SWARM INTELLIGENCE - I

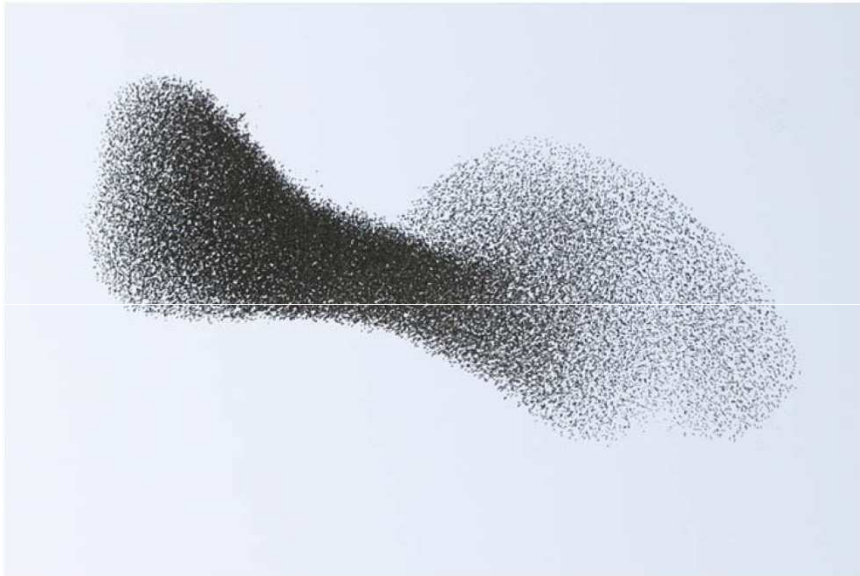
Swarm Intelligence

- Any attempt to design algorithms or distributed problem solving devices inspired by the collective behaviour of social insect colonies and other animal societies

(Bonabeau et al., 1999)

Why Imitate Swarms?

- Emergent, collective intelligence of groups of simple agents.



Harmonious Flight

The ability of animal groups—such as this flock of starlings—to shift shape as one, even when they have no leader, reflects the genius of collective behavior.

http://ngm.nationalgeographic.com/2007/07/swarms/swarms-photography

Swarm Intelligence in Nature

- Two categories
  - Species whose individuals form a swarm because they benefit in some way and
  - Social insects – which live in colonies whose members cannot survive on their own.

Interesting Characteristics of Social Colonies

- Flexible: the colony can respond to internal perturbations and external challenges
- Robust: Tasks are completed even if some individuals fail
- Decentralized: there is no central control in the colony
- Self-organized: paths to solutions are emergent rather than predefined

Emergent Behaviour

- **Emergence** is the way complex systems and patterns arise out of a multiplicity of relatively simple interactions.

http://en.wikipedia.org/wiki/Emergence

Interactions

- Self-organization in social insects often requires interactions among insects.
- Interactions
  - Direct
  - Indirect
    - Stigmergy

Stigmergy

- Stigmergy is a method of indirect communication in a self-organizing emergent system where its individual parts communicate with one another by modifying their local environment.
- The two main characteristics of stigmergy that differentiate it from other forms of communication are the following.
  - Stigmergy is an indirect, non-symbolic form of communication mediated by the environment: insects exchange information by modifying their environment; and
  - Stigmergic information is local: it can only be accessed by those insects that visit the locus in which it was released (or its immediate neighborhood).

