Lecture #4

Swarm Intelligence

Dr. Charnon Pattiyanon
Assistant Director of IT, Instructor
Department of Artificial Intelligence and Computer Engineering
CMKL University

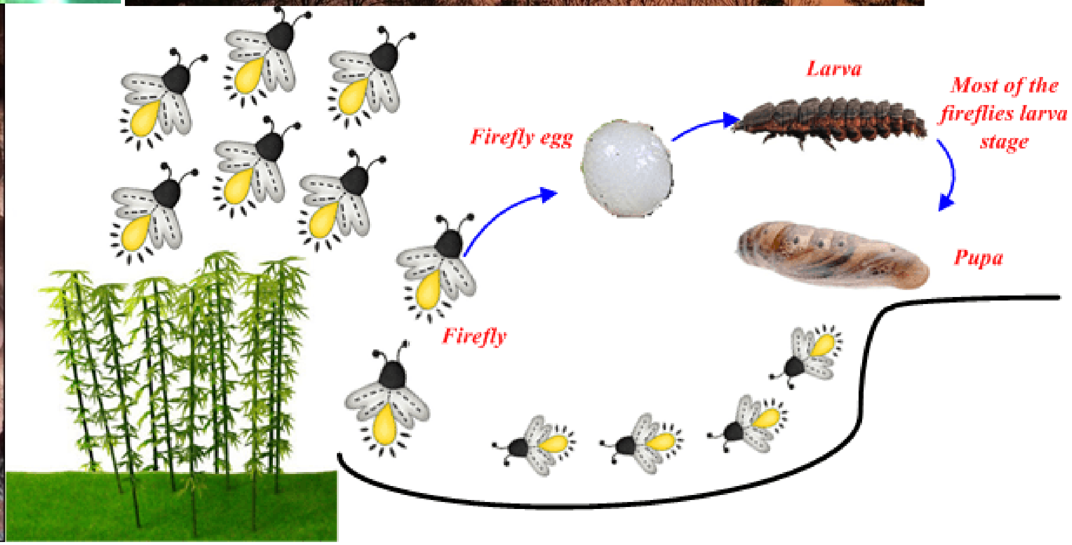
Modified from Companion Slides for "D. Floreano and C. Mattiussi, Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies, 2008"

An Introduction to Swarm Intelligence

- **Swarm Intelligence** is inspired by collaborative behaviors in social animals.

These social animals require **no leader**.

Their collaborative behaviors emerge from interactions among individuals.



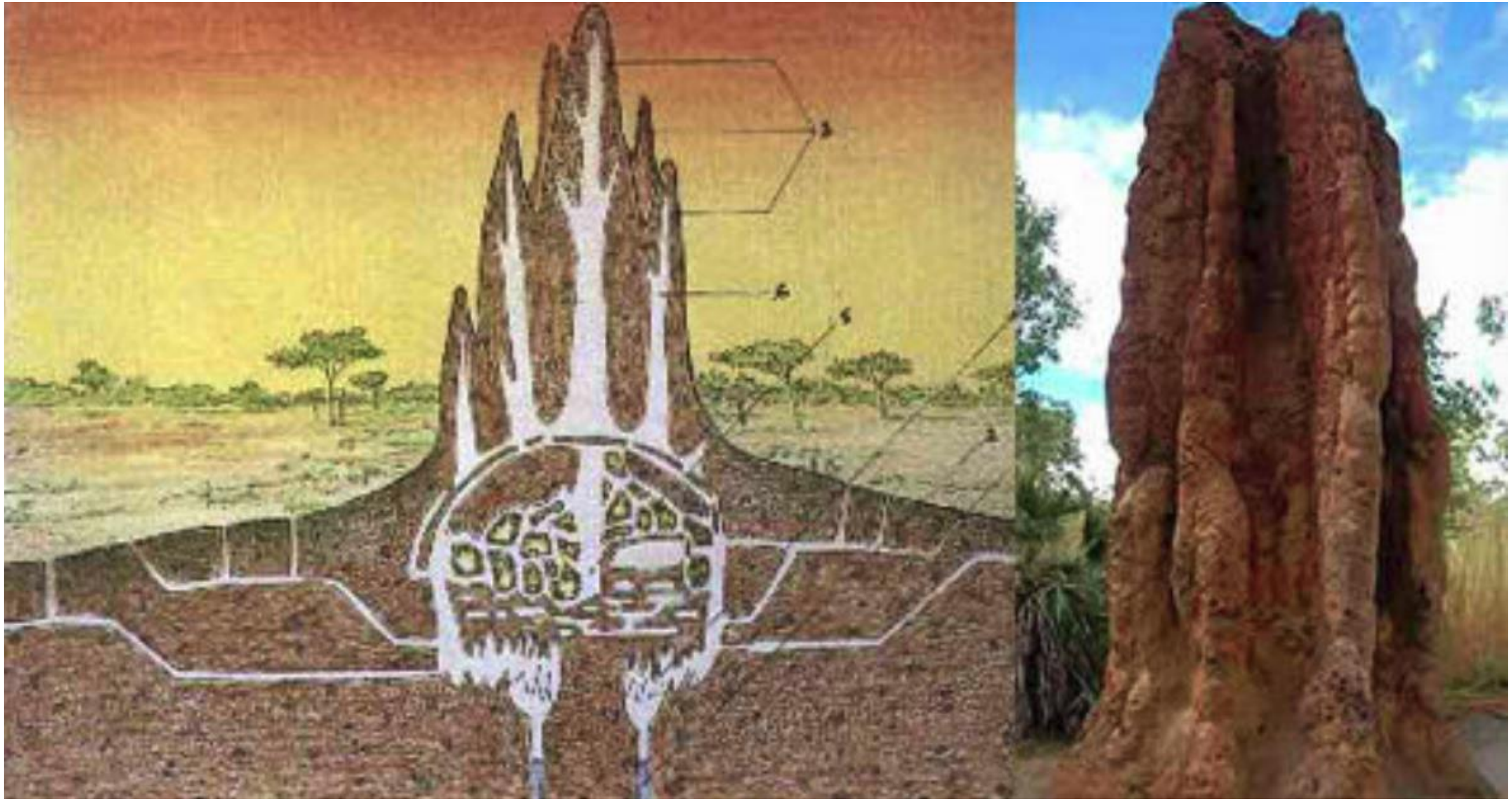
Ant

- Oecophylla longinoda or Weaver ant
- Building nests from leaves bound together with silk secreted by larvae.

Ant Foraging Behavior

Source: https://www.youtube.com/watch?v=8ZmO948g4Q4

Termite



Termite Mound Construction

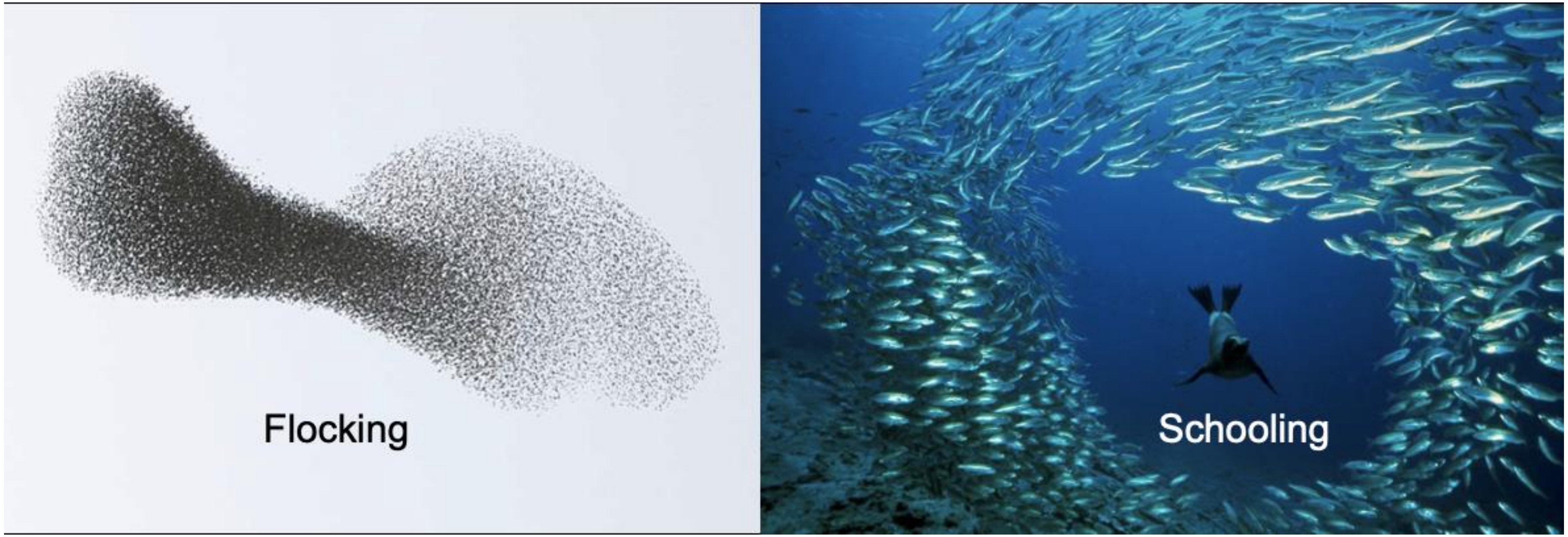
Source: https://www.youtube.com/watch?v=0m7odGafpQU

