

# Optimizacija kolonijom pčela Bee Colony Optimization (BCO)

Tatjana Davidović,

Matematički institut SANU

3. decembar 2015.



# Sadržaj

- 1 Biološke osnove
- 2 Optimizacija kolonijom pčela
- 3 Detalji implementacije
- 4 Primeri primene



# Inteligencija grupe (Swarm Intelligence, SI)

Algoritmi optimizacije inspirisani SI

- Algoritmi inspirisani prirodnim procesima;
- Rade nad populacijom rešenja;
- Koriste inteligenciju grupe (roja);



# Inteligencija grupe (Swarm Intelligence, SI)

## Algoritmi optimizacije inspirisani SI

- Algoritmi inspirisani prirodnim procesima;
- Rade nad populacijom rešenja;
- Koriste inteligenciju grupe (roja);
- Interakcijom i razmenom znanja između jedniki usmerava se pretraga ka boljim regionima;



# Inteligencija grupe (Swarm Intelligence, SI)

## Algoritmi optimizacije inspirisani SI

- Algoritmi inspirisani prirodnim procesima;
- Rade nad populacijom rešenja;
- Koriste inteligenciju grupe (roja);
- Interakcijom i razmenom znanja između jedniki usmerava se pretraga ka boljim regionima;
- Najpoznatiji predstavnici:  
Optimizacija mravlјim kolonijama (Ant Colony Optimization, ACO),  
optimizacija kolonijom pčela (Bee Colony Optimization, BCO),  
optimizacija rojevima čestica (Particle Swarm Optimization, PSO).



# Ponašanje pčela u prirodi

[1] S. Camazine, and J. Sneyd, "A model of collective nectar source by honey bees: Self-organization through simple rules", J. Theor. Biol. vol. 149, 1991, pp. 547-571.

- Pčele izviđači kreću u potragu za hranom u blizini košnice;



# Ponašanje pčela u prirodi

[1] S. Camazine, and J. Sneyd, "A model of collective nectar source by honey bees: Self-organization through simple rules", J. Theor. Biol. vol. 149, 1991, pp. 547-571.

- Pčele izviđači kreću u potragu za hranom u blizini košnice;
- Po povratku, odlučuju se za neku od sledećih mogućnosti:
  - ① postaju *regruteri*, tako što plesom obaveštavaju druge pčele o lokaciji, količini i kvalitetu nektara;
  - ② ostaju *lojalne*, vraćaju se na polje nektara i nastavljaju skupljanje;
  - ③ odustaju od pronađenog izvora hrane i postaju *neopredeljene*.



# Ponašanje pčela u prirodi

[1] S. Camazine, and J. Sneyd, "A model of collective nectar source by honey bees: Self-organization through simple rules", J. Theor. Biol. vol. 149, 1991, pp. 547-571.

- Pčele izviđači kreću u potragu za hranom u blizini košnice;
- Po povratku, odlučuju se za neku od sledećih mogućnosti:
  - ① postaju *regruteri*, tako što plesom obaveštavaju druge pčele o lokaciji, količini i kvalitetu nektara;
  - ② ostaju *lojalne*, vraćaju se na polje nektara i nastavljaju skupljanje;
  - ③ odustaju od pronađenog izvora hrane i postaju *neopredeljene*.
- Neopredeljene pčele na plesnom podiju biraju novu lokaciju nektara i slede uputstva izabranog regrutera;



# Ponašanje pčela u prirodi

[1] S. Camazine, and J. Sneyd, "A model of collective nectar source by honey bees: Self-organization through simple rules", J. Theor. Biol. vol. 149, 1991, pp. 547-571.

- Pčele izviđači kreću u potragu za hranom u blizini košnice;
- Po povratku, odlučuju se za neku od sledećih mogućnosti:
  - ① postaju *regruteri*, tako što plesom obaveštavaju druge pčele o lokaciji, količini i kvalitetu nektara;
  - ② ostaju *lojalne*, vraćaju se na polje nektara i nastavljaju skupljanje;
  - ③ odustaju od pronađenog izvora hrane i postaju *neopredeljene*.
- Neopredeljene pčele na plesnom podiju biraju novu lokaciju nektara i slede uputstva izabranog regrutera;
- Odluke o lojalnosti i proces regrutacije direktno zavise od kvaliteta i kvantiteta pronađenog izvora hrane.



