Particle Swarm Optimization with Oscillation Control

Javier López
III-LIDI - Fac.de Informática -UNLP
50 y 120 – La Plata (1900)
Bs.As. - Argentina
Phone: +54 221 4227707
jlopez@telefe.com.ar

Laura Lanzarini
III-LIDI - Fac.de Informática -UNLP
50 y 120 – La Plata (1900)
Bs.As. - Argentina
Phone: +54 221 4227707
laural@lidi.info.unlp.edu.ar

Armando De Giusti
III-LIDI - Fac.de Informática -UNLP
50 y 120 – La Plata (1900)
Bs.As. - Argentina
Phone: +54 221 4227707
degiusti@lidi.info.unlp.edu.ar

ABSTRACT
Particle Swarm Optimization (PSO) is a metaheuristic that has been successfully applied to linear and non-linear optimization problems in functions with discrete and continuous domains. This paper presents a new variation of this algorithm – called oscPSO – that improves the inherent search capacity of the original (canonical) version of the PSO algorithm. This version uses a deterministic local search method whose use depends on the movement patterns of the particles in each dimension of the problem. The method proposed was assessed by means of a set of complex test functions, and the performance of this version was compared with that of the original version of the PSO algorithm. In all cases, the oscPSO variation equaled or surpassed the performance of the canonical version of the algorithm.

Categories and Subject Descriptors
I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *Heuristic methods.*

General Terms
Performance.

Keywords
Evolutionary Computing, Particle Swarm Optimization, Function Optimization, Oscillation Detector, Local Search

1. INTRODUCTION
Problem optimization is a very frequent process in real life. We constantly face the need of finding good solutions to any given problem. Particle Swarm Optimization (PSO) is a metaheuristic proposed by Kennedy and Eberhart [2]. Its operation is based on the simulation of simple social models, and it has been successfully used in function optimization as well as in neural network training [1] [3].

In this paper, a new version, called oscPSO, is presented. This version adds a local search method that is automatically enabled based on particle movement characteristics.

2. OBJECTIVE
PSO is an optimization technique that has proven to be easily implementable, stable, and scalable, and which has yielded good results in function optimization [6]. However, the exploration strategy of the PSO algorithm used around local optimums is inefficient, since it tends to present oscillation around these optimums in repeated occasions [4]. Taking this last issue into account, oscPSO changes the strategy to approach local optimums when oscillation is detected around them. To achieve this objective, a deterministic search procedure that minimizes the number of steps required to reach optimal points is used.

GECCO'09, July 8–12, 2009, Montréal, Québec, Canada.
ACM 978-1-60558-325-9/09/07.

3. OSCILLATION CONTROL
In PSO, each individual represents a possible solution to the problem and adapts based on three factors: its knowledge of the environment (fitness value), its historical knowledge or previous experiences (memory), and the historical knowledge or previous experiences of the individuals in its neighborhood [2]. Its purpose is to make its knowledge evolve so as to resemble the most successful individuals within its environment.

The purpose of oscPSO is to minimize the oscillations in the movement of each particle around an optimum, thus obtaining a more efficient approach strategy. Therefore, with oscPSO, any given particle will be able to use two different strategies to modify the dimensions of its velocity vector. The dimensions that are in an oscillation state will use the local search process described in Section 3.2, whereas those dimensions that are not in an oscillation state will follow the conventional PSO algorithm.

3.1 Oscillation Detection
A particle dimension will be considered to be in an oscillation state if there is evidence of a behavior that alternates successive forward and backward movements. That is, for each dimension of any given particle, the sign of the resulting speed corresponding to each iteration will be analyzed. When a sequence of opposite speed signs is detected in successive iterations, the particle is considered, *for that specific dimension*, to be in an *oscillation state.* Figure 1 illustrates this situation.

3.2 Local search procedure
This procedure is used to update the velocity vector of a given particle in the dimensions that are in an oscillation state. It requires that the last N positions of the particle be stored within the solution space together with their corresponding fitness values. Within the memorized locations for this particle, the mid point between both locations with greatest fitness values and opposite speed sign for the dimension is calculated. This procedure is repeated for successive iterations until a location whose fitness value is worse than the previous one is found. The operation of the *Local Search Procedure* is shown in Figure 2.

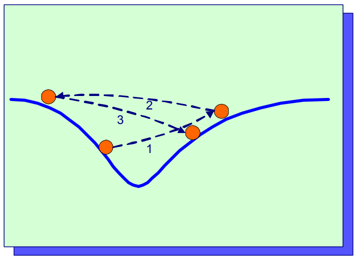

Figure 1. Oscillation Detector

It should be noted that this procedure is run independently by each dimension. Considering that the *Oscillation Detector* is also independent for each dimension, in an iteration of the same algorithm, it could happen that the speeds of some of the dimensions of a particle are updated based on the standard formula of the PSO algorithm, whereas other dimensions of that same particle, in the same iteration, are updated using this *Deterministic Local Search Procedure*.

