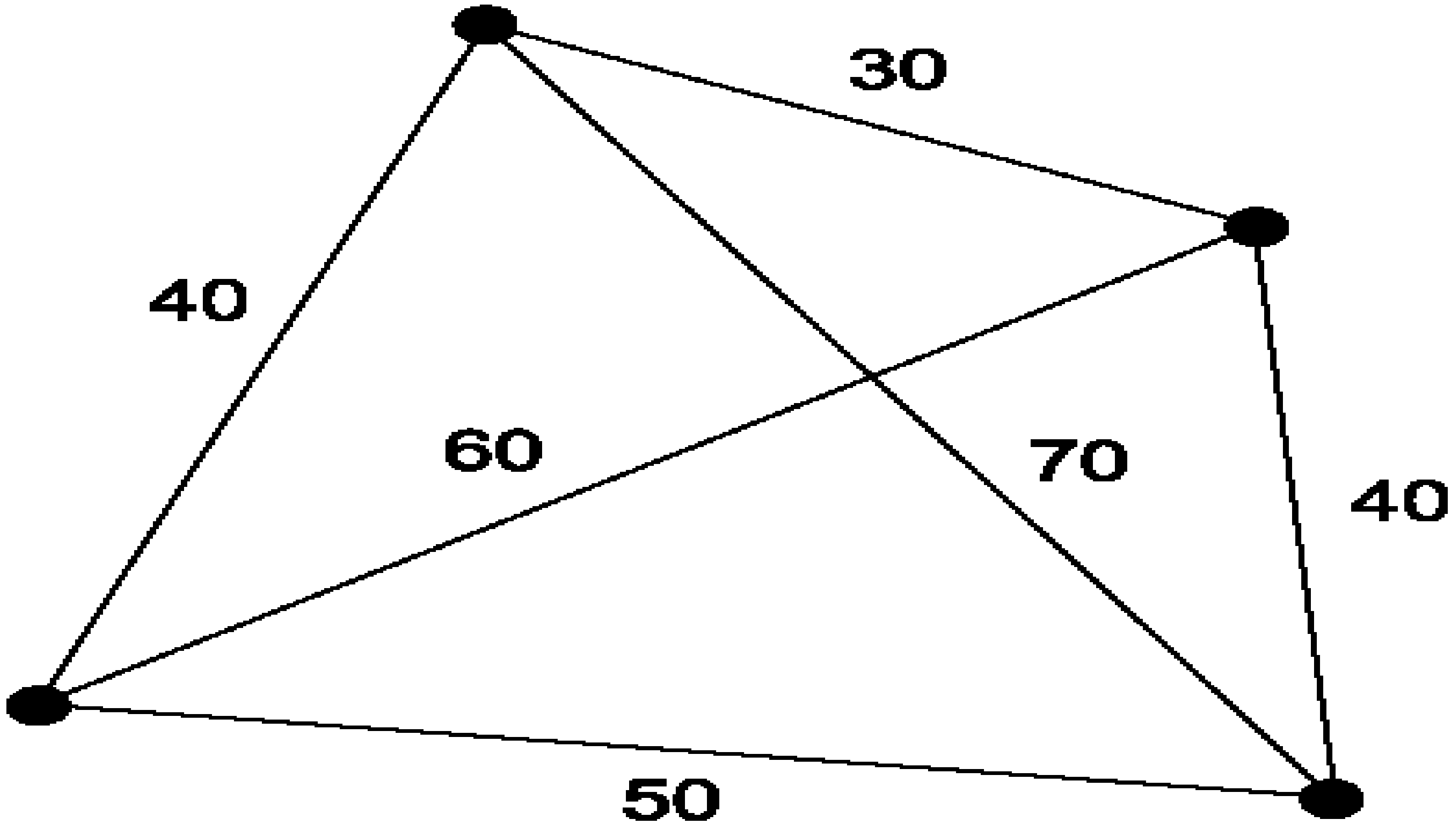
Ant Colony Optimization to solve Traveling Salesman Problem



M. A. H. Akhand
A. B. M. Junaed
Md. Forhad Hossain

Topics

1. History of Ant Colony Optimization (ACO)
2. Real Ant’s Behavior
3. The Concept of Ant System
4. ACO System Concept
5. Pheromone Update Formula
6. Ant Colony Optimization Algorithm
7. Apply Ant Colony Optimization Algorithm to solve Traveling Salesman Problem
8. Example of a Simple AS to solve TSP
9. Application of ACO
10. Advantage and Disadvantages

History of Ant Colony Optimization

- The first ACO system was introduced by Marco Dorigo in his Ph.D. thesis (1992),
- It was called Ant System (AS).
- AS was initially applied to the traveling salesman problem.
- Applied later to various hard optimization problems

Macro Dorigo

Gambardella

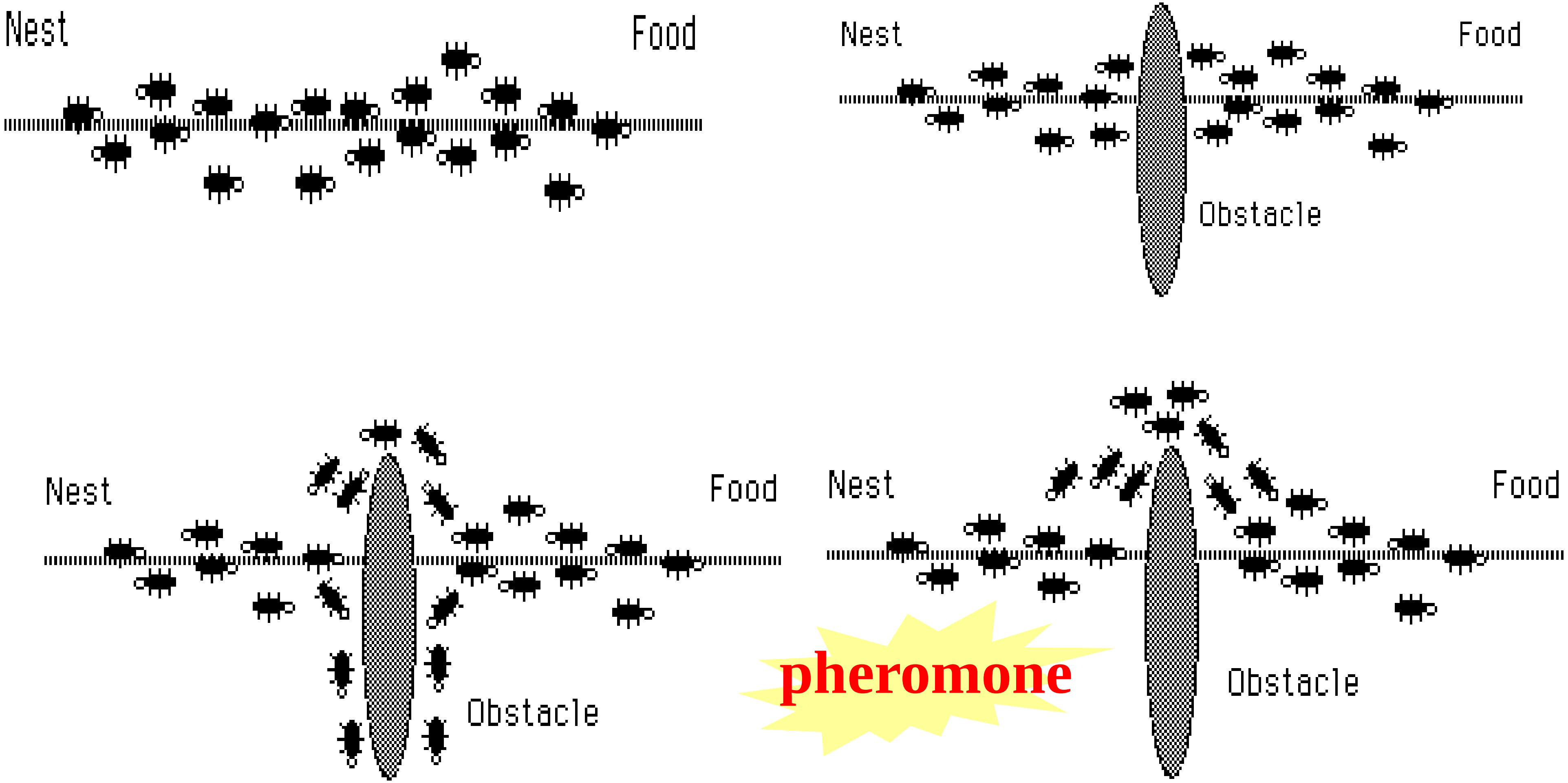
Real Ant’s Behavior

- Natural behavior of ants have inspired scientists to mimic insect operational methods to solve real-life complex problems.

- By observing ant behavior, scientists have begun to understand their means of communication.

- Ants choose paths depending on pheromone
- After collecting food, paths are marked
- Pheromone accumulation is faster on the shorter path
- After some time, the shortest path has the highest probability

Real Ant’s Behavior



The Concept of Ant System

- Ants (blind) navigate from nest to food source
- Shortest path is discovered via pheromone trails
  - each ant moves at random
  - pheromone is deposited on path
  - ants detect lead ant's path, inclined to follow
  - more pheromone on path increases the probability of path being followed

- These pheromones evaporate with time.

The Concept of Ant System

- The shorter path will be reinforced by the pheromones further.
- Finally , the ants arrive at the shortest path.
- Starting node selected at random
- Path selected at random
  - based on amount of “trail” present on possible paths from starting node
  - higher probability for paths with more “trail”
- Ant reaches next node, selects next path
- Continues until reaches starting node
- Finished “tour” is a solution

The Concept of Ant System

- A completed tour is analyzed for optimality
- “Trail” amount adjusted to favor better solutions
  - better solutions receive more trail
  - worse solutions receive less trail
  - higher probability of ant selecting path that is part of a better-performing tour
- New cycle is performed
- Repeated until most ants select the same tour on every cycle

Ant Colony Algorithms

- Algorithm was inspired by observation of real ant colonies.
- Ants are essentially blind, deaf and dumb.
- Q: how can ants find the shortest path to food sources?
- Ants deposit *pheromones* on ground that form a trail. The trail attracts other ants.