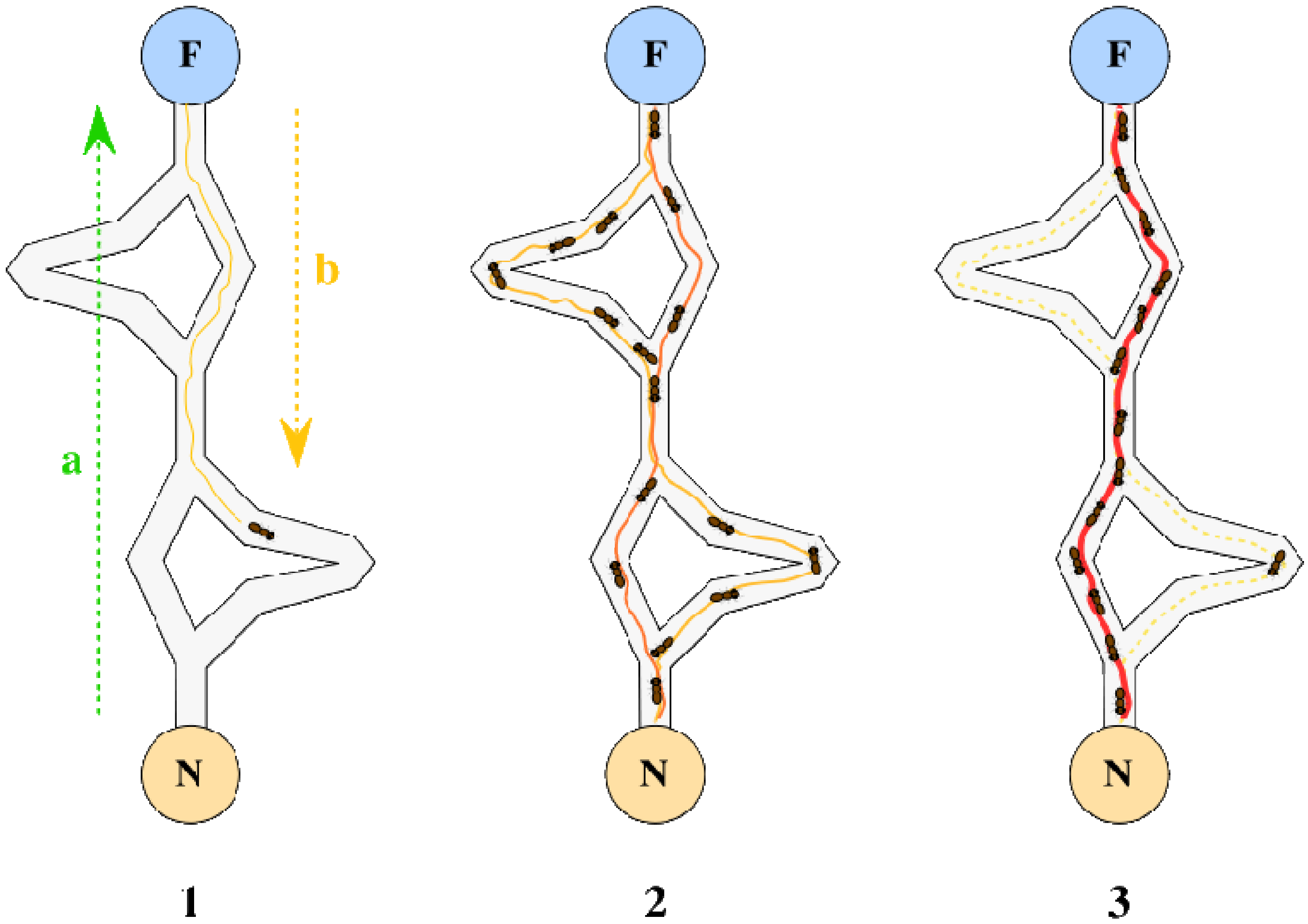
http://en.wikipedia.org/wiki/Stigmergy

Stigmergy

- Stigmergy was first observed in social insects.
- Ants:
  - Ants exchange information by laying down pheromones on their way back to the nest when they have found food.
  - In that way, they collectively develop a complex network of trails, connecting the nest in the most efficient way to the different food sources.

http://en.wikipedia.org/wiki/Stigmergy

Real Ants



http://en.wikipedia.org/wiki/Ant_colony_optimization

StarLogo Demo: Ants

StarLogo is developed at Media Laboratory and Teacher Education Program, MIT, Cambridge, Massachusetts.

- A programmable modeling environment for exploring the workings of decentralized systems -- systems that are organized without an organizer, coordinated without a coordinator.
- With StarLogo, you can model many real-life phenomena, such as bird flocks, traffic jams, ant colonies, and market economies.
- Designed to help students (as well as researchers) develop new ways of thinking about and understanding decentralized systems.
- StarLogo is an extension of the Logo programming language.

http://education.mit.edu/starlogo/

Sigmergy in General

- Stigmergy is not restricted to eusocial creatures, or even to physical systems.
- On the Internet there are many emergent phenomena that arise from users interacting only by modifying local parts of their shared virtual environment.
  - Wikipedia is an example of this.
  - The massive structure of information available in a wiki, or an open source software project such as the Linux kernel could be compared to a termite nest
  - one initial user leaves a seed of an idea (a mudball) which attracts other users who then build upon and modify this initial concept, eventually constructing an elaborate structure of connected thoughts.

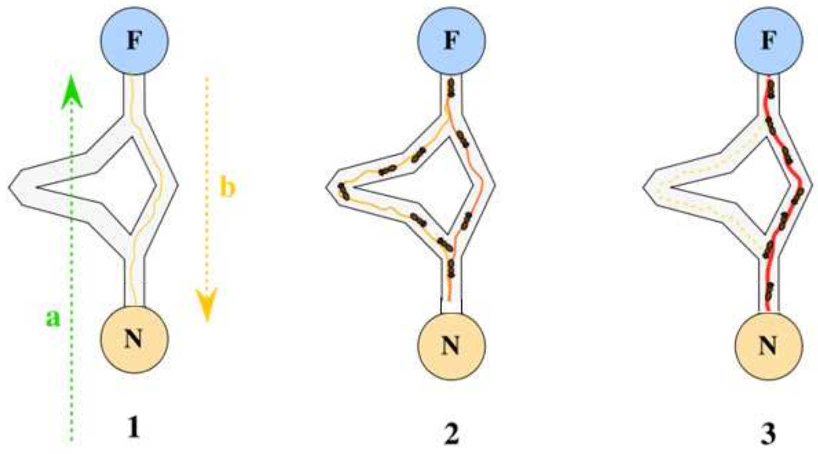
Ant Colony Optimization (ACO)

ACO - Inspiration

- The inspiring source of ACO is the pheromone trail laying by real ants.
- The pheromone trails in ACO serve as a distributed, numerical information which the ants use to probabilistically construct solutions to the problem being solved and which the ants adapt during the algorithm's execution to reflect their search experience

Towards an ACO Algorithm

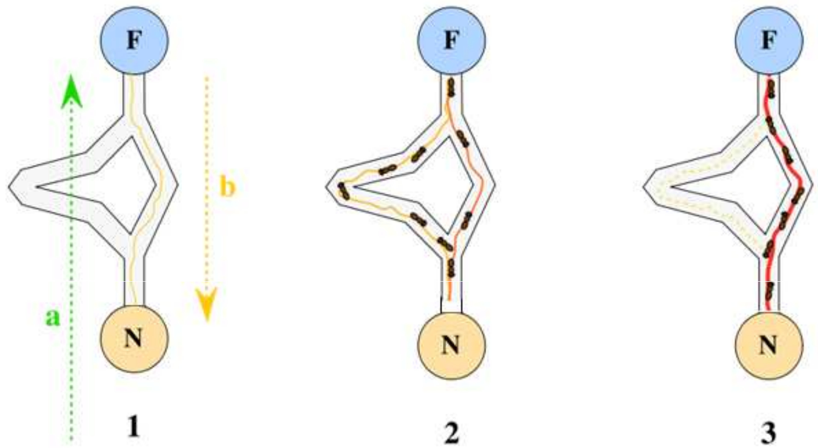
- Assume, to begin with, all ants are in the nest. There is no pheromone in the environment.
- The foraging starts. In probability, 50% of the ants take the short path and 50% take the long path to the food source