# Jezik pčela (Waggle dance)



# Ilustracija ponašanja pčela u prirodi

(PceliceSaVirtuelnomKamerom.swf)



# BCO: Opis metode

- Optimizaciona metoda inspirisana ponašanjem pčela u potrazi za hranom;
- Predložili je Lučić i Teodorović, 2001;
- Svaka pčela odgovara jednom dopustivom rešenju razmatranog problema;
- Postoje dve osnovne varijante: konstruktivna i metoda sa popravkom;
- Stohastika i verovatnoća koriste se u procesu odlučivanja o lojalnosti i regrutaciji.



# Razlike između pčela u prirodi i veštačkih pčela

- Veštačkih pčela ima mnogo manje nego u pravoj košnici;
- Košnica je virtualni pojam, nema fizičku lokaciju;
- Sve veštačke pčele su uključene u potragu;
- Komunikacija je sinhrona;
- Veštačke pčele su podeljene u samo dve grupe:
  - ① Regruteri i
  - ② neopredeljene pčele.



# BCO - Opis Algoritma

- Gradi/popravlja rešenja za svaku pčelu kroz iteracije (let unapred/let unazad, forward/backward pass);



# BCO - Opis Algoritma

- Gradi/popravlja rešenja za svaku pčelu kroz iteracije (let unapred/let unazad, forward/backward pass);
- Komunikacija između pčela podrazumeva razmenu informacija o kvalitetu (parcijalnih) rešanja:



# BCO - Opis Algoritma

- Gradi/popravlja rešenja za svaku pčelu kroz iteracije (let unapred/let unazad, forward/backward pass);
- Komunikacija između pčela podrazumeva razmenu informacija o kvalitetu (parcijalnih) rešanja;
- Na osnovu toga svaka pčela izvrši jednu od sledećih akcija:
  - ① Napušta svoje trenutno rešenje, postaje neopredeljena (uncommitted) i preuzima rešenje neke druge pčele;
  - ② Nastavlja da dograđuje/modifikuje trenutno rešenje i privlači neopredeljene pčele da ga preuzmu (recruiter).



# BCO - Opis Algoritma

- Gradi/popravlja rešenja za svaku pčelu kroz iteracije (let unapred/let unazad, forward/backward pass);
- Komunikacija između pčela podrazumeva razmenu informacija o kvalitetu (parcijalnih) rešanja;
- Na osnovu toga svaka pčela izvrši jednu od sledećih akcija:
  - ① Napušta svoje trenutno rešenje, postaje neopredeljena (uncommitted) i preuzima rešenje neke druge pčele;
  - ② Nastavlja da dograđuje/modifikuje trenutno rešenje i privlači neopredeljene pčele da ga preuzmu (recruiter).
- Najbolje dobijeno rešenje prijavljuje se korisniku kao konačno;



# BCO - Opis Algoritma

- Gradi/popravlja rešenja za svaku pčelu kroz iteracije (let unapred/let unazad, forward/backward pass);
- Komunikacija između pčela podrazumeva razmenu informacija o kvalitetu (parcijalnih) rešanja;
- Na osnovu toga svaka pčela izvrši jednu od sledećih akcija:
  - ① Napušta svoje trenutno rešenje, postaje neopredeljena (uncommitted) i preuzima rešenje neke druge pčele;
  - ② Nastavlja da dograđuje/modifikuje trenutno rešenje i privlači neopredeljene pčele da ga preuzmu (recruiter).
- Najbolje dobijeno rešenje prijavljuje se korisniku kao konačno;
- Parameteri:
  - ①  $B$  - broj pčela;
  - ②  $NC$  - broj koraka tokom jednog leta unapred/broj letova unapred.



# BCO - pseudokod

Inicijalizacija: Čitanje ulaznih podataka, postavljanje vrednosti parametara BCO-a i kriterijuma zaustavljanja.