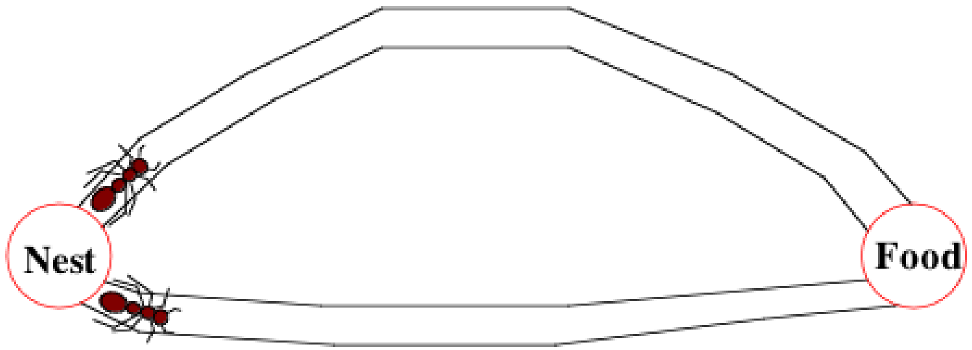
- The ants evaluate the cost of the paths they have traversed.
- The shorter paths will receive a greater deposit of pheromones.
- An evaporation rule will be tied with the pheromones, which will reduce the chance for poor quality solutions.

How Ants find foods

How Ants find foods

Ant foraging

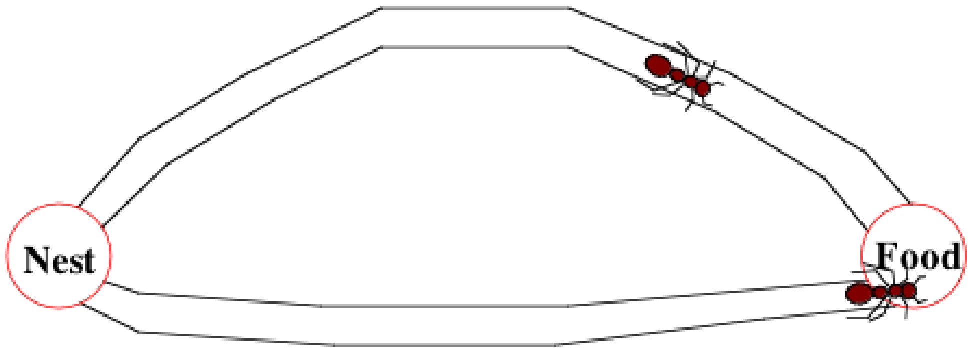
Cooperative search by pheromone trails



How Ants find foods

Ant foraging

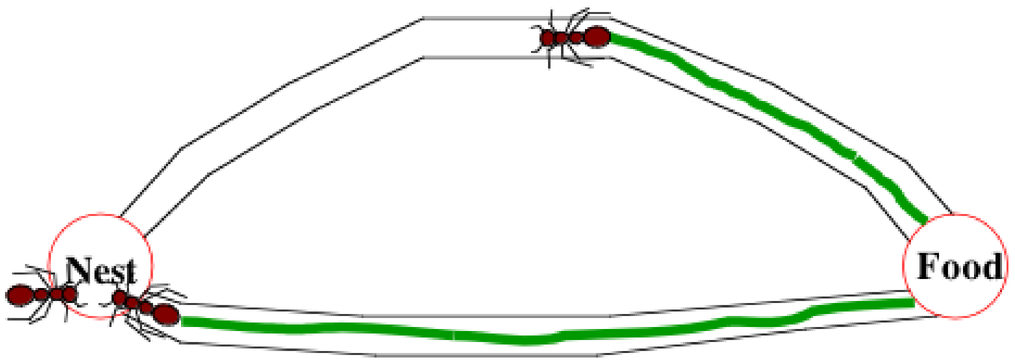
Cooperative search by pheromone trails



How Ants find foods

Ant foraging

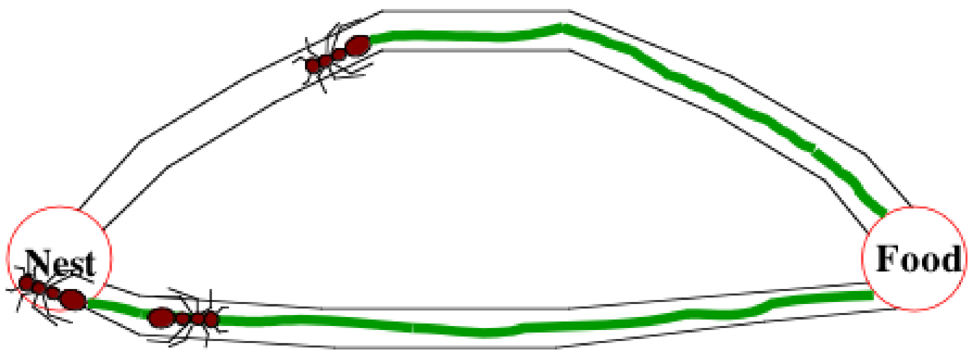
Cooperative search by pheromone trails



How Ants find foods

Ant foraging

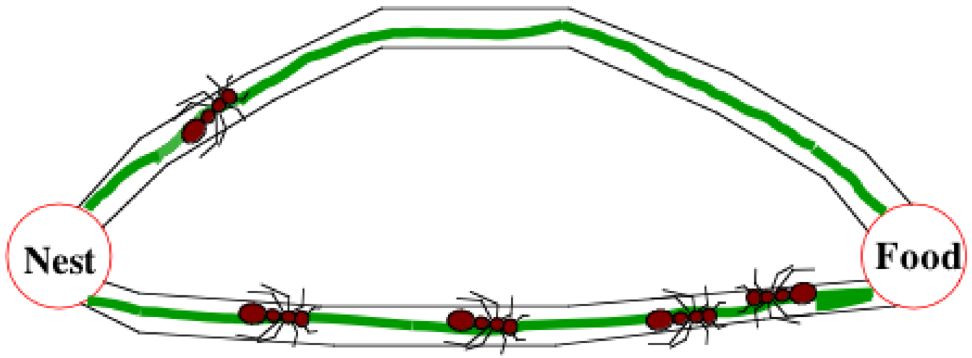
Cooperative search by pheromone trails



How Ants find foods

Ant foraging

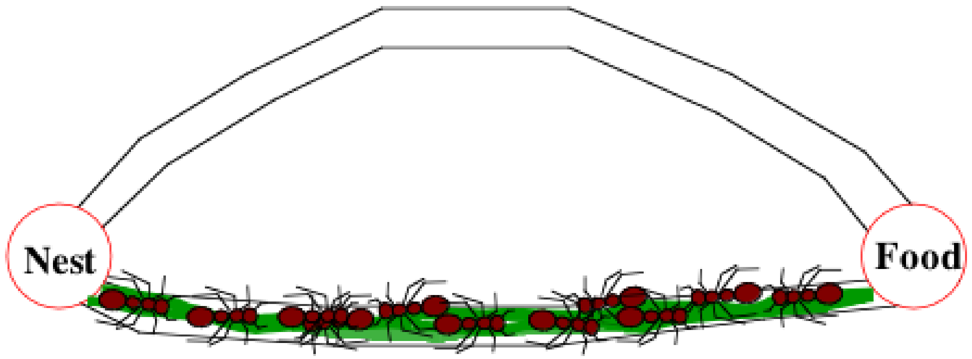
Cooperative search by pheromone trails



How Ants find foods

Ant foraging

Cooperative search by pheromone trails



Ant System (AS)

- Three different versions are proposed:
  - Ant Density
  - Ant quantity
  - Ant cycle

Pheromone updated after each move of ant from one city to another

Pheromone updates after all ants construct tour

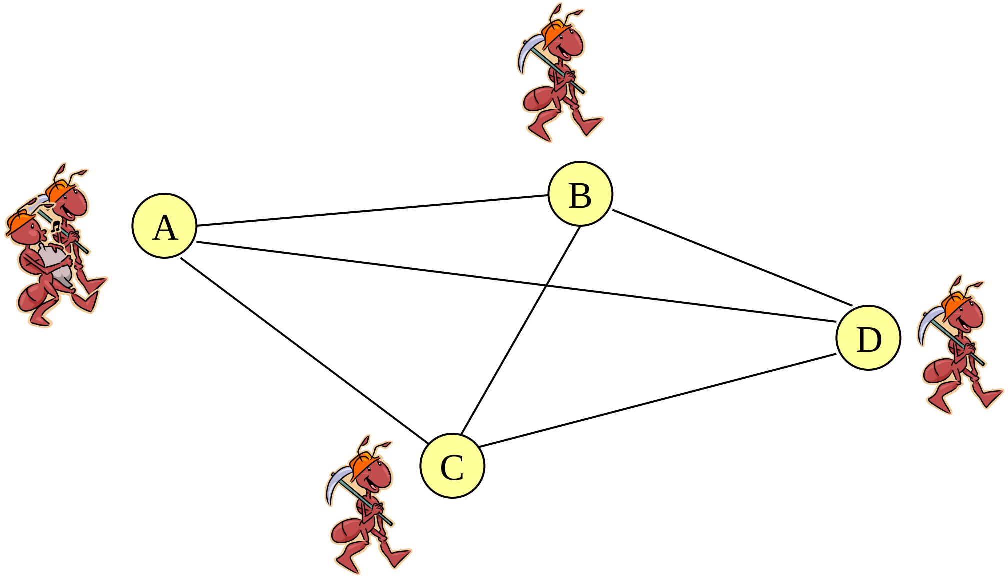
Ant cycle in performed mush better than other two.
We will present Ant-cycle algorithm

Ant Systems for TSP:

Graph (N,E): where N = cities/nodes, E = edges

= *the tour cost from city i to city j (edge weight)*

Ant move from one city i to the next j with some transition probability.



