

Elektronski fakultet Niš



Primena Swarm algoritma pristupom
Ant Colony optimizacije

Student: Jovan Artonovic

Br. Indeksa: 1425

Sadržaj

1. Uvod
2. Primena Swarm algoritama zasnovana na ACO (Ant Colony Optimization) pristupu
3. Ant Colony Optimization
4. Konkretni problem primene ACO optimizacije
5. Rešenje zadatog problema u odabranom programskom okruženju

1. Uvod

Modeliranje ponašanja prirode je jedno od najintuitivnijih načina istraživanja danasnjice. Mnogi algoritmi su implementirani tako da oponasaju prirodu i na taj način dođu do efikasnog rešenja nekog problema. Jedna od najboljih tehnika danas za kreiranje algoritama zasnovanih na desavanjima u prirodi se odnosi na Swarm inteligenciju. Swarm metoda je zasnovana na tehnikama koje oponasaju insekte u prirodi i uz pomoć ovakvih tehnika dolazi se do rešenja nekih problema iz svakodnevnice. Ant Colony Optimization (u nastavku ACO) je tehnika koja se zasniva na oponasanju mrava u prirodi i njegovog prirodnog načina dolazanja do hrane.

U ovom dokumentu obradimo primenu ACO (Ant Colony Optimization) pristupa u primeni SWARM algoritma i pokazati da je ACO dobar optimizacijski algoritam za resavanje raznih problema zasnovanih na kombinatorici. U ovom dokumentu obradimo konkretan problem nalazanja najkraceg puta između dve lokacije uz pomoć ACO optimizacijskog algoritma.

2. Primena Swarm Algoritama zasnovana na ACO pristupu

ACO tehnika se prvi put pojavljuje u doktorskoj disertaciji napisanoj od strane Marco Dorigo. Dorigo je u svom radu pisao o problemu prodavaca za nalazanje najkracih puteva. Njegovo rešenje zasnovano je na feromonima koje upravo ostavljaju mravi prilikom nalazanja njihovog izvora hrane. Dakle, pravi insekti ostavljaju upravo hemijsko jedinjenje po kome se mravi u koloniji prepoznaju zvane feromoni. Dakle proces kreiranja najkraceg puta mravi formiraju upravo ostavljanjem ovakvih podataka na njihovom putu do hrane. Upravo ovaj pristup je pristup uz pomoć kojeg oni formiraju putanju od startne pozicije do krajnje pozicije koja predstavlja dolazak do hranljivog izvora. Mravi koji putuju kracom putanjom do hrane ostavljaju više feromona na putu i na taj način sugerisu da je to pravi i najkraci put do cilja.

ACO algoritam se još koristi za resavanje problema nalazanja najkraceg puta u grafu. I ovaj problem predstavlja najbolji primer primene ovog algoritma.

3. Opis problema za primenu ACO algoritma optimizacije

U ovom dokumenta prikazacemo problem nalazenja najkraceg puta u graphu od startnog cvora do ciljanog cvora. ACO algoritam je upravo dizajniran za resavanje slicnih ili istih problema. Svaki cvor ima informaciju o svojim susedima i izmedju svakog cvora postoji veza sa tezinom odnosno razdaljinom. Startuje se od polazne lokacije i sledeca lokacija bira se na osnovu minimalne razdaljine izmedju cvorova. Mrav ce najpre izabrati sledecu lokaciju na osnovu minimalne razdaljine i na tom putu ce ostaviti pheromone. Nivo pheromone-a uvek mora biti veci od treshold-a. Pheromone koji je mrav ostavio bice dostupan samo neko vreme na putanji do sledece lokacije nakon toga ce se njegova vrednost smanjivati. Kako bi izbegli isparavanje pheromone-a na putanju sa najkracom razdaljinom, kolicina pheromone-a bice onoliko koliko je potrebno da se sledeci mrav odluci na putanju na kojoj se nalazi neka kolicina pheromone-a. Dakle, mravi ce se kretati od cvora do cvora i to na osnovu kolicine pheromone-a na putanjama izmedju cvorova. Mravi ce na taj nacin odluciti koji sledeci cvor treba biti izabran.

Jednacina koja se odnosi na nalazenje najkraceg puta uz pomoc ACO algoritma prikazana je u nastavku:

$$P_{i,j} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_l \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}}$$

P_{ij} - Verovatnoca da se predje sa cvora i 'th to j 'th.

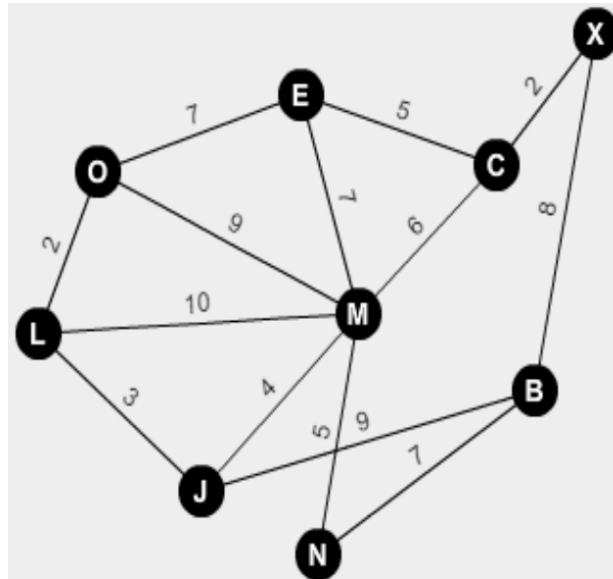
α - Kontrolise kolicinu pheromone-a

β - Pozeljna kolicina pheromone-a

4. Resenje problema uz pomoc ACO optimizaijskog algoritma

Prikazacemo graf u kome cvorovi predstavljaju lokacije sa obelezenim vrednostima razdaljine izmedju svakog cvora u grafu. Postojanje putanje izmedju dva cvora je oznaceno sa P_{ij} . Ukoliko postoji put izmedju dva cvora duzina puta izmedju ta dva cvora je oznacena sa D_{ij} . D_{ij} se ažurira uvek kada se doda nova lokacija u putanju izmedju

dva krajnja cvora. Lista suseda se definise za svaku lokaciju u grafu i prelazak na sledecu lokaciju zavisi upravo od liste suseda.



