

Elektromagnetna metaheuristika

Seminarski rad u okviru kursa
Metodologija stručnog i naučnog rada
Matematički fakultet

Antić Dimitrije, Novaković Andrija, Golubović Stefan,
Stanojević Nikola

mi16128@alas.matf.bg.ac.rs, mi16068@alas.matf.bg.ac.rs,
mi16135@alas.matf.bg.ac.rs, mi16092@alas.matf.bg.ac.rs

15. april 2020.

U ovom tekstu je ukratko prikazana metaheuristika za globalnu optimizaciju inspirisana elektromagnetizmom, uz osvrt na njenu konkretnu primenu na *problem alokacije habova* i *problem izbora atributa*. Prikazana metaheuristika koristi princip privlačenja i odbijanja, kako bi se rešenja približila optimalnom rešenju. Laka implementacija metaheuristike daje veliki značaj u njenom potencijalu za primenu na razne probleme iz prakse.

Sadržaj

1	Uvod	2
2	Osnovni pojmovi	2
3	Elektromagnetna metaheuristika	2
3.1	Motivacija	2
3.2	Opšta procedura	4
3.2.1	Inicijalizacija	4
3.2.2	Lokalna pretraga	5
3.2.3	Računanje rezultujuće sile	5
3.2.4	Pomeranje čestica	7
4	UMAHLP problem	8
4.1	Formulacija problema	8
4.2	Primena EM heuristike	9
5	Izbor atributa	10
5.1	Opis algoritma	10
5.2	Primena EM heuristike	11
5.3	Rezultati	11
6	Zaključak	12
	Literatura	12

1 Uvod

Globalna optimizacija je polje koje se u bliskoj istoriji ubrzano razvija i tema je mnogobrojnih istraživanja. Veliki broj problema iz realnog sveta kao npr. fizike, hemije ili molekularne biologije uključuje nelinearne funkcije više promenljivih sa određenim osobinama (više ekstremuma, prekidnost itd.) koje je teško optimizovati postojećim, konvencionalnim matematičkim alatima, kao što su gradijentne metode[2]. U cilju prevazilaženja navedenih poteškoća, kao i mnogih drugih, 1980-ih godina počele su da se razvijaju stohastičke metode. Nažalost, takve metode ne dovode uvek do željenih rezultata zbog visoke dimenzionalnosti i raznih drugih nepravilnosti. Metaheuristika koja će biti prikazana je jedna od takvih metoda, i koristi se upravo za globalnu optimizaciju. Poglavlje 2 opisuje osnovne pojmove za dalje razumevanje, poglavlje 3 sadrži motivaciju za uvođenje elektromagnetne metaheuristike i prikazuje opštu proceduru metode. Poglavlja 4 i 5 prikazuju direktne primene na probleme iz prakse.

2 Osnovni pojmovi

Definicije preuzete iz [1] i [7]. Optimizacione metode u opštem slučaju traže optimum u prostoru dopustivih rešenja.

Rešenje je *dopustivo* ako ispunjava ograničenja problema.

Funkcija cilja je funkcija čiji optimum, odnosno minimalnu ili maksimalnu vrednost, želimo da nađemo, a da pritom sva ograničenja nad domenom njenih vrednosti budu ispunjena.

Prema tipu domena nad kojim se vrši, optimizacija se deli na:

- *globalnu* - dopustivi skup vrednosti je iz domena realnih brojeva.
- *kombinatornu (diskretnu)* - dopustivi skup vrednosti iz konačnog ili beskonačnog skupa celih brojeva.

Prema fokusu pretrage, optimizacija se deli na:

- *globalnu* - traži globalni optimum.
- *lokalnu* - traži lokalni optimum.

3 Elektromagnetna metaheuristika

Bitno je napomenuti razliku između pojmova *metaheuristika* i *heuristika*. Naime, *metaheuristika* u opštem smislu nije svesna problema na koji se primenjuje, te služi kao opšti metod za rešavanje, dok se pojam *heuristika* odnosi na konkretnu primenu metoda na problem za koji je prilagođena. Primeri heuristika su dati u poglavljima 4 i 5.