Bee

Bee Foraging Behavior

Source: https://www.youtube.com/watch?v=M0Gy99-pTuE

Emergent Collective Behavior



- There is **no** predefined group leader.
- Each individual uses **only local information** about the presence of other individuals and of the environment.

Emergent Collective Behavior

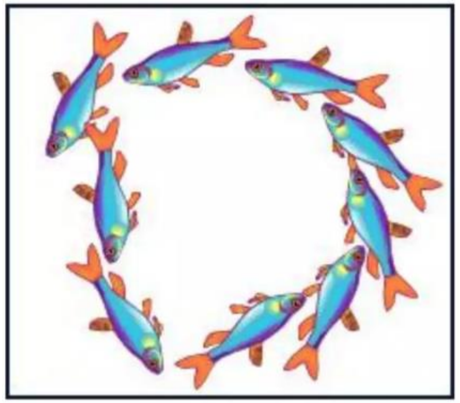
- There is **no** predefined group leader.
- Each individual uses **only local information** about the presence of other individuals and of the environment.
- In some cases, there is a leader and more restrictive rules on relative motion, **but individuals still use local information to decide how to move**.

Main Principles of Collective Behavior

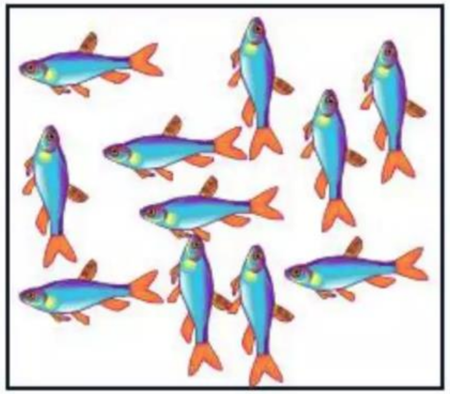
- The main principles of the collective behavior are:
  - **Homogeneity:** every bird in flock has the same behavioral model. The flock moves without a leader, even though temporary leaders seem to appear.
  - **Locality:** the motion of each bird is only influenced by its nearest flock mates. Vision is considered to be the most important senses for flock organization.
  - **Collision Avoidance:** avoid collision with nearby flock mates.
  - **Velocity Matching:** attempt to match velocity with nearby flock mates.
  - **Flock Centering:** attempt to stay close to nearby flock mates.

Collective Dynamical Behaviors

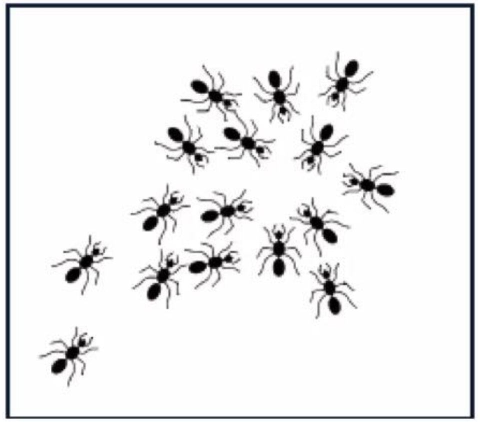
- **Torus:** Individuals continuously rotate around an empty core (milling), with the direction of rotation chosen randomly.



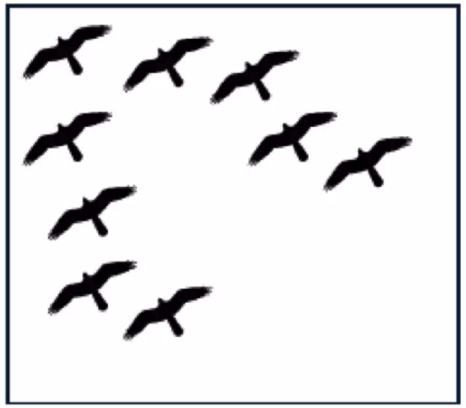
- **Dynamic Parallel Group:** Individuals are polarized and move as a coherent group. However, they can shift positions within the group, leading to fluctuations in density and overall group formation.



Collective Dynamical Behaviors



- **Swarm:** An aggregate characterized by cohesion but with a low degree of polarization (parallel alignment) among its members.



- **Highly Parallel Group:** Much more static in terms of exchange of spatial positions within the group than the dynamic parallel group and the variation in density and form is minimal.

What is Swarm Intelligence?

- The property of a system whereby the collective behaviors of unsophisticated agent **interacting locally** with their environment cause coherent functional global patterns to emerge.

- Swarm intelligence has also been referred to as **collective intelligence**.

Objective of Swarm Intelligence Techniques

To model the simple behaviors of individuals, and the local interactions with the environment and neighboring individuals, in order to obtain more complex behaviors that can be used to solve complex problems, mostly optimization problems.

Main Principles of Swarm Intelligence

- The swarm is composed of several individuals, some of which may be lost or make mistakes, but its overall performance is not affected.
- Individuals in a swarm rely on local sensory information, perform simple actions, and have little or no memory. These simple agents do not know the global state of the swarm or its goals.
- The swarm can solve complex problems that a single individual with simple abilities (computational or physical) could not solve.

The Most Popular Algorithms in the SI Domain

- **Ant Colony Optimization (ACO)**

  The ant colony optimization (ACO) algorithm is a probabilistic technique used to solve computational problems that can be reduced to finding optimal paths through **graphs**.

