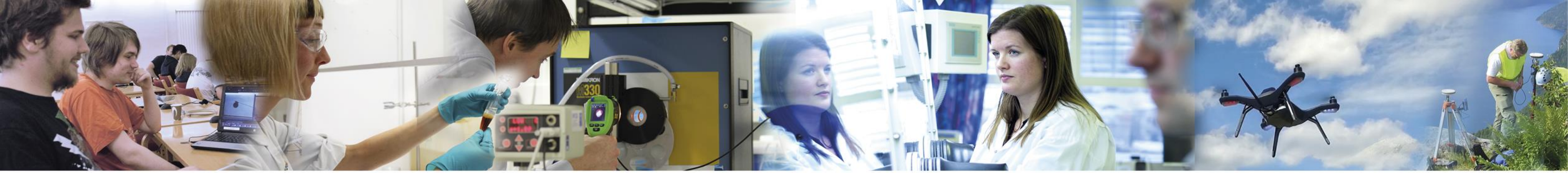


UiO : Department of Technology Systems
University of Oslo

TEK5010/9010 - Multiagent systems 2020
Lecture 3

Swarm intelligence

Jonas Moen

Corona restrictions at UiO

Remember to keep everyone safe by:

1. Washing hands
2. Keeping your distance (1 metre)
3. Staying home if you are sick

https://www.uio.no/english/about/hse/corona/index.html

Highlights lecture 3 – Swarm intelligence (SI)*

- A new form of AI is needed – the social insect metaphore
- SI (emergence) = stigmergy + self-organization
- Ant Colony Optimization (stigmergy)
- Particle Swarm Optimization (self-organization)
- Taxis
- Artificial Potential Field

*Bonabeau, Dorigo & Theraulaz, 1999: chapter 1, 2 and preface,
A collection of papers on ACO and PSO

Types of Agents

- Deductive reasoning agents (1956–present)
  Propose that agents use explicit logical reasoning in order to decide what to do.
- Reactive agents (1985–present)
  Problems with symbolic reasoning led to a reaction against this - the reactive agent movement.
- Hybrid agents (1990–present)
  Hybrid architectures attempt to combine the best of symbolic and reactive architectures.

Swarm intelligence (SI)

Swam intelligence is the emergent collective intelligence of groups of simple agents, typically based on the social insect metaphor.

SI is an appealing research since the world has become «so complex that no single human being can understand it» in terms of utilizing the increasing information and computational resources available [Bonabeau et *al.*, 1999].

Swarm intelligence (SI)

SI was first used by [Beni, 1989; Beni and Wang, 1989/1991; Beni and Hackwood, 1992; Hackwood and Beni, 1991/1992] in the context of cellular robotics.

SI is here extended to include any attempt to design algorithms of distributed problem solving devices inspired by the collective behaviour of social insect colonies and other animal societies.

Swarm intelligence (SI)

[Bonabeau et *al.*, 1999] is one of the first attempts to describe the SI research field.

Based on modelling a biological example and then use this model as a metaphor to design an algorithm, a multiagent system or a group of robots.

Swarm intelligence (SI)

Social insect metaphor emphasizing:

1. Autonomy (only requirement in MAS)
2. Distributedness (no central control)
3. Simple agents (reactive agents)
4. Emergence (properties from the many simple interactions)
5. Stigmergy (local direct and indirect communication)
6. Self-organization (flexibility and robustness)

Swarm intelligence (SI)

Social insects metaphor applications:

1. Optimization
2. Networks
3. Robotics

Social insects

Ants, bees, wasps and termites live in social colonies, what is it that governs them?

Every single insect seems to have its own agenda and yet an insect colony looks so organized.