Tour Construction

1. Initially, each ant is put on some randomly chosen city.
2. Tabu list: N_i^k set of all cities that ant k has not visited
3. $n_{ij} = 1/d_{ij}$, **visibility**: Heuristic desirability of choosing city j when in city i.
4. **Pheromone trail**: $T_{ij}(t)$ This is a global type of information

Transition probability: Ant *k*, currently at city *i*, chooses to go to city *j* at *t* th iteration is:

$$P_{ij}^{k}(t) = \frac{[T_{ij}(t)]^{\alpha} \cdot [n_{ij}]^{\beta}}{\sum_{l \in J_i^k} [T_{lk}(t)]^{\alpha} \cdot [n_{lk}]^{\beta}} \quad \text{If } j \in N_i^k$$

Closest cities are selected based on pheromone and distance.

- α,β determines relative influence of pheromone trail and heuristic information
- If α=0, closest cities are more likely to be selected. Approaches to greedy algorithm
- If β=0, only pheromone amplification is at work

Pheromone Update Rule

- After all ants constructed their tours, pheromone trails are updated.
- It is done by
  - Firstly, lowering the pheromone strength on all arcs by a constant factor
  - Secondly, allowing each ant to add pheromone on the arcs it has visited.

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \sum_{k=1}^{m}\Delta\tau_{ij}^{k}(t)$$

(Trail pheromone decay)

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} 1/L^{k}(t) \quad \text{or } Q/L & \text{if arc } (i,j) \text{ is used by ant } k \\ 0 & \text{otherwise} \end{cases}$$

$L^{k}(t)$ is length of *k*th ant's tour. 0 < ρ <= 1 is pheromone trail evaporation.

- ➢ The better the ant's tour is, more pheromone is received by arcs belonging to the tour.
- ➢ In general, arcs used by many ants and contained in shorter tours will receive more pheromone.

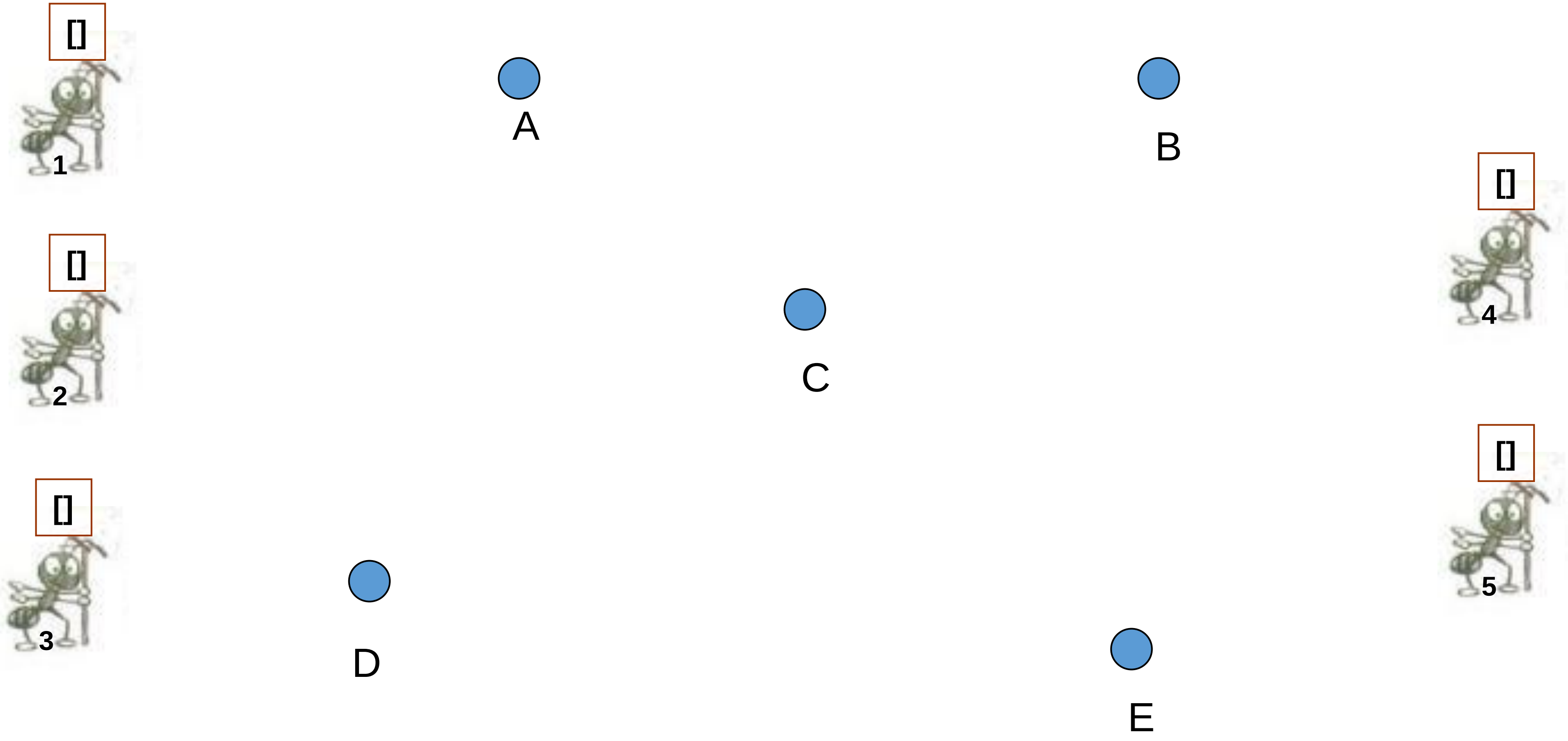
Key Parameters

$$P_{ij}^{k}(t) = \frac{[T_{ij}(t)]^{a} \cdot [n_{ij}]^{\beta}}{\sum_{l \in J_i^k} [T_{lk}(t)]^{a} \cdot [n_{lk}]^{\beta}}$$

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t)$$

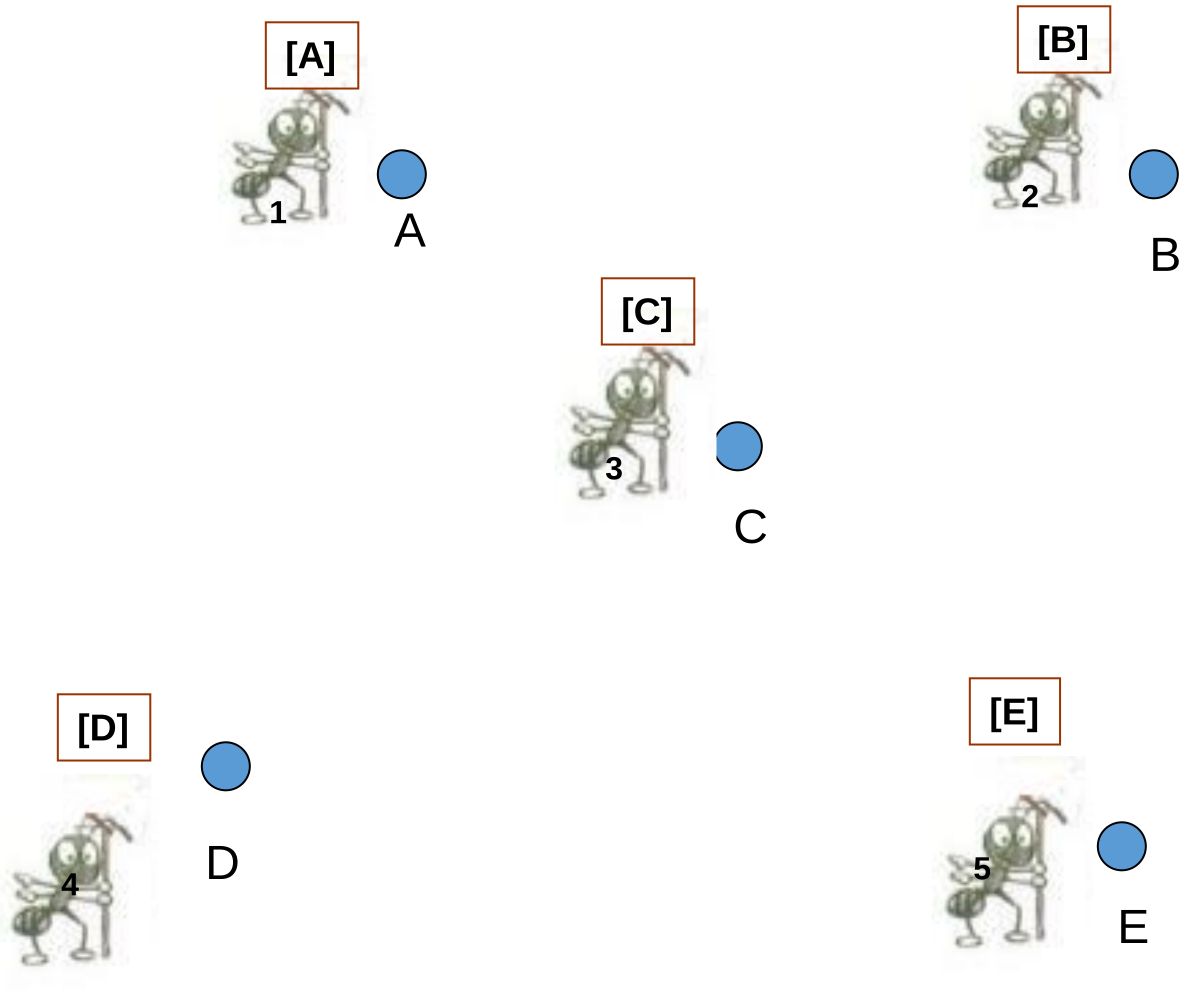
- Trail intensity is given by value of τ_{ij} which indicates the intensity of the pheromone on the trail segment, (*ij*)
- Trail visibility is $\eta_{ij} = 1/d_{ij}$: a heuristic function of the desirability of adding edge (d_{ij} is distance from i to j)
- importance of the intensity in the probabilistic transition is α
- The importance of the visibility of the trail segment is β
- The trail persistence or evaporation rate is given as ρ
- *Q* is a constant and the amount of pheromone laid on a trail segment employed by an ant (for $\Delta\tau_{ij}^{(k)} \leftarrow Q/L_k$)
- Initial pheromone is a small amount on all edges

