Ant Colony Optimization

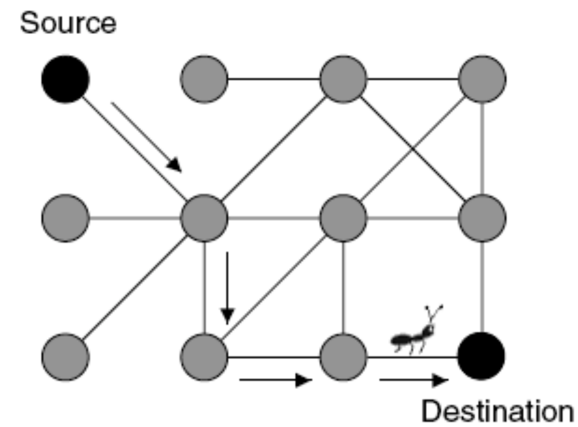
March 6 2012

Melanie Moses, Assistant Professor, Computer Science
University of New Mexico

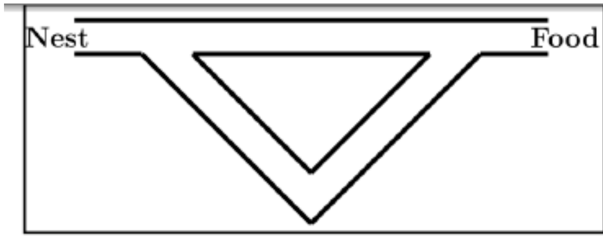
Key Concepts From Dorigo's ACO

- Ant algorithms use 'self-organizing principles' to coordinate agents to solve computational problems
- Stigmergy: indirect communication and coordination through signals that modify the environment and stimulate other agents
- Pheromones: a chemical signal that triggers a response in another agent
  - Pheromone concentration increases the probability that an ant will follow a path
  - Evaporation removes stale solutions and mitigates premature convergence

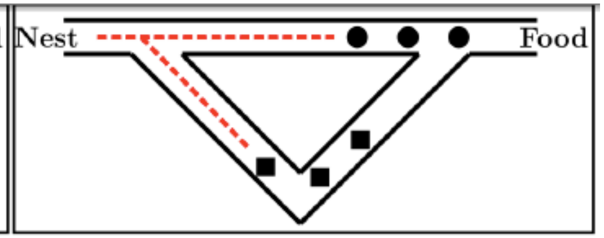
Simple-ACO



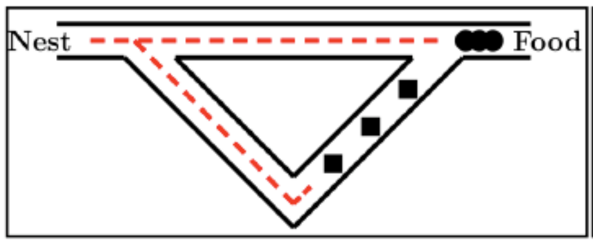
- Ants remember their paths
- Only backward pheromone deposition
- Deterministic backward path
- Pheromone evaporation
- Pheromone deposition rate depends on quality of solution (ants deposit more pheromone on shorter paths)
- Loop avoidance



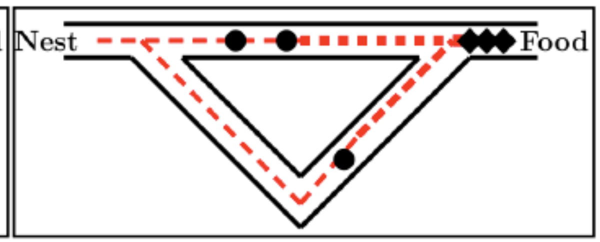
(a) All ants are in the nest. There is no pheromone in the environment.



(b) The foraging starts. In probability, 50% of the ants take the short path (symbolized by circles), and 50% take the long path to the food source (symbolized by rhombs).



(c) The ants that have taken the short path have arrived earlier at the food source. Therefore, when returning, the probability to take again the short path is higher.



(d) The pheromone trail on the short path receives, in probability, a stronger reinforcement, and the probability to take this path grows. Finally, due to the evaporation of the pheromone on the long path, the whole colony will, in probability, use

$$G = (V, E)$$

$$vertices: v_s, v_d$$

$$edges: e_1, e_2$$

$$p_i = \frac{\tau_i}{\tau_1 + \tau_2}, i = 1,2$$

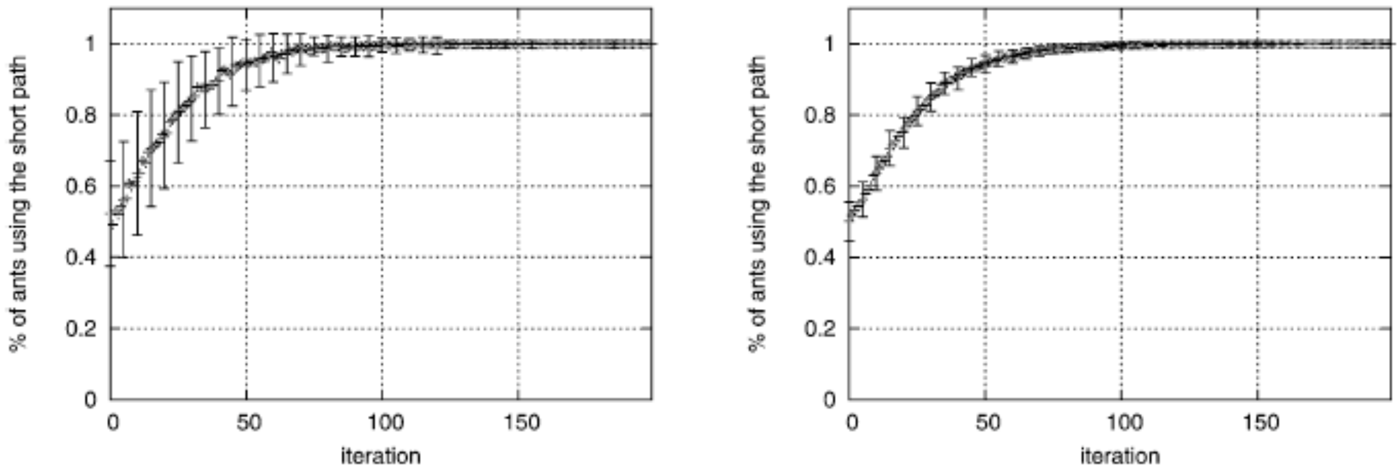
PheromoneUpdate

$$\tau_i \leftarrow \tau_i + \frac{Q}{l_i}$$

PheromoneEvaporation

$$\tau_i \leftarrow (1 - \rho)\tau_i$$

$$\rho \in (0,1]$$



(a) Colony size: 10 ants

(b) Colony size: 100 ants

Fig. 2. Results of 100 independent runs (error bars show the standard deviation for each 5th iteration). The x-axis shows the iterations, and the y-axis the percentage of the ants using the short path.

