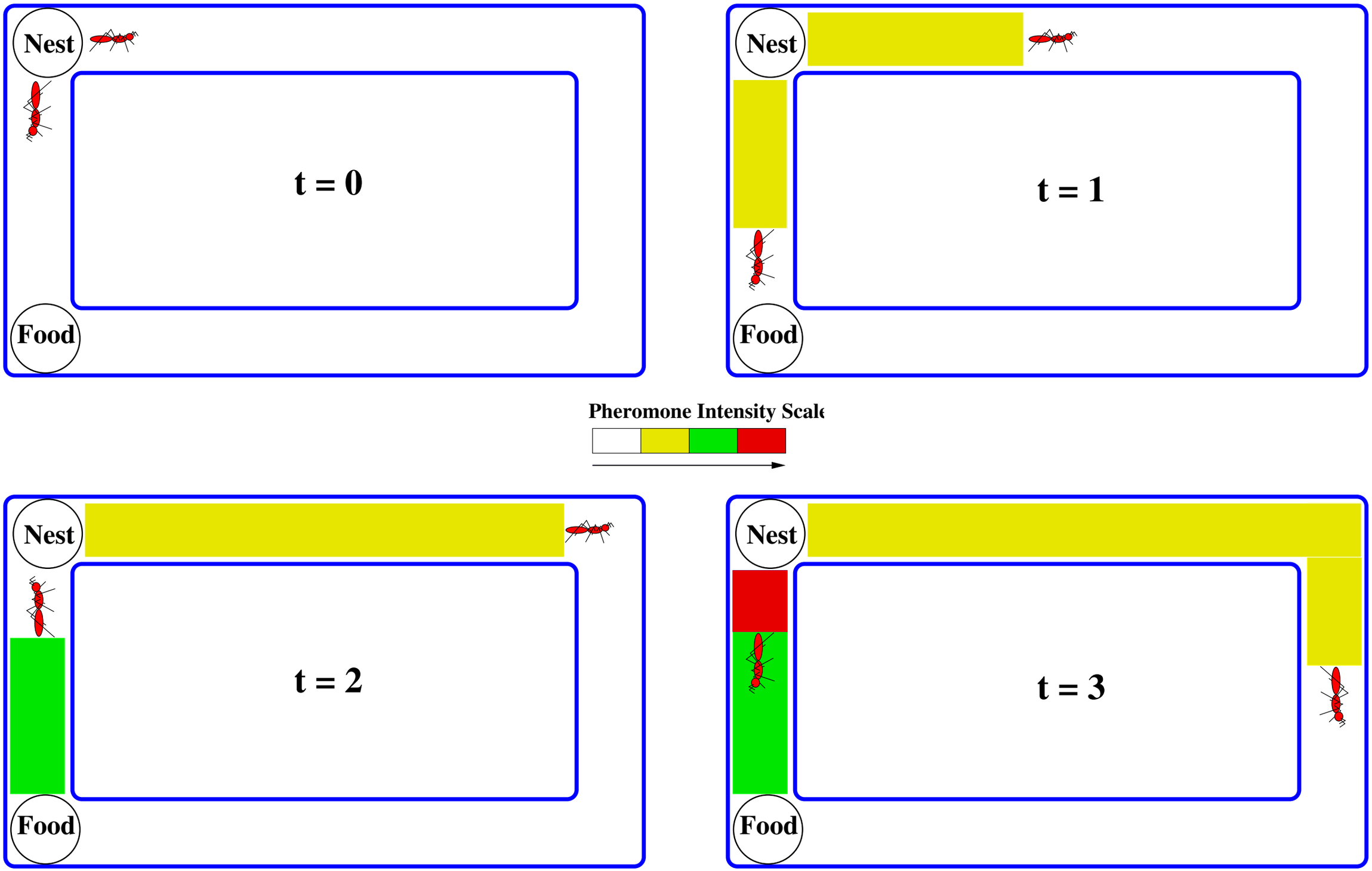
15-382 Collective Intelligence - S19

Lecture 25: Swarm Intelligence 6 / Ant Colony Optimization 2

