Ant Systems and Particle Swarms

Thiemo Krink

EVALife Group, Dept. of Computer Science, University of Aarhus

Part I: Ant Systems

Ant Systems

Why are ants interesting?

- ants solve complex tasks by simple local means
- ant productivity is far better than the sum of their single activities
- ants are ‘grant masters’ in search and exploitation

Ant Systems

Why are ants interesting?

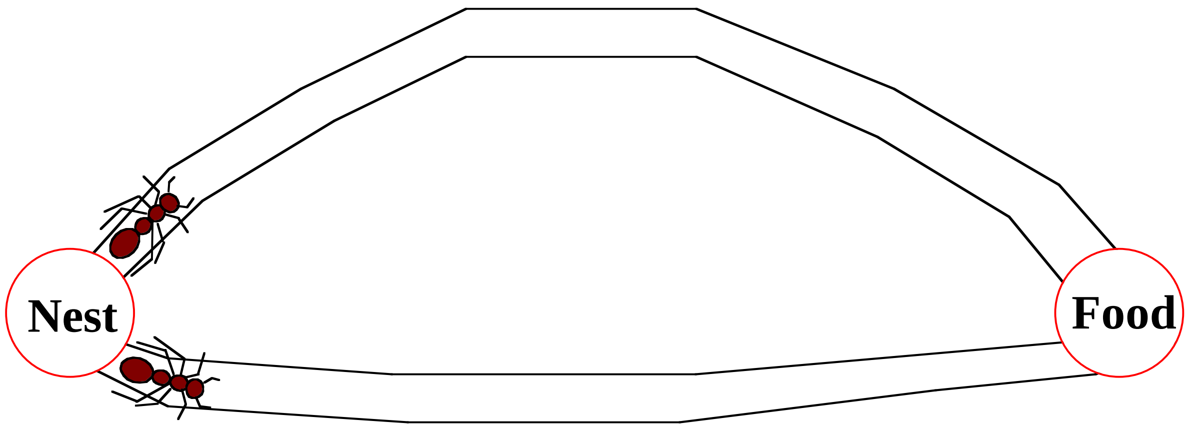
- ants solve complex tasks by simple local means
- ant productivity is far better than the sum of their single activities
- ants are ‘grant masters’ in search and exploitation

Which mechanisms are important?

- cooperation and division of labour
- task adaptation
- local interactions
- pheromones

