

A0M33EOA:  
Other types of Metaheuristics

Petr Pošík

Czech Technical University in Prague  
Faculty of Electrical Engineering  
Department of Cybernetics

<b>Introduction</b>	<b>2</b>
Swarm Intelligence .....	3
<b>PSO</b>	<b>4</b>
PSO .....	5
PSO Algorithm .....	6
PSO: Demo .....	7
<b>Ant Colonies</b>	<b>8</b>
Ant colonies .....	9
ACO .....	10
Algorithm parts .....	11
Applications .....	13
<b>Conclusions</b>	<b>14</b>
Summary .....	15

### Swarm Intelligence

Swarm intelligence:

- In nature: swarm (cz: roj, hejno) of small simple 'units' is able to create very complex behavioral patterns via cooperation.
- **Emergence**: non-linear interactions of simple rules  $\longrightarrow$  complex behavior of the whole system.
- Analogy to the behavior of bees, wasps, ants, fish, birds, ...

An engineering view:

- Is it possible to model these systems *in silico* and use that model to solve a practical task?
- How to design the simple units and their interactions such that a practically useful system emerges?

### Particle Swarm Optimization

#### Particle Swarm Optimization

**Particle Swarm Optimization (PSO)**: an optimization algorithm inspired by the behavior of birds.

Inspiration:

- Birds fly over the landscape and lands on the highest hill.
- Birds are modeled by particles in a multidimensional vector space.
- The particles have their *position* and *speed* (and momentum).
- They remember their own best position (i.e., the highest place of the landscape they flew over), but also
- they communicate and use the best position of their neighboring particles to update their own position and speed.
- The communication is usually of 2 types:
  1. **Globally best position** is known to all particles and is updated as soon as any particle finds an improvement.
  2. **Best position in neighborhood** is shared among a group of neighboring particles.

# PSO Algorithm

## Algorithm 1: Canonical PSO

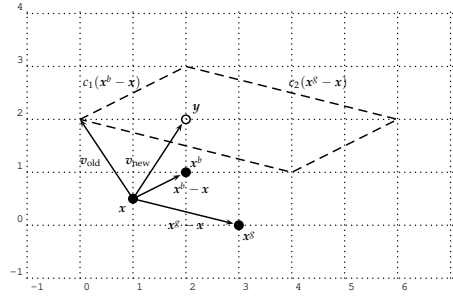
```

1 begin
2   Initialize positions  $x_i$  and velocities  $v_i$ .
3   Initialize personal best positions  $x_i^b \leftarrow x_i$ .
4   Initialize globally best position  $x^g \leftarrow x_k, \forall i : f(x_k) \leq f(x_i)$ 
5   for  $i = 1, \dots, N$  do
6      $v_i \leftarrow \omega v_i + c_1 r_1 \circ (x_i^b - x_i) + c_2 r_2 \circ (x^g - x_i)$ 
7      $x_i \leftarrow x_i + v_i$ 
8     If  $f(x_i) < f(x_i^b)$ ,  $x_i^b \leftarrow x_i$ .
9     If  $f(x_i) < f(x^g)$ ,  $x^g \leftarrow x_i$ .
10  If termination condition not satisfied, go to 5.

```

Meaning of symbols:

$f$  objective function (landscape)  $f : \mathcal{R}^D \rightarrow \mathcal{R}$   
 $N$  the number of particles  
 $x_i$  particle positions,  $x_i \in \mathcal{R}^D$   
 $v_i$  particle velocities,  $v_i \in \mathcal{R}^D$   
 $x_i^b$  personal best position