Teacher:
Gianni A. Di Caro

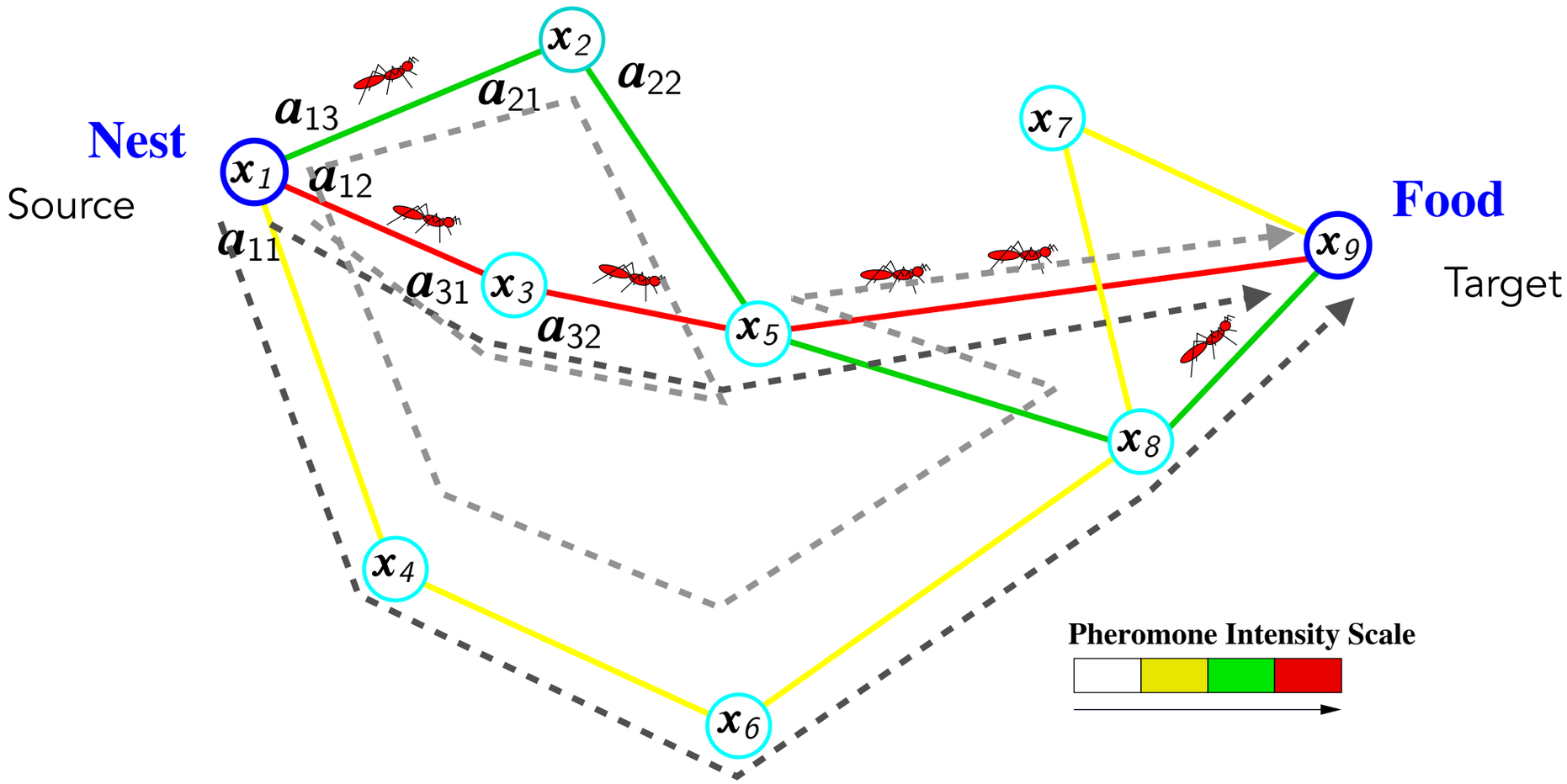
جامعة كارنيجي ميلون في قطر
Carnegie Mellon University Qatar