3.1 Motivacija

U stohastičkoj globalnoj optimizaciji, algoritmi bazirani na populaciji se inicijalizuju tako što se iz dopustivog skupa vrednosti biraju slučajne tačke koje će predstavljati rešenje problema. U zavisnosti od vrednosti ciljne funkcije za

svako od rešenja, određuju se regioni u kojima postoji šansa da se nađe pogodno rešenje. Nakon toga se, izabranim mehanizmom, utiče na tačke rešenja tako da se izabrani regioni dodatno pretraže, u cilju nalaženja optimalnog rešenja. U dvofaznim metodama, pretraga dopustivih rešenja se izvodi slučajnim izborom, koji je praćen eng. *hill-climbing* metodama ili metodama zasnovanim na gradijentima[5].

Predložena elektromagnetna heuristika ohrabruje tačke rešenja da konvergiraju ka tzv. *dolinama* odnosno obeshrabruju tačke da se kreću uz tzv. *strma brda*. Motivacija leži u tome što se u *dolinama* može pronaći vrednost funkcije koja je vrlo verovatno manja od one na koju se može naići kada bi se tačka kretala uz *strma brda*. Ovakav metod pretraga omogućava primenu koncepta na kom se zasniva elektromagnetizam - *privlačenje* i *odbijanje*.

Za svaku tačku koja predstavlja rešenje, u daljem tekstu **čestica**, može se pretpostaviti da ima svoje naelektrisanje i da se tako naelektrisana kreće kroz prostor. Naelektrisanje će, kao i u elementarnom elektromagnetizmu, uticati na jačinu kojom će se čestice međusobno privlačiti, odnosno odbijati. Ciljna funkcija opisuje kvalitet čestice u smislu traženja optimalne vrednosti. Jasno je da česticama koje imaju veću vrednost ciljne funkcije treba dodeliti veće naelektrisanje, kako bi više privlačila ostale čestice iz populacije.

Nakon što se za svaku česticu izračuna naelektrisanje, ono se dalje koristi za nalaženje smera u pretrazi skupa dopustivih rešenja u narednim iteracijama. Smer kretanja se određuje izračunavanjem rezultujućeg vektora svih sila kojima ostale čestice deluju na posmatranu.

Iako je ideja za nastanak metode proistekla kao posledica koncepta na kojima se zasniva elektromagnetizam, postoje i bitne razlike koje će biti pomenute kasnije. Metaheuristika se uglavnom primenjuje uz male izmene. Često se kao modifikacije dodaju *lokalne pretrage*, koje za očekivani rezultat imaju poboljšanje ciljnih funkcije određenih čestica populacije. Takve modifikacije biće detaljnije opisane u poglavljima 4 i 5.

U narednom poglavlju, biće reč o sledećem optimizacionom problemu sa ograničenjima:

$$\min f(\mathbf{x})$$

gde je

$$\mathbf{x} \in [\mathbf{l}, \mathbf{u}]$$

a vrednosti $\mathbf{l}, \mathbf{u} \in \mathbb{R}^n$. Preciznije $[\mathbf{l}, \mathbf{u}] := \{\mathbf{x} \mid l_k \leq x_k \leq u_k, k = \overline{1, n}\}$.

Biće korišćena notacija kao u originalnom radu[2]. Dati ulazni parametri problema su:

1. n - dimenzija problema
2. u_k - gornja granica k -te koordinate
3. l_k - donja granica k -te koordinate
4. $f(x)$ - funkcija čiji optimum tražimo

3.2 Opšta procedura

U opštem slučaju, algoritam se odvija u 4 faze:

- inicijalizacija algoritma
- izračunavanje ukupne sile na svaku česticu
- pomeranje čestica duž rezultujućeg vektora sile
- primena pretrage u okolini kako bi se pronašao lokalni minimum

Kao dobar rezultat za konvergenciju dovoljno je izabrati $max_iter = 25n$, gde je n dimenzija dozvoljenog prostora. Postoje i drugi kriterijumi zaustavljanja koji se koriste u optimizacionim metodama. Jedan od kriterijuma koji je pokazao dobre rezultate je zaustavljanje nakon određenog broja iteracija bez promene odnosno poboljšanja rezultata. U literaturi se može pronaći još različitih kriterijuma, oni su detaljnije obrađivani u radu [8].

