

Computational Intelligence

Samaneh Hosseini

Isfahan University of Technology

Ant Colony Optimization

Ants: An Introduction

Inspiration From Nature



1 quadrillion ants
7 billion humans

Ants

- There are about 10^{18} living insects
- About 2% of all insects are social
- About 50% of all social insects are ants
- The total weight of ants is about the total weight of humans
- Ant colony size can be as few as 30 ants and as large as million ants
- Ants colonize the world since 100,000,000 years, humans only since 50,000 years

Ants

- Ants use Stigmergy that was discovered by Pierre-Paul Grasse who was studying termites.
- Properties of Stigmergy
 - • indirect agent interaction modification of the environment
 - • environmental modification serves as external memory
 - • work can be continued by any individual
- Ants walking, to or from, a food source deposit a chemical substance on its way,
 - This substance is referred to as the **pheromone**.



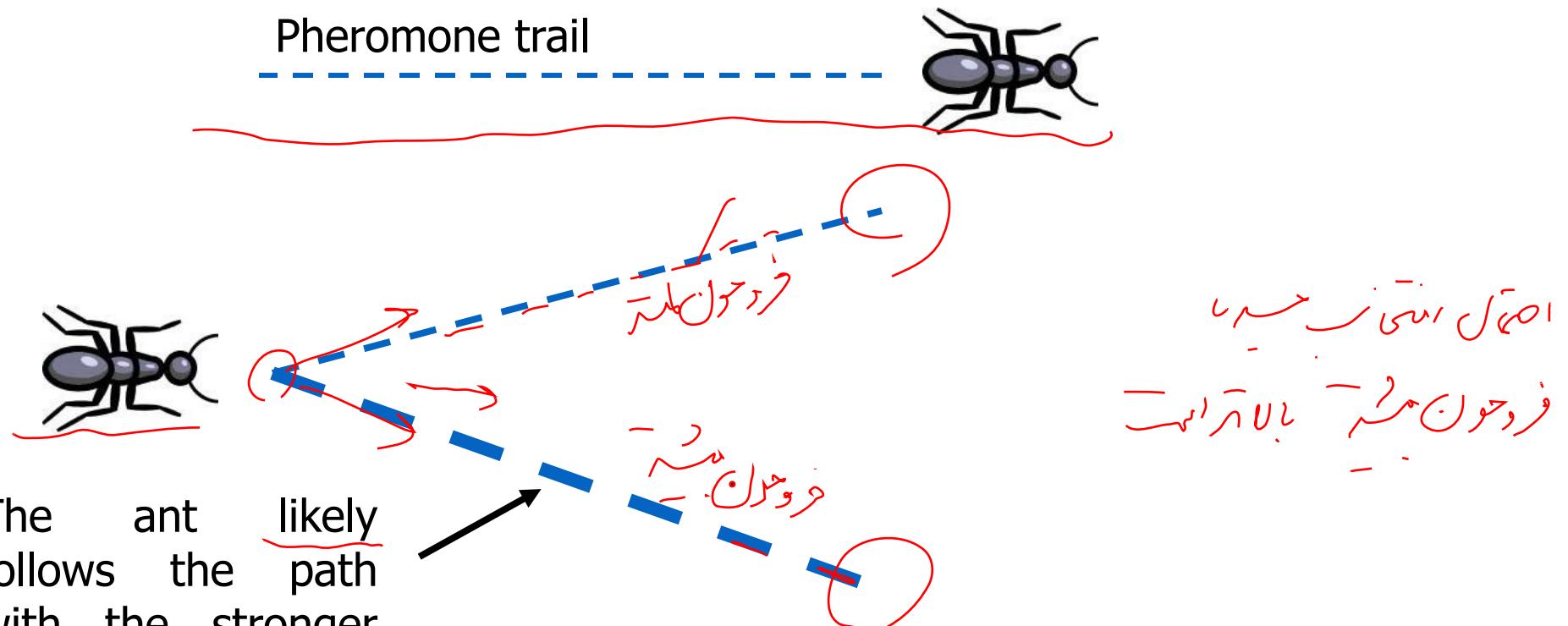
- P. P. Grasse. "Recherches sur la biologie des termites champignonnistes (*Macrotermes*)". Ann. Sc. Nat., Zool. Biol. anim., 6, 97, 1944.

- P. P. Grasse. "La reconstruction du nid et les coordinations interindividuelles chez *bellicositermes natalensis* et *cubitermes* sp. La theorie de la stigmergie: essai d'interpretation du comportement des termites constructeurs". Insectes Sociaux, 6, 41, 1959.

Pheromone

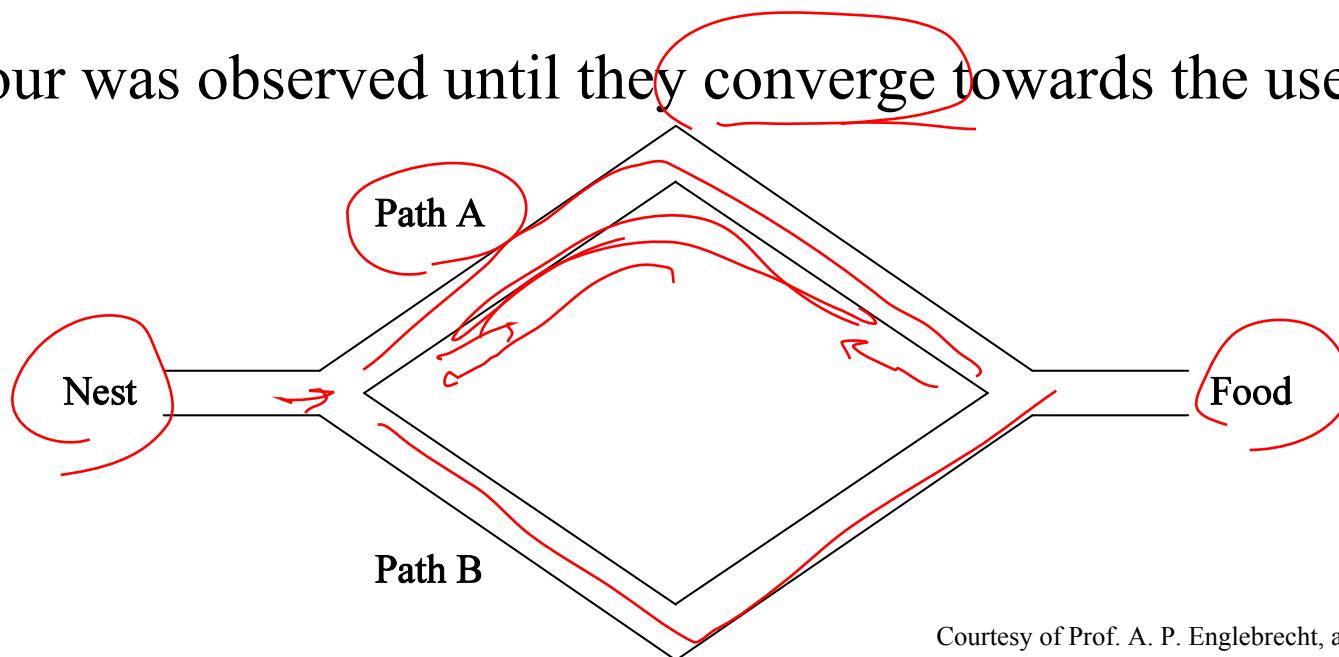
- Other ants can smell this pheromone,
- Its presence will influence the ants' choice of a certain path,
- Ants tend to follow strong pheromone concentrations.
- Pheromone trails, is formed by the pheromone deposited on the ground,
- Pheromone trails help the ants to reach good food sources that have been previously identified by other ants.

Pheromone



Binary Bridge Experiment

- A *binary bridge experiment* was done by Deneubourg et al.,
- The ants nest was connected to a food source through two equal length bridges,
- The ants behaviour was observed until they converge towards the use of the same bridge.



