TJHSST Senior Research Project
Particle Swarm Optimization and Social Interaction Between Agents
2007-2008

Kenneth Lee

November 1, 2007

Abstract

Particle Swarm Optimization is a method of optimization for n-dimensional infinte search spaces. This project aims to test different social situations between the particles and their ability to converge on a solution. The different versions of the social interactions are tested using a benchmark function and then compared to eachother.

Keywords: Particle Swarm Optimization, Fully Informed Particle Swarm.

1 Introduction - Purpose and Scope

Particle Swarm Optimization(PSO) is a technique used to optimize n-dimensional infinite search space problems. A large amount of particles exist in the search space and "fly" though it searching for the global minimum. Particles are influenced by both cognitive and social interactions changing their course of "flight". This project aims to alter the social interactions in order to increase the effeciency of the swarm.

If a way of social interaction is proven to be more effecient than the others, then it can replace the older method, and thus make the algorithm more powerful. This could also lead to other improvements increasing the quality of algorithm.

Anyone who has a problem in which a optimal result needs to be found quickly can use this algorithm. It has seen extensive use in neural networks and finds potential in time-critical optimization problems where the constraints quickly change.

This project will deal only with the social interactions between agents. It will not deal with inertia or cognitive influences and they will remain constant thoughout the program. It should also be noted that thusfar in the project, the only form of comparing the different influences will be by iteration count not by actual time to run the program.

2 Background

PSO is a relatively new swarm intelligence technique. It was first created in 1995, inspired from flocks of birds and schools of fish. It is considered a good technique because it is both inexpensive in time and in memory. PSO is used for n-dimensional optimization problems, because it is relatively easy to implement. A set of particles is randomly created in the search space. Each particle is given a random velocity to move about the search space. Its velocity can be adjusted during the run by both cognitive and social interactions. The congnitive interactions involve the particle remembering where it had the highest fitness value, and wanting to return there. The social influences are where the particle in influenced based on the other particles, either by their current position and fitness value or their personal best(pbest) fitness value.[1]

Though different types of social interactions have been tested in the past, the conculsions have not been conclusive.[2]

3 Procedures

Procedure and Methodology

The first step for this project was to correctly recreate the basic PSO for a simple situation. This basic PSO had to include a method for the social interaction between agents. The method was then altered to include some different social intereactions. More specifically, those interactions are the Fully Informed Particle Swarm(FIPS) and the No Influence Particle Swarm(NIPS).

3.1 Testing

For this project, it would not be very possible to use a mathematical formula to judge performance for the swarm, due to the fact that a great part of the algorithm(including starting position and velocity) are derived randomly. Therefore, the program will be tested by running the program multiple times and determining the average running time and number of time steps needed for the swarm to converge on the correct answer.

I plan on using the following tools and sticking with the proposed time scale below.

3.2 Software

Currently, I am using C for the main coding, with use of the OpenGL library to graphically show 2 dimensions of the problem.

3.3 Algorithms/Programs

I'll be using the following algorithms/programs, in addition to designing my own:

1. A PSO problem that I create and altering the social interaction methods.

4 Schedule

In the first quarter, I will focus on creating a basic PSO program and learning about previous attempts at social interactions between particles and the results of those attemps.

In the second quarter, I will add to the program in order to test different social situations, possibly over different benchmark functions than only the one used thus far in the project.

In the third quarter, I will bring together both earlier parts to form a complete Particle Swarm Optimizer application. It will be able to test different types of social interactions between agents with different benchmark functions and output the results.

5 Expected Results

Expected Results and Value to Others

I expect that the FIPS will fare better than some of the other types of interaction tested, but it will not be the most effecient in terms of time due to the n^2 addition required for that method. Overall, I believe that the Single Influenced Particle Swarm will do the best overall because of its simplicity and robustness. The NIPS will do the worst due to the tendancy of its particles to reach and maintain at local extrema.

References

[1] Hirotaka Yoshida et al. A particle swarm optimization for reactive power and voltage control considering voltage sability. 1999.

[2] Rui Mendes, James Kennedy, and José Neves. The fully informed particle swarm: Simpler, maybe better. *IEEE Trans. Evolutionary Computation*, 8(3):204–210, 2004.