Problem: Find the maximum of the function

$$f(x) = -x^2 + 5x + 20 \text{ with } -10 \le x \le 10$$

using the PSO algorithm. Use 9 particles with the initial positions $x_1 = -9.6$, $x_2 = -6$, $x_3 = -2.6$, $x_4 = -1.1$, $x_5 = 0.6$, $x_6 = 2.3$, $x_7 = 2.8$, $x_8 = 8.3$, and $x_9 = 10$. Show the detailed computations for iterations 1, 2 and 3.

Solution:

Step1: Choose the number of particles: $x_1 = -9.6$, $x_2 = -6$, $x_3 = -2.6$, $x_4 = -1.1$, $x_5 = 0.6$, $x_6 = 2.3$, $x_7 = 2.8$, $x_8 = 8.3$, and $x_9 = 10$.

The initial population (i.e. the iteration number $t = 0$) can be represented as $x_i^0, i = 1,2,3,4,5,6,7,8,9$:

$$x_1^0 = -9.6\,,\ x_2^0 = -6,\ x_3^0 = -2.6,$$
$$x_4^0 = -1.1,\ x_5^0 = 0.6,\ x_6^0 = 2.3,$$
$$x_7^0 = 2.8,\quad x_8^0 = 8.3,\ x_9^0 = 10.$$

Evaluate the objective function values as

$$f_1^0 = -120.16,\ f_2^0 = -46,\ f_3^0 = 0.24,$$
$$f_4^0 = 13.29,\ f_5^0 = 22.64\,,\ f_6^0 = 26.21,$$
$$f_7^0 = 26.16,\ f_8^0 = -7.39,\ f_9^0 = -30.$$

Let $c_1 = c_2 = 1$.

Set the initial velocities of each particle to zero:
$v_i^0 = 0, i.e. v_1^0 = v_2^0 = v_3^0 = v_4^0 = v_5^0 = v_6^0 = v_7^0 = v_8^0 = v_9^0 = 0.$

Step2: Set the iteration number as $t = 0 + 1 = 1$ and go to step 3.

Step3: Find the personal best for each particle by

$$P_{best,i}^{t+1} = \begin{cases} P_{best,i}^{t} & \text{if } f_i^{t+1} > P_{best,i}^{t} \\ x_i^{t+1} & \text{if } f_i^{t+1} \le P_{best,i}^{t} \end{cases}$$

So,

$P^1_{best,1} = -9.6$, $P^1_{best,2} = -6$, $P^1_{best,3} = -2.6$,

$P^1_{best,4} = -1.1$, $P^1_{best,5} = 0.6$, $P^1_{best,6} = 2.3$,

$P^1_{best,7} = 2.8$, $P^1_{best,8} = 8.3$, $P^1_{best,9} = 10$.

Step4: Find the global best by

$$G_{best} = \min\{P^t_{best,i}\} \text{ where } i = 1,2,3,4,5,6,7,8,9.$$

Since, the maximum personal best is $P^1_{best,6} = 2.3$, thus $G_{best} = 2.3$.

Step5: Considering the random numbers in the range (0, 1) as $r^1_1 = 0.213$ and $r^1_2 = 0.876$, and find the velocities of the particles by

$$v^{t+1}_i = v^t_i + c_1 r^t_1\left[P^t_{best,i} - x^t_i\right] + c_2 r^t_2 [G^t_{best} - x^t_i];\ i = 1, \ldots ,9.$$

so

$v^1_1 = 0 + 0.213(-9.6 + 9.6) + 0.876(2.3 + 9.6) = 10.4244$

$v^1_2 = 7.2708$, $v^1_3 = 4.2924$, $v^1_4 = 2.9784$, $v^1_5 = 1.4892$,

$v^1_6 = 0$, $v^1_7 = -0.4380$, $v^1_8 = 5.256$, $v^1_9 = -6.7452$.

Step6: Find the new values of $x^1_i, i = 1, \ldots ,9$ by

$$x^{t+1}_i = x^t_i + v^{t+1}_i$$

So

$x^1_1 = 0.8244$, $x^1_2 = 1.2708$, $x^1_3 = 1.6924$,

$x^1_4 = 1.8784$, $x^1_5 = 2.0892$, $x^1_6 = 2.3$,

$x^1_7 = 2.362$, $x^1_8 = 3.044$, $x^1_9 = 3.2548$.

Step7: Find the objective function values of x_i^1 , $i = 1, \dots, 9$:

$$f_1^1 = 23.4424, f_2^1 = 24.7391, f_3^1 = 25.5978,$$
$$f_4^1 = 25.8636, f_5^1 = 26.0812 , f_6^1 = 26.21,$$
$$f_7^1 = 26.231, f_8^1 = 25.9541, f_9^1 = 25.6803.$$

Step 8: Stopping criterion:

If the terminal rule is satisfied, go to step 2,
Otherwise stop the iteration and output the results.

Step2: Set the iteration number as $t = 1 + 1 = 2$, and go to step 3.

Step3: Find the personal best for each particle.

$$P_{best,1}^2 = 0.8244, P_{best,2}^2 = 1.2708, P_{best,3}^2 = 1.6924,$$
$$P_{best,4}^2 = 1.8784, P_{best,5}^2 = 2.0892, P_{best,6}^2 = 2.3,$$
$$P_{best,7}^2 = 2.362, P_{best,8}^2 = 3.044, P_{best,9}^2 = 3.2548.$$

Step4: Find the global best.

$$G_{best} = 2.362 .$$

Step5: By considering the random numbers in the range (0, 1) as $r_1^2 = 0.113$ and $r_2^2 = 0.706$, find the velocities of the particles by

$$v_i^{t+1} = v_i^t + c_1 r_1^t \left[P_{best,i}^t - x_i^t\right] + c_2 r_2^t [\, G_{best} - x_i^t]; \; i = 1, \dots, 9.$$

so

$$v_1^2 = 11.5099, v_2^2 = 8.0412, v_3^2 = 4.7651,$$
$$v_4^2 = 3.3198, v_5^2 = 1.6818, v_6^2 = 0.0438,$$
$$v_7^2 = -0.4380, v_8^2 = -5.7375, v_9^2 = -7.3755.$$

Step6: Find the new values of x_i^2 , $i = 1, \dots, 9$ by

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

so

$x_1^2 = 12.3343$, $x_2^2 = 9.312$, $x_3^2 = 6.4575$,

$x_4^2 = 5.1982$, $x_5^2 = 3.7710$, $x_6^2 = 2.3438$,

$x_7^2 =1.9240$, $x_8^2 = -2.6935$, $x_9^2 = -4.1207$.

Step7: Find the objective function values of $f_i^2, i = 1, \dots, 9$:

$f_1^2 = -70.4644$, $f_2^2 = -20.1532$, $f_3^2 = 10.5882$,

$f_4^2 = 18.9696$, $f_5^2 = 24.6346$, $f_6^2 = 26.2256$,

$f_7^2 = 25.9182$, $f_8^2 = -0.7224$, $f_9^2 = -17.5839$.

Step 8: Stopping criterion:

If the terminal rule is satisfied, go to step 2,
Otherwise stop the iteration and output the results.

Step2: Set the iteration number as $t = 1 + 1 = 3$, and go to step 3.

Step3: Find the personal best for each particle.

$$P^3_{best,1} = 0.8244, P^3_{best,2} = 1.2708, P^3_{best,3} = 1.6924,$$
$$P^3_{best,4} = 1.8784, P^3_{best,5} = 2.0892, P^3_{best,6} = 2.3438,$$
$$P^3_{best,7} = 2.362, P^3_{best,8} = 3.044, \quad P^3_{best,9} = 3.2548.$$

Step4: Find the global best.

$$G_{best} = 2.362$$

Step5: By considering the random numbers in the range (0, 1) as
$r^3_1 = 0.178$ and $r^3_2 = 0.507$, find the velocities of the particles by

$$v^{t+1}_i = v^t_i + c_1 r^t_1 [P^t_{best,i} - x^t_i] + c_2 r^t_2 [G_{best} - x^t_i]; \; i = 1, \ldots, 9.$$

so

$$v^3_1 = 4.4052, v^3_2 = 3.0862, v^3_3 = 1.8405,$$
$$v^3_4 = 1.2909, v^3_5 = 0.6681, v^3_6 = 0.053,$$
$$v^3_7 = -0.1380, v^3_8 = -2.1531, v^3_9 = -2.7759.$$

Step6: Find the new values of x^3_i, $i = 1, \ldots, 9$ by

$$x^{t+1}_i = x^t_i + v^{t+1}_i$$

so

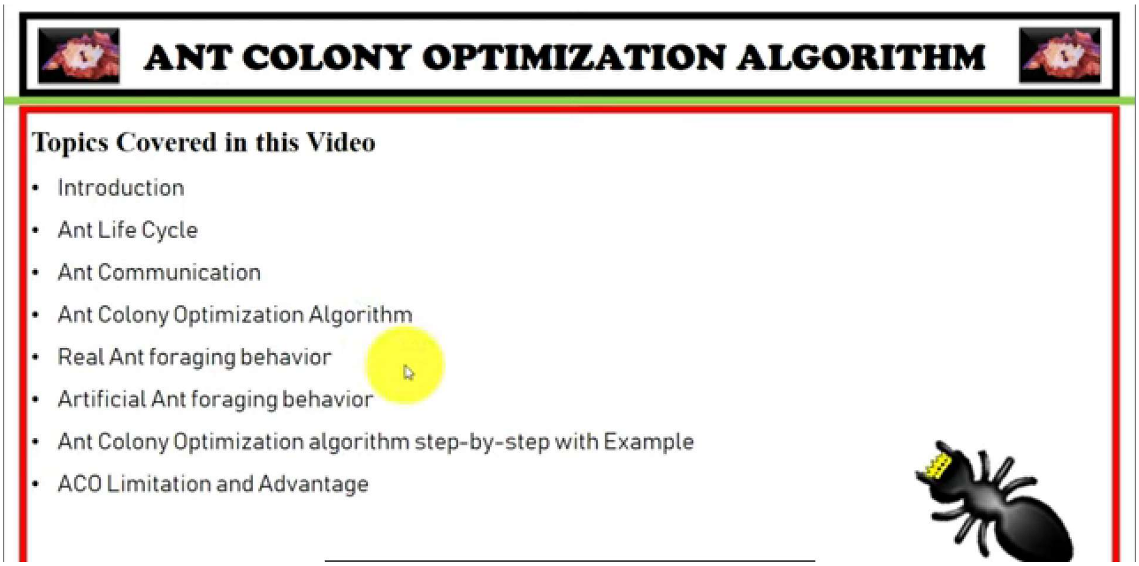
$$x^3_1 = 16.7395, x^3_2 = 12.3982, x^3_3 = 8.298,$$
$$x^3_4 = 6.4892, x^3_5 = 4.4391, x^3_6 = 2.3968,$$
$$x^3_7 = 1.786, x^3_8 = -4.8466, x^3_9 = -6.8967.$$