EM(*broj_cestica*, *max_iter*, *ls_iter*, δ)

```
1: inicijalizuj() 3.2.1
2: iter  $\leftarrow$  1
3: while iter < max_iter do
4:   lokalna_pretraga(ls_iter,  $\delta$ ) 3.2.2
5:    $\mathbf{F} \leftarrow$  izracunaj_sile() 3.2.3
6:   pomeri( $\mathbf{F}$ ) 3.2.4
7:   iter  $\leftarrow$  iter + 1
8: end while
```

3.2.1 Inicijalizacija

Procedura inicijalizacije podrazumeva slučajan izbor *broj_cestica* tačaka iz prostora dopustivih rešenja. Za svaku koordinatu izabrane tačke se podrazumeva da je $x_k \in U[l_k, u_k]$ ¹. Koristi se uniformna raspodela kako bi sve tačke imale istu verovatnoću izbora. Nakon što su tačke izabrane, za svaku tačku se računa vrednost ciljne funkcije $f(x)$, i tačka sa najvećom vrednošću ciljne funkcije se zapamti.

inicijalizuj()

```
1: iter  $\leftarrow$  1
2: while i < broj_cestica do
3:   while j < n do
4:     pom  $\leftarrow$   $U[0, 1]$ 
5:      $x_k^i \leftarrow l_k + pom \cdot (u_k - l_k)$ 
6:     j  $\leftarrow$  j + 1
7:   end while
8:   izracunaj_f( $x^i$ )
9:   i  $\leftarrow$  i + 1
10: end while
11:  $x^{najbolje} \leftarrow \operatorname{argmax}\{f(x^i), i = \overline{1, broj\_cestica}\}$ 
```

¹Uniformna raspodela intervala $[l_k, u_k]$

3.2.2 Lokalna pretraga

Lokalna pretraga kao parametre dobija ls_iter i δ koji predstavljaju broj iteracija za lokalnu pretragu za rešenje x^i i multiplikativnu konstantu za pretragu okoline, redom.

Procedura se odvija u nekoliko koraka:

- Maksimalan korak se računa u odnosu na parametar δ .
- Za svaku česticu, poboljšanje se računa za svaku koordinatu. Preciznije, za datu koordinatu, vrednost i -te čestice se privremeno čuva. Slučajan broj se bira kao dužina koraka i privremena vrednost se pomera za dužinu koraka, ako nova vrednost ima bolju vrednost ciljne funkcije vrednost čestice se menja njom i lokalna pretraga se završava. Inače, isti proces se ponavlja ls_iter puta.

Moguće je koristiti i moćnije metode lokalne pretrage, ali i sa ovako jednostavnom metodom algoritam pokazuje veliku sklonost konvergenciji.

lokalna_pretraga(ls_iter, δ)

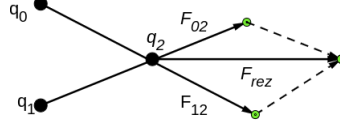
```
1:  $br \leftarrow 1$ 
2:  $duzina \leftarrow \delta(\max_k(u_k - l_k))$ 
3: for  $i = 1$  to  $broj\_cestica$  do
4:   for  $k = 1$  to  $n$  do
5:      $\lambda \leftarrow U[0, 1]$ 
6:     while  $br < ls\_iter$  do
7:        $tmp \leftarrow x^i$ 
8:        $\eta \leftarrow U[0, 1]$ 
9:       if  $\lambda > 0.5$  then
10:         $y_k \leftarrow y_k + \eta \cdot (duzina)$ 
11:       else
12:         $y_k \leftarrow y_k - \eta \cdot (duzina)$ 
13:       end if
14:       if  $f(y) < f(x^i)$  then
15:         $x^i \leftarrow y$ 
16:         $br \leftarrow ls\_iter - 1$ 
17:       end if
18:        $br \leftarrow br + 1$ 
19:     end while
20:   end for
21: end for
22:  $x^{najbolje} \leftarrow \operatorname{argmax}\{f(x^i), i = \overline{1, n}\}$ 
```

Promenljive λ i η se koriste kao promenljive odlučivanja, za dodavanje slučajnosti tokom pretrage okoline.