Differences between real and artificial ants

1) Real ants are asynchronous
2) Real ants lay pheromones both ways, ACO only on return (NOT TRUE!)
3) Real ants implicit, ACO explicit evaluation of path length (partially true)

ACO

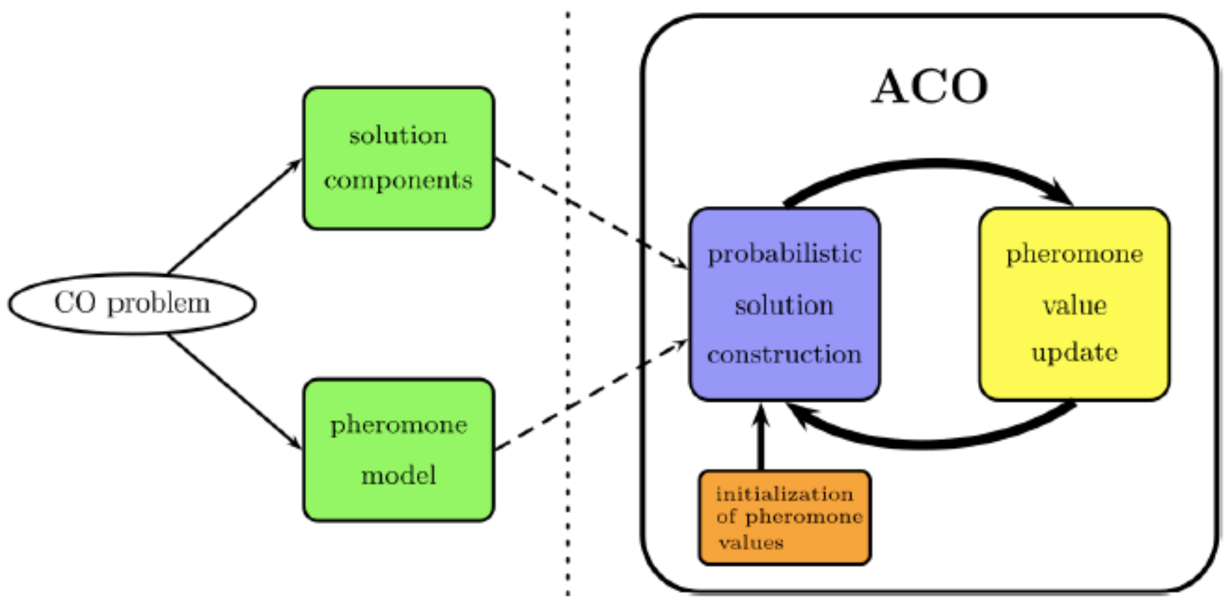
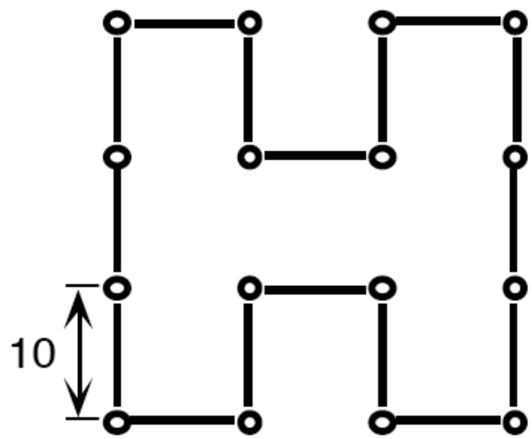


Fig. 4. The working of the ACO metaheuristic.

ACO for TSP



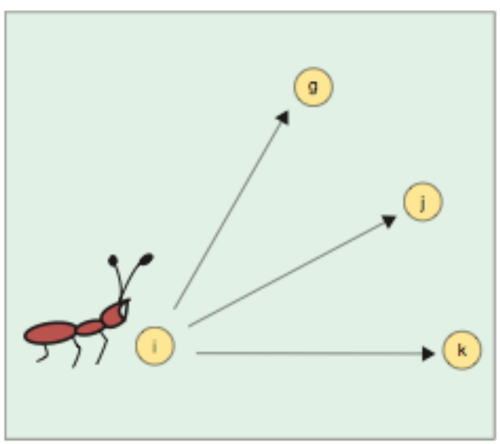
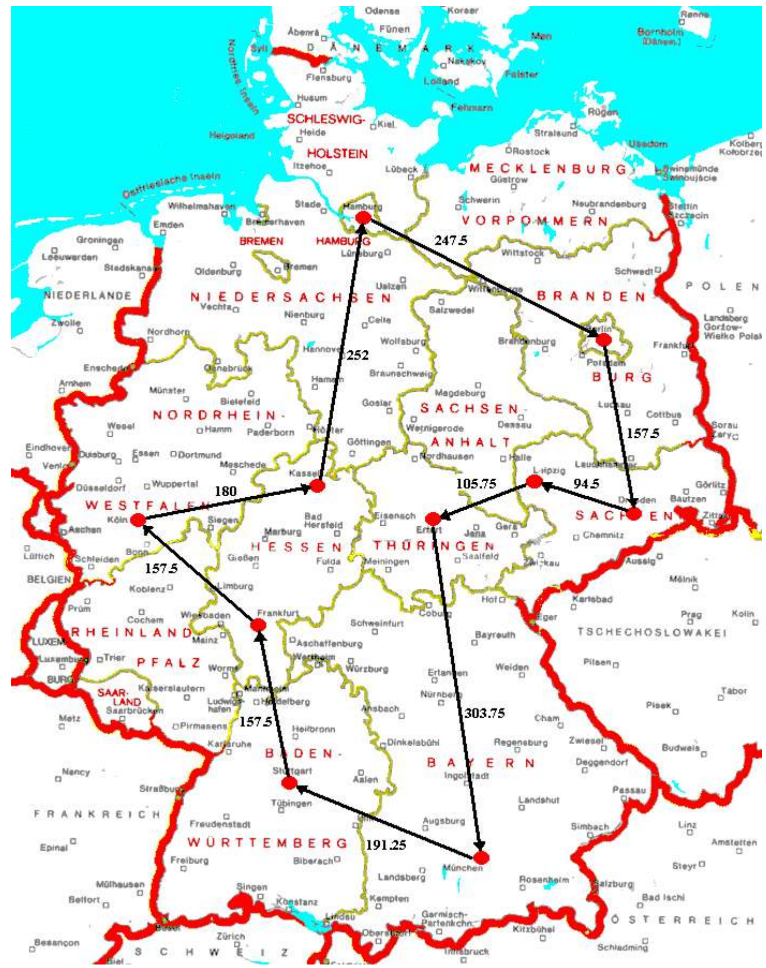


FIGURE 2 An ant in city *i* chooses the next city to visit via a stochastic mechanism: if *j* has not been previously visited, it can be selected with a probability that is proportional to the pheromone associated with edge (*i*, *j*).



Traveling Salesman Problem

TSP is an NP hard optimization problem

- Find the shortest tour through a set of cities back home, visiting each city exactly once.

G = (V,E) with V vertices and E edges

Each edge $(i,j) \in E$ has length d_{ij}.