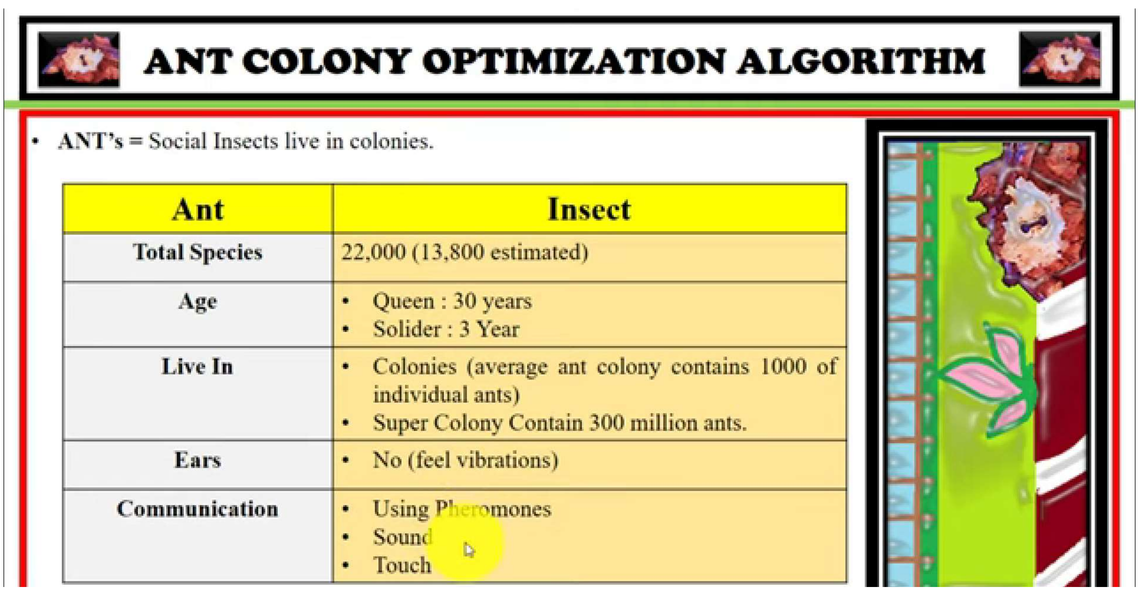
Step7: Find the objective function values of $f^3_i, i = 1, \ldots, 9$:

Step 8: Stopping criterion:

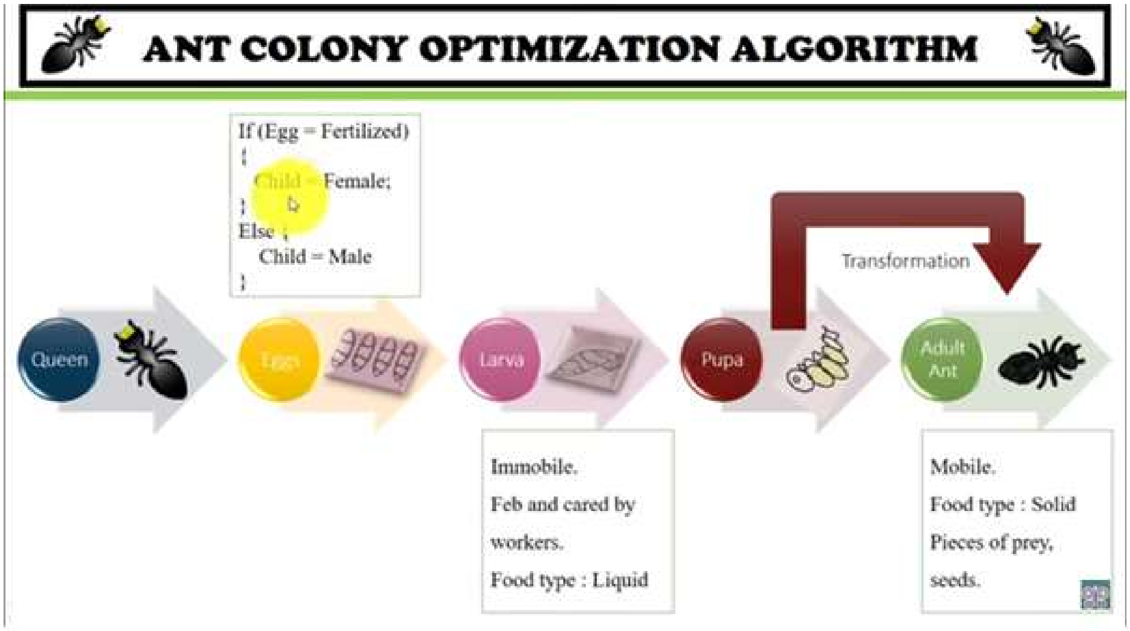
If the terminal rule is satisfied, go to step 2,
Otherwise stop the iteration and output the results.

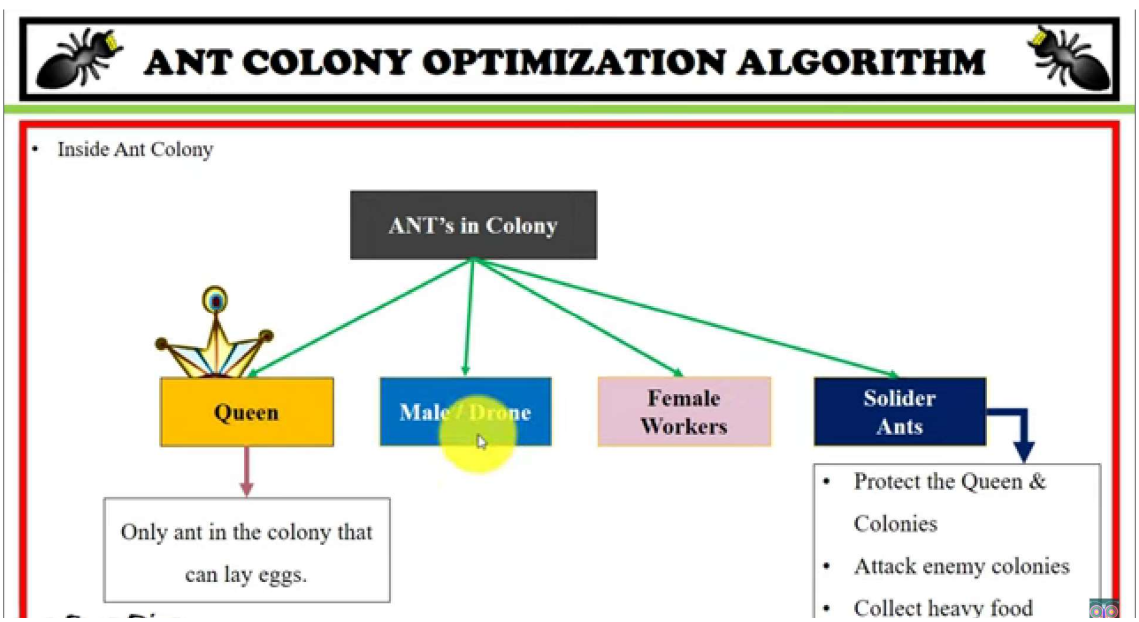
Finally, the values of $x_i^2, i = 1,2,3,4,5,6,7,8,9$ did not converge, so we increment the iteration number as $t = 4$ and go to step 2. When the positions of all particles converge to similar values, then the method has converged and the corresponding value of x_i^t is the optimum solution. Therefore the iterative process is continued until all particles meet a single value.





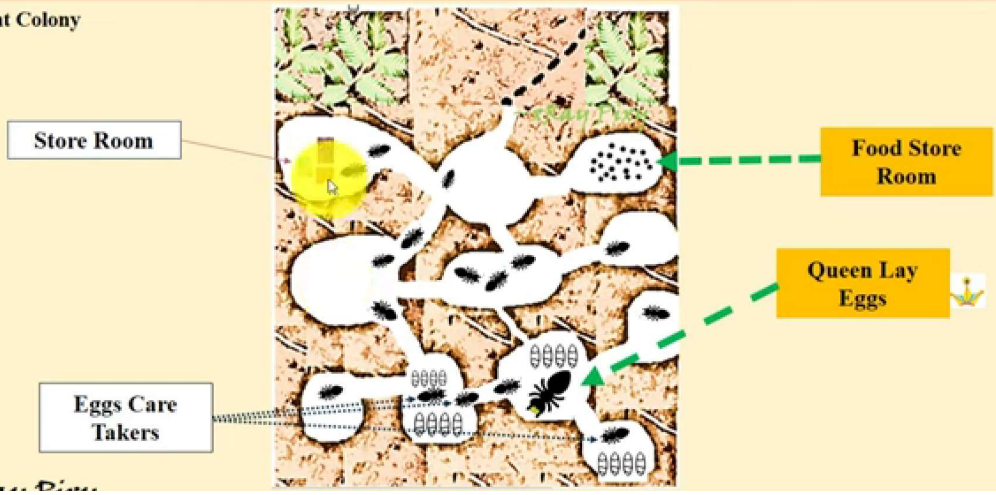
Ant	Insect
Total Species	22,000 (13,800 estimated)
Age	• Queen : 30 years • Solider : 3 Year
Live In	• Colonies (average ant colony contains 1000 of individual ants) • Super Colony Contain 300 million ants.
Ears	• No (feel vibrations)
Communication	• Using Pheromones • Sound • Touch


ANT COLONY OPTIMIZATION ALGORITHM
If (Egg = Fertilized)
{
Child = Female;
}
Else {
Child = Male
}
Transformation
Queen
Eggs
Larva
Pupa
Adult Ant
Immobile.
Feb and cared by workers.
Food type : Liquid
Mobile.
Food type : Solid
Pieces of prey, seeds.