SHORTEST PATHS WITH PHEROMONE LAYING-FOLLOWING



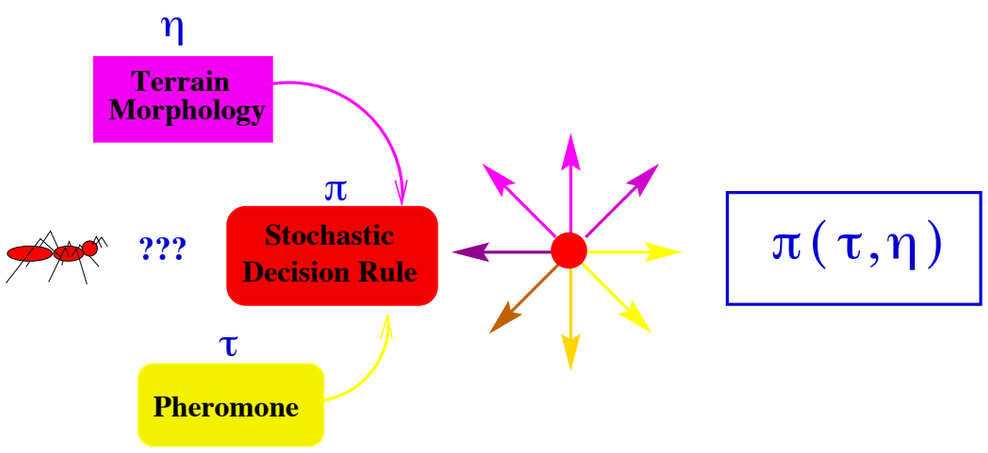
#Pheromone on a branch ∝ Frequency of fw/bw crossing ∝ Length (quality) of paths