Courtesy of Prof. A. P. Englebrecht, author of text

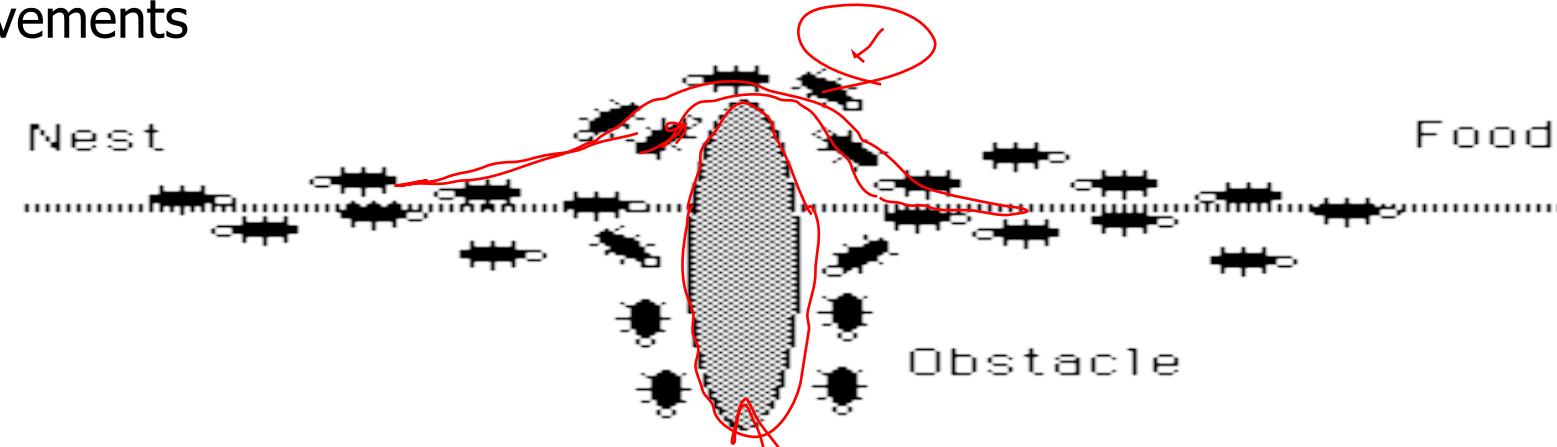
J. L. Deneubourg, S. Aron, S. Goss and J. M. Pasteels. "The self- organizing exploratory pattern of the Argentine ant". Journal of Insect Behaviour, 3, 159, 1990.

Binary Bridge Experiment

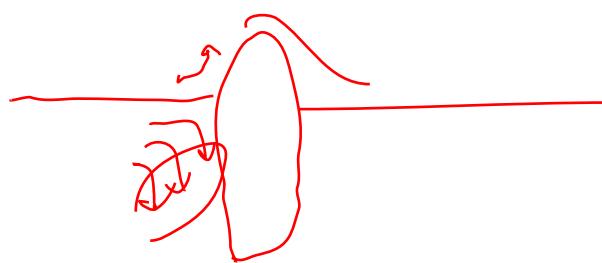
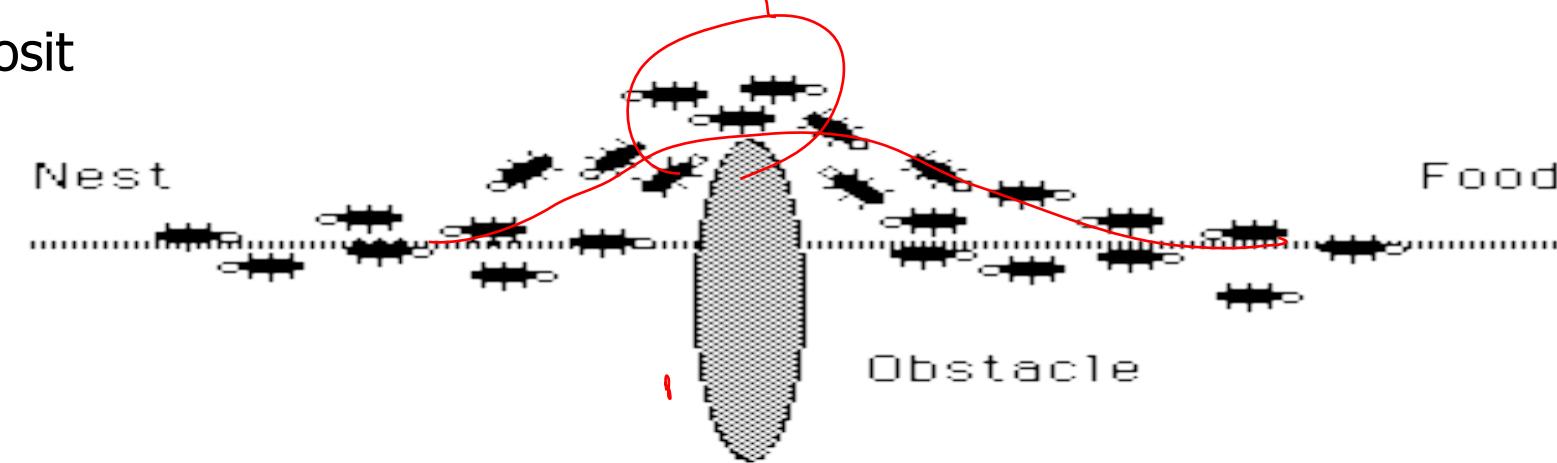
- Initially, the ants randomly choose the bridge to follow,
- After a while, the ants will tend to follow the bridge with *stronger pheromone trails*,
- The selection of one bridge is due to *random fluctuations* causing it to have higher pheromone concentration.
- Pheromone trail acts as a *collective memory* for the ants to communicate through by sensing and recording their foraging experience
- Pheromone *evaporates* over time introducing some changes in the environment.