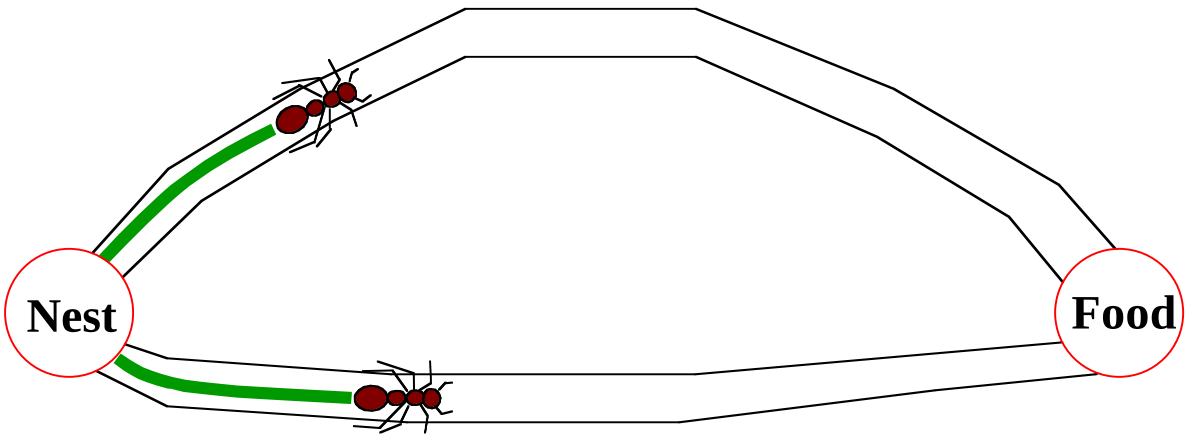
Ant Systems

Cooperative search by pheromone trails



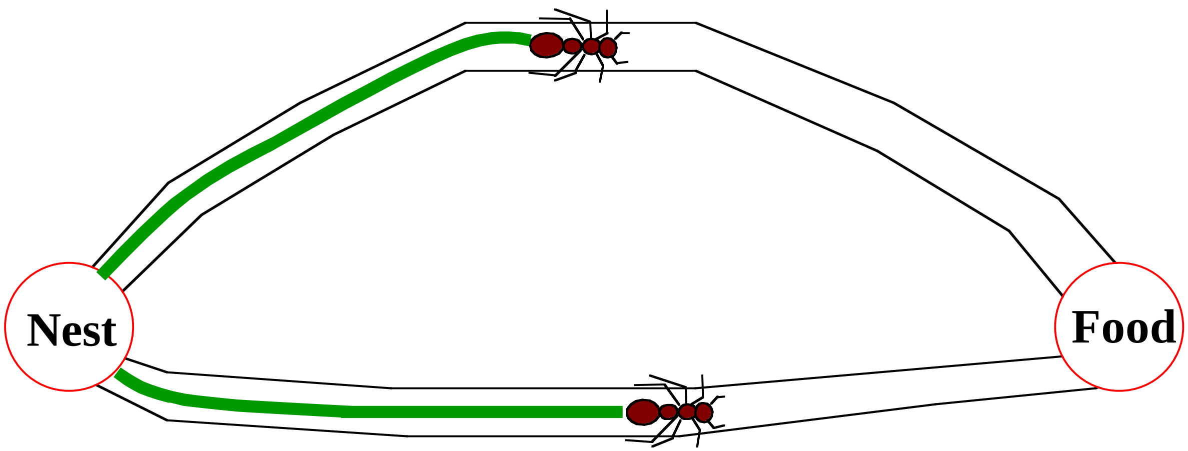
Ant Systems

Cooperative search by pheromone trails



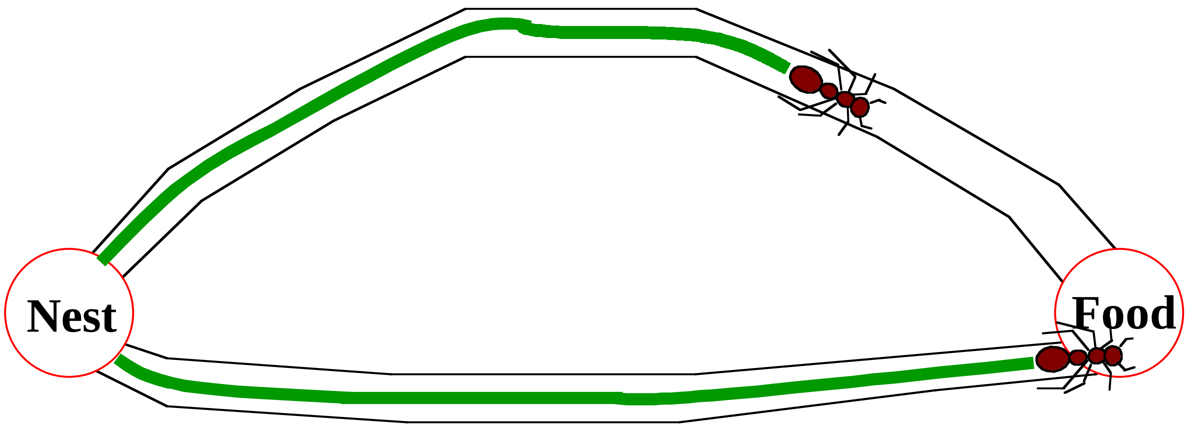
Ant Systems

Cooperative search by pheromone trails



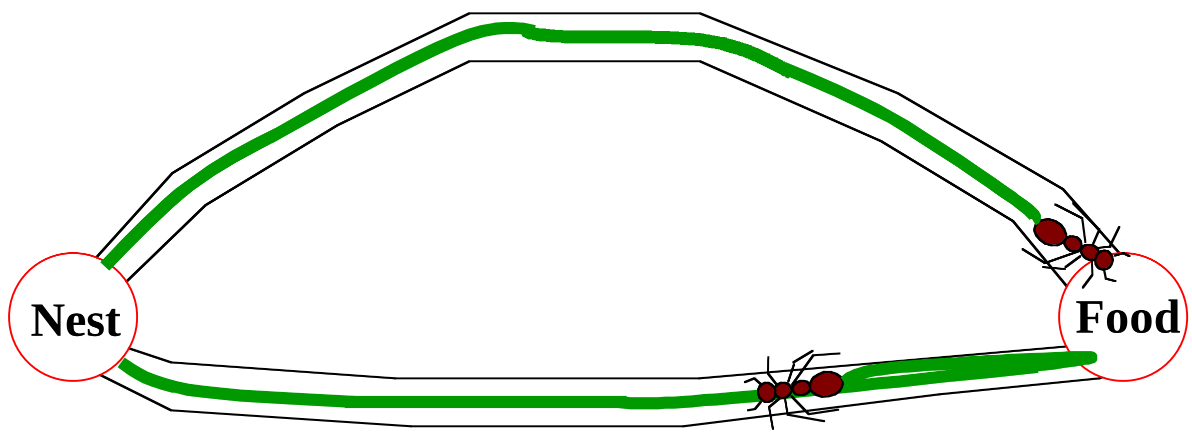
Ant Systems

Cooperative search by pheromone trails



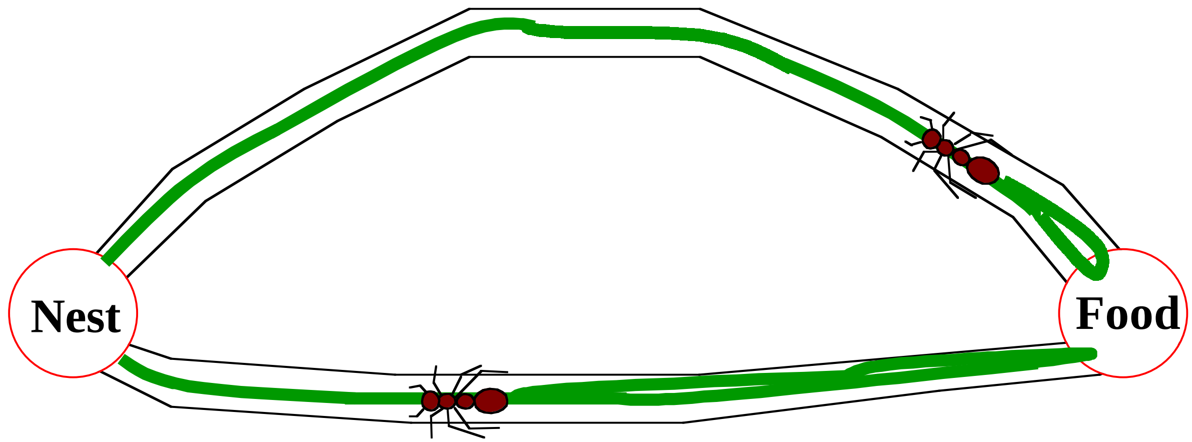
Ant Systems

Cooperative search by pheromone trails



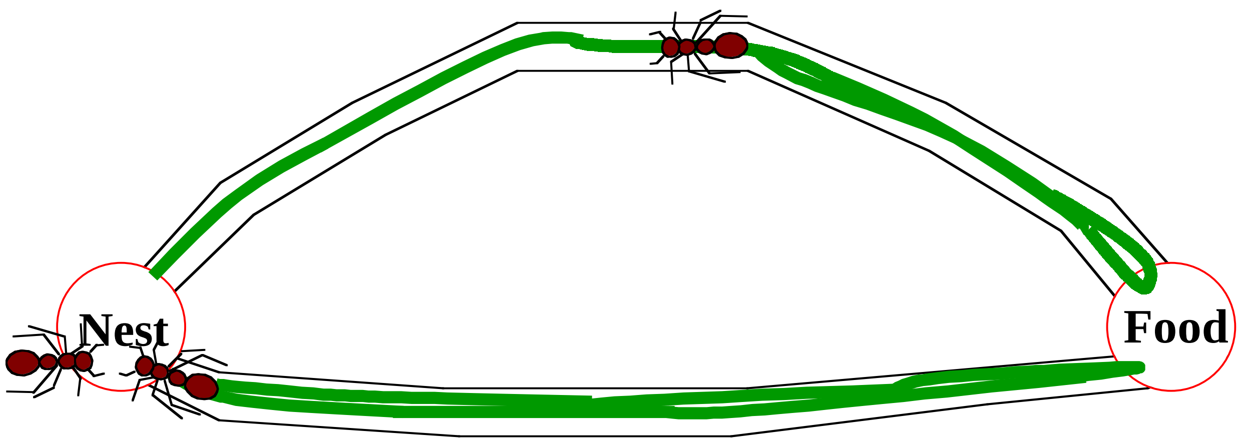
Ant Systems

Cooperative search by pheromone trails



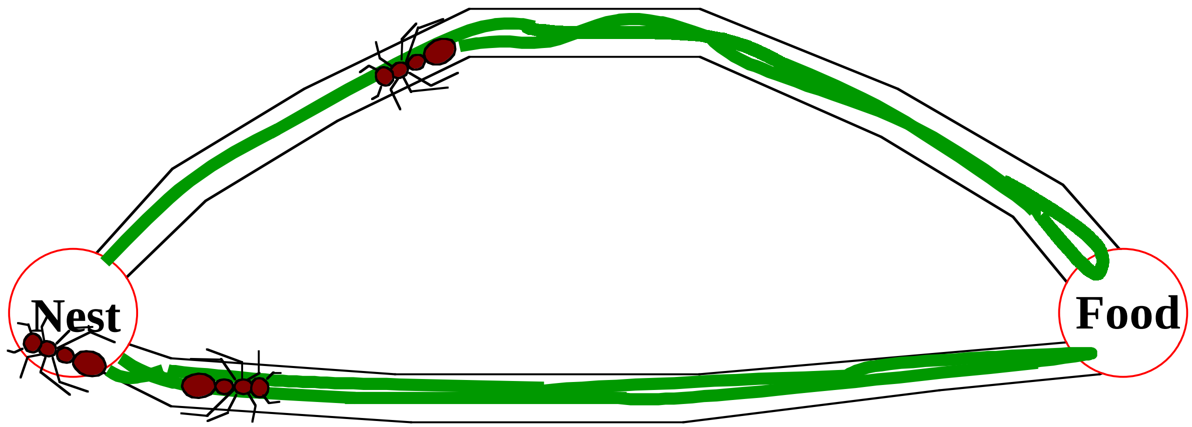
Ant Systems

Cooperative search by pheromone trails



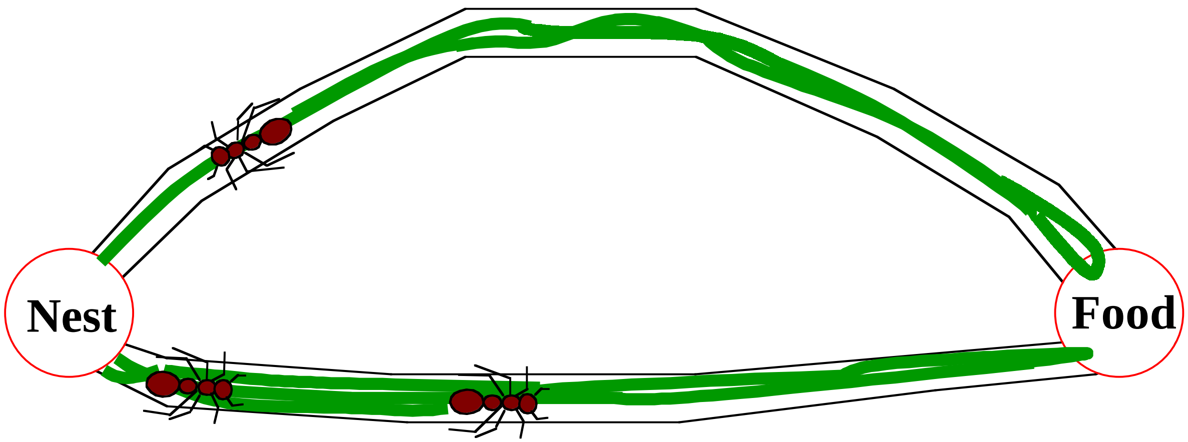
Ant Systems

Cooperative search by pheromone trails



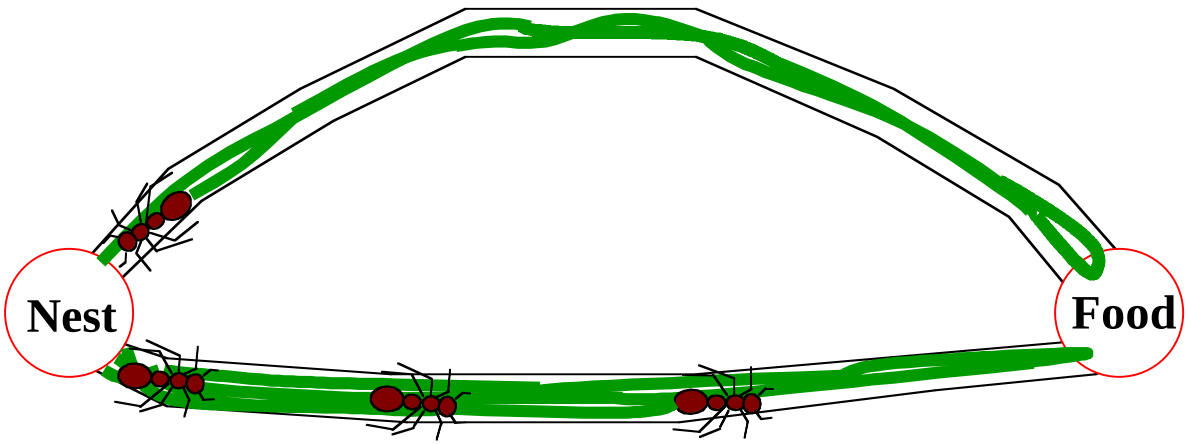
Ant Systems

Cooperative search by pheromone trails



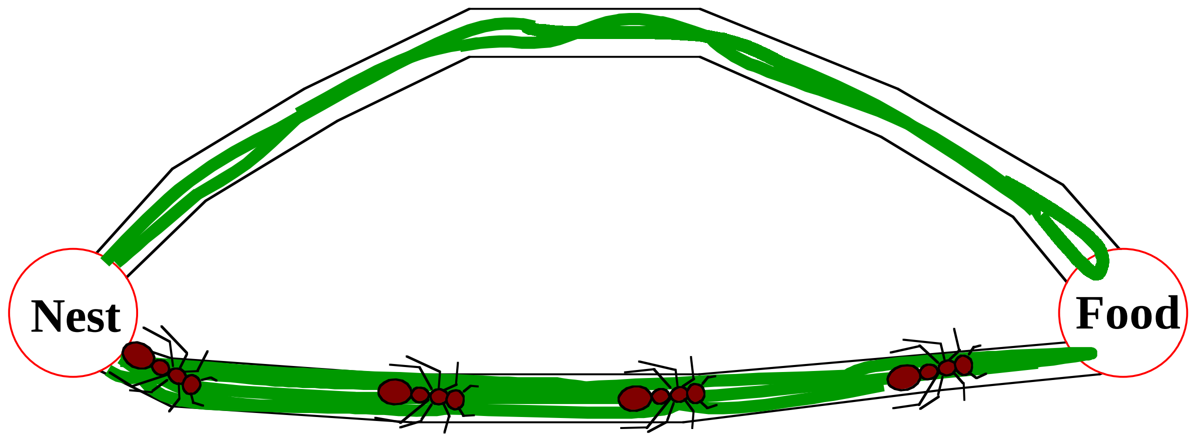
Ant Systems

Cooperative search by pheromone trails



Ant Systems

Cooperative search by pheromone trails



Ant Systems

Tackling the TSP by pheromone trails

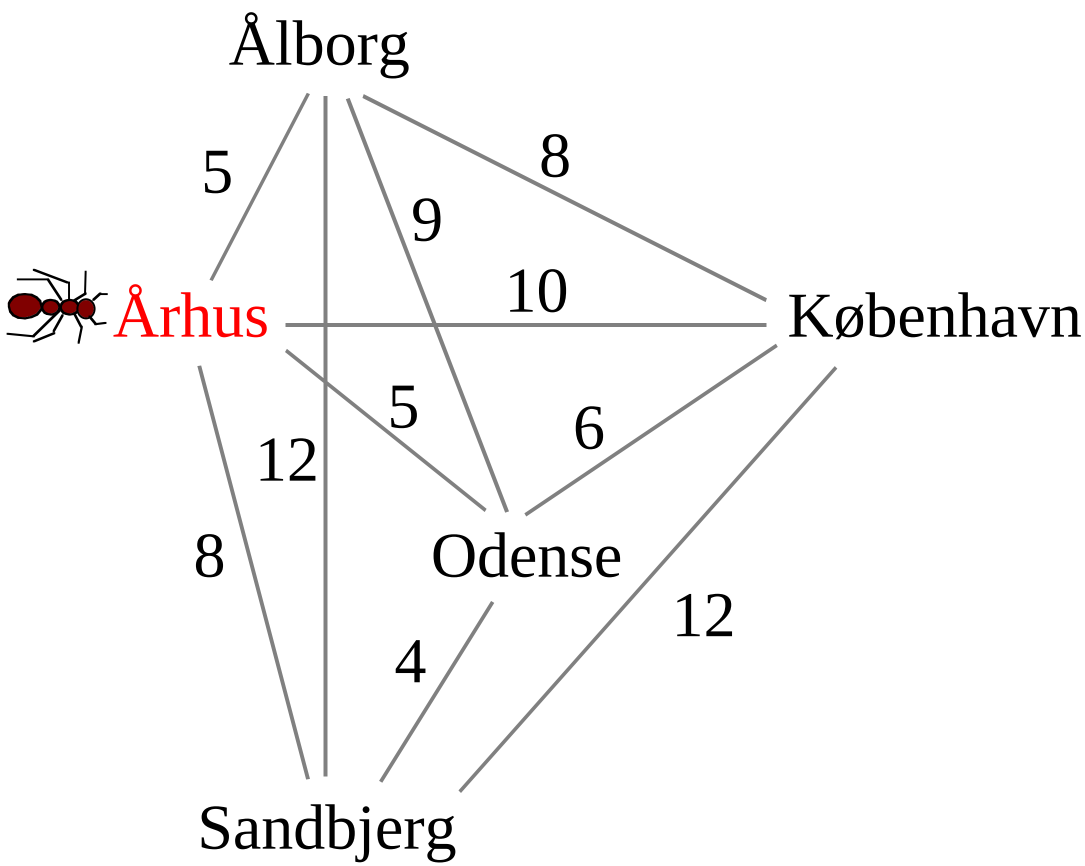
Task: Minimize $COST(i_1,\ldots,i_n) = \sum_{j=1}^{n-1} d(C_{i_j}, C_{i_{j+1}}) + d(C_{i_n}, C_{i_1})$

$d(C_x, C_y)$ distance between cities x and y

‘Find a path *i* between *n* cities (including an edge from the last city to the home city), such that the costs for travelling are minimal and each city is visited once.’

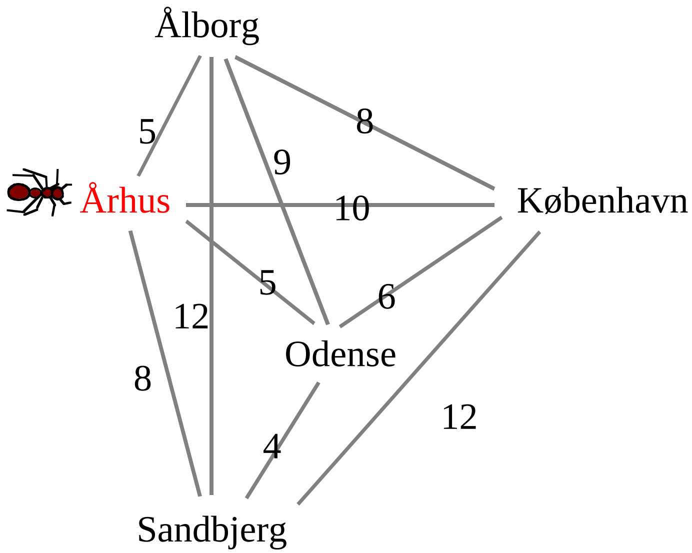
(Dorigo et al, 1996)

Ant Systems



Ant Systems

The algorithm

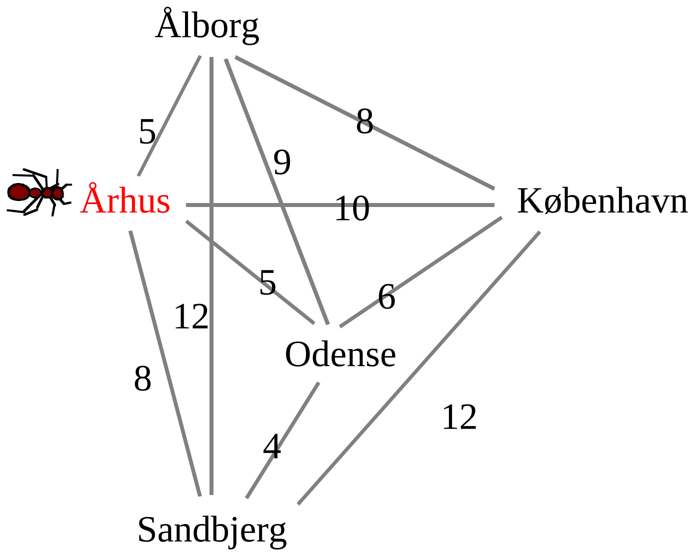


procedure ant system
initialize
for t=1 **to** number of cycles **do**
begin
  for k=1 **to** m **do** {for all ants}
  begin
    repeat {let each ant k run through all cities}
      select city j to be visited next with probablity P_{ij}^{k}
    until ant k has completed a tour
    calculate the length L_k of the tour generated by ant k
  end
  save the best solution found so far
  update the pheromone trail levels τ_{ij} on all paths based on the costs
end

(Dorigo et al, 1996)

Ant Systems

The algorithm



procedure ant system
initialize
for t=1 **to** number of cycles **do**
begin
  for k=1 **to** m **do** {for all ants}
  begin
    repeat {let each ant k run through all cities}
      select city j to be visited next with probablity P_{ij}^{k}
    until ant k has completed a tour
    calculate the length L_k of the tour generated by ant k
  end
  save the best solution found so far
  update the pheromone trail levels τ_{ij} on all paths based on the costs
end

(Dorigo et al, 1996)

Ant Systems

Laying of the pheromone trails

$\tau_{ij}(t+n)$ intensity of the pheromone ‘trail’ connecting the cities (i j) at time t+n

Ant Systems

Laying of the pheromone trails

$\tau_{ij}(t+n)$ intensity of the pheromone 'trail' connecting the cities (i j) at time t+n

$$\tau_{ij}(t+n) = \underbrace{(1-\rho)\tau_{ij}(t)}_{\text{pheromone evaporation}} + \underbrace{\Delta\tau_{ij}(t, t+n)}_{\text{new pheromone}} \qquad 0 < \rho \le 1$$

evaporation factor

Ant Systems

Laying of the pheromone trails

$\tau_{ij}(t+n)$ intensity of the pheromone 'trail' connecting the cities (i j) at time t+n

$$\tau_{ij}(t+n) = \underbrace{(1-\rho)\tau_{ij}(t)}_{\text{pheromone evaporation}} + \boxed{\Delta\tau_{ij}(t,t+n)} \qquad 0 < \rho \le 1$$

evaporation factor

$$\boxed{\Delta\tau_{ij}(t,t+n)} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t,t+n)$$

new pheromone

pheromone quantity added to (i j) by ant k; m=#ants

Ant Systems

Laying of the pheromone trails

$\tau_{ij}(t+n)$ intensity of the pheromone 'trail' connecting the cities (i j) at time t+n

$$\tau_{ij}(t+n) = \underbrace{(1-\rho)\tau_{ij}(t)}_{\text{pheromone evaporation}} + \boxed{\Delta\tau_{ij}(t,t+n)} \qquad 0 < \rho \le 1$$

evaporation factor

$$\boxed{\Delta\tau_{ij}(t,t+n)} = \sum_{k=1}^{m} \boxed{\Delta\tau_{ij}^{k}(t,t+n)}$$

new pheromone

pheromone quantity added to (i j) by ant k; m=#ants

constant

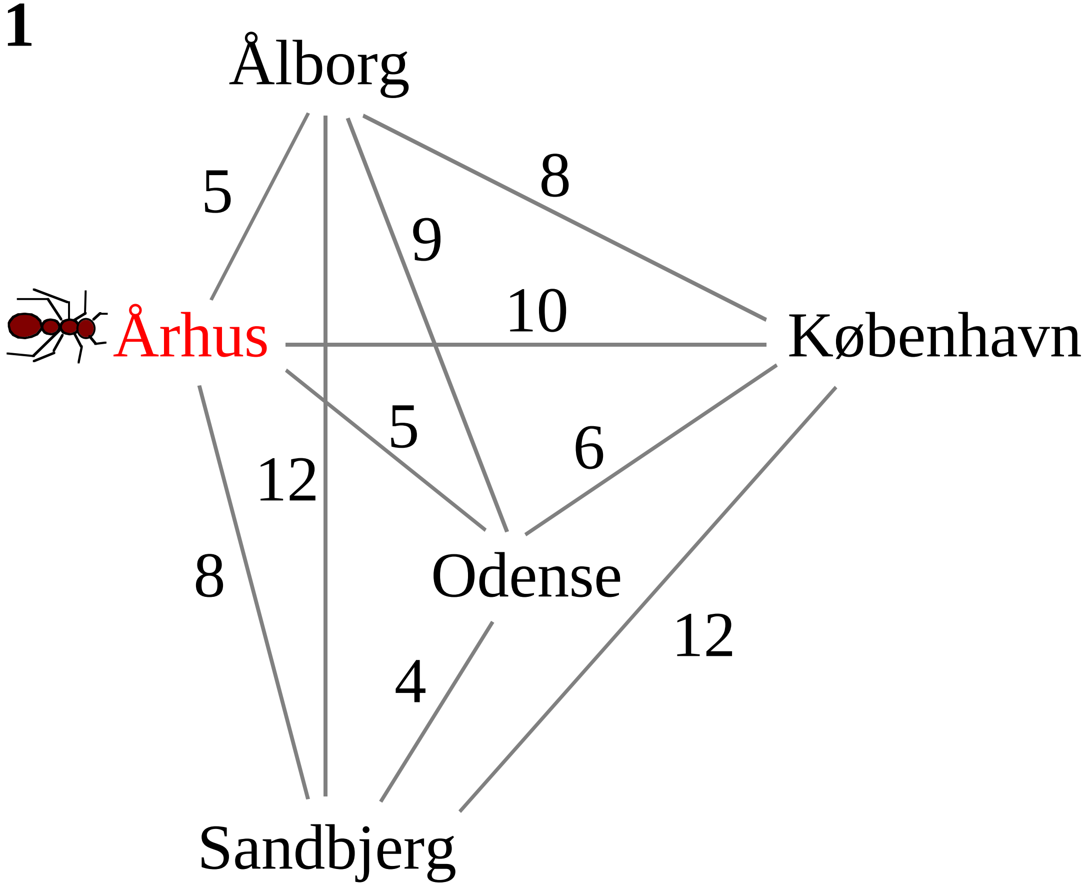
$$\boxed{\Delta\tau_{ij}^{k}(t,t+n)} = \begin{cases} \dfrac{Q}{L_k} & \text{if k-th ant uses edge (i j) in its tour} \\ 0 & \text{otherwise} \end{cases}$$