- **Particle Swarm Optimization (PSO)**

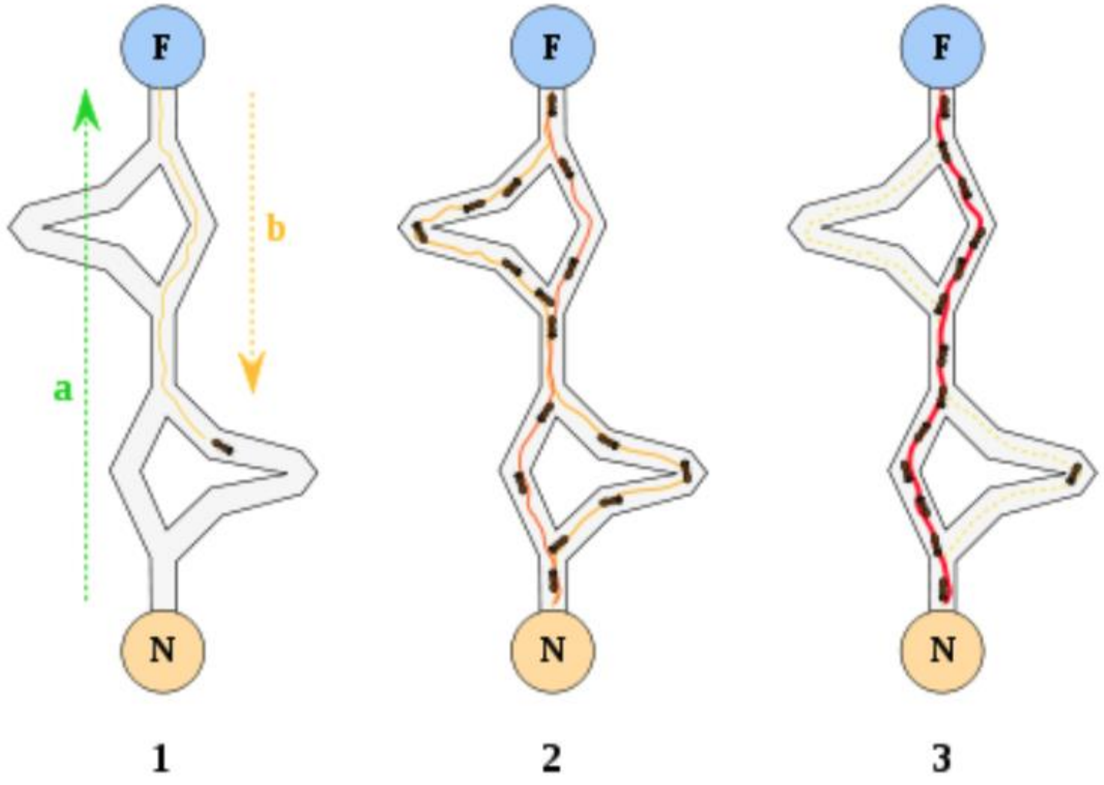
  PSO is **a population-based stochastic optimization** technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA).

Ant Colony Optimization (ACO)

Variations of Swarm Intelligence

Ant Colony Optimization (ACO)

- ACO is inspired by foraging behaviors in ants.
- There are three simple rules:
  1. If it comes across something smell like food, it will pick up.
  2. If it has something picked up, it will release pheromone.
  3. If it senses the pheromone, it will decide if or not it will follow the trail.



The ants choose the shortest path. **Why?**

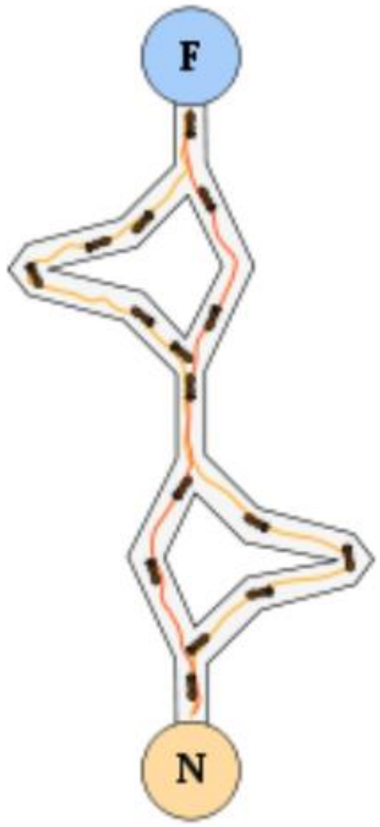
Ant Colony Optimization (ACO)

- Finding the shortest path

1. As they moved, ants deposit pheromone.
2. Pheromone decays in time.
3. Ants follow path with highest pheromone concentration.
4. Without pheromone, equal probability of choosing short or long path.

Shorter path allows higher number of passages and therefore pheromone level will be higher on shorter path.

Ants will increasingly tend to choose shorter path.



Source: Goss et al., 1989, Deneubourg et al., 1990

What If Ants Encounter Obstacle?

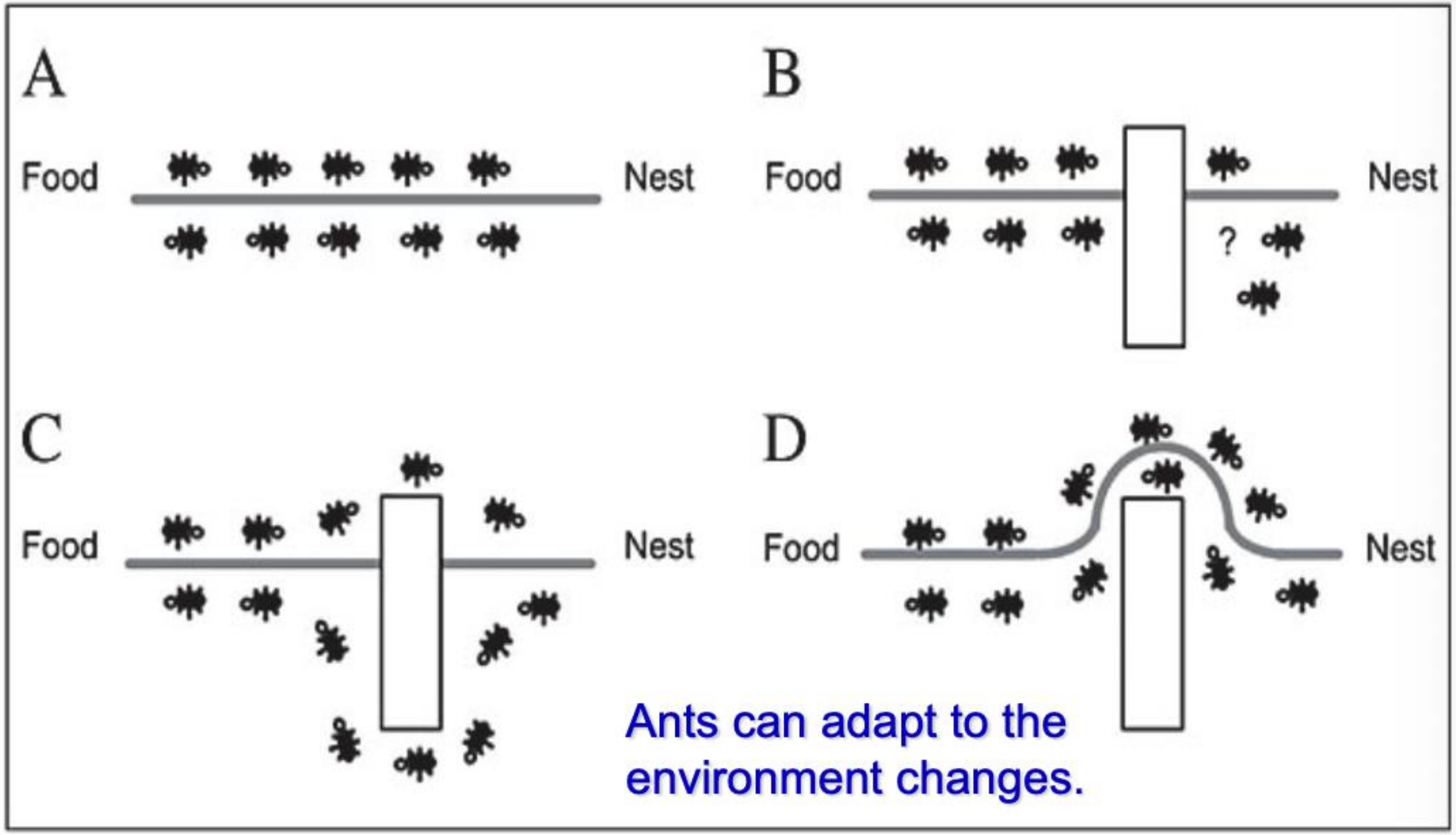
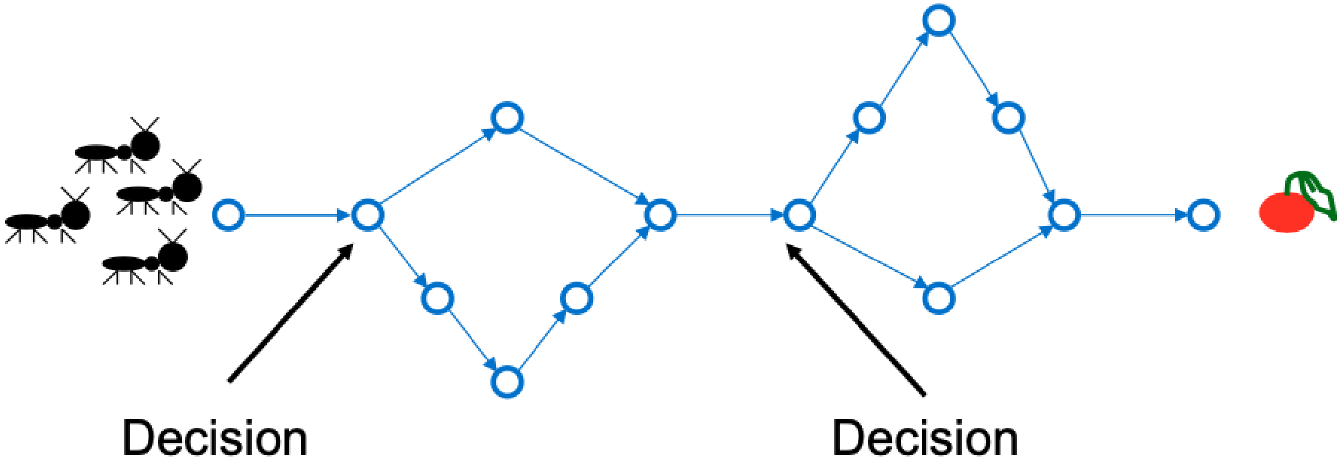