http://en.wikipedia.org/wiki/Ant_colony_optimization

Towards an ACO Algorithm

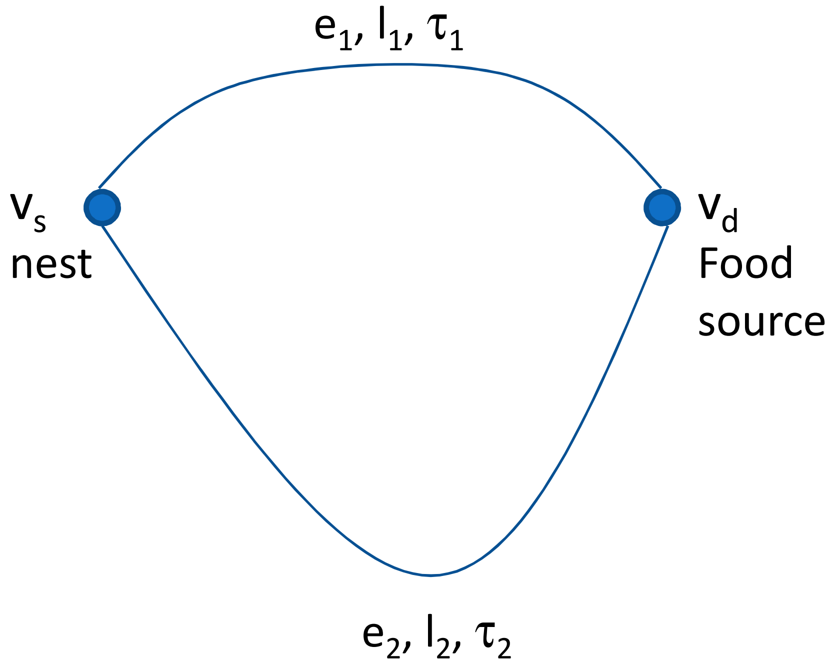
- The ants that have taken the short path have arrived earlier at the food source. Therefore, while returning, the probability that they will take the shorter path is higher.
- The pheromone trail on the short path receives, in probability, a stronger reinforcement and the probability of taking this path grows. Finally due to evaporation of the pheromone on the long path, the whole colony will, in probability, use the shorter path



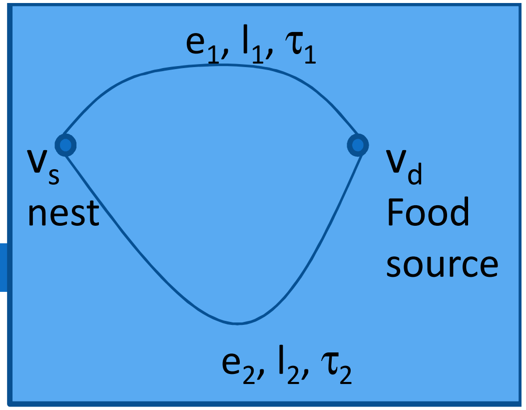
http://en.wikipedia.org/wiki/Ant_colony_optimization

Towards an ACO Algorithm

- $l_2 > l_1$
- Real ants deposit pheromone on the paths on which they move.
- We introduce an artificial pheromone value τ_i for each of the two links e_i, $i = 1, 2$.
- This indicates the strength of the pheromone trail on the corresponding path.
- Introduce n_a artificial ants.



Towards an ACO Algorithm



- Each ant behaves as follows:
  Starting from v_s, an ant chooses with probability

$$\mathbf{p}_i = \frac{\tau_i}{\tau_1 + \tau_2} \quad , \; i = 1, 2,$$

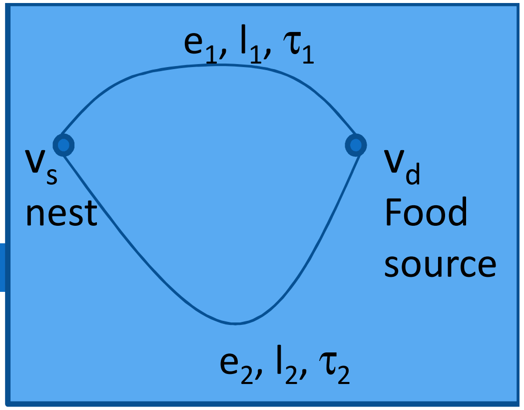
  between path e1 and path e2 for reaching the food source v_d.

- For returning from v_d to v_s, an ant uses the same path as it chose to reach v_d, and it changes the artificial pheromone value associated with the used edge.

$$\tau_i \leftarrow \tau_i + \frac{Q}{l_i}$$

  where the positive constant Q is a parameter of the model.

Towards an ACO Algorithm



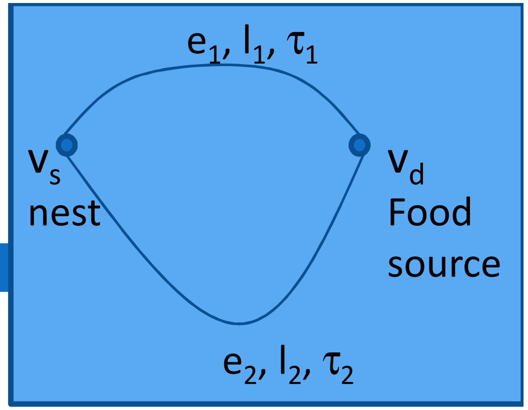
- In nature the deposited pheromone is subject to an evaporation over time. This is simulated as