Shortest Path Around an Obstacle

Initial movements

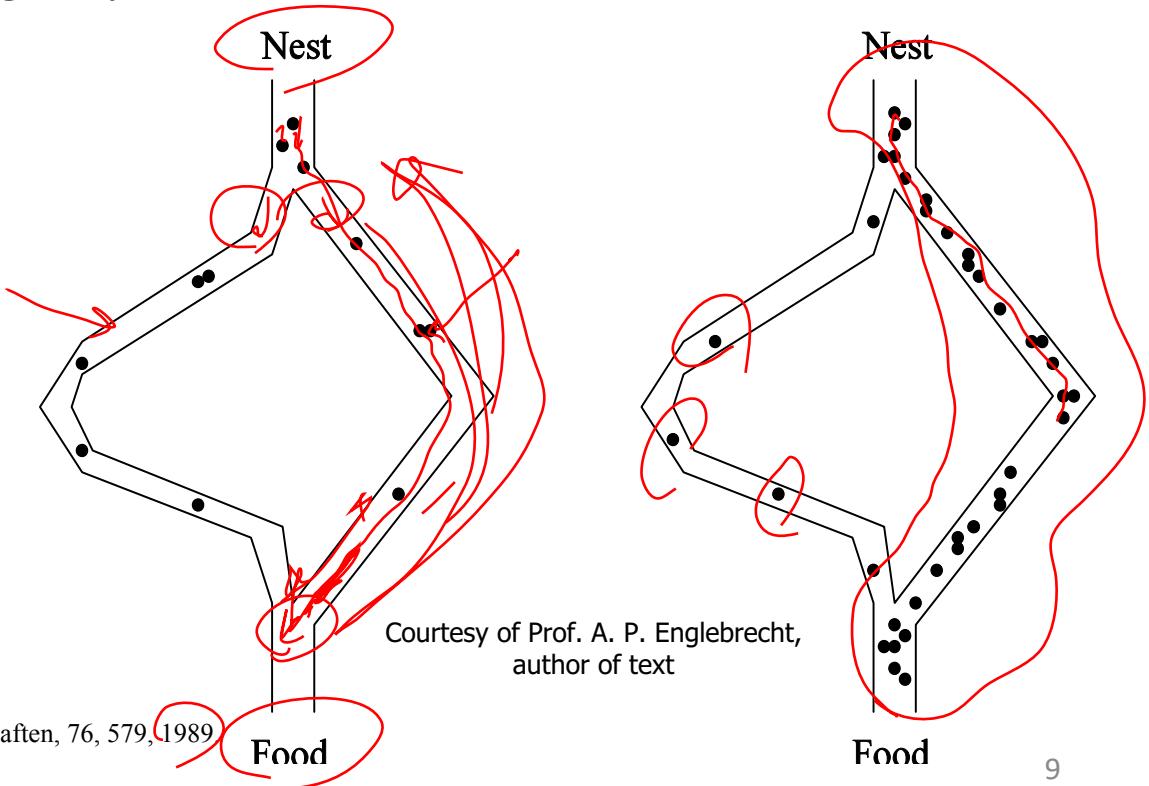


After deposit

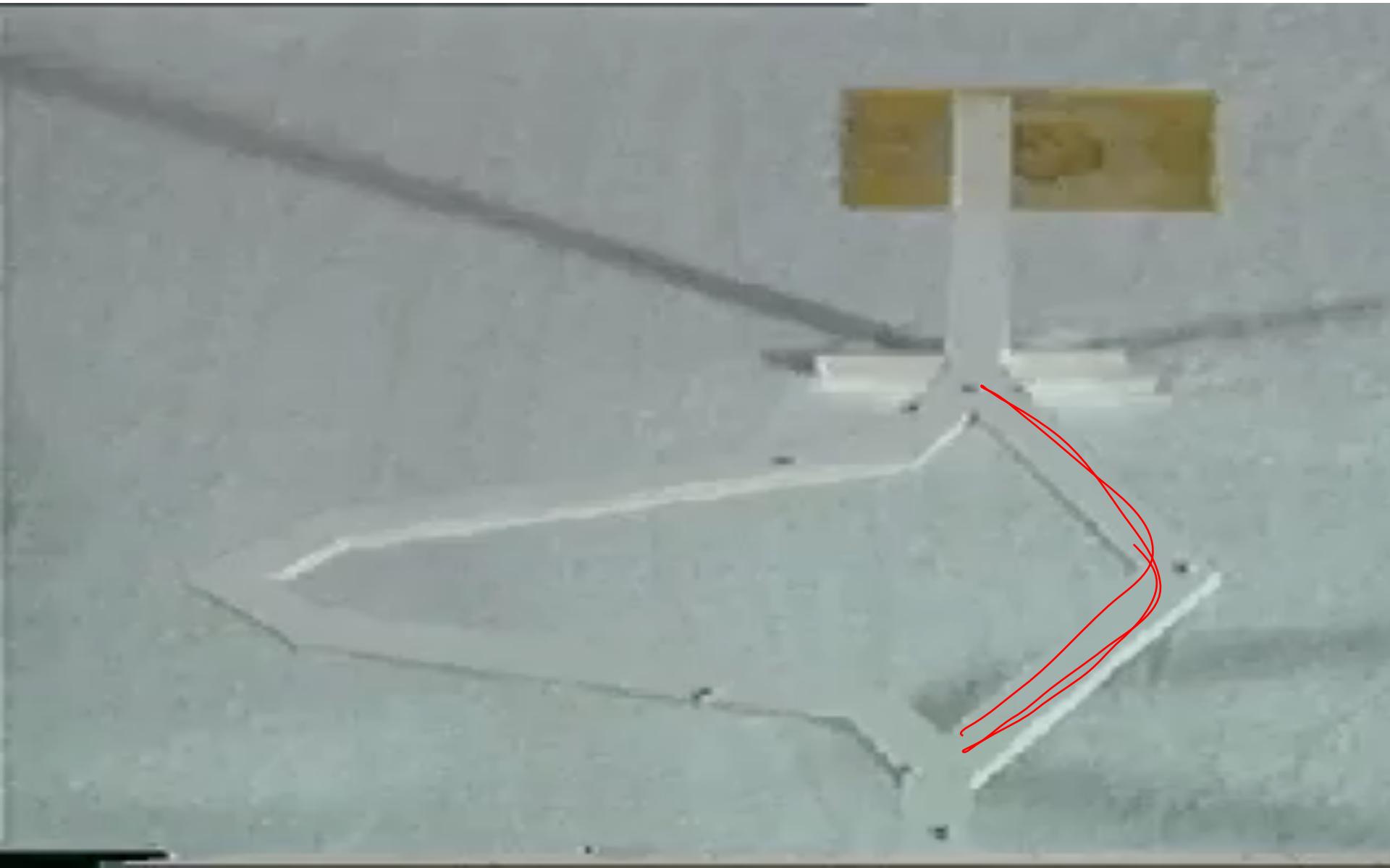


Bridges with Non-equal Length

- Another experiment was carried by Goss et al. [4] where the bridges were not of equal length. One bridge was significantly shorter than the other,
- In this case, the ants following the shorter bridge by chance will be the first to reach the food source.
- They will be also the first to reach the nest as they will take the same path home; since it will have more pheromone,
- The ants finally converged to the shorter path due to the pheromone depositing mechanism.



Bridges with Non-equal Length



⇒ Ant Colony Optimization^(ACO)
Algorithm

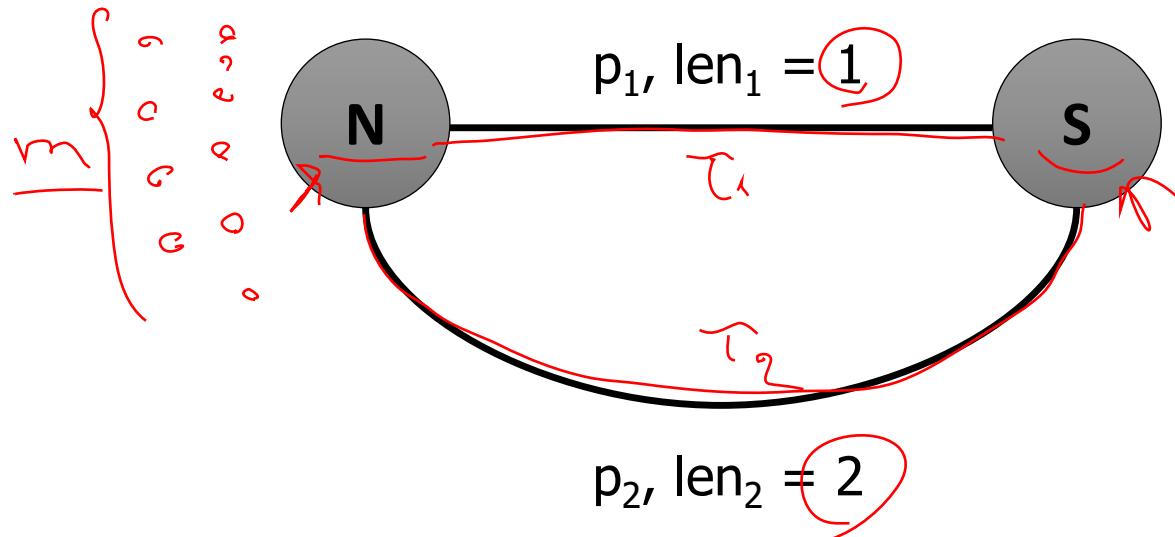
ACO
AS
Min-Max AS

ACO - Introduction

- Introduced by M. Dorigo in 1992,
- Inspired by the foraging behaviour of real ants,
- Successfully applied in many application including:
 - Traveling Salesman Problem (TSP),
 - Packet Routing in Network Telecommunications,
 - Scheduling Problems,
 - Vehicle Routing Problems.

ACO - Simulation

- To simulate the behaviour of ants: assume we have a nest and a food source connected through two paths with different lengths.



