CSE/ECE 848
Introduction to
Evolutionary Computation

Module 3 - Lecture 12 - Part 4
Particle Swarm Optimization

Wolfgang Banzhaf, CSE
John R. Koza Chair in Genetic Programming

Particle Swarm Optimization

Introduced by James Kennedy and Russell Eberhart (1995) “Particle Swarm Optimization.” Proceedings of the IEEE International Conference on Neural Networks, Australia, pp. 1942-1945.

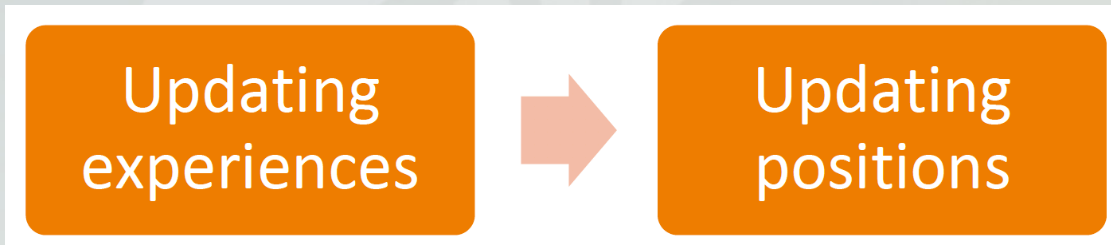
General features of Swarm Algorithms:

- Population-based
- Stochastic
- Derivative free
- Dynamic: Velocity important

Influenced through using the experiences

- Personal (inertia)
- Global (society)
- Neighbors

PSO Main Steps



- Update individual experience
- Update velocities
- Update positions

Initialization of Swarm

- Swarm size NP
- Initialize positions of individuals in search space [x_{min}, x_{max}]

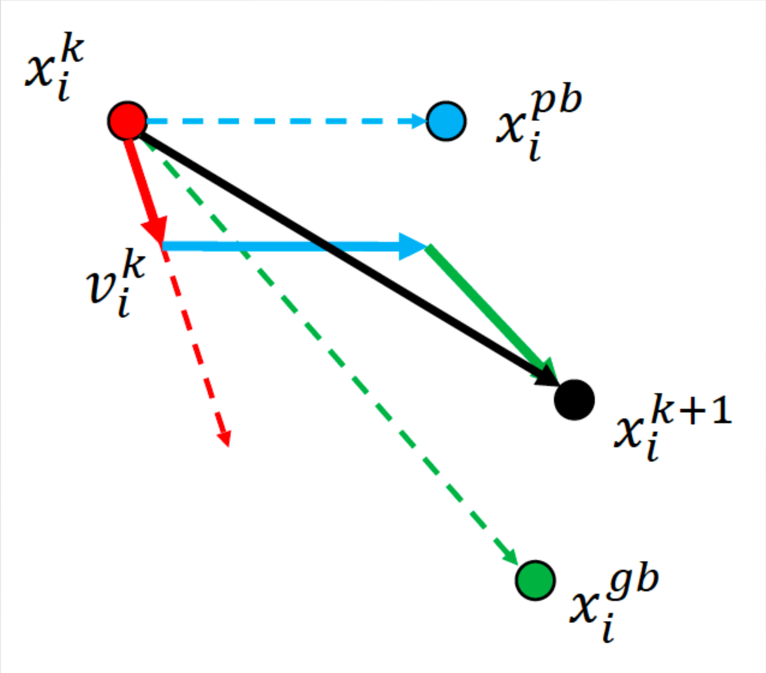
$$x_i^0 = x_{min}^0 + rand \times \left(x_{max}^0 - x_{min}^0 \right)$$
$$i \in \{1, 2, \cdots, \mathrm{NP}\}$$

- Initialize velocities

$$v_i^0 = \frac{x_i^0}{\Delta t} \quad or \quad v_i^0 = 0 \quad or \quad v_i^0 = rand$$