Example of a Simple AS to solve TSP

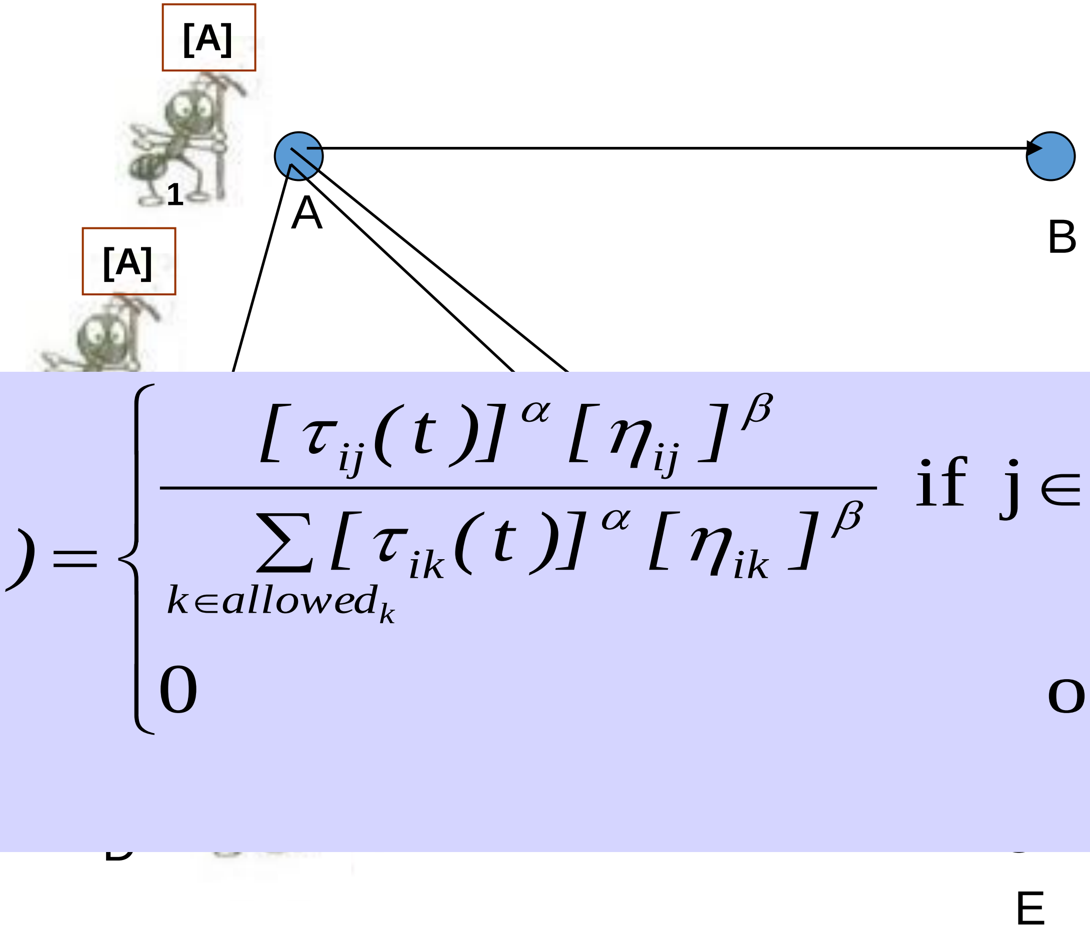


d_{AB} =100;d_{BC} = 60…;d_{DE} =150

Example of a Simple AS to solve TSP

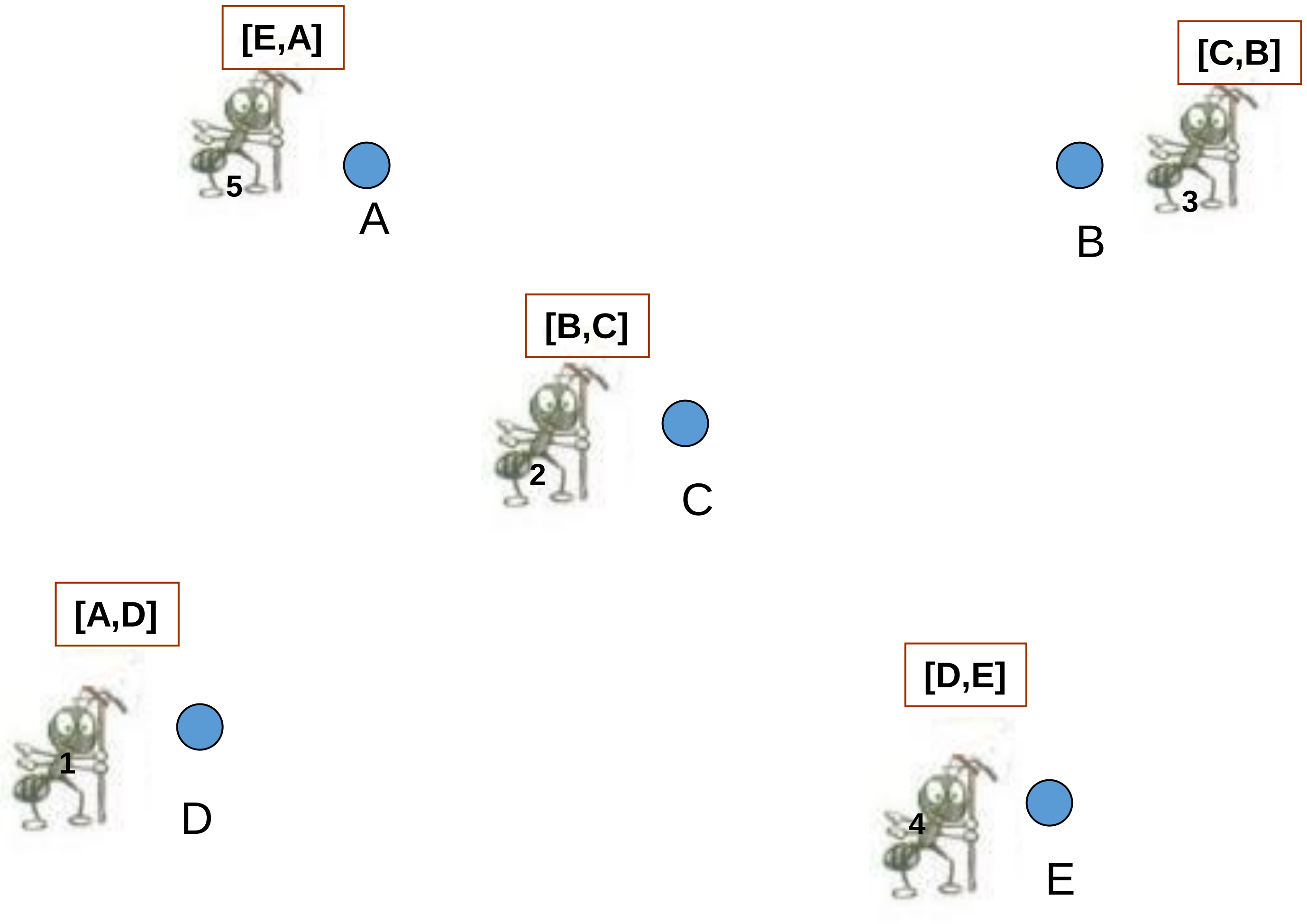


Example of a Simple AS to solve TSP

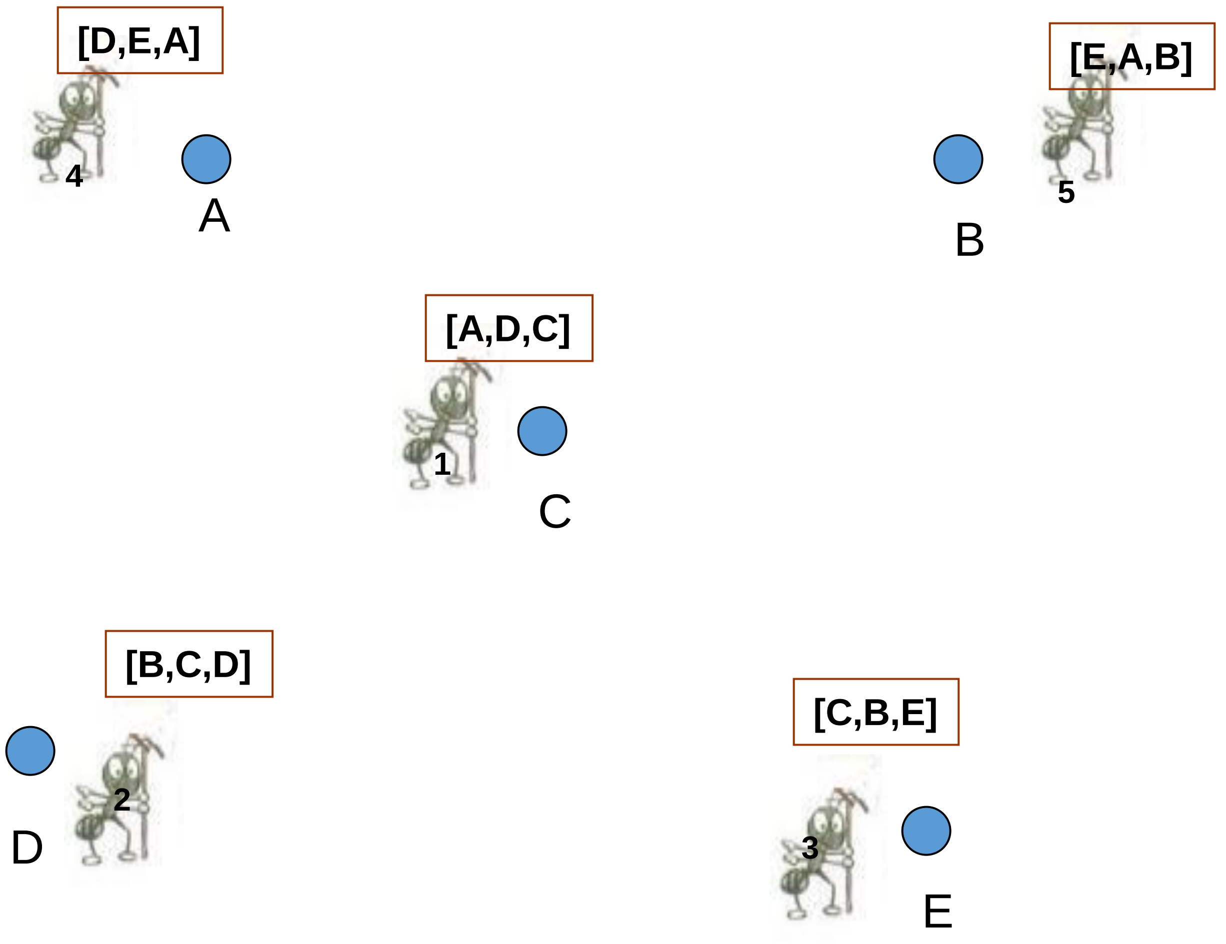


$$p_{ij}^{k}(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum\limits_{k\in allowed_k}[\tau_{ik}(t)]^{\alpha}[\eta_{ik}]^{\beta}} & \text{if } j\in \text{allowed}_{k}\\ 0 & \text{otherwise}\end{cases}$$

