

Optimizacija kolonijom pčela Bee Colony Optimization (BCO)

Tatjana Davidović,

Matematički institut SANU

11. januar 2013.



Sadržaj

- ① Inteligencija grupe (Swarm Intelligence, SI)
- ② Optimizacija kolonijom pčela
- ③ Detalji implementacije
- ④ Primeri primene



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).

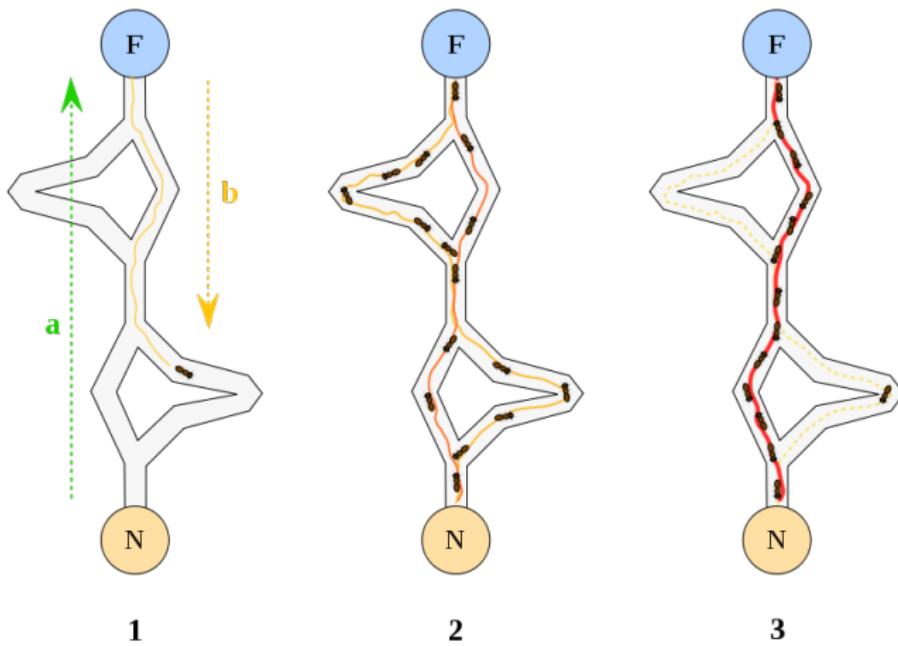


Optimizacija mravljim kolonijama

- Metoda inspirisana ponašanjem mrava u potrazi za hranom;
- Na putu ka izvoru hrane mravi ostavljaju feromone;
- Veća količina feromona privlači druge mrave da krenu istim putem;
- Kao posledica toga, bliži izvor hrane posetiće više mrava;
- Na putu kojim mravi ređe prolaze feromoni isparavaju.



Ilustracija kretanja mrava u potrazi za hranom



Optimizacija mravljim kolonijama: opis algoritma

Procedure ACO //**algoritam za probleme kombinatorne optimizacije**

Begin

Inicijalizacija.

Do

KonstruisiResenjeZaSvakogMrava;

PrimeniLokalnoPretrazivanje; //**opciono**

GlobalnoAzuriranjeVrednostiFeromona.

While (nije zadovoljen kriterijum zaustavljanja).

end //**ACO**



Opis algoritma (nastavak)

- Algoritam je konstruktivan;
- Osnovni parametar je broj mrava;
- Svaki mrav generivse po jedno rešenje slučajnim izborom komponenti na osnovu vrednosti feromona;
- Rešenja se porede i nalazi se najbolje među njima;
- Vrši se popravka rešenja primenom neke LS procedure (opciono);
- PRETHODNA DVA KORAKA MOGU DA ZAMENE MESTA



Opis algoritma (nastavak)

- Svim komponentama smanji se nivo feromona za koeficijent isparavanja (parametar);
- Komponentama koje pripadaju najboljem rešenju povećava se vrednost feromona;
- Pravila za ažuriranje vrednosti feromona mogu biti različita;
- Granice za vrednost feromona i koeficijent povećanja su takođe parametri.