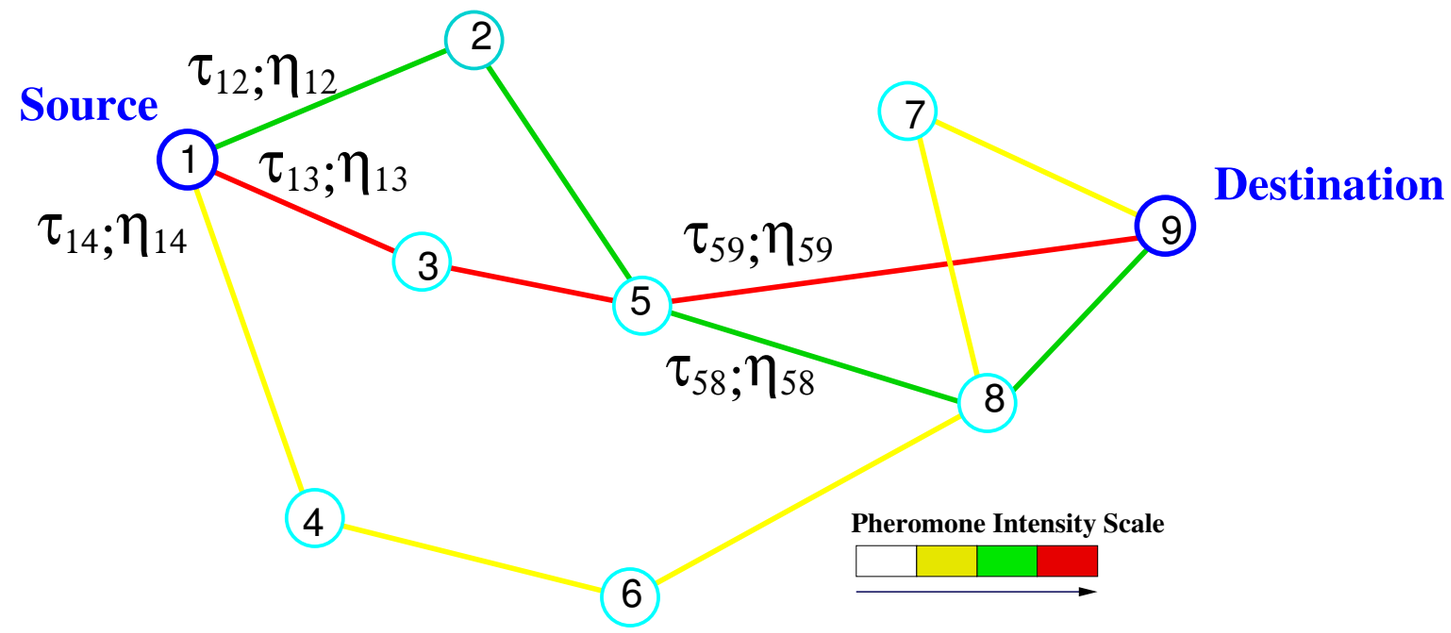
LET'S ABSTRACT A MORE COMPLEX SCENARIO



- Multiple decision nodes: n decision states/nodes, $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \boldsymbol{X}$
- Set $\boldsymbol{A}$ of decisions / actions, $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots \boldsymbol{a}_m$, such that at each state $\boldsymbol{x}$ a subset $\mathscr{A}(\boldsymbol{x})$ of actions is available or feasible
- A **path (ant *solution*)** is *constructed* through a **sequence decisions, for each visited state**
- **Multiple ants iterating path construction (i.e., *foraging*) in parallel**
- A traveling cost is associated to each state transition: colony's goal is to let the ants moving over the ***minimum-cost path between nest and food***