Figure 2. A. Ants in a pheromone trail between nest and food; **B.** an obstacle interrupts the trail; **C.** ants find two paths to go around the obstacle; **D.** a new pheromone trail is formed along the shorter path.

Source: M. Perretto, and H. S. Lopes, Reconstruction of Phylogenetic trees using the ant colony optimization paradigm, Genet. Mol. Res. 4 (3): 581-589, 2005.

Simple ACO Algorithm



- Finding the shortest path between two nodes on a graph $G = (V, E)$

Transition Probability:

Decision policy to determine the next link of the path

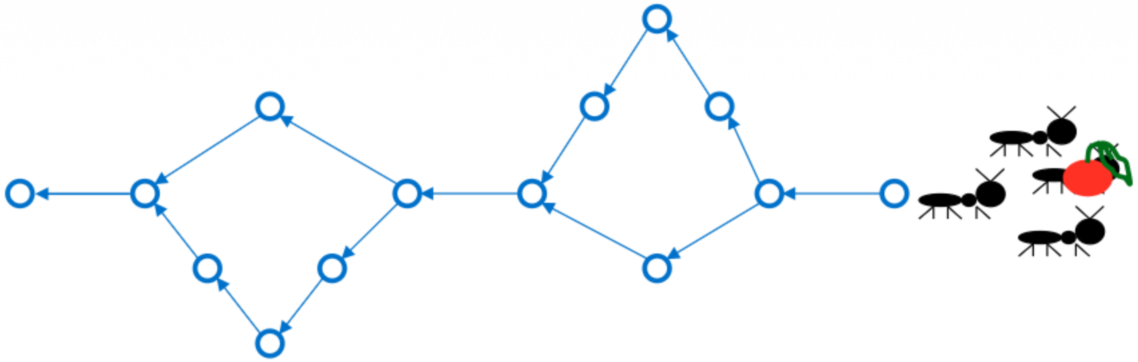
$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}(t)}{\Sigma \tau_{ij}^{\alpha}(t)} & \text{If } j \in N_i^k \\ 0 & \text{If } j \notin N_i^k \end{cases}$$

Pheromone concentration of edge (i, j)

α is a positive constant used to amplify the influence of pheromone concentrations.

Set of feasible nodes connected to node i, with respect to ant k

Simple ACO Algorithm



Once all ants have constructed a complete path from the origin node to the destination node, each ant retraces its path to the source node deterministically, and deposits a pheromone amount.

$$\Delta\tau_{ij}^{a}(t) \propto \frac{1}{L^{k}(t)}$$

Length of the path constructed by ant k at time step t

$$\tau_{ij}(t) = (1-\rho)\tau_{ij}(t)$$

Pheromone evaporating rate

Number of ants

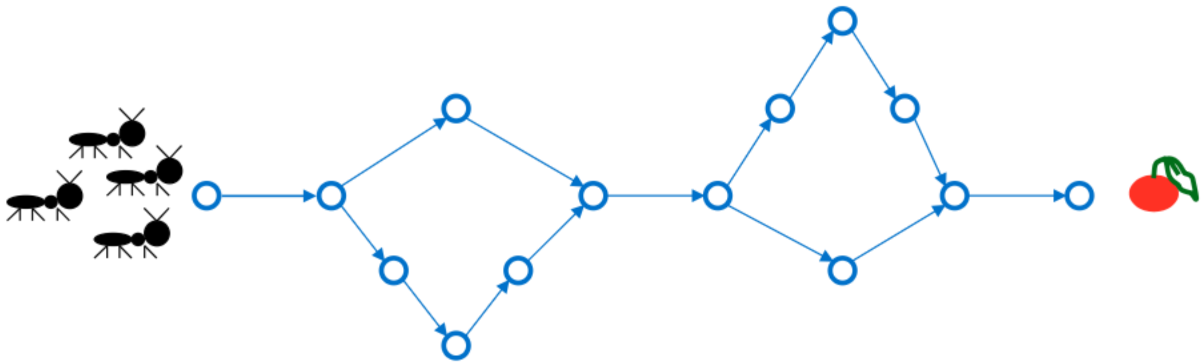
$$\tau_{ij}(t+1) = \tau_{ij}(t) + \Sigma_{k=1}^{n}\Delta\tau_{ij}^{k}(t)$$

The more pheromones evaporate, the more random the search becomes, facilitating better exploration.

Simple ACO Algorithm

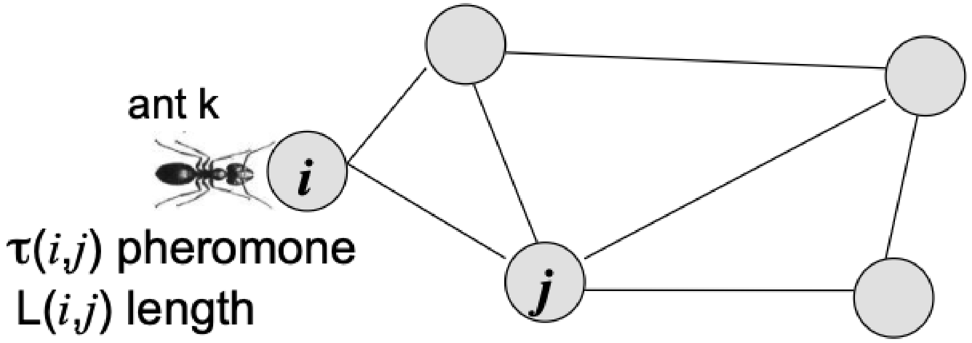
- If $x^k(t)$ denotes a solution at time step t, then

  $f(x^k(t))$ expresses the quality of the solution (the constructed path)