3.2.3 Računanje rezultujuće sile

Zakon *superpozicije*, definisan u teoriji elektromagnetizma, kaže da je rezultujuća sila koja deluje na posmatranu tačku obrnuto srazmerna rastojanju

između tačaka, a direktno proporcionalna proizvodu njihovih naelektrisanja. Ilustracija zakona superpozicije je prikazana na slici 1.



Slika 1: *Princip superpozicije*

Kao što je opisano u algoritmu 1, u svakoj iteraciji se za svaku česticu računa njeno naelektrisanje imajući u vidu njenu vrednost ciljne funkcije. Jasno je da se naelektrisanje svake čestice menja u svakoj iteraciji.

Neka je sa q^i označeno naelektrisanje čestice x^i . Tako dato naelektrisanje se računa po formuli:

$$q^i = \exp \left(-n \cdot \frac{f(x^i) - f(x^{najbolje})}{\sum_{k=1}^{broj_cestica} (f(x^k) - f(x^{najbolje}))} \right), i = \overline{1, broj_cestica} \quad (1)$$

Iz jednačine 1 jasno sledi da će čestice koje imaju bolju vrednost ciljne funkcije imati veće naelektrisanje, i time više uticati na ostale čestice iz populacije, što je i bio jedan od zahteva opisan u poglavlju 3.1.

Kako se sav račun odvija u računaru, kod velikih populacija, razlomak može biti veoma mali pa može dovesti do prekoračenja ili potkoračenja pri izračunavanju eksponencijalne funkcije. Da bi se zaobišli takvi problemi, uvodi se množenje sa n^2 .

Naelektrisanje se računa u odnosu na ciljnu funkciju, postoje i alternativne metode, ali one neće biti opisane u ovom radu.

Kao što je navedeno u poglavlju 3.1, postoje bitne razlike između elektromagnetizma u fizičkom smislu i onoga što se koristi u metaheuristikama. Naime, kako se može primetiti u formuli 1 nikakav znak se ne dodaje naelektrisanju q^i , što u fizici nije slučaj. Umesto davanja znakova naelektrisanjima, o smeru kretanja čestice se odlučuje drugačijom metodom - poređenjem ciljnih funkcija posmatrane i svake čestice koja deluje na nju. Tako se dobija F^i koja predstavlja rezultujuću silu u čestici i :

$$F^i = \sum_{j \neq i}^{broj_cestica} \begin{cases} (x^j - x^i) \cdot \frac{q^i q^j}{\|x^j - x^i\|^2}, & \text{ako je } f(x^j) < f(x^i). \\ (x^i - x^j) \cdot \frac{q^i q^j}{\|x^j - x^i\|^2}, & \text{ako je } f(x^j) \geq f(x^i) \end{cases} \quad (2)$$

²Dimenzija prostora dopustivih rešenja

izracunaj_sile()

```

1: for  $i = 1$  to  $broj\_cestica$  do
2:    $q^i \leftarrow \exp\left(-n \frac{f(x^i) - f(x^{najbolje})}{\sum_{k=1}^{broj\_cestica} (f(x^k) - f(x^{najbolje}))}\right)$ 
3:    $F^i \leftarrow 0$ 
4: end for
5: for  $i = 1$  to  $broj\_cestica$  do
6:   for  $j = 1$  to  $broj\_cestica$  do
7:     if  $f(x^j) < f(x^i)$  then
8:        $F^i \leftarrow F^i + (x^j - x^i) \cdot \frac{q^i q^j}{\|x^j - x^i\|^2}$  privlačenje
9:     else
10:       $F^i \leftarrow F^i - (x^j - x^i) \cdot \frac{q^i q^j}{\|x^j - x^i\|^2}$  odbijanje
11:    end if
12:  end for
13: end for