ANT COLONY OPTIMIZATION ALGORITHM
• Inside Ant Colony
ANT's in Colony
Queen
Male / Drone
Female Workers
Solider Ants
Only ant in the colony that can lay eggs.
• Protect the Queen & Colonies
• Attack enemy colonies
• Collect heavy food

ANT COLONY OPTIMIZATION ALGORITHM

- **Ant Colony**



ANT COLONY OPTIMIZATION ALGORITHM

ANT Communication :

***#1.* How Ant's Communicate with each other?**

- **Ant's** can easily communicate with each other using **Pheromones**.

***#2.* Explain Pheromones?**

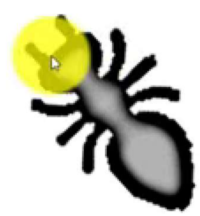
- Chemical Signals
- Used by ants for communication in the environment.
- Ant's release pheromones in Danger (to alert other ants/ for help).

***#3.* How do ants detect pheromones?**

- Through their mobile Antennas

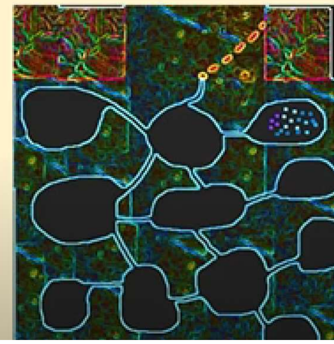
***#4.* How do other ants follow pheromones?**

- Ants leave pheromones on the soil. That can be easily followed by other ants.



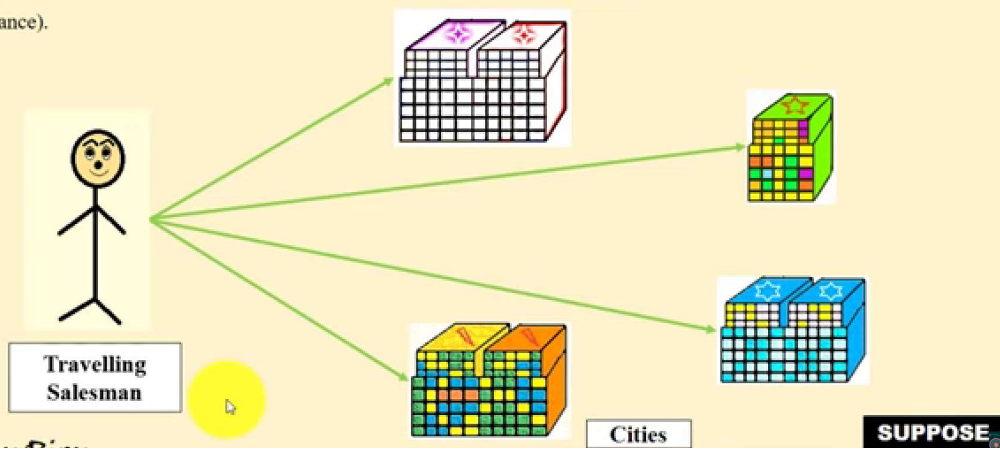
ANT COLONY OPTIMIZATION ALGORITHM

- Ant Colony Optimization (ACO) algorithm is inspired by social behavior of real ants.
- ACO is basically inspired by pheromone based Ant communication.
- ACO is developed by Marco Dorigo in 1922.
- This technique is used to find Optimal paths.
- ACO is used for Routing and load balancing problems.
- For Example: Travelling Salesman Problem (TSP).
- ACO can be applied to continuous optimization problems.



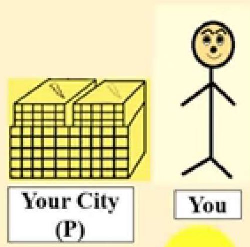
ANT COLONY OPTIMIZATION ALGORITHM

- **In Travelling Salesman Problem (TSP):** Salesman want to visit each city exactly once (for minimum travelling distance).



ANT COLONY OPTIMIZATION ALGORITHM

- In Travelling Salesman Problem (TSP): Salesman want to visit each city exactly once (for minimum travelling distance).
- You = Salesperson



List of Cities you want to visit

1. City X
2. City Y
3. City Z
4. City XYZ

TASK

1. **Find Shortest route.**
2. **Visit each city exactly once.**
3. **Return to the current city.**

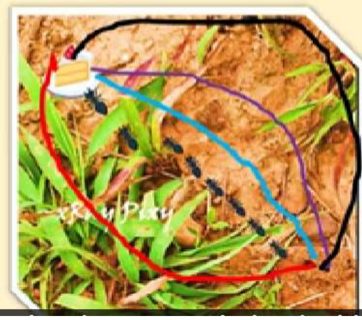
(*distance between all cities is known*)

ANT COLONY OPTIMIZATION ALGORITHM

- **Ant Colony Optimization (ACO)** algorithm is basically inspired by the foraging behavior of ants searching for suitable paths between their colonies and food source.

ANT COLONY OPTIMIZATION ALGORITHM

How do ants find out the shortest path between Nest and Food?



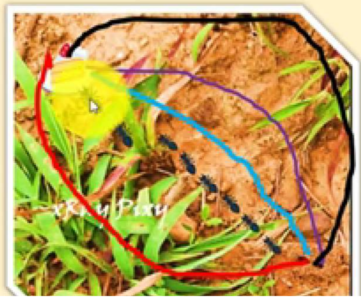
ANT COLONY OPTIMIZATION ALGORITHM

- **Ant Colony Optimization (ACO)**
- Ants communicate with each other indirectly using a chemical known as a **pheromone**.
- With the help of Pheromone signals, ants can easily find shortest path between Nest/Colony and Food.

ANT COLONY OPTIMIZATION ALGORITHM

- **Ant Colony Optimization (ACO)**
- ANTS Foraging Behavior

When ants find food source they choose the way have strong pheromones.



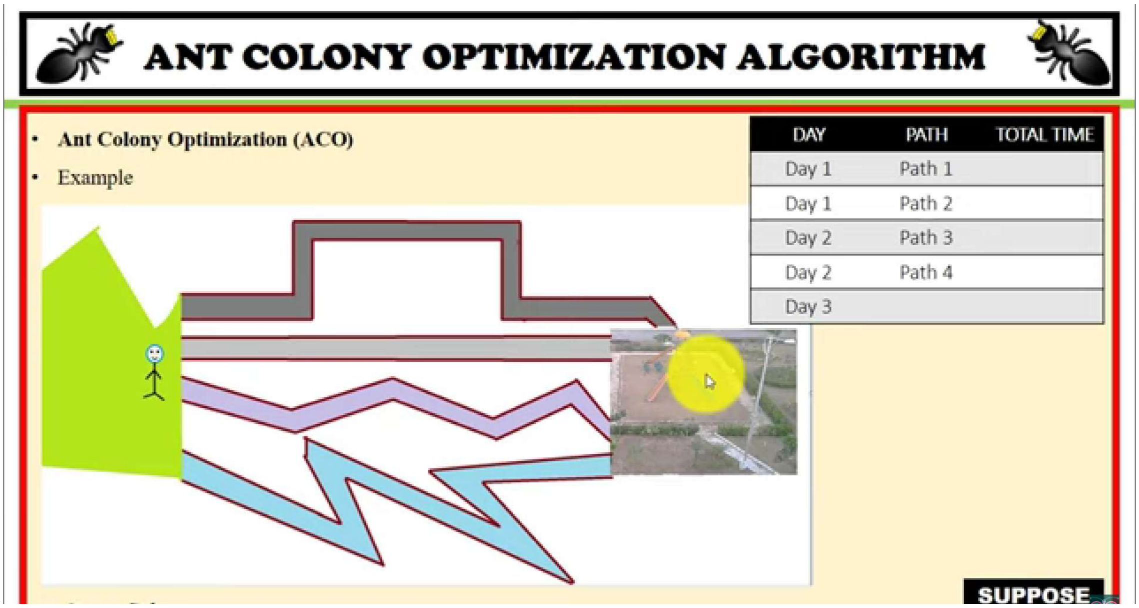
ANT COLONY OPTIMIZATION ALGORITHM

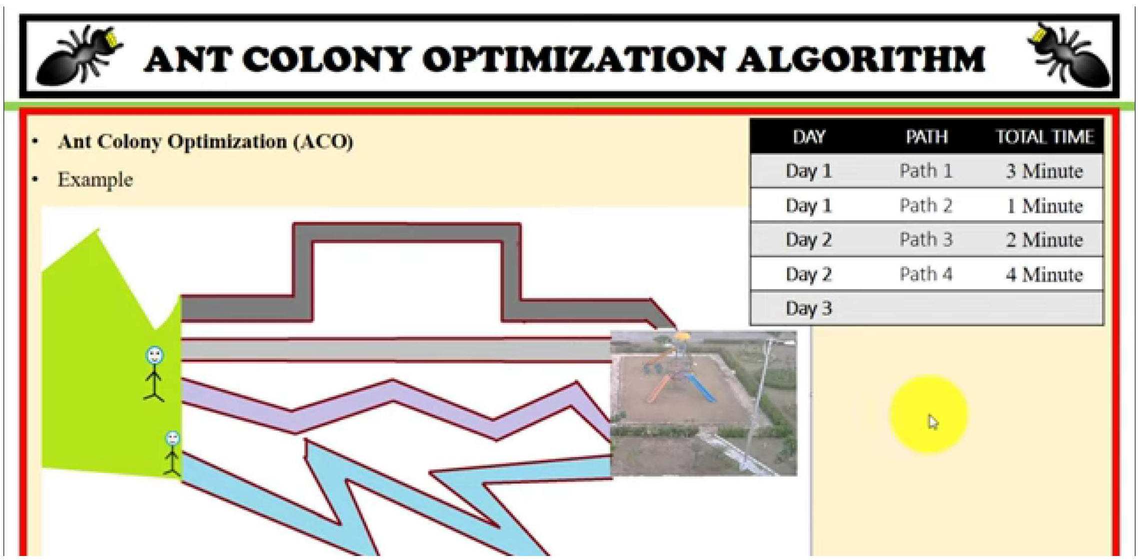
ANT Communication EXAMPLE:

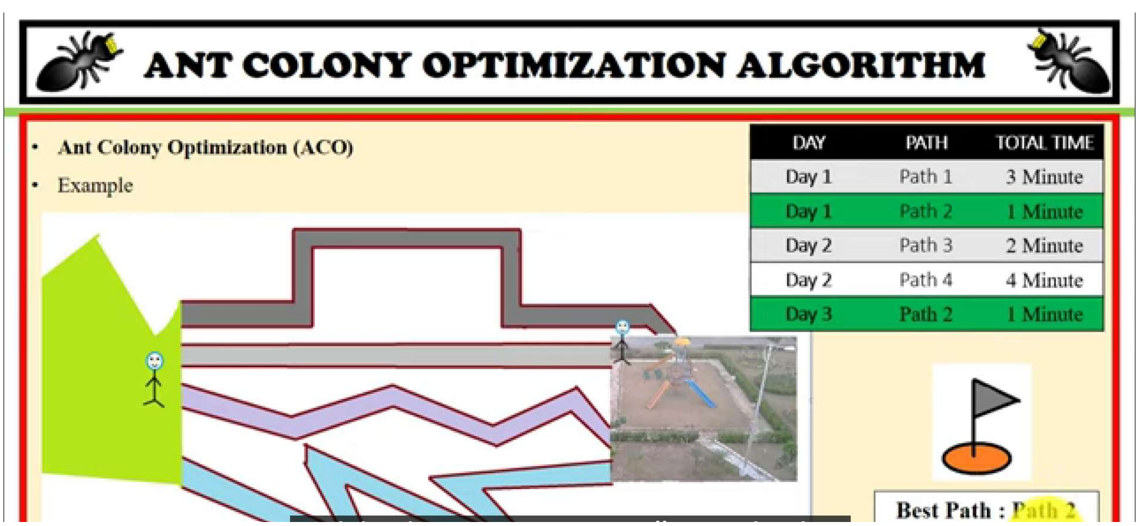
#1. Forger will marks trails on the way using pheromones while going back to the colony.

Followed by other ants.

Food Source


ANT COLONY OPTIMIZATION ALGORITHM
• Ant Colony Optimization (ACO)
• Example
DAY PATH TOTAL TIME
Day 1 Path 1
Day 1 Path 2
Day 2 Path 3
Day 2 Path 4
Day 3
SUPPOSE


ANT COLONY OPTIMIZATION ALGORITHM
• Ant Colony Optimization (ACO)
• Example
DAY PATH TOTAL TIME
Day 1 Path 1 3 Minute
Day 1 Path 2 1 Minute
Day 2 Path 3 2 Minute
Day 2 Path 4 4 Minute
Day 3


ANT COLONY OPTIMIZATION ALGORITHM
• Ant Colony Optimization (ACO)
• Example
DAY PATH TOTAL TIME
Day 1 Path 1 3 Minute
Day 1 Path 2 1 Minute
Day 2 Path 3 2 Minute
Day 2 Path 4 4 Minute
Day 3 Path 2 1 Minute
Best Path : Path 2

- **Ant Colony Optimization (ACO)**
- ANTS Foraging Behavior

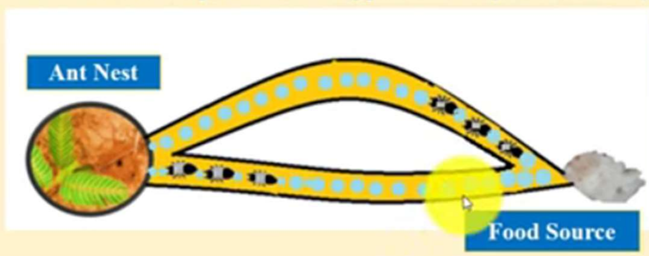
When ants find food source they choose the way have strong pheromones.

CASE 01 : All ants are in the nest. There is no pheromone marks on the ground.

CASE 02 : Foraging start: 50% ants will take shortest path, 50% ants will take longest path.

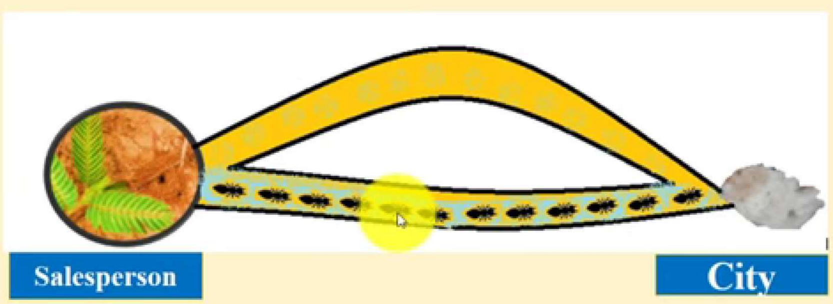
CASE 03 : Ants used shortest path arrive earlier to the food source.

CASE 04 : Pheromone marks on the shorter paths have strong pheromone signals. Probability of this path selected by other ants increases.



ANT COLONY OPTIMIZATION ALGORITHM

- **Ant Colony Optimization (ACO)**



ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

1. Initialize ACO parameters
2. Ant Solution Construction
3. Position Each ant in the stating node.
4. Each ant will select next node by applying state transition rule.
5. Repeat until ant build the best solution, then Compute the fitness value.
6. Update best solution.
7. Apply offline pheromone update.
8. Display the best result.

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

1. Initialize ACO parameters

PopulationSize(k) = 20; % Total number of artificial Ants or Agents.

MaxT = 200; % Maximum number of iterations.

Tau = 0.5; % Pheromone initial value.

Alpha = 1; % Pheromone exponential weight.

Beta = 1; % Pheromone heuristic weight.

rho = 0.05; % Pheromone evaporation rate.

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

1. Initialize ACO parameters

2. Ant Solution Construction

First start main loop:

CurrentIteration = 1 to Maximum Number of iterations

CurrentIteration = 1 to MaxT

CurrentIteration = 1 to 200

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

3. Position Each ant in the stating node.

Here, kth ant move from node i to node j with probability. [transition probability]

$$P_{ij}^{k} = \frac{\left(\tau_{ij}^{\alpha}\right)\left(\eta_{ij}^{\beta}\right)}{\sum_{z\epsilon allowed_i}\left(\tau_{ij}^{\alpha}\right)\left(\eta_{ij}^{\beta}\right)}$$

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

3. Position Each ant in the stating node.

Here, kth ant move from node i to node j with probability. [transition probability]

Total Pheromone deposit by ant on path ij

Ant

Value of Path ij.

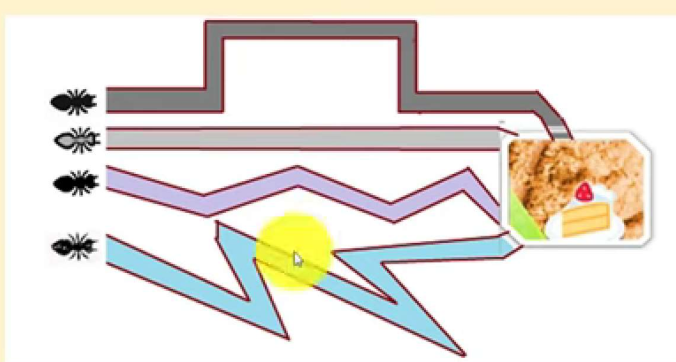
Probability of moving from node i to node j.

$$P_{ij}^{k} = \frac{\left(\tau_{ij}^{\alpha}\right)\left(\eta_{ij}^{\beta}\right)}{\sum_{z\epsilon allowed_i}\left(\tau_{ij}^{\alpha}\right)\left(\eta_{ij}^{\beta}\right)}$$

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm

n = 4; (Population Size)



ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

5. Repeat until ant build the best solution, then Compute the fitness value.

6. Update best solution.

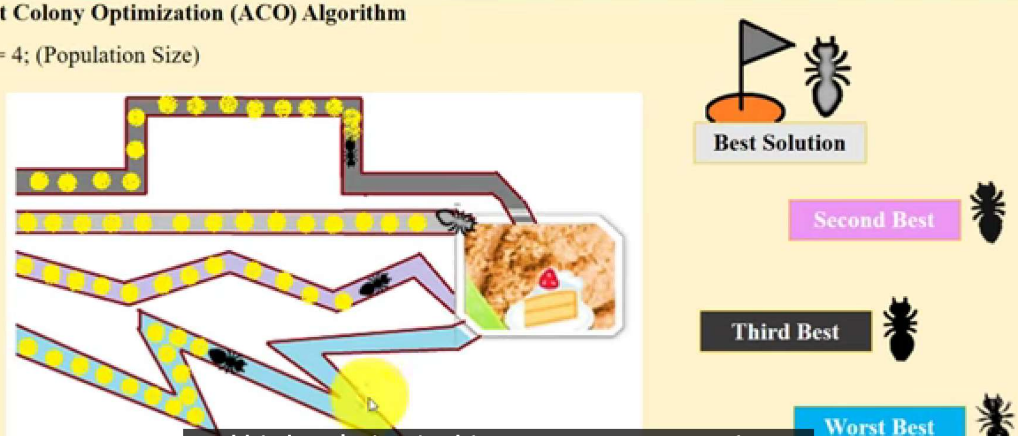
Compare the best solution with each ant solution.

```
If  (Ant(3) Solution < Best Solution)
{
        Consider Ant(3) best solution.
}
```

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm

n = 4; (Population Size)



Ant Colony Optimization (ACO) Algorithm Step-by-Step

7. Apply offline pheromone update.

Note: Pheromone trials are updated when all ants completed their cost/solution.