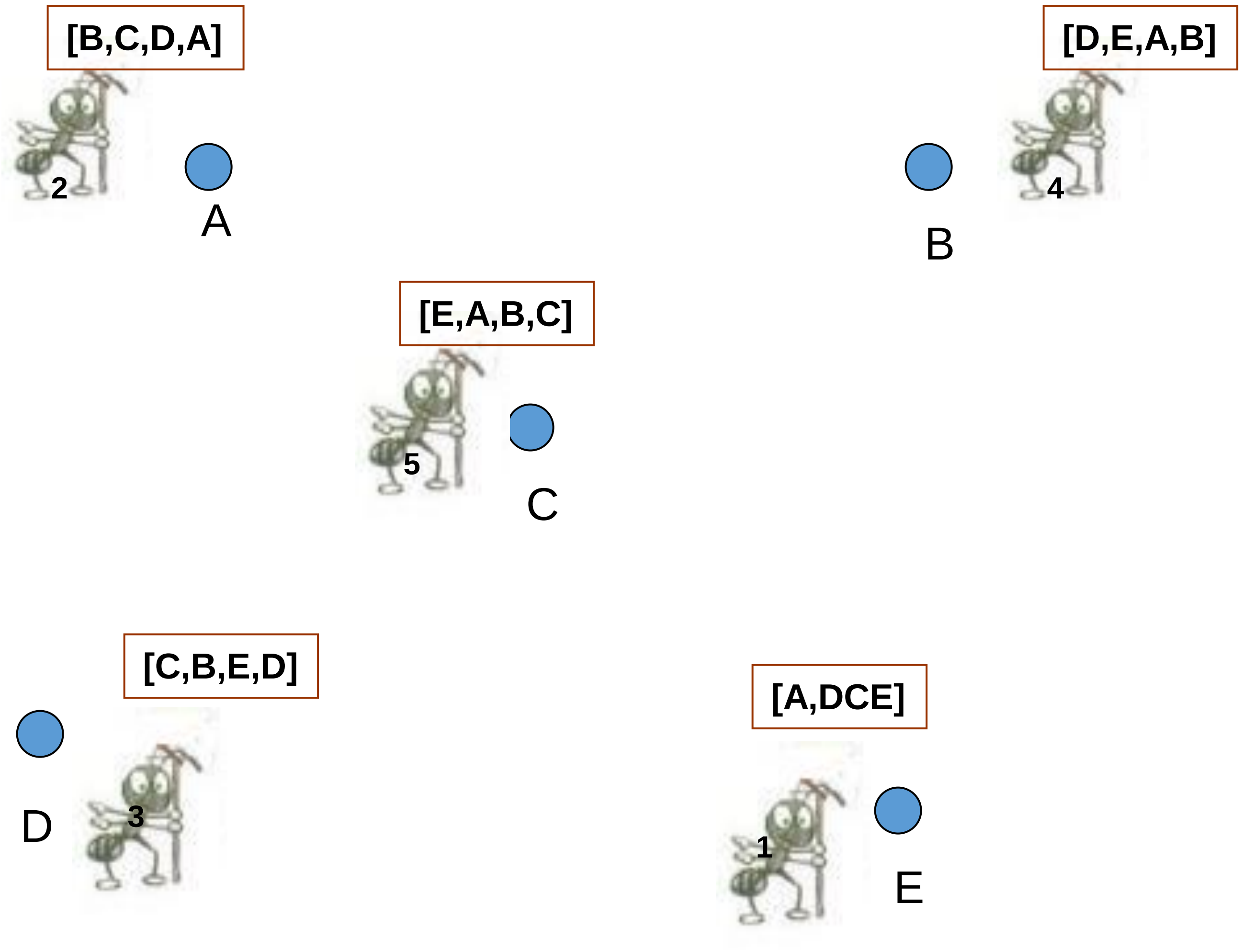
Example of a Simple AS to solve TSP



Example of a Simple AS to solve TSP

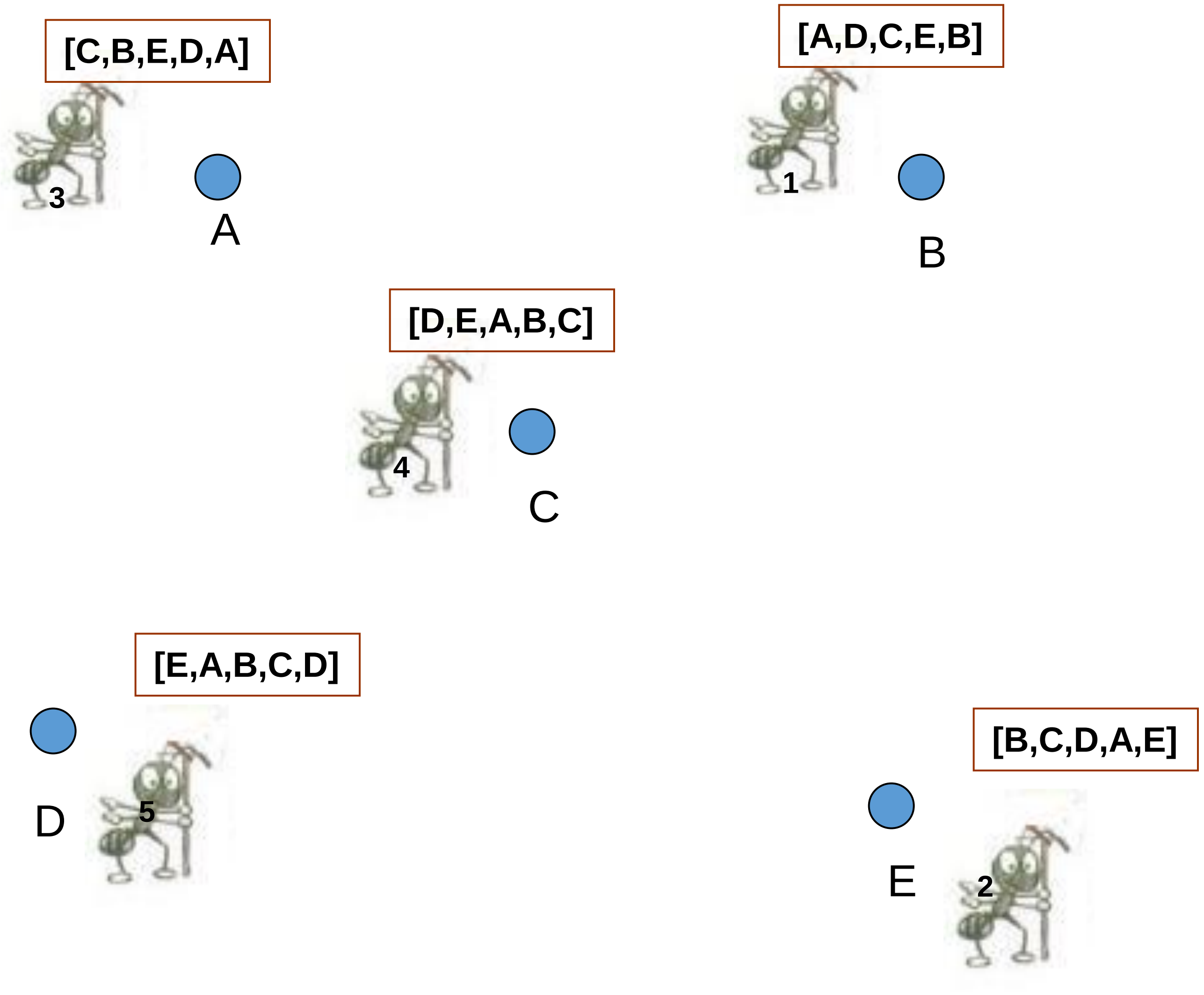


Example of a Simple AS to solve TSP



Presented By: Md. Robiul Islam, CSE KUET (ID: 0907552)

Example of a Simple AS to solve TSP



Example of a Simple AS to solve TSP

[A,D,C,E,B]

L_1 =300

[B,C,D,A,E]

L_2 =450

$$\Delta\tau_{i,j}^{k} = \begin{cases} \dfrac{Q}{L_k} & if\,(i,j) \in \text{tour} \\ 0 & \text{otherwise} \end{cases}$$

[C,B,E,D,A]

$$\Delta\tau_{A,B}^{total} = \Delta\tau_{A,B}^{1} + \Delta\tau_{A,B}^{2} + \Delta\tau_{A,B}^{3} + \Delta\tau_{A,B}^{4} + \Delta\tau_{A,B}^{5}$$

[D,E,A,B,C]