tour length found by k-th ant

Ant Systems

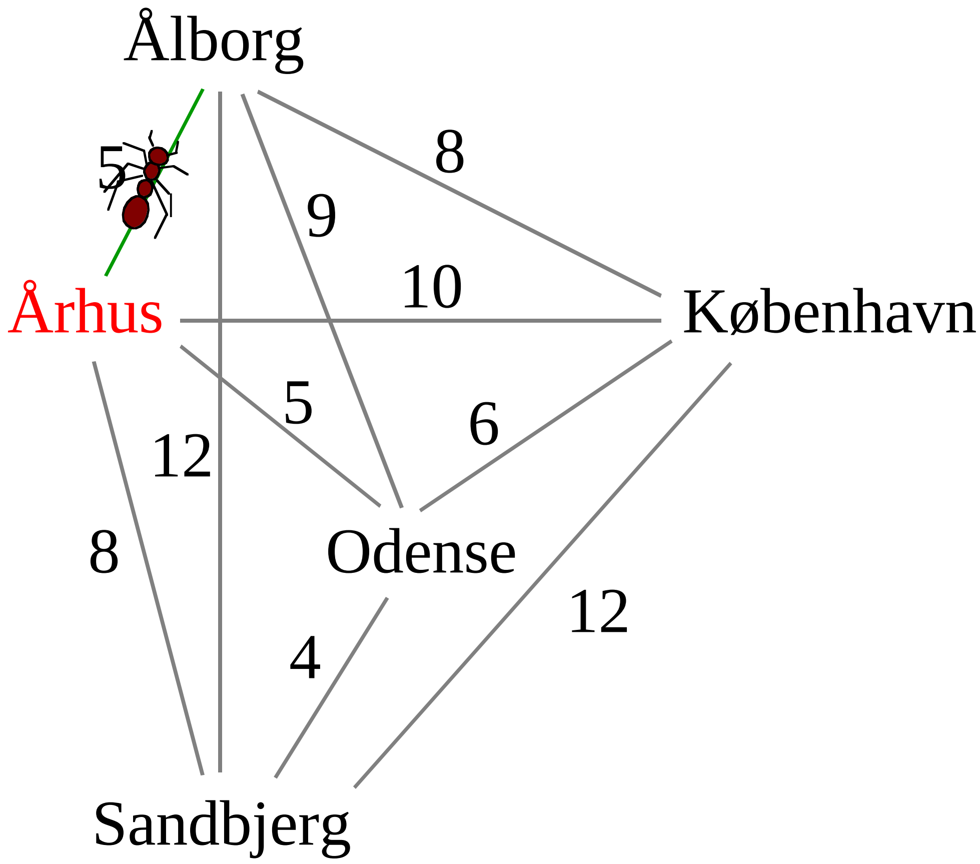
time: t=1

ant: k=1



Ant Systems

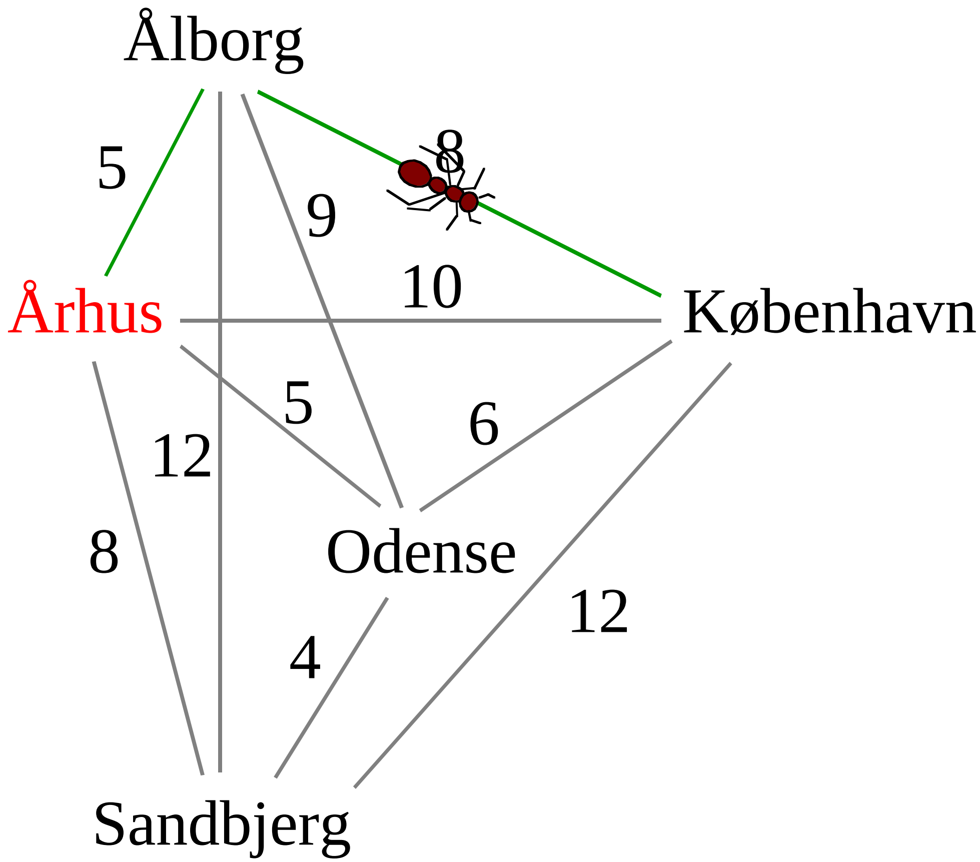
time: t=1
ant: k=1



Ant Systems

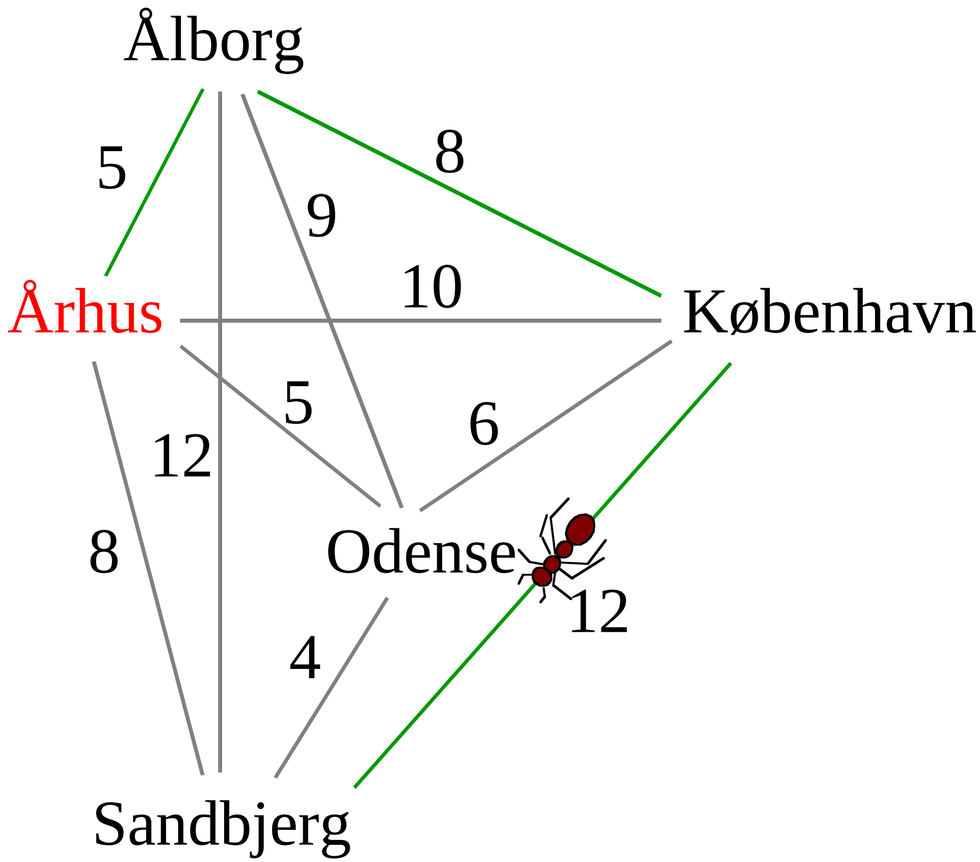
time: t=1

ant: k=1



Ant Systems

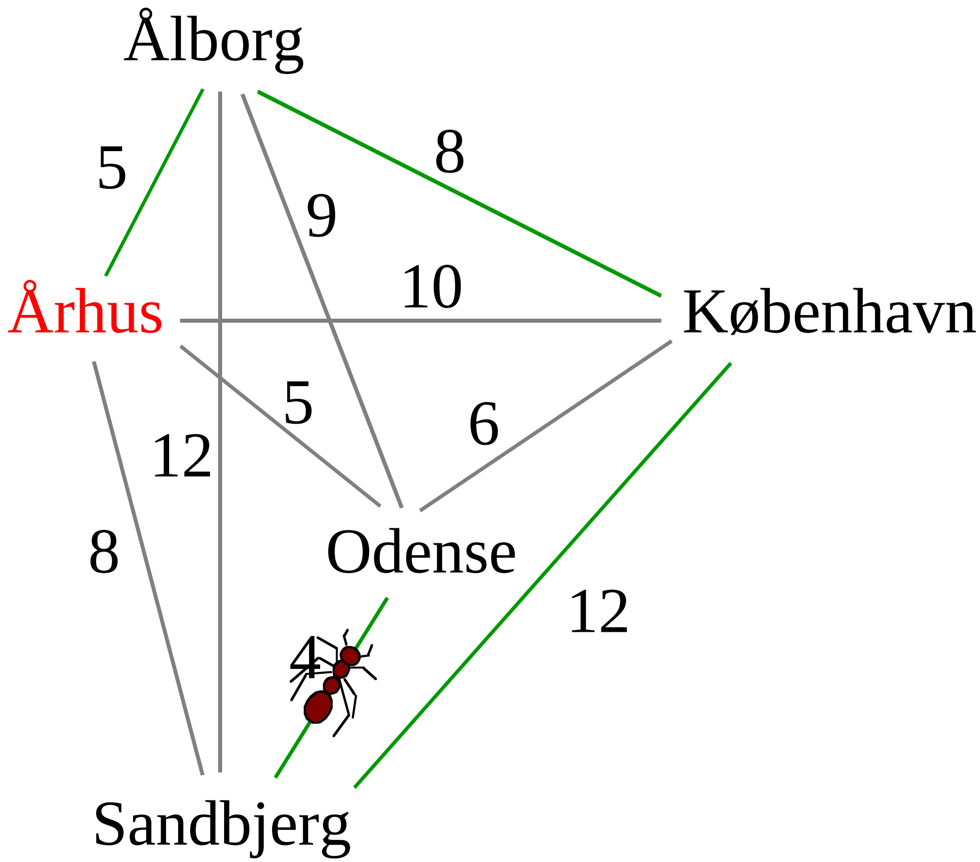
time: t=1
ant: k=1



Ant Systems

time: t=1

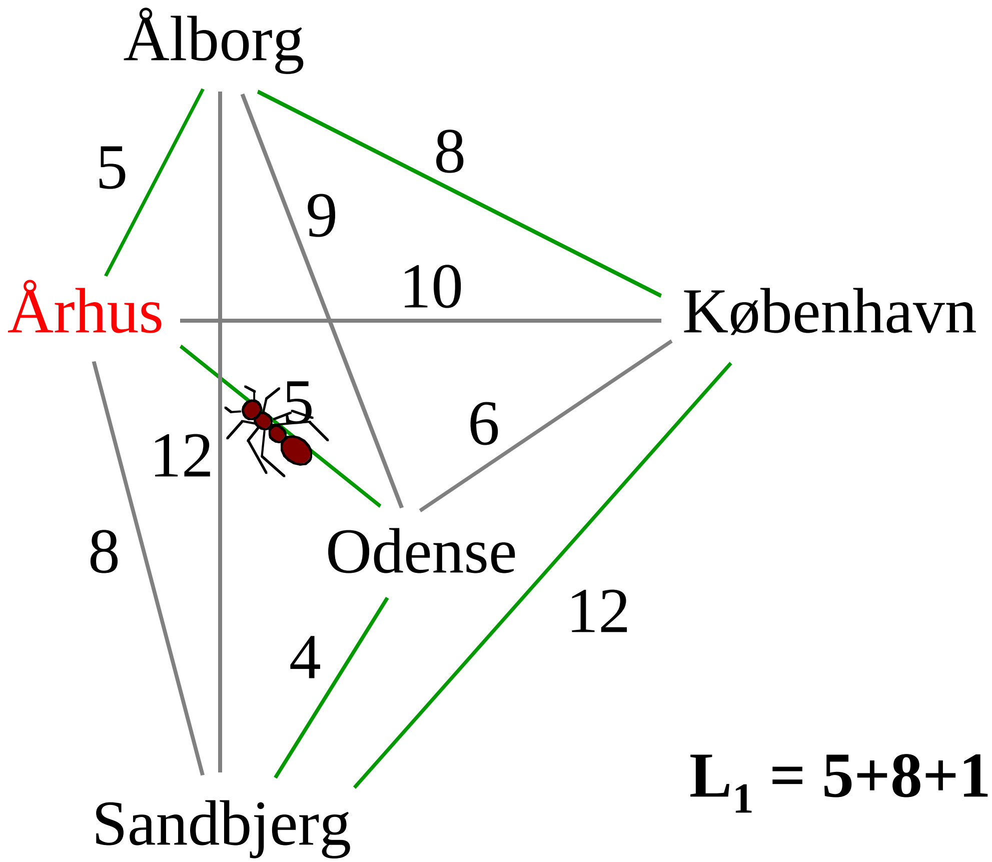
ant: k=1



Ant Systems

time: t=1

ant: k=1

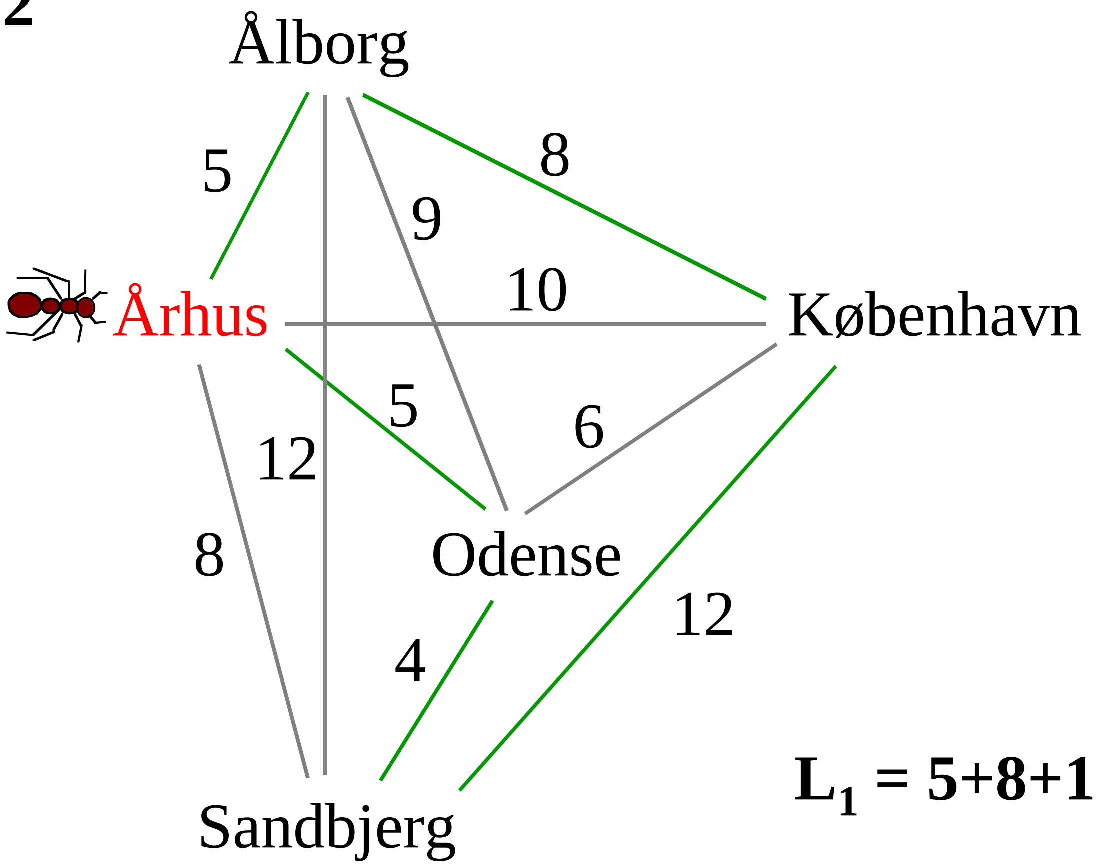


$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$

Ant Systems

time: t=1

ant: k=2

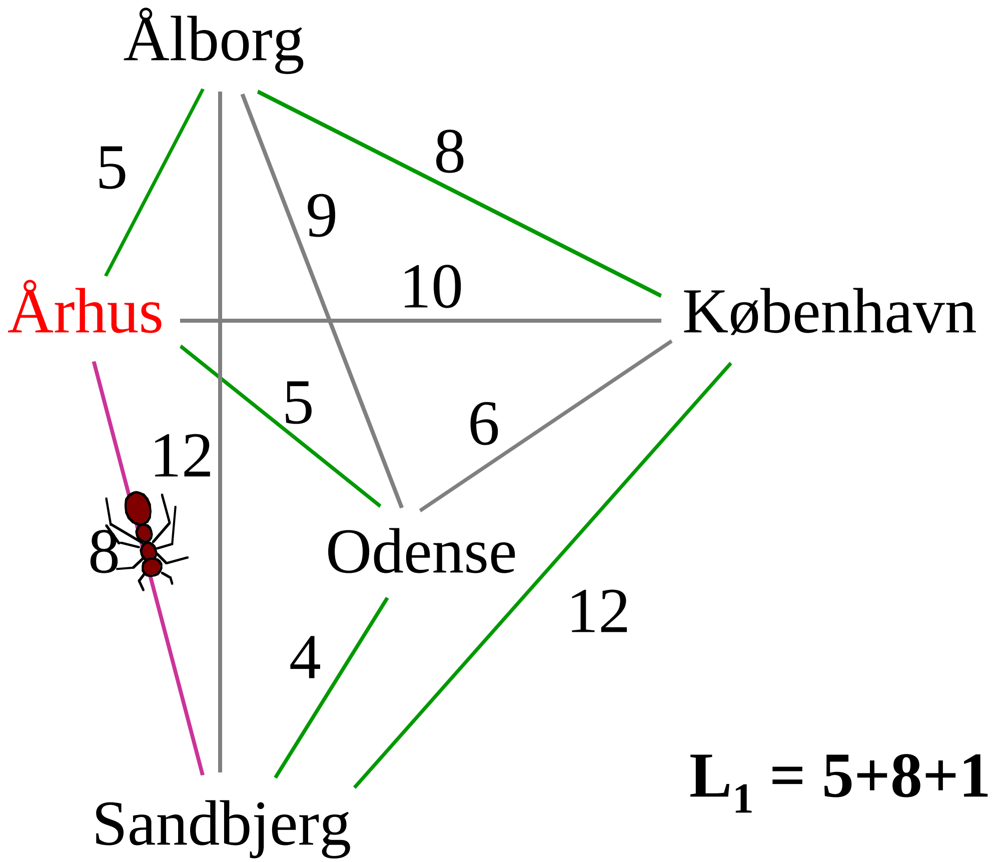


$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$

Ant Systems

time: t=1

ant: k=2

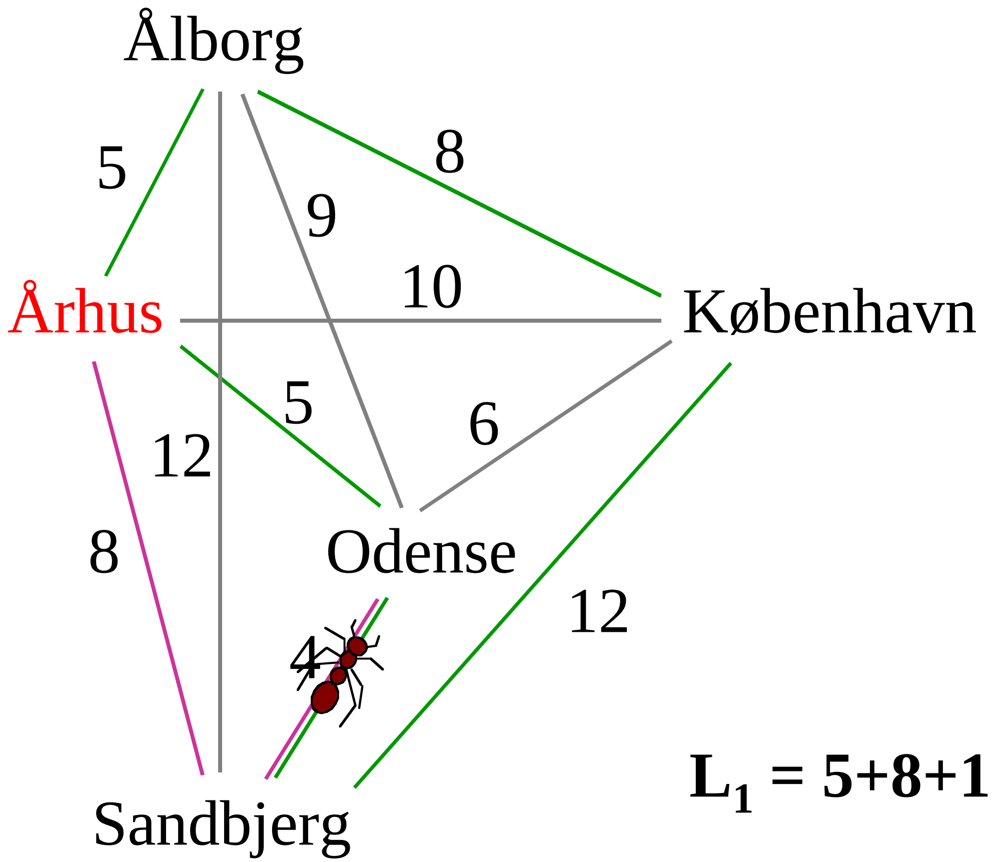


$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$

Ant Systems

time: t=1

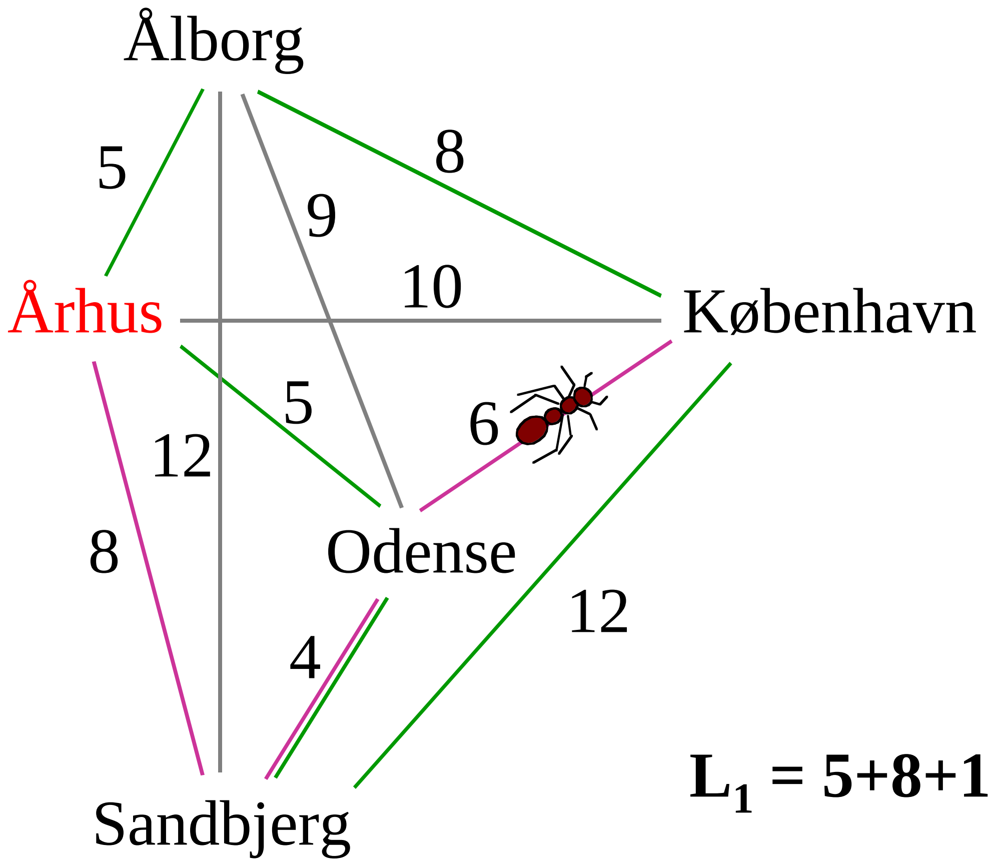
ant: k=2



$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$

Ant Systems

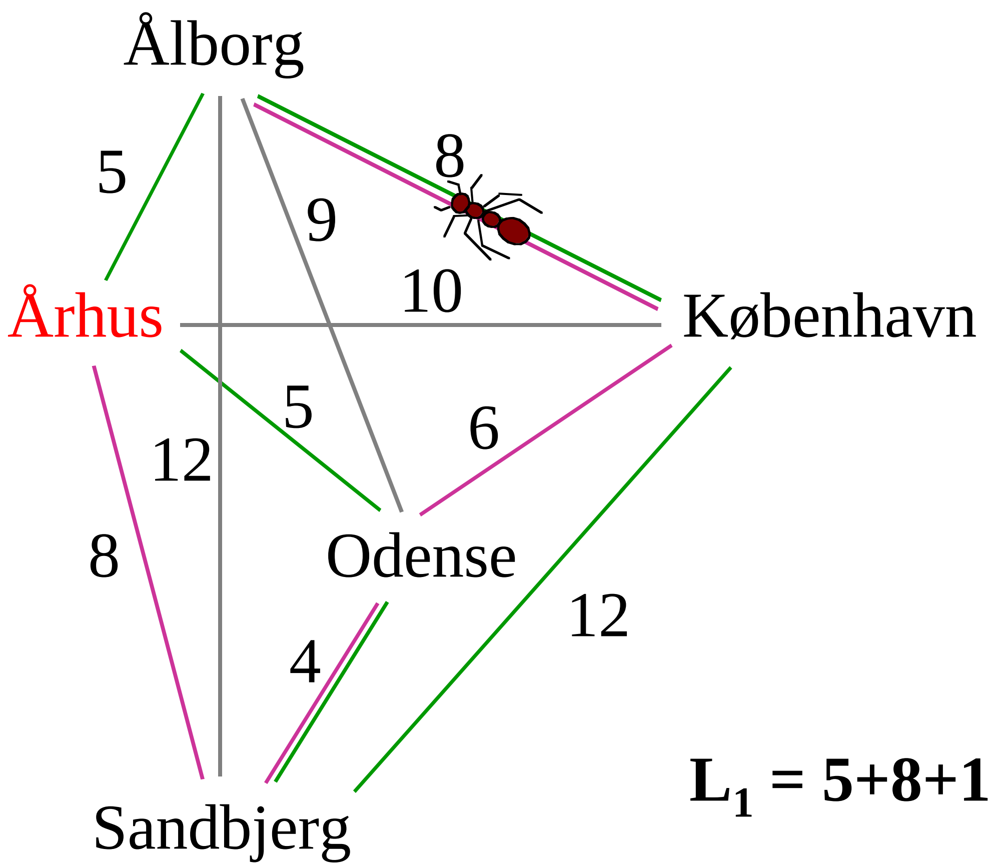
time: t=1
ant: k=2



$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$

Ant Systems