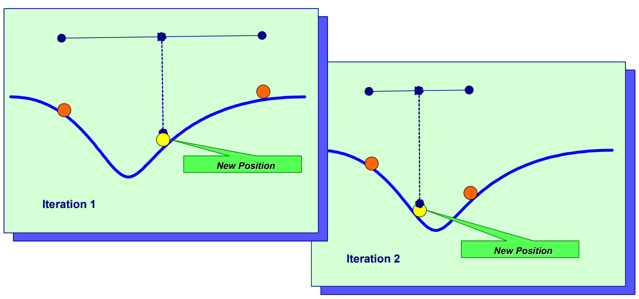

Figure 2. Local Search Procedure

4. PARAMETER ADAPTATION

The incorporation of both procedures requires configuring new parameters which are not present in the original version of the algorithm. In order not to increase complexity, a set of good performance parameters was selected, particles were initialized with different parameter values, and a self-adaptation process (Binary Tournament) was incorporated based on the success rate of the *Deterministic Local Search Procedure*.

5. RESULTS

Table 1 shows the performance of oscPSO in ten test functions [5]. As it can be gathered from the summary chart, the performance of the oscPSO algorithm was better than that of the original PSO version. For 5 out of the 10 test problems, it yielded a better result. For the remaining 5 problems, it was not possible to determine which of the two algorithms had the best performance.

6. CONCLUSIONS

This article presents a new variation of the PSO algorithm – called oscPSO – that allows improving search processes, particularly during the final approximation phase of the particles to the optimal values. This final approximation process was found to be inefficient in the original version of the PSO algorithm, since the canonical version uses too many iterations to achieve the desired approximation.

Table 1: Average values of 25 independent runs.

Function	PSO		oscPSO	
	Average	Deviation	Average	Deviation
Shifted Sphere	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
Shifted Schwefel's Problem	3.9327E-06	4.4077E-06	**6.0800E-12**	**2.9361E-11**
Shifted Rotated High Conditioned Elliptic	7.4384E+05	2.3456E+05	**3.3724E+05**	**1.3475E+05**
Shifted Schwefel's Problem 1.2 with Noise in Fitness	6.3575E-01	5.3254E-01	8.4022E-01	8.7679E-01
Schwefel's Problem 2.6 with Global Optimum on Bounds	2.4861E+03	4.6604E+02	2.6997E+03	4.2652E+02
Shifted Rosenbrock's	7.0806E+01	6.5885E+01	**7.4804E+00**	**1.6092E+01**
Shifted Rotated Griewank's without Bounds	1.3296E-02	1.0803E+01	1.2017E-02	1.1374E-02
Shifted Rotated Ackley's Global Optimum on Bounds	2.0953E+01	4.3397E-02	2.0932E+01	6.3025E+00
Shifted Rastrigin's	2.4675E+01	5.6866E+00	**2.0973E+01**	**6.2460E+00**
Shifted Rotated Rastrigin's	1.2545E+02	6.2869E+01	**8.7198E+01**	**4.7837E+01**

Bearing this in mind, there are two major issues to be solved: 1) Determining when the algorithm is oscillating around some local (global) optimum; 2) Developing a procedure that increases the efficiency of the final approximation of the particle to the optimum. In order to solve the first issue, an *Oscillation Detector* was incorporated to analyze sign variations in the speed component of each particle. In the case of the second issue, a *Deterministic Local Search Procedure* was incorporated. The purpose of this procedure is finding the location of the optimum in the smallest number of iterations possible.

Results show that the objectives were reached, and that the behavior of this new version of the PSO algorithm is better than that of the original version.

7. REFERENCES

[1] Hu X., Eberhart X., Shi Y. Recent Advances in Particle Swarm, *IEEE Congress on Evolutionary Computation 2004*, Portland, Oregon, USA.2004

[2] Kennedy J. , Eberhart R. C. Particle Swarm Optimization. *IEEE International Conference on Neural Networks*, vol. IV, (Perth, Australia), pp. 1942–1948, IEEE Service Center 1995

[3] Meissner, M., Schmuker, M., Schneider, G. Optimized Particle Swarm Optimization (OPSO) and its Application to Artificial Neural Networks Training. *In: BMC Bioinformatics 2006 (Published online 2006 March 10)* pp. 7–125 (2006)

[4] Monson C. No Free Lunch, Bayesian Inference, and Utility: A Decision-Theoretic Approach to Optimization. *Ph.D. dissertation*. Department of Computer Science. Brigham Young University. August 2006

[5] Suganthan P., Hansen N., Liang J., Deb K., Chen Y., Auger A. , Tiwari S. Problem Definitions and Evaluation Criteria for de CEC 2005 Special Session on Real-Parameter Optimization. *CEC-05*, Edinburgh, UK, 2-5 Sept. 2005

[6] Van den Bergh F. An Analysis of Particle Swarm Optimizers. *Ph.D. dissertation*. Department Computer Science. University Pretoria. South Africa. 2002