$$\tau_i \leftarrow (1-\rho)\cdot\tau_i \qquad , \; i = 1,2$$

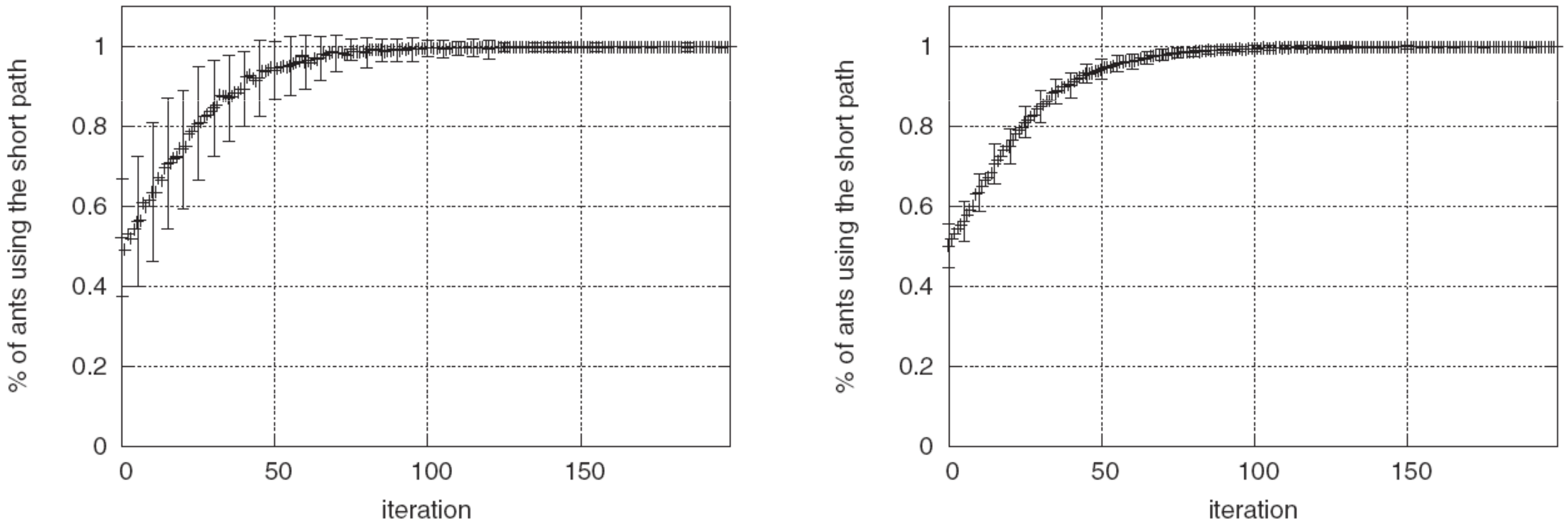
  The parameter $\rho \in (0, 1]$ is a parameter that regulates the pheromone evaporation.

- The foraging of an ant colony is in this model iteratively simulated as follows:
  - At each step (or iteration) all the ants are initially placed in node v_s.
  - Then, each ant moves from v_s to v_d choosing a path with probability.
  - Pheromone evaporation is performed
  - Finally, all ants conduct their return trip and reinforce their chosen path.

A Simulation



- $l_1 = 1$, $l_2 = 2$, $Q = 1$.
- The two pheromone values were initialized to 0.5



(a) Colony size: 10 ants (b) Colony size: 100 ants

Results of 100 independent runs (error bars show the standard deviation for each 5th iteration).

Swarm Intelligence in Optimization, Christian Blum and Xiaodong Li

Combinatorial Optimization Problem

A *model* $P = (\mathbf{S}, \Omega, f)$ *of a combinatorial optimization problem consists of:*

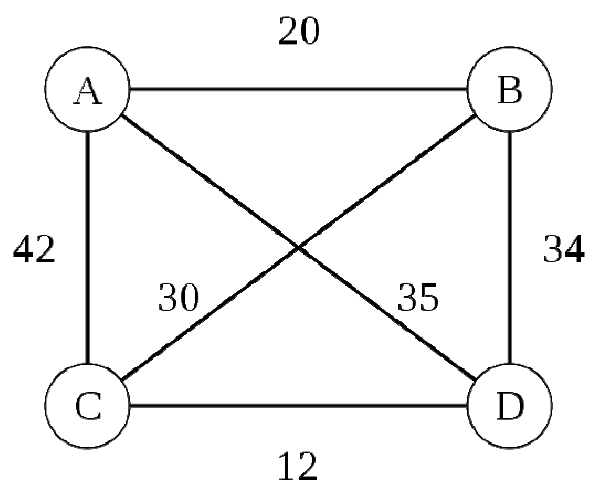
- a *search space* $\mathbf{S}$ *defined over a finite set of discrete decision variables* $X_i, \; i = 1, \ldots, n$;
- *a set* Ω *of constraints among the variables; and*
- *an objective function* $f : \mathbf{S} \to \mathbb{R}_0^+$ *to be minimized.*

The generic variable X_i *takes values in* $\mathbf{D}_i = \{v_i^1, \ldots, v_i^{|\mathbf{D}_i|}\}$. *A feasible solution* $s \in \mathbf{S}$ *is a complete assignment of values to variables that satisfies all constraints in* Ω. *A solution* $s^* \in \mathbf{S}$ *is called a global optimum if and only if:* $f(s^*) \le f(s) \; \forall s \in \mathbf{S}$.

Ant Colony Optimization, Marco Dorigo, Mauro Birattari, and Thomas Stutzle

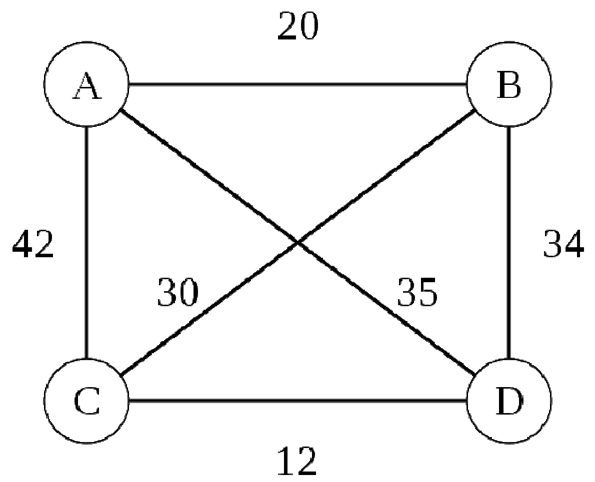
The Travelling Salesman Problem

- Given a completely connected, undirected graph G = (V,E) with edge weights.
  - Vertices V represent the cities, and
  - edge weights represent the distances between the cities.
- **Goal**: find a closed path in G that contains each node exactly once (a *tour*) and whose length is minimal.
- Search space *S* consists of all tours in *G*. The objective function value *f(s)* of a tour *s* ∈ *S* is defined as the sum of the edge weights of the edges that are in *s*.