Basic Updating

- pb: Previous best (individual best fitness so far)
- gb: Global best (global, swarm best fitness so far)
- Sometimes, gb is replaced by lb: Local best (local in neighbourhood)



$$v_i^{k+1} = w^{k+1} v_i^k + c_1 rand_1 \frac{\left(x_i^{pb} - x_i^k\right)}{\Delta t} + c_2 rand_2 \frac{\left(x_i^{gb} - x_i^k\right)}{\Delta t}$$

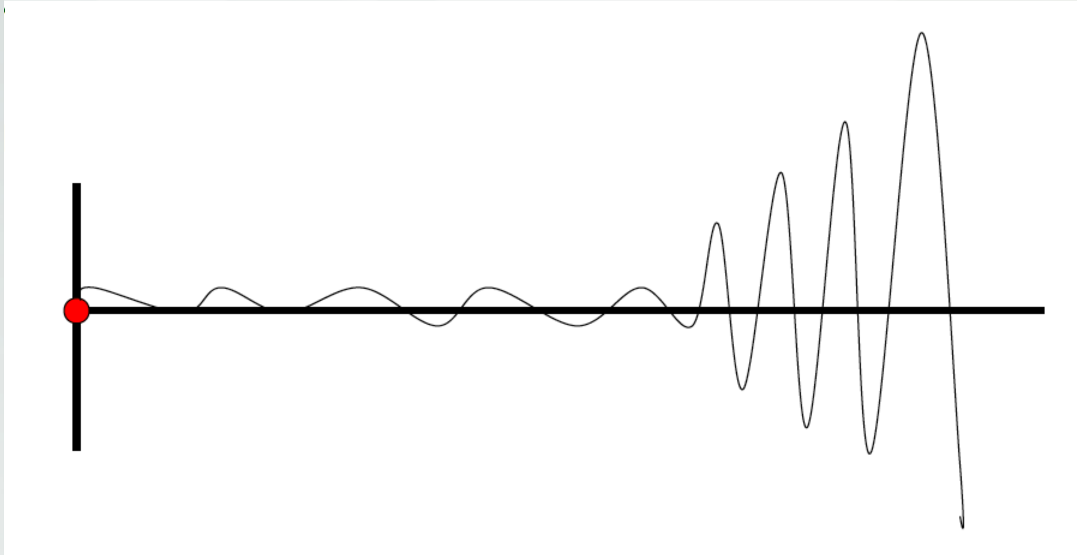
$$x_i^{k+1} = x_i^k + v_i^{k+1} \Delta t$$

Synchronous vs. Asynchronous Updates

- Synchronous updates
  - Personal best and neighbourhood best updated separately from velocity and position vectors
  - Slower feedback about best position
  - Better for gbest PSO
- Asynchronous updates
  - New best position updates after each particle position update
  - Immediate feedback about best regions of the search space
  - Better for lbest PSO

Problems

- Velocity has a tendency to explode to large values



Control Parameters

$$v_i^{k+1} = w^{k+1} v_i^k + c_1 rand_1 \frac{\left(x_i^{pb} - x_i^k\right)}{\Delta t} + c_2 rand_2 \frac{\left(x_i^{gb} - x_i^k\right)}{\Delta t}$$

- Convergence depends on parameter settings
- c_1 and c_2 control exploration vs exploitation tendency
- rand are random numbers from [0,1]
- Speed limit V_{max}

$$v_{ij}^{k+1} = \begin{cases} v_{ij}^{k+1} & \left|v_{ij}^{k+1}\right| < V_{max,j} \\ V_{max,j} & otherwise \end{cases}$$

Control Parameters II

- w is a weight for the previous velocity, therefore the inertia
  - $0 < w < 1$: Velocity decreases, leading to convergence of swarm
  - $w > 1$: Velocity increases, leading to divergence of swarm
- w decreasing over run, [0.9 …. 0.4], or constant at $w=0.7298$ with c_1 and c_2 as $c_1 = c_2 = 1.49618$ (empirical results)