Find π, a permutation of the node indices that minimizes f(π)

$$f(\pi) = \sum_{i=1}^{n-1} d_{\pi(i)+\pi(i+1)} + d_{\pi(n)\pi(i)}$$

e.g. $\pi = \{5,7,3,8,2,1\}$

ACO Metaheuristic

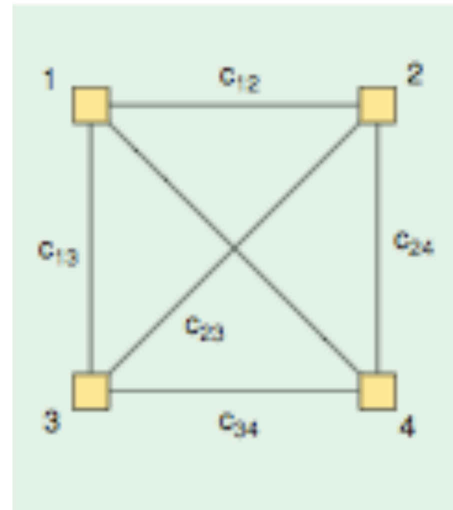
- A model $P = (S,\Omega,f)$ of a combinatorial optimization problem
  - Search space S is defined over a set of decision variables X_i, I = 1, …,n
  - A set Ω of constraints among variables
  - An objective function f: $S \rightarrow \mathbb{R}^+$ to be minimized

    For TSP, minimize tour length

X_i is assigned from a set of discrete values that satisfy Ω and minimizes f.

ACO Metaheuristic

- Ants build a construction graph of vertices and edges G(V,E)
- The solutions are constructed by depositing pheromone values on the edges.
- Solution components $c_{i,j}$ = (i,j) specify that city j should be visited after city i



Ant System

-Pheromone is stored in a matrix
-Heuristic information (distances between nodes) is stored in another matrix
-Ants remember where they've been on a given tour and do not repeat cities

```
Initialize Pheromone
While termination condition not met
    Construct Ant Solutions
        For each ant,
            choose a start city,
            construct a tour, biasing steps by pheromone, until it returns home
    Optionally Apply local search
    Update Pheromone
endwhile
```

Variations: elitist, rank based, max-min: alter pheromone deposition and update

Ant System

- Ant cycle: pheromone deposit is determined globally (not very ant like) based on the length of the tour
- Initialization:

Initialize pheromone and heuristic information for all i,j:

$$\tau_{i,j} = \frac{m}{C^{nn}}$$

$$\eta_{i,j} = \frac{1}{d_{i,j}}$$

m = # ants

C^{nn} = length of nearest neighbor CYCLE

$d_{i,j}$ = distance from i to j

We define the transition probability from town i to town j for the k-th ant as

- Tour construction formula

- What do alpha and beta represent?

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha} \cdot \left[\eta_{ij}\right]^{\beta}}{\sum\limits_{k \in allowed_k} \left[\tau_{ik}(t)\right]^{\alpha} \cdot \left[\eta_{ik}\right]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

- Pheromone update

$$\tau_{ij} \leftarrow \tau_{ij} + \Delta\tau^k.$$

$$\Delta\tau^k_{i,j} = \frac{1}{C^k}$$

C^k is tour length of k^{th} ant
If (i,j) are on the tour of the k^{th} ant
0 otherwise

- Pheromone evaporation

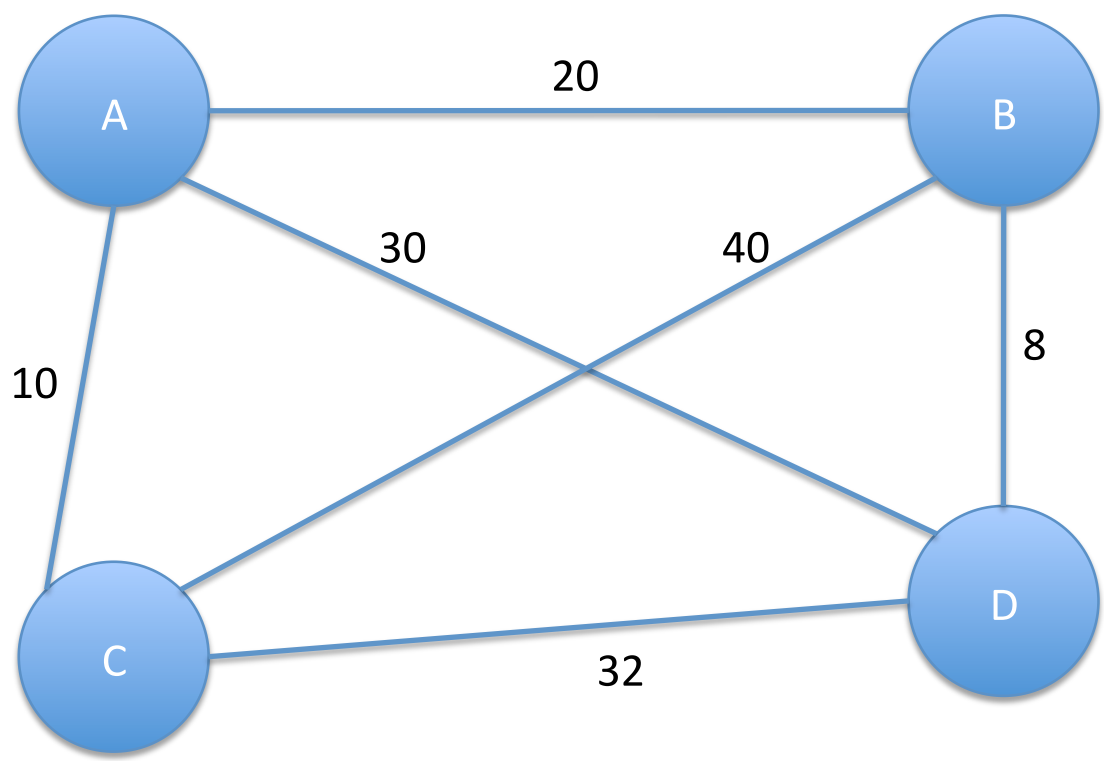
$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij}, \quad \forall (i,j) \in A,$$

where $\rho \in (0,1]$ is a parameter.

- AS parameter settings:
  - alpha = 1
  - beta = 2 to 5
  - rho = 0.5
  - m = n (number of ants = number of cities)
  - Tau_0 initialization = m/C^{nn}
- Ant Cycle: pheromone update depends on tour length,
  So it is updated only after a completed tour
- Elistist AS: Reinforce T^{bs} (best so far tour)

$$\tau_{i,j} = \tau_{i,j} + \sum_{k=1}^{m} \Delta\tau^k_{i,j} + e\Delta\tau^{bs}_{i,j}$$