Solution vs Solution Components

- Solution
  - A complete tour
- Solution Components
  - The edges of the TSP graph



Heuristic Search

- Construction Algorithms
- Local Search
- Population Based

Construction Algorithms

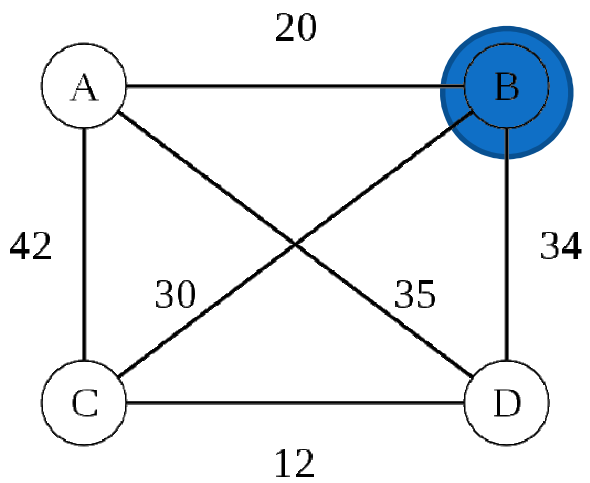
```
Procedure GreedyConstructionHeuristic
  s_p = emptySolution;
  while s_p not a complete solution do
      e = GreedyComponent();
      s_p = s_p ⊗ e;
  end
      return s_p;
end
```

The Ant Colony Optimization Heuristic: Algorithms, Applications and Advances, Dorigo and Stutzle

Construction Algorithms and TSP

- Nearest Neighbour Tour

  Build a tour to start from some initial city and always choose to go to the closest still unvisited city before returning to the start city.

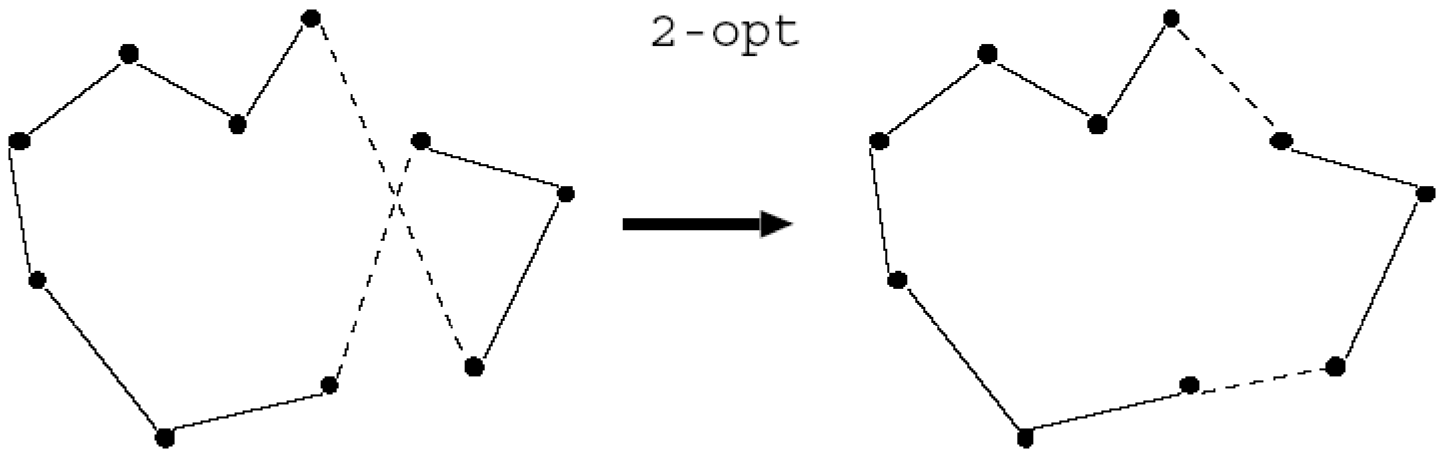


Local Search

```
Procedure IterativeImprovement (s∈S)
  s' = Improve(s);
  while s' ≠ s do
      s = s';
      s' = Improve(s);
  end
      return s;
end
```

The Ant Colony Optimization Heuristic: Algorithms, Applications and Advances, Marco Dorigo and Thomas Stutzle

Local Search and TSP



- Need: a neighborhood examination scheme that defines how the neighborhood is searched and which neighbour solution replaces the current one

