traci.

Kao funkcije evaluacije rješenja korištene su zadane globalne funkcije cilja, te je za samu evaluaciju korišten njihov omjer tj. omjer funkcije maksimizacije i funkcije minimizacije. Problem optimizacije u konačnici je bio maksimizacija dobivenog omjera.

Pseudokod primijenjenog algoritma

dok stog nije prazan:

uzmi gornji čvor

ako je vozilo u trenutnom čvoru zadnje vozilo po redu:

dodaj u listu mogućih rješenja

nastavi provjeru za sljedeći čvor iz stoga

uzmi sljedeće vozilo po redu

za sve linije:

ako (legalno postaviti vozilo u liniju):

dodaj trenutni čvor proširen za sljedeće vozilo u provjerenom redu u
stog

sortiraj sva pronađena rješenja po funkcijama za evaluaciju iz uputa, func2/func1

uzmi najvrjednije rješenje

Opis rezultata i diskusija

Uspjeli smo dobiti rezultate samo za instancu 1. Ostale instance su imale previše automobila i nismo uspjeli pronaći rješenje. Probali smo maknuti uvjete za blokirane trake ali nismo uspjeli pronaći rezultat u razumnim vremenima.

Rješenja za instancu 1 sa računanjem od 5 minuta i beskonačnim računanjem (u razumnom vremenu) su jednaka što znači da smo ili našli globalni maksimum ili smo prekratko vrtili algoritam tijekom beskonačnog računanja.

Broj pronađenih rješenja u instanci 1 za 1 minutu je bio oko 19 000, za 5 minuta je bio 97000 a za beskonačno smo stali na 256000 rješenja.

Instanca 3 je dolazila do 49 složenih vozila što je valjda maksimum do kojega se može doći sa DFS-om.

Kratki zaključak

Iako je DFS veoma brz i daje dobre rezultate kada je prostor pretrage relativno malen, ako se prostor pretrage poveća preko određene dimenzije DFS neće biti dovoljno dobar da pronađe rezultat u razumnom vremenu. Korištenjem neke druge heuristike bi se vjerojatno postigao bolji rezultat.