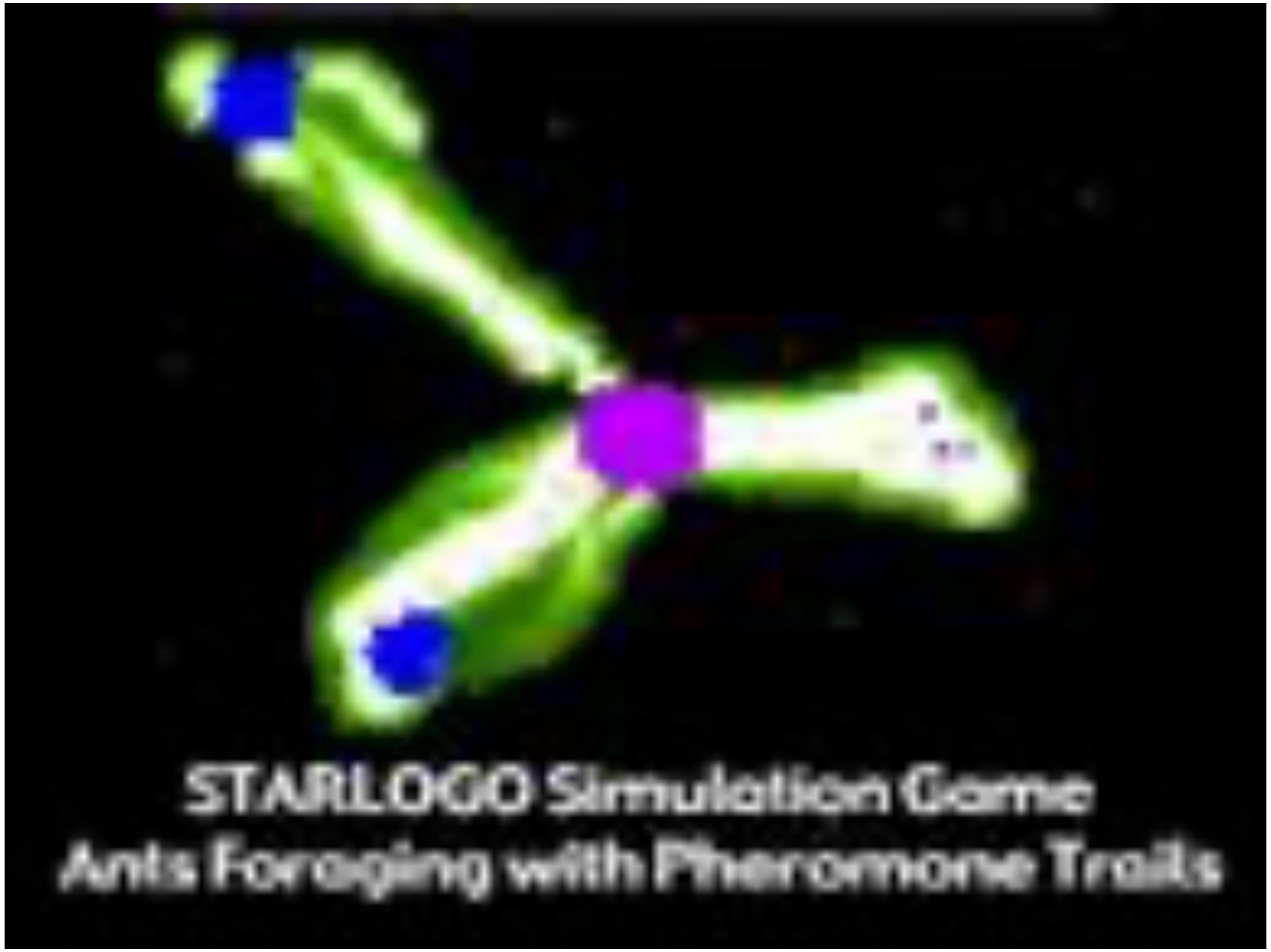
- **Termination Criteria:**
  - Terminate when a maximum number of iternations, n_t, has been exceeded.
  - Terminate when an acceptable solution has been found, with $f\left(x^k(t)\right) \leq \epsilon$ or $f\left(x^k(t)\right) \geq \epsilon$
  - Terminate when all ants (or most of the ants) follow the same path.

Ant Colony Optimization (ACO)



- Each ant generates a complete tour of nodes using **probabilistic transition rule** encouraging choice of edge with high pheromone and short distance.
- **Pheromone level** on each edge is updated by considering evaporation and deposit by each ant.
- Pheromone levels only of edges traveled by best ant are increased in inverse proportion to length of path.
- Result is that edges belong to short tours receive greater amount of pheromone.

Ant Colony Optimization (ACO)

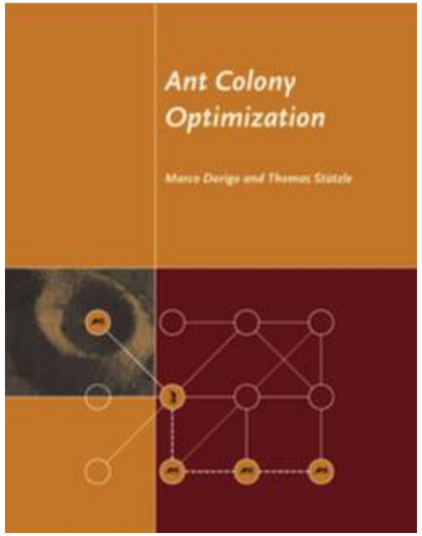


Source: https://www.youtube.com/watch?v=kN0M49iqFRc

ACO Performance

- Find **best** solution on "**small**" problems.
- Find **good** solutions on **large** problems compared to other techniques.
- Find **best** solution on **large** problems when coupled with other search techniques.
- Can operate on **dynamic problems** (e.g., node malfunctioning) that require fast rerouting.

Good Material: Dorigo and Stuetzle, 2005, MIT Press.

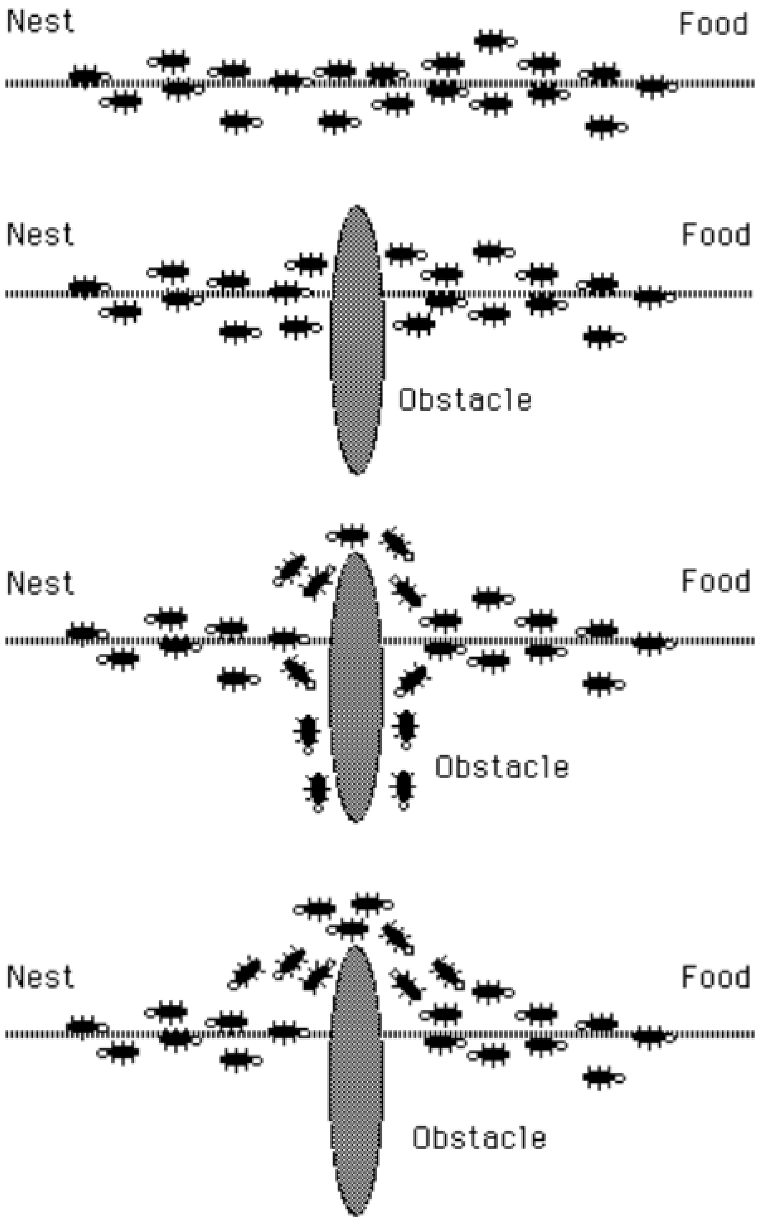


Design Issues of ACO

- As problems become more complex, the parameter settings of the simple ACO become increasingly important to obtain convergence to the optimal solution:
  - Pheromone updates based on solution quality are important for fast convergence.
  - Large values for parameter α lead to a strong emphasis of initial, random fluctuations and to bad algorithm behavior as it gives excessive importance to pheromone, specially the initial random pheromones, which may lead to rapid convergence to sub-optimal paths.
  - The larger the number of ants, the better the convergence behavior of the algorithm, although this comes at the cost of longer simulation times.
  - Pheromone evaporation is important when trying to solve more complex problems. If $\rho = 0$ (no evaporation), the algorithm does not converge; if pheromone evaporates too much, the algorithm often converged to sub-optimal solutions for complex problems.