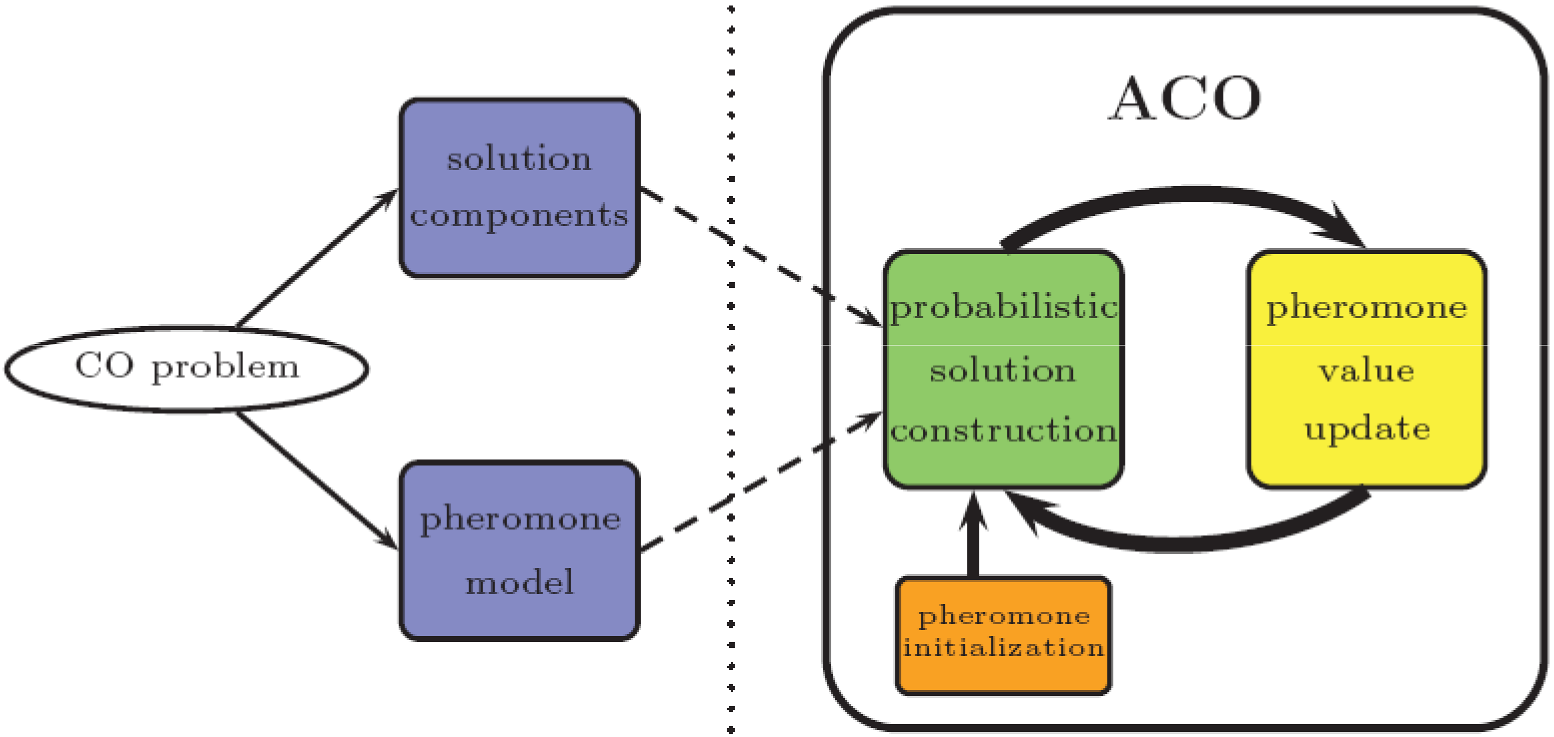
ACO: A Construction Algorithm

- Artificial ants used in ACO are stochastic solution construction procedures that probabilistically build a solution by iteratively adding solution components to partial solutions by taking into account
  - heuristic information on the problem instance being solved, if available, and
  - (artificial) pheromone trails which change dynamically at run-time to reflect the agents' acquired search experience.

Some Successful ACO Algorithms

ALGORITHM	AUTHORS
ANT SYSTEM (AS)	DORIGO ET AL.
ELITIST AS	DORIGO ET AL.
ANT-Q	GAMBARDELLA & DORIGO
ANT COLONY SYSTEM	DORIGO & GAMBARDELLA
$\mathcal{MAX}$-$\mathcal{MIN}$ AS	STÜTZLE & HOOS
RANK-BASED AS	BULLNHEIMER ET AL.
ANTS	MANIEZZO
BWAS	CORDON ET AL.
HYPER-CUBE AS	BLUM ET AL.

The ACO Metaheuristic



Swarm Intelligence in Optimization, Christian Blum and Xiaodong Li

Combinatorial Optimization Problem

A *model* $P = (\mathbf{S}, \Omega, f)$ *of a combinatorial optimization problem consists of:*

- a *search space* $\mathbf{S}$ *defined over a finite set of discrete decision variables* X_i, $i = 1, \ldots, n$;
- *a set* Ω *of constraints among the variables; and*
- *an objective function* $f : \mathbf{S} \to \mathbb{R}_0^+$ *to be minimized.*

The generic variable X_i *takes values in* $\mathbf{D}_i = \{v_i^1, \ldots, v_i^{|\mathbf{D}_i|}\}$. *A feasible solution* $s \in \mathbf{S}$ *is a complete assignment of values to variables that satisfies all constraints in* Ω. *A solution* $s^* \in \mathbf{S}$ *is called a global optimum if and only if:* $f(s^*) \leq f(s)\ \forall s \in \mathbf{S}$.

Ant Colony Optimization, Marco Dorigo, Mauro Birattari, and Thomas Stutzle

Pheromone Model

- The model of a combinatorial optimization problem is used to define the pheromone model of ACO.
- A value that can be assigned to a decision variable is a *solution component*.
- Let C be the set of all possible solution components.
- A pheromone value is associated with each possible *solution component*.
- Formally, the pheromone value τ_{ij} is associated with the solution component c_{ij}, which consists of the assignment $X_i = v_i^j$.

Problem Representation

- In ACO, an artificial ant builds a solution by traversing the fully connected *construction graph* G_C **(V, E)**
  - **V is a set of vertices and**
  - **E is a** set of edges.
- This graph can be obtained from the set of solution components **C in two ways:**
  - **Components** may be represented either by vertices or by edges.

Ant Behaviour

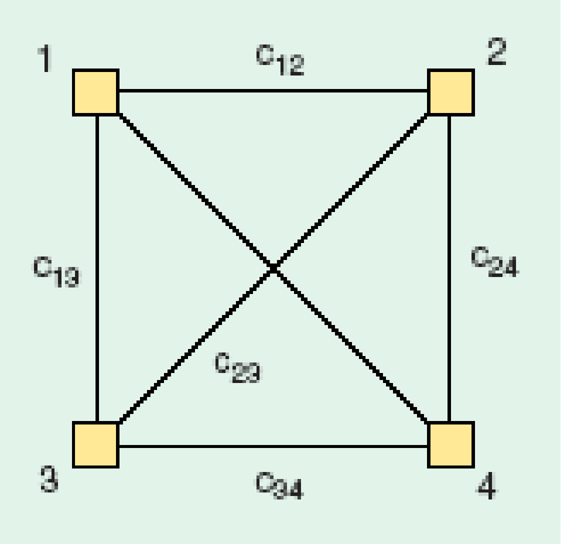
- An artificial ant moves from vertex to vertex along the edges of the graph, incrementally building a *partial solution.*
- *Additionally, the* ant deposits a certain amount of pheromone on the components; that is, either on the vertices or on the edges that it traverses.
- The amount τ *of pheromone deposited may* depend on the quality of the solution found.
- Subsequent ants use the pheromone information as a guide toward promising regions of the search space.