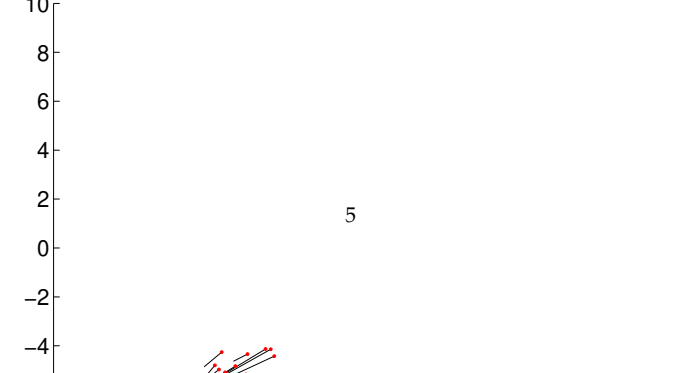
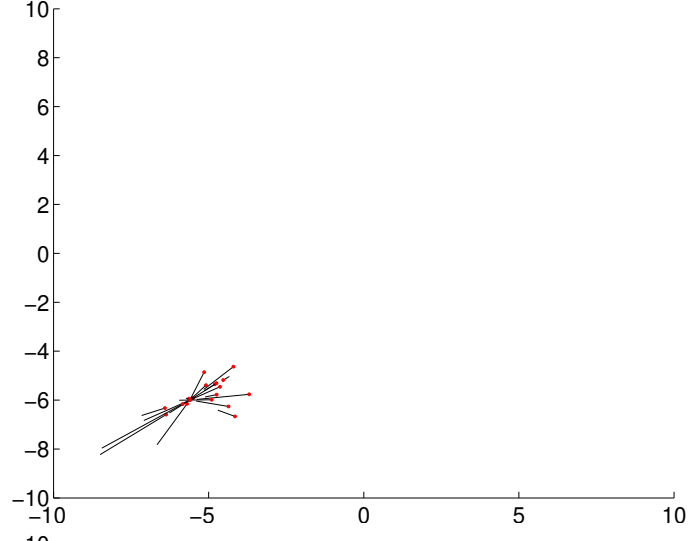
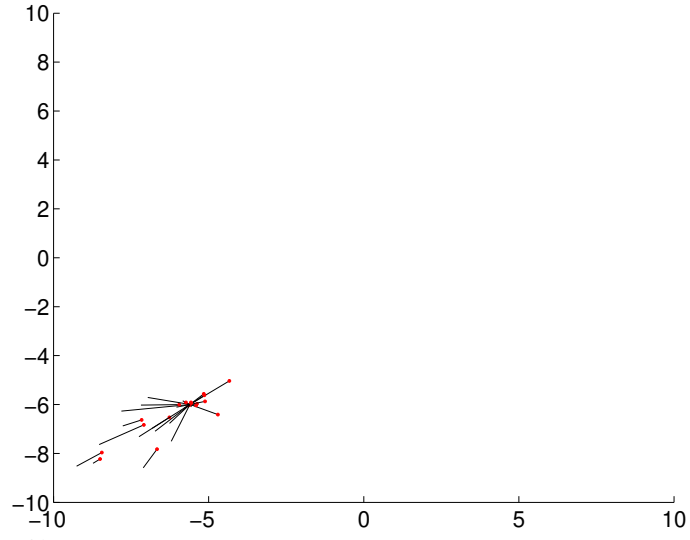
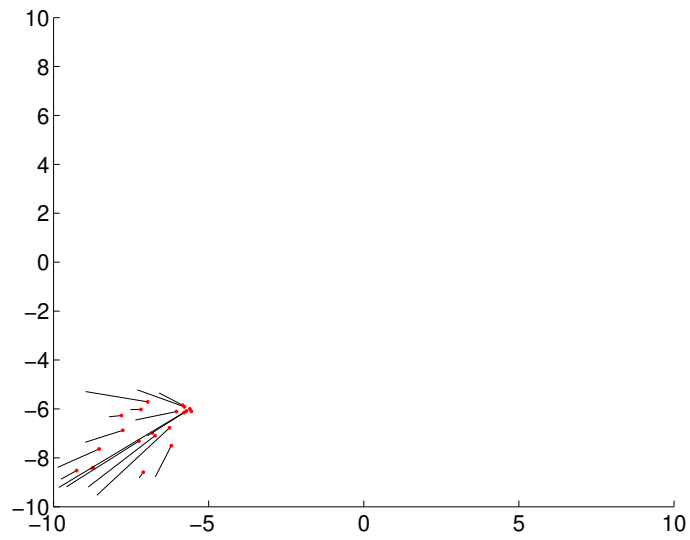
$x^g$  globally best position  
 $\omega$  particle momentum, suitable value is e.g. 0.9, sometimes it decreases during simulation e.g. to 0.4.  
 $c_1, c_2$  attraction constants, 'cognitive' and 'social' components, suitable values between 1 and 2  
 $r_1, r_2$  random vectors from  $U(0,1)^D$   
 $\circ$  vector multiplication by items

[KE02] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948, August 2002.