Control Parameters III

- c_1 and c_2 are termed acceleration coefficients
- $c_1 > 0$, $c_2 = 0$: Independent hill climbers, local search by each particle
- $c_1 = 0$, $c_2 > 0$: Swarm is one stochastic hill climber
- $c_1 = c_2 > 0$: Particles are attracted to the average of pb and gb
- $c_1 < c_2$: More beneficial for uni-modal problems
- $c_1 > c_2$: More beneficial for multi-modal problems
- Low c_1 and c_2 : Smooth particle trajectories
- High c_1 and c_2 : More acceleration, abrupt movement of particles

Convergence

- Van den Bergh (2002) and Trelea (2003) provided formal proof that particles converge to an equilibrium
- In the limit, for gbest PSO, this is:

$$\lim_{t \to \infty} \vec{x}_i(t) = \frac{c_1 \vec{p}_i(t) + c_2 \vec{p}_g(t)}{c_1 + c_2}$$

- A single point!

Convergence II

- However, this does not mean that this weighted average between personal and global best is actually a local minimum
- Particles may prematurely converge to a stable state

- For example, what happens if $\vec{x}_i = \vec{p}_i = \vec{p}_g$??
- Then, only the inertia term $w\vec{v}_i$ contributes
- Over a number of iterations, this could mean $w\vec{v}_i \to 0$
- Add mutation!

Variants

- Without a maximum velocity, a method to restrict the update: a constriction factor

$$v_i^{k+1} = \chi\left(v_i^k + c_1 rand_1 \left(x_i^{pb} - x_i^k\right) + c_2 rand_2 \left(x_i^{gb} - x_i^k\right)\right)$$

$$\chi = \frac{2\kappa}{\left|2 - \phi - \sqrt{\phi^2 - 4\phi}\right|}$$

$$\phi = \phi_1 + \phi_2$$
$$\phi_1 = c_1 rand_1$$
$$\phi_2 = c_2 rand_2$$

- $\phi \geq 4$ and $\kappa \in [0, 1]$ guarantee convergence

Variants II

- Neighborhood (nb) wo/w global best

$$v_i^{k+1} = w^{k+1} v_i^k + c_1 rand_1 (x_i^{pb} - x_i^k) + c_2 rand_2 (x_i^{nb} - x_i^k)$$

$$v_i^{k+1} = w^{k+1} v_i^k + c_1 rand_1 (x_i^{pb} - x_i^k) + c_2 rand_2 (x_i^{nb} - x_i^k) + c_3 rand_3 (x_i^{gb} - x_i^k)$$

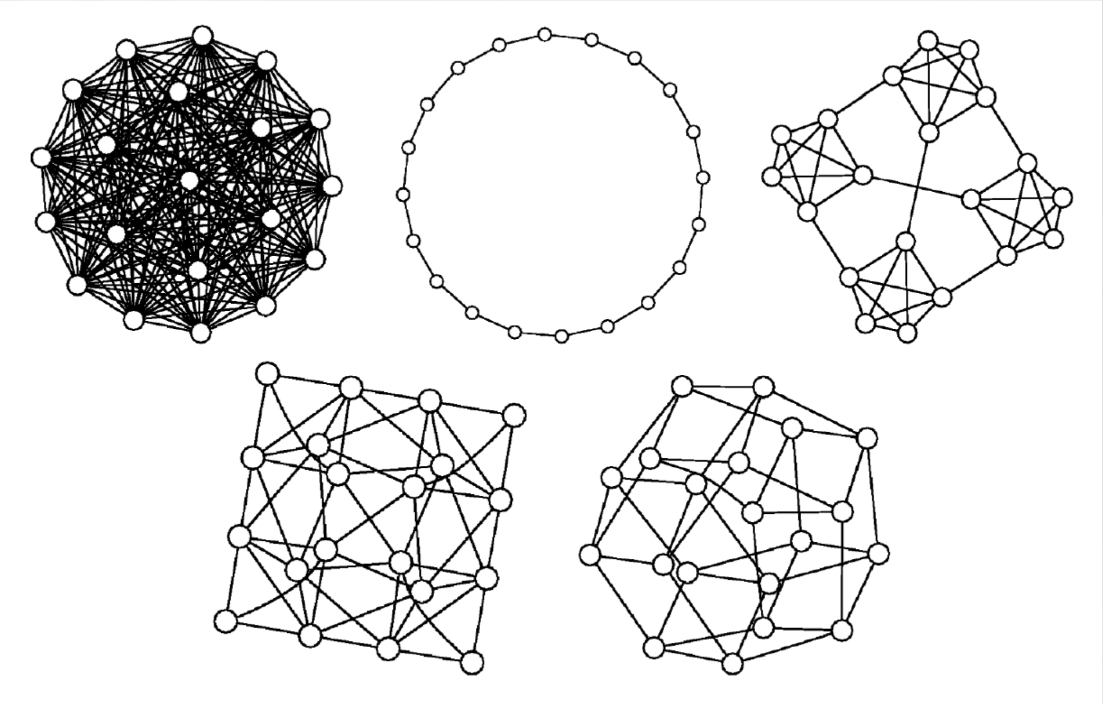
- Worst experience (global or personal)