Social insects

Leaf cutter ants (*Atta*)

Ants that cut leafs from plants and trees to grow fungi. Workers forage for leaves hundreds of meters away from the nest, literally organizing highways to and from foraging sites. [Hölldobler and Wilson, 1978]

Image: inhabitat.com

Social insects

Weaver ants (*Oecophylla*)

Weaver ants form chains of their own bodies, allowing them to cross wide gaps. Workers run back and forth. Such chains are powerful enough to pull leaf edges together and connect the edges with a strong thread of silk emitted from a larva held by workers. [Hölldobler and Wilson, 1978 and 1990]

Image: http://ngm.nationalgeographic.com

Social insects

Army ants (*Eciton*)

Army ants organize hunting raids, involving hundreds of thousands of ants, during which they collect thousands of prey.

[Burton and Franks, 1985; Rettenmeyer, 1963; Schneirla, 1971]

Image: en.wikipedia.org

Social insects

Hive of paper wasp*

Termite hive**

Image: *en.wikipedia.org and **inhabitat.com

Social insects

Specialization

Division of labour reflected in specialization in morphology, age or chance allowing simultaneous parallell work to be performed.

Social insects

Robustness

A removal of a class of workers is often compensated by other workers. Divison of labour exhibits a high degree of plasticity.

Social insects

Emergent behaviour and self-organization (SO)

The sensory system of individual insects is reactive, though when decribing detailed interaction with nest mates and decision-making on the basis of large amounts of information can be quite complex. Yet the complexity of individual insects is still not sufficient to explain the complexity of what social insects can do in terms of coordinated behaviour and nest building.

Social insects

Emergent behaviour and self-organization (SO)

The self-organizing properties of social insects seems to require no need for low level individual comlexity to explain collective behaviour. SO is a major component in a wide range of phenomena in social insects [Bonabeau et *al.*, 1999].

Social insects

Emergent behaviour and self-organization (SO)

The most difficult question is how to connect the individual behaviours with collective performance.

How do complex collective behaviours emerge from interaction among individuals that exhibit simple behaviours?

Social insects

Self-organization (SO)

Hard to build SI systems because what individual behaviour produce desired global behaviour?

- Make a catalogue of collective behaviours?
- Model a few biological systems and use this as basis for modelling engineering problems and their parameter space?

Stigmergy

Stigmergy is indirect interaction [Grassé,1946, 1959]:

Two individuals interact indirectly when one of them modifies the environment and the other responds to the new environment at a later time.

(Direct interaction in insects could be antennation, trophallaxis (food or liquid exchange), mandibular contact (nebb, kjeve), etc.)

Stigmergy

Stigmergy is indirect interaction:

1. Indirect, non-symbolic form of interaction mediated by the environment. Insects exchange information by modifying their environment.
2. Local information is only accessible by those insects that can visit the locus in which it was released/deployed.

Stigmergy

Stigmergy is indirect interaction:

The environment serves as a medium for communication and coordination between simple reactive agents with reduced communication abilities.

- Incremental construction is possible.
- Agents respond to perturbation without being specifically reprogrammed to deal with particular disturbances.

Ant foraging behaviour

The binary bridge experiment [Deneubourg et *al.*, 1990]

Many ant species have trail-laying and trail-following behaviour when foraging. Individual ants deposit a chemical substance called *pheromones* as they move to and from a food source. Foragers follow such pheromone trails.

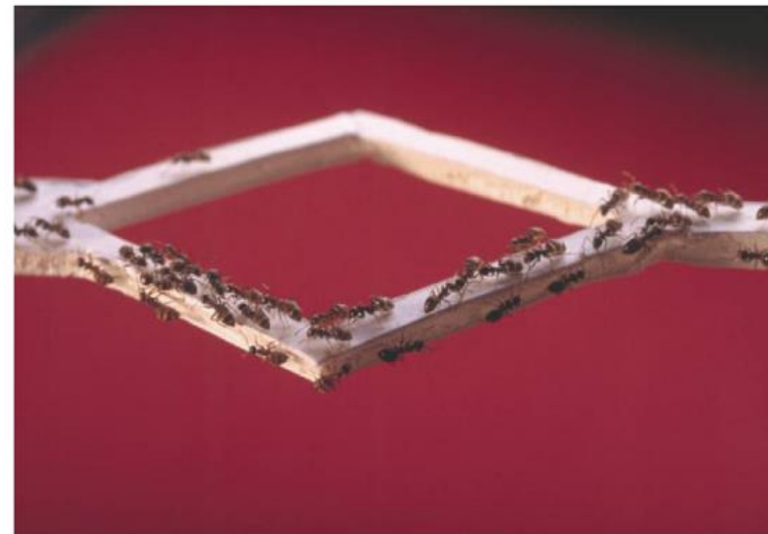


Image: http://home.iitk.ac.in/~adityat/se367/project/

Ant foraging behaviour

The binary bridge experiment [Deneubourg et *al.*, 1990]

