Ant Colony Optimization: A New Meta-heuristic

Authors:
Marco Dorigo Gianni A. Di Caro

Presented by:

Md. Zawad Abdullah (1605002)
Bishwajit Bhattacharjee (1605003)
Md. Mahir Shahriyar (1605024)

Department of CSE,
Bangladesh University of Engineering and Technology

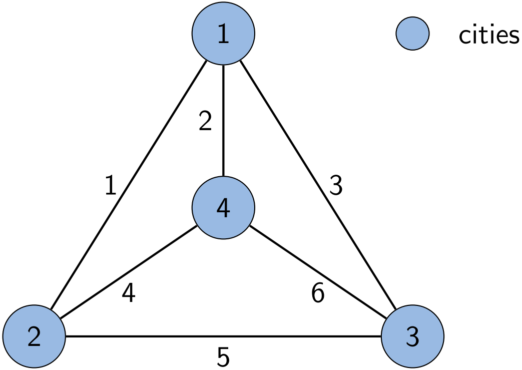
August 30, 2019

Problem Definition

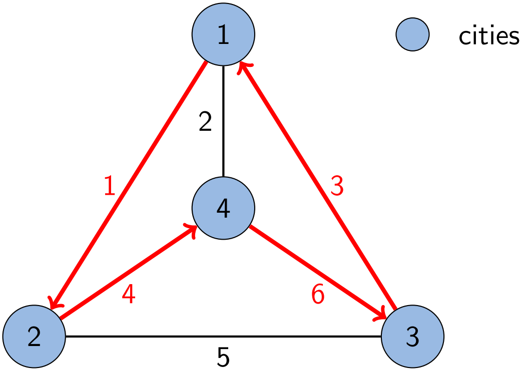
The Traveling Salesman Problem

A salesman needs to visit a number of customers located in different cities and return to the starting city using the shortest route.

Input:



Output:



Known Methods

- Backtracking

Known Methods

- Backtracking
  Issue - Complexity is exponential.

Known Methods

- Backtracking
  Issue - Complexity is exponential.
  Not Good Enough!

Known Methods

- Backtracking
  Issue - Complexity is exponential.
  Not Good Enough!

- Bitmask DP

Known Methods

- Backtracking
  Issue - Complexity is exponential.
  Not Good Enough!

- Bitmask DP
  Issue - Works efficiently when input is small.

Known Methods

- Backtracking
  Issue - Complexity is exponential.
  Not Good Enough!

- Bitmask DP
  Issue - Works efficiently when input is small.
  Not Good Enough!

Motivation

We will use Ant Colony Optimization (ACO) to solve TSP more efficiently.