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

7. Apply offline pheromone update.

Pheromone Updating Equation:

Number of Ants

$$\tau_{ij} \leftarrow (1-\rho)\,\tau_{ij} + \sum_{k}^{m} \Delta\tau_{ij}^{k}$$

Total Pheromone deposit by kth ant on path ij

Total Pheromone deposit by ant on path ij

Pheromone Evaporation

7. Apply offline pheromone update.

Pheromone Updating Equation:

$$\tau_{ij} \leftarrow (1-\rho)\,\tau_{ij} + \sum_{k}^{m} \Delta\tau_{ij}^{k}$$

Constant (Q =1)

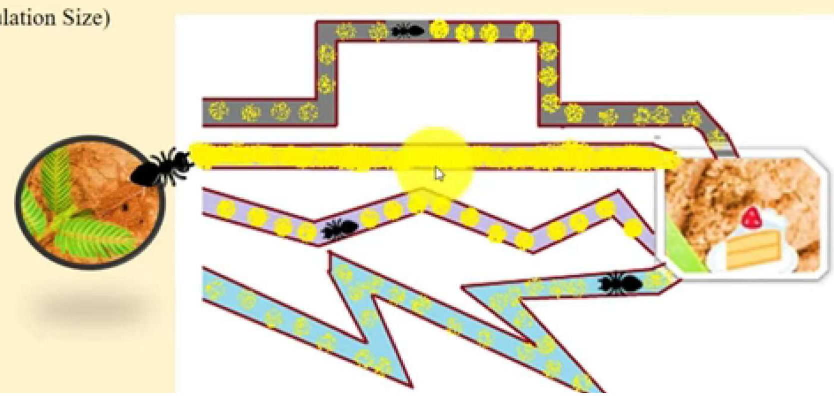
$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{L_k} & \textit{if ant k uses curve ij in its path} \\ 0 & \textit{Otherwise} \end{cases}$$

Path Length

ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm

n = 4; (Population Size)



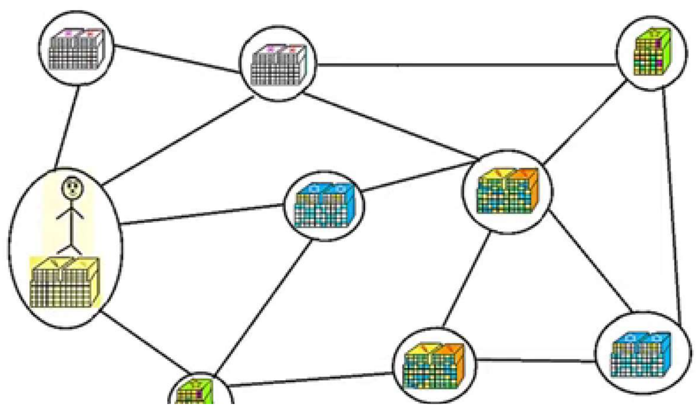
ANT COLONY OPTIMIZATION ALGORITHM

Ant Colony Optimization (ACO) Algorithm Step-by-Step

8. Display the best solution.

Loop will Repeat until maximum number of iterations (i.e., 200) after that it will display best solution (optimal solution).

ANT COLONY OPTIMIZATION ALGORITHM



ACO Limitation:

Use more Parameters.

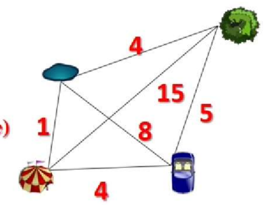
ACO Advantage:

Provide better solution with fewer iterations.

PHEROMONE MATRIX

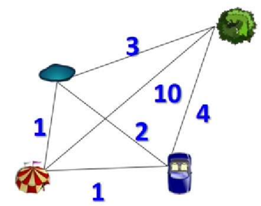
$$\begin{bmatrix} 0 & 5 & 15 & 4 \\ 5 & 0 & 4 & 8 \\ 15 & 4 & 0 & 1 \\ 4 & 8 & 1 & 0 \end{bmatrix}$$

Cost matrix (distance)



$$\begin{bmatrix} 0 & 4 & 10 & 3 \\ 4 & 0 & 1 & 2 \\ 10 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}$$

Pheromone matrix



$$\Delta\tau_{i,j}^{k} = \begin{cases} \dfrac{1}{L_k} & k^{th} \, ant \; travels \; on \; the \; edge \; i,j \\ 0 & otherwise \end{cases}$$

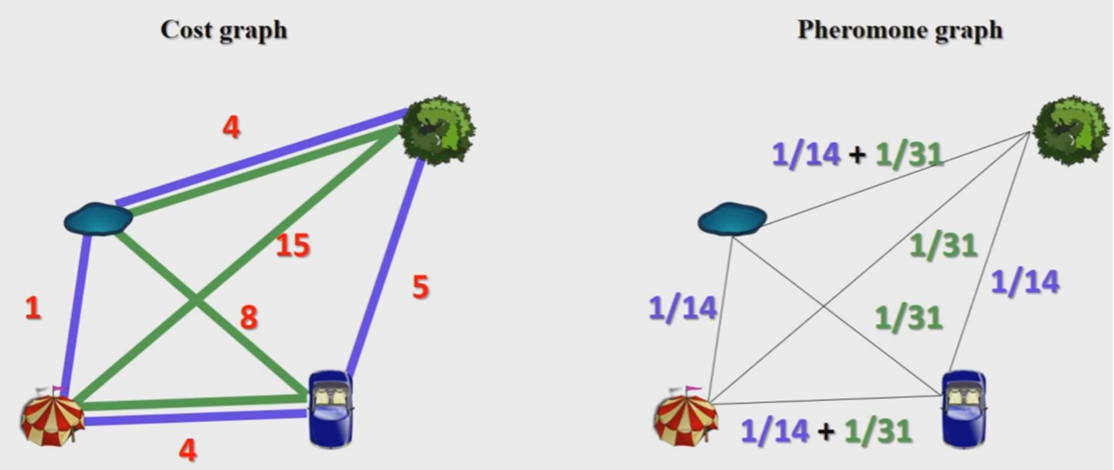
$$\tau_{i,j}^{k} = \sum_{k=1}^{m} \Delta\tau_{i,j}^{k}$$