Biological model of Argentine ants (*Linepithema humile*)

1. Pheromones ∝ number of ants
2. No evaporation of pheromones

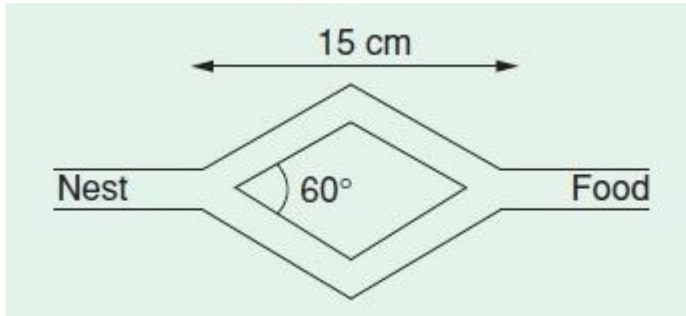


Image: Figure 1, Dorigo et *al.*, 2006

Ant foraging behaviour

The binary bridge experiment [Deneubourg et *al.*, 1990]

$$P_A = \frac{(k+A_i)^n}{(k+A_i)^n+(k+B_i)^n} = 1 - P_B$$

where P_A is probability of an ant choosing branch A
i is total number of ants travelled across bridge
A_i is the number of ants that have used branch A
n determines the degree of nonlinearity, n high gives strong bifurcation
k determines the degree of attraction of unmarked branches, high k gives randomness for low i

Ant foraging behaviour

The binary bridge experiment [Deneubourg et *al.*, 1990]

$$P_A = \frac{(k+A_i)^n}{(k+A_i)^n+(k+B_i)^n}$$

where choice dynamic are

$$A_{i+1} = \begin{cases} A_i + 1 \text{ if } \delta \leq P_A \\ A_i \text{ if } \delta > P_A \end{cases}$$

$$i = A_i + B_i$$

best fit using $n \approx 2$ and $k \approx 20$

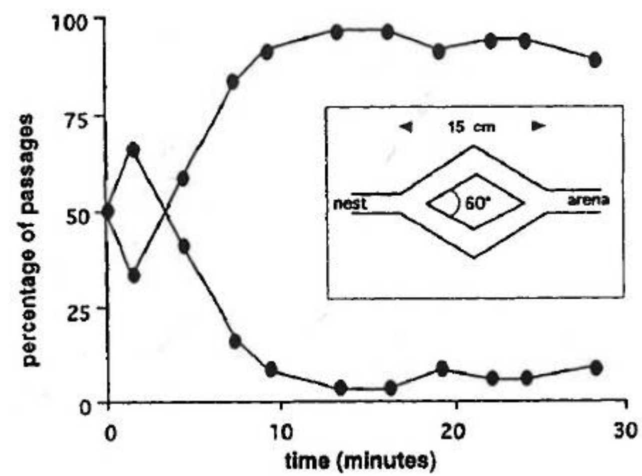
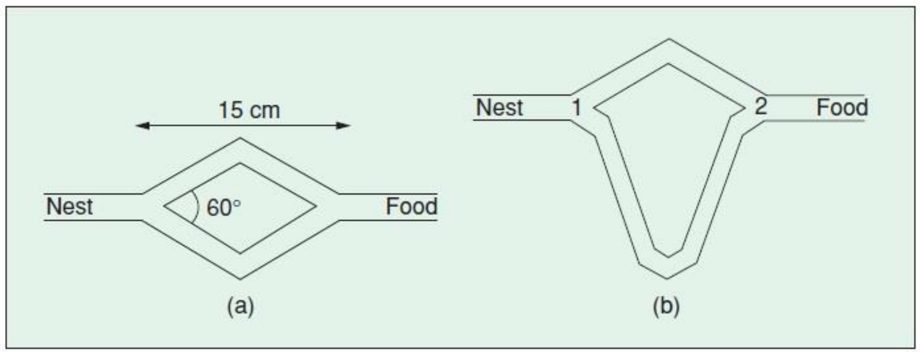