# PSO: Demo

PSO on 2D Sphere function:





**Ant colonies**

Ants:

- Social insects
- Ant colonies exhibit an intelligent behavior:
  - labor division, work coordination
  - complex nests
  - ability to find 'low-energy' path between the nest and a food source
- They communicate by
  1. physical contact (they touch with their antennas)
  2. interaction with the environment (pheromone trails)

*"In nature, ants first search their environment randomly, until they find a source of food. Then, they return to the nest and lay a pheromone trail behind. Other ants are able to sense this pheromone trail and are able to follow it, and thus make it stronger. The pheromone evaporates; after the food source is exhausted, the random foraging reemerges."*

P. Pošík © 2018

A0M33EOA: Evolutionary Optimization Algorithms – 9 / 15

**Ant Colony Optimization**

**Ant Colony Optimization (ACO)** is a class of stochastic optimization algorithms for solving combinatorial problems.

Similarities with the real ants:

- a colony of cooperating individuals
- pheromone trail
- indirect communication via pheromone (stigmergy)
- probabilistic decision making, local strategies

Differences from the real ants:

- (usually) discrete world (a graph)
- inner state, memory
- the amount of pheromone train can depend on the solution quality
- may use several types of pheromones

**Algorithm 2: ACO**

```

1 begin
2   Initialize the pheromone trails on graph edges:  $\tau_{ij}(0) = \tau_0$ .
3   Set the initial position of ants in the graph.
4   while not termination condition do
5     foreach ant do
6       Build a solution.
7       Apply local search. // Optional, but used very often.
8     Update pheromone trails.
```

P. Pošík © 2018

A0M33EOA: Evolutionary Optimization Algorithms – 10 / 15

## Algorithm parts

### Ant $k$ constructs a solution:

- Probability the ant will move from the current node  $i$  to neighboring node  $j$  is

$$p_{ij}^k(t) = \frac{(\tau_{ij}(t))^\alpha (\eta_{ij})^\beta}{\sum_{l \in \mathcal{N}_i^k} (\tau_{il}(t))^\alpha (\eta_{il})^\beta}, \text{ kde } j \in \mathcal{N}_i^k,$$

where  $\tau_{ij}$  is the amount of pheromone on edge  $i \rightarrow j$ ,  
 $\eta_{ij} = \frac{1}{d_{ij}}$  is a known heuristic information,  
 $\alpha, \beta$  are coefficients of the relative influence of pheromone and heuristic information,  
 $\mathcal{N}_i^k$  is a set of graph nodes accessible to ant  $k$  from node  $i$ .

- If  $\alpha = 0$ , only the heuristic information has an effect, and the solution construction reduces to greedy algorithm (nearest neighbor heuristic).
- If  $\beta = 0$ , only the pheromone trail has an effect. The paths found in the first iteration have a big influence. Moreover, if  $\alpha > 1$ , stagnation occurs very fast, i.e. all ants use the same (not optimal) path.
- Suggested values of parameters:

$$\alpha = 1 \qquad \beta = 2 \text{ až } 5 \qquad \rho = 0.5 \qquad m = n \text{ (TSP)} \qquad \tau_0 = m / C^m \text{ (TSP)}$$

$m$  is the number of ants,  $n$  is the number of cities,  $C^m$  is the length of the path constructed by the nearest neighbor heuristic.

## Algorithm parts (cont.)

### Pheromone update on all edges

- Done after all ants find their solution.
- Pheromone evaporation:  $\tau_{ij} \leftarrow (1 - \rho)\tau_{ij}$   
.  $\rho$  is the evaporation rate, allows to 'forget' bad paths.
- Pheromone deposition from all ants:  $\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k$ , where

$$\Delta\tau_{ij}^k = \begin{cases} 1/C^k & \text{if ant } k \text{ used edge } i \rightarrow j \\ 0 & \text{otherwise,} \end{cases}$$

$C^k$  is the length of the path of ant  $k$ .

### Other options:

- The best path is reinforced the most.
- The amount of deposited pheromone is proportional to the ant rank according to the path lengths (i.e., not directly proportional to path lengths).
- Update of pheromone trails as soon as an ant uses and edge.
- More types of pheromones can be used:
  - Ants can start from both the nest and the food source.
  - We can have more types of ants.
- ...



## Applications

ACO was able to find good solutions in the following tasks:

- Traveling salesperson problem
- Network routing, vehicle routing
- Scheduling
- Quadratic assignment problem
- Shortest common supersequence
- Classification rule learning
- ...

Advantages:

- The graph topology can change in time (e.g. in routing problems)

Demo: ant foraging

## Conclusions

14 / 15

### Summary

- There are plenty of nature-inspired techniques, other than EAs.
- Swarm intelligence takes advantage of the emergent swarm behavior which is a result of simple interactions among individual swarm members.
- Particle swarm optimization primarily aims at real-parameter optimization, but there are also variants suitable for discrete spaces.
- Ant colonies are used to solve problems which can be reduced to search for the shortest path in a graph (combinatorial problems). Again, variants for real-parameter optimization exist (but are somewhat 'unnatural').