Particle Swarm Optimization

The Basic Idea

We assume a set of particles moving in a hypothesis parameter space searching for an optimum set of parameters. Each particle is moving with some velocity and remembers the position where it had the best result so far. The particles in this swarm cooperate, where they exchange information about what they have discovered.

Each particle has a neighborhood associated with it. A particle knows about the fitness of those in its neighborhood, and uses the position of the best fit, to adjust the velocity of the particle.

In each timestep, a particle moves to a new position by adjusting its velocity. The velocity is updated as a weighted increment of its current position, a random portion in the direction of its personal best and a random portion in the direction of the neighborhood's best.

This is done until maximum iterations are achieved or minimum error is obtained.

The velocity of each particle in each dimension is also capped.

Describing the Problem

The problem description can be given by

1. Number of particles - swarm size
2. Importance of personal best c1
3. Importance of neighborhood best c2
4. V^{max} the limit on the velocity in each dimension

Adaptive swarm sizes can also be used where if there is not much improvement, the particle is killed. Also, if a particle has the best value so far but is not improving, it tries to generate a new particle from its position to search other alternatives. Similarly, coefficients can be adapted by seeing how the neighbours perform, where if the neighbours are better, the particle tries to shift in their direction.

Comparison of PSO and GA

Both PSO and GA are similar in the sense that they are both population-based search approaches that both depend on information sharing between population entities. PSO is more computationally efficient.