LET'S ABSTRACT A MORE COMPLEX SCENARIO



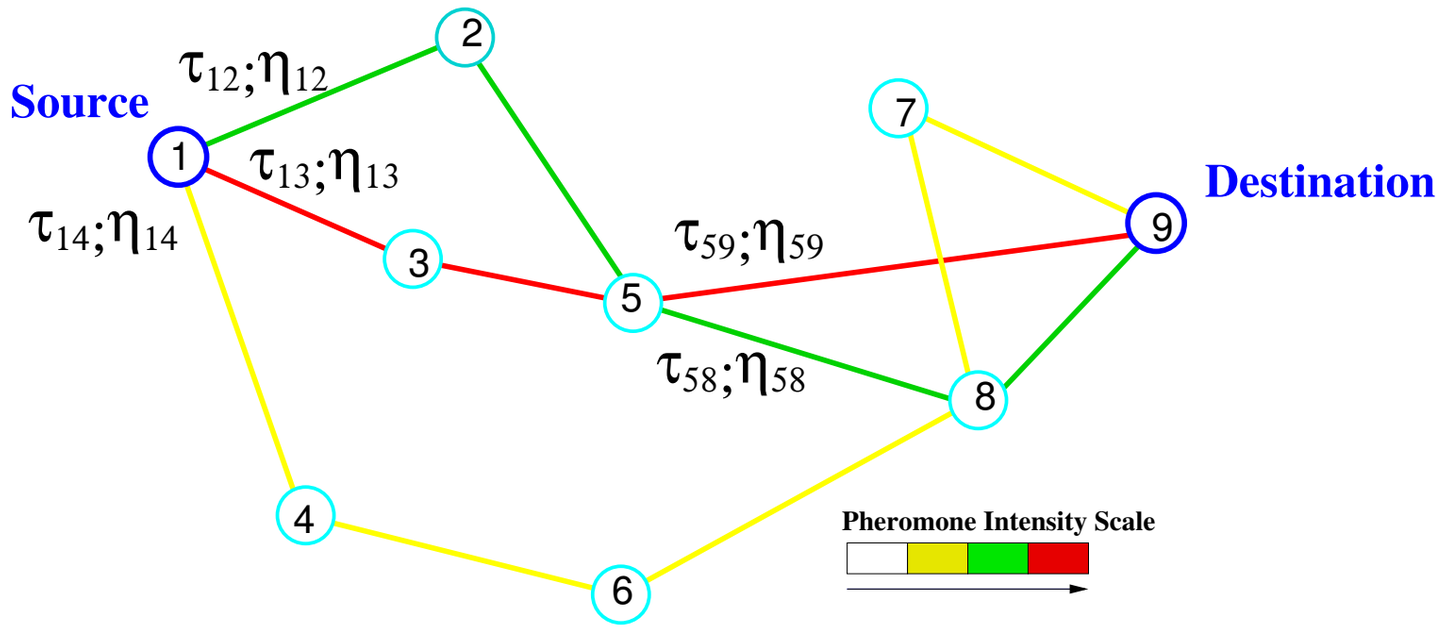


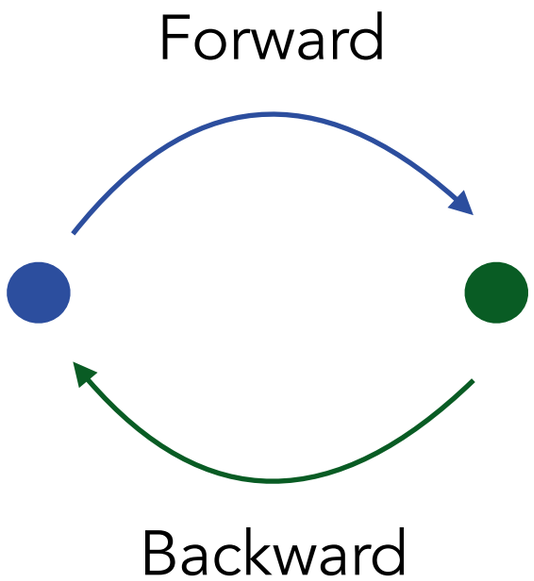
- <u>Distributed Optimization Problem</u>
- **At each state x_k only <u>local information / constraints</u> (+ some ant <u>memory)</u> is available for taking (a possibly optimized) *decision* $a \in \mathscr{A}(x_k)$**
  - **Pheromone information (dynamic), parametrized as a vector** τ^k *(stigmergic variables)*
  - **Heuristic information (static, scenario-related)** parametrized as a vector η^k
- <u>Ant behavior:</u> **Stochastic decision policy** $\pi_\varepsilon(x_k; \tau^k, \eta^k)$, $\pi_\varepsilon : X \longmapsto A$