$$\Delta\tau^{bs}_{i,j} = \frac{1}{C^{bs}}$$


A
B
C
D
20
30
40
10
8
32

Initialize

$$\tau_{i,j} = \frac{m}{C^{nn}}$$

$$\eta_{i,j} = \frac{1}{d_{i,j}}$$

C^{nn} = length of nearest neighbor CYCLE

$d_{i,j}$ = distance from i to j

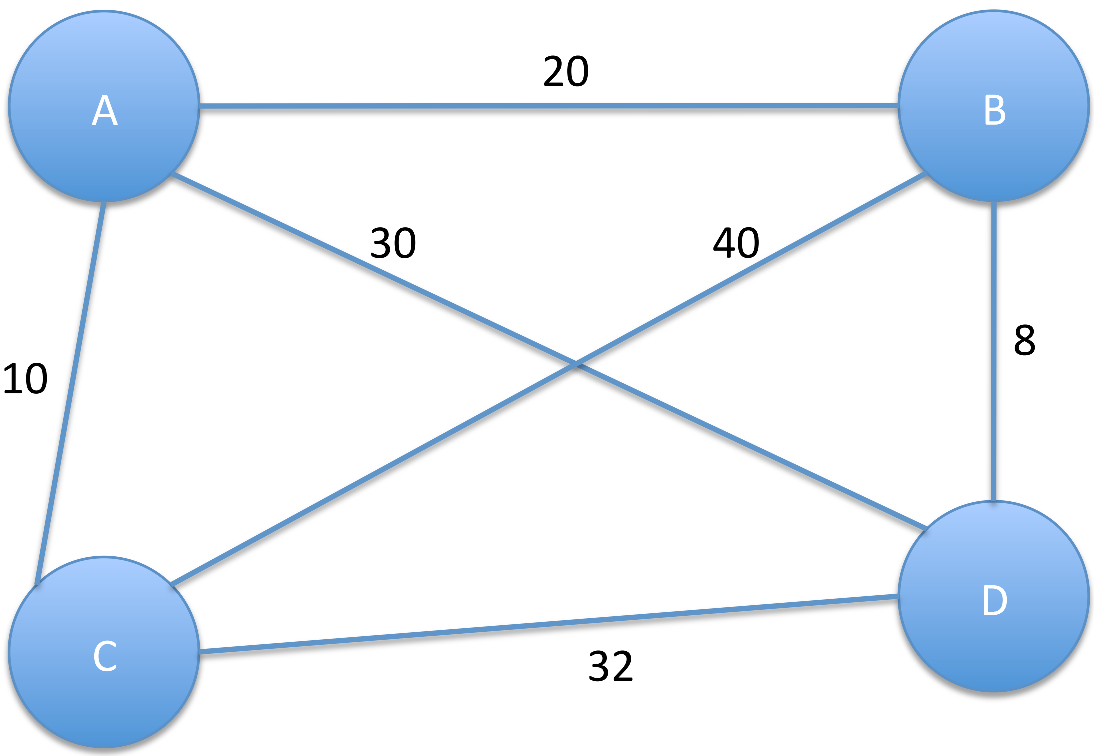
Start at A

C^{nn} = 10 + 32 + 8 + 20 = 70

A, C, D, B

m = # cities = # ants = 4

All τ = 4/70



Initialize

$$\tau_{i,j} = \frac{m}{C^{nn}}$$

$$\eta_{i,j} = \frac{1}{d_{i,j}}$$

C^{nn} = length of nearest neighbor CYCLE