BCO: Biološke osnove

- Metoda inspirisana ponašanjem pčela u potrazi za hranom;
- Predložili je Lučić i Teodorović, 2001;
- Pčele u prirodi tragaju za hranom i donose uzorke nektara u košnicu;
- U košnici se ispituje kvalitet donetog nektara;
- Kvalitetniji nektar reklamira se plesom odgovarajućih pčela kojim se ukazuje na pravac i daljinu njegovog izvora.



Ilustracija ponašanja pčela u prirodi

(PceliceSaVirtuelnomKamerom.swf)



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

Do

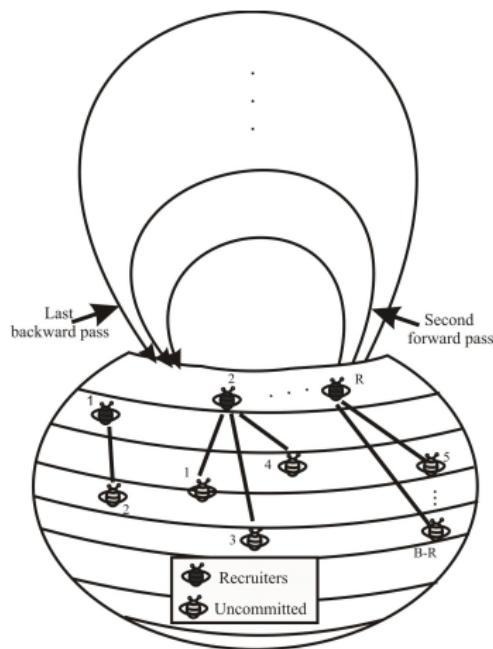
- Inicializacija: početno (prazno) rešenje dodeljuje se svakoj pčeli;
- *Do*
 - //forward pass
 - ① For ($b = 0; b < B; b++$)
 - For ($k = 0; k < NC; k++$) //brojač koraka
 - 1) Proceni sve moguće korake;
 - 2) Izaberi naredni korak koristeći pravila ruleta.
 - //backward pass (Razmena informacija među pčelama);
 - ① Odredi kvalitet (parcijalnog) rešenja za svaku pčelu;
 - ② Odredi lojalnost na osnovu pravila ruleta;
 - ③ Za svaku neopredeljenu pčelu izaberi jedno novo rešenje na osnovu pravila ruleta.

while (iteracija nije kompletirana);

- Uporedi sva rešenja i odredi najbolje;

while (nije zadovoljen kriterijum zaustavljanja).

BCO - ilustracija



BCO - Forward pass

- Zavisi od problema;



BCO - Forward pass

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



BCO - Forward pass

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



BCO - Forward pass

- 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 - Backward pass

- 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 - Backward pass

- 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 - Backward pass

- 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;
- URAĐENO: Uvođenje globalnog znanja;



BCO modifikacije

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



BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- URAĐENO: Uvođenje globalnog znanja;
- URAĐENO: Varijanta sa poboljšanjem (modifikacijom) kompletних rešenja;
- URAĐENO: Paralelizacija (MPI i openMP);



BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- URAĐENO: Uvođenje globalnog znanja;
- URAĐENO: Varijanta sa poboljšanjem (modifikacijom) kompletних rešenja;
- URAĐENO: Paralelizacija (MPI i openMP);
- URAĐENO: Teoretska verifikacija (dokaz konvergencije);



BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- URAĐENO: Uvođenje globalnog znanja;
- URAĐENO: Varijanta sa poboljšanjem (modifikacijom) kompletних rešenja;
- URAĐENO: Paralelizacija (MPI i openMP);
- URAĐENO: Teoretska verifikacija (dokaz konvergencije);
- U PLANU: Kombinacija konstruktivnog i algoritma sa poboljšanjem;



BCO modifikacije

- Inicijalno: Konstruktivni algoritam koji se sastoji od nezavisnih iteracija;
- URAĐENO: Uvođenje globalnog znanja;
- URAĐENO: Varijanta sa poboljšanjem (modifikacijom) kompletних rešenja;
- URAĐENO: Paralelizacija (MPI i openMP);
- URAĐENO: Teoretska verifikacija (dokaz konvergencije);
- U PLANU: Kombinacija konstruktivnog i algoritma sa poboljšanjem;
- U PLANU: Hibridizacija sa drugim metodama.