Without vaporization

$$\tau_{i,j}^{k} = (1-\rho)\,\tau_{i,j} + \sum_{k=1}^{m} \Delta\tau_{i,j}^{k}$$

With vaporization

PHEROMONE MATRIX

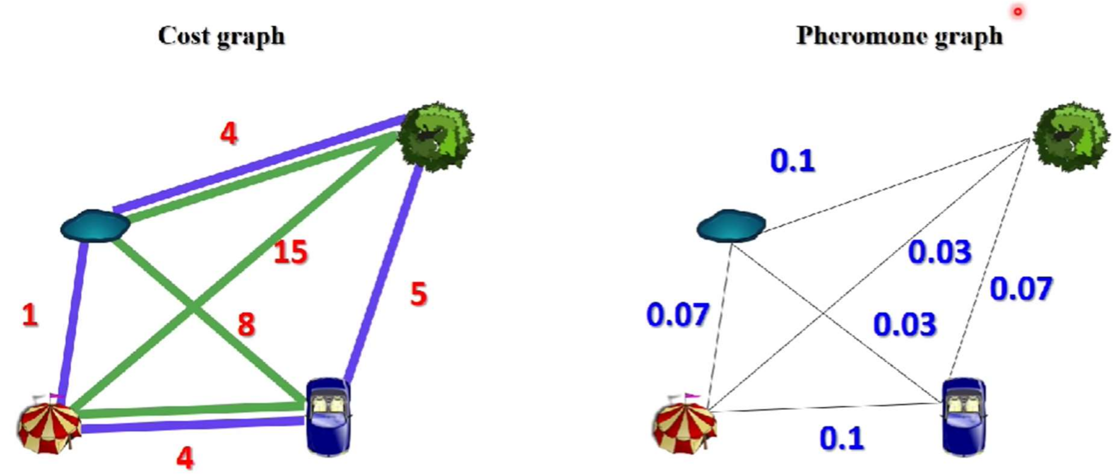


$L_1 = 14 \rightarrow \Delta\tau_{i,j}^1 = \frac{1}{14}$

$L_2 = 31 \rightarrow \Delta\tau_{i,j}^2 = \frac{1}{31}$

$$\tau_{i,j} = \sum_{k=1}^{m} \Delta\tau_{i,j}^k$$

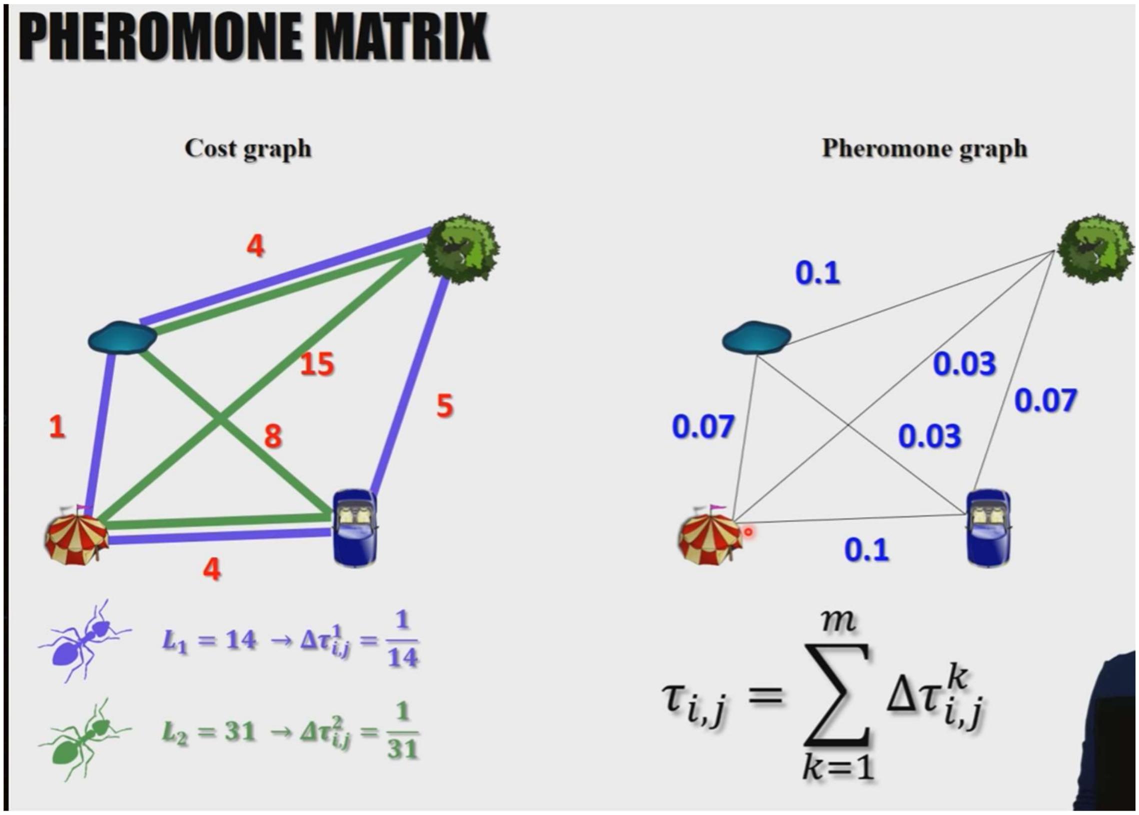
PHEROMONE MATRIX



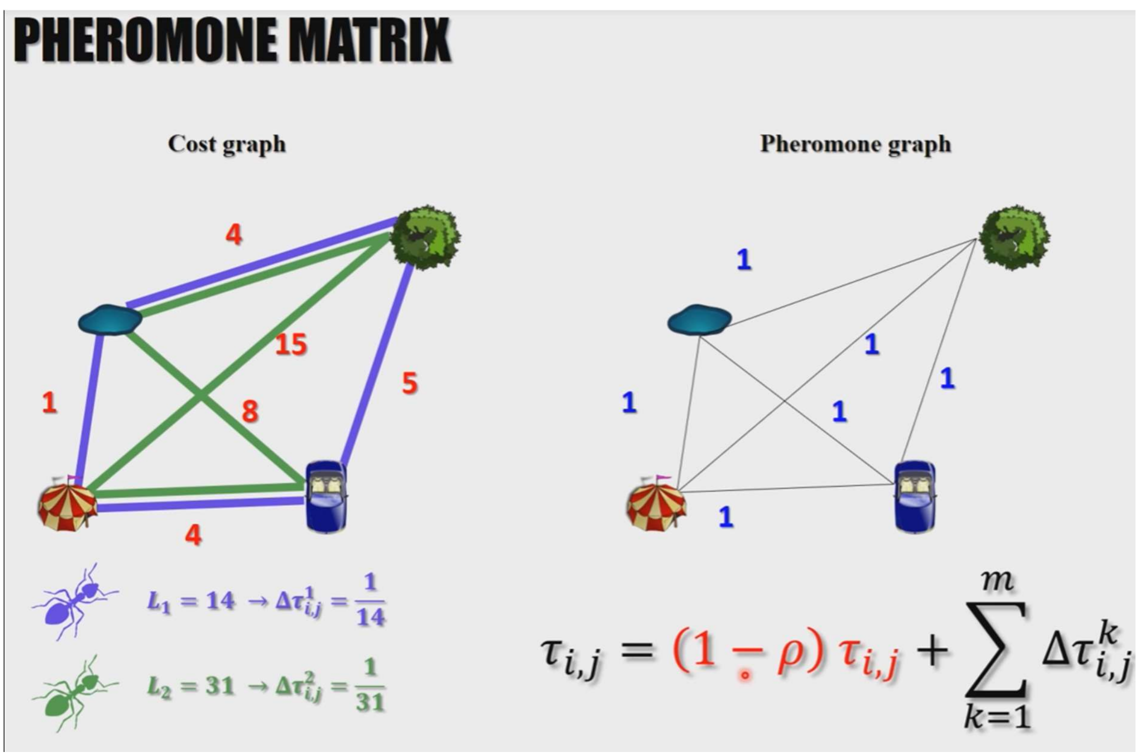
$L_1 = 14 \rightarrow \Delta\tau_{i,j}^1 = \frac{1}{14}$

$L_2 = 31 \rightarrow \Delta\tau_{i,j}^2 = \frac{1}{31}$

$$\tau_{i,j} = \sum_{k=1}^{m} \Delta\tau_{i,j}^k$$



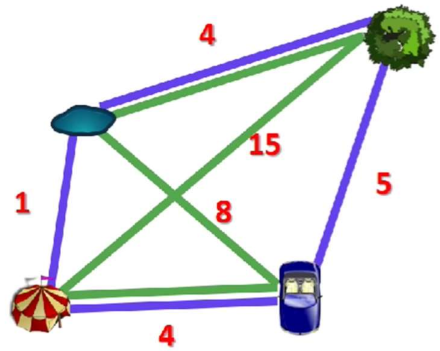
New ant selection path??



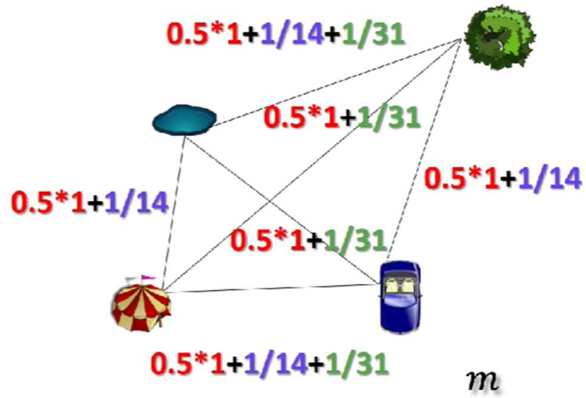
PHEROMONE MATRIX

$\rho = 0.5$

Cost graph



Pheromone graph



$L_1 = 14 \rightarrow \Delta\tau^1_{i,j} = \frac{1}{14}$

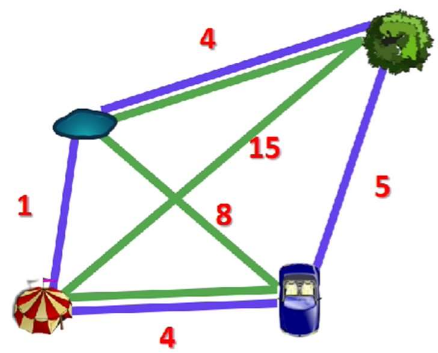
$L_2 = 31 \rightarrow \Delta\tau^2_{i,j} = \frac{1}{31}$

$$\tau_{i,j} = (1-\rho)\,\tau_{i,j} + \sum_{k=1}^{m} \Delta\tau^k_{i,j}$$

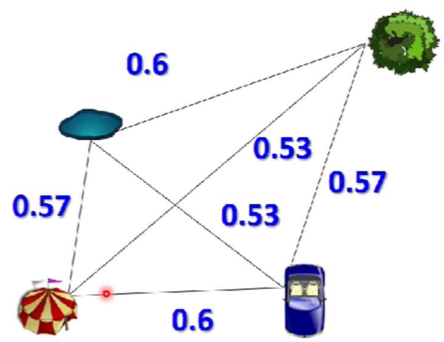
PHEROMONE MATRIX

$\rho = 0.5$

Cost graph



Pheromone graph



$L_1 = 14 \rightarrow \Delta\tau^1_{i,j} = \frac{1}{14}$

$L_2 = 31 \rightarrow \Delta\tau^2_{i,j} = \frac{1}{31}$

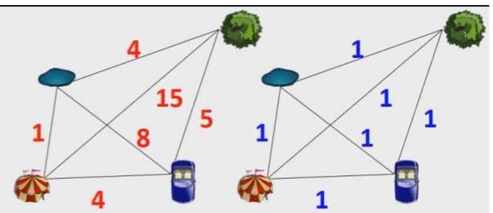
$$\tau_{i,j} = (1-\rho)\,\tau_{i,j} + \sum_{k=1}^{m} \Delta\tau^k_{i,j}$$

CALCULATING THE PROBABILITIES

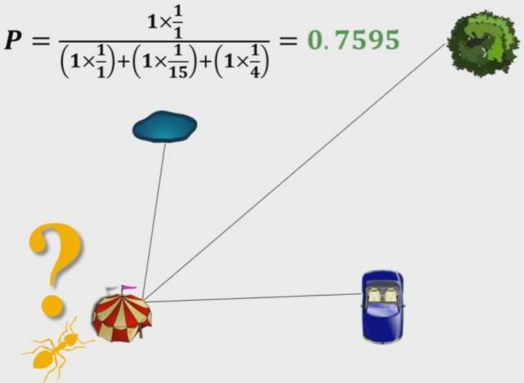
$$P_{i,j} = \frac{\left(\tau_{i,j}\right)^{\alpha}\left(\eta_{i,j}\right)^{\beta}}{\sum\left(\left(\tau_{i,j}\right)^{\alpha}\left(\eta_{i,j}\right)^{\beta}\right)}$$

$$where:\ \eta_{i,j} = \frac{1}{\mathrm{L}_{i,j}}$$

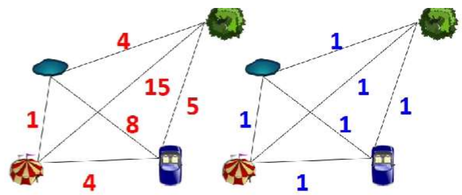
NUMERICAL EXAMPLE



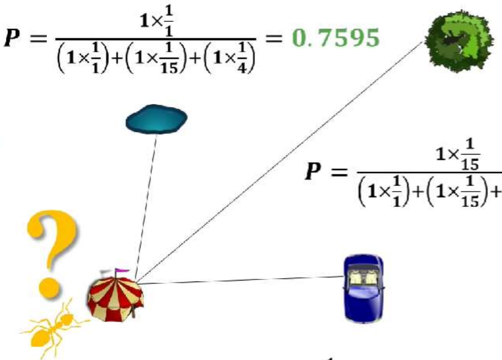
$$P = \frac{1\times\frac{1}{1}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}\right)+\left(1\times\frac{1}{4}\right)} = 0.7595$$



NUMERICAL EXAMPLE



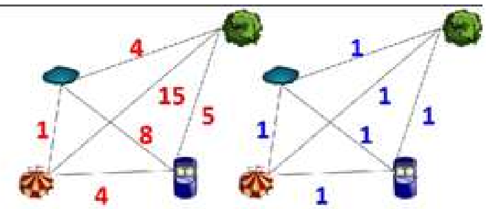
$$P = \frac{1\times\frac{1}{1}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}\right)+\left(1\times\frac{1}{4}\right)} = 0.7595$$

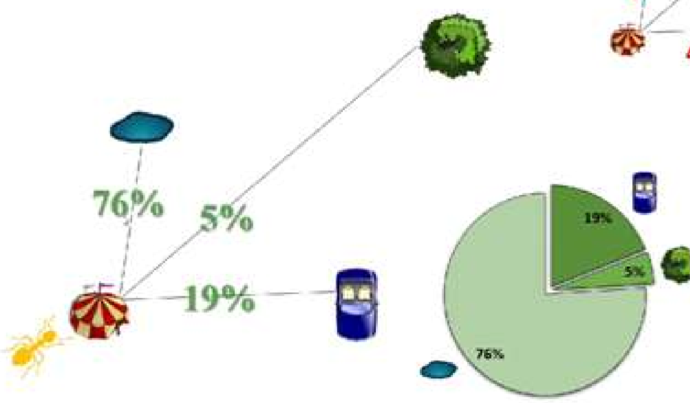


$$P = \frac{1\times\frac{1}{15}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}\right)+\left(1\times\frac{1}{4}\right)} = 0.0506$$