- Assign initial artificial pheromone values for every path, τ_1 and τ_2 ,
- Initially, both values are equal: $\tau_1 = \tau_2 = C > 0$

ACO - Simulation

- Place m ants at the nest,

- For each ant k ,

- Traverses path 1 with a probability:

$$\text{J} \xrightarrow{\tau_1} \text{J} \xrightarrow{\tau_2} \dots \quad p_k = \frac{\tau_1}{\tau_1 + \tau_2}$$

$$\frac{1}{2}$$

$$\frac{c}{c+c} = \frac{1}{2}$$

- Traverses path 2 with a probability:

$$\text{P} \xrightarrow{\tau_1} \text{P} \xrightarrow{\tau_2} \dots \quad p_k = 1 - p_k$$

ACO - Simulation

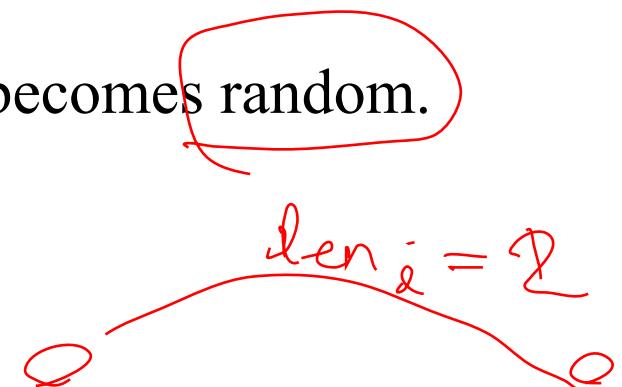
- An evaporation phase is applied:
 - To simulate the evaporation of real pheromone,
 - To avoid quick convergence to sub-optimal paths.

$$\tau_i = \underbrace{(1 - \rho)}_{\rho \in (0,1)} \times \underbrace{\tau_i}_{\text{initial pheromone}}$$

ρ specifies the rate of evaporation, when equal to 1, the move becomes random.

- Each ant leaves more pheromone on its traversed path:

$$\tau_i = \tau_i + \frac{1}{len_i}$$



Real AC vs. Artificial AC

Real AC

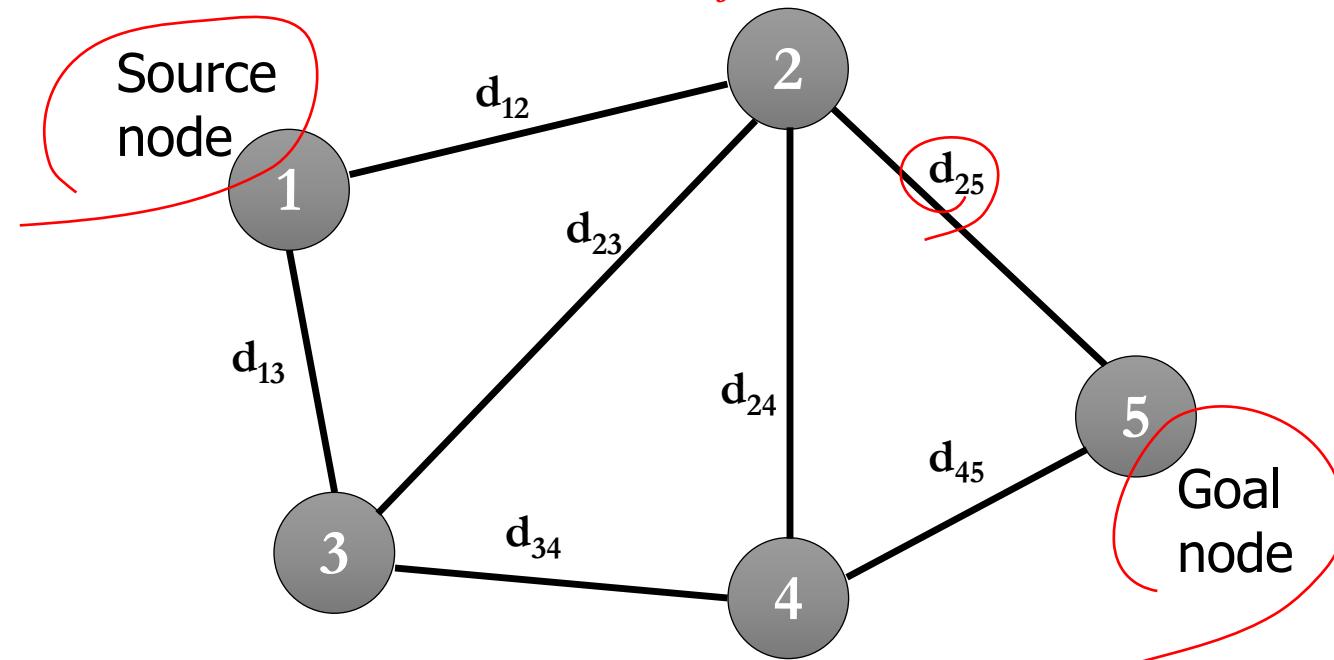
- Real ant and pheromone
- Food
- Continuous
- Pheromone update while moving
- Solutions are evaluated implicitly

Artificial AC

- Artificial ant (agent) and pheromone (value)
T,
- Solution
- Discrete
- Pheromone update after traverse
- Explicit function to evaluate solutions

ACO - Algorithm

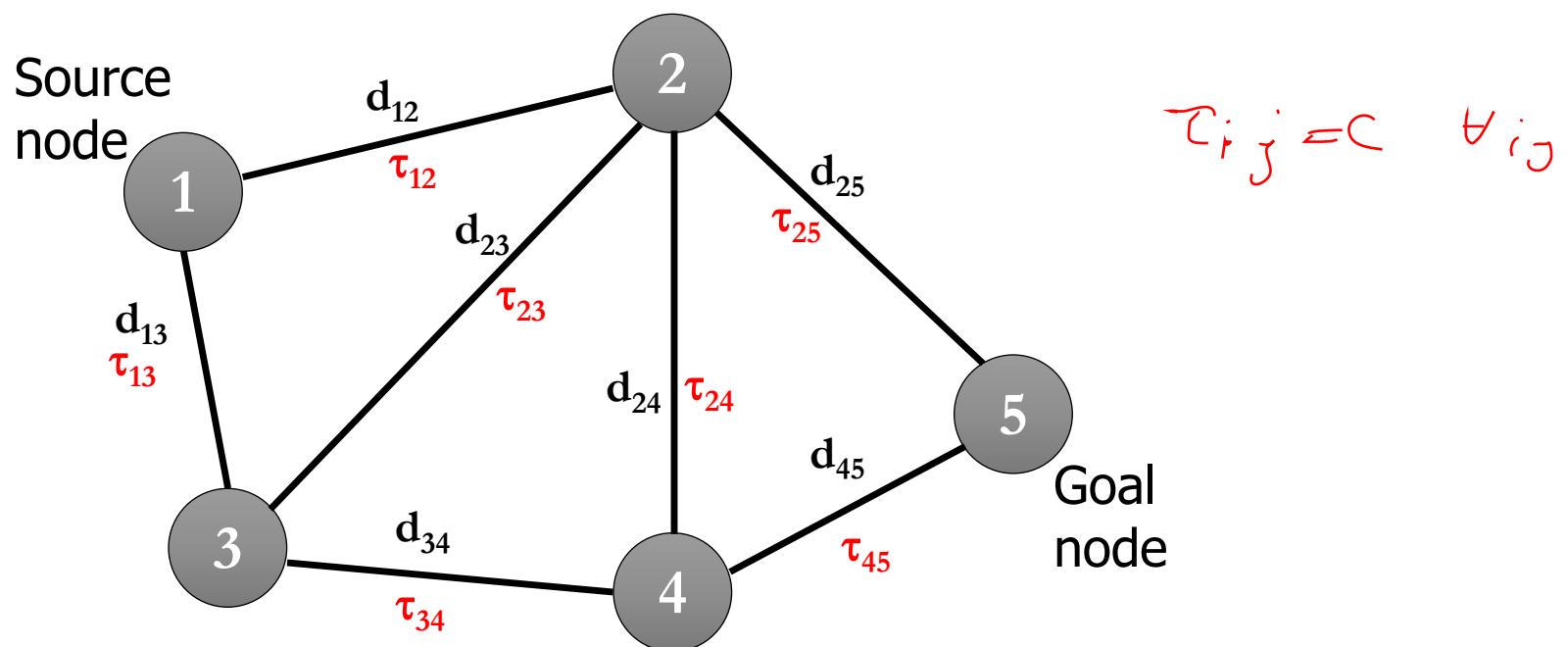
- Let $\underline{G=(N,E)}$ denote a graph, where N is the set of nodes (vertices) and E is the set of arcs (edges),
- Each arc (i, j) is associated with a value d_{ij} denoting the *distance* between nodes i and j ,



- A simple ant algorithm could be used to find the shortest path between two given nodes in the graph.