```

3.2.4 Pomeranje čestica

Nakon što sve čestice dobiju svoje naelektrisanje, i smer kretanja, potrebno ih je pomeriti kako bi se približili ka boljim rešenjima odnosno udaljili od loših. Za tu proceduru potrebno je izabrati parametar η koji će predstavljati dužinu koraka. Ovde će biti izabran kao vrednost iz uniformne raspodele između 0 i 1, naravno da je moguće koristiti i neku drugu raspodelu, ali takvi slučajevi neće biti opisivani. Veličina koraka se u ovom slučaju uzima kao slučajan broj iz navedene raspodele kako bi postojala ne-nula verovatnoća kretanja ka neposećenim regionima.

$$x^i = x^i + \eta \frac{F^i}{\|F^i\|} V, i = \overline{1, broj_cestica} \quad (3)$$

Sa V je dat vektor kretanja, čije komponente predstavljaju dozvoljena kretanja ka gornjoj granici u_k i donjoj l_k . Normalizacija se vrši kako bi održali dopustivost rešenja.

pomeri()

```

1: for  $i = 1$  to  $broj\_cestica$  do
2:   if  $i \neq najbolje$  then
3:      $\eta \leftarrow U[0, 1]$ 
4:      $F^i \leftarrow \frac{F^i}{\|F^i\|}$ 
5:     for  $k = 1$  to  $n$  do
6:       if  $F_k^i > 0$  then
7:          $x_k^i \leftarrow x_k^i + \eta \cdot F_k^i \cdot (u_k - x_k^i)$ 
8:       else
9:          $x_k^i \leftarrow x_k^i + \eta \cdot F_k^i \cdot (x_k^i - l_k)$ 
10:      end if
11:    end for
12:  end if
13: end for

```

4 UMAHLP problem

Kako bi se razumeo kontekst problema, potrebno je prvo objasniti osnovne pojmove. Naime, u savremenim mrežama, kao što su telekomunikacione ili čak mreže pošte, moraju da postoje čvorovi koji odlučuju kome će sledećem poruka biti prosleđena kako bi došla do odredišta. Takvi čvorovi se nazivaju *eng. hub*, i u daljem tekstu će se razlikovati od krajnjih čvorova koji učestvuju u komunikaciji.

4.1 Formulacija problema

Uncapacitated multiple allocation hub location problem je problem alokacije habova, odnosno izbora koji čvorovi u mreži treba da budu oni koji su zaduženi za rutiranje. Pritom nema ograničenja za broj čvorova koji mogu biti hubovi. Takođe, ne postavlja se ograničenje koliko količinu saobraćaja jedan hub može da prenese, niti jedan krajnji čvor mora pripadati samo jednom hubu. Uspostavljanje svakog huba zahteva određen trošak resursa. Cilj je rešiti problem tako da ukupna cena prenosa poruka i ukupan trošak za uspostavljanje habova budu minimizovani.

Detaljnija objašnjenja problema i njegovih varijacija data su u radu [4]. U nastavku biće data matematička formulacija problema. Korišćena je notacija iz literature [9].

Pretpostavimo da je dato n različitih čvorova u mreži. Neka je taj skup označen $S = \{1, \dots, n\}$, pritom svaki od čvorova može biti ili hub ili krajnji čvor. Neka je:

- C_{ij} rastojanje od i -tog do j -tog čvora
- W_{ij} zahtev od i -tog do j -tog čvora
- $y_k = \begin{cases} 1, & \text{ako je } k\text{-ti čvor hub.} \\ 0, & \text{inače.} \end{cases}$
- x_{ijkm} predstavlja deo protoka W_{ij} iz čvora i koji je prikupljen od strane haba k , a prosleđen od strane haba m do odredišnog čvora j

Kako se svaka putanja sastoji iz tri koraka:

1. od početnog čvora do prvog haba
2. transfer između habova
3. prenos od poslednjeg haba do odredišnog čvora

