Function value optimization with PSO

* PSO algorithm:

A particle knows the fitnesses of those in its neighborhood, and uses the position of the one with best fitness.

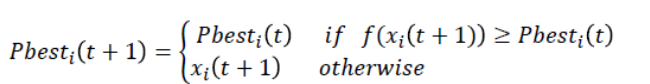
This position is simply used to adjust the particle’s velocity

**Neighborhood**

* Star topology -> communication between all
* Ring topology
* Wheel topology ->follow the leader
* The basic concept of PSO lies in accelerating each particle toward its **pbest** and the **gbest** locations, with a random weighted acceleration at each time.
* **Inertia Weight**

**

*d*  is the dimension, *c1* and *c2* are positive constants, *rand1* and *rand2* are random numbers, and *w* is the inertia weight





w = wMax-[(wMax-wMin) \* iter]/maxIter

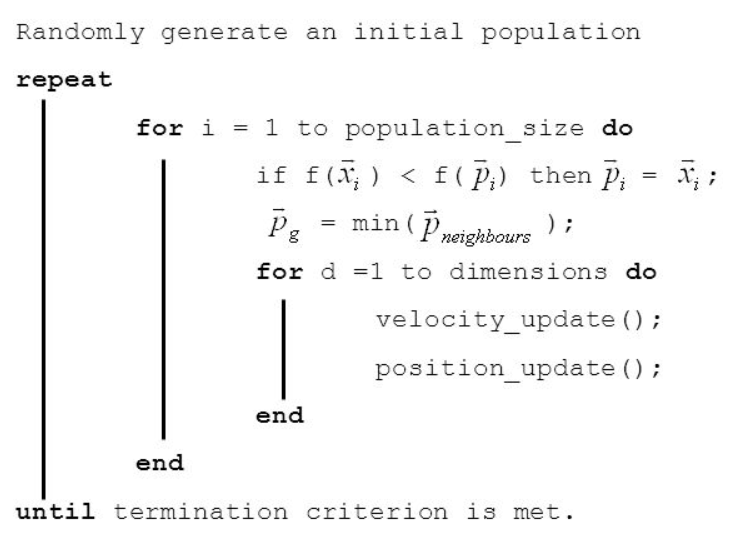
wMax= initial weight, wMin = final weight,

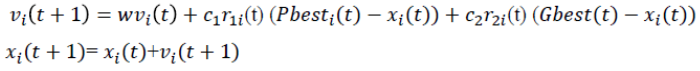
maxIter = maximum iteration number,

iter = current iteration number.

* Large inertia weight facilitates global exploration
* small on facilitates local exploitation

By linearly decreasing the inertia weight gives best performance





velocity: take positive and negative 10% of particles position as velocity.

Dimension: it is number of variables in the problem;

Number of particles: varies between 10 to 60 (normally gives good result)

It is claimed in previous research that inertia weight is used during the implementation PSO and many research work is conducted where the value is chosen as: w(initial) = 0.9 and decrease to w(final) =0.2.