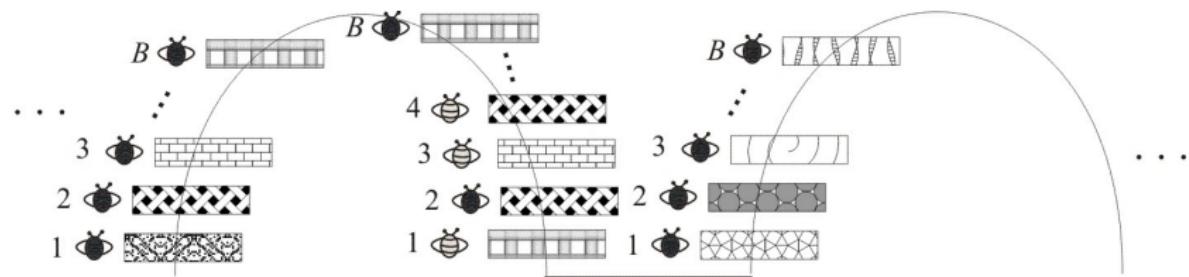
*Do*

- (1) Dodeli (prazno) rešenje svakoj pčeli.
- (2) *For* ( $i = 0; i < NC; i++$ )
  - // let unapred
  - (i) *For* ( $b = 0; b < B; b++$ )
    - (a) Proceni moguće poteze;
    - (b) Izaberi potez pomoću ruleta;
  - // let unazad
  - (ii) *For* ( $b = 0; b < B; b++$ )
    - Proceni (parcijalna) rešenja pčele  $b$ ;
  - (iii) *For* ( $b = 0; b < B; b++$ )
    - Odluči o lojalnosti pomoću ruleta za pčelu  $b$ ;
  - (iv) *For* ( $b = 0; b < B; b++$ )
    - If* ( $b$  je nelojalna), izaberi regrutera pomoću ruleta.
- (3) Oceni sva rešenja i pronađi najbolje. Ažuriraj  $x_{best}$  i  $f(x_{best})$ .  
*while* kriterijum zaustavljanja nije zadovoljen.  
**return** ( $x_{best}, f(x_{best})$ )

Figure: Pseudo-kod za BCO algoritam



# BCO - ilustracija



# BCO - Let unapred

- Zavisi od problema;



# BCO - Let unapred

- Zavisi od problema;
- Cilj je da se izgradi/popravi rešenje pridruženo svakoj pčeli;



# BCO - Let unapred

- Zavisi od problema;
- Cilj je da se izgradi/popravi rešenje pridruženo svakoj pčeli;
- Koristi randomizovane (stohastičke) pohlepne (proždrljive) postupke;



# BCO - Let unapred

- Zavisi od problema;
- Cilj je da se izgradi/popravi rešenje pridruženo svakoj pčeli;
- Koristi randomizovane (stohastičke) pohlepne (proždrljive) postupke;
- Komponente/transformacije boljih karakteristika imaju više šansi da budu izabrane.



# BCO - Let unazad

- Evaluacija rešenja (Normalizacija za min)

$$O_b = \frac{y_{\max} - y_b}{y_{\max} - y_{\min}}, \quad O_b \in [0, 1], \quad b = 1, 2, \dots, B$$



# BCO - Let unazad

- Evaluacija rešenja (Normalizacija za min)

$$O_b = \frac{y_{\max} - y_b}{y_{\max} - y_{\min}}, \quad O_b \in [0, 1], \quad b = 1, 2, \dots, B$$

- Verovatnoća lojalnosti:

$$p_b^{u+1} = e^{-\frac{O_{\max} - O_b}{u}}, \quad b = 1, 2, \dots, B$$



# BCO - Let unazad

- Evaluacija rešenja (Normalizacija za min)

$$O_b = \frac{y_{\max} - y_b}{y_{\max} - y_{\min}}, \quad O_b \in [0, 1], \quad b = 1, 2, \dots, B$$

- Verovatnoća lojalnosti:

$$p_b^{u+1} = e^{-\frac{O_{\max} - O_b}{u}}, \quad b = 1, 2, \dots, B$$

- Regrutovanje:

$$p_b = \frac{O_b}{\sum_{k=1}^R O_k}, \quad b = 1, 2, \dots, R$$



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;
- Uvođenje novih funkcija lojalnosti;



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;
- Uvođenje novih funkcija lojalnosti;
- Uvođenje heterogenih veštačkih pčela;



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;
- Uvođenje novih funkcija lojalnosti;
- Uvođenje heterogenih veštačkih pčela;
- Paralelizacija (MPI i openMP);



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;
- Uvođenje novih funkcija lojalnosti;
- Uvođenje heterogenih veštačkih pčela;
- Paralelizacija (MPI i openMP);
- Teoretska verifikacija (dokaz konvergencije);



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;
- Uvođenje novih funkcija lojalnosti;
- Uvođenje heterogenih veštačkih pčela;
- Paralelizacija (MPI i openMP);
- Teoretska verifikacija (dokaz konvergencije);
- Kombinacija konstruktivnog i algoritma sa poboljšanjem;



# BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- Uvođenje globalnog znanja;
- Varijanta sa poboljšanjem (modifikacijom) kompletnih rešenja;
- Uvođenje novih funkcija lojalnosti;
- Uvođenje heterogenih veštačkih pčela;
- Paralelizacija (MPI i openMP);
- Teoretska verifikacija (dokaz konvergencije);
- Kombinacija konstruktivnog i algoritma sa poboljšanjem;
- Hibridizacija sa drugim metodama.