$d_{i,j}$ = distance from i to j

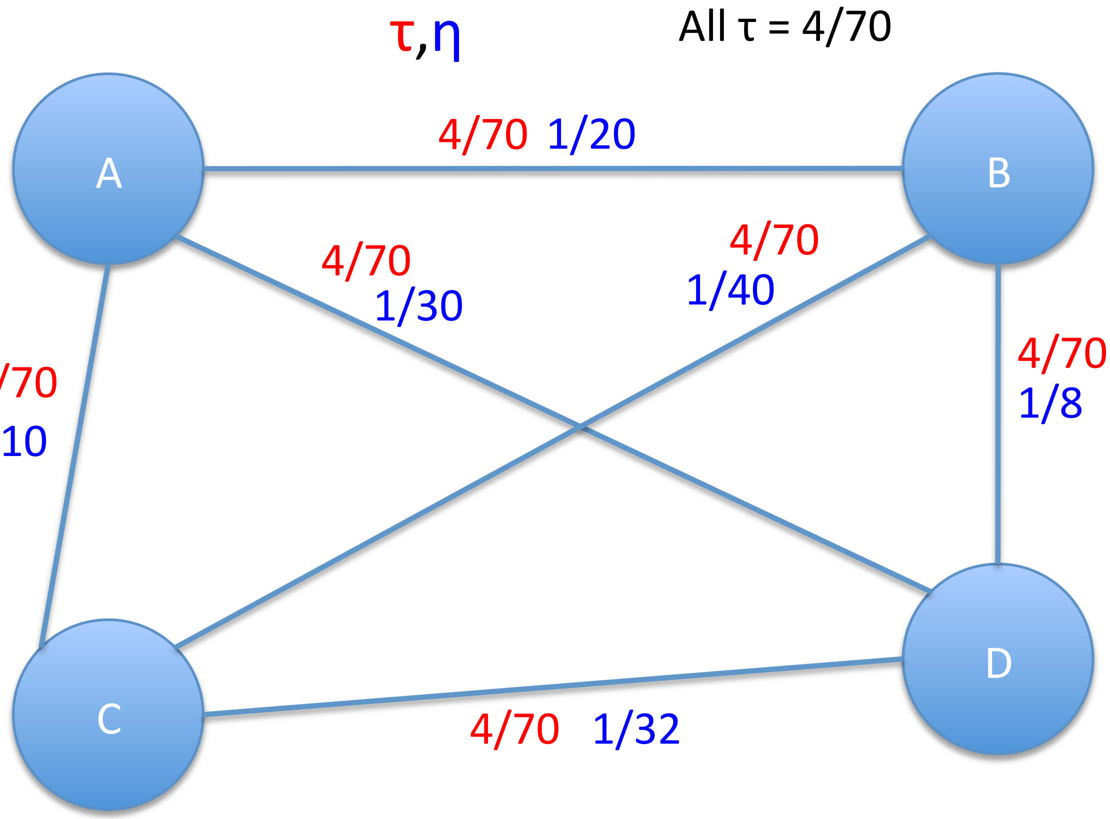
Start at A

C^{nn} = 10 + 32 + 8 + 20 = 70

A, C, D, B

m = # cities = # ants = 4

All τ = 4/70



We define the transition probability from town i to town j for the k-th ant as

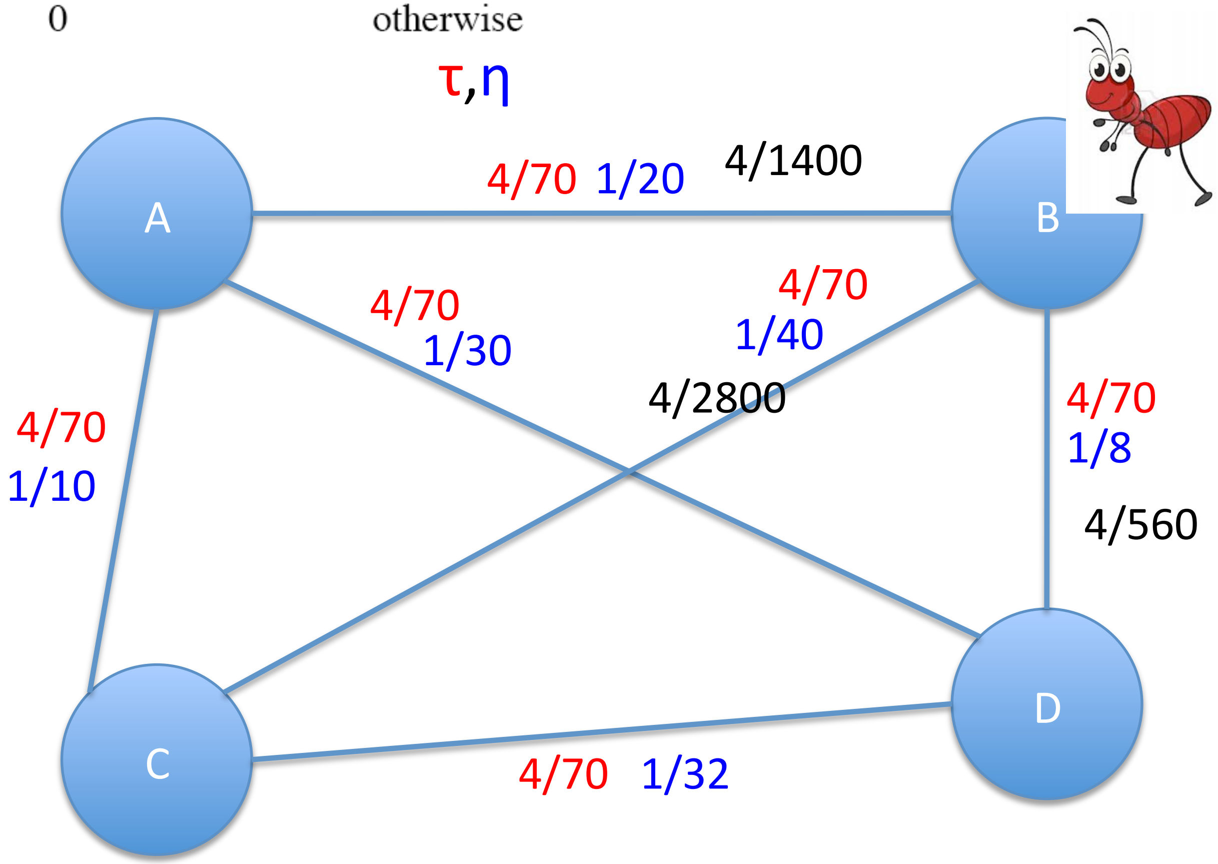
TRAVERSE

$$p_{ij}^k(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

denominator=(8+4+20)/2800

α = 1

β = 1 (usually 2 – 5)



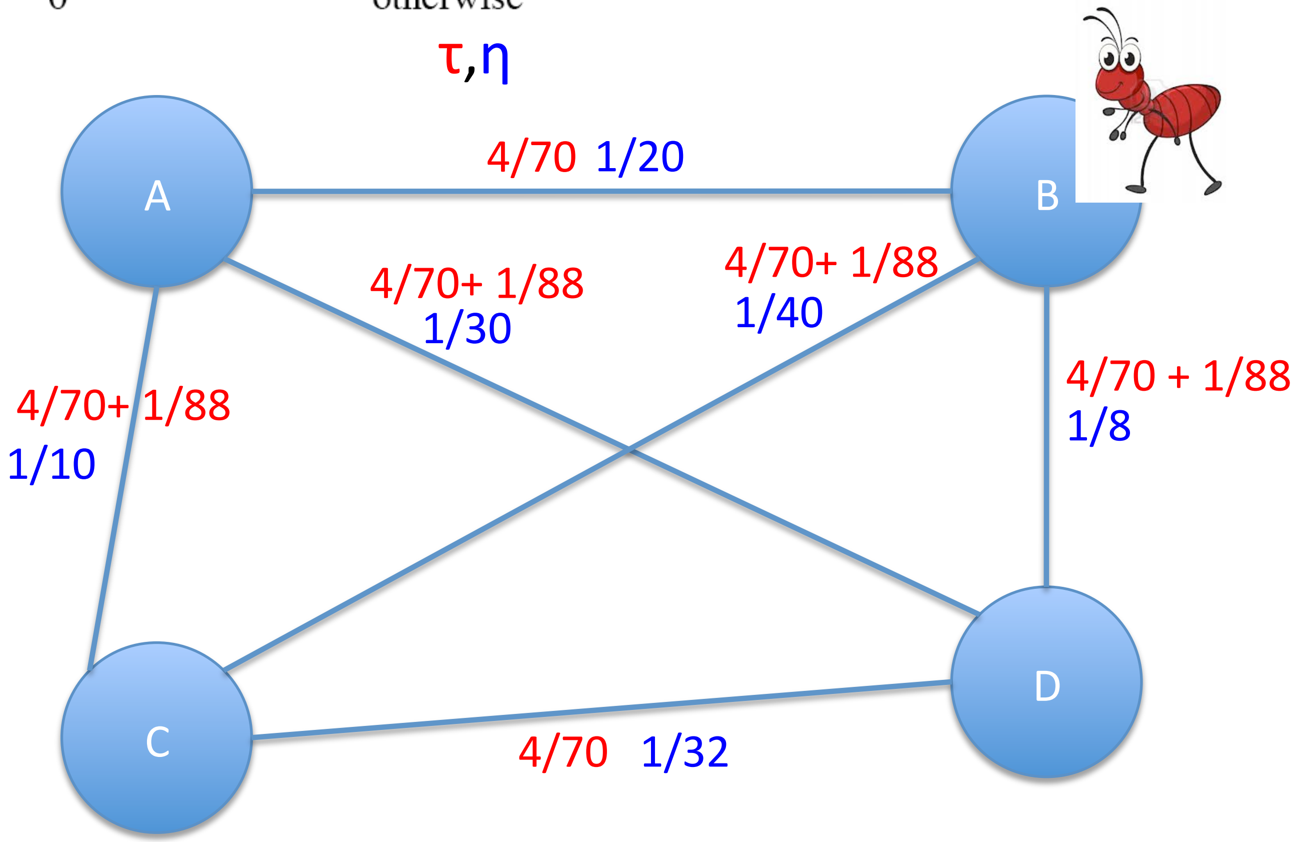
Usually : B, D, A, C, B: Cycle = 8+30+10 +40 = 88