Odluka o narednom cvoru za obilazak donosi se na osnovu liste suseda od trenutne lokacije i takodje na osnovu obrasca navedenog u dokumentu. Jednom kada se detektuje ivica izmedju lokacija tada se nivo pheromone-a povecava na tom putu. Potrebno je da pheromone nivo uvek bude veci od vrednosti oznacene kao threshold na putanji izmedju dva cvora. Kad god je dostignuta nova lokacija u grafu proverava se nivo pheromone-a. Ukoliko je nivo pheromone-a na odredjenoj granici manji od vrednosti threshold/a ta putanja se vise ne prati od strane drugih ucesnika (mrava). Proverava se ime trenutne lokacije sa krajnjom lokacijom i ukoliko su lokacije iste algoritam prestaje.

Dakle, mrav k u nodu r ce izabrati sledcu lokaciju cvora s sa verovatnocom:

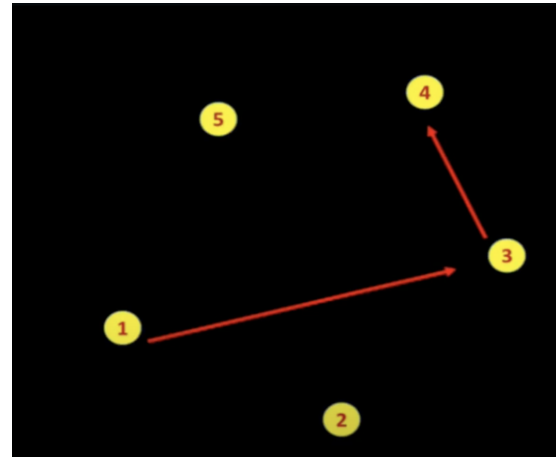
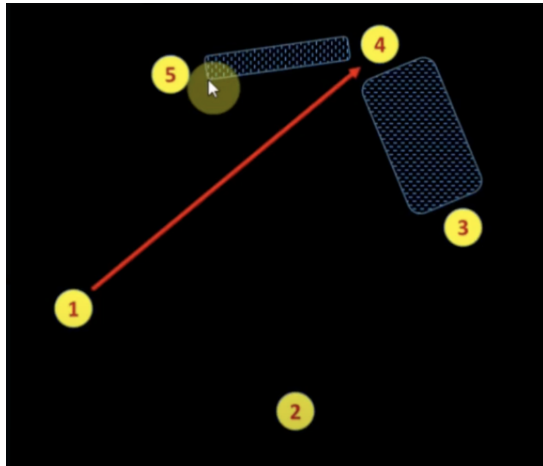
$$p_k(r, s) = \begin{cases} \tau(r, s)^\alpha \eta(r, s)^\beta / \sum_{u \in M_k} \tau(r, u)^\alpha \eta(r, u)^\beta, & \text{for } s \in M_k \\ 0, & \text{otherwise} \end{cases}$$

α - Predstavlja stepen vaznosti cvora zasnovan na nivou pheromone-a

β - Predstavlja vrednosti vidljivost

$u \in M_k$ - Predstavlja izbor mrava k kada je on bio u cvoru r

Svaki cvor poseduje listu cvorova koje moze da adresira iz cvora r osim cvorova koji su prethodno poseceni. Vidljivost cvora se uglavnom predstavlja kao $1 / (\text{rastojanje između cvora } r \text{ i cvora } s)$.



Na slikama su prikazana dva faktora koja ucestvuju u odlucivanju sledeceg cvora. Prvi je pheromone a drugi je vidljivost (visibility). Na prvoj slici se vidi da je vrednost pheromone-a od cvora 4 -> 3 veca u odnosu od cvora 4 -> 5.

ACO Algoritam

- Nasumicno smestamo mrave na lokacije
- Za svakog mrava :
 - Izaberi jos uvek ne posecenu lokaciju sve dok sve lokacije nisu posecene.
 - Optimizuj putanju od startnog cvora do dosegnutog
 - Azuriraj vrednost pheromone-a $T_{ij} += 1 / (\text{length}(\text{tour}))$
 - Oslabi pheromone sa $T_{ij} = (1 - \rho) * T_{ij}$

Kod aplikacije smesten je u okviru repozitorijuma pod sekcijom *Inteligentni Sistemi/Domaci Zadatak 1 Ant Colony*.

Link do repozitorijum-a :

<https://github.com/jocacode/Master-Elfak/tree/master/Inteligentni%20Sistemi/Domaci%20Zadatak%201%20-%20Ant%20Colony>

5. Literatura

1. ACO Based Shortest Path between Locations within a Campus, Maria Wenisch S 1 , Amali Asha A 2., Loyola Institute of Business Administration, Chennai, India 2 Loyola College, Chennai, India
2. Swarm Intelligence, Focus on Ant and Particle Swarm Optimization, Felix T. S. Chan and Manoj Kumar Tiwari
3. <https://pypi.org/project/ACO-Pants/>