Ants collecting food-1

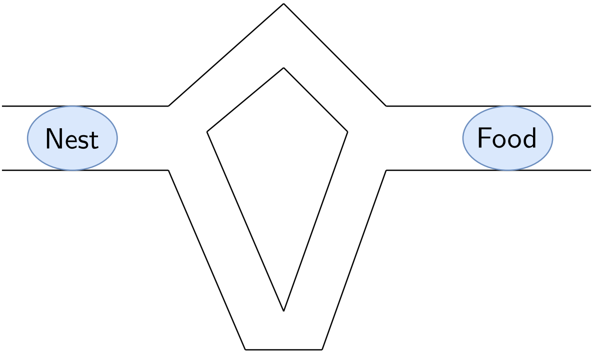


Figure: Paths From Food to Ants' Nest

Ants collecting food-2

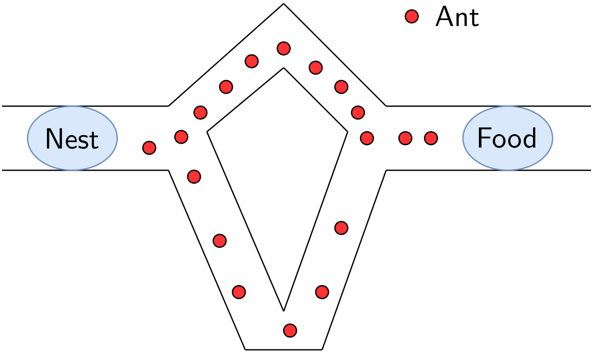


Figure: Ants Searching for Food

Ants collecting food-3

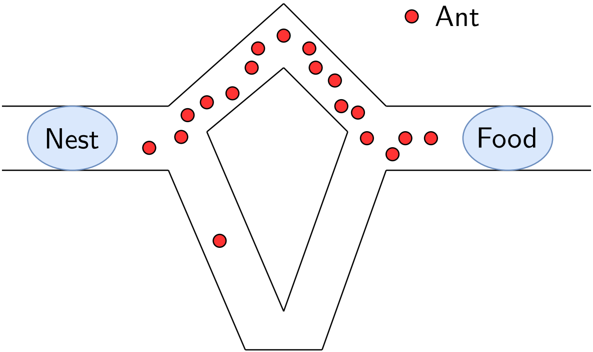


Figure: Ants Following An Optimal Path

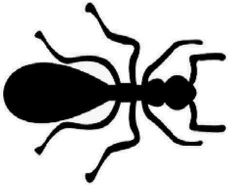
So how do they communicate??

So how do they communicate??

Pheromone

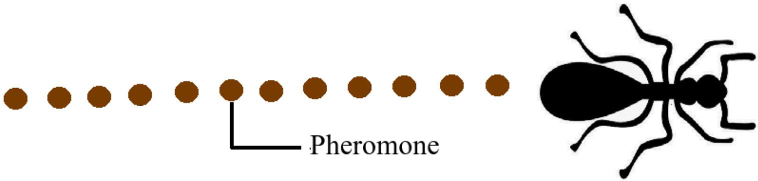
So how do they communicate??

Pheromone



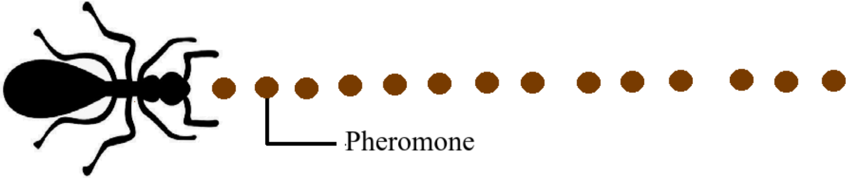
So how do they communicate??

Pheromone



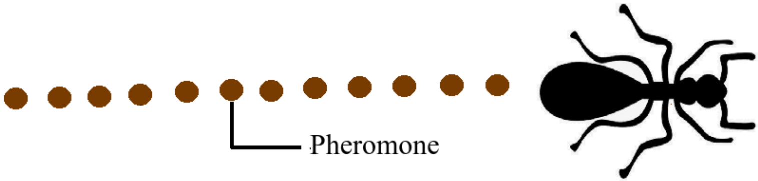
So how do they communicate??

Pheromone



So how do they communicate??

Pheromone



Previous Works

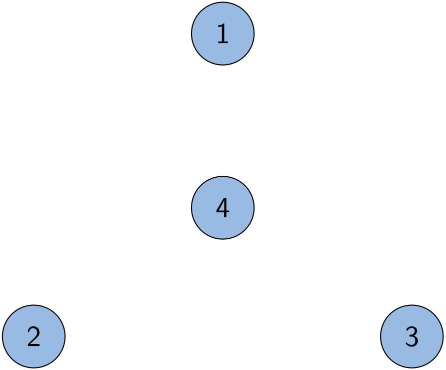
- In the year 1991, Marco Dorigo proposed an algorithm called "Ant System".
- AS was first applied to the Traveling Salesman Problem.

Figure: Marco Dorigo

Results

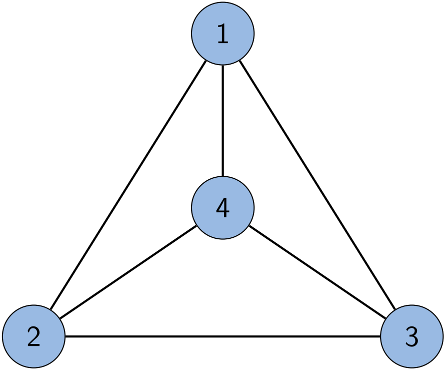
A more efficient algorithm to solve TSP is the ACO meta-heuristic.

Notations



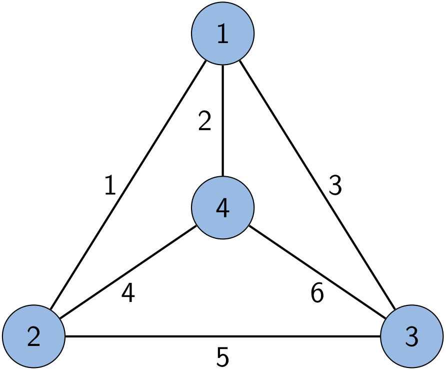
$C = \{c_1, c_2, ..., c_{N_C}\}$ is the set of *cities*.