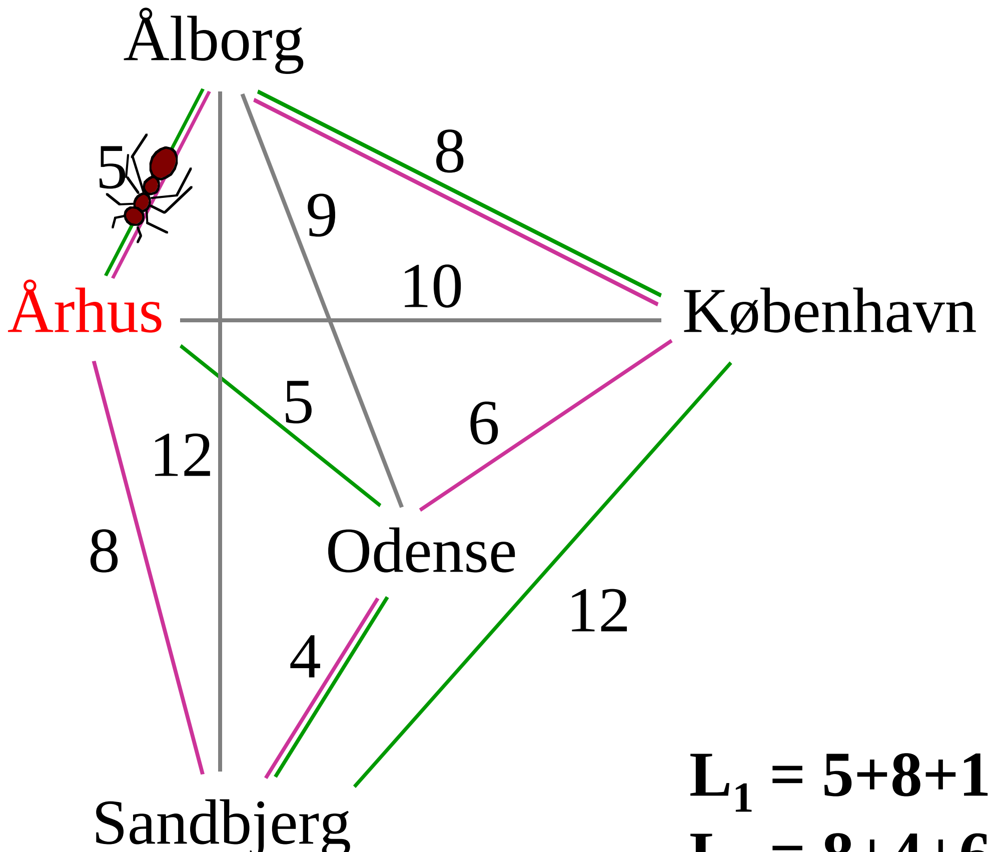
time: t=1
ant: k=2



$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$

Ant Systems

time: t=1
ant: k=2

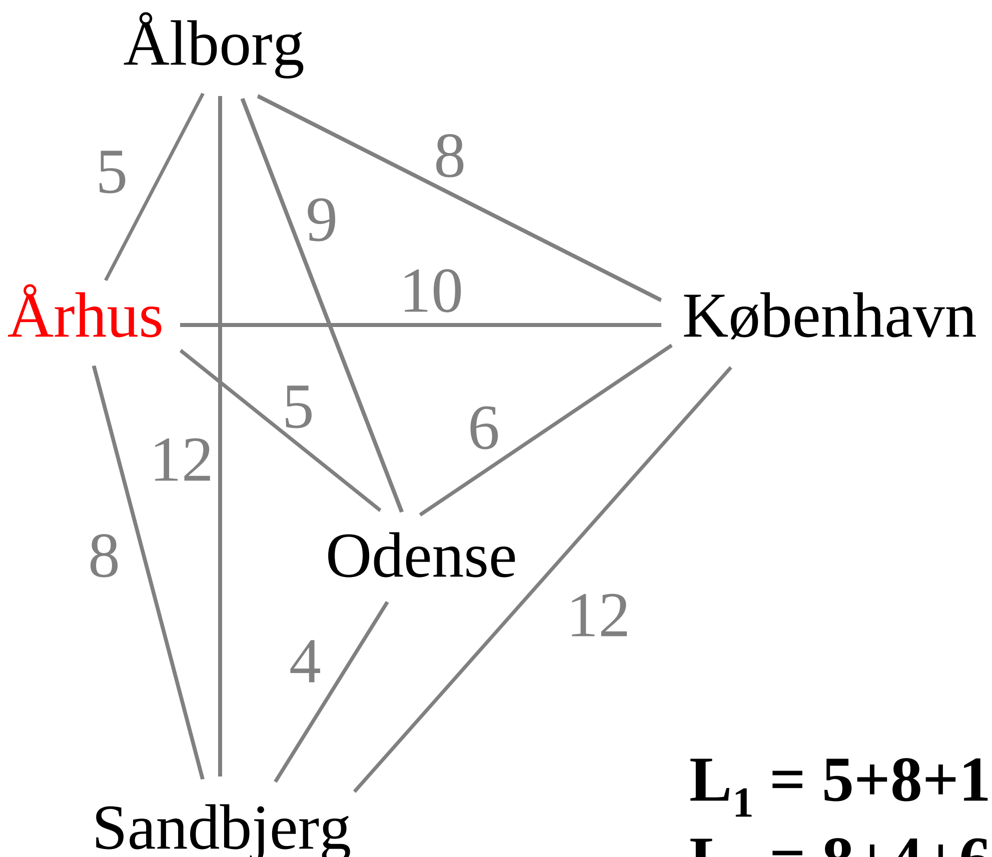


$$\mathbf{L_1 = 5+8+12+4+5 = 34}$$
$$\mathbf{L_2 = 8+4+6+8+5 = 31}$$

Ant Systems

$$\tau_{Århus,Ålborg}(1) = (1-\rho)\tau_{Århus,Ålborg}(0) + \Delta\tau_{Århus,Ålborg}(1)$$

time: t=1
ant: k=2



L_1 = 5+8+12+4+5 = 34
L_2 = 8+4+6+8+5 = 31

Ant Systems

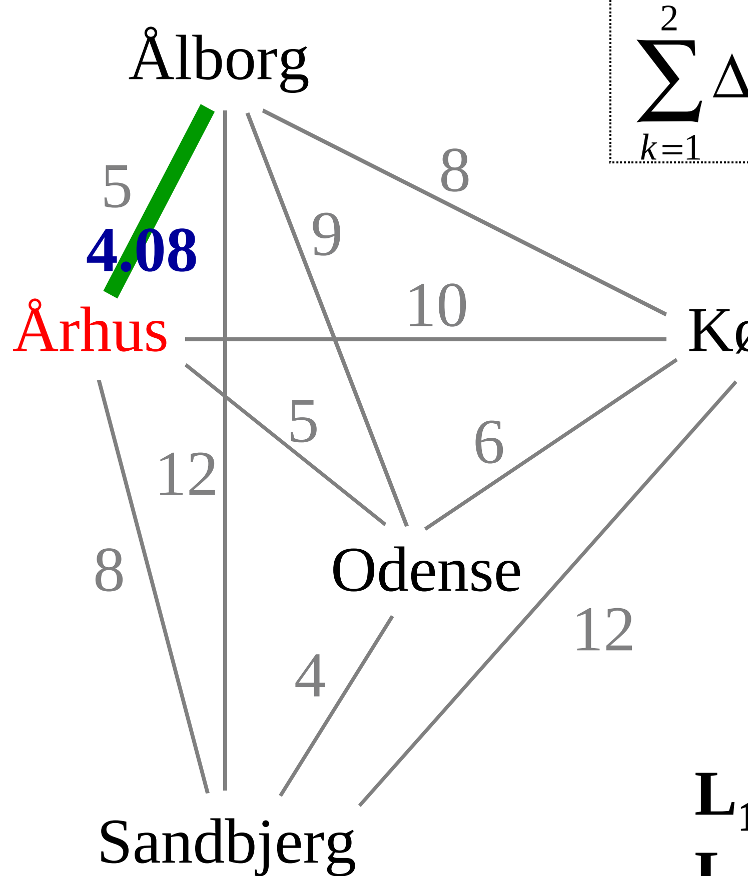
$$\tau_{\text{Århus},\text{Ålborg}}(1) = \overbrace{(1-\rho)\tau_{\text{Århus},\text{Ålborg}}(0)}^{1.0} + \underbrace{\Delta\tau_{\text{Århus},\text{Ålborg}}(1)}$$

time: t=1

ant: k=2

$$\boxed{\sum_{k=1}^{2}\Delta\tau_{ij}^{k}(1) = \frac{50}{34} + \frac{50}{31} \approx 3.08}$$

Q (arbitrary)



L_1 = 5+8+12+4+5 = 34

L_2 = 8+4+6+8+5 = 31

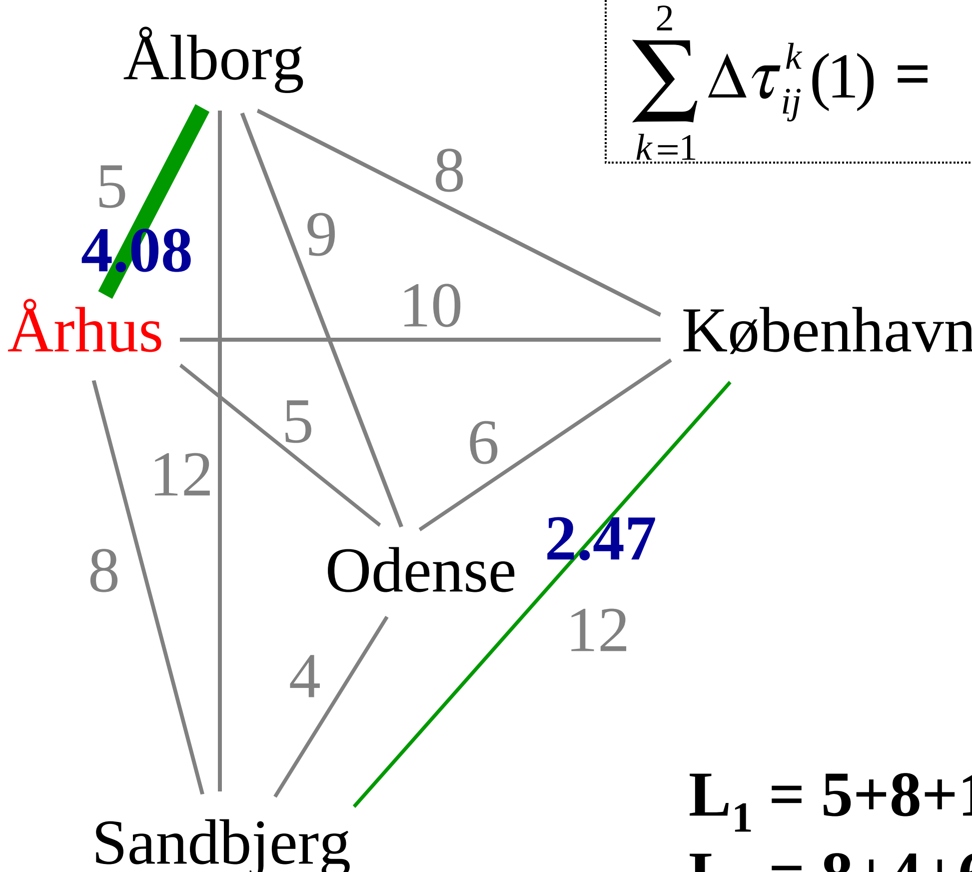
Ant Systems

$$\tau_{K\o bhv,Sandbjg}(1) = \overbrace{(1-\rho)\tau_{K\o bhv,Sandbjg}(0)}^{1.0} + \underbrace{\Delta\tau_{K\o bhv,Sandbjg}(1)}$$

time: t=1

ant: k=2

$$\boxed{\sum_{k=1}^{2}\Delta\tau_{ij}^{k}(1) = \frac{50}{34} + 0 \approx 1.47}$$