Rarely: B,D,C,A, B: Cycle = 8+32+10 + 20 = 70

We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

UPDATE Pheromone



Usually : B, D, A, C, B: Cycle = 8+30+10 +40 = 88

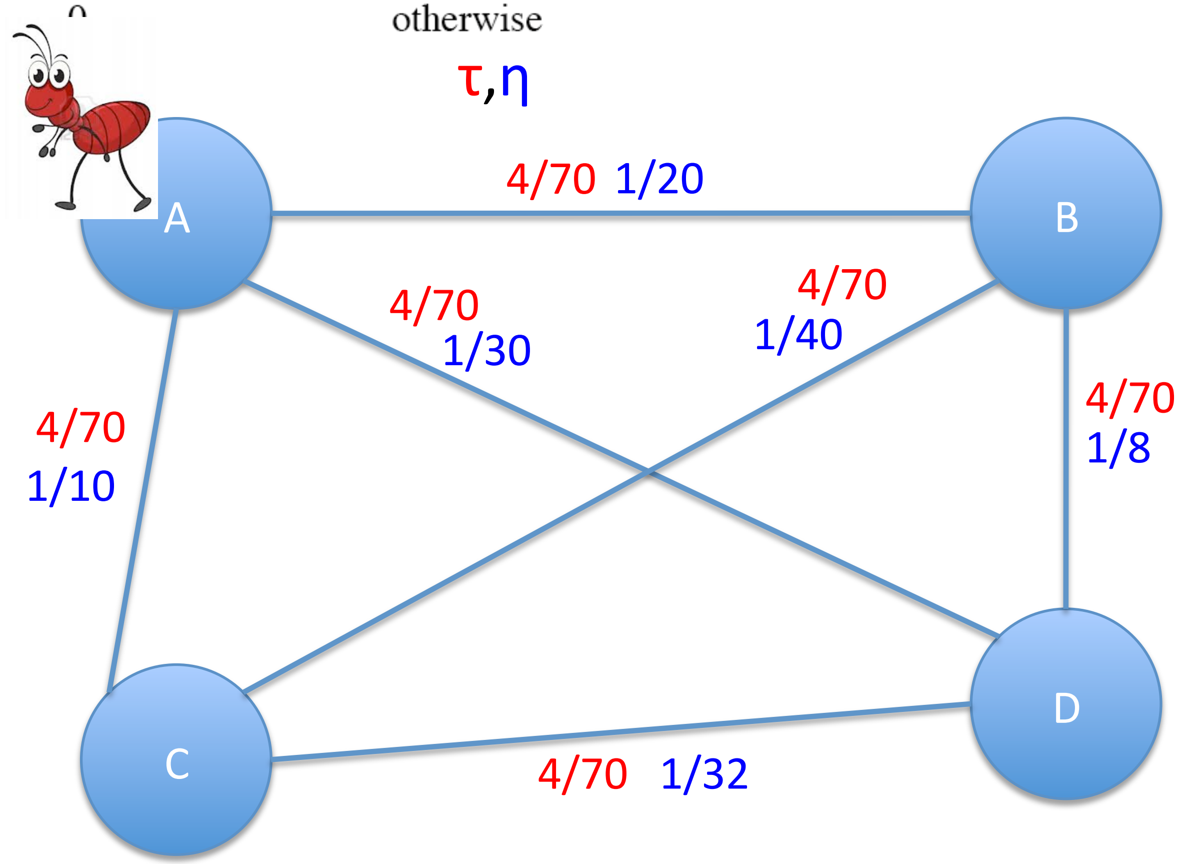
Rarely: B,D,C,A, B: Cycle = 8+32+10 + 20 = 70

We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

$\alpha = 1$

$\beta = 1$

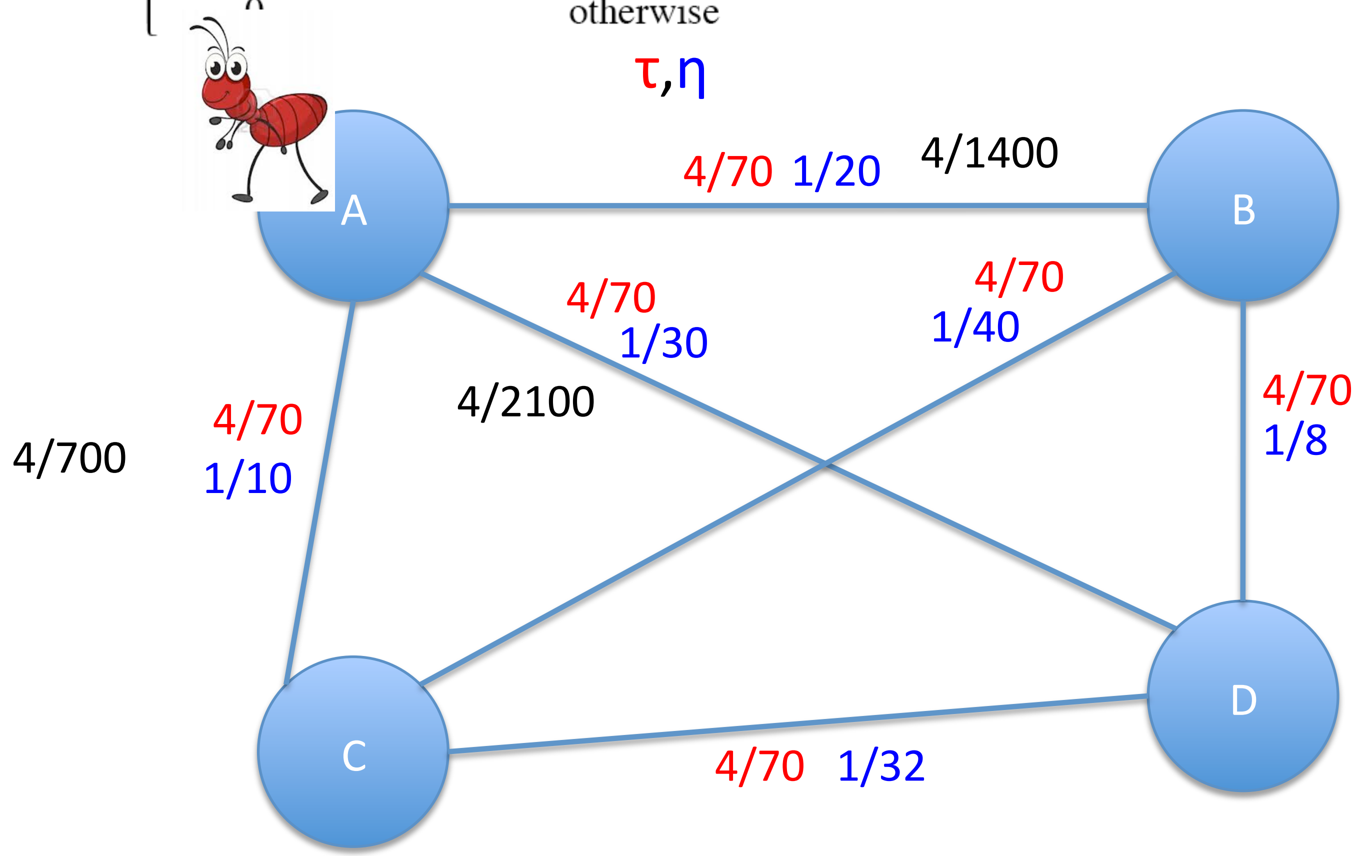


We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

α = 1

β = 1 (usually 2 – 5)



