Particle Swarm Optimization

- Inspired by biological and sociological motivations
  - Bird flocks
  - Fish schools
  - Swarms of insects
PSO: Characteristics

• Population-based optimization technique – originally designed for solving real-valued function optimizations
• Applicable for optimizations in rough, discontinuous and multimodal surfaces
• Does not require any gradient information of the function to be optimized
• Conceptually very simple
PSO: Characteristics

- Each candidate solution of continuous optimization problem is described (encoded) by a real vector N-dimensional search space: \( x = x_1, \ldots, x_n \).
- Each candidate solution is called PARTICLE and represents one individual of a population called SWARM.
- The particles change their components and FLY through the multi-dimensional search space.
- Particles calculate their FITNESS function as the quality of their actual position in the search space using w.r.t. the function to be optimized.
- Particles also compare themselves to their neighbors and imitate the best of that neighbors.
PSO: Fundamentals

- Swarm of particles is flying through the parameter space and searching for the optimum
- Each particle is characterized by
  - Position vector… $x_i(t)$
  - Velocity vector… $v_i(t)$
PSO: Velocity Update

- Velocity update ($i^{th}$ particle):

$$v_i(t+1) = \omega v_i(t) + C_1 \varphi_1 (pbest_i(t) - x_i(t)) + C_2 \varphi_2 (gbest(t) - x_i(t))$$

$pbest_i(t)$ … personal best experience; the best value of the fitness function found by the $i$-th particle up to time $t$.

$gbest(t)$ … global best experience; the best value out of $pbest_i(t)$ values of all particles in the swarm found up to time $t$.

$\omega$ … inertial factor

$\varphi_1$ and $\varphi_2$ … uniformly distributed random numbers that determine the influence of $pbest_i(t)$ and $gbest(t)$.

$C_1$ … particle’s self-confidence; controls the contribution towards the self-exploration.

$C_2$ … swarm confidence; controls the contribution towards the global direction.
• Velocity update ($i^{\text{th}}$ particle):

\[ v_i(t+1) = \omega v_i(t) + C_1 \varphi_1 (p_{best}(t) - x_i(t)) + C_2 \varphi_2 (g_{best}(t) - x_i(t)) \]
• Velocity update ($i^{th}$ particle):

\[ v_i(t+1) = \omega v_i(t) + C_1 \varphi_1 (p_{best}(t) - x_i(t)) + C_2 \varphi_2 (g_{best}(t) - x_i(t)) \]
PSO: Velocity Update

- Velocity update ($i^{th}$ particle):

\[ v_i(t+1) = \omega v_i(t) + C_1 \varphi_1 (p_{best}(t) - x_i(t)) + C_2 \varphi_2 (g_{best}(t) - x_i(t)) \]
PSO: Velocity Update

- Velocity update ($i^{th}$ particle):

\[ v_i(t+1) = \omega \ v_i(t) + C_1 \varphi_1 (p_{best}(t) - x_i(t)) + C_2 \varphi_2 (g_{best}(t) - x_i(t)) \]
PSO: Velocity Update

- Velocity update ($i^{th}$ particle):

$$v_i(t+1) = \omega v_i(t) + C_1 \varphi_1 (p_{best_i}(t) - x_i(t)) + C_2 \varphi_2 (g_{best}(t) - x_i(t))$$
PSO: Position Update

- Position update ($i^{th}$ particle):

$$x_i(t+1) = x_i(t) + v_i(t+1)$$
PSO: Algorithm

Input: Randomly initialized position and velocity of the particles: \( X_i(0) \) and \( V_i(0) \)

Output: Position of the approximate global optima \( X^* \)

begin
  while terminating condition is not reached do
  begin
    for i=1 to number_of_particles
    begin
      calculate fitness \( f(X_i) \)
      update \( p_i \) and \( g_i \)
      adapt velocity of the particle
      update position of the particle
      increase i
    end
  end
end
PSO: Setting the Inertia Factor $\omega$

- **Static parameter setting**
  - $\omega < 1$ … only little momentum is preserved from the previous time-step.
  - $\omega = 0$ … the particle moves in each step totally ignoring information about the past velocity.
  - $\omega > 1$ … particles can hardly change their direction which implies a reluctance against convergence towards optimum.
    $\omega > 1$ is always used with $V_{\text{max}}$ to avoid *swarm explosion*.

- **Dynamic parameter setting** – annealing scheme; $\omega$ decreases linearly with time from $\omega=0.9$ to $\omega=0.4$.

  Globally explores the search space in the beginning of the run.

  Performs local search in the end.

- $V_{\text{max}}$ can be set to the full search range of the particle’s position in order to allow global search.
PSO: Swarm Size

- Swarm size has no significant effect on the performance of the PSO. Typical values are 20-60.
PSO: Acceleration Coefficients $C_1$ and $C_2$

- **Static setting**
  Usually $C_1=C_2$ and range from $[0, 4]$, for example $C_1=C_2=1.494$.

- **Dynamic setting** - coefficients vary with time according to
  
  \[
  C_1 = (C_{1f} - C_{1i}) \frac{i}{\text{MAXITER}} + C_{1i}
  \]

  \[
  C_2 = (C_{2f} - C_{2i}) \frac{i}{\text{MAXITER}} + C_{2i}
  \]

  where $C_{1f}$ and $C_{2f}$ are final values for $C_1$ and $C_2$, $C_{1i}$ and $C_{2i}$ are current values at iteration $i$, and $\text{MAXITER}$ is the maximum number of iterations.

  Particular scheme: $C_1$ decreases from 2.5 to 0.5; $C_2$ increases from 0.5 to 2.5.

  Effect: Global search during the early phase of the optimization process; convergence to global optimum at the final stage of the optimization process.
http://www.softcomputing.net/aciis.pdf