Image: Figure 2.1, Bonabeau et *al.*, 1999

Ant foraging behaviour

The binary bridge experiment [Deneubourg et *al.*, 1990]



Length is important: more ants go to and back on shortest branch leaving more pheromones. Validated experimentally by [Gross et *al.*, 1989].

Image: Figure 1, Dorigo et *al.*, 2006

Ant Colony Optimization (ACO)

ACO takes inspiration from the foraging behaviours of some ant species.

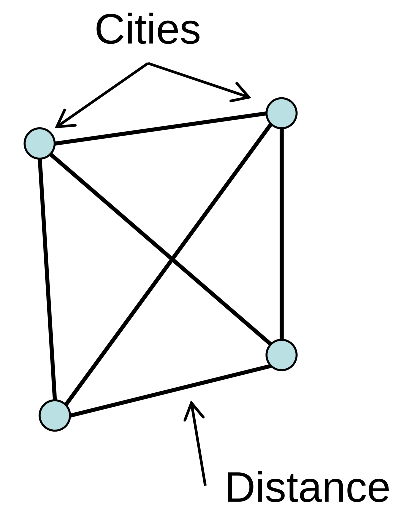
These ants deposit pheromones on the ground in order to make some favourable path that should be followed by other members of the colony.

ACO exploits a similar mechanism for solving optimization problems [Dorigo, 2006].

Ant Colony Optimization (ACO)

Traveling Salesman Problem (TSP)

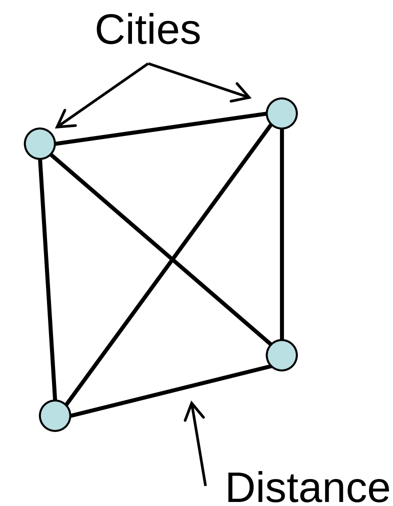
A set of cities is given and the distances between each of them is known. The goal is to find the shortest tour that allows each city to be visited once and only once.



Ant Colony Optimization (ACO)

Traveling Salesman Problem (TSP)

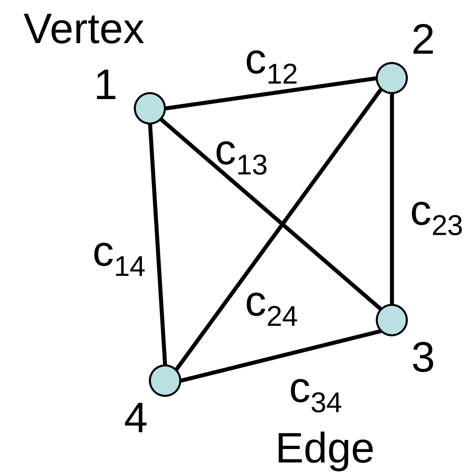
1. ACO is easily adapted to TSP
2. NP-hard
3. Benchmark problem
4. Didactic problem



Ant Colony Optimization (ACO)

Traveling Salesman Problem (TSP)

In more formal terms, the goal is to find a Hamiltonian tour of minimal length on a fully connected graph $G(\boldsymbol{V}, \boldsymbol{E})$.



