



A dynamic neighborhood learning based particle swarm optimizer for global numerical optimization

Md Nasir^{a,c}, Swagatam Das^{b,c,*}, Dipankar Maity^{a,c}, Soumyadip Sengupta^{a,c},
Udit Halder^{a,c}, P.N. Suganthan^{b,c}

^a Dept. of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700 032, India

^b Electronics and Communication Sciences Unit, Indian Statistical Institute, Kolkata 700 108, India

^c School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore

ARTICLE INFO

Article history:

Received 28 July 2011

Received in revised form 7 March 2012

Accepted 4 April 2012

Available online 6 May 2012

Keywords:

Particle swarm

Neighborhood

Exemplar

Learning strategy

ABSTRACT

The concept of particle swarms originated from the simulation of the social behavior commonly observed in animal kingdom and evolved into a very simple but efficient technique for optimization in recent past. Since its advent in 1995, the Particle Swarm Optimization (PSO) algorithm has attracted the attention of a lot of researchers all over the world resulting into a huge number of variants of the basic algorithm as well as many parameter selection/control strategies. PSO relies on the learning strategy of the individuals to guide its search direction. Traditionally, each particle utilizes its historical best experience as well as the global best experience of the whole swarm through linear summation. The Comprehensive Learning PSO (CLPSO) was proposed as a powerful variant of PSO that enhances the diversity of the population by encouraging each particle to learn from different particles on different dimensions, in the metaphor that the best particle, despite having the highest fitness, does not always offer a better value in every dimension. This paper presents a variant of single-objective PSO called Dynamic Neighborhood Learning Particle Swarm Optimizer (DNLPSO), which uses learning strategy whereby all other particles' historical best information is used to update a particle's velocity as in CLPSO. But in contrast to CLPSO, in DNLPSO, the exemplar particle is selected from a neighborhood. This strategy enables the learner particle to learn from the historical information of its neighborhood or sometimes from that of its own. Moreover, the neighborhoods are made dynamic in nature i.e. they are reformed after certain intervals. This helps the diversity of the swarm to be preserved in order to discourage premature convergence. Experiments were conducted on 16 numerical benchmarks in 10, 30 and 50 dimensions, a set of five constrained benchmarks and also on a practical engineering optimization problem concerning the spread-spectrum radar poly-phase code design. The results demonstrate very competitive performance of DNLPSO while locating the global optimum on complicated and multimodal fitness landscapes when compared with five other recent variants of PSO.

© 2012 Published by Elsevier Inc.

1. Introduction

In its simplest form, an unconstrained D -dimensional real-parameter optimization problem can be formulated as a search for the optimal parameter vector $\vec{X}^*$, which minimizes an objective function $F(\vec{X}) (F: \Omega \subseteq \Re^D \rightarrow \Re)$ i.e. $F(\vec{X}^*) < F(\vec{X})$ for all

* Corresponding author at: Electronics and Communication Sciences Unit, Indian Statistical Institute, Kolkata 700 108, India.

E-mail addresses: nasir795@gmail.com (M. Nasir), swagatam.das@isical.ac.in (S. Das), dipankarmaity1991@gmail.com (D. Maity), senguptajuetce@gmail.com (S. Sengupta), udithalder99@gmail.com (U. Halder), epnsugan@ntu.edu.sg (P.N. Suganthan).

$\vec{X} \in \Omega$, where Ω is a non-empty large finite set serving as the domain of the search. For an unconstrained optimization problem, $\Omega \in \mathbb{R}^D$. Since $\max\{F(\vec{X})\} = -\min\{-F(\vec{X})\}$, the restriction to minimization is without loss of generality. In general, the optimization task is complicated by the existence of non-linear objective functions with multiple local minima. A local minimum $F_\ell = F(\vec{X}_\ell)$ may be defined as: $\exists \varepsilon > 0 \forall \vec{X} \in \Omega : \|\vec{X} - \vec{X}_\ell\| < \varepsilon \Rightarrow F_\ell \leq F(\vec{X})$, where $\|\cdot\|$ indicates any p -norm distance measure.