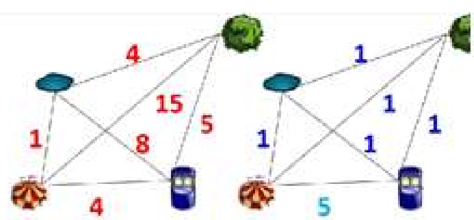
$$P = \frac{1\times\frac{1}{4}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}\right)+\left(1\times\frac{1}{4}\right)} = 0.1899$$

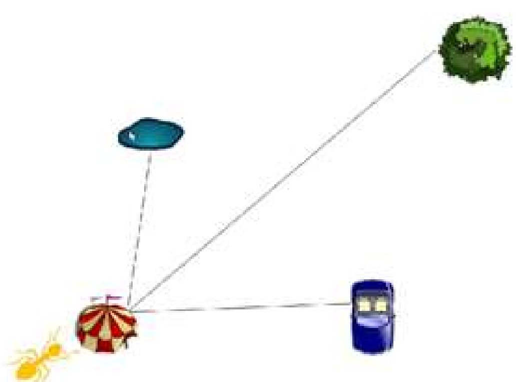
NUMERICAL EXAMPLE

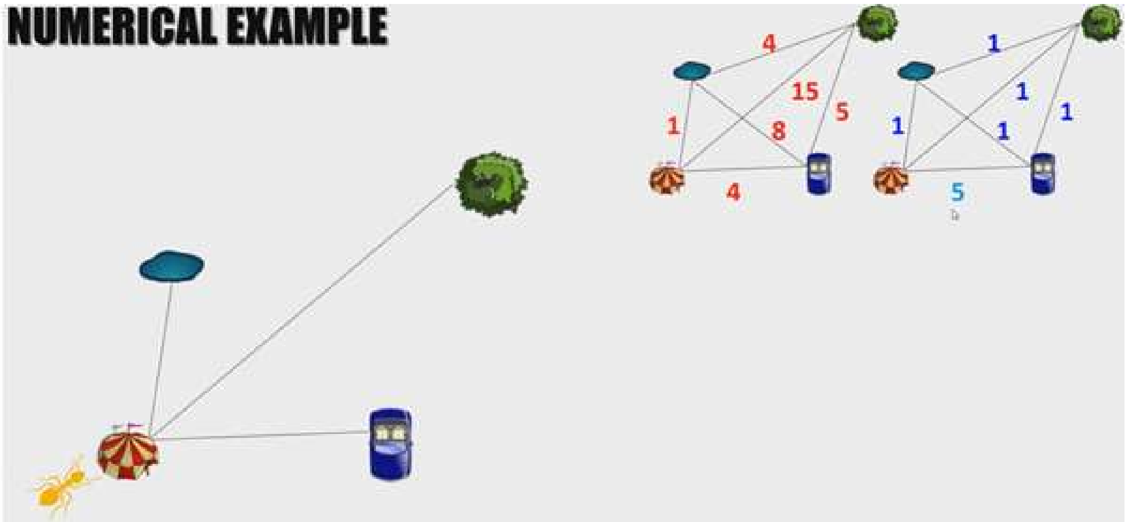


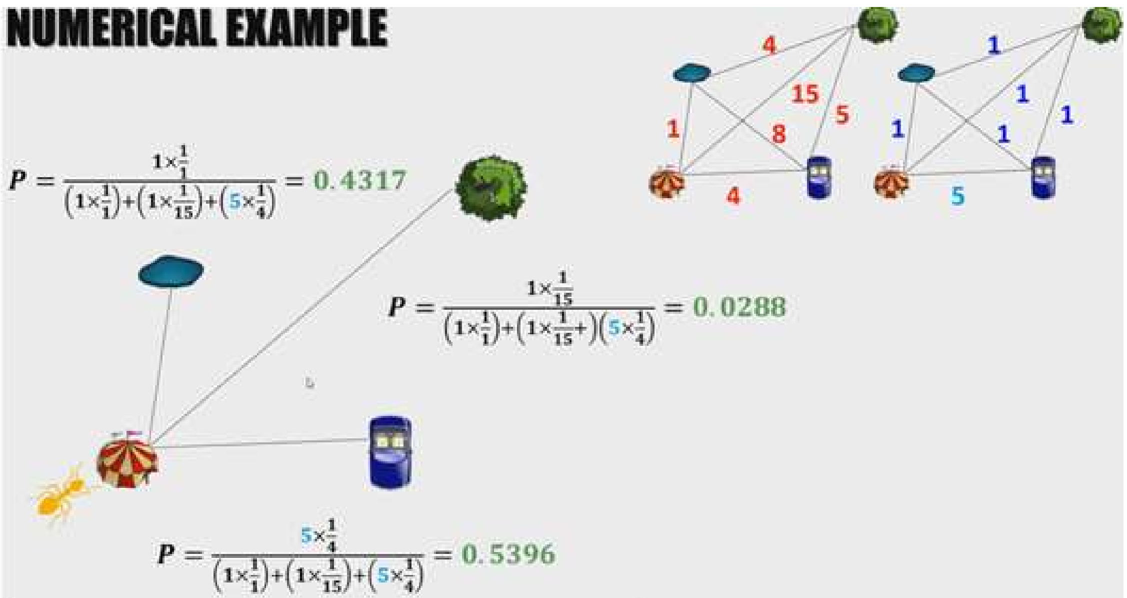


NUMERICAL EXAMPLE

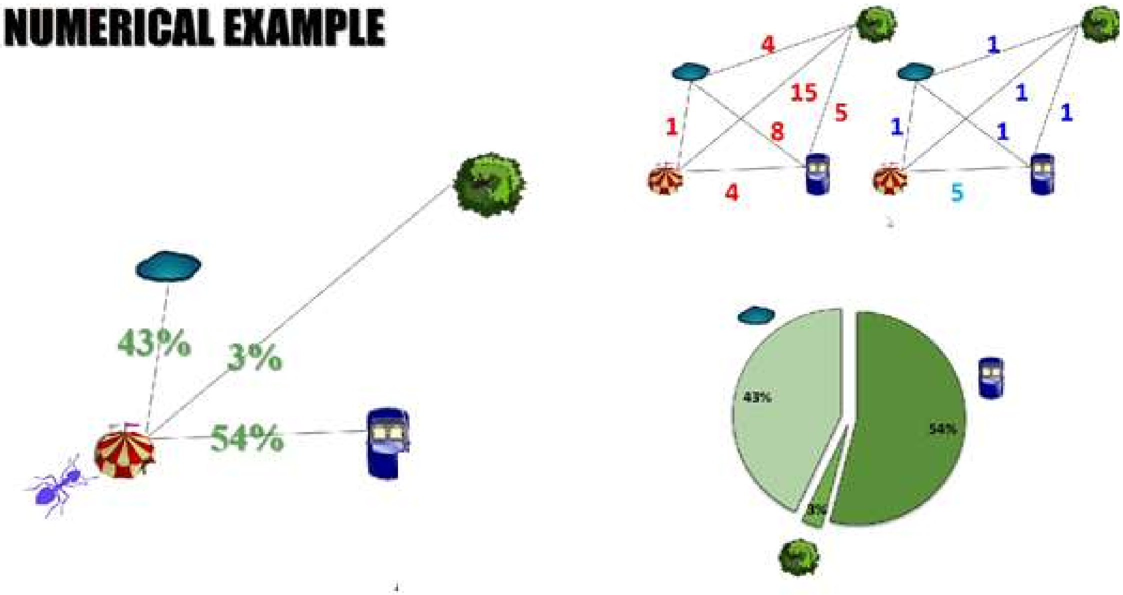




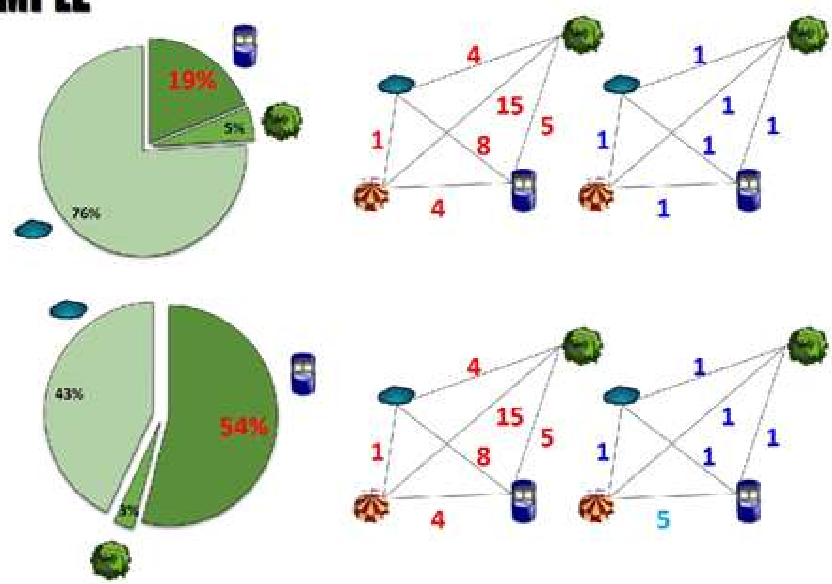

NUMERICAL EXAMPLE
4
15
1
8
5
4
1
1
1
1
1
5

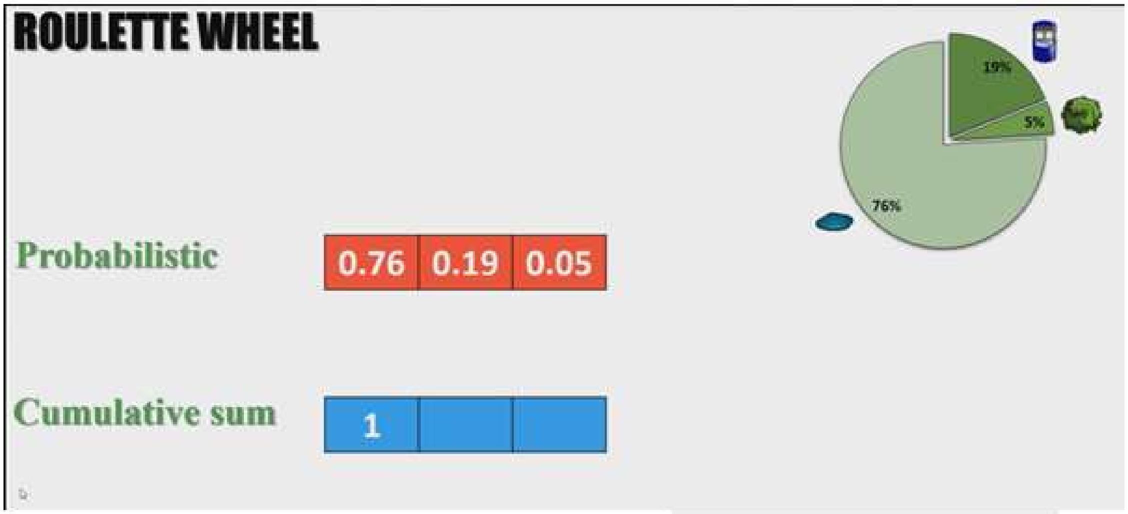

NUMERICAL EXAMPLE
4
15
1
8
5
4
1
1
1
1
1
5
$P = \frac{1\times\frac{1}{1}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}\right)+\left(5\times\frac{1}{4}\right)} = 0.4317$
$P = \frac{1\times\frac{1}{15}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}+\right)\left(5\times\frac{1}{4}\right)} = 0.0288$
$P = \frac{5\times\frac{1}{4}}{\left(1\times\frac{1}{1}\right)+\left(1\times\frac{1}{15}\right)+\left(5\times\frac{1}{4}\right)} = 0.5396$