L_4 =280

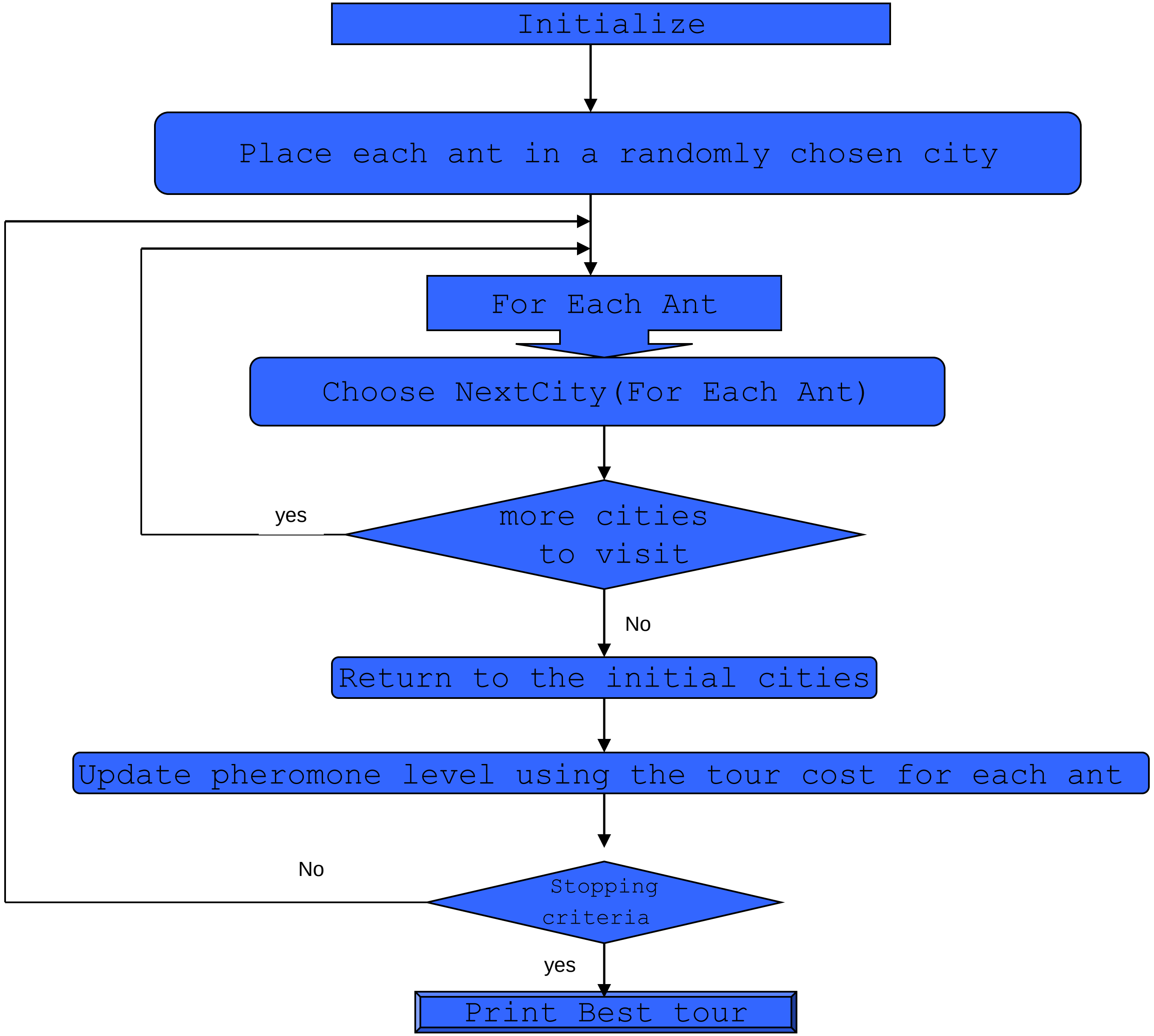
[E,A,B,C,D]

L_5 =420

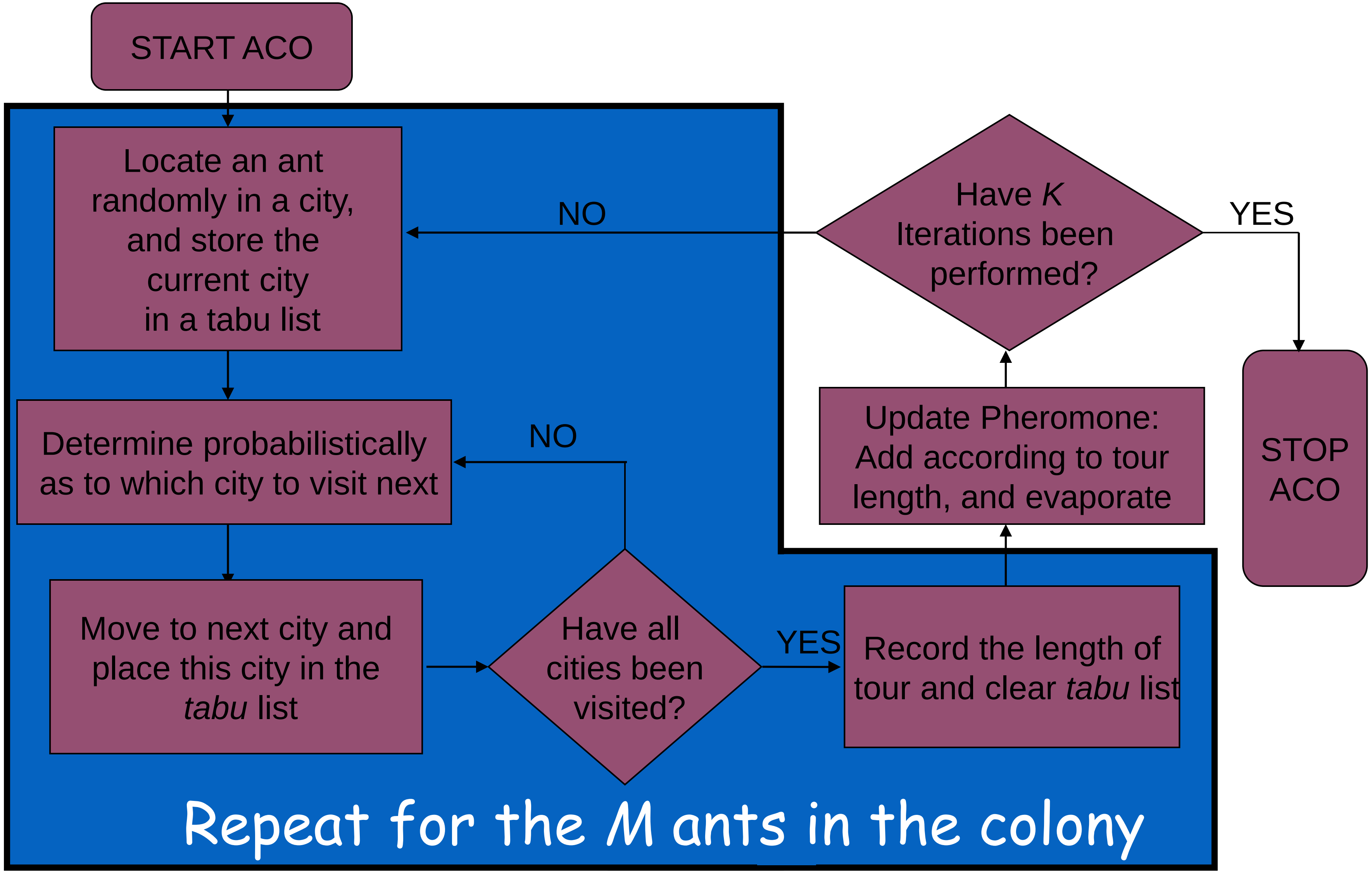
Application of ACO

- Traveling Salesman Problem (TSP)
- Class Scheduling Problem (CSP)
- Function Optimization
- Vehicle Routing
- Sequential Ordering
- Graph Coloring
- Frequency Assignment
- Train Time Scheduling
- Water Distribution Network
- Quadratic assignment problems (QAP)
- Dynamic routing problems in networks and so on…

Ant Systems Algorithm for TSP



Another ACO Flowchart for TSP



ACO Algorithm for TSP

n = number of cities
α, β = relative importance of pheromones vs. heuristic information
Q = deposition constant
ρ = evaporation rate $\in (0, 1)$
$\tau_{ij} = \tau_0$ (initial pheromone between cities i and j) for $i \in [1, n]$ and $j \in [1, n]$
d_{ij} = distance between cities i and j for $i \in [1, n]$ and $j \in [1, n]$
While not(termination criterion)
    For $q = 1$ to $n - 1$
        For each ant $k \in [1, N]$
            Initialize the starting city c_{k1} of each ant $k \in [1, N]$
            Initialize the set of cities visited by ant k: $C_k \leftarrow \{c_{k1}\}$ for $k \in [1, N]$
            For each city $j \in [1, n]$, $j \notin C_k$
                probability $p_{ij}^{(k)} \leftarrow \left(\tau_{ij}^{\alpha}/d_{ij}^{\beta}\right) / \left(\sum_{m=1, m \notin C_k}^{n} \tau_{im}^{\alpha}/d_{im}^{\beta}\right)$
            Next j
            Let ant k go to city j with probability $p_{ij}^{(k)}$
            Use $c_{k,q+1}$ to denote the city selected in the previous line
            $C_k \leftarrow C_k \cup \{c_{k,q+1}\}$
        Next ant
    Next q
    $L_k \leftarrow$ total path length constructed by ant k, for $k \in [1, N]$
    For each city $i \in [1, n]$ and each city $j \in [1, n]$
        For each ant $k \in [1, N]$
            If ant k went from city i to city j
                $\Delta\tau_{ij}^{(k)} \leftarrow Q/L_k$
            else
                $\Delta\tau_{ij}^{(k)} \leftarrow 0$
            End if
        Next ant
        $\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} + \sum_{k=1}^{N} \Delta\tau_{ij}^{(k)}$
    Next city pair
Next generation

Figure 10.6 A simple ant system (AS) for solving a TSP. Each generation, some of the pheromone between cities i and j evaporates, but the pheromone also increases due to ants that travel between the two cities.

ACO Algorithm for TSP

Algorithm 6.3.1: Pseudocode for Ant System.

