Applied Soft Computing 18 (2014) 39–58

Contents lists available at ScienceDirect

Applied Soft Computing

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

Teaching and peer-learning particle swarm optimization

Wei Hong Lim, Nor Ashidi Mat Isa *

Imaging and Intelligent System Research Team (ISRT), School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia

ARTICLE INFO

Article history:
Received 13 March 2013
Received in revised form
30 September 2013
Accepted 15 January 2014
Available online 24 January 2014

Keywords:
Teaching and peer-learning particle swarm optimization (TPLPSO)
Teaching–learning-based optimization (TLBO)
Metaheuristic search
Global optimization

ABSTRACT

Most of the recent proposed particle swarm optimization (PSO) algorithms do not offer the alternative learning strategies when the particles fail to improve their fitness during the searching process. Motivated by this fact, we improve the cutting edge teaching–learning-based optimization (TLBO) algorithm and adapt the enhanced framework into the PSO, thereby develop a teaching and peer-learning PSO (TPLPSO) algorithm. To be specific, the TPLPSO adopts two learning phases, namely the teaching and peer-learning phases. The particle firstly enters into the teaching phase and updates its velocity based on its historical best and the global best information. Particle that fails to improve its fitness in the teaching phase then enters into the peer-learning phase, where an exemplar is selected as the guidance particle. Additionally, a stagnation prevention strategy (SPS) is employed to alleviate the premature convergence issue. The proposed TPLPSO is extensively evaluated on 20 benchmark problems with different features, as well as one real-world problem. Experimental results reveal that the TPLPSO exhibits competitive performances when compared with ten other PSO variants and seven state-of-the-art metaheuristic search algorithms.

1. Introduction

Particle swarm optimization (PSO) algorithm is initially introduced by Kennedy and Eberhart in 1995 [1], to emulate the collaborative behavior of bird flocking and fish schooling in searching for foods [1–4]. In PSO, each individual (namely particle) represents the potential solution of the optimization problem, while the location of food source is the global optimum solution. Being a population-based metaheuristic search (MS) algorithm, PSO simultaneously evaluates many points in the search space. Besides searching for the food independently and stochastically, each particle collaborates and shares information with each other, to ensure all of them move toward the optimal solution of the problem and eventually leads to the convergence [2,3]. Since the introduction of the PSO, it becomes an overwhelming choice of optimization technique due to its simplistic implementation and excellent performance on various benchmark and engineering design problems [4–9].

Despite having the competitive performance, PSO has some undesirable dynamical properties that degrade its searching ability. One of the most important issues is the premature convergence, where the particles tend to be trapped in the local optima solution, due to the rapid convergence and diversity loss of the swarm [10]. Another issue is regarding the ability of PSO to balance the exploration/exploitation search. Overemphasize of the exploration prevents the swarm convergence, while too much exploitation has high tendency to cause the premature convergence of swarm [11].

Although extensive amounts of works [11–29] are reported to address the aforementioned issues, most of the current existing PSO variants do not provide the alternative learning strategies to particles when they fail to update their fitness during the searching process. This problem inevitably limits the algorithms' searching capabilities. Recently, Rao et al. [30,31] proposed a teaching–learning-based optimization (TLBO) algorithm, inspired by the philosophy of teaching and learning. The process of TLBO is divided into two parts, namely the teacher phase and the learner phase, where the individuals can learn from the teacher and the interaction of other individuals, respectively. Motivated by these two facts, we propose a teaching and peer-learning PSO (TPLPSO). To be specific, we improve the current existing TLBO framework and adapt this enhanced framework into the PSO. Similar with TLBO, the TPLPSO adapts two learning phase, namely the teaching and peer-learning phases. Each particle first enters into the teaching phase and updates its velocity according to its historical best and the global best information. Particle that fails to improve its fitness in the teaching phases then enters into the peer-learning phase, where an exemplar particle is selected as the guide for the particle to search for a better solution. The roulette

* Corresponding author. Tel.: +60 45996093; fax: +60 45941023.
E-mail addresses: limweihong87@yahoo.com (W.H. Lim), ashidi@eng.usm.my (N.A. Mat Isa).

1568-4946/$ – see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2014.01.009

wheel selection technique is employed to ensure fitter particle has higher probability to be selected as the exemplar, thereby provide a more promising searching direction toward the global optimum. To resolve the premature convergence issue, we employ a stagnation prevention strategy (SPS) module that will be triggered when the PSO swarm fails to improve the global best fitness in m successive function evaluations (FEs).

The remainder of this paper is organized as follows. Section 2 briefly presents some related works. Section 3 details out the methodologies of the TPLPSO. Section 4 provides the experimental settings and results, respectively. Finally, Section 5 concludes the work done.

2. Related works

In this section, we discuss the mechanism of the basic PSO. Next, the state-of-art PSO variants are reviewed. For self-completeness purpose, we also provide the brief description of TLBO.