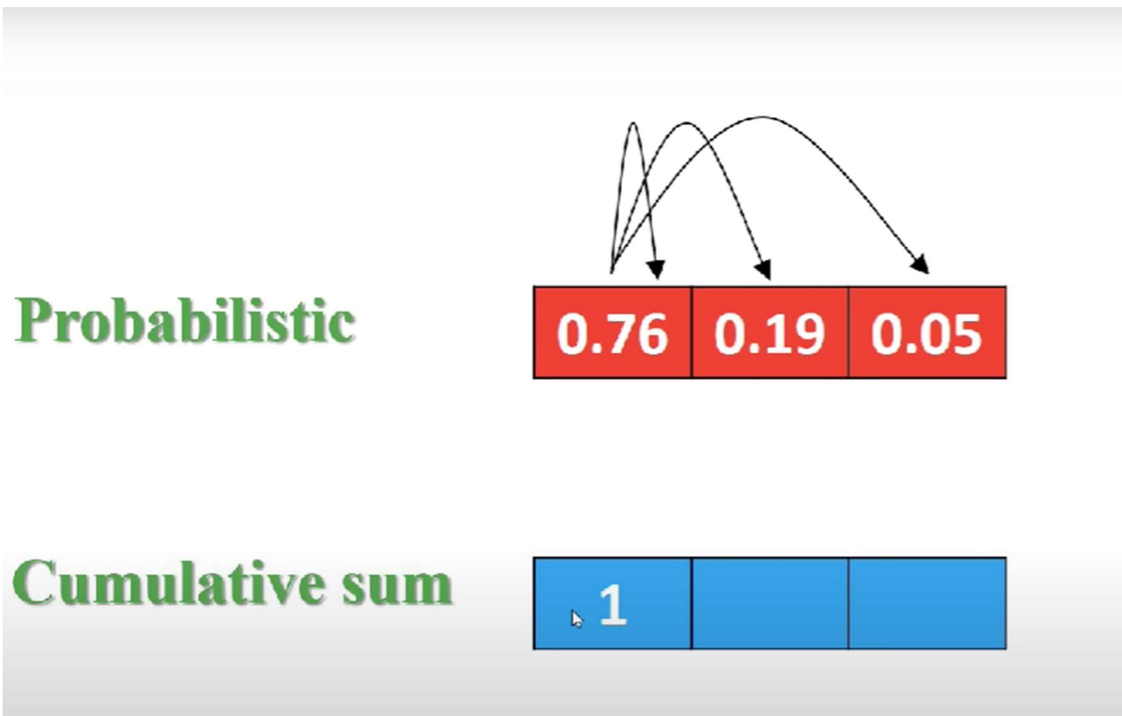
NUMERICAL EXAMPLE

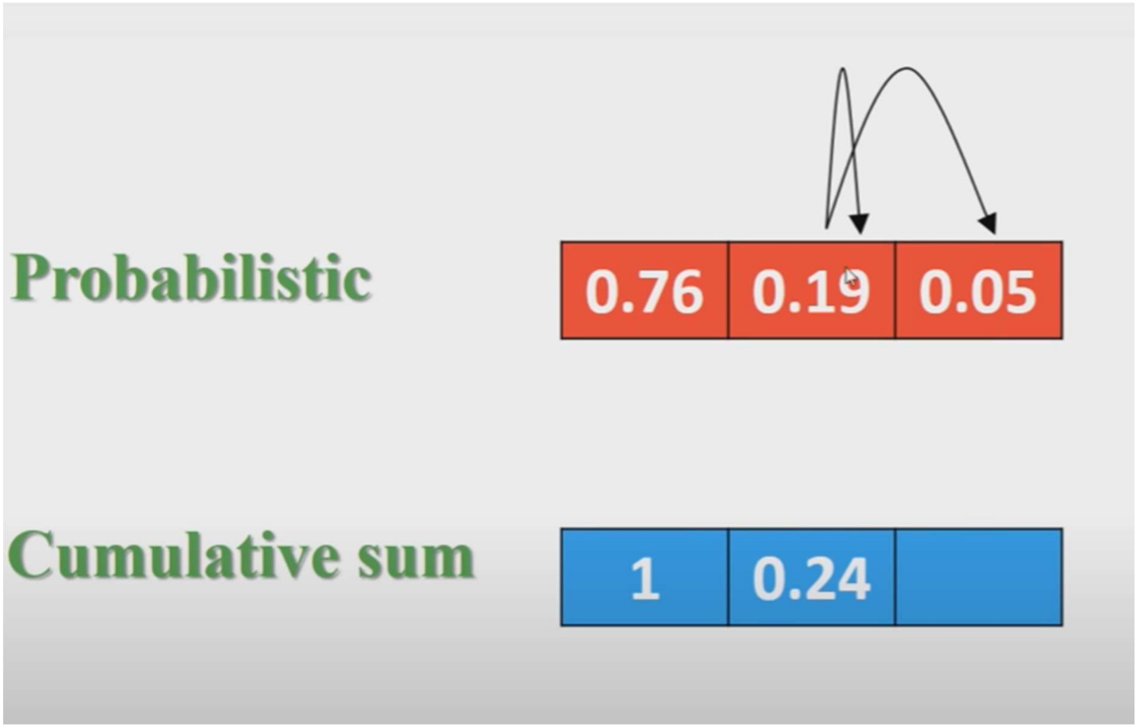


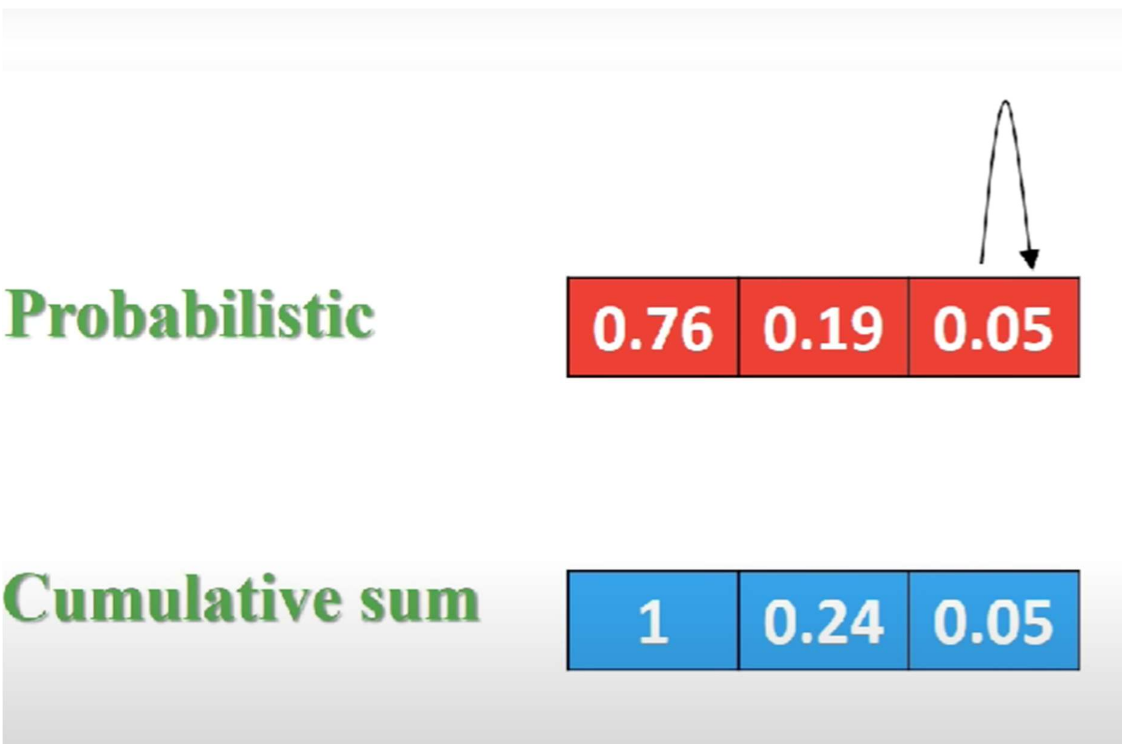
NUMERICAL EXAMPLE

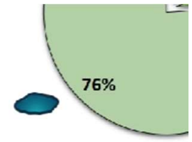



ROULETTE WHEEL
19%
5%
76%
Probabilistic
0.76
0.19
0.05
Cumulative sum
1


Probabilistic
0.76
0.19
0.05
Cumulative sum
1


Probabilistic
0.76
0.19
0.05
Cumulative sum
1
0.24


Probabilistic
0.76
0.19
0.05
Cumulative sum
1
0.24
0.05



Probabilistic

0.76	0.19	0.05

Cumulative sum

1	0.24	0.05

A random number (r) in [0,1]

$$\begin{cases} 0.24 < r \leq 1.00 \\ 0.05 < r \leq 0.24 \\ 0.00 \leq r \leq 0.05 \end{cases}$$