Input: ProblemSize, $Population_{size}$, m, ρ, α, β
Output: P_{best}

$P_{best} \leftarrow$ `CreateHeuristicSolution(`ProblemSize`)`;
$Pbest_{cost} \leftarrow$ `Cost(`S_h`)`;
Pheromone $\leftarrow$ `InitializePheromone(`$Pbest_{cost}$`)`;
while $\neg$`StopCondition()` **do**
    Candidates $\leftarrow \emptyset$;
    for $i = 1$ **to** m **do**
        $S_i \leftarrow$ `Probabilis` … `truction(`Pheromone, ProblemSize, α, β`)`;
        $Si_{cost} \leftarrow$ `Cost(`S_i`)`;
        if $Si_{cost} \leq Pbest_{cost}$ **then**
            $Pbest_{cost} \leftarrow Si_{cost}$;
            $P_{best} \leftarrow S_i$;
        end
        Candidates $\leftarrow S_i$;
    end
    `DecayPheromone(`Pheromone, ρ`)`;
    foreach $S_i \in$ Candidates **do**
        `UpdatePheromone(`Pheromone, S_i, Si_{cost}`)`;
    end
end
return P_{best};

$$\Delta\tau_{ij}^{(k)} \leftarrow Q/L_k$$

ACO Algorithm for TSP

// Initialize pheromone trails
for (every edge i, j) {
$\tau = \tau_0$
ng town for every ant
++) {
on a randomly chosen city
t tour and length
our found from the beginning
the best tour

// Initialize pheromone trails
for (every edge i, j) {
$\tau = \tau_0$
}
// Choose the starting town for every ant
for ($k = 1; k \leq m; k{+}{+}$) {
Place ant k on a randomly chosen city
}
// Initialize the best tour and length
T^+ = the shortest tour found from the beginning
L^+ = the length of the best tour
// Main loop
for ($t = 1; t \leq Tmax; t{+}{+}$) {
// Compute a tour for every ant
for ($k = 1; k \leq m; k{+}{+}$) {
Build tour $T_k(t)$ by applying $n - 1$ times the following step:
Choose the next node (city) j with the probability
$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}(t)]^\beta}$, if $j \in J$
$p_{ij}^k(t) = 0$, if $j \notin J$
where i is the current city.
}
// Compute the tour lengths for all ants
for ($k = 1; k \leq m; k{+}{+}$) {
Compute the length $L_k(t)$ of the tour $T_k(t)$ produced by ant k
}
// Update the best tour if an improved tour is found
if (an improved tour is found) {
Update T^+ and L^+
print T^+ and L^+
}

// Global update for the pheromone trails
for (every edge i, j) {
Update the pheromone trails by applying the rule:
$\tau_{ij}(t) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) + e \cdot \tau_{ij}^e(t)$, where
$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t)$
$\Delta\tau_{ij}^k(t) = \left\{ \frac{Q/L^k(t), if (i,j) \in T^k(t)}{0, otherwise} \right\}$
and
$\tau_{ij}^e(t) = \left\{ \frac{Q/L^+ \ if (i,j) \in T^+}{0, otherwise} \right\}$
}
// Calculate the intensity of the pheromone for next iteration
for (every edge i, j) {
$\tau_{ij}(t + 1) = \tau_{ij}(t)$
}
}

$_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t)$
$_{ij}^k(t) = \left\{ \frac{Q/L^k(t), if (i,j) \in T^k(t)}{0, otherwise} \right\}$
d
$(t) = \left\{ \frac{Q/L^+ \ if (i,j) \in T^+}{0, otherwise} \right\}$
the intensity of the pheromone for next iteration
edge i, j) {
$_{ij}(t + 1) = \tau_{ij}(t)$

Advantage and Disadvantages

- For TSPs (Traveling Salesman Problem), relatively efficient
  - for a small number of nodes, TSPs can be solved by exhaustive search
  - for a large number of nodes, TSPs are very computationally difficult to solve (NP-hard) – exponential time to convergence

- Performs better against other global optimization techniques for TSP (neural net, genetic algorithms, simulated annealing)

- Compared to GAs (Genetic Algorithms):
  - retains memory of entire colony instead of previous generation only
  - less affected by poor initial solutions (due to combination of random path selection and colony memory)

Advantage and Disadvantages

- Can be used in dynamic applications (adapts to changes such as new distances, etc.)
- Has been applied to a wide variety of applications
- As with GAs, good choice for constrained discrete problems (not a gradient-based algorithm)

- Theoretical analysis is difficult:
  - Due to sequences of random decisions (not independent)
  - Probability distribution changes by iteration
  - Research is experimental rather than theoretical
- Convergence is guaranteed, but time to convergence uncertain

Advantage and Disadvantages

- Tradeoffs in evaluating convergence:
  - In NP-hard problems, need high-quality solutions quickly – focus is on quality of solutions
  - In dynamic network routing problems, need solutions for changing conditions – focus is on effective evaluation of alternative paths

- Coding is somewhat complicated, not straightforward
  - Pheromone "trail" additions/deletions, global updates and local updates
  - Large number of different ACO algorithms to exploit different problem characteristics

References

[1]. Thomas STÄUTZLEz and Marco DORIGO. ACO Algorithms for the Traveling Salesman Problem

[2]. Marco Dorigo. The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances, Technical Report IRIDIA-2000-32

[3]. Eric Bonabeau , Marco Dorigo, Guy Theraulaz. Swarm Intelligence - From Natural to Artificial Systems

[4] Prasanna BALAPRAKASH. Ant Colony Optimization under Uncertainty. IRIDIA – Technical Report Series,Technical Report No.TR/IRIDIA/2005-028 November 2005

[5]. Marco Dorigo, *Member, IEEE,* Vittorio Maniezzo and Alberto Colorni. The Ant System:Optimization by a colony of cooperating agents

[6] Michael Guntsch Jürgen Branke. Hand book of Bio-inspired Algorithm and Application: Ant Colony Optimization

[7] Mohamed Belal, Jafaar Gaber, Hoda El-Sayed,Abdullah Almojel. Hand book of Bio-inspired Algorithm and Application: Swarm Intelligence

References

[8]. Jakob Kierkegaard & Jean-Luc Ngassa. ACO and TSP, 29th of May, 2007

[9]. Jürgen Branke, Michael Stein, Hartmut Schmeck. A Unified View on Metaheuristics and Their Hybridization

[10]. Borut Robiˇc, Peter Korošec, Jurij Šilc. Ant Colonies and the Mesh-Partitioning Problem

[11]. Marco Dorigo, Luca Maria Gambardella. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, TR/IRIDIA/1996-5

[12]. Marco Dorigo, *Member, IEEE,* Vittorio Maniezzo and Alberto Colorni. The Ant System: Optimization by a colony of cooperating agents

[13]. M. Dorigo and Gianni Di Caro. The Ant colony optimization Meta-heuristic

[14]. Fred Glover, Gari A. Kochenberger. Hand book of Metaheuristic

[15]. Yoshiyuki Nakamichi, Takaya Arita. Diversity control in ant colony optimization

[16]. Jade Herbots, Willy Herroelen,Roel Leus. Experimental Investigation of the Applicability of Ant Colony Optimization Algorithms for Project Scheduling, November 26, 2004

References

[17]. Mohammed Al-Fayoumi, 2P .Mahanti and 3Soumya Banerjee. OptiTest: Optimizing Test Case Using Hybrid Intelligence. Proceedings of the World Congress on Engineering 2007 Vol I

WCE 2007, July 2 - 4, 2007, London, U.K.

[18]. Dario Floreano and Claudio Mattiussi. Bio-Inspired Artificial Intelligence: THEORIES, METHODS, AND TECHNOLOGIES

[19]. *Pierre Delisle, Michaël Krajecki, Marc Gravel, Caroline Gagné.* PARALLEL IMPLEMENTATION OF AN ANT COLONY OPTIMIZATION METAHEURISTIC WITH OPENMP

[20]. *Vittorio Maniezzo, Luca Maria Gambardella, Fabio de Luigi.* Ant Colony Optimization

[21]. Sorin C. Negulescu, Claudiu V. Kifor, Constantin Oprean. Ant Colony Solving Multiple Constraints Problem: Vehicle Route Allocation. Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844 Vol. III (2008), No. 4, pp. 366-373

[22]. Christian Blum. Review Ant colony optimization: Introduction and recent trends *ALBCOM, LSI, Universitat Politècnica de Catalunya, Jordi Girona 1-3, Campus Nord, 08034 Barcelona, Spain* Accepted 11 October 2005

References

[23]. James Lin and David Ansari. Instruction Scheduling – Variation of Max-Min Ant Colony System Optimization Parameters

[24]. Marco Dorigo and Thomas Stützle: Ant Colony Optimization. MIT Press, Cambridge, MA, 2004.

[25]. LinQuan Xie and HongBiao Mei. The Application of the Ant Colony Decision Rule Algorithm on Distributed Data Mining

[26]. Gianni Di Caro. Ant Colony Optimization and its Application to Adaptive Routing in Telecommunication Networks, September 2004

[27]. Saad Ghaleb Yaseen andNada M. A.AL-Slamy. Ant Colony Optimization, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008 351

[28]. T. Stuitzle and H. Hoos. Improving the ant system: a detail report on MAX-MIN ant system

[29]. Caroline Herssens, Amin Mantrach and Marco Saerens. Ant Colony Optimization Revisited from a Randomized Shortest Path Perspective

[30]. Osvaldo G´omez and Benjam´ın Bar´anOmicron ACO. A New Ant Colony Optimization Algorithm

References

[31]. Gianni A. Di Caro, Frederick Ducatelle and Luca M. Gambardella. Theory and practice of Ant Colony Optimization for routing in dynamic telecommunications networks

[32]. *Vittorio Maniezzo, Luca Maria Gambardella, Fabio de Luigi.* Ant Colony Optimization

[33]. *A. Ketabi, and R. Feuillet.* Ant Colony Search Algorithm for Optimal Generators Start-up during Power System Restoration

[34]. J. Dréo, P. Siarry. Continuous interacting ant colony algorithm based on dense heterarchy

[35]. C. Blum and M. Sample. ACO for FOP shop scheduling: A case study on different pheromone representation

[36]. Jun Zhang, Xiaomin Hu, X. Tan, J.H. Zhong and Q. Huang. Implementation of an Ant Colony Optimization technique for job shop scheduling problem

[37]. Marcin L. Pilat and Tony White. Using Genetic Algorithms to optimize ACS-TSP

[38]. Daniel Angus Ant Colony Optimization: From Biological Inspiration to an Algorithmic Framework, Swinburne University of Technology, Melbourne, Australia, April 21, 2006

References

[39]. *Vittorio Maniezzo, Anthonilla and Carbonaro.* Ant colony optimization: an overview

[40]. Hozefa M. Botee and Eric Bonabeauy. Evolving Ant Colony Optimization, Santa Fe Institute
1399 Hyde Park Road, Santa Fe, NM 87501, USA