ACO metaheuristic

A metaheuristic is a set of algorithmic concepts that can be used to define heuristic methods applicable to a wide set of different problems.

ACO metaheuristic

ACO metaheuristic pseudocode [Dorigo et *al.*, 2006]

Set parameteres, initialize pheromone trails
while termination condition not met **do**
    ConstructAntSolutions
    ApplyLocalSearch (optional)
    UpdatePheromones
endwhile

ACO metaheuristic

ACO parameters:

t is iterations or time (i.e. one increment in while-loop)

$k = \{1,2,\ldots,m\}$ is ants

$i = \{1,2,\ldots,n\}$ is the set of vertices (i.e. cities)

c_{ij} is edge from vertex i to vertex j

d_{ij} is length of edge c_{ij}

τ_{ij} is pheromone concentration on edge c_{ij}

ρ is evaporation rate of pheromone

ACO metaheuristic

ConstructSolution:

Artificial ants incrementally build a partial solution by moving from city i to city j. Each of the m ants construct solutions by adding edges between vertices not visited yet.

The choice of edge is biased by the pheromone concentration of the available edges τ_{ij}. This stochastic rule vary across different ACO.

ACO metaheuristic

ApplyLocalSearch:

After all k ant solutions have been obtained, but before updating the pheromones, a local search could be performed. Local search is often applied in state-of-the-art ACO.

ACO metaheuristic

UpdatePheromone:

Increase value of edges associated with good or promising performance and decrease bad ones.

1. Decrease all solutions by evaporation.
2. Increase only good solutions by adding pheromones.

Main ACO algorithms

1. Ant System (AS) [Dorigo et *al*., 1991,1992,1996]

2. Max-Min Ant System (MMAS) [Stützle and Hoos, 2000]

3. Ant Colony Systems (ACS) [Dorigo et *al*., 1996, 1997]

ACO – Ant System (ACO-AS)

The pheromone update rule:

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

where τ_{ij} is pheromone concentration on edge (i, j)

ρ is evaporation rate

m is number of ants

$\Delta\tau_{ij}^{k}$ is pheromones laid on edge (i, j) by ant k

ACO – Ant System (ACO-AS)

The pheromone update rule:

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_k} \text{ if ant } k \text{ used edge } (i,j) \text{ on its tour} \\ 0 \text{ otherwise} \end{cases}$$

where Q is a constant

L_k is the length of the tour constructed by ant k

ACO – Ant System (ACO-AS)

The transition rule (the probability of going to city j):

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{c_{il} \in N(s^p)} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}} \text{ if } c_{ij} \in N(s^p) \\ 0 \text{ otherwise} \end{cases}$$

where $N(s^p)$ is the set of feasible components (cities not visited yet)

α, β are nonlinear control parameters

$\eta_{ij} = \dfrac{1}{d_{ij}}$ is invers distance between city i and j

Current hot topics in ACO

- Dynamic optimization problems
- Multi-objective optimization
- Stochastic optimization problems

- ACO in real world applications

Bird flocking

Starling birds (*Sturnidae*)

"The starlings are generally a highly social family. Most species associate in flocks of varying sizes throughout the year. This sociality is particularly evident in their roosting behaviour; in the non-breeding season some roosts can number in the thousands of birds" [Wikipedia, 2017].

Image: National Geographic

Fish schooling

Sardines (*Clupeidae*)

Fish may derive benefits from shoaling behaviour including:

- Defence against predators
- Enhanced foraging success
- Higher success in finding a mate
- Increased hydrodynamic efficiency

Image: en.wikipedia.org

Mathematical modelling

The observational approach is complemented by the mathematical modelling of schools. The most common mathematical models of schools instruct the individual animals to follow three rules [Wikipedia, 2017]:

1. Move in the same direction as your neighbours
2. Remain close to your neighbours
3. Avoid collisions with your neighbours

i.e. Boids*

*Reynolds, “Flocks, Herds And Schools”, ACM 1987

Particle Swarm Optimization (PSO)

PSO* is a metaheuristic for optimization of continuous nonlinear functions inspired by bird flocking and fish schooling (in contrast to ACO often used in dynamic combinatorial optimization problems).