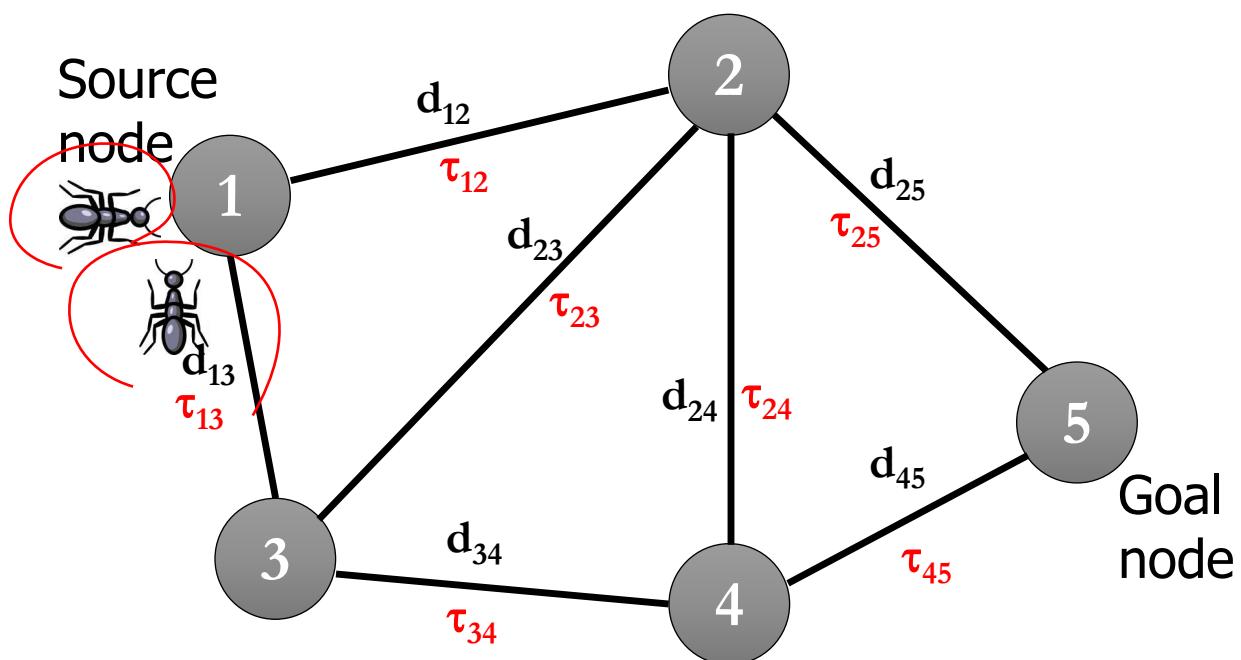
ACO - Algorithm

- Let each arc (i, j) be associated with a value τ_{ij} called the artificial pheromone,
- At the beginning, the same small amount of artificial pheromone is placed on all arcs,



ACO - Algorithm

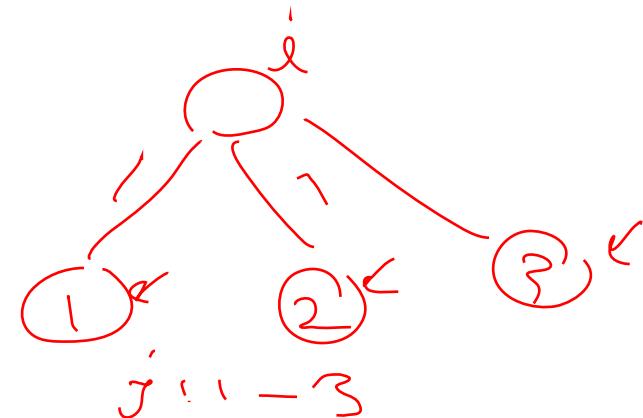
- Place a group of m ants at the source node.



ACO - Algorithm

- At each node \underline{i} , the ant has a choice to move to any of the \underline{j} nodes connected to it,
- Each node $j \in N_i$ connected to i has a probability to be selected by ant K equal to:

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha / d_{ij}^\beta}{\sum_{n \in N_i} \tau_{in}^\alpha / d_{in}^\beta} & \text{if } j \in N_i \\ 0 & \text{if } j \notin N_i \end{cases}$$



- $\underline{\alpha}$ and $\underline{\beta}$ are chosen to balance the $\underline{\text{local}}$ vs. the $\underline{\text{global}}$ search ability of the ant.

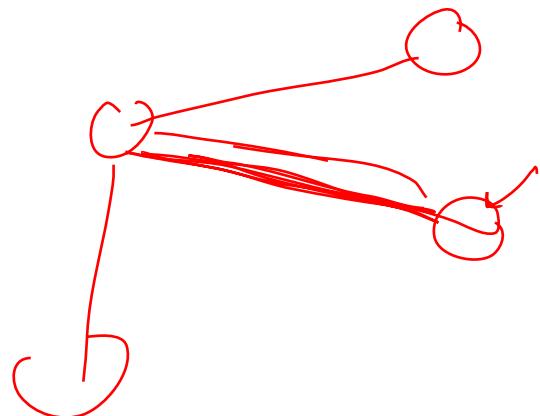
ACO - Algorithm

- Evaporate the artificial pheromone:

$$\tau_i = (1 - \rho) \times \tau_i, \quad \rho \in (0,1]$$

- The ant deposits extra pheromone on the arc it chooses:

$$\tau_{ij} = \tau_{ij} + \Delta\tau$$



- This will increase the probability of a subsequent ant choosing the same arc and referred to as ***online step_by_step*** pheromone update.

A Co

ACO - Algorithm

- Different approaches exist for choosing the value of $\Delta \tau$:
 - **Ant density model**, adding Q , a constant value, hence the final pheromone added to the edge will be proportional to the number of ants choosing it. This doesn't take the edge length into account,
 - **Ant quantity model**, adding Q/d_{ij} , taking the edge length into account, hence enforcing the ant local search ability.
 - **Online delayed** pheromone update, sometimes referred to as the **ant cycle model**,
 - After the ant builds the solution, it traces the path backward and updates the pheromone trails on the visited arcs according to the solution quality.
- online
step-by-step

