

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA**

# **Heurističke metode optimizacije**

Problem usmjeravanja školskih autobusa

Domagoj Krivošić

Luka Novak

**Zagreb, 2018.**

# SADRŽAJ

<b>OPIS PROBLEMA</b>	<b>3</b>
<b>OGRANIČENJA I PRETPOSTAVKE</b>	<b>3</b>
<b>PODACI</b>	<b>5</b>
ULAZNI PODACI	5
FORMAT RIJEŠENJA	5
<b>ALGORITAM</b>	<b>6</b>
OPIS KORIŠTENOG ALGORITMA	6
HIPERMUTACIJA	8
HIPERPARAMETRI	8
<b>IMPLEMENTACIJA</b>	<b>9</b>
<b>REZULTATI</b>	<b>10</b>
<b>ZAKLJUČAK</b>	<b>10</b>

## OPIS PROBLEMA

Jedan od tipičnih problema usmjeravanja jest usmjeravanje vozila. U tom problemu postaje su poznate, treba samo odrediti puteve. Problem usmjeravanja autobusa je podproblem usmjeravanja vozila koji je nešto složeniji. Dakle, problem usmjeravanja autobusa je složeniji iz nekoliko razloga. Prvi od njih je taj da nisu poznate postaje, nego samo *potencijalne* postaje. Drugi razlog je taj da odabir postaja i oblikovanje autobusnih linija ovisi o kapacitetu autobusa koji vozi na toj liniji te mjestu stanovanja učenika.

Krajnji rezultat rješavanja ovog problema je:

- Odrediti skup stanica
- Odrediti stanicu za svakog učenika
- Odrediti autobusne linije tako da ukupna udaljenost koju autobusi prelaze budu što manje moguće

## OGRANIČENJA I PRETPOSTAVKE

Prvo od ograničenja odnosi se na učenike. Svakom učeniku mora biti dodijeljena jedna stanica, ali udaljenost te stanice od učenikove kuće mora biti manja od zadane maksimalne dozvoljene udaljenosti.

Drugo ograničenje odnosi se na autobuse i kaže da svaki autobus mora imati jednaki kapacitet te da se taj kapacitet ni u jednom slučaju ne smije prekoračiti.

Treće ograničenje odnosi se na postaje. Jednu postaju smije posjećivati samo jedan autobus, što znači da dva učenika koji idu na istu stanicu nužno idu na isti autobus.

Zadnja pretpostavka jest da početna i krajnja točka svake autobusne linije mora biti škola. Dakle, nijednom učeniku škola ne smije biti dodijeljena postaja, iako mu je u dometu. Odnosno, iako je učenik udaljen od škole za udaljenost manju od maksimalne dozvoljene udaljenosti, mora mu biti dodijeljena neka autobusna postaja, ne smije pješačiti do škole.

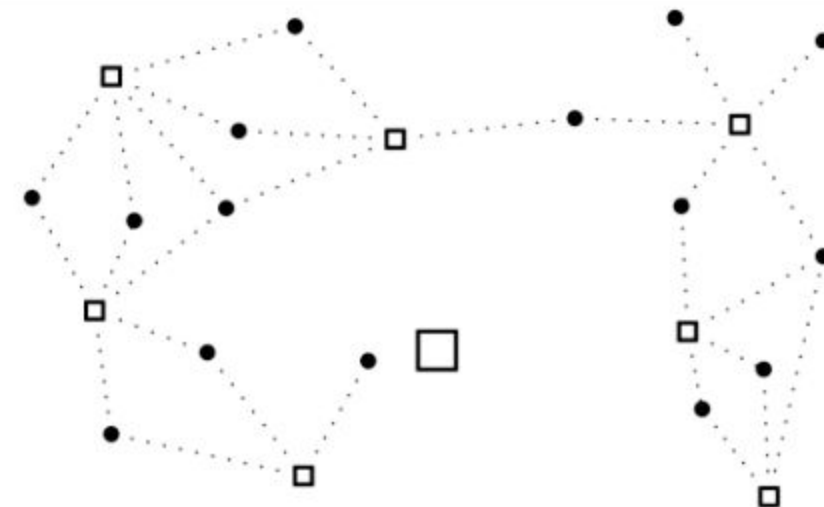


Figure 1: Primjer instance problema.