Stigmergy

- The term indicates indirect communication among individuals through modification of the environment.
- For example, some ants leave a chemical (pheromone) trail behind to trace the path. The chemical decays over time.
- This allows other ants to find the path between the food and the nest. It also allows ants to find the shortest path among alternative paths.



Stigmergy provide flexibility and robustnes

Self-Organization

- Four key components to perform self-organization:
  - **Positive Feedbacks:**

    e.g., the recruitment of the individuals with pheromone trails
  - **Negative Feedbacks:**

    e.g., negative feedback in the form of pheromone evaporation.
  - **Amplification of Fluctuations - Randomness:**

    Randomness can promote the exploration and discovery of new solutions.
  - **Multiple Interactions:**

    e.g., the action of pheromone following can interact with pheromone-laying action if the density of the pheromone is sufficient. As the pheromone substance can evaporate over time, multiple interactions are required to maintain the pheromone density level.

Particle Swarm Optimization (PSO)

Variations of Swarm Intelligence

Particle Swarm Optimization (PSO)

Particle Swarm Optimization is an optimization algorithm inspired upon birds flocking to find the best food area.

A caricature scenario:

The flock wants to find the area with the highest concentration of food (insects).

Birds

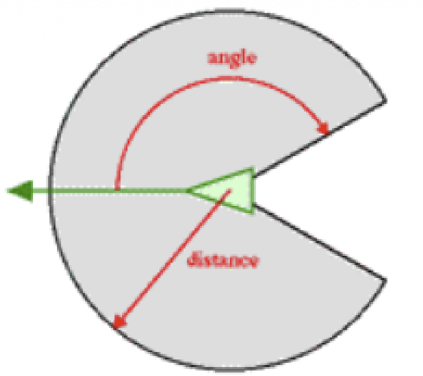
- Do not know where that area is, but each bird can shout to their neighbors how many insects are at its location, and
- Can remember their own location where they found the highest concentration of food so far.

Flock of Birds

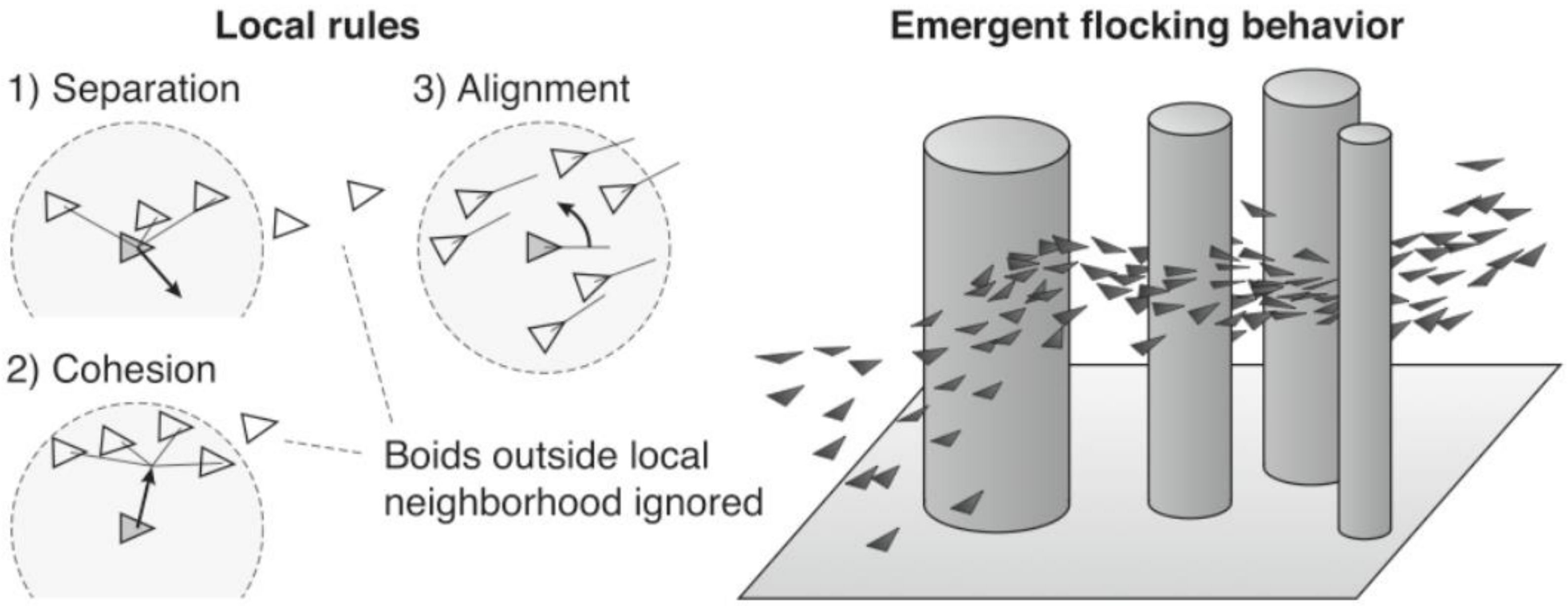
Particle Swarm Optimization (PSO)

Source: https://www.youtube.com/watch?v=0dskCpuxgtI

Raynolds Flocking (1987)



- **Sensing : Boid** (*Bird-oid Object*) perceives angle and distance of neighboring boids



1. **Separation:** Boid maintains a given distance from other boids.
2. **Cohesion:** Boid moves towards center of mass of neighboring boids
3. **Alignment:** Boid aligns its angle along those of neighboring boids.

Examples of Character Animation

Emergent coordinated behavior. **The approach is applicable to any type of animated characters in groups where behavior coordination is used.**

The Lion King, 1994 (Walt Disney)

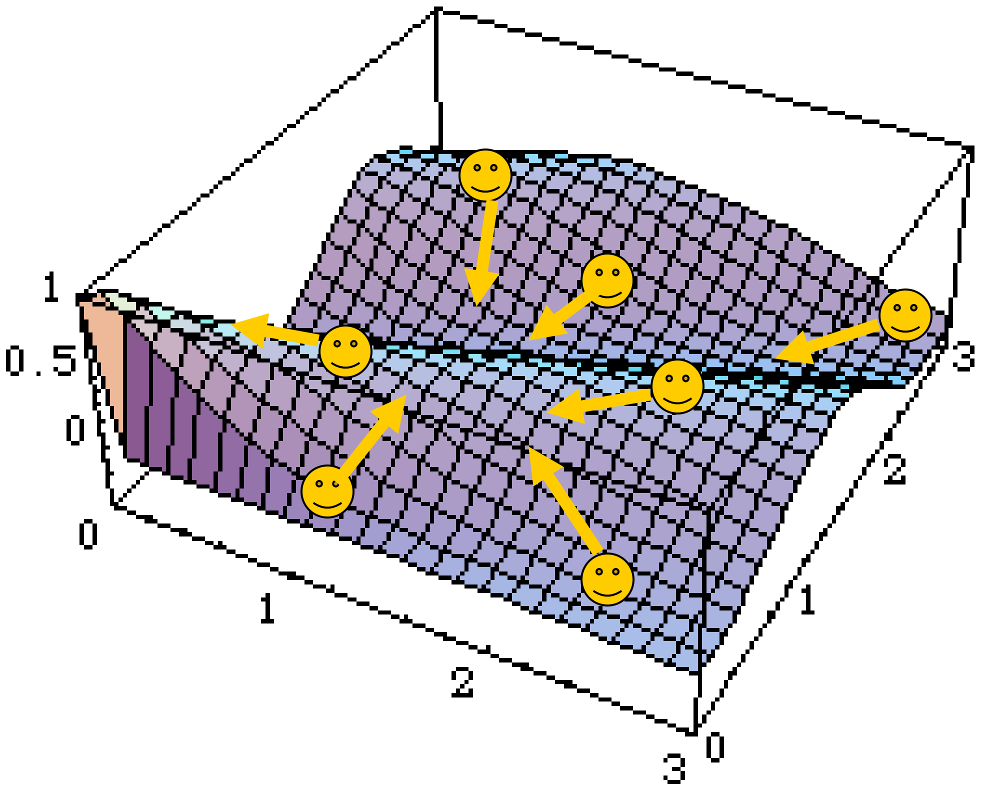
Source: https://www.red3d.com/cwr/boids/

From Birds to Particles

The food concentration describes the search space of the optimization problem and the birds are the local solution for that problem. They are called particles because they are very simple.