Real-parameter optimization problems are rife in different fields of engineering, social and physical sciences and many of them pose severe challenge to the classical derivative-based techniques. This fact has led the researchers to develop various optimization techniques founded on the simulation of natural phenomena. Some of the real parameter optimizers that are in wide use these days are Genetic Algorithm (GA) [14], Simulated Annealing (SA) [21], Tabu Search (TS) [13], Covariance Matrix Adaptation Evolution Strategies (CMA-ESs) [15], Teaching–Learning–Based Optimization [32,33], Harmony Search [38], Immune Algorithm [40], Differential Evolution (DE) [5,43], and Particle Swarm Optimization (PSO) [8]. PSO [2,8,10,20] marks one of the most popular classes of nature-inspired optimizers and has its roots in artificial life and social psychology. In fact, it emulates the flocking behavior of birds and fish schooling. PSO has become very popular these days as an efficient algorithm for intelligent search and optimization. It does not require any gradient information of the function to be optimized, uses only primitive mathematical operators, and is conceptually very simple. Since its inception in 1995, PSO has attracted a great deal of attention of the researchers all over the globe resulting into nearly uncountable number of variants of the basic algorithm, theoretical and empirical investigations of the dynamics of the particles, parameter selection and control, and applications of the algorithm to a wide spectrum of real world problems from diverse fields of science and engineering. For a comprehensive knowledge on the foundations, perspectives, and applications of PSO, see [1,3,6,9,18,22,37].

PSO is, however, not free from premature convergence, especially over multimodal, rugged, and non-separable fitness landscapes. Several PSO-variants were proposed by the researchers over the past decade to circumvent these problems and make PSO more efficient as a black-box optimizer over the continuous search space for real parameter optimization problems. Some of the most significant variants of PSO that attracted much interest from the researchers in recent past are Comprehensive Learning PSO (CLPSO) [24], Dynamic Multi-Swarm PSO (DMSPSO) [23], Fully Informed Particle Swarm (FIPS) [26], Unified PSO (UPSO) [30], Adaptive PSO (APSO) [42], scale-free FIPS [45], Orthogonal Learning PSO (OLPSO) [41], etc. In addition, there are several hybrid PSO variants, which incorporate tested methods of other algorithms with PSO, such as GA-PSO [29], Evolutionary PSO (EPSO) [27], DE-PSO [34], and C-PSO [16].

In this paper, we present an improved variant of the CLPSO algorithm, called Dynamic Neighborhood Learning PSO (DNLPSO), for the purpose of solving single-objective optimization problems. CLPSO uses a novel learning strategy whereby all other particles' historical best information is used to update a particle's velocity. In order to achieve a better balance of the explorative and exploitative behaviors of CLPSO, we incorporated a few novel strategies regarding the selection of the exemplar particles that contribute to the velocity updating of the other learning particles in the swarm. By the consistent updating of their velocities in course of the algorithm, all the particles tend to reach better solutions and eventually converge to an optimal solution. The learning strategies which we have incorporated in the proposed algorithm are as follows:

- (1) The learner particle learns not only from its own experience, but also from the experience of the particles in the swarm. However, unlike CLPSO, the exemplar particles are not chosen randomly, but from a predefined neighborhood, thus making the particles diverse enough for yielding good results.
- (2) The neighborhood of the learner particles have been made dynamic in nature i.e. the entire swarm is regrouped into a number of neighborhoods after certain intervals. This enhances the explorative nature of the algorithm and thus helps the particles to come out from local optima in case of premature convergence.

Rest of the article is organized in the following way: Section 2 provides an overview of the PSO family of algorithms and also discusses the CLPSO algorithm. In Section 3, the proposed DNLPSO is discussed in sufficient details. Section 4 provides a theoretical analysis of the explorative power of the algorithm justifies the motivation of the method. Effects of control parameters on the performance of DNLPSO are presented in Section 5. The experimental results on numerical unconstrained and constrained benchmarks as well as a practical engineering optimization problem are presented and discussed in Section 6. Finally Section 7 concludes the paper.

2. An overview of classical PSO and CLPSO

In PSO, each solution vector is known as a particle and several such particles collectively form a swarm. Each member in the swarm adapts its search patterns by learning from its own experience as well as other particles. The particle has a tendency to move towards a better search area with a definite velocity determined by the information collected by the particle over the course of the search process. While searching in a D -dimensional hyperspace, each particle i has a velocity vector $V_i = [v_i^1, v_i^2, \dots, v_i^D]$ and a position vector $X_i = [x_i^1, x_i^2, \dots, x_i^D]$ to indicate its current state where i is a positive integer indexing the particle in the swarm and D is the dimension of the problem under study. Moreover, particle i will keep its personal historical best position vector $pbest_i = [pbest_i^1, pbest_i^2, \dots, pbest_i^D]$. The best position of all the particles is $gbest = [gbest^1, gbest^2, \dots, gbest^D]$. The vectors V_i and X_i initialized randomly and are updated by using the following formulae:

Download English Version:

<https://daneshyari.com/en/article/394078>

Download Persian Version:

<https://daneshyari.com/article/394078>

[Daneshyari.com](https://daneshyari.com)