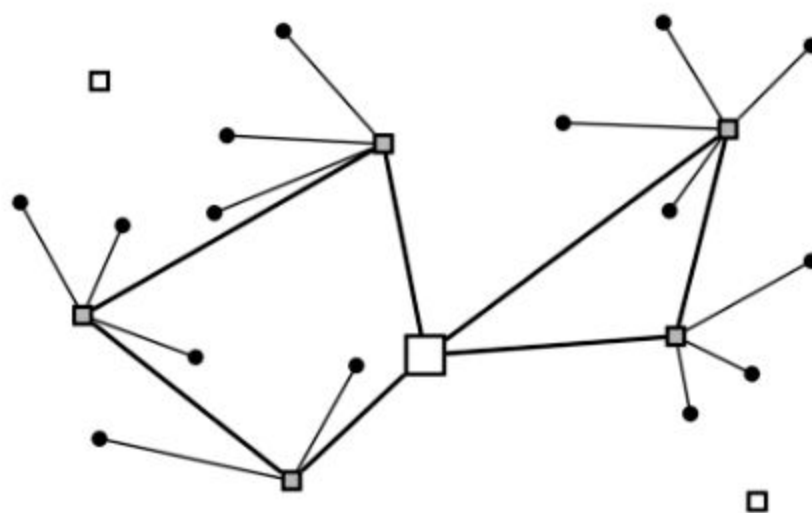


Figure 2: Primjer rješenja problema.

# PODACI

## ULAZNI PODACI

Ulazni podatak je jedna instanca problema. U svakoj instanci problema prvo se navodi broj potencijalnih autobusnih postaja. Nakon toga naveden je kapacitet autobusa, iza čega slijedi maksimalna dozvoljena udaljenost učenika od stanice te kapacitet autobusa.

Nakon toga slijede koordinate potencijalnih autobusnih postaja, od kojih je prva škola (postaja s indeksom 0).

Na kraju, slijede koordinate mjesta stanovanja učenika.

Primjer datoteke instance problema:

```
81 stops, 400 students, 40.000 maximum walk, 25 capacity
```

```
0      50.000      50.000
1      48.318      46.209
2      52.009      49.907
3      59.662      43.504
4      54.935      48.097
(.....)
```

```
1      48.387      54.189
2      47.385      47.888
3      64.043      10.534
4      49.000      17.578
```

## FORMAT RIJEŠENJA

Datoteke sa dobivenim rješenjima moraju biti formata navedenog u nastavku.

Prvo su napisane autobusne linije (identifikator stanice odgovara onima zadanim instancom problema). Primjerice, prva autobusna linija u ovom rješenju je:

**0 → 6 → 18 → 33 → 0**

Svaka nova autobusna linija zapisana je u novom retku. Nakon autobusnih linija zapisane su dodijeljene autobusne stanice za svakog učenika. Primjerice, učeniku 1 dodijeljena je autobusna stanica 22, učeniku 2 autobusna stanica 13 itd.

6 18 33  
25 35 2 14 3  
(.....)

1 22  
2 13  
3 2  
4 53  
(.....)

## ALGORITAM

### OPIS KORIŠTENOG ALGORITMA

Za rješavanje problema korištena je inačica imunološkog algoritma - CLONALG.

Imunološki algoritam spada u skupinu prirodnim inspiriranih optimizacijskih algoritama. On je, kao što mu i samo ime kaže, razvijen na temelju načina rada imunološkog mehanizma kod živih bića. Teoriju rada imunološkog sustava dao je Burnet 1957. godine.

*CLONALG (Clonal Selection Algorithm)* razvili su De Castro i Von Zuben 1999. te ga 2002. godine i razvili. CLONALG je populacijski algoritam koji radi s populacijom antitijela. Pri tome je svako antitijelo zapravo jedno rješenje problema koji se optimira. Antigen u ovom kontekstu predstavlja samu funkciju čiji se optimum traži. Afinitet pojedinog antitijela (rješenja) prema antigenu (funkciji) tada je predstavljen kvalitetom (tj. dobrotom; engl. fitness) samog rješenja. Ukoliko se radi o maksimiziranju funkcije, tada je afinitet najčešće jednak upravo iznosu same funkcije u promatranom rješenju.

```

inicijaliziraj  $P^{(0)} = \{x_1, x_2, \dots, x_d\}$ 
ponavljaj dok nije zaustavi( $P^{(t)}$ )
    evaluiraj( $P^{(t)}, Ag$ )
     $P^{(t)} = \text{odaberi}(P^{(t)}, n)$ 
     $P^{clo} = \text{kloniraj}(P^{(t)}, \beta)$ 
     $P^{hyp} = \text{hipermutiraj}(P^{clo})$ 
    evaluiraj( $P^{hyp}, Ag$ )
     $P' = \text{odaberi}(P^{hyp}, n)$ 
     $P^{birth} = \text{stvorinove}(d)$ 
     $P^{(t+1)} = \text{zamijeni}(P', P^{birth})$ 
     $t = t+1$ 
kraj ponavljanja