<u>How ant colonies solve the Distributed MCP problem?</u>

Exploiting pheromone for learning the best (parameters) of the decision policy

ANT COLONIES: INGREDIENTS FOR SHORTEST PATHS

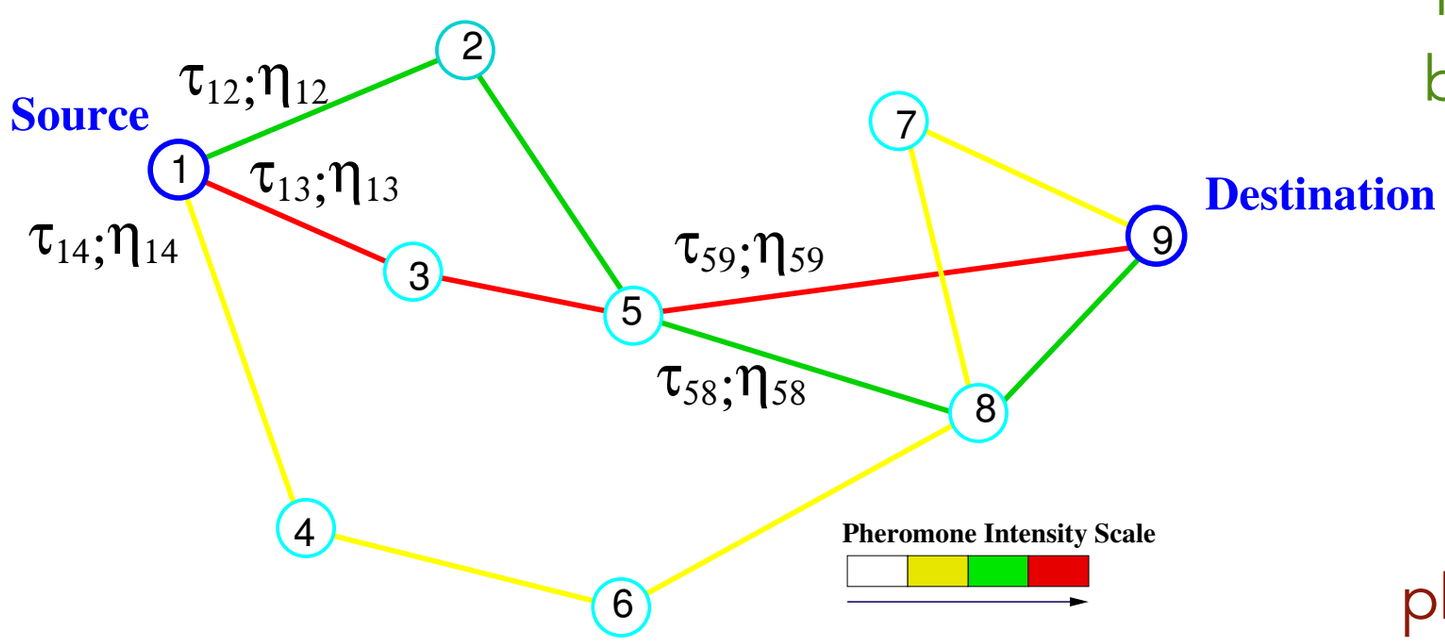




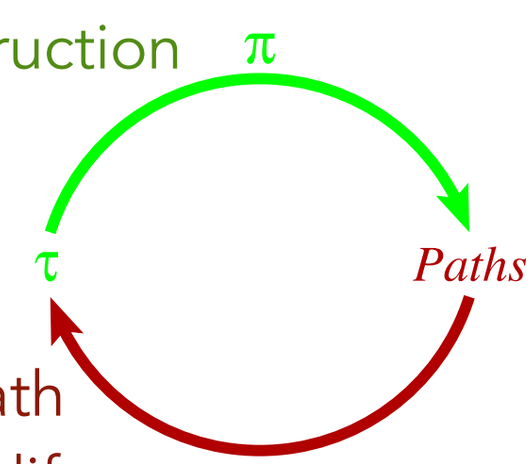
- A number of concurrent autonomous (simple?) agents (ants)
- Forward-backward constructive **path sampling** based on the stochastic policy $\pi_{\boldsymbol{\varepsilon}}$
- Local laying and sensing of pheromone → ***Pheromone is dynamically updated***
- Step-by-step stochastic decisions biased by local pheromone intensity and by other local heuristic aspects (e.g., terrain)
- Multiple paths are concurrently tried out and implicitly evaluated
- **Positive feedback effect** (local reinforcement of good decisions)
- Iteration over time of the path sampling actions
- Persistence (**exploitation**) and evaporation (**exploration**) of pheromone

FROM ANTS TO ACO

- Let's mimic ant colonies, with some pragmatic modifications ....
- <u>Once completed a solution / path:</u>
  - The sampled solution is **<u>evaluated</u>** (e.g., sum of the individual costs)
  - "**Credit**" is assigned to each individual decision belonging to the solution
  - **Pheromone updating:** the value of the pheromone variables τ^k associated to each decision in the solution are **modified** according to the "credit"
- Pheromone values can also <u>decade/change</u> for other reasons (e.g., ***evaporation***)
- **Pheromone values locally encode how good is to take decision i vs. j as collectively estimated/learned by the agent population through repeated solution sampling**