Notations Continued..



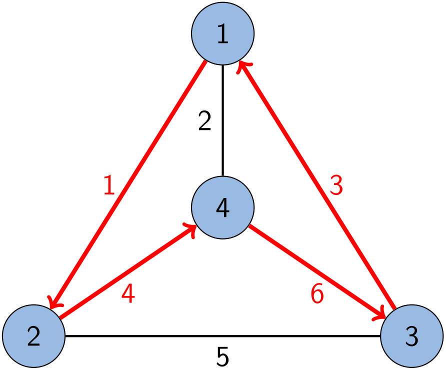
$L = \{l_{c_i c_j} \mid (c_i, c_j) \in \tilde{C}\}, \; |L| \leq N_C^2$ is the set of *connections* between *cities*.

Notations Continued..



$J_{c_i c_j} \equiv J(l_{c_i c_j})$ is a *cost* function associated with each *connection* $l_{c_i c_j} \in L$.

Notations Continued..

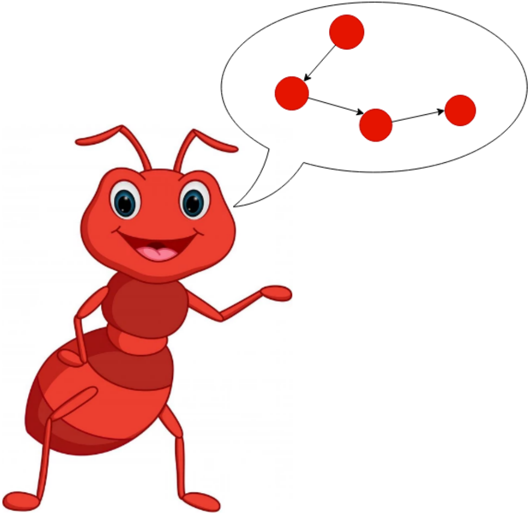


$J_\psi(L)$ is the *cost function* associated to each solution ψ. It is the summation of all the costs $J(c_i, c_j)$ of all the connections belonging to the solution ψ .

The optimal cost here is $1 + 4 + 6 + 3 = 14$

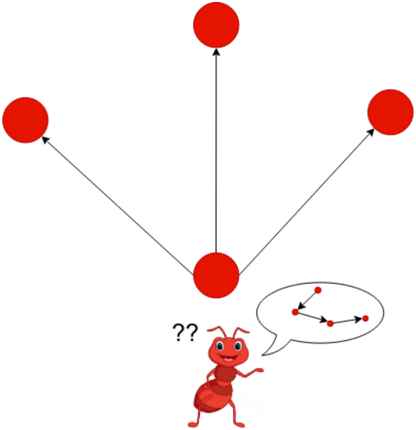
Ant Properties

- Every ant has its own memory.



Ant Properties Continued

- An ant chooses the next node to visit from its memory and the ant-routing table



Ant Properties Continued

$$\begin{pmatrix} 0 & 5 & 3 & 1 \\ 4 & 0 & 1 & 6 \\ 9 & 3 & 0 & 9 \\ 12 & 4 & 15 & 0 \end{pmatrix}$$

Ant-routing table, a

Here, a_{ij} is a measurement of the quality of the edge from node i to node j.

Formula for ant-routing table

The formula for updating the ant-routing table is:

$$a_{ij} = \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in \mathcal{N}_i} [\tau_{il}(t)]^{\alpha}[\eta_{il}]^{\beta}} \qquad \forall j \in \mathcal{N}_i$$

- τ_{ij} is the intensity of pheromone trail of the edge l_{ij}
- η_{ij} is the heuristic value of the edge between i and j.

$$\eta_{ij} = \frac{1}{J_{c_i c_j}}$$

- α and β are two parameters that control the relative weight of pheromone trail and heuristic value.

Formula for ant-routing table Continued

The probability $p_{ij}^k(t)$ with which an ant k located in city i chooses the city $j \in \mathcal{N}_i^k$ to move to at the t-th iteration is:

$$p_{ij}^k(t) = \frac{a_{ij}(t)}{\displaystyle\sum_{l \in \mathcal{N}_i^k} a_{il}(t)}$$

where $\mathcal{N}_i^k \subseteq \mathcal{N}_i$ is the feasible neighborhood of node i for ant k.