ACO - Algorithm

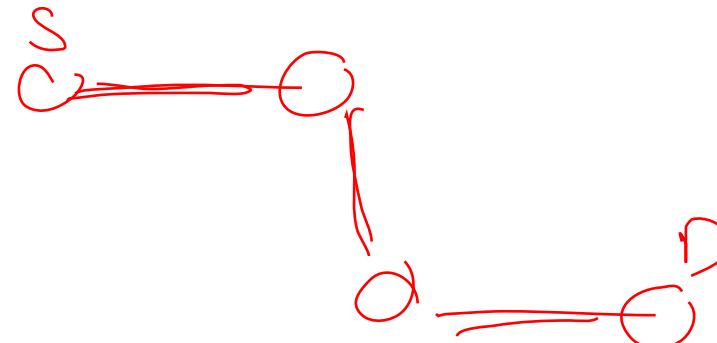
- The online delayed pheromone update is done by adding:

$$\Delta \tau_{ij} = \frac{Q}{L^k}$$

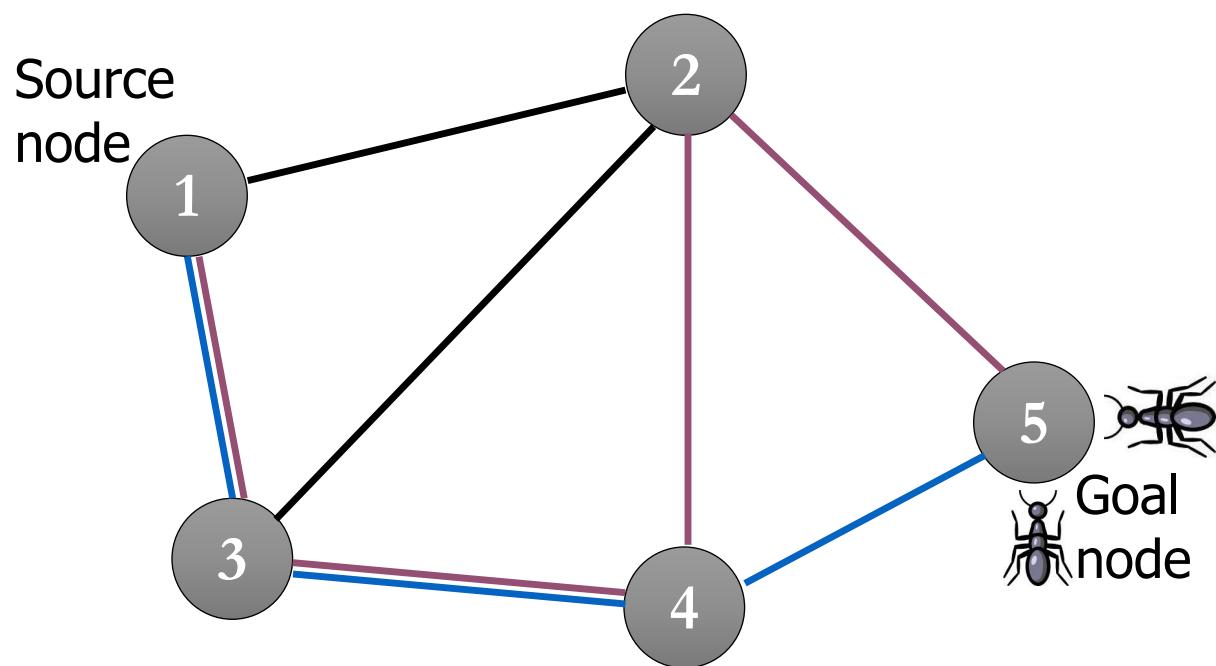
for every arc (i, j) on the path, where L^k is the length of the path found by ant k .

- Done after the evaporation phase

- Repeat the process for different initialization of pheromone

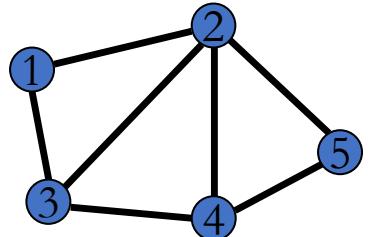


ACO - Algorithm

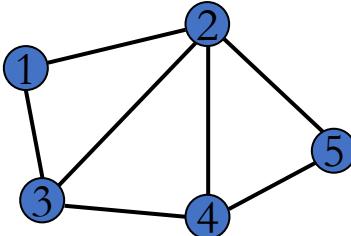


ACO - Algorithm

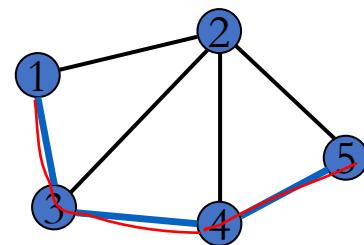
Start



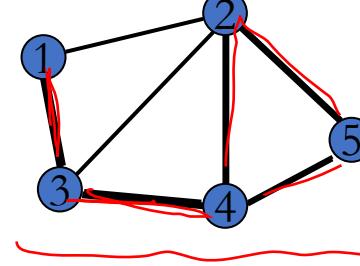
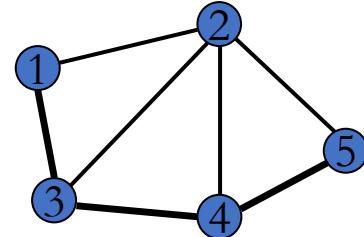
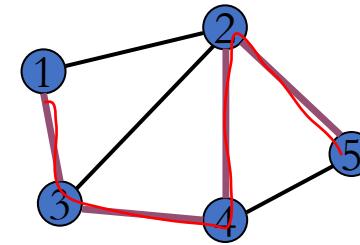
Evaporation



Solution 1



Solution 2



ACO Metaheuristic

- Algorithm Ant colony optimization metaheuristic

- Set parameters, initialize pheromone trails $\propto \rho^m Q \tau_{ij}$
- while termination conditions not met do

- ConstructAntSolutions P
- ApplyLocalSearch (optional)
- UpdatePheromones

- end while

ACO algorithm

- Termination conditions:
 - Max number of iterations reached
 - Acceptable solution reached
 - All ants (or most of them) follow the same path, i.e stagnation.

Parameters

- Number of ants:
 - More ants more computations, but also more exploration.
- Max number of iterations: has to be enough to allow convergence.
- Initial pheromone: constant, random
- Pheromone decay parameter ρ

$\alpha \beta$

Components

- **Transition rule:** probability of selection for the ant
- **Pheromone evaporation rule**
- **Pheromone update rule**
- **Problem heuristic if used**
- **Quality of solution measure**
- **Termination Criteria**

Ant System Algorithm

- *Ant System Algorithm (AS):*

- Proposed by Dorigo et al.,

- Uses the same basic steps outlined in ACO,

- The online delayed pheromone update is adopted using all the solutions of the current iteration.

probability with which ant k in city r chooses to move to the city s

$$p_k(r, s) = \begin{cases} \frac{[\tau(r, s)] \cdot [\eta(r, s)]^\beta}{\sum_{u \in J_k(r)} [\tau(r, u)] \cdot [\eta(r, u)]^\beta}, & \text{if } s \in J_k(r) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where τ is the pheromone, $\eta = 1/\delta$ is the inverse of the distance $\delta(r, s)$, $J_k(r)$ is the set of cities that remain to be visited by ant k positioned on city r (to make the solution feasible), and β is a parameter which determines the relative importance of pheromone versus distance ($\beta > 0$).

$$\tau(r, s) \leftarrow (1 - \alpha) \cdot \tau(r, s) + \sum_{k=1}^m \Delta\tau_k(r, s) \quad (2)$$

where

$$\Delta\tau_k(r, s) = \begin{cases} \frac{1}{L_k}, & \text{if } (r, s) \in \text{tour done by ant } k \\ 0, & \text{otherwise} \end{cases}$$



- M. Dorigo."Optimization, Learning and Natural Algorithms". Ph.D. thesis, DEI, Politecnico di Milano, Italy, pp. 140, 1992.
- M. Dorigo, V. Maniezzo and A. Colorni." Ant System: Optimization by a Colony of Cooperating Agents". IEEE Trans. Syst., Man, and Cybern. Part B, vol. 26, no. 1, 1996.