Pheromone distribution biases path construction



Outcomes of path construction modify pheromone distribution

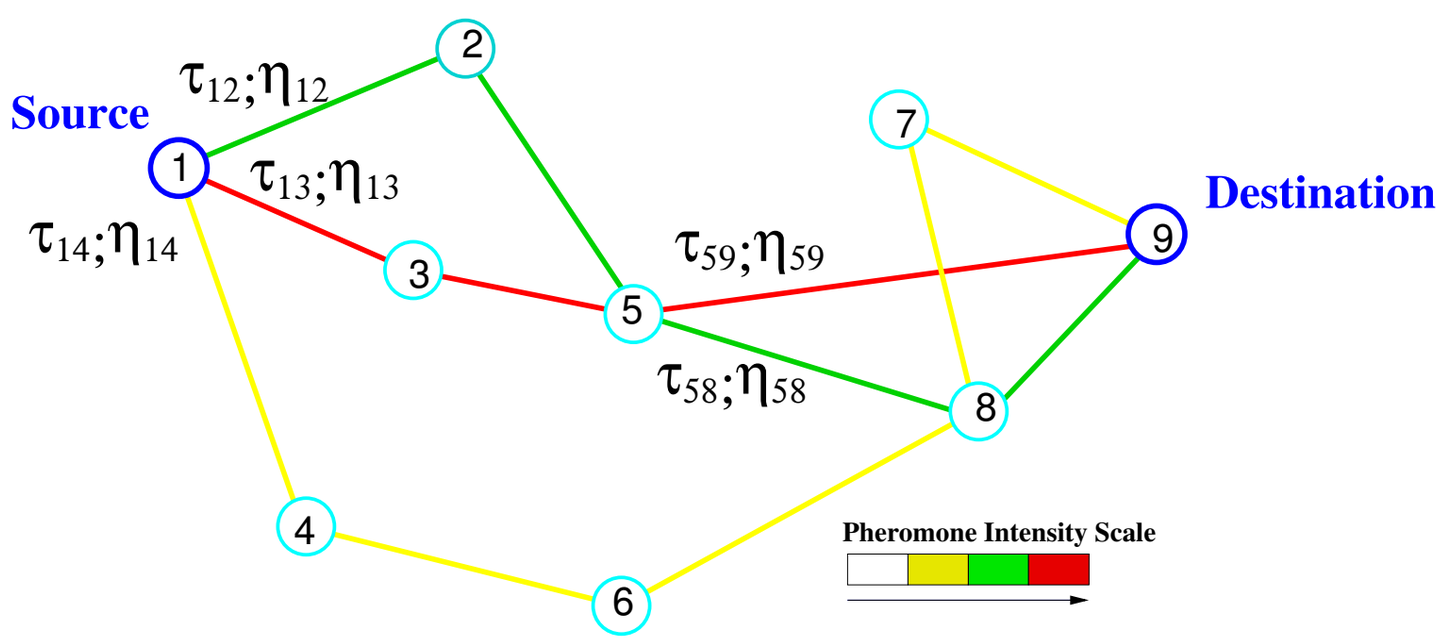
ANT COLONY OPTIMIZATION METAHEURISTIC: (VERY) GENERAL ARCHITECTURE

```
procedure ACO_metaheuristic()
  while (¬ stopping_criterion)
    schedule_activities
      ant_agents_construct_solutions_using_pheromone();
      pheromone_updating();
      daemon_actions(); /* optional */
    end schedule_activities
  end while
  return best_solution_generated;
```

- Solution construction
- Monte Carlo path sampling by N (# states) joint probability distributions parametrized by τ and η variable arrays
- Sequential learning by Generalized Policy Iteration (GPI)

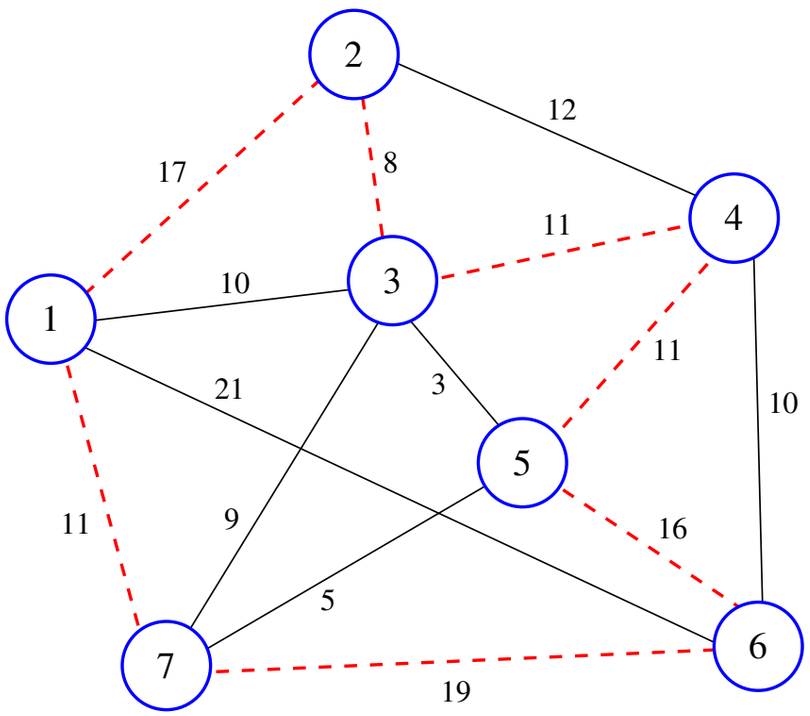
PHEROMONE AND HEURISTIC ARRAYS

- At each decision node i an array of pheromone variables: $\vec{\tau}_i = [\tau_{ij}] \in \mathbb{R},\ \forall j \in \mathcal{N}(i)$ is available to the ant agent to issue the routing decision
- $\tau_{ij} = q(j|i)$: estimate of the quality of moving to next node j conditionally to the fact of being in i. Pheromone values are collectively learned by the ants through path sampling
- At each decision node i an array of heuristics variables $\vec{\eta}_i = [\eta_{ij}] \in \mathbb{R},\ \forall j \in \mathcal{N}(i)$ is also available to the ant agent to take the routing decision
- η_{ij} is also an estimate of $q(j|i)$ but it is derived from a process or a priori knowledge not related to the ant actions (e.g., node-to-node distance)