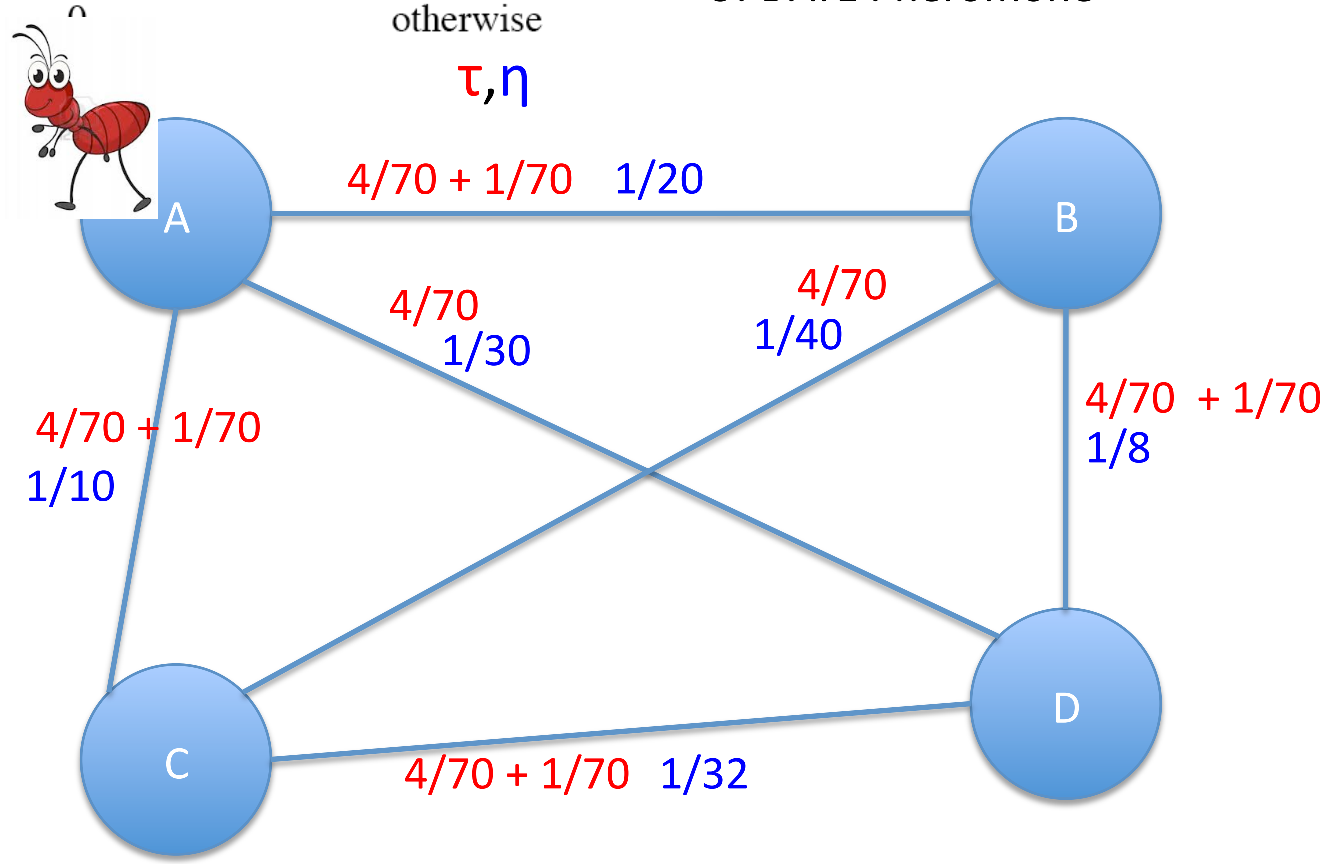
Usually : A, C, D, B,A Cycle = 10+32+8 +20 =70

Very Rarely: A,D, C, B, A Cycle = 30+32+40+20 = 122

We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha} \cdot \left[\eta_{ij}\right]^{\beta}}{\sum\limits_{k \in allowed_k} \left[\tau_{ik}(t)\right]^{\alpha} \cdot \left[\eta_{ik}\right]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

UPDATE Pheromone



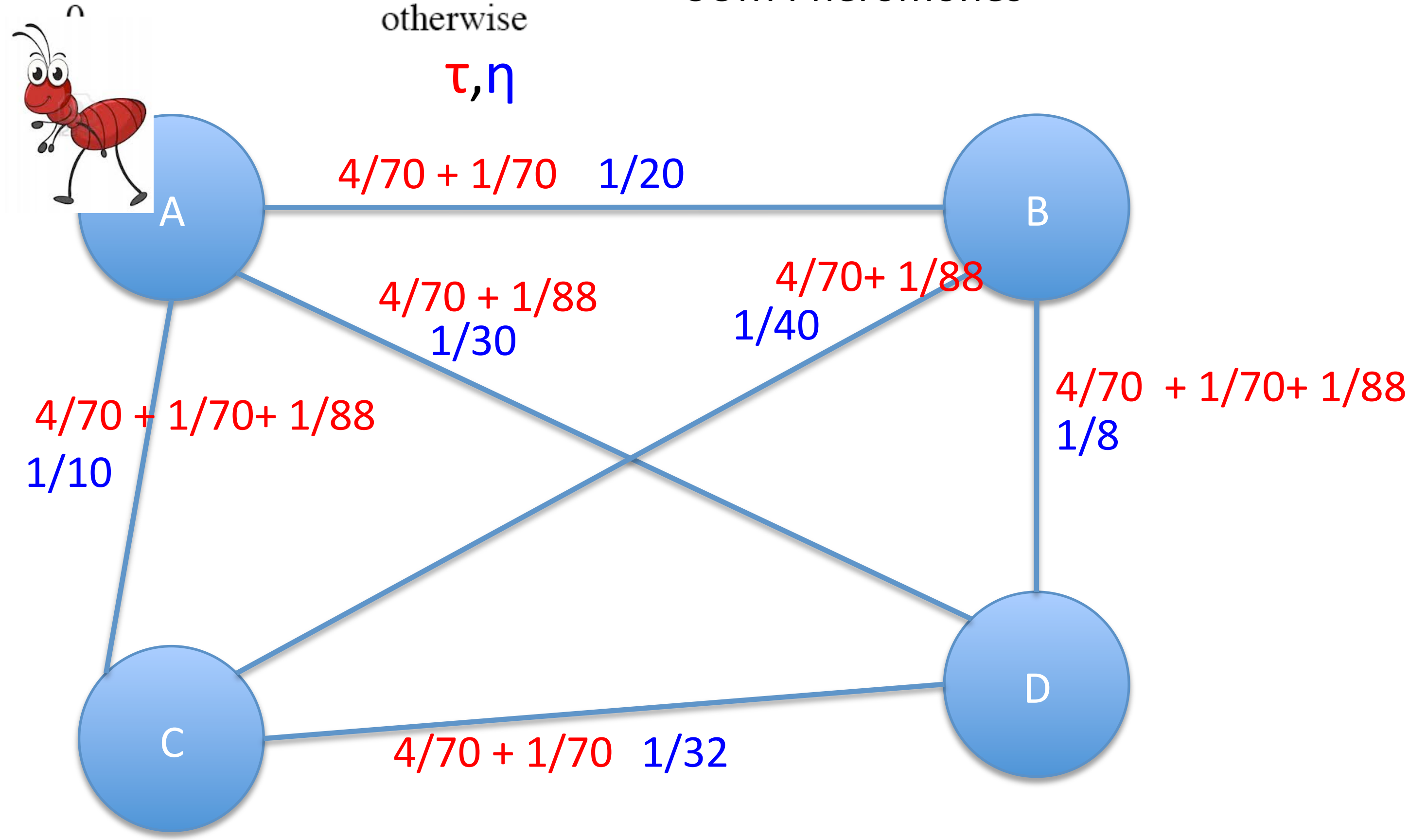
Usually : A, C, D, B, A Cycle = 10+32+8 +20 =70