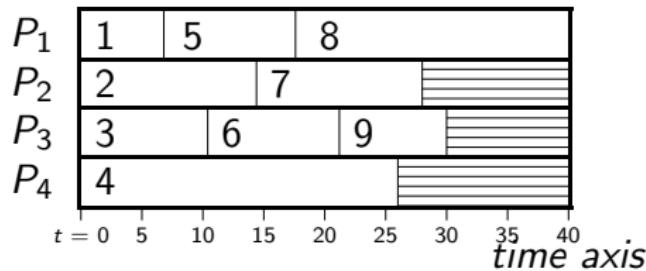
# Raspoređivanje nezavisnih zadataka na identične mašine

$T = \{1, 2, \dots, n\}$  - Skup zadataka,

$M = \{1, 2, \dots, m\}$  - Skup mašina,

$I_i$  - dužina izvršavanja svakog zadatka  $i$  ( $i = 1, 2, \dots, n$ ).

Cilj: Minimizacija ukupnog vremena izvršavanja svih zadataka (*makespan*).



Gantt diagram–raspodela zadataka po mašinama



# Koraci BCO algoritma

Gradi se rešenje:  $n/NC$  zadataka se dodaju trenutnom parcijalnom rešenju u svakom letu unapred.

Verovatnoća izbora zadatka  $i$  je:

$$p_i = \frac{l_i}{K}, \quad i = 1, 2, \dots, n \quad (1)$$

$$\sum_{k=1}^n l_k$$

gde je  $K$  broj neraspoređenih zadataka.

Odgovarajući procesor bira se pomoću **best fit** heuristike, ali tako da rezultujuće rešenje ne bude gore od trenutno najboljeg - globalno znanje, pohlepni princip.



# *p*-centar problem

- Dat je skup od  $n$  čvorova (lokacija, korisnika);



# *p*-centar problem

- Dat je skup od  $n$  čvorova (lokacija, korisnika);
- Data je  $D = [d_{ij}]_{n \times n}$  matrica rastojanja između čvorova;



# *p*-centar problem

- Dat je skup od  $n$  čvorova (lokacija, korisnika);
- Data je  $D = [d_{ij}]_{n \times n}$  matrica rastojanja između čvorova;
- Cilj je locirati  $p$  uslužnih centara tako da se minimizira maksimalno rastojanje između korisnika i pridruženog centra;



# *p*-centar problem

- Dat je skup od  $n$  čvorova (lokacija, korisnika);
- Data je  $D = [d_{ij}]_{n \times n}$  matrica rastojanja između čvorova;
- Cilj je locirati  $p$  uslužnih centara tako da se minimizira maksimalno rastojanje između korisnika i pridruženog centra;
- Korisnici se pridružuju najbližem uspostavljenom centru;

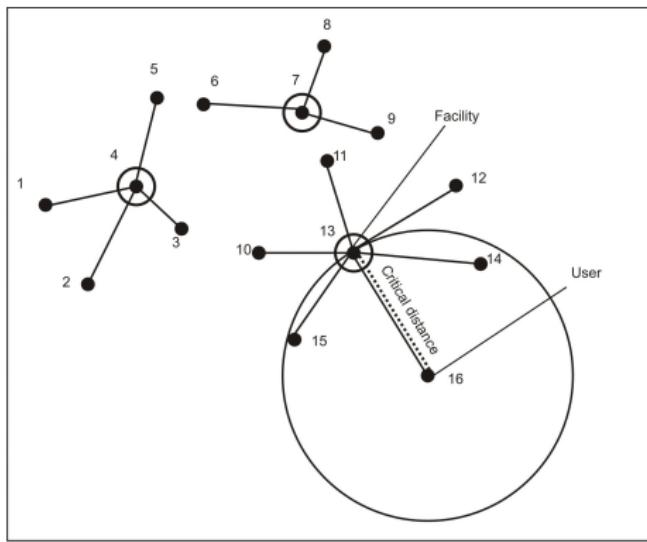


# *p*-centar problem

- Dat je skup od  $n$  čvorova (lokacija, korisnika);
- Data je  $D = [d_{ij}]_{n \times n}$  matrica rastojanja između čvorova;
- Cilj je locirati  $p$  uslužnih centara tako da se minimizira maksimalno rastojanje između korisnika i pridruženog centra;
- Korisnici se pridružuju najbližem uspostavljenom centru;
- Centri mogu biti locirani u bilo kom od  $n$  čvorova.