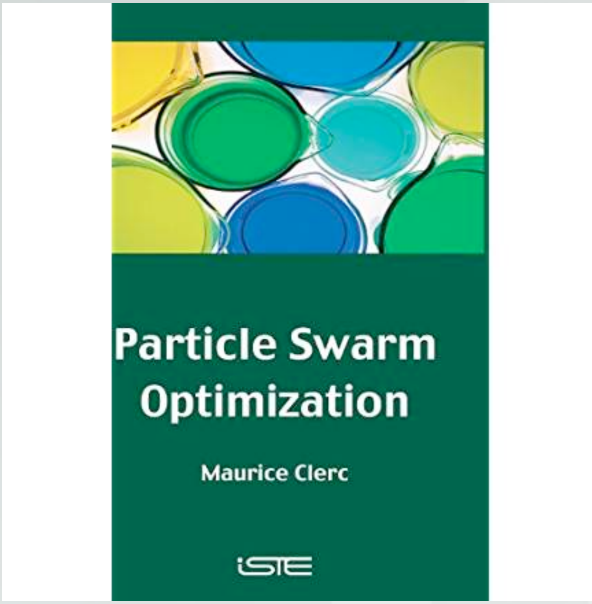
$$v_i^{k+1} = w^{k+1} v_i^k + c_1 rand_1 (x_i^{pb} - x_i^k) + c_2 rand_2 (x_i^k - x_i^{gw})$$

$$v_i^{k+1} = w^{k+1} v_i^k + c_1 rand_1 (x_i^k - x_i^{pw}) + c_2 rand_2 (x_i^{gb} - x_i^k)$$

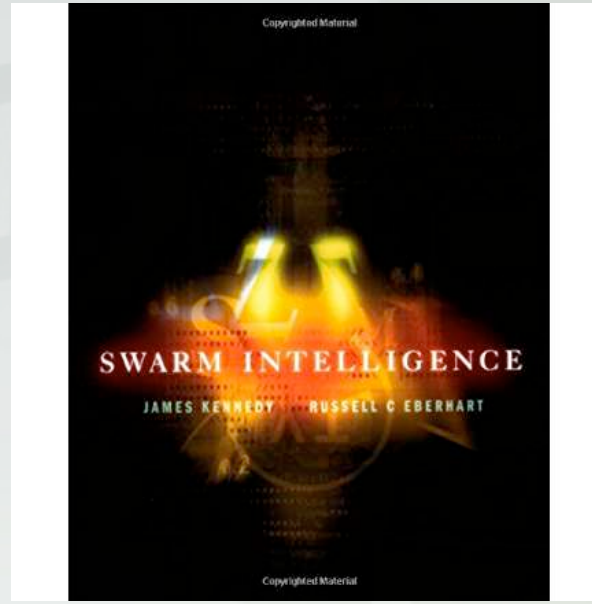
Topologies

- Different topologies to define neighbourhoods
- All
- Ring
- Four clusters
- …




Particle Swarm Optimization
Maurice Clerc
ISTE

2006


Copyrighted Material
SWARM INTELLIGENCE
JAMES KENNEDY
RUSSELL C EBERHART
Copyrighted Material

2001