According to Google Scholar PSO* ~50.000 citings

*Kennedy & Eberhart, “Particle swarm optimization”, IEEE 1995

Particle Swarm Optimization (PSO)

PSO metaheuristic pseudocode*

```
Set parameteres, initialize particles
while termination condition not met do
    for each particle
        for each dimension
            UpdateParticleVelocity
        UpdateParticlePosition
        UpdateParticleAndSwarmBestPosition
endwhile
```

*Clerc, “Standard Particle Swarm Optimisation”, 2012
https://hal.archives-ouvertes.fr/hal-00764996

PSO metaheuristic

PSO parameters:

t is iterations or time (i.e. one increment in while-loop)

$i = \{1,2,\ldots,N\}$ is particle

$d = \{1,2,\ldots,M\}$ is dimension

v_{id} is velocity of particle i in dimension d

x_{id} is position of particle i in dimension d

p_{id} is the best position of particle i in dimension d

p_{gd} is the best position of all particles in dimension d

PSO metaheuristic

UpdateParticleVelocity:

$$v'_{id} = \underbrace{w \cdot v_{id}}_{\text{Inertia term}} + \underbrace{w_1 \varphi_1 (p_{id} - x_{id})}_{\text{Cognition term}} + \underbrace{w_2 \varphi_2 \left(p_{gd} - x_{id}\right)}_{\text{Social term}}$$

where v'_{id} is updated velocity of particle i in dimension d

φ_1 and φ_2 are uniform random variables $[0,1]$

w, w_1 and w_2 are paremeters that need to be tuned

PSO metaheuristic

UpdateParticlePosition:

$$x'_{id} = x_{id} + v'_{id}$$

PSO metaheuristic

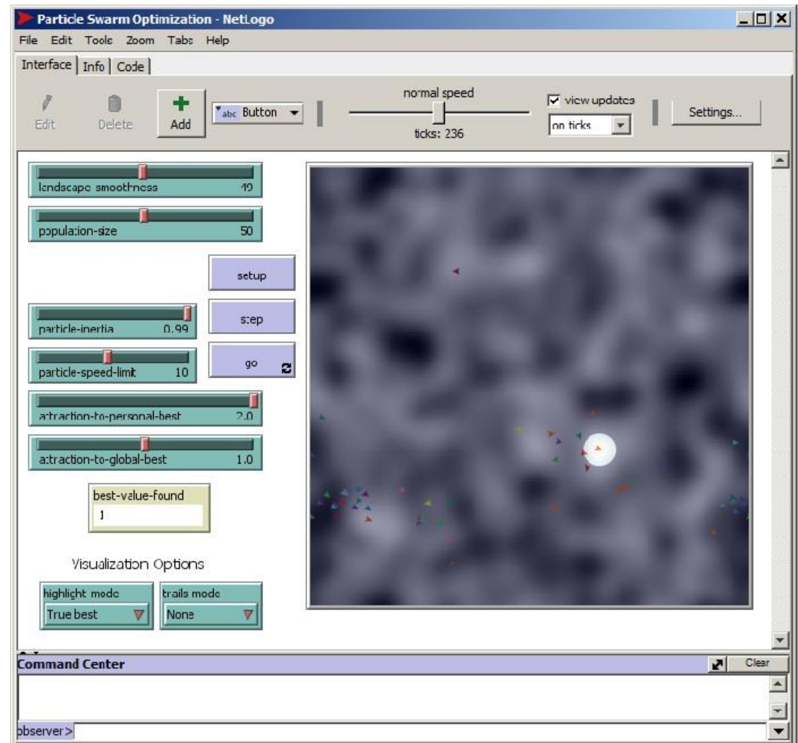
UpdateParticleAndSwarmBestPosition :

$$p'_{id,best} = \max\left(objective\left(p_{id,best}\right), objective\left(x_i\right)\right)$$

$$p'_{gd,best} = \max\left(objective\left(p_{gd,best}\right), objective\left(p_{gd}\right)\right)$$