ACO FOR THE TRAVELING SALESMAN PROBLEM (TSP)

Given G(V, E) find the Hamiltionian tour of minimal cost : NP-Hard



Every *cyclic permutation* of *n* integers is a feasible solution

$$\pi_1 = (1, 3, 4, 2, 6, 5, 7, 1), \;\; \pi_2 = (2, 3, 4, 5, 6, 7, 1, 2)$$
$$c(\pi_2) = d_{23} + d_{34} + d_{45} + d_{56} + d_{67} + d_{71} + d_{12} = 93$$

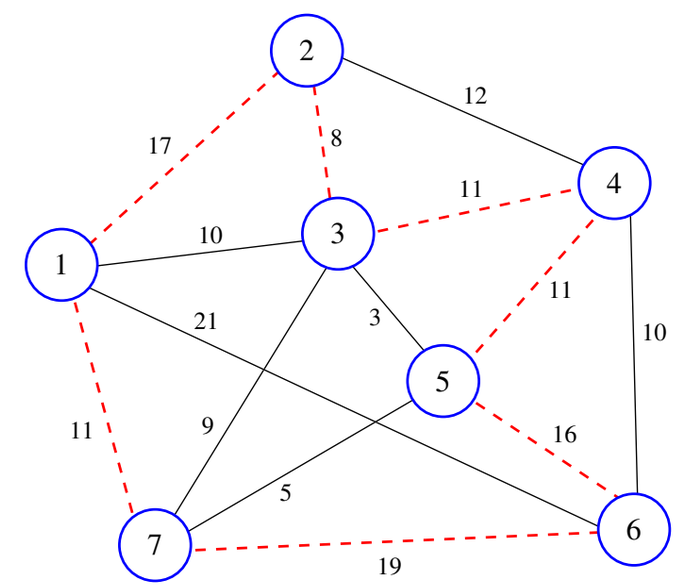
Read also as set of edges:
{(2,3), (3,4), (4,5), (6,7), (7,1), (1,2)}

It's easier to consider fully connected graphs, |E| = |V| |V-1|:
If two nodes are not connect, *d* is infinite

"Related" <u>combinatorial optimization</u> problems : VRPs, SOP, TO, QAP, …

ACO FOR THE TRAVELING SALESMAN PROBLEM (TSP)

- **Pheromone variables:** $\tau_{ij} \in \mathbb{R}^+$ expresses how beneficial is (*estimated*, up to now) to have edge (i,j) in the solution to optimize final tour length → |E| variables
- **Heuristic values** $\eta_{ij} \in \mathbb{R}^+$: problem costs $c_{ij} \in \mathbb{R}^+$ for traveling from i to j → |E| variables



Solution construction strategies (no repair, no look-ahed)

- ***Extension:*** when ant k is in city i, how good is expected to include (feasible) city j (next in the solution sequence $x^k(t)$? → $f(\tau_{ij}, \eta_{ij})$
- ***Insertion:*** how good is expected to insert (feasible) edge (m,p) in the partial solution $x^k(t)$? → $f(\tau_{mp}\ \eta_{mp})$

(META-)ACO FOR CO PROBLEMS (CENTRALIZED SCHEDULE)

```
Initialize τ_ij(0) to small random values and let t = 0;
repeat
    Place n_k ants on randomly chosen origin nodes;
    foreach ant k = 1, ..., n_k do
        Construct a path x^k(t) [Update pheromone step-by-step];
        Evaluate path x^k(t);
    end
    foreach [selected] edge (i, j) of the graph do
        Pheromone evaporation;
    end
    foreach [selected] ant k = 1, ..., n_k do
        foreach [selected] edge (i, j) of x^k(t) do
            Update τ_ij using path evaluation results;
        end
    end
Daemon actions [Local search];
t = t + 1;
until stopping condition is true;
return best solution generated;
```