Neurocomputing 129 (2014) 315–333

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

An improved particle swarm optimizer with difference mean based perturbation

Rupam Kundu [a], Swagatam Das [b,*], Rohan Mukherjee [a], Shantanab Debchoudhury [a]

[a] Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700032, India
[b] Electronics and Communications Sciences Unit, Indian Statistical Institute, Kolkata 700108, India

ARTICLE INFO

Article history:
Received 27 May 2013
Accepted 16 September 2013
Communicated by A. Konar
Available online 12 December 2013

Keywords:
Particle swarm optimization
Difference mean based perturbation
Aging guideline
Time varying parameters
Swarm intelligence

ABSTRACT

Concept of the particle swarms emerged from a simulation of the collective behavior of social creatures and gradually evolved into a powerful global optimization technique, now well-known as the Particle Swarm Optimization (PSO). PSO is arguably one of the most popular nature-inspired algorithms for real parameter optimization at present. The very basic PSO model does not ensure convergence to an optimal solution and it also suffers from its dependency on external parameters like acceleration parameters and inertia weight. Owing to its comparatively poor efficiency, a multitude of measures has been taken by the researchers to improve the performance of PSO. This paper presents a scheme to modify the very basic framework of PSO by the introduction of a novel dimensional mean based perturbation strategy, a simple aging guideline, and a set of nonlinearly time-varying acceleration coefficients to achieve a better tradeoff between explorative and exploitative tendencies and thus to avoid premature convergence on multimodal fitness landscapes. The aging guideline is used to introduce fresh solutions in the swarm when particles show no further improvement. A systematically rendered comparison between the proposed PSO framework and several other state-of-the-art PSO-variants as well as evolutionary algorithms on a test-suite comprising 16 standard numerical benchmarks and two real world problems indicates that the proposed algorithm can enjoy a statistically superior performance on a wide variety of problems.

1. Introduction

Kennedy and Eberhart [1,2] introduced the concept of function-optimization by means of a particle swarm in 1995. Currently the basic PSO and its variants constitute one of the most well-known families of global optimizers over real parameter space. In PSO, each trial solution is modelled as a particle and several such particles collectively form a swarm. Particles fly through the multi-dimensional search space following a typical dynamics in search of the global optima. At any particular instance, each particle has a position and a velocity. At the beginning, a population of particles is initialized with random position-vectors and random velocities. Each particle in the swarm adapts its search pattern by learning from its own experience as well as other particles. A particle has the tendency to move towards a better search area with a definite velocity determined by the information collected by its own self and the other members of the swarm over the course of the search process. PSO does not require any derivative information of the function to be optimized, uses only rudimentary 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, the readers are directed to see references [3–9].

Being a stochastic search process, PSO is not free from false and/or premature convergence, especially over multimodal fitness landscapes. As there is a direct link of the information flow between particles and the globally best member of the swarm, multifariousness is lost. For example, a globally best particle located at one of the local optima may trap the whole swarm and leads to premature convergence. Various modifications and PSO variants have been proposed to eradicate this problem. The modifications can be regarded as algorithmic components that provide an improved performance. These approaches include tuning the control parameters so as to achieve a better exploration/exploitation trade-off [10–12], designing various proximity

* Corresponding author.
E-mail addresses: rupam2422@gmail.com (R. Kundu), swagatam.das@isical.ac.in (S. Das), rohan.mukherjii@gmail.com (R. Mukherjee), sdch10@gmail.com (S. Debchoudhury).

0925-2312/$ - see front matter
http://dx.doi.org/10.1016/j.neucom.2013.09.026

topologies to replace the traditional global topology [13–17], using multiswarm techniques [18–20], and hybridizing auxiliary search methods with PSO [21–26]. Recent works include a quantum-behaved particle swarm optimization with Gaussian distributed local attractor point [78], median-oriented Particle Swarm Optimization (MPSO) to execute global search with accelerating convergence speed and avoiding local basins of attraction [79], a fitness evaluation strategy integrated with particle swarm optimization [80], a novel particle swarm optimization algorithm emerging from the concept of ecological population evolution, called the ecological particle swarm optimization [81], even locating multiple optima using particle swarm optimization [82], and using a discrete variant of PSO [33] etc. However, most such variants preserve the population diversity at the cost of slow convergence or complicated algorithmic structures. Avoiding premature convergence without significantly reducing the convergence speed and without making the algorithmic structure too complicated still remains a challenge to the PSO researchers. The findings reported in this paper can be considered as a humble attempt to contribute in this context.

This paper proposes a simple, dimensional mean based perturbation scheme, which when integrated with the classical PSO, results into an improved variant of the algorithm, referred to here as DMP-PSO (Difference Mean based Perturbation in PSO). Under this scheme, in each iteration and after updating of velocity and position of each particle, the particles are further perturbed with a vector formed by scaling a unit vector along any random direction with the difference of the dimensional means of the current best particle and the target particle to be perturbed. The current best particle may be the best among the swarm for the global best or gbest PSO topology or a neighborhood best of the target particle for the local best or lbest topology. We will explain the lbest and gbest topologies in detail in Section 2. The difference between the dimensional mean of the current best and that of any other particle, referred to as the difference mean, implants within the algorithm a blend of explorative and exploitative capabilities to ensure effective search of the neighborhood of the current best. An aging mechanism takes care of the unwanted stagnation and improves search efficiency. In this paper, both DMP-PSO-G (global topology) and DMP-PSO-L (local neighborhood topology) frameworks have been investigated. In addition the acceleration coefficients of PSO are made to vary with time in a nonlinear fashion, as this scheme appeared to improve the performance of PSO harmoniously with the DMP scheme on the tested benchmarks. The DMP-PSO is tested on a set of 16 standard benchmark functions with various characteristic features. The results, thus obtained, have been shown to be statistically better than those obtained with 11 other classical and state-of-the-art PSO-variants which successfully assert the effectiveness of our suggested modifications. Further the proposed algorithm is validated on two engineering optimization problems taken from the competition on testing Evolutionary Algorithms (EAs) on real world problems [27] under the 2011 IEEE Congress on Evolutionary Computation (CEC).