Ant Colony System Algorithm

- *Ant Colony System Algorithm (ACS):*
 - Proposed by Gambardella and Dorigo,
 - Based on AS but is different in:
 - Transition rule based on Elitist strategy (balances exploitation vs exploration)
 - Pheromone update rule
 - Local pheromone update
- Informally, the ACS works as follows: ants are initially positioned on cities chosen according to some initialization rule (e.g., randomly). Each ant builds a tour (i.e., a feasible solution to the TSP) by repeatedly applying a stochastic greedy rule (the state transition rule). While constructing its tour, an ant also modifies the amount of pheromone on the visited edges by applying the local updating rule. Once all ants have terminated their tour, the amount of pheromone on edges is modified again (by applying the global updating rule). As was the case in ant system, ants are guided, in building their tours, by both heuristic information (they prefer to choose short edges) and by pheromone information. An edge with a high amount of pheromone is a very desirable choice. The pheromone updating rules are designed so that they tend to give more pheromone to edges which should be visited by ants

ACS

- If $q < q_o$ (q is a random number, q_o is in $(0,1)$):

$$j = \arg \max_{j \in N_i^k} \left\{ \tau_{ij}(t) \eta_{ij}^\beta \right\}$$

- Else use:

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

- J element of N_i^k is a randomly selected node give probability P_{ij}^k

- This creates bias towards choices of better quality, q_o controls this bias.

ACS

- Uses the Ant density model update, where all the ants update their last traversed edge:

$$\tau_{ij} = (1 - \rho_2)\tau_{ij} + \rho_2\tau_0$$

- Introduced a new pheromone update rule (***using the best solution (ant) only***).

- delayed update: $\tau_{ij}(t+1) = (1 - \rho_1)\tau_{ij}(t) + \rho_1\Delta\tau_{ij}^{best}(t)$

- Best here can be ***iteration best or global best***

$$\Delta\tau(r, s) = \begin{cases} (L_{gb})^{-1}, & \text{if } (r, s) \in \text{global-best-tour} \\ 0, & \text{otherwise} \end{cases}$$

Max-Min Ant System Algorithm

- **Max-Min Ant System Algorithm (MMAS):**

- Proposed by Stutzle and Hoos to overcome stagnation.
- The update is done using the best solution (best ant) in the current iteration or the best solution over all, also decay in the update of the pheromone

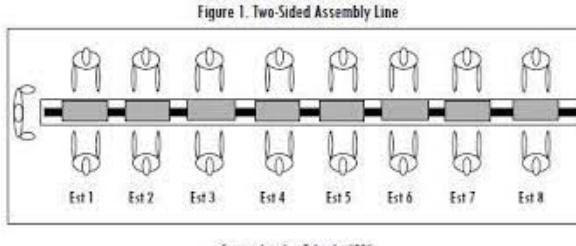
$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \rho\Delta\tau_{ij}^{best}(t)$$

- The values of the pheromone are restricted between τ_{min} and τ_{max} , allows high exploration in the beginning (τ_{max}) and more intensification later.
- The values of τ_{min} and τ_{max} are chosen experimentally, although they could be calculated analytically if the optimal solution is known,
- Improved the performance significantly over AS.

Ant Colony Optimization Applications

ACO Applications

- Travelling Salesman Problem (TSP),
- Assembly Line Balancing (ALB), assigning tasks to workstations in a serial production system
- Cell Assignment in PCS Networks (CA).
- Other



Source: Jurado y Tabora, 2006.

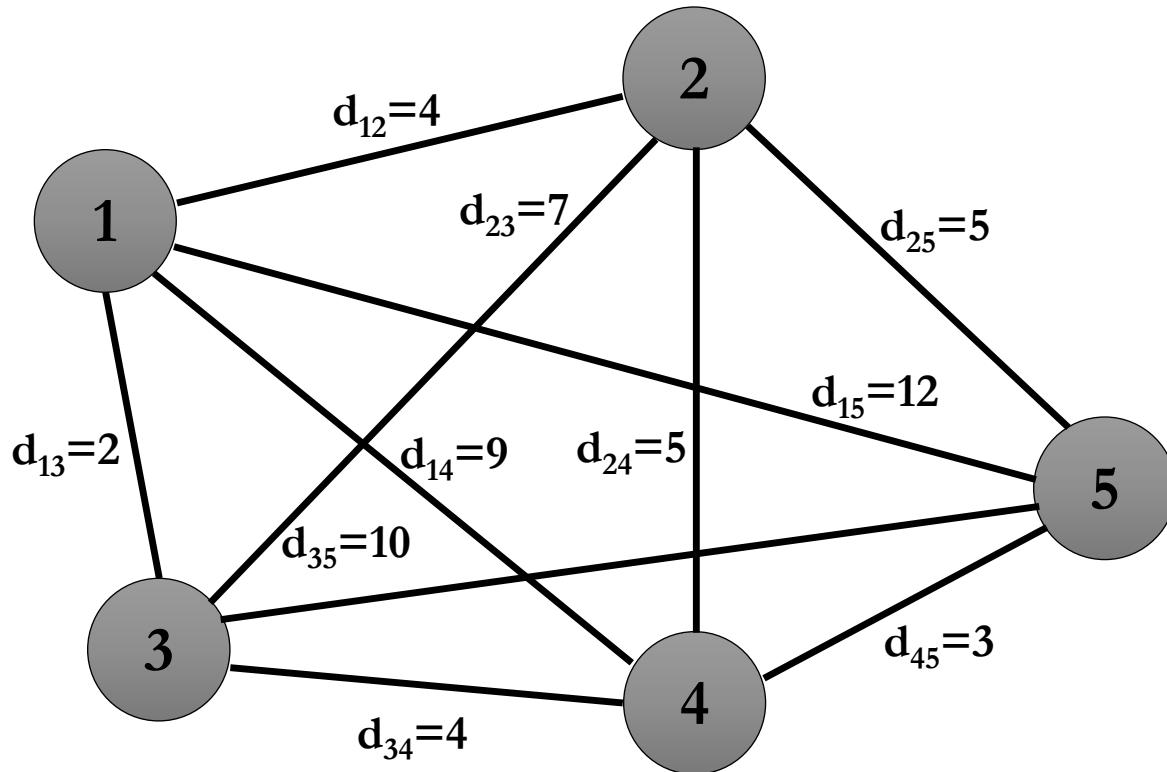


@ dreamstime.com

ACO Applications

Travelling Salesman Problem

ACO for TSP - Example



ACO for TSP - Example

- A small artificial pheromone value is placed on all the edges,
- M ants are placed on the graph divided among all the nodes,
- M (number of ants) roundtrips are created in n (number of cities) steps.

ACO Algorithm Overview

- Initialize the pheromone trails,
- While termination condition not met
 - **ConstructAntSolutions**,
 - **ApplyLocalSearch** (*Optional*),
 - **UpdatePheromones**,
- end

ACO for TSP - Example

- The ConstructAntSolutions step:
 - For each ant:
 - A new node is randomly chosen according to the selection probability,
 - This node is added to the ants' *visited list* (so as not to be re-selected because every city is visited once).
 - After the completion of the roundtrips, the visited lists are emptied,

ACO Algorithm Overview

- Initialize the pheromone trails,
- While termination condition not met
 - ConstructAntSolutions,
 - *ApplyLocalSearch (Optional),*
 - UpdatePheromones,
- end

ACO for TSP - Example

- Usually, the online step-by-step update rule is *not adopted*,
- The ApplyLocalSearch step:
 - These are optional actions referred to as *daemon actions*,
 - One action is to apply a local search method in order to improve the constructed solution,
 - Another action is to add extra pheromone (*delayed update*) based on the collection of some global information.

ACO Algorithm Overview

- Initialize the pheromone trails,
- While termination condition not met
 - ConstructAntSolutions,
 - ApplyLocalSearch (*Optional*),
 - **UpdatePheromones**,
- end

ACO for TSP - Example

- The UpdatePheromones step:
 - Apply pheromone evaporation,
 - Update the pheromone trails using a chosen set of good solutions, different approaches use (*Ant cycle model*):
 - All the solutions found in the current iteration,
 - The best solution found in the current iteration,
 - The best solution found so far,

ACO for TSP - Example

- A simple iteration:
 - Assume that an ant k was placed at node 1,
 - The neighbours of node 1 are :

$$N_1 = \{2, 3, 4, 5\}$$

- Initially, all the edges have the same pheromone value (assume $\tau=1$).