L_1 = 5+8+12+4+5 = 34

L_2 = 8+4+6+8+5 = 31

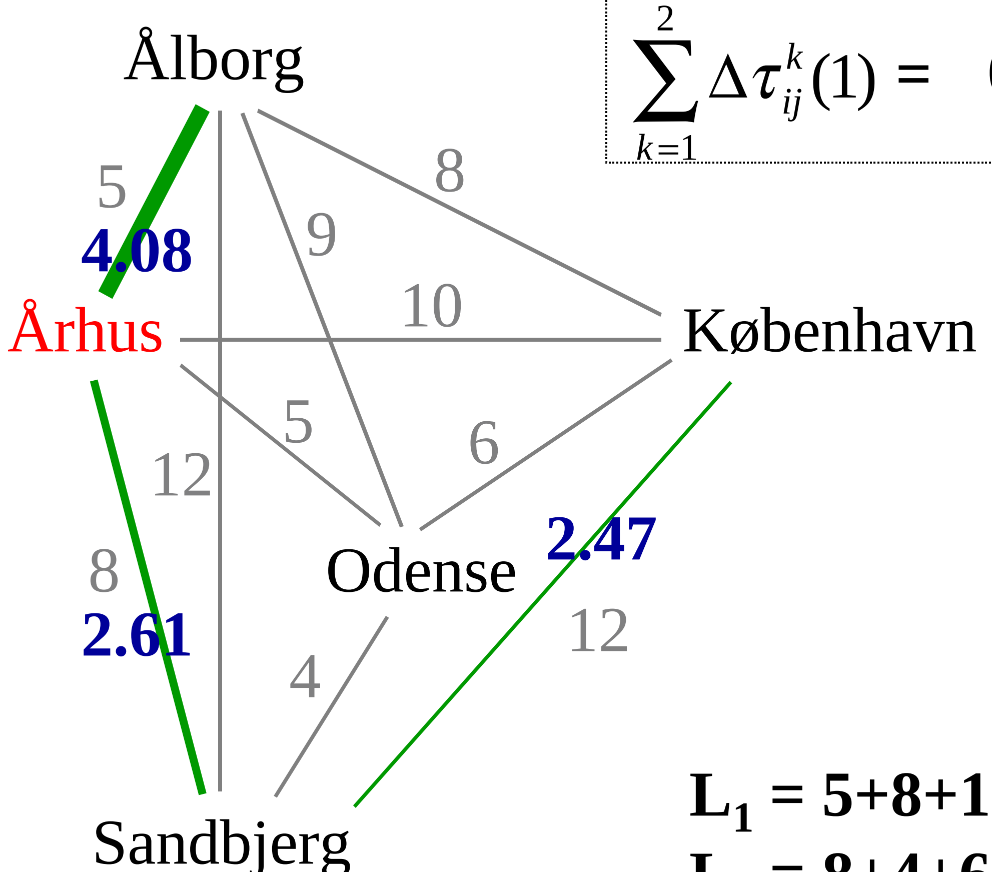
Ant Systems

$$\tau_{Århus,Sandbjg}(1) = \overbrace{(1-\rho)\tau_{Århus,Sandbjg}(0)}^{1.0} + \underbrace{\Delta\tau_{Århus,Sandbjg}(1)}$$

time: t=1

ant: k=2

$$\boxed{\sum_{k=1}^{2} \Delta\tau_{ij}^{k}(1) = 0 + \frac{50}{31} \approx 1.61}$$

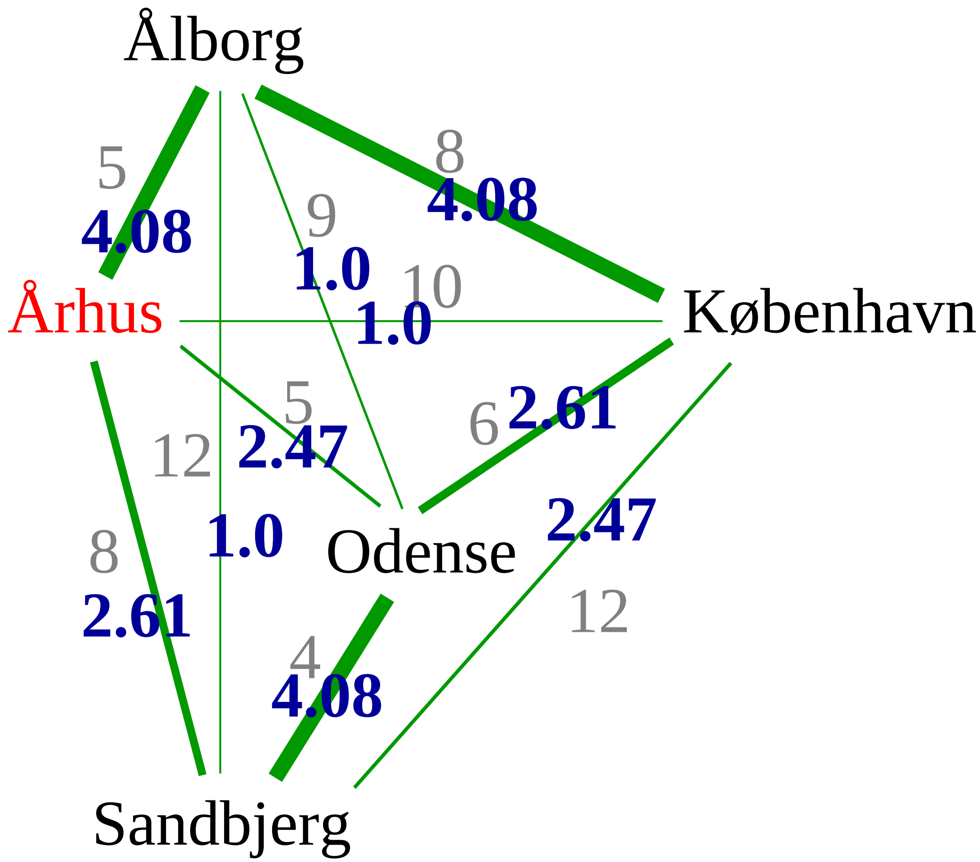


L_1 = 5+8+12+4+5 = 34

L_2 = 8+4+6+8+5 = 31

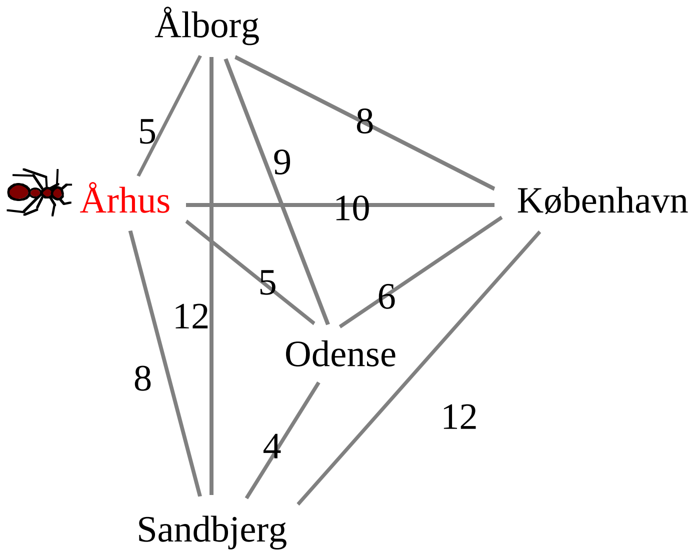
Ant Systems

time: t=1
ant: k=2



Ant Systems

The algorithm



procedure ant system
initialize
for t=1 **to** number of cycles **do**
begin
  for k=1 **to** m **do** {for all ants}
  begin
    repeat {let each ant k run through all cities}
      select city j to be visited next with probablity P_{ij}^{k}
    until ant k has completed a tour
    calculate the length L_k of the tour generated by ant k
  end
  save the best solution found so far
  update the pheromone trail levels τ_{ij} on all paths based on the costs
end

Ant Systems

How does an ‘ant’ know which city it should visit?

probability that ant k in city i visits city j

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{h \in allowed_k(t)} \left[\tau_{ih}(t)\right]^{\alpha}\left[\eta_{ih}\right]^{\beta}} & \text{if } j \in allowed_k(t) \\ 0 & \text{otherwise} \end{cases}$$

Ant Systems

How does an ‘ant’ know which city it should visit?

pheromone ‘trail’

$$\boxed{\eta_{ij}} = 1/d(C_i, C_j)$$ ‘visibility’

probability that ant k in city i visits city j

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

Ant Systems

How does an 'ant' know which city it should visit?

pheromone 'trail'

$\boxed{\eta_{ij}} = 1/d(C_i, C_j)$ 'visibility'

probability that ant k in city i visits city j

parameters to control the relative importance of the trail vs. visibility

$$p_{ij}^k(t) = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{h \in allowed_k(t)} \left[\tau_{ih}(t)\right]^{\alpha}\left[\eta_{ih}\right]^{\beta}} & \text{if } j \in allowed_k(t) \\ 0 & \text{otherwise} \end{cases}$$

Ant Systems

How does an ‘ant’ know which city it should visit?

pheromone ‘trail’

$\boxed{\eta_{ij}} = 1/d(C_i, C_j)$ ‘visibility’

probability that ant k in city i visits city j

parameters to control the relative importance of the trail vs. visibility

$$p_{ij}^k(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{h \in allowed_k(t)} [\tau_{ih}(t)]^{\alpha}[\eta_{ih}]^{\beta}} & \text{if } j \in allowed_k(t) \\ 0 & \text{otherwise} \end{cases}$$

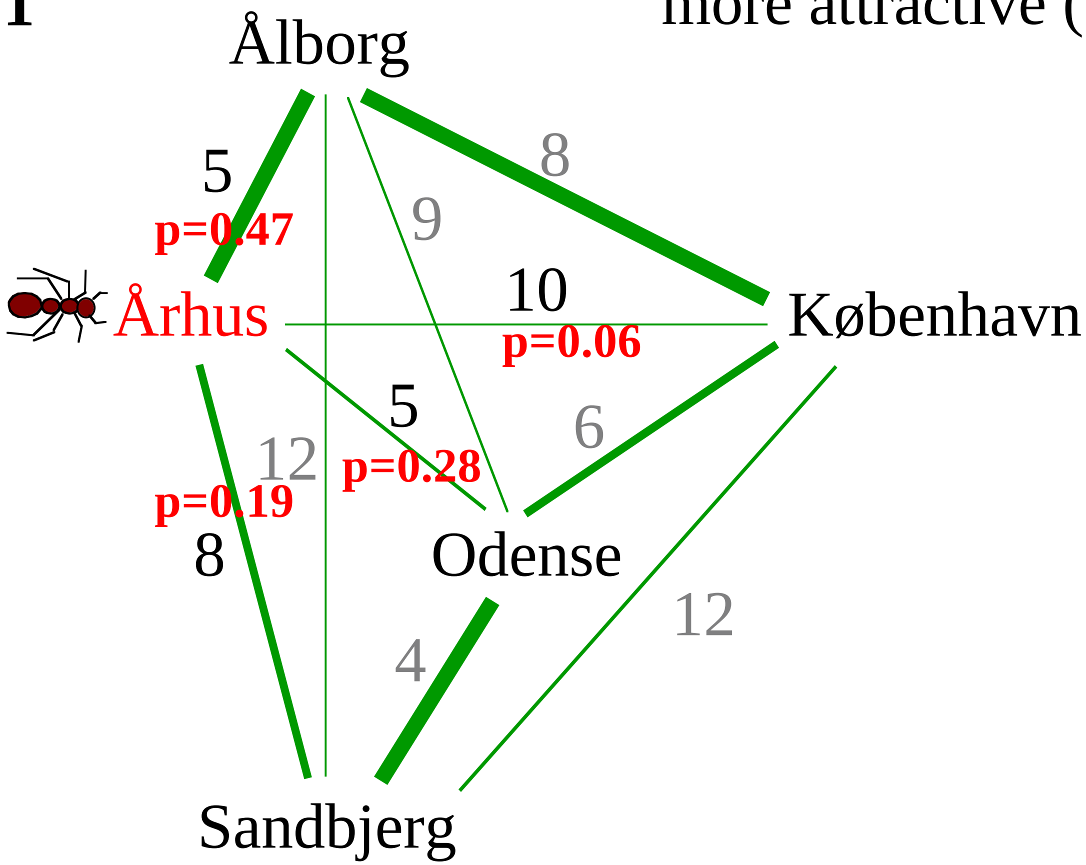
$$\boxed{allowed_k(t)} = \{ j \mid j \notin \underbrace{tabu_k(t)} \}$$

set of visited cities up to time t by the k-th ant

Ant Systems

time: t=2
ant: k=1

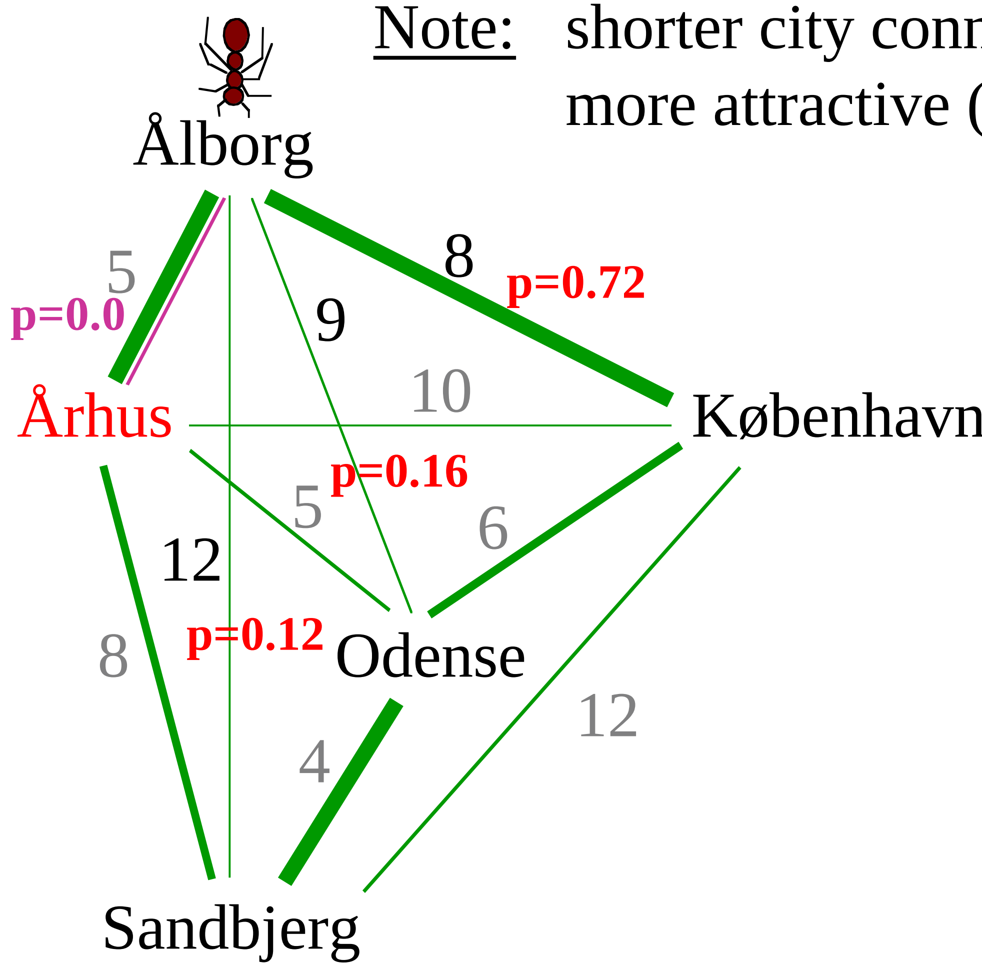
Note: shorter city connections are more attractive ('visibility')!