Uvode se parametri α koji predstavlja jediničnu cenu komunikacije hub-hub, i po definiciji je manji od 1. Parametri χ, δ predstavljaju jedinične cene prikupljanja odnosno prenos do odredišnog čvora, redom. Tada je ciljna funkcija:

$$\min \sum_{i,j,k,m} W_{ij} \cdot (\chi \cdot C_{ik} + \alpha \cdot C_{km} + \delta \cdot C_{mj}) \cdot x_{ijkm} + \sum_k f_k \cdot y_k \quad (4)$$

čiji ćemo minimum tražiti pri uslovima:

$$\sum_{k,m} x_{ijkm} = 1, \text{ za sve } i, j \quad (5)$$

$$\sum_m x_{ijkm} + \sum_{m, m \neq k} x_{ijmk} \leq y_k, \text{ za sve } i, j, k \quad (6)$$

$$y_k \in \{0, 1\}, \text{ za svako } k \quad (7)$$

$$x_{ijkm} \geq 0, \text{ za sve } i, j, k, m \quad (8)$$

Ograničenje 5 govori da se svi protoci šalju između svih krajnjih čvorova, ograničenje 6 garantuje da protok putuje samo preko otvorenih hubova. Ograničenja 7 i 8 garantuju ne-negativnost i binarnost promenljivih odlučivanja.

4.2 Primena EM heuristike

Koristi se pristup već opisan u poglavlju 3.2. U nastavku će biti dat pseudo-kod i objašnjenja procedura koje su modifikovane.

EM nad UMAHLP

```

1: inicijalizuj()
2: iter  $\leftarrow$  1
3: while iter < max_iter do
4:   for all  $x^k$  u populaciji do
5:     izracunaj_ciljnu()
6:     1_swap_lokalna_pretraga( $x^k$ )
7:     normalizuj( $x^k$ )
8:   end for
9:   izracunaj_naelektrisanja_i_silu()
10:  pomeri()
11:  if nema_promena_maksBrIteracija() then
12:    kraj()
13:  end if
14:  iter  $\leftarrow$  iter + 1
15: end while

```

Kako je y_k binarna promenljiva, njena vrednost se dobija po formuli:

$$y_k = \begin{cases} 1, & \text{ako je } x_{k,i} \geq 0.5 \\ 0, & \text{inače} \end{cases}$$

Za svaki izabrani skup hubova, najkraći putevi do krajnjih čvorova u mreži se dobijaju Floyd-Warshallovim algoritmom (opisan u [3]). Nakon nalaženja najkraćih puteva, između parova čvorova, vrši se evaluacija ciljne funkcije kako bi česticama bila dodeljena naelektrisanja i izračunati rezultujući vektori (sila kojom ostale čestice deluju na posmatranu i smer njenog kretanja). Sve procedure ostaju iste kao u poglavlju 3.2, osim lokalne pretrage. U ovoj heuristici se koristi tzv. *1_swap_lokalna_pretraga* i biće ukratko opisana.

Korišćena lokalna pretraga pokušava u svakoj iteraciji da zameni (*eng. swap*) jedan element y_k njenom komplementarnom vrednošću. Naime, ako je $y_k = 1$, dobija vrednost 0 i obrnuto. Što zapravo predstavlja da ako je čvor bio hub sada više nije. Proces lokalne pretrage se nastavlja kao što je opisano u poglavlju 3.2.2. Procedura normalizacije opisana je u poglavlju 3.2.3. Detaljniji opis rezultata metode i podacima nad kojim je primenjivana opisani su u radu [9] i prevazilaze okvire ovog teksta, te neće biti opisani.

5 Izbor atributa

Izbor atributa (eng. *feature selection*) je proces odabira podskupa atributa nekog skupa podataka. Izabrani podskup za cilj ima da najbolje, u poređenju sa ostalim podskupovima, ispunjava određeni kriterijum, npr. preciznost klasifikacije. Originalna formulacija problema i rešenja pomoću EM metaheuristike dati su u radu [6].