ACO for TSP - Example

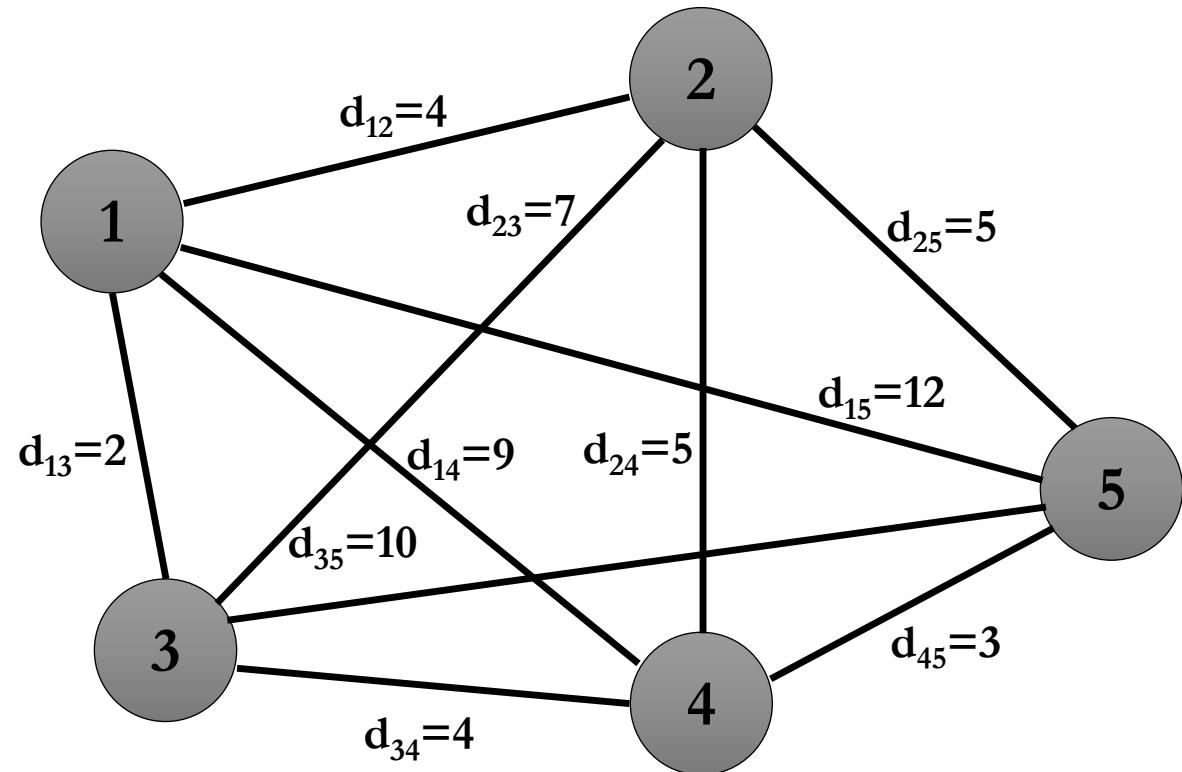
- Assuming α and β are both equal to 1,

- Then:

$$\sum_{n \in N_1} \tau_{1n}^\alpha / d_{1n}^\beta = \frac{1}{4} + \frac{1}{2} + \frac{1}{9} + \frac{1}{12} \approx 0.95$$

$$p_{12}^k = \frac{0.25}{0.95} \approx 0.26, p_{13}^k = \frac{0.5}{0.95} \approx 0.53$$

$$p_{14}^k = \frac{0.11}{0.95} \approx 0.12, p_{15}^k = \frac{0.08}{0.95} \approx 0.08$$



ACO for TSP - Example

- Node 3 has the highest probability of being selected,
- If node 3 gets selected, ant ***k*** moves to that node and continues with the next iteration where:

$$N_3 = \{2, 4, 5\}$$

ACO for TSP - Example

- Hence:

$$\sum_{n \in N_3} \tau_{1n}^\alpha / d_{1n}^\beta = \frac{1}{7} + \frac{1}{4} + \frac{1}{10} \cong 0.49$$

$$p_{32}^k = \frac{0.14}{0.49} \cong 0.29, p_{34}^k = \frac{0.25}{0.49} \cong 0.51$$

$$p_{35}^k = \frac{0.1}{0.49} \cong 0.20$$

- Another node is selected, and so on ...

ACO for TSP - Example

- If ant k completes the tour:

$$Tour = \{1, 3, 2, 4, 5\}$$

$$Cost = 29$$

- First, all the pheromone trails get evaporated,
- And then, if this ant is selected to update the pheromone trails, it enforces the edges

$$\begin{aligned} & \{1, 3\}, \{3, 2\}, \{2, 4\}, \\ & \{4, 5\}, \{5, 1\} \end{aligned}$$

with the value $Q/29$.

References

Text: Fundamentals of Computational Swarm Intelligence by A. P. Engelbrecht, Wiley, 2005.
Chapter 1, 22, and 23.

Papers:

- ID. Couzin, J. Krause, R. James, GD. Ruxton, NR. Franks."Collective Memory and Spatial Sorting in Animal Groups". *Journal of Theoretical Biology*, 218, pp. 1–11, 2002.
- C. Grosan, A. Abraham and C. Monica."Swarm Intelligence in Data Mining". In *Swarm Intelligence in Data Mining*, A. Abraham, C. Grosan and V. Ramos (Eds), Springer, pp. 1–16, 2006.
- MM. Millonas .“Swarms, phase transitions, and collective intelligence”. In CG. Langton Ed., *Artificial Life III*, Addison Wesley, Reading, MA, 1994.

References

- P. P. Grasse.“Recherches sur la biologie des termites champignonnistes (*Macrotermitina*)”. Ann. Sc. Nat., Zool. Biol. anim., 6, 97, 1944.
- P. P. Grasse.”La reconstruction du nid et les coordinations interindividuelles chez *bellicositermes natalensis* et *cubitermes* sp. La theorie de la stig- merge: essai d'interpretation du comportement des termites constructeurs”. Insectes Sociaux, 6, 41, 1959.
- J. L. Deneubourg, S. Aron, S. Goss and J. M. Pasteels.”The self- organizing exploratory pattern of the Argentine ant”. Journal of Insect Behaviour, 3, 159, 1990.
- S. Goss, S. Aron, J. L. Deneubourg and J. M. Pasteels.”Self-organized shortcuts in the Argentine ant”. Naturwissenschaften, 76, 579, 1989.
- M. Dorigo.”Optimization, Learning and Natural Algorithms”. Ph.D. thesis, DEI, Politecnico di Milano, Italy, pp. 140, 1992.

References

- M. Dorigo, V. Maniezzo and A. Colomi.” Ant System: Optimization by a Colony of Cooperating Agents”. IEEE Trans. Syst., Man, and Cybern. Part B, vol. 26, no. 1, 1996.
- T. Stutzle and H. H. Hoos.”*MAX-MIN* Ant System”. *Future Generation Comput. Syst.*, vol. 16, no 8, 2000.
- M. Dorigo and L. M. Gambardella.”Ant Colony System: a cooperative learning approach to the traveling salesman problem”. IEEE Trans. Evolutionary Computation, vol. 1, no. 1, 1997.
- J. Bautista and J. Pereira. “Ant Algorithms for A Time and Space Constrained Assembly Line Balancing Problem”. European Journal of Operation Research, 177, pp. 2016-2032, 2002.
- S. J. Shyu, B. M. T. Lin and T. S. Hsiao.“An Ant Algorithm for Cell Assignment in PCS Netwroks”. IEEE International Conference on Networking, Sensing and Control, pp. 1081-1086, 2004.