[41]. Hongcheng Zeng, Timo Pukkala, Heli Peltola and Seppo Kellomäki Application of Ant Colony Optimization for the Risk Management of Wind Damage in Forest Planning

[42]. XIAO Jie , ZHOU Ze-kui, ZHANG Guang-xin. Ant colony system algorithm for the optimization of beer fermentation control, *China,* revision accepted Oct. 12, 2003

[43]. Joon-Woo Lee, Jeong-Jung Kim, Byoung-Suk Choi, and Ju-Jang Lee. Improved Ant Colony Optimization Algorithm by Potential Field Concept for Optimal Path Planning

[44]. Bin Yu, ZhongzhenYang, Chuntian Cheng and Chong Liu. OPTIMIZING BUS TRANSIT NETWORK WITH PARALLELANT COLONY ALGORITHM

[45]. Hao Mei, Yantao Tian*, Linan Zu. A Hybrid Ant Colony Optimization Algorithm for Path Planning of Robot in Dynamic Environment1

[46]. Chia-Ho CHEN. A HYBRID ANT COLONY SYSTEM FOR VEHICLE ROUTING PROBLEM WITH TIME WINDOWS, Society for Transportation Studies, Vol. 6, pp. 2822 - 2836, 2005

References

[47]. Carlos A. Silva and Thomas A. Runkler, Ant Colony Optimization for dynamic Traveling Salesman Problems

[48]. Tony White, Simon Kaegi, Terri Oda, Revisiting Elitism in Ant Colony Optimization

[49]. Stephen Gilmour and Mark Dras. Understanding the Pheromone System within Ant Colony Optimization

[50]. Mesut G¨unes¸, Udo Sorges, Imed Bouazizi. ARA – The Ant-Colony Based Routing Algorithm for MANETs

[51]. Yanjun Li and Tie-Jun WU. A nested Ant Colony Algorithms for hybrid production scheduling

[52]. Ajay C Solai Jawahar . Ant Colony Optimization for Mobile Ad-hoc Networks

[53]. Ryan M. Garlick and Richard S. Barr. Dynamic Wavelength Routing in WDM Networks via Ant Colony Optimization

[54]. Siriluck Lorpunmanee, Mohd Noor Sap, Abdul Hanan Abdullah, and Chai Chompoo-inwai. An Ant Colony Optimization for Dynamic Job Scheduling in Grid Environment

[55]. Nicolas Durand and Jean-Marc Alliot. Ant Colony Optimization for Air Traffic Conflict Resolution DTI R&D/POM

References

[56]. Rafael S. Parpinelli, Heitor S. Lopes and Alex A. Freitas. An Ant Colony Algorithm for Classification Rule Discovery

[57]. Thomas Weise. Global Optimization Algorithm Version: 2009-06-26

[58]. Jamaludin sallimr, wan Muhammad Syahrir wan Hussin, Rosni Abdullah, Ahamad Tajudin Abdul Khadera. A Background Study on Ant Colony Optimization Metaheuristic and its Application Principles in Resolving Three Combinatorial Optimization Problems

[59]. In`es Alaya, Christine Solnon and Khaled Gh´edira. ANT ALGORITHM FOR THE MULTIDIMENSIONAL KNAPSACK PROBLEM

[60]. Ehsan Salari and Kourosh Eshghi, An ACO Algorithm for the Graph Coloring Problem Int. J. Contemp. Math. Sciences, Vol. 3, 2008, no. 6, 293 - 304

[61]. Rafael S. Parpinelli1, Heitor S. Lopes, and Alex A. Freitas. Data Mining with an Ant Colony Optimization Algorithm

[62]. Fangqing Zhao1, Fanggeng Zhao2, Tao Li1 and Donald A. Bryant. A new pheromone trail-based genetic algorithm for comparative genome assembly Published online 29 April 2008 Nucleic Acids Research, 2008, Vol. 36, No. 10 3455–3462

References

[63]. Mauro Birattari¤ Prasanna Balaprakash¤ Marco Dorigo. ACO/F-Race: Ant Colony Optimization and Racing Techniques for Combinatorial Optimization Under Uncertainty, MIC2005.

[64]. Jeong-Jun Suh, Shan Guo Quan, Seung-Hoon Park, and Young Yong Kim. Adaptive File Distribution in P2P Network Using Ant Colony Optimization for Smart Home Environment

[65]. MARCO DORIGO and THOMAS ST¨UTZLE. An Experimental Study of the Simple Ant Colony Optimization Algorithm

[66]. Veronica Lopez, Jose A. Gamez, and Luis delaOssa. Improvement of a car racing controller by means of Ant Colony Optimization algorithms

[67]. *Mostafa Abd-El-Barr, Sadiq M. Sait, Bambang A. B. Sarif, Uthman Al-Saiari.* COMBINATIONAL LOGIC CIRCUITS DESIGN THROUGH ANT COLONY OPTIMIZATION ALGORITHM

[68]. Xingguo Chen, Hao Wang, Weiwei Wang, Yinghuan Shi, and Yang Gao.Apply Ant Colony Optimization to Tetris

[69]. Wong, L.-H., & Looi. Adaptable Learning Pathway Generation with Ant Colony Optimization

Wong, L.-H., & Looi, C.-K. (2009).

References

[63]. Mauro Birattari¤ Prasanna Balaprakash¤ Marco Dorigo. ACO/F-Race: Ant Colony Optimization and Racing Techniques for Combinatorial Optimization Under Uncertainty, MIC2005.

[64]. Jeong-Jun Suh, Shan Guo Quan, Seung-Hoon Park, and Young Yong Kim. Adaptive File Distribution in P2P Network Using Ant Colony Optimization for Smart Home Environment

[65]. MARCO DORIGO and THOMAS ST¨UTZLE. An Experimental Study of the Simple Ant Colony Optimization Algorithm

[66]. Veronica Lopez, Jose A. Gamez, and Luis delaOssa. Improvement of a car racing controller by means of Ant Colony Optimization algorithms

[67]. *Mostafa Abd-El-Barr, Sadiq M. Sait, Bambang A. B. Sarif, Uthman Al-Saiari.* COMBINATIONAL LOGIC CIRCUITS DESIGN THROUGH ANT COLONY OPTIMIZATION ALGORITHM

[68]. Xingguo Chen, Hao Wang, Weiwei Wang, Yinghuan Shi, and Yang Gao.Apply Ant Colony Optimization to Tetris

[69]. Wong, L.-H., & Looi. Adaptable Learning Pathway Generation with Ant Colony Optimization

Wong, L.-H., & Looi, C.-K. (2009).

References

[70]. Kwang Mong Sim and Weng Hong Sun, *Member, IEEE.* Ant Colony Optimization for Routing and Load-Balancing: Survey and New Directions

[71]. Hua Chen and Albert M. K. Cheng. Applying Ant Colony Optimization to the Partitioned Scheduling Problem for Heterogeneous Multiprocessors

[72]. Cristian Martinez. An ACO algorithm for image compression, October 29, 2006

[73]. Aruna A. Priyanto, Adiwijaya, W. Maharani. IMPLEMENTATION OF ANT COLONY OPTIMIZATION ALGORITHM ON THE PROJECT RESOURCE SCHEDULING PROBLEM

[74]. Philip Christian C. Zuniga, Maia Malonzo, Henry Adorna and Prospero Naval. An Ant Colony Optimization Algorithm for the Network Inference and Parameter Estimation of S-Systems International Conference on Molecular Systems Biology 2009

[75]. Mehrabi, Saeed Mehrabi† and Abbas Mehrabi. Ali D. A PRUNING BASED ANT COLONY ALGORITHM FOR MINIMUM VERTEX COVER PROBLEM

[76]. Marc REIMANN ANALYZING A VEHICLE ROUTING PROBLEM WITH STOCHASTIC DEMANDS USING ANT COLONY OPTIMIZATION

References

[77]. P. Jaganathan1*, K.Thangavel2 A. Pethalakshmi3, M. Karnan4 -Classification Rule Discovery with Ant Colony Optimization and Improved Quick Reduct Algorithm

[78]. Vittorio Maniezzo, Matteo Roffilli. VERY STRONGLY CONSTRAINED PROBLEMS:
AN ANT COLONY OPTIMIZATION APPROACH

[79]. Chung-wen Chiang, Yi-chan Lee, Chungnan Lee and Ta-yuan Chou. Ant Colony Optimization for Task Matching and Scheduling

[80]. Joao Pedro, Joao Pires, and Joao Paulo Carvalho . Ant Colony Optimization for Distributed Routing Path Optimization in Optical Burst-Switched Networks

[81]. Pan Junjie and Wang Dingwei. An Ant Colony Optimization Algorithm for Multiple Travelling Salesman Problem

[82]. Carlos M. Fernandes, Agostinho C. Rosa and Vitorino Ramos. Binary Ant Algorithm

[83]. Malika Bessedik, Rafik Laib, Aissa Boulmerka et Habiba Drias. Ant Colony System for Graph Coloring Problem

References

[84]. TAN Guan-Zheng1;3 HE Huan2 SLOMAN Aaron. Ant Colony System Algorithm for Real-Time Globally Optimal Path Planning of Mobile Robots

[85]. Sofie Demeyer,* Marc De Leenheer, Jurgen Baert, Mario Pickavet, and Piet Demeester Ant colony optimization for the routing of jobs in optical grid networks

[86]. Ying-Shiuan You. Parallel Ant System for Traveling Salesman Problem on GPUs

[87]. Nada M. A. Al Salami. Ant Colony Optimization Algorithm

[88]. Holger R. Maier1; Angus R. Simpson2; Aaron C. Zecchin3; Wai Kuan Foong4; Kuang Yeow Phang5;Hsin. Ant Colony Optimization for Design of Water Distribution Systems

[89]. Karl O. Jones, André Bouffet. COMPARISON OF BEES ALGORITHM, ANT COLONY OPTIMISATION AND PARTICLE SWARM OPTIMISATION FOR PID CONTROLLER TUNING

[90]. Farzaneh Jalalinejad, Farhang Jalali-Farahania, Navid Mostoufi, Rahmat Sotudeh-Gharebagh Ant Colony Optimization: A Leading Algorithm in Future Optimization of Chemical Processes