Metode izbora atributa za cilj imaju smanjenje vremenskih i računskih troškova u rešavanju problema istraživanja podataka. Kako raste potreba za istraživanjem podataka, tako raste i njihova količina i dimenzionalnost. Samim tim, bitno je imati metod kojim se ti podaci mogu redukovati bez gubitka informacija koje njihovi prvobitni atributi sadrže. U narednom poglavlju biće opisana heuristika bazirana na elektromagnetizmu primenjena na pomenuti problem.

5.1 Opis algoritma

Sledeći parametri će biti korišćeni kao ulazni:

- `br_iter` - broj iteracija glavne petlje
- `M` - broj čestica
- α - skalirajući parametar iz $[0, 1]$
- `r` - instanca problema (skup podataka)
- `N` - dimenzija instance problema

Proces inicijalizacije čestica, detaljnije opisan u 3.2.1, se izvršava pozivanjem metode *create random points* kojom se generiše vektor dimenzije `M`, dalje označen sa p . Svaka od koordinata jedne čestice je slučajno izabran broj iz intervala $[0, 1]$. Svaka od koordinata i -te čestice:

$$p_k^i = \begin{cases} 1, & \text{ako je } k\text{-ti atribut uključen u rešenje.} \\ 0, & \text{inače.} \end{cases}, k = \overline{1, N} \quad (9)$$

Kako se vrednosti biraju uniformno iz intervala $[0, 1]$, koordinata se transformiše u binarnu po formuli:

$$\overline{p_k^i} = \begin{cases} 1, & \text{ako je } p_k^i \geq 0.5. \\ 0, & \text{inače.} \end{cases}, k = \overline{1, N} \quad (10)$$

Vrednost praga 0.5 daje jednaku šansu svakom atributu ulaznog skupa da bude uključen ili ne. Nakon inicijalizacije, u svakoj iteraciji glavne petlje se izračunava ciljna funkcija za svaku tačku. Postupak lokalne pretrage (poglavlje 3.2.2) se izvršava za tačku ako su ispunjeni uslovi:

- Tačka ima najbolji ili drugi najbolji rezultat
- Lokalna pretraga nije bila primenjivana na tu tačku ili joj se vrednost promenila u odnosu na poslednju loklanu pretragu.
- Najbolja vrednost se nije promenila u najmanje 10 poslednjih generacija

Kao algoritam lokalne pretrage, korišćena je metoda `1_swap_lokalna_pretraga()` opisana u poglavlju 4.

5.2 Primena EM heuristike

Sama procedura algoritma se ne razlikuje od onog opisanog u poglavljima 3.2 i 4. Kako je jedina razlika u izračunavanju ciljne funkcije, ona će biti opisana:

```

ciljna_funkcija( $p_i$ , klasifikator)


---


1:  $s_i = \text{dekodiraj}(p_i)$ 
2:  $\text{pronadjen} \leftarrow \text{pretrazi\_kes}(p_i)$ 
3: if  $\text{pronadjen} == \text{null}$  then
4:   if klasifikator == 1NN then
5:      $\text{obj} \leftarrow \text{1nn\_5fold\_cv}(s_i, r)$ 
6:   else
7:      $\text{obj} \leftarrow \text{svm\_2fold\_cv}(s_i, r)$ 
8:   end if
9:    $\text{dodaj\_u\_kes}(s_i, \text{obj})$ 
10: else
11:    $\text{obj} = \text{dohvati\_iz\_kesa}(s_i)$ 
12:    $\text{kesiraj}(s_i)$ 
13: end if

```

Kako bi se izbegla ponovljena izračunavanja nad istim podskupovima atributa, koristi se procedura keširanja, odnosno čuvanje rezultata računa u memoriji.

Metode $\text{1nn_5fold_cv}(s_i, r)$ i $\text{svm_2fold_cv}(s_i, r)$ klasifikuju dati ulazni skup podataka r koristeći podskup skupa atributa s_i , i kao rezultat vraćaju preciznost klasifikatora. Izračunata preciznost predstavlja vrednost ciljne funkcije.

Naelektrisanje i rezultujuća sila svake čestice se računaju po postupku opisanom u poglavlju 3.2.3.