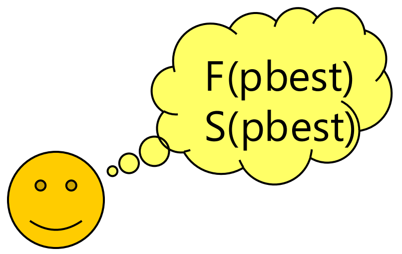
A particle p is described by:

- $x(\cdot)$ denotes its position in either 2 or 3 dimensional space, e.g., $x(x, y)$ or $x(x, y, z)$
- $v(\cdot)$ denotes its velocity, e.g., (for discrete cases) angle and distance of next step.
- $f(\cdot)$ denotes its performance, e.g., value of the function at its location.

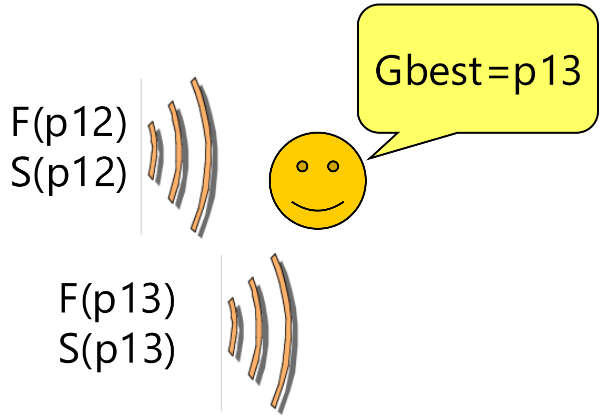


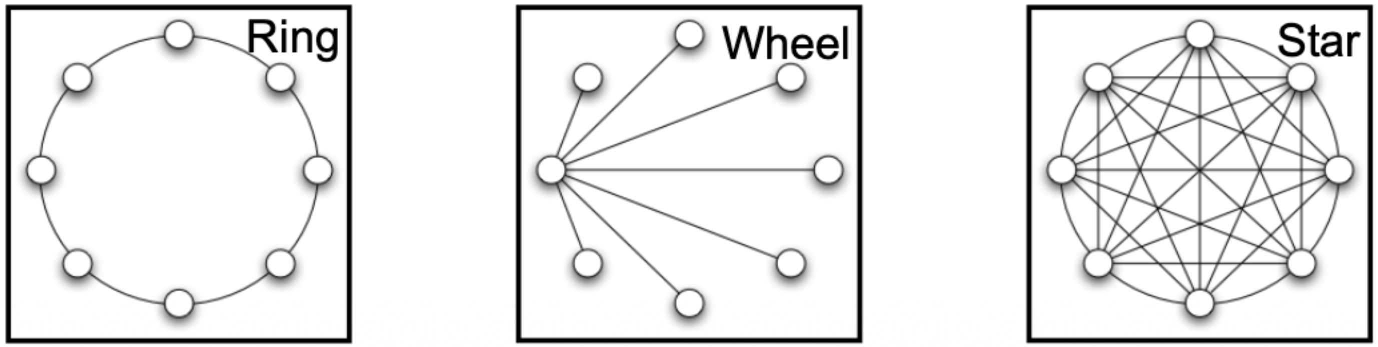
Particle’s Perception

- A particle remembers the position where it obtained the best performance so far (i.e., personal best or “pbest”)



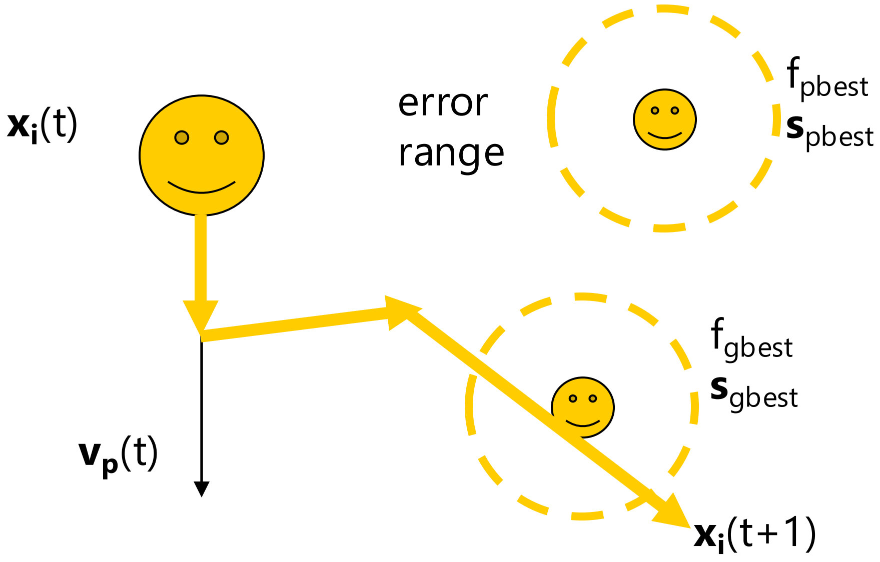
- A particle perceives performances and positions of neighboring particles.
- It can also tell which is the best particle among its neighbors (i.e., global best or “gbest”, or local best or “lbest” depending on the neighborhood topology





Particle's Actions

A particle computes the next position by taking into account a fraction of its current velocity v, the direction to its previous best location $pbest$, and the direction to the location of the best neighbor $gbest$. The movement towards other particles has some errors.



$$v_i(t+1) = \underbrace{\mathfrak{w} v_i(t)}_{\text{Previous Velocity}} + \underbrace{c_1 r_1 \left(x_{pbest,i} - x_i(t)\right)}_{\text{Recognition Component}} + \underbrace{c_2 r_2 (x_{gbest} - x_i(t))}_{\text{Social Component}}$$