```

Slika 1: Pseudokod CLONALG algoritma

CLONALG je zapravo nadogradnja SIA algoritma (*Simple Immunological Algorithm*). Nadogradnja se odnosi na kloniranje i mutacije. Dok se kod SIA algoritma sve jedinke kloniraju i mutiraju jednako, kod CLONALG algoritma sve se to događa ovisno o afinitetu (dobroti) pojedine jedinke. Kloniranje se događa *proporcionalno* sa afinitetom odnosno, bolje jedinke će se klonirati više puta. Hipermutacije se događaju *obrnuto proporcionalno* afinitetu, odnosno, bolje jedinke se mutiraju manje.

Parametri algoritma su:

- $Ag$  - antigen, funkcija koju je potrebno optimirati
- $n$  - broj antitijela u populaciji
- $d$  - broj novih jedinki koje ćemo u svakoj iteraciji stvoriti i dodati u populaciju
- $\beta$  - parametar koji određuje veličinu populacije klonova

Algoritam započinje stvaranjem inicijalne populacije antitijela  $P^{(0)}$  veličine  $d$ . Zatim se ulazi u petlju koja se ponavlja tako dugo dok neki od uvijeta zaustavljanja nije zadovoljen (pronalazak dovoljno dobrog rješenja ili prekoračen maksimalan broj iteracija). Zatim dolazi do kloniranja. Kao što je već spomenuto, broj klonova određenog antitijela proporcionalan je njegovom afinitetu (veći afinitet, više klonova). Ukupan broj klonova koji ulazi u populaciju označen je sa  $N_c$  (izraz pod sumom zaokružujemo na najbliži cijeli broj).

$$N_c = \sum_{i=1}^n [(\beta \cdot n) / i]$$

Zatim svako antitijelo iz  $P^{clo}$  prolazi kroz proces hipermutacije čime nastaje nova populacija  $P^{hyp}$  iste veličine. Potom se za sva antitijela iz populacije  $P^{hyp}$  računaju afiniteti, odnosno, vrednuje dobrotu rješenja. Na kraju se iz  $P^{hyp}$  izabire  $n$  najboljih antitijela i ona tada čine novu populaciju  $P^{(t+1)}$  ( $t$  je oznaka iteracije). Na kraju se, radi diverzifikacije, slučajnim mehanizmom stvori  $d$  novih antitijela, te tih  $d$  novih antitijela zamjenjuje  $d$  antitijela sa najmanjim afinitetom.

## HIPERMUTACIJA

Hipermutacija antitijela se sastoji od više različitih mutacija. U ovoj implementaciji algoritma koriste se tri mutacije. Prva mutacija s određenom vjerojatnošću mijenja stanicu učenika na neku drugu u dozvoljenom radijusu kretanja. Druga mutacija s određenom vjerojatnošću odbacuje jedan bus, a stanice koje je posjećivao taj bus raspodjeljuje između ostalih busova. Treća mutacija mijenja redoslijed dviju stanica busa. Vjerojatnosti mutacija su hiperparametri koji se zadaju na početku izvođenja programa.