Ant Systems

time: t=2
ant: k=1

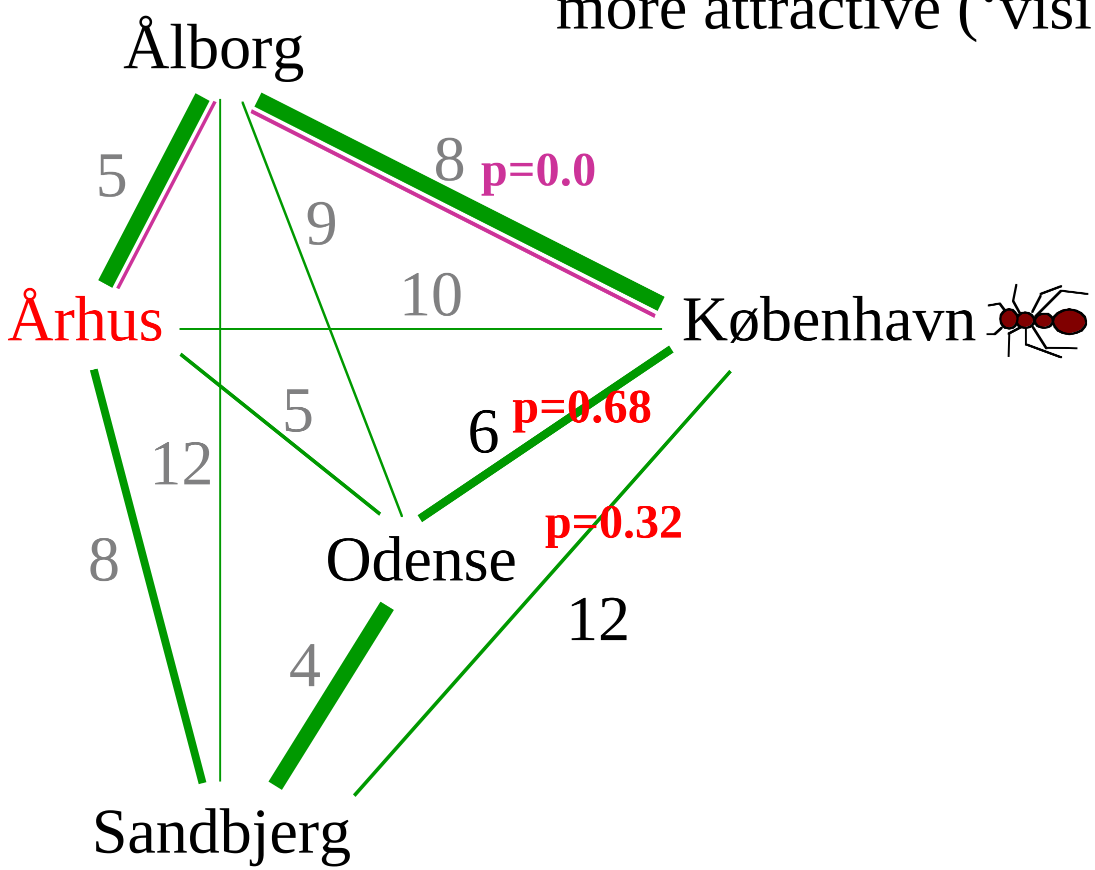
<u>Note:</u> shorter city connections are more attractive ('visibility')!



Ant Systems

time: **t=2**
ant: **k=1**

Note: shorter city connections are more attractive ('visibility')!

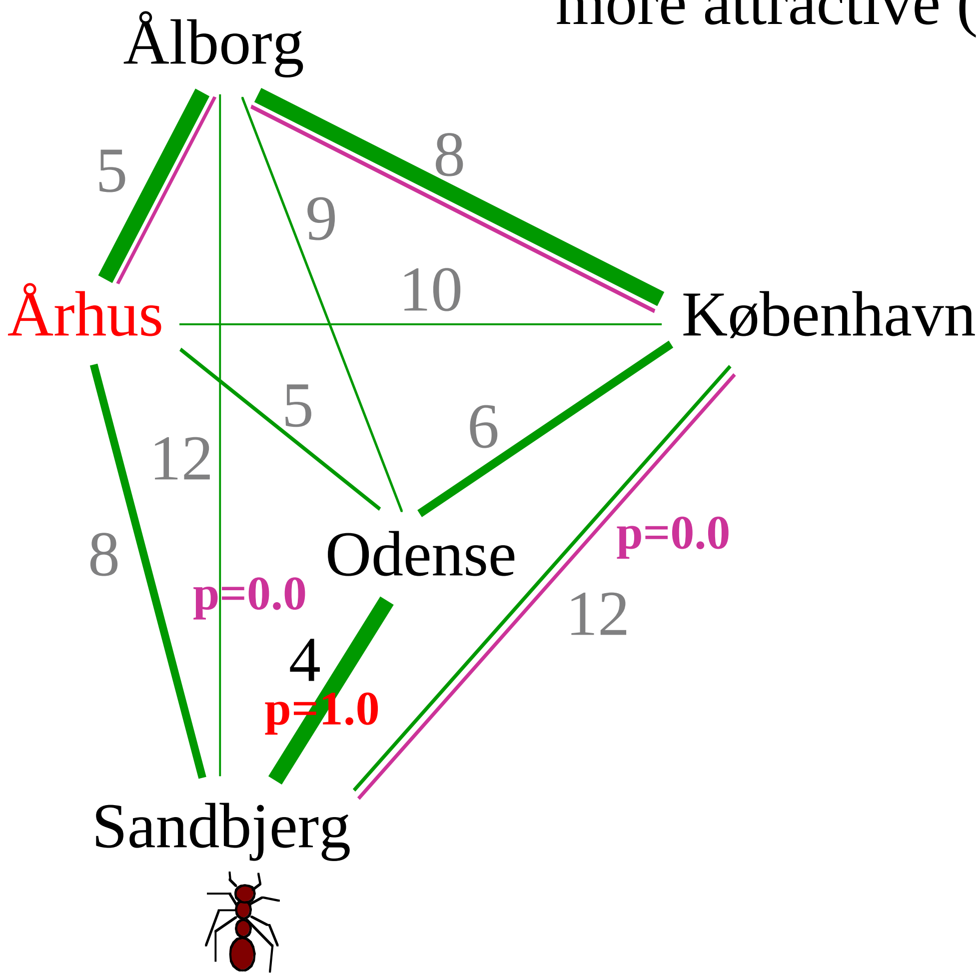


Ant Systems

time: t=2

ant: k=1

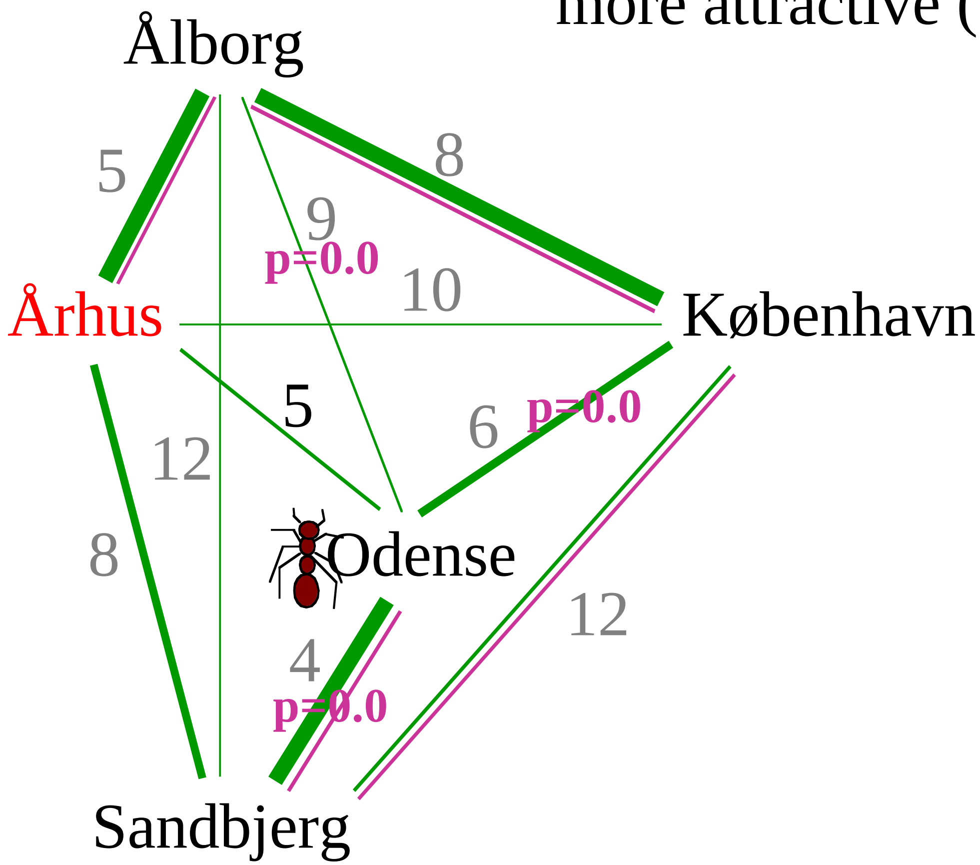
<u>Note:</u> shorter city connections are more attractive ('visibility')!



Ant Systems

time: t=2
ant: k=1

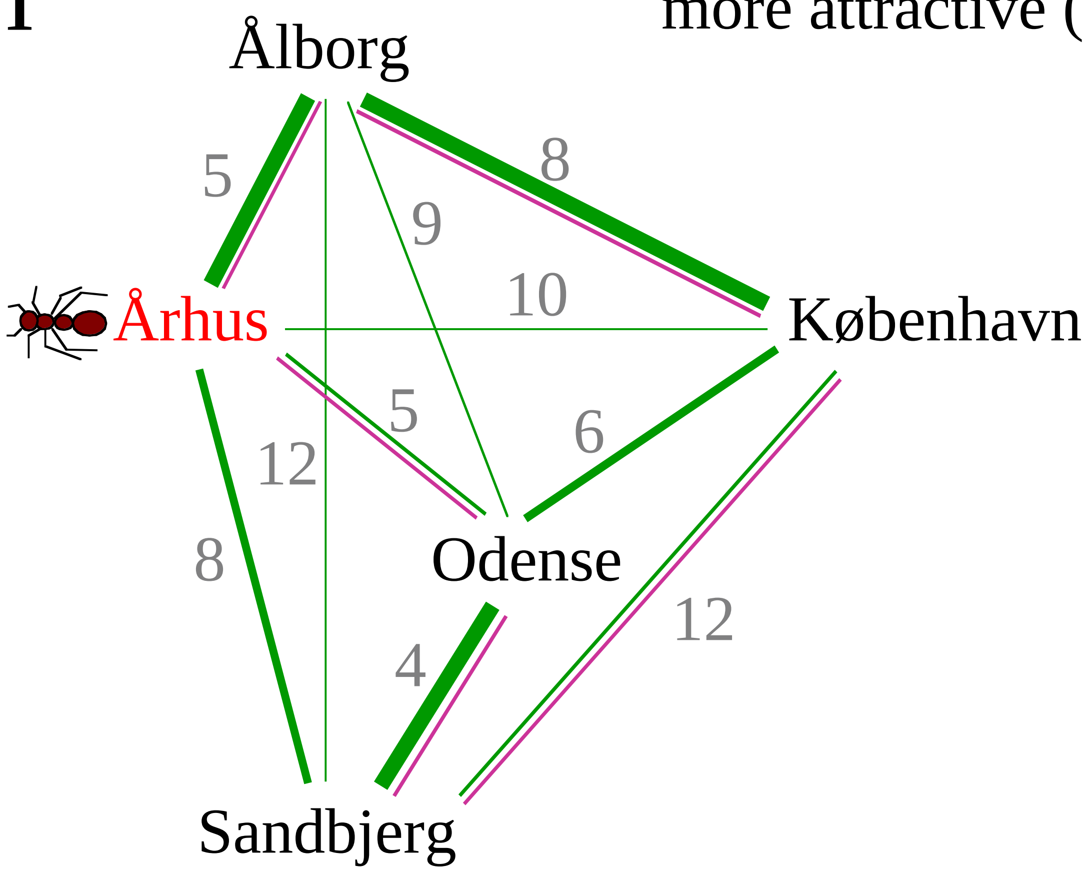
Note: shorter city connections are more attractive ('visibility')!



Ant Systems

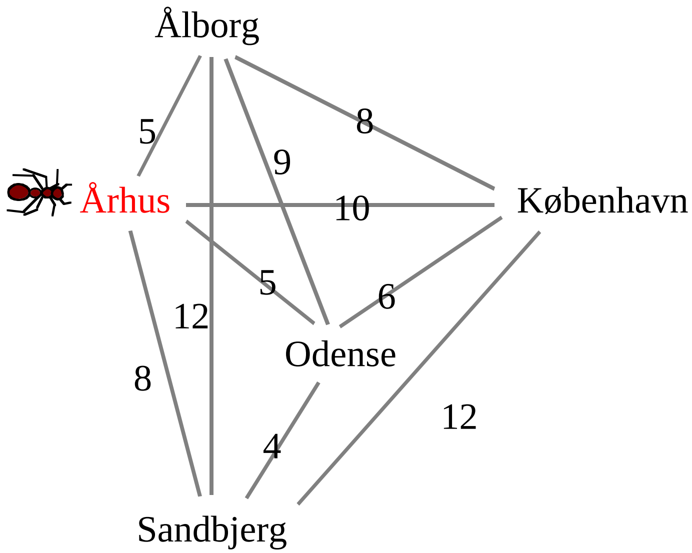
time: t=2
ant: k=1

Note: shorter city connections are more attractive ('visibility')!



Ant Systems

The algorithm



procedure ant system
initialize
for t=1 **to** number of cycles **do**
begin
  for k=1 **to** m **do** {for all ants}
  begin
    repeat {let each ant k run through all cities}
      select city j to be visited next with probablity P_{ij}^{k}
    until ant k has completed a tour
    calculate the length L_k of the tour generated by ant k
  end
  save the best solution found so far
  update the pheromone trail levels τ_{ij} on all paths based on the costs
end

Ant Systems

What are the applications?

- Combinatorial optimization
- Large-scale network routing

Ant Systems

What are the applications?

- Combinatorial optimization
- Large-scale network routing

Advantages / Disadvantages

- Pro: Can adapt to a dynamically changing problem!
- Contra: No guarantees! Solution approximation

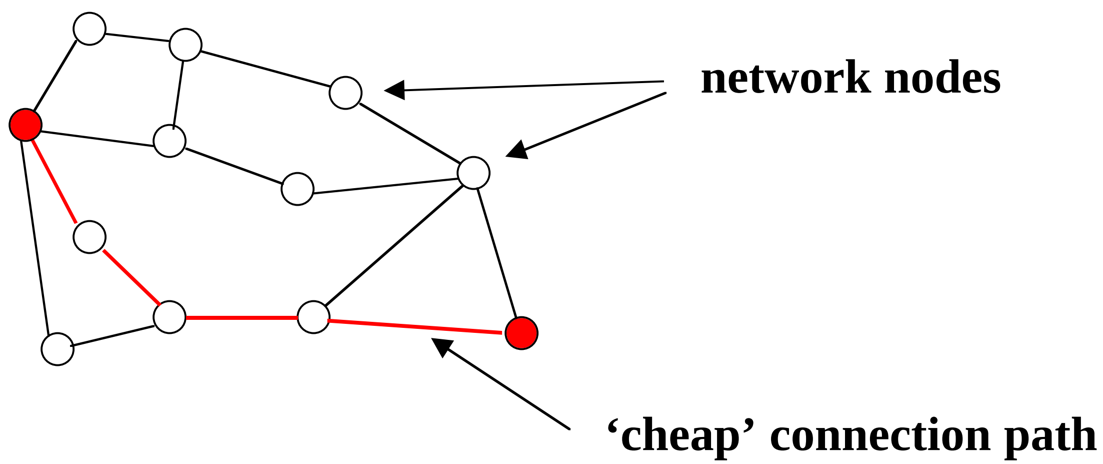
Ant Systems

What are the applications?

- Combinatorial optimization
- Large-scale network routing

Advantages / Disadvantages

- Pro: Can adapt to a dynamically changing problem!
- Contra: No guarantees! Solution approximation



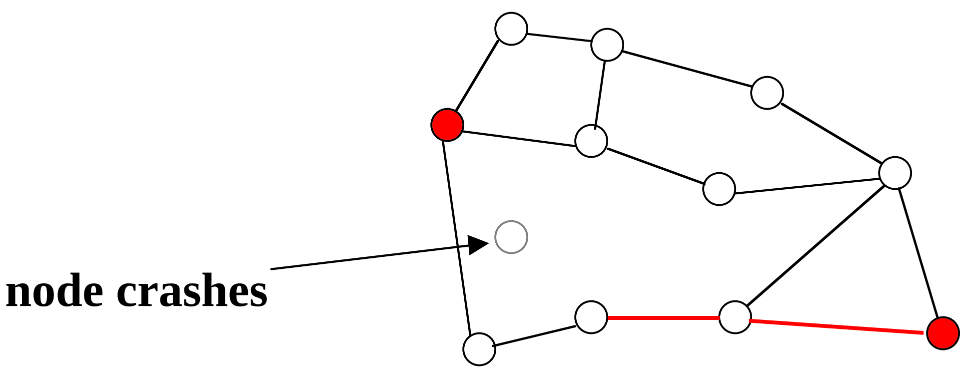
Ant Systems

What are the applications?

- Combinatorial optimization
- Large-scale network routing

Advantages / Disadvantages

- Pro: Can adapt to a dynamically changing problem!
- Contra: No guarantees! Solution approximation



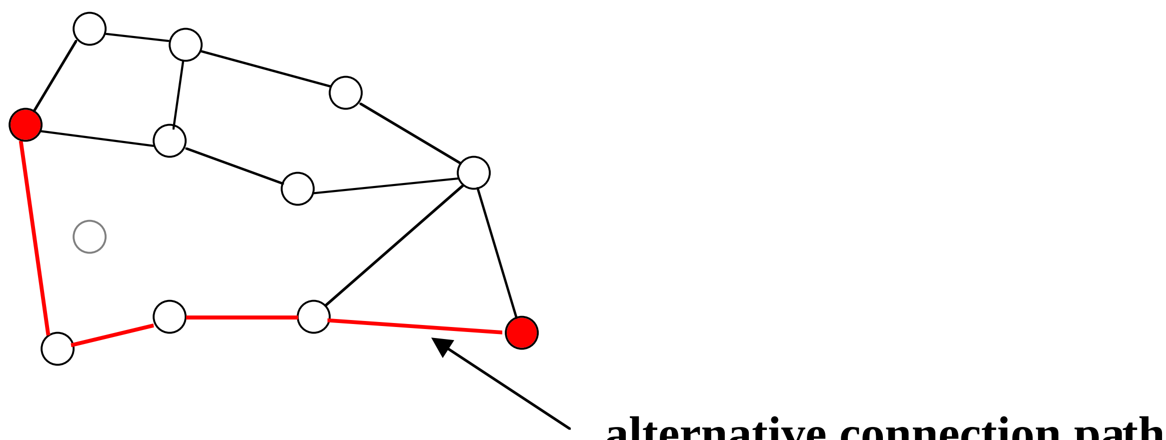
Ant Systems

What are the applications?