U slučajevima kada su vrednosti koordinata čestica bliske 0.5, dolazi do čestih promena koordinate iz 0 u 1 i obrnuto. Usled tih promena javlja se nepotrebno rasipanje pretrage i konvergencija algoritma opada. Kako bi se ovo ponašanje izbeglo, standardni algoritam se modifikuje dodavanjem specijalne procedure skaliranja. Ako sa p_k^i obeležimo k -tu koordinatu i -te čestice:

$$p_k^i = \alpha \cdot s_k^i + (1 - \alpha) \cdot p_k^i \quad (11)$$

Naime, uvedena procedura omogućava bolju kontrolu variranja vrednosti koordinata. Takođe, skaliranje ima svoje mane. Usled smanjenja prostora pretrage, može doći do prerane konvergencije, što dalje dovodi do dobijanja neoptimalnog rešenja. Nalaženje dobrog balansa između širenja i sužavanja prostora pretrage je ključno za ovakve metode.

Faktor skaliranja uzima vrednosti iz skupa $[0, 1]$. U opštem slučaju, velike vrednosti ovog parametra dovode do lokalnog optimuma, a premale vrednosti neće ubrzati konvergenciju dovoljno.

5.3 Rezultati

U narednoj tabeli 1 prikazani su rezultati dobijeni u primenom elektromagnetne heuristike, i upoređeni su sa rezultatima dobijenim primenom metoda potpune pretrage, *eng. Full Search* i genetskim algoritmom. *Full Search* je metod pretrage po svim mogućim podskupovima skupa atributa. Rezultati preuzeti iz rada [6].

Tabela 1: Rezultati

skup podataka	br atributa	br klasa	br instanci	FS	genetski alg	EML
abalone	8	11	3842	23.99	24.37	opt
iris	4	3	150	99.39	98.00	opt
spambase	57	2	4601	-	91.55	94.35
letter	16	26	20000	96.39	95.24	opt

Vrednost *opt* opisuje slučaj kada je rezultat pri klasifikaciji, uz korišćenje metode za izbor atributa, bio identičan onome koji dobije metod potupne pretrage.

6 Zaključak

Elektromagnetna metaheuristika se može koristiti kao samostalan algoritam ili kao podrška nekim drugim algoritmima. Glavni koncept algoritma podrazumeva usmeravanja čestica prema lokalnim optimumima koristeći mehanizam privlačenja i odbijanja, i kao takav pokazuje dobre rezultate u praksi. Laka implementacija čini je pogodnom za rešavanje širokog skupa problema, uključujući i neke NP-teške.

Literatura

- [1] Engelbrecht Andries. *Computational intelligence - An introduction*. John Willey & Sons, 2007.
- [2] S. C. Fang. Birbil S. I. An electromagnetism-like mechanism for global optimization. *Journal of Global Optimization*, 25:263–282, 2003.
- [3] M. Krishnamoorthy Ernst A. T. An exact solution approach based on shortest-paths for p-hub median problem. *INFORMS Journal of Computing*, 10:149–162, 1998.
- [4] Campbell J. F. Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72:387–405, 1994.
- [5] Timmer G. T Kan, A. H. G. R. Stochastic global optimization methods Part II: Multilevel methods. *Mathematical programming*, 39:57–78, 1987.
- [6] Aleksandar Kartelj. An Improved Electromagnetism-like Method for Feature Selection. *Journal of Mult.-Valued Logic Soft Computing*, 0:1–19, 2015.
- [7] Aleksandar Kartelj. Ukratko o optimizaciji, 2019. on-line at: <http://poincare.matf.bg.ac.rs/~kartelj/nastava/RI2019/04.Optimizacija.ukratko.pdf>.
- [8] Ali M. M. Törn, A. and S. Viitanen. Stochastic global optimization: Problem classes and solution techniques. *Journal of Global Optimization*, 14:437–447, 1999.
- [9] Filipović V. An electromagnetism metaheuristic for the uncapacitated multiple allocation hub location problem. *Serdica Journal of Computing*, 5:261–272, 2011.