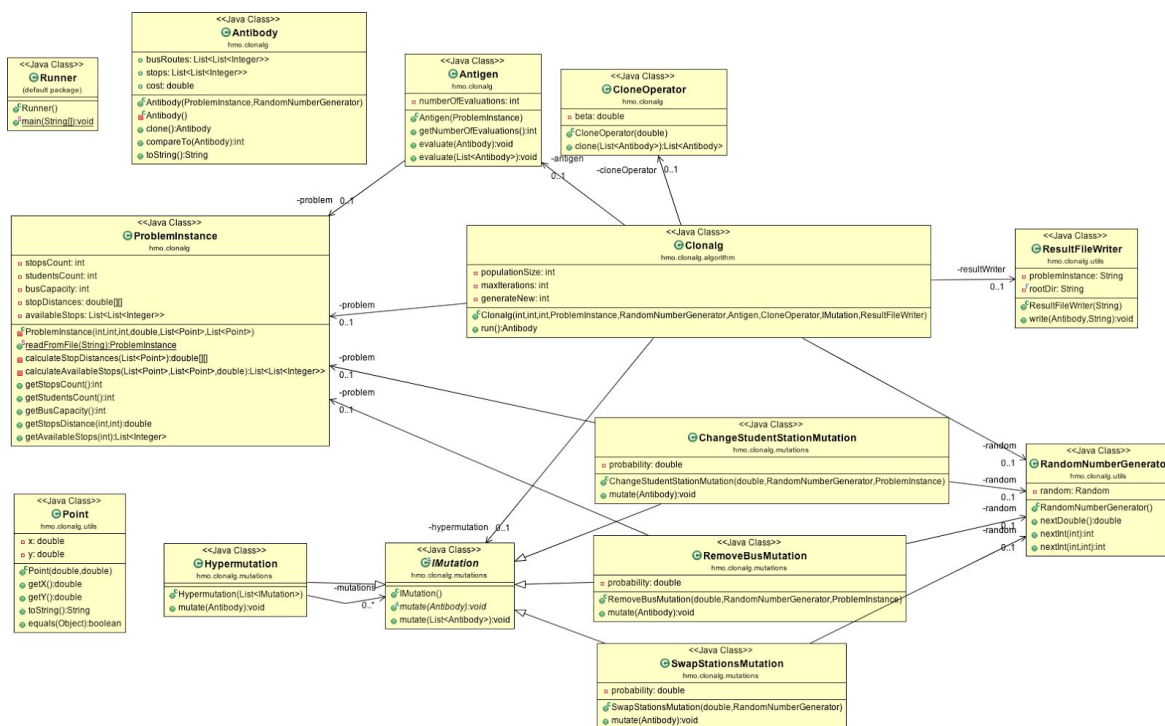
## HIPERPARAMETRI

- Veličina populacije - 100
- Broj novostvorenih antitijela u svakoj iteraciji - 5
- Ukupan broj iteracija - 150000
- Vjerojatnost zamjene stanice učenika - 0.05
- Vjerojatnost odbacivanja busa - 0.05
- Vjerojatnost zamjene redoslijeda dviju stanica - 0.1
- $\beta$  - 0.5



# IMPLEMENTACIJA

Kao što je već spomenuto, korišten je imunološki algoritam CLONALG. Na slici 2. nalazi se dijagram razreda.



Slika 2. Dijagram razreda

Cilj rješavanja ovog problema je minimizacija sume udaljenosti koje autobusi prelaze. Matematički se to zapis funkcije cilja (antigena):

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} \sum_{k=1}^n x_{ijk}$$

Inicijalna populacija dobiva se na način da se prvo učenici rasporede na postaje, pri tome pazeći na uvjet da na jednoj postaji ne bude više učenika nego što je kapacitet jednog autobusa. Zatim se napravi slučajna permutacija stanica te se te stanice tim redoslijedom dodijeljuju busevima. Dodijeljivanje stanica busevima radi se na način da se autobusu dodijeljuju stanice dok broj ljudi koji bi trebao ući u autobus ne prijeđe njegov kapacitet. Jedini kriterij zaustavljanja je ukupan broj iteracija.

## REZULTATI

Algoritam je testiran na 10 različitih instanci problema. Za svaku instancu zapisana su rješenja nakon 1 minute, 5 minuta izvođenja te bez vremenskog ograničenja (ograničenje je broj iteracija). U ovom slučaju kada nema vremenskog ograničenja, algoritam se izvede u 150 000 iteracija te se zaustavi.

## ZAKLJUČAK

Kvaliteta rezultata se poboljšava s duljinom izvođenja algoritma pa bi se duljim izvođenjem programa mogla dobiti još bolja rješenja. Dodatno poboljšanje rješenja bi se moglo postići dodavanjem novih mutacija. Povećavanjem vjerojatnosti mutacija, povećava se prostor pretraživana rješenja te prevelikim povećanjem pretraga postaje nasumična. Smanjivanjem vjerojatnosti mutacija, brzo se smanji prostor pretraživanja rješenja i algoritam lakše ostane u nekom lokalnom optimumu. Prebrza konvergencija se može izbjeći i povećavanjem broja novo generiranih antitijela.