Razlike između ACO i BCO

- Komunikacija između mrava je indirektna i bazirana na pojedinačnim komponentama rešenja, a pčele komuniciraju direktnim poređenjem (parcijalnih) rešenja;



Razlike između ACO i BCO

- Komunikacija između mrava je indirektna i bazirana na pojedinačnim komponentama rešenja, a pčele komuniciraju direktnim poređenjem (parcijalnih) rešenja;
- ACO uglavnom koristi LS za popravljanje slučajno generisanih rešenja, što u BCO još nije korišćeno;



Razlike između ACO i BCO

- Komunikacija između mrava je indirektna i bazirana na pojedinačnim komponentama rešenja, a pčele komuniciraju direktnim poređenjem (parcijalnih) rešenja;
- ACO uglavnom koristi LS za popravljanje slučajno generisanih rešenja, što u BCO još nije korišćeno;
- Tokom razmene informacija u BCO koriste se samo informacije o trenutnim (parcijalnim) rešenjima dok je u ACO merodavno kompletno prethodno iskustvo;



Razlike između ACO i BCO

- Komunikacija između mrava je indirektna i bazirana na pojedinačnim komponentama rešenja, a pčele komuniciraju direktnim poređenjem (parcijalnih) rešenja;
- ACO uglavnom koristi LS za popravljanje slučajno generisanih rešenja, što u BCO još nije korišćeno;
- Tokom razmene informacija u BCO koriste se samo informacije o trenutnim (parcijalnim) rešenjima dok je u ACO merodavno kompletno prethodno iskustvo;
- za razliku od BCO, ACO algoritmi često koriste reinicijalizaciju za nivo feromona da spreče preuranjenu konvergenciju;



Razlike između ACO i BCO

- Komunikacija između mrava je indirektna i bazirana na pojedinačnim komponentama rešenja, a pčele komuniciraju direktnim poređenjem (parcijalnih) rešenja;
- ACO uglavnom koristi LS za popravljanje slučajno generisanih rešenja, što u BCO još nije korišćeno;
- Tokom razmene informacija u BCO koriste se samo informacije o trenutnim (parcijalnim) rešenjima dok je u ACO merodavno kompletno prethodno iskustvo;
- za razliku od BCO, ACO algoritmi često koriste reinicijalizaciju za nivo feromona da spreče preuranjenu konvergenciju;
- BCO ima manje parametara.



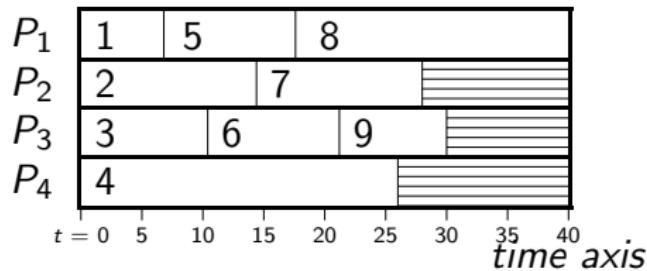
Raspoređivanje nezavisnih zadataka na identične mašine

$T = \{1, 2, \dots, n\}$ - 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 se bira tako da rezultujuće rešenje ne bude gore od trenutno najboljeg - globalno znanje, pohlepljni 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.



BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;



BCOi - Modifikacija rešenja

- To je glavni doprinos implementaciji BCO algoritma;
- Treba obezbiti 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.



Hvala na pažnji!

Pitanja?