Organization of the paper is in order. Section 2 provides a precise description of the basic PSO framework and also briefly reviews the developments on the algorithmic research with PSOs. Section 3 presents the proposed algorithm in sufficient details with proper justification for the added features. Experimental setup and results are discussed in Section 4 which includes the parameter settings with empirical analyses, description of the functions used, comparison table, scalability of the proposed approach, changes in other algorithms using the added features proposed here, and lastly application of DMP-PSO on selected real world optimization problems. Finally Section 5 concludes the paper and unearths some interesting future avenues of research.

2. Particle swarm optimization—an overview

PSO, as the name suggests, uses a swarm of particles each of which model a trial solution of the problem at hand. A particle is characterized by its position vector $\overrightarrow{x_i} = x_i^1, x_i^2, \ldots, x_i^D$, basically a coordinate in the D-dimensional frame, and its velocity vector $\overrightarrow{v_i} = v_i^1, v_i^2, \ldots, v_i^D$. The velocity and position of any particle are updated in the following way:

$$v_i^d \leftarrow v_i^d + c_1 * rand1_i^d * (pbest_i^d - x_i^d) + c_2 * rand2_i^d * (gbest_i^d - x_i^d), \tag{1}$$

$$x_i^d \leftarrow x_i^d + v_i^d, \tag{2}$$

where x_i^d and v_i^d respectively represent the d-th component of the position and velocity of the i-th particle. The best position of the particle, i.e. the position at which the i-th particle yields its best fitness value is termed $\overrightarrow{pbest}_i$ where $\overrightarrow{pbest}_i = [pbest_i^1, pbest_i^2, \ldots, pbest_i^D]$. Using similar terminology the global best position of the i-th particle is given by $\overrightarrow{gbest}_i = [gbest_i^1, gbest_i^2, \ldots, gbest_i^D]$. Note that for the *gbest* PSO topology $\overrightarrow{gbest}_i$ denotes the best position found so far in the entire swarm. On the other hand for *lbest* PSO model, $\overrightarrow{gbest}_i$ denotes the best position found by some particle in the neighborhood of the i-th particle. Neighborhoods are defined by some topological structure, such as the ring structure, the pyramid structure, or the von Neumann structure [15,16]. Without loss of generality, our method can be integrated with both the *gbest* PSO and *lbest* PSOs with any proximity topology.

The constants c_1 and c_2 scale the attraction of the particle towards the personal best and global/neighborhood best positions. These are often called acceleration coefficients [11]. Venter and Sobieski [28] termed c_1 as self-confidence and c_2 as swarm confidence. $rand1_i^d$ and $rand2_i^d$ are uniformly distributed random numbers bound within the range [0,1] and are instantiated freshly for each ordered pair (d, i). A particles velocity may be optionally clamped to a maximum denoted as $v_{max} = [v_{max}^1, v_{max}^2, \ldots, v_{max}^D]$. If the velocity exceeds this limiting condition, appropriate steps are taken to reduce it. Reassignment of the velocity component to a value of sign$(|v_i^d|)v_{max}^d$ remains a popular means of limitation [8].

Parameter control and adaptation have attracted the PSO researchers for a long time. The inertia weight ω was first introduced by Shi and Eberhart [29] in 1998 to dampen the inertial velocity from previous iteration and thus to influence convergence. Their work indicates that while larger inertia weight is better for global search, a smaller one may increase the ability of local refinement. Since then, almost all the PSO-variants have used the inertia weight as an integral part of the velocity update rule. To balance the local and global search abilities, Shi and Eberhart [10] put forth a scheme to decrease ω linearly with number of iterations in the following way:

$$\omega = \omega_{max} - (\omega_{max} - \omega_{min}) * \frac{g}{G}, \tag{3}$$

where g is the iteration index and G is a predefined maximum number of iterations. ω_{max} and $\omega_{max}\omega_{max}$ are usually set as 0.9 and 0.4 respectively [10]. Thus, the use of time varying inertia weight ω provides the necessary balance between the local and global search abilities of a particle. A fuzzy rule based nonlinear adaptation of ω was proposed in [30]. In context to the dynamic system optimization a significant modification of inertia weight as $\omega = 0.5 + random(0,1) * 0.5$ was successfully experimented in [31]. Logarithmic and exponential decreases of the inertia weight have been investigated in [31,32]. Quande et al. presented a novel PSO with a piecewise-varied inertia weight. The piecewise function, thus chosen, has two parts: one is nonlinear decreasing to enhance the explorative ability, while the other is linear decreasing just as