Particle Swarm Optimization (PSO)

Demonstration of PSO using NetLogo



Current hot topics in PSO

- Dynamic optimization problems
- Multi-objective optimization
- Stochastic optimization problems

- PSO in real world applications

Taxis

“A taxis (plural taxes, from Ancient Greek, meaning 'arrangement') is the movement of an organism in response to a stimulus such as light or the presence of food. Taxes are innate behavioural responses.” [Wikipedia, 2017]

Taxis is often applied to source seeking, i.e. finding a hidden resource using chemotaxis, thermotaxis or phototaxis to name a few taxes.

Chemotaxis

Source seeking with inspiration from bacteria like *E.coli**. E.coli search for nutrients by moving randomly some distance before selecting a new course influenced by the environment.

1. An unfavourable environment makes the variation in the new course large.
2. A favourable environment makes the variation in the new couse smaller.

Also the direction changing frequency is higher in varying environments.

*Berg & Brown, “Chemotaxis in E.coli analysed by 3D tracking”, Nature, 1972

Chemotaxis

*Image: FreeScienceLectures.com

Chemotaxis

An E.coli algorithm*

Set parameters, initialize robots
while termination condition not met **do**
  read sensor s_t
  if $s_t > s_{t-1}$ **then**
    turn $\pm random(5^{\circ})$ and move forward
  else
    turn $\pm random(180^{\circ})$ and move forward
endwhile

*Russel et *al.*, “A comparison of reactive robot chemotaxis algorithms”, Nature, 1972

Artificial Potential Field (APF)

Obstacle avoidance can be a challenge in many mobile robot systems.

APF* create repulsive force fields around obstacles in the environment, either preprogrammed off-line or deposited on-line dynamically in the environment.

*Khatib, "Real-time obstacle avoidance for manipulators and mobile robots", Nature, 1972

Artificial Potential Field (APF)

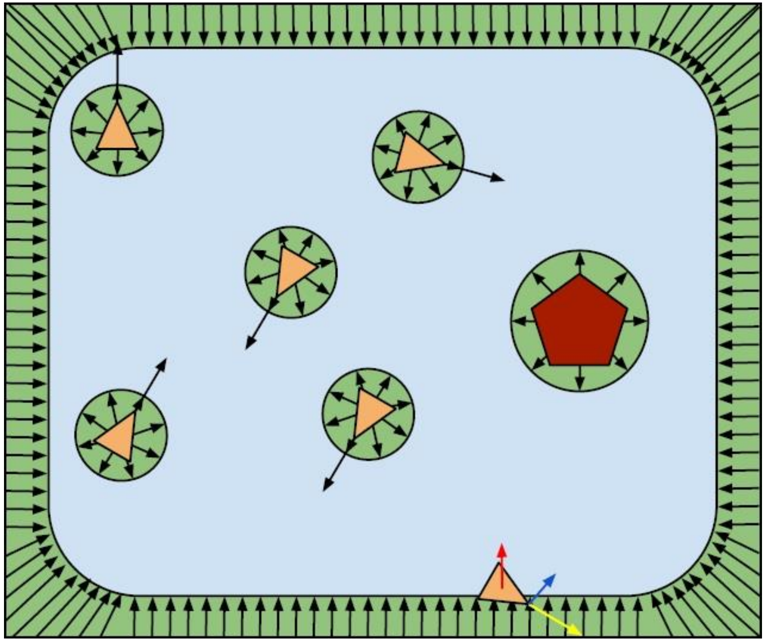


*Image: Figure 2.10 in Jørgen Nordmoen, MSc NTNU, 2013

Summary lecture 3 – Swarm intelligence*

- A new form of AI is needed – the social insect metaphore
- SI (Emergence) = stigmergy + self-organization
- Ant Colony Optimization (ant foraging behaviour)
- Particle Swarm Optimization (self-organization)
- Taxis
- Artificial Potential Field

*Bonabeau, Dorigo & Theraulaz, 1999: chapter 1, 2 and preface,
A collection of papers on ACO and PSO