# Ilustracija problema



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbediti različit tretman za ista rešenja pridružena različitim pčelama (slučajna konstruktivna transformacija);



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbiti različit tretman za ista rešenja pridružena različitim pčelama (slučajna konstruktivna transformacija);
- Slučajno izabranih  $Q$  centara zamenjuje se ne-centrima;



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbediti različit tretman za ista rešenja pridružena različitim pčelama (slučajna konstruktivna transformacija);
- Slučajno izabranih  $Q$  centara zamenjuje se ne-centrima;
- $Q$  - se bira slučajno za svaku pčelu;



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbediti različit tretman za ista rešenja pridružena različitim pčelama (slučajna konstruktivna transformacija);
- Slučajno izabranih  $Q$  centara zamenjuje se ne-centrima;
- $Q$  - se bira slučajno za svaku pčelu;
- Prvo se  $Q$  ne-centara doda (dobija se nedopustivo rešenje) tako da se smanji kritično rastojanje korisnik-centar;



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbediti različit tretman za ista rešenja pridružena različitim pčelama (slučajna konstruktivna transformacija);
- Slučajno izabranih  $Q$  centara zamenjuje se ne-centrima;
- $Q$  - se bira slučajno za svaku pčelu;
- Prvo se  $Q$  ne-centara doda (dobija se nedopustivo rešenje) tako da se smanji kritično rastojanje korisnik-centar;
- $Q$  čvorova se izbaci iz spiska centara koristeći pohlepno pravilo;



# BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbediti različit tretman za ista rešenja pridružena različitim pčelama (slučajna konstruktivna transformacija);
- Slučajno izabranih  $Q$  centara zamenjuje se ne-centrima;
- $Q$  - se bira slučajno za svaku pčelu;
- Prvo se  $Q$  ne-centara doda (dobija se nedopustivo rešenje) tako da se smanji kritično rastojanje korisnik-centar;
- $Q$  čvorova se izbaci iz spiska centara koristeći pohlepno pravilo;
- Eksperimentalno dobijene granice za  $Q$ :  $Q \in [0, p]$  ako  $5 \cdot p < n$ , a inače  $Q \in [0, \frac{n-2.5 \cdot p}{2.5}]$ .



# Jos neke primene

- BAP - problem dodele vezova u luci;



# Jos neke primene

- BAP - problem dodele vezova u luci;
- PSAT - zadovoljivost formula u verovatnosnim i infinitezimalnim logikama;



# Jos neke primene

- BAP - problem dodele vezova u luci;
- PSAT - zadovoljivost formula u verovatnosnim i infinitezimalnim logikama;
- Razni primeri iz saobraćaja i transporta;



# Jos neke primene

- BAP - problem dodele vezova u luci;
- PSAT - zadovoljivost formula u verovatnosnim i infinitezimalnim logikama;
- Razni primeri iz saobraćaja i transporta;
- Sve implementacije su nad kombinatornom formulacijom problema.



# Doktorske disertacije

- [1] M. Šelmić, *Location problems on transport networks by computational intelligence methods*, PhD thesis, Faculty of Traffic and Transportation, University of Beograde, 2011.
- [2] M. Nikolić, *Resolving the consequences of traffic disturbances by bee colony optimization*, PhD thesis, Faculty of Traffic and Transportation, University of Beograde, 2015.
- [3] T. Stojanović, *The development and analisys of metaheuristics for satisfiability in probabilistic logics*, Faculty of Science, University of Kragujevac, 2015.
- [4] T. Jakšić Krüger, *The development, parallelization and theoretical verification of bee colony optimization*, Faculty of Technical Sciences, University of Novi Sad, 2015.



# Hvala na pažnji!

Pitanja?

Tatjana Davidović  
tanjad@mi.sanu.ac.rs