- Combinatorial optimization
- Large-scale network routing

Advantages / Disadvantages

- Pro: Can adapt to a dynamically changing problem!
- Contra: No guarantees! Solution approximation



Part II: Particle Swarms

Particle Swarms

Idea

- moving points in the search space, which refine their knowledge by interaction

(Kennedy and Eberhart, 1995)

Particle Swarms

Idea

- moving points in the search space, which refine their knowledge by interaction

What is a particle?

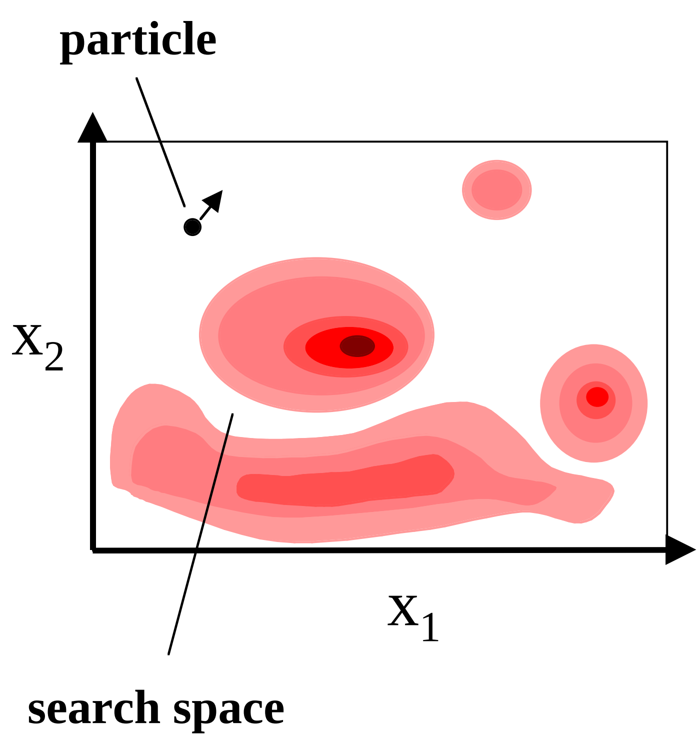
- a particle consists of:

  $\vec{x}_i$ position

  $\vec{v}_i$ velocity

  $\vec{p}_i$ best position found so far

  velocity and position update rules



(Kennedy and Eberhart, 1995)

Particle Swarms

The velocity update rule

$$\vec{v}_i \leftarrow \vec{v}_i + \varphi_1(\vec{p}_i - \vec{x}_i) + \varphi_2(\vec{p}_g - \vec{x}_i)$$

φ_1, φ_2 random numbers (upper limit usually 2.0)

g index of the particle with the best performance so far

$\vec{p}_g$ best vector found by particles of the neighbourhood

Particle Swarms

The velocity update rule

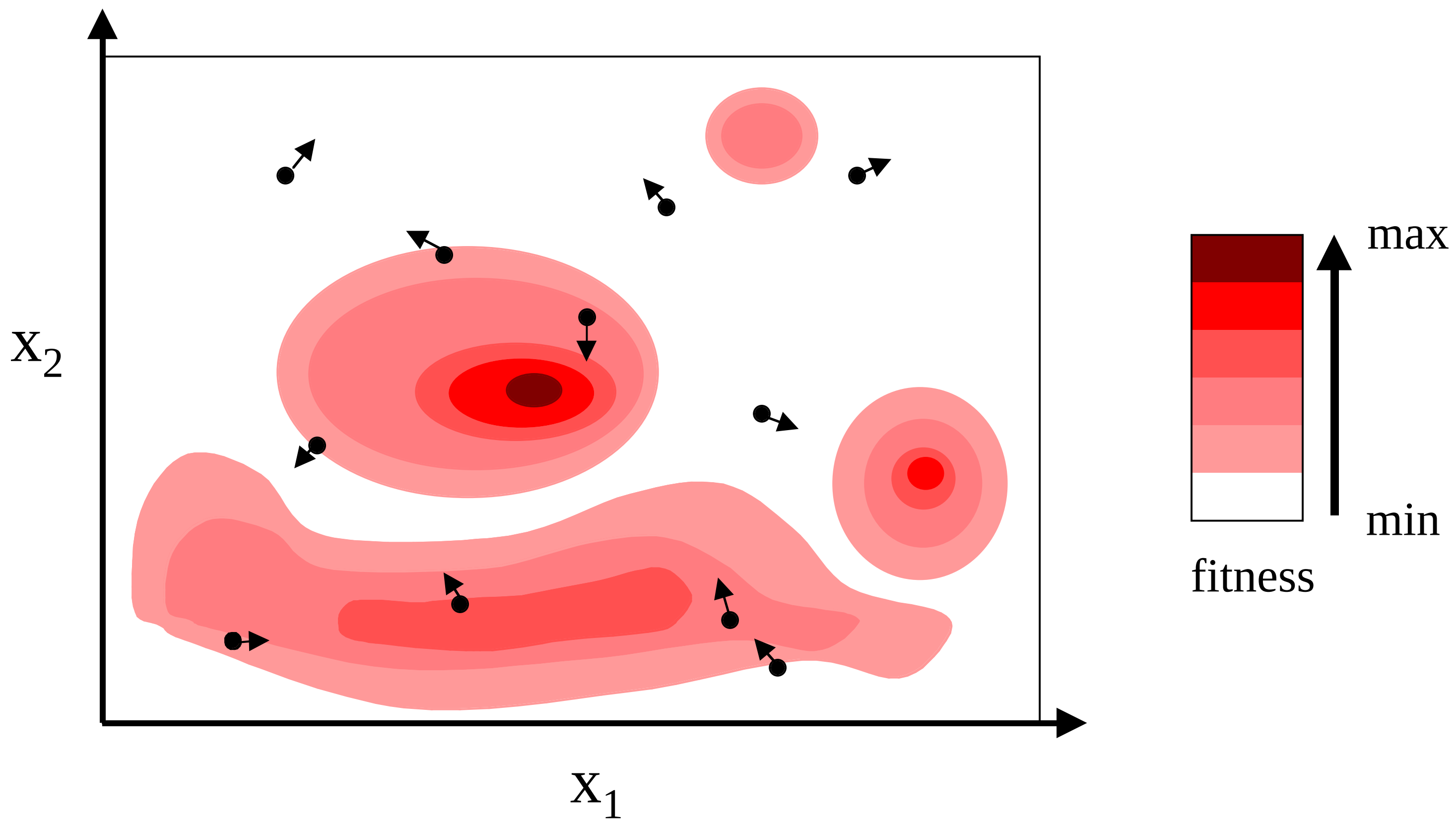
$$\vec{v}_i \leftarrow \vec{v}_i + \varphi_1(\vec{p}_i - \vec{x}_i) + \varphi_2(\vec{p}_g - \vec{x}_i)$$

φ_1, φ_2	random numbers (upper limit usually 2.0)
g	index of the particle with the best performance so far
$\vec{p}_g$	best vector found by particles of the neighbourhood