TSP

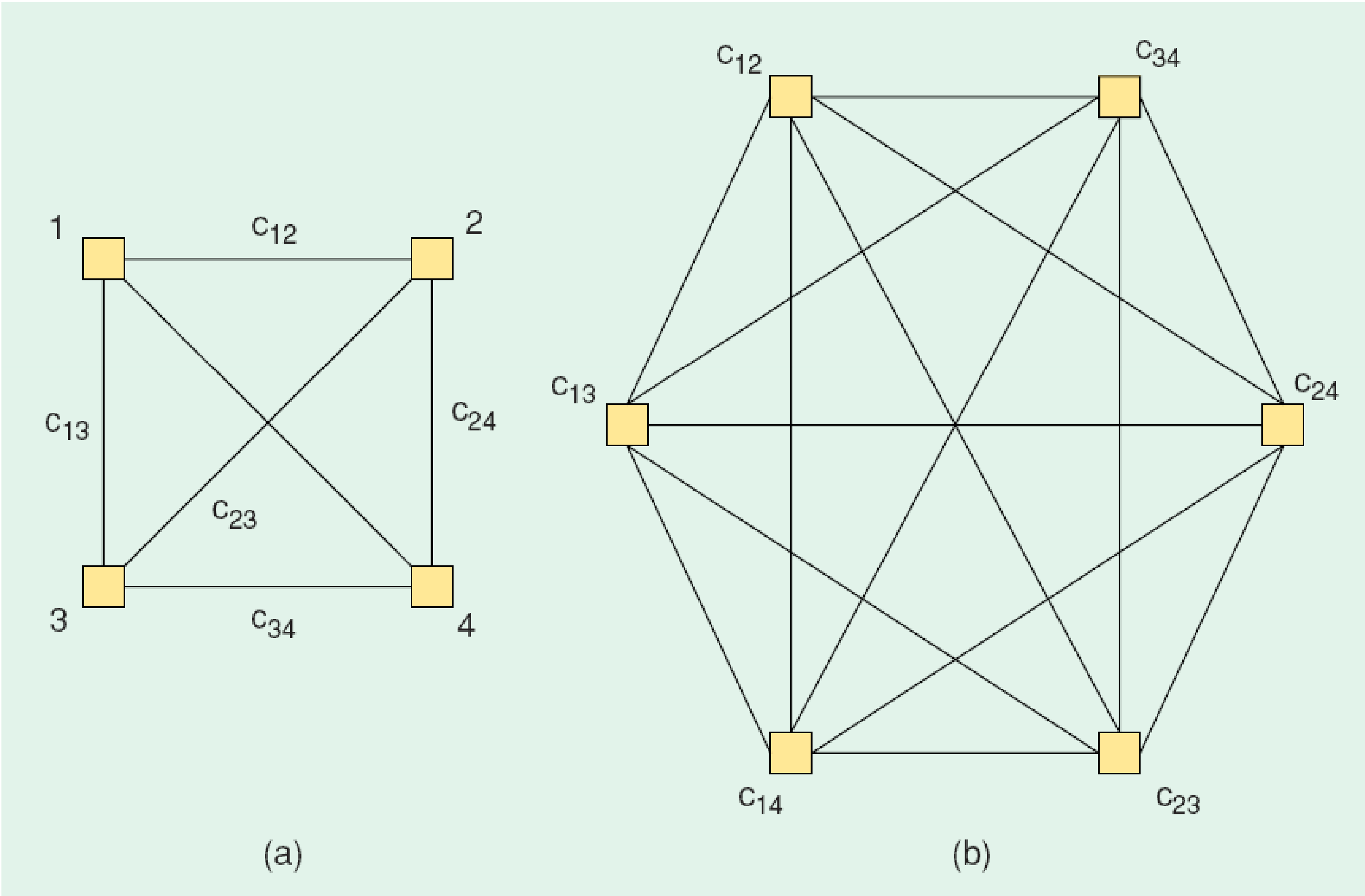
- In the TSP, a solution can be represented through a set of *n* variables, where *n* is the number of cities.
- Each of these variables is associated with a city.
- The variable X_i indicates the city to be visited after city *i*.
- Solution components are pairs of cities to be visited one after the other, in the given order
  - The solution component $c_{ij} = (i, j)$ indicates that the solution under analysis prescribes that city j should be visited immediately after city i.

TSP

- The construction graph is a graph in which the vertices are the cities of the original traveling salesman problem, and the edges are solution components.
- As a consequence, ants deposit pheromone on the edges of the construction graph.