Rarely: A,D, B, C, A Cycle = 30+8+40+10 = 88

We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

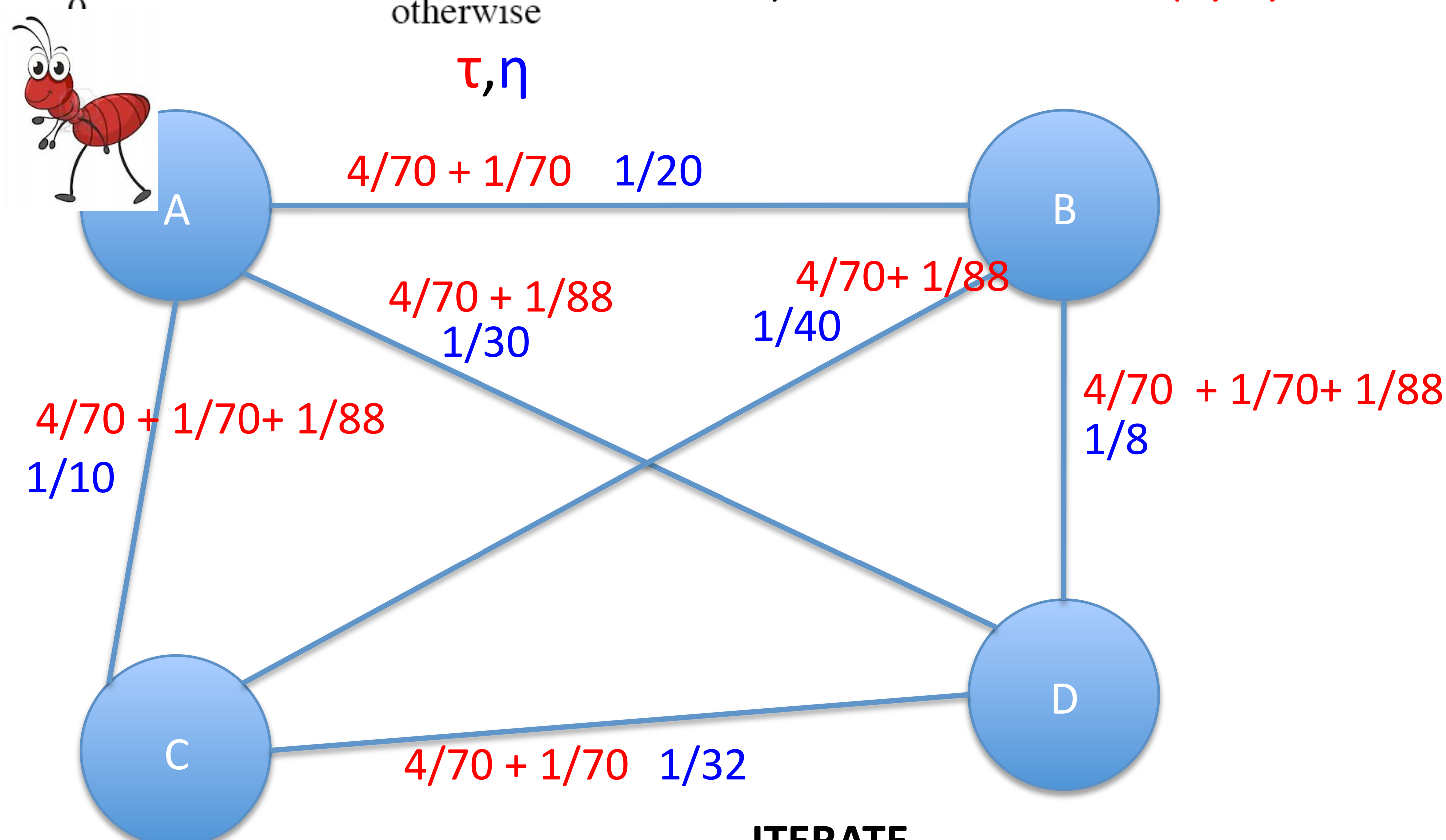
SUM Pheromones



We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases}$$

Evap Pheromones: multiply by 0.5



ITERATE

η doesn't change

τ varies as ants choose and assess tours

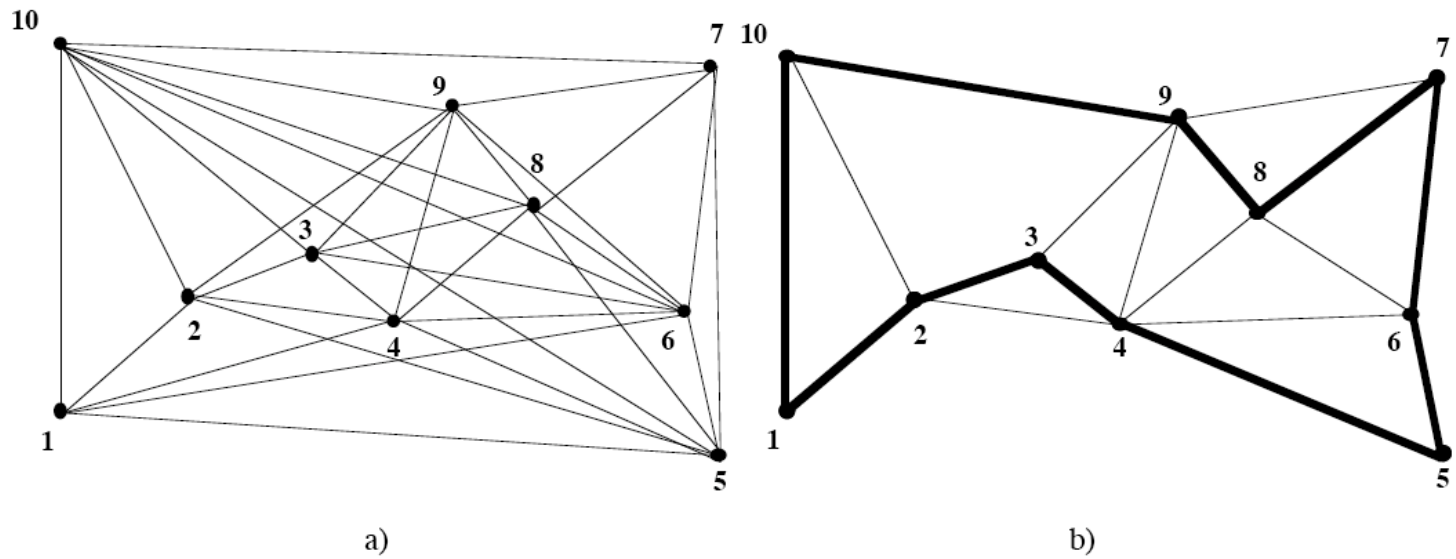


Fig. 6. Evolution of trail distribution for the CCA0 problem.
a) Trail distribution at the beginning of search.
b) Trail distribution after 100 cycles.

Dorigo 1996

Variations

- Min Max Ant System (Eqns. 6 -8)
  Only the best ant lays pheromones
  (ant B would not add +1/80)
  best: in iteration or best ant so far
  pheromone value is bounded

- Ant Colony System adds a local pheromone update (Eqn. 9)
  ants remove pheromone as they go to encourage subsequent ants to explore other edges

- Add local search

- The complexity of the *ant-cycle algorithm is* $O(NC*n^2*m)$
  - *NC = Number of Cycles (completed tours)*
  - *n = number of cities*
  - *m = number of ants*
- Experimentally AS works best when m = n
- Complexity is $O(NC*n^3)$
- Note: global communication in AntCycle restricts parallelization
  - Lamarkian pheromones vs Darwinian pheromones
- Why does it work?
  - Reduces the size of the search space (focus the search)
  - How quickly do tours converge or stagnate?

- Pheromone bias vs pheromone evaporation (Alpha vs rho)?