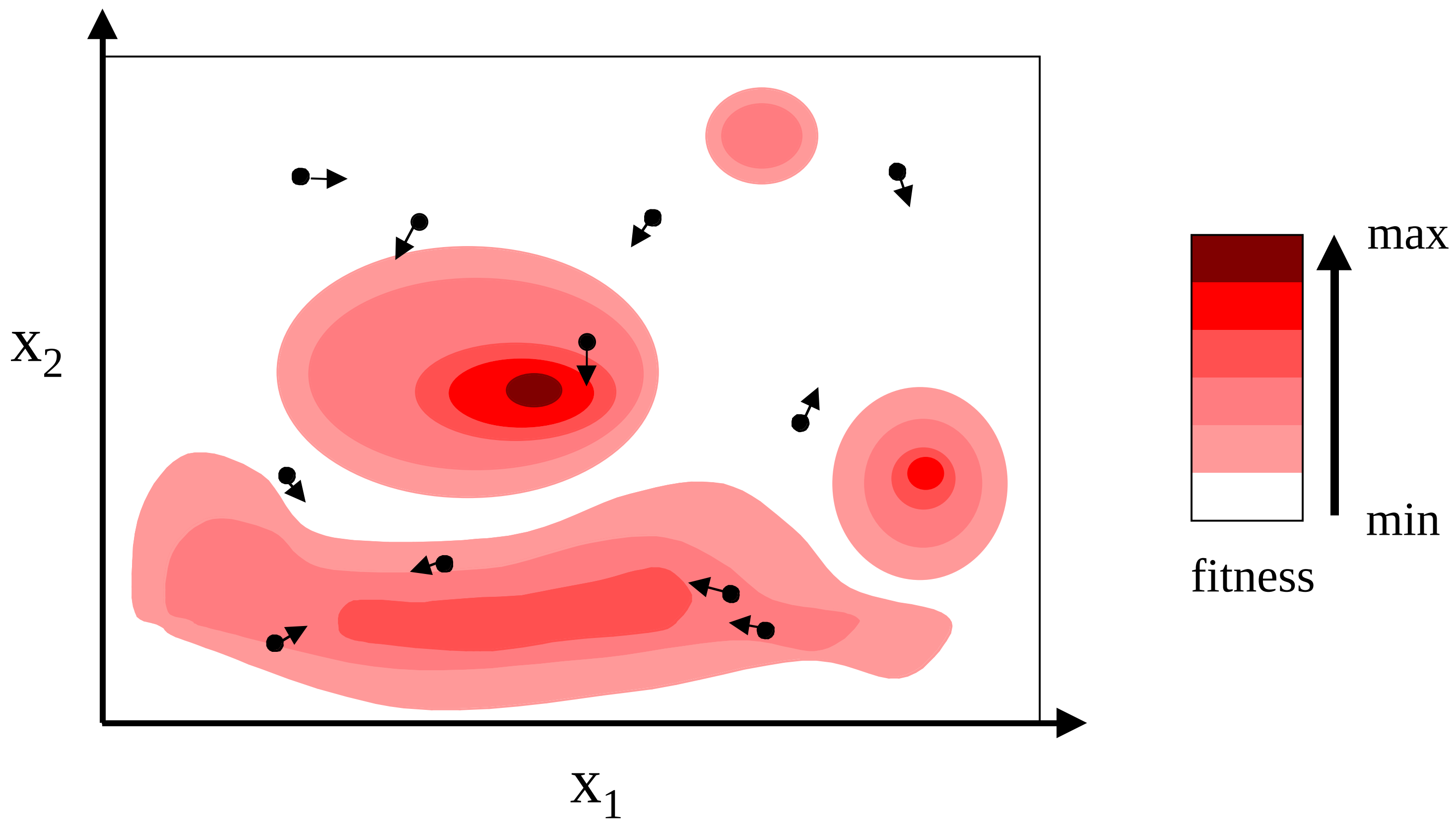
The position update rule

$$\vec{x}_i \leftarrow \vec{x}_i + \vec{v}_i$$

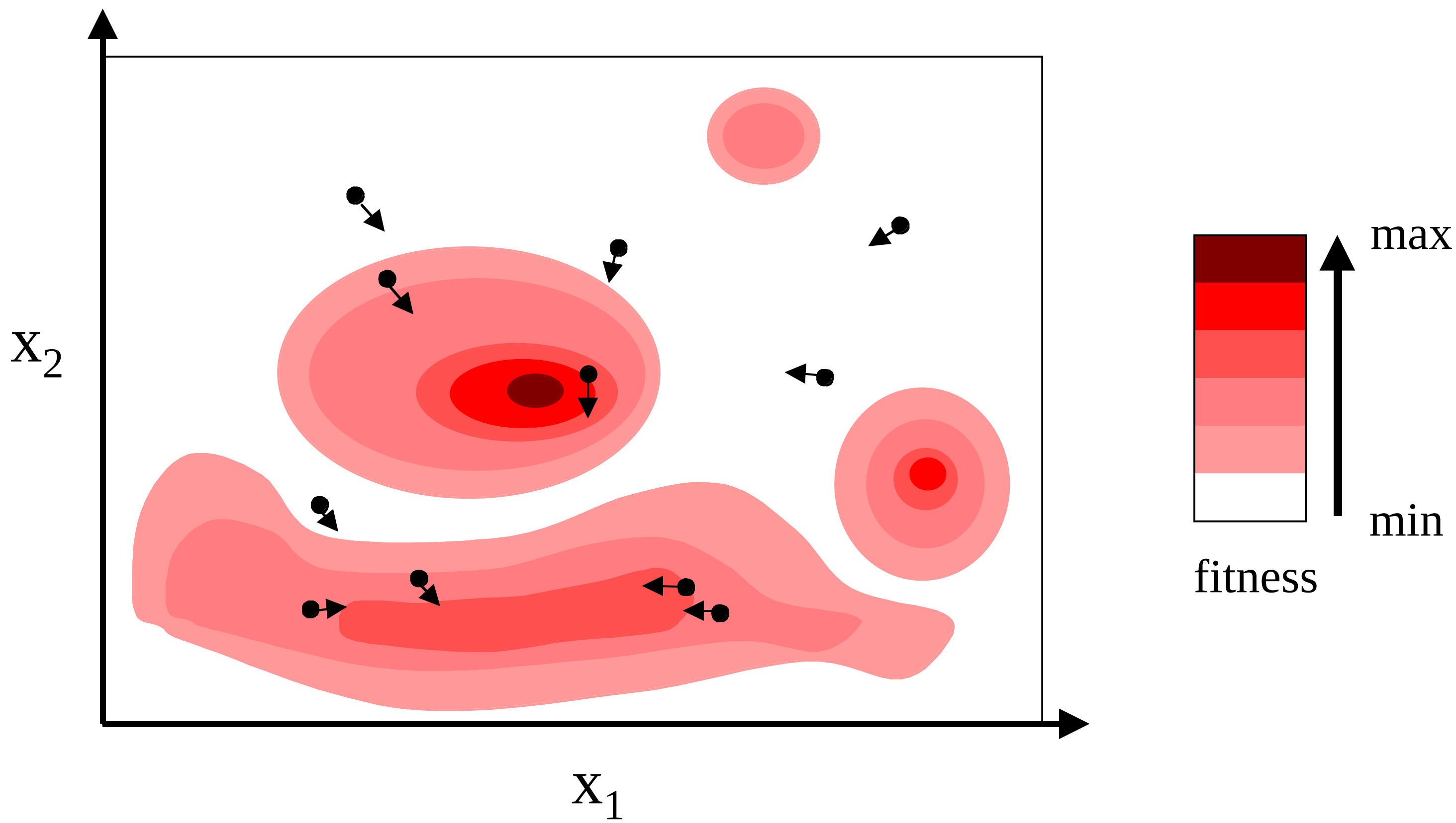
Particle Swarms



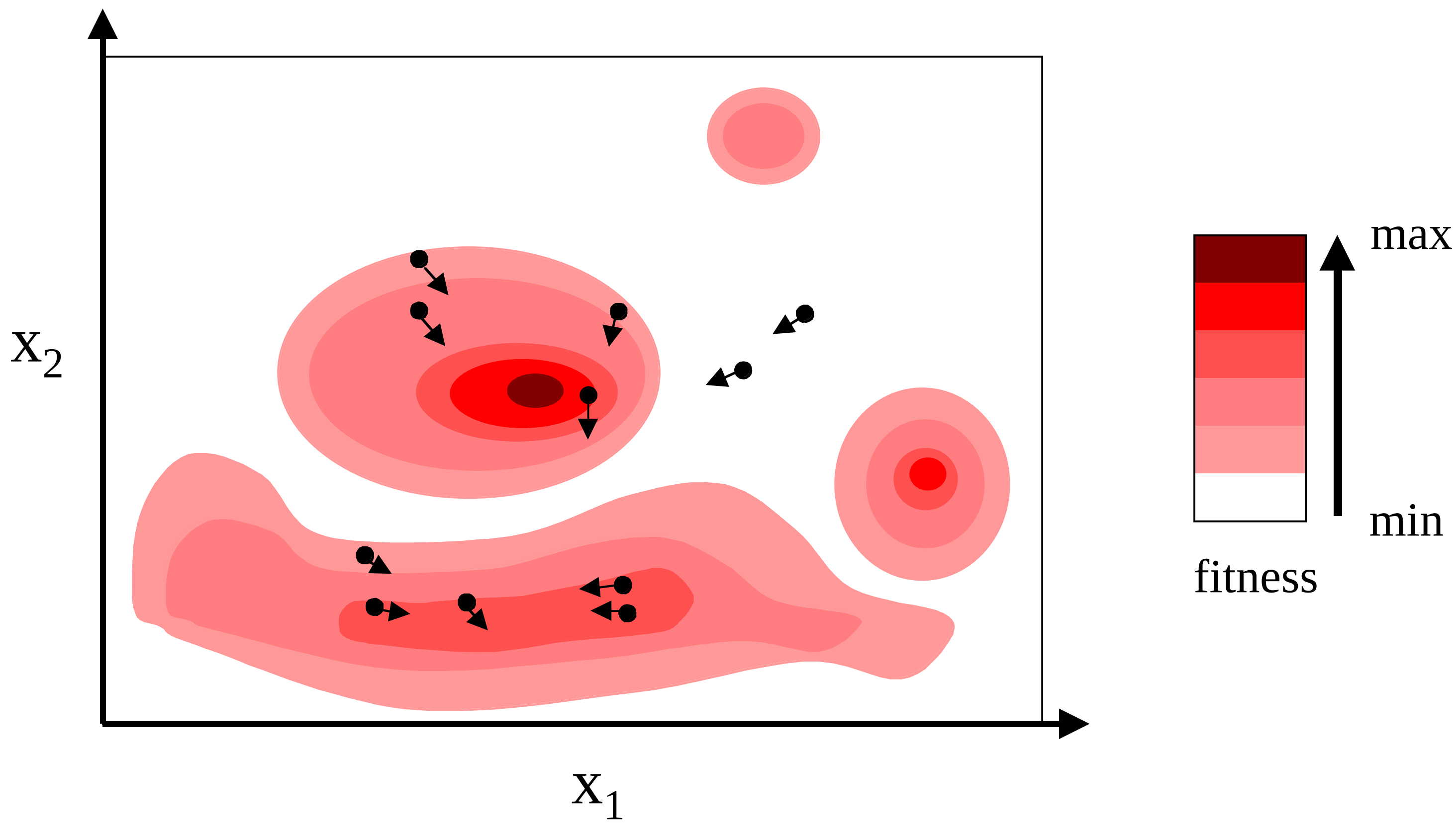
Particle Swarms



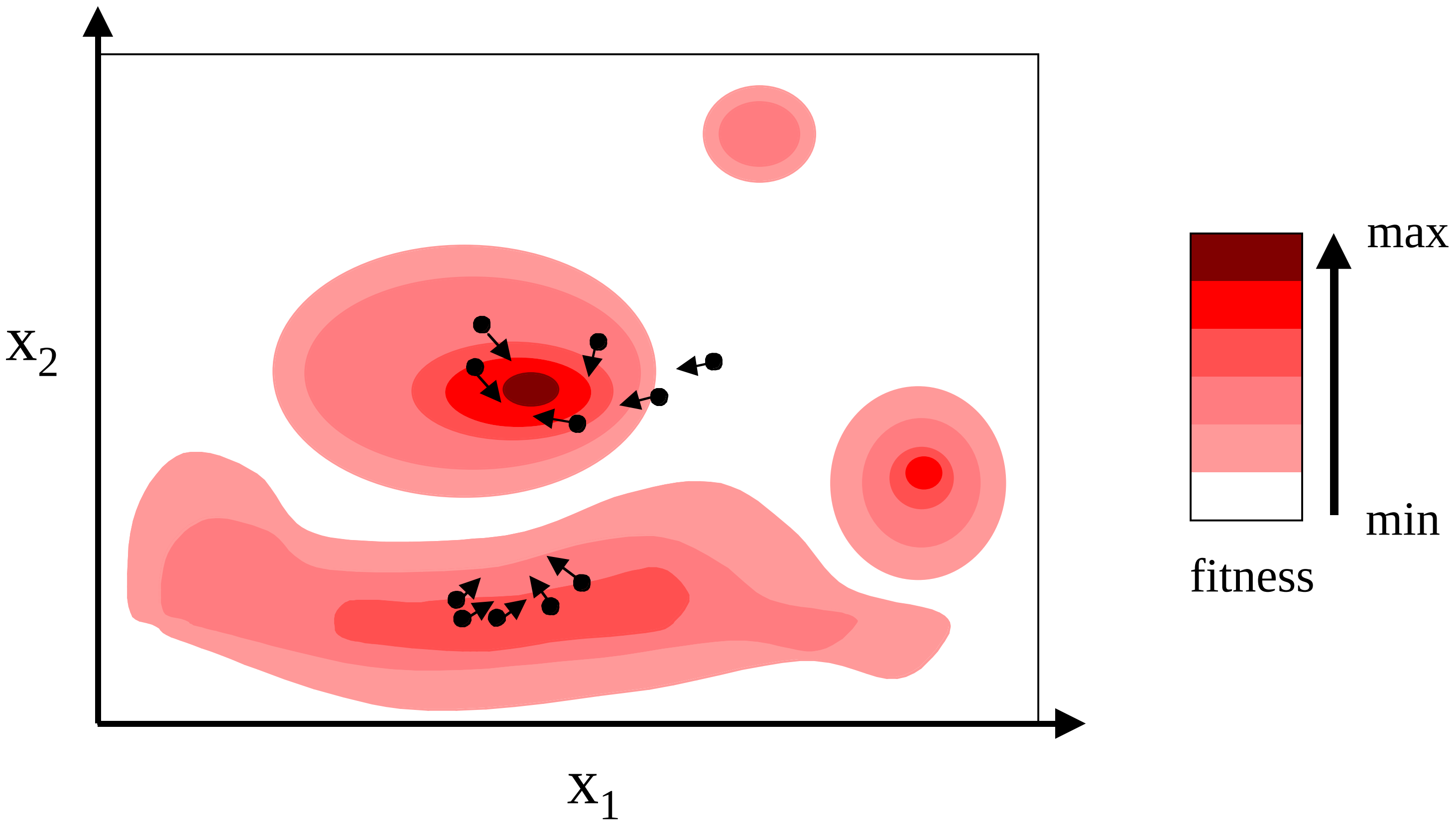
Particle Swarms



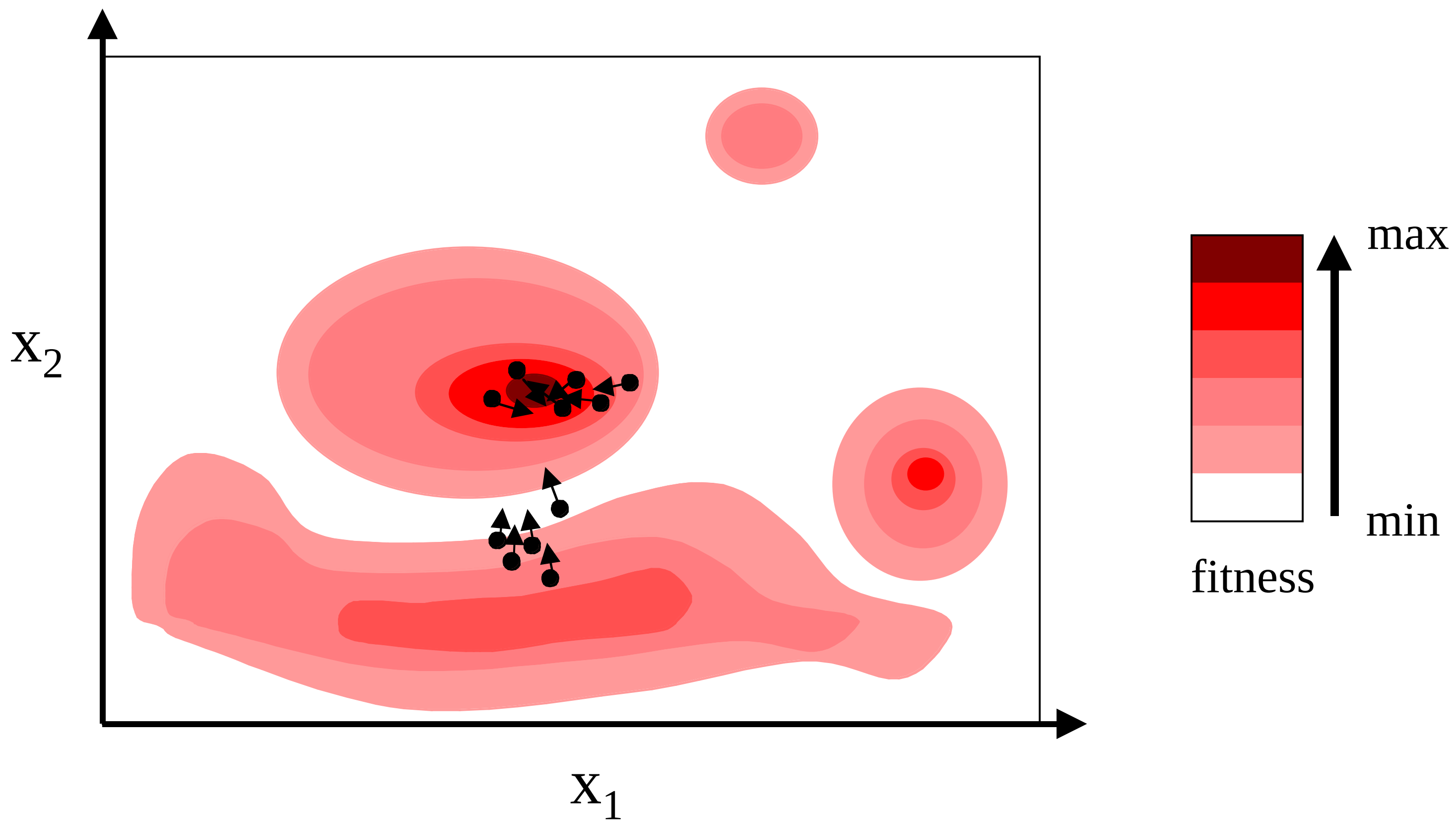
Particle Swarms



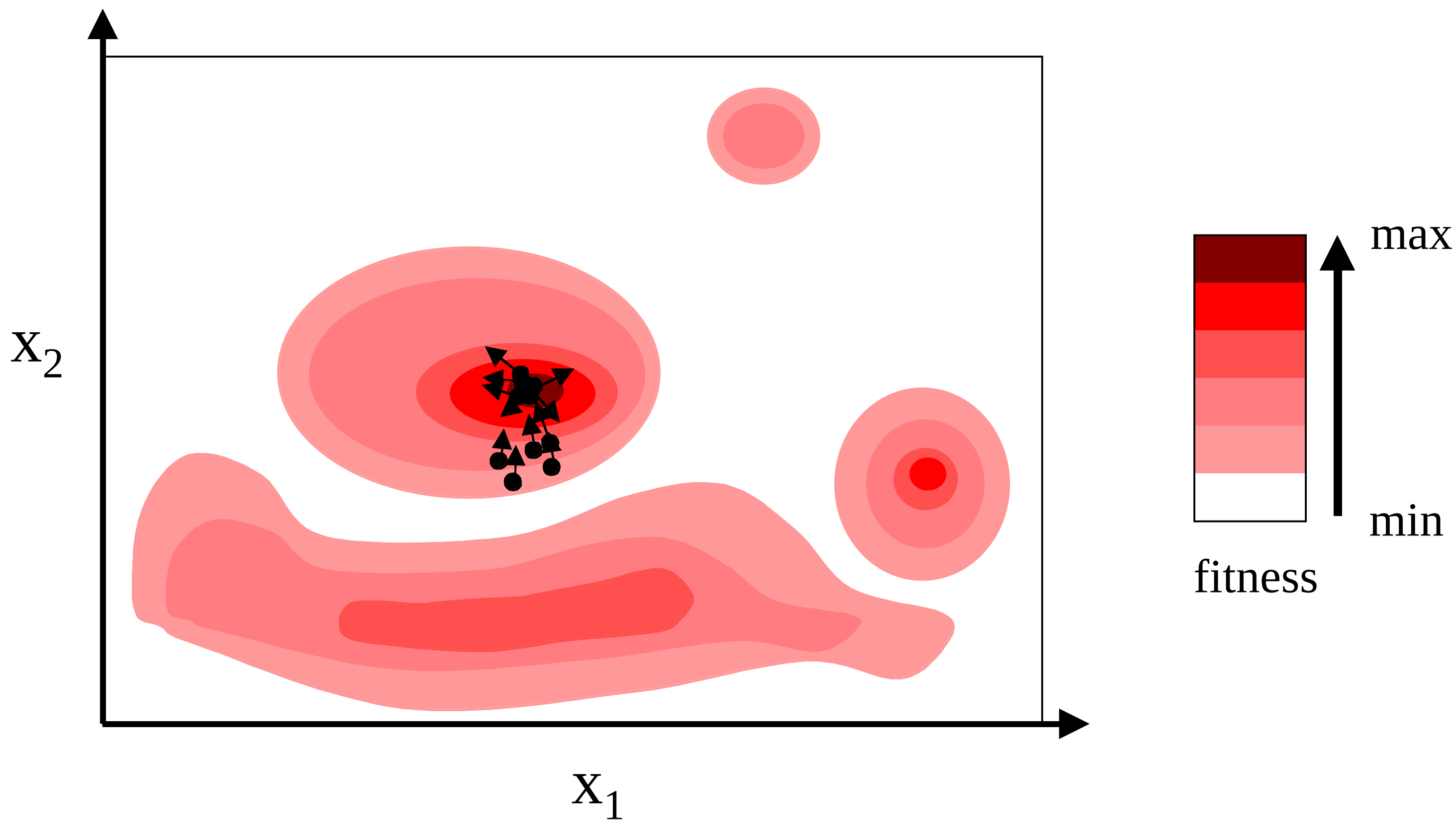
Particle Swarms



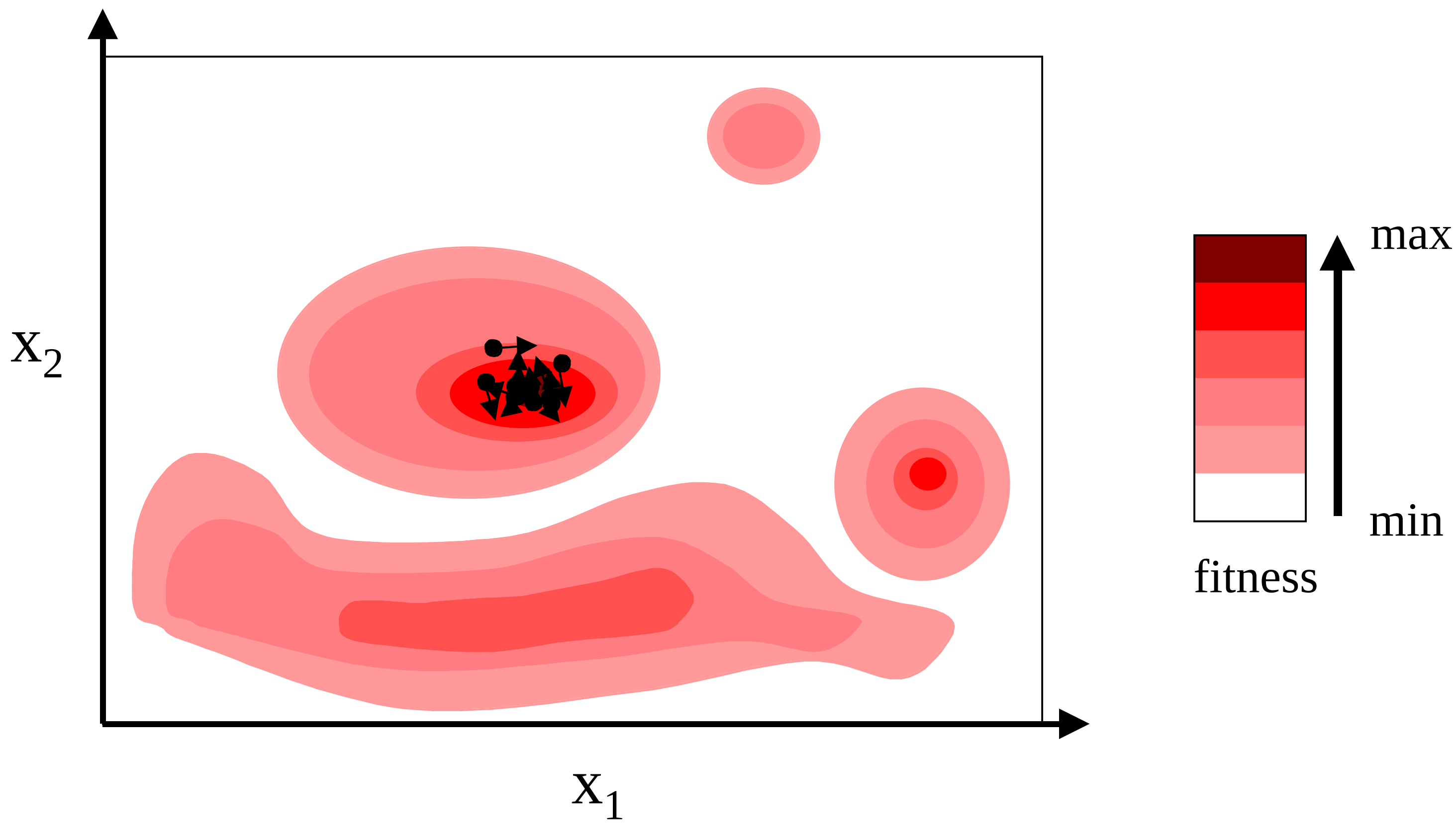
Particle Swarms



Particle Swarms



Particle Swarms



Particle Swarms vs Multinational GAs

	Particle Swarms	**Multinational GA**
Search type	swarm	GA with subpop.
Attractor range	local	global
Space level	genotype (search space)	genotype (search s)
Space topology	continuous	continuous
Extensionality	none	none
Movement	real vector based	mutation&crossover
Reproduction	none	in subpopulations

Particle Swarms vs the Patchwork Model

	Particle Swarms	**Patchwork Model**
Search type	swarm	swarm
Attractor range	local	local
Space level	genotype (search space)	phenotype
Space topology	continuous	discrete
Extensionality	none	discrete
Movement	real vector based	discrete / one step
Reproduction	none	between neighbours

Summary of Ant Systems

- algorithm inspired by cooperative foraging in ants
- can approximate solutions of combinatorial problems
- application: network routing
- pros/cons: can adapt to the problem, but no guarantees

Summary of Ant Systems

- algorithm inspired by cooperative foraging in ants
- can approximate solutions of combinatorial problems
- application: network routing
- pros/cons: can adapt to the problem, but no guarantees

Summary of Particle Systems

- based on swarming points in the search space
- velocity update by best own position and best neighbourhood
- related to Multinational GAs and Patchwork Models
- pros/cons: easy to implement, but no comparison to EAs