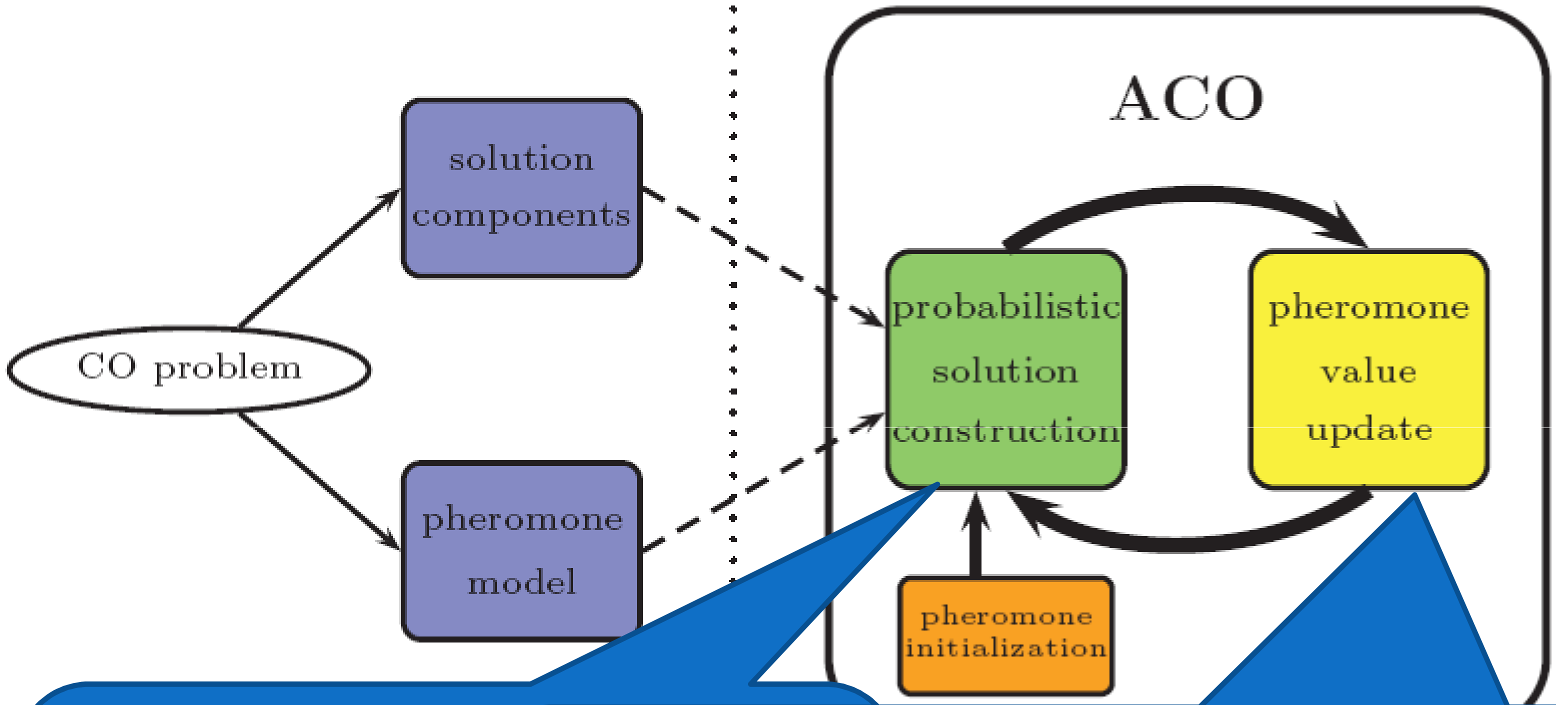


TSP



Components associated with edges Components associated with vertices

The ACO Metaheuristic



The process of const[…]
can be regarded a[…]
construction graph[…]
solution compone[…]
stochastic mechanis[…]
by the pheromone as[…]
of the ele[…]

The aim of the pheromone update is to increase the pheromone values associated with good or promising solutions, and to decrease those that are associated with bad ones. This is achieved (i) by decreasing all the pheromone values through *pheromone evaporation*, and (ii) by increasing the pheromone levels associated with a chosen set of good solutions.

The ACO Metaheuristic

Set parameters, initialize pheromone trails

while termination condition not met do

ProbabilisticSolution Construction

ApplyLocalSearch (optional)

PheromoneValueUpdate

endwhile

Each ant has a memory that it uses to store information about the path it has followed so far. Memory can be used for:

•Building feasible solutions

•Evaluating the solution found

•Retracing the path backward to deposit pheromone.

nd before
o improve the
local search. This
s optional
-the-art ACO

The Ant System (AS) and TSP

- The First ACO proposed in the literature
- Let the number of ants be m and cities n

The Ant System (AS) and TSP

Initialize the pheromone τ_{ij}, associated with the edge joining cities i and j ;

Place each ant on a randomly selected city;

Let *best* be the best tour;

t = 1;

While t < max_iterations **do**

    for each ant **do**

        build tour;

    end

    (*Perform local search*);

    Evaluate the length of the tour performed by each ant;

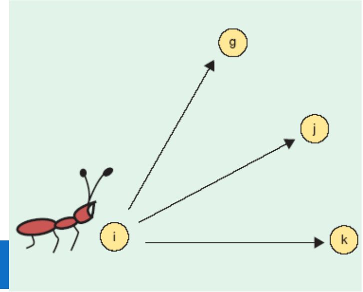
    If a shorter tour has been found update *best*;

    Perform pheromone update;

    t = t+1;

end

Solution Construction



- When ant k is in city i and has so far constructed the partial solution s_p, the probability of going to city j is given by:

$$p_{ij}^{k} = \begin{cases} \dfrac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in N(s_p)} [\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}} & \text{if } c_{ij} \in N(s_p) \\ 0 & \text{otherwise} \end{cases}$$

- $N(s_p)$ is the set of feasible components; that is, edges (i, l) where l is a city not yet visited by the ant k.
- α, β are parameters

$$p_{ij}^{k} = \begin{cases} \dfrac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum\limits_{l \in N(s_p)} [\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}} & \text{if } c_{ij} \in N(s_p) \\ 0 & \text{otherwise} \end{cases}$$

- The parameters α and β control the relative importance of the pheromone value versus the *heuristic information* η_{ij}, which is given by:

$$\eta_{ij} = \frac{1}{d_{ij}}$$

  where d_{ij} is the distance between cities *i* and *j*

  - If α is small then the closest cities are favored – classic greedy algorithm
  - A high value for α means that trail is very important and therefore ants tend to choose edges chosen by other ants in the past.
  - If β is small only pheromone +ve feedback at work – may choose non-optimal paths too quickly

Pheromone Update

- The main characteristic of AS is that, at each iteration, the pheromone values are updated by *all* the *m* ants that have built a solution in the iteration itself.

Pheromone Update

- The pheromone τ_{ij}, associated with the edge joining cities i and j, is updated as follows:

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

- where ρ= (0,1] is the evaporation rate, *m* is the number of ants, and $\Delta\tau_{ij}^{k}$ is the quantity of pheromone laid on edge (i, j) by ant k:

$$\Delta\tau_{ij}^{k} = \begin{cases} Q/L_k & \text{if ant k uses edge i,j} \\ 0 & \text{otherwise} \end{cases}$$

where *Q* is a constant, and L_k is the length of the tour constructed by ant *k*.

Elitist Strategy

- Give the best tour since the start of the algorithm, Tgb, a strong additional weight

$$\Delta\tau_{ij}^{gb} = \begin{cases} e / L^{gb} & \text{if } (i,j) \in T^{gb} \\ 0 & \text{otherwise} \end{cases}$$

Flocks, Herds and Schools

- What are the advantages for herd animals, flocks of birds and schools of fish that cause the formation of swarms?
  - Defense against predators
  - The disadvantage of sharing food sources can be outweighed by the reduced chances of finding no food at all, whenever the food is unpredictably distributed
  - Individuals may also increase their chances of finding a mate
  - For animals that travel great distances – like migratory birds– there is a decrease in energy consumption when moving in a tight formation.

Flocks, Herds and Schools

- Flocks, herds and schools can become very large and the individuals are both limited in their mental capacity and their perception
- Can be assumed that only simple, local rules control the movements of a single animal.
- The most basic behaviors seem to be an urge to stay close to the swarm and one to avoid collisions

- References in the next lecture slides

Questions?