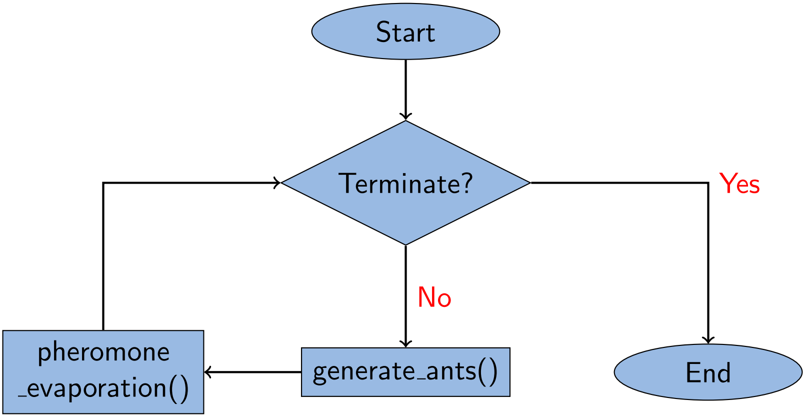
Pheromone Trail Evaporation

After pheromone updating has been performed by the ants, pheromone evaporation is triggered: the following rule is applied to all the edges l_{ij} of the graph G

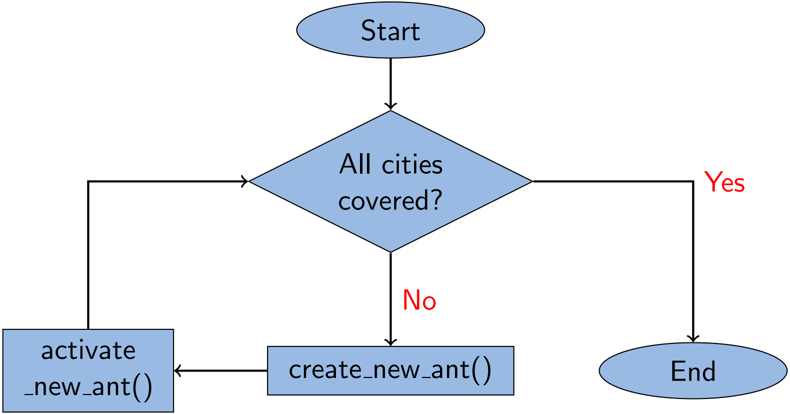
$$\tau_{ij}(t) \leftarrow (1-\rho)\tau_{ij}(t)$$

where $\rho \in (0,1]$ is the pheromone trail decay coefficient.

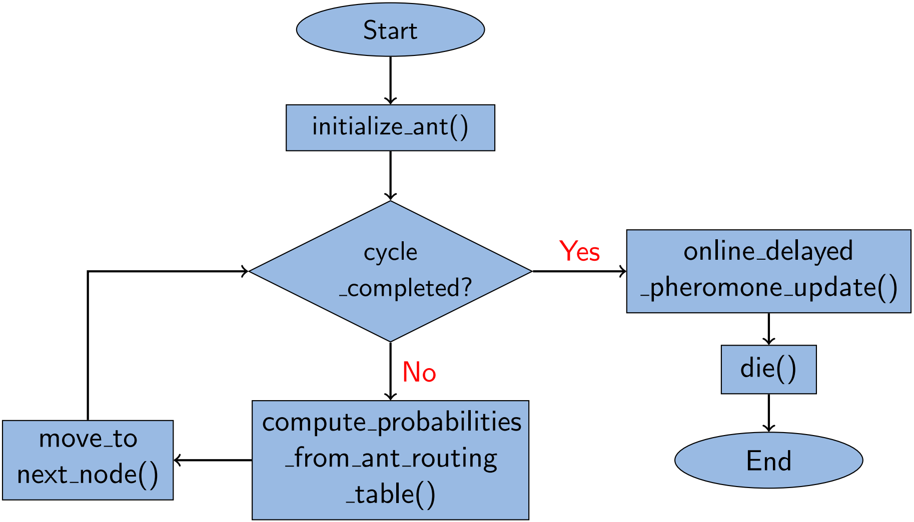
procedure ACO_meta-heuristic()



procedure generate_ants()



procedure activate_new_ant() {Ant lifecycle}



Conclusions

ACO is a new weapon to attack NP-hard graph problem!!