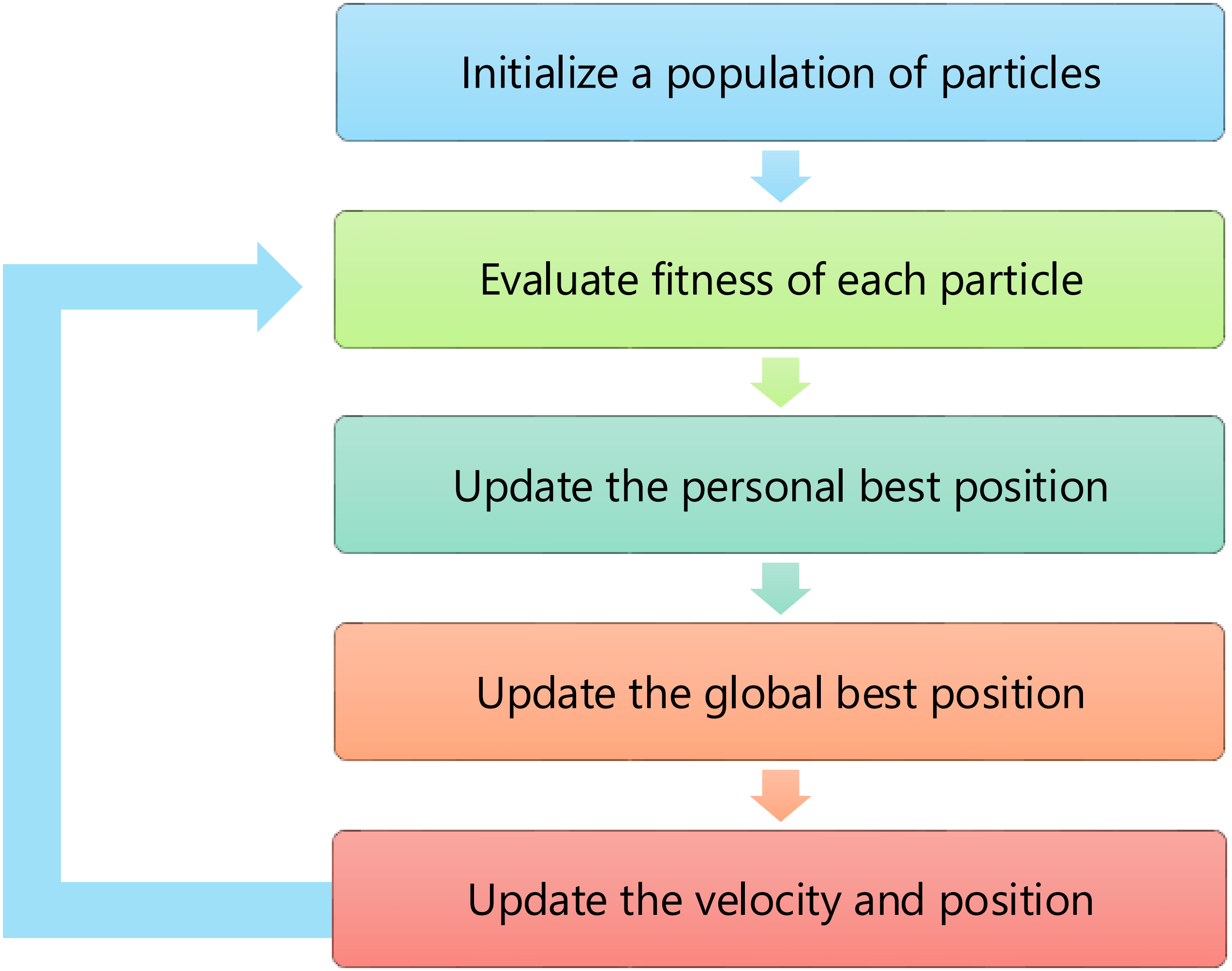
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

Where $\mathfrak{w}$ is a coefficient between 0 and 1,

c_1 and c_2 are acceleration constants between 0 and 1,

r_1 and r_2 are random numbers between 0 and 1

Particle Swarm Optimization Algorithm

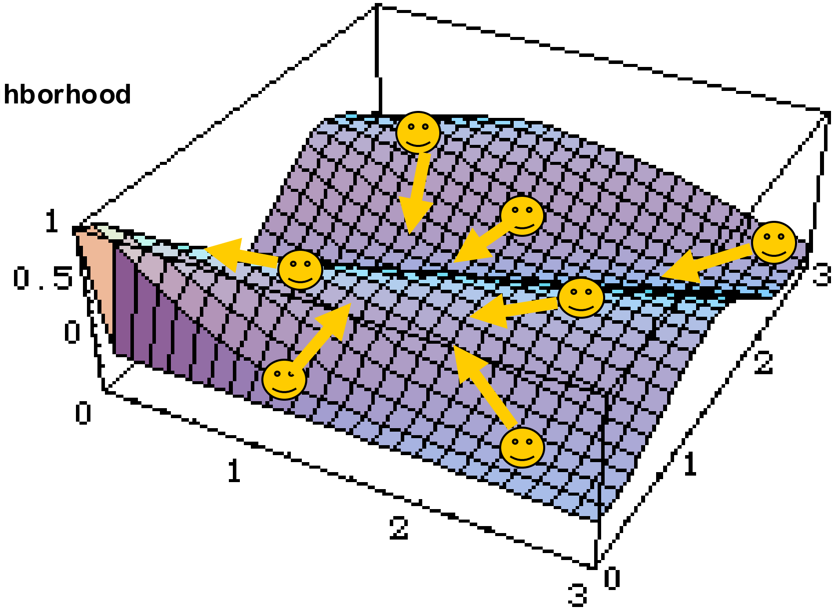


Source: Eberhart, R. and Kennedy, J., 1995. A New Optimizer Using Particle Swarm Theory". Proceedings of 6th International Symposium on Micro Machine and Human Science

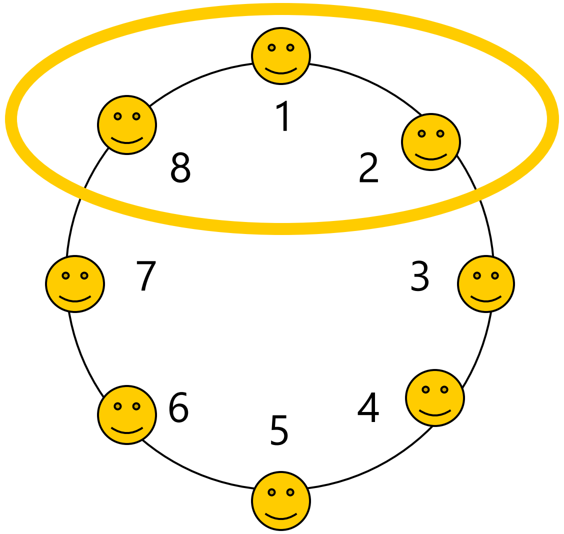
Particle Swarm Optimization - Settings

- **Swarm size:** Typically 20 particles for problems with dimensionality 2 – 200
- **Initial position of each particle:** Random
- **Neighborhood topology:** Global Geographical or sodea : Global, Geographical or Social (list based)

Geographical neighborhood



Social neighborhood



- **Neighborhood size:** Typically 3 to 5
- Set **max_velocity** to v_{max}; If $v(i+1)$ is larger, clip it
- Set **coefficient values** for ϖ, c1, c2
- Iterate until best solution is found or no further improvement or max iteration is reached.

Design Issues of PSO

- Due to the greater particle interconnectivity of the $gbest$ PSO, it converges faster than the $lbest$ PSO. However, this faster convergence comes at the cost of reduced diversity compared to the $lbest$ PSO.
- As a consequence of its greater diversity (which allows larger portions of the search space to be explored), the $lbest$ PSO is less susceptible to being trapped in local minima. In general, depending on the problem, neighborhood structures such as the ring topology used in $lbest$ PSO improve performance.
- The stopping condition should not cause the PSO to converge prematurely, as this would lead to suboptimal solutions.
- The stopping condition should also prevent oversampling of the fitness function. If frequent evaluations of the fitness function are required, the computational complexity of the search process can increase significantly.

PSO Versus Evolution Computation

- As in Evolutionary Computation, PSO works with *a population* and incorporates *random factors* to update solutions.
- Contrary to Evolutionary Computation, there is no generational change, no genome, and no competition among individuals; instead, cooperation is emphasized.
- A major challenge in PSO is transforming the problem parameters into a form that can be encoded and effectively searched by particles.
- The most successful applications to date include the large class of Traveling Salesman Problems and the optimization of neural network weights.

Challenges in Swarm Intelligence Implementation

- Identify individual behavioral rules that produce the desired swarm behavior (reverse engineering).
  - This challenge may be manageable because the behavioral rules are expected to be relatively simple. Often, these rules are hand-designed, though in some cases they are evolved.
- Ensure that the emergent behavior is stable.
  - Dynamical systems theory can help characterize and predict swarm behavior, as a swarm can be described as a system of elements with both negative and positive interactions moving through space and time. However, modeling non-linear interactions remains difficult.

Project Assignment 2: Swarm Intelligence

- Use the problem you formulated and find an optimal solution using one of the following algorithms:
  - **Hybrid between ACO and EC**
  - **Hybrid between PSO and EC**
  - **Artificial Bee Colony (ABC)**
  - **Bat Algorithm**
  - **Firefly Algorithm**

End of the Lecture

Please don’t hesitate to raise your hand and ask questions if you're curious about anything!