2.1. Basic particle swarm optimization (PSO) algorithm

In the basic PSO, the PSO swarm consists of a group of particles with negligible mass and volume that roam through the D-dimensional problem hyperspace. Each particle i represents a potential solution of the problem and it is associated with two vectors, namely the position vector $X_i = [X_{i1}, X_{i2}, \ldots, X_{iD}]$ and the velocity vector $V_i = [V_{i1}, V_{i2}, \ldots, V_{iD}]$ to indicate its current state. One salient feature of PSO that distinguishes it from other MS algorithms is the capability of particle to remember its personal best experience, that is, the best position that it ever achieves. During the searching process, each particle of the population stochastically adapts its trajectory through its personal best experience and the group best experience [1,2]. Specifically, the d-th dimension of particle i's velocity, $V_{i,d}(t+1)$ and position $X_{i,d}(t+1)$ at $(t+1)$-th iteration of the searching process are updated as follows:

$$V_{i,d}(t+1) = \omega V_{i,d}(t) + c_1 r_1 (P_{i,d}(t) - X_{i,d}(t)) + c_2 r_2 (P_{g,d}(t) - X_{i,d}(t)) \quad (1)$$

$$X_{i,d}(t+1) = X_{i,d}(t) + V_{i,d}(t+1) \quad (2)$$

where $i = 1, 2, \ldots, S$ is the particle's index; S is the population size; $P_i = [P_{i1}, P_{i2}, \ldots, P_{iD}]$ represents the particle i's personal best experience; $P_g = [P_{g1}, P_{g2}, \ldots, P_{gD}]$ is the group best experience found by all of the particles so far; c_1 and c_2 are the acceleration coefficients that control the influences of personal and group best experiences, respectively; r_1 and r_2 are two random numbers that generated with the uniform distribution in the range of [0,1]; and ω is the inertia weight that is used to balance the global/local searches of particles [11]. The implementation of basic PSO is illustrated in Fig. 1.

2.2. State-of-the-art PSO variants

Substantial amount of researches are performed to improve the PSO's performance. Among these works, parameter adaptation strategy has become one of the promising approaches. Shi and Eberhart [11] proposed a PSO with linearly decreasing inertia weight (PSO-LDIW) by introducing a parameter called inertia weight ω into the basic PSO. Accordingly, the parameter ω is linearly decreased to balance the exploration/exploitation search of PSO. Based on a thorough theoretical study on the convergence properties of PSO swarm, Clerc and Kennedy [12] proposed a constriction factor χ into basic PSO to prevent the swarm explosion, thereby developed the Constricted PSO (CPSO). Ratneweera et al. [15] proposed a time varying acceleration coefficient (TVAC) strategy into PSO, where the acceleration coefficients of c_1 and c_2 are decreased and increased linearly with time, to regulate the exploration/exploitation behaviors of swarm. In [15], two variants of PSO-TVAC, namely the PSO-TVAC with mutation (MPSO-TVAC) and Self-Organizing Hierarchical PSO-TVAC (HPSO-TVAC) were developed. Tang et al. [23] proposed a Feedback Learning PSO with quadratic inertia weight (FLPSO-QIW) by introducing a fitness feedback mechanism into the TVAC scheme. The particle's fitness is incorporated into the modified TVAC to adaptively determine the c_1 and c_2 values. By proposing an evolutionary state estimation (ESE) module, Zhan et al. [21] developed an Adaptive PSO (APSO) that is capable to identify the swarm's evolutionary states. The outputs of the ESE module are then used to adaptively adjust the particles' ω, c_1 and c_2. Leu and Yeh [27] proposed a Grey PSO, by employing the grey relational analysis to tune the particles' ω, c_1 and c_2. Hsieh et al. [19] developed an efficient population utilization strategy for PSO (EPUS-PSO). Accordingly, a population manager is proposed to adaptively adjust the population size according to the population's searching status.

Population topology is another crucial factor that determines the PSO performance as it decides the information flow rate of the best solution within the swarm [32,33]. In [32,33], different topologies with different connectivity, such as the fully connected, ring, and wheel topologies, were studied. Carvalho and Bastos-Filho [17] developed a clan topology according to the social behavior of clan. In the Clan PSO, the PSO population is divided into several clans. Each clan will first perform the search and the particle with best fitness is selected as the clan leader. A conference is then performed among the leaders to adjust their position. Bastos-Filho et al. [18] proposed a Dynamic Clan PSO by employing a migration mechanism into the clan topology. This improvement allows particles in one clan migrate to another clan. Meanwhile, Pontes et al. [22] hybridized the concept of clan topology into the APSO [21] to produce the ClanAPSO. Based on the evolutionary state of each clan, ClanAPSO enables different clans to employ different search operations. Parsopoulos and Vrahatis [14] proposed a Unified PSO (UPSO) to balance the exploration/exploitation search. Mendes et al. [13] advocated that each particle's movement is influenced by all its topological neighbors and thereby proposed the fully informed PSO (FIPSO). Montes de Oca [20] integrated the concepts of time-varying population topology, FIPSO's velocity updating mechanism [13], and decreasing ω [11], to develop the Frankenstein PSO (FPSO). Initially, the particles in FPSO are connected with fully connected topology. The topology connectively is then reduced over time with certain pattern.

Another area of research is to explore the PSO's learning strategies. Liang et al. [16] proposed the Comprehensive Learning PSO (CLPSO). Accordingly, each particle is allowed to learn from its or other particle's historical best position in each dimension, to ensure a larger search space is explored. Wang et al. [24] proposed a CLPSO variant by employing a generalized opposition-based learning to the CLPSO. Motivated by a social phenomenon where multiple of good exemplars assist the crowd to progress better, Huang et al. [26] proposed an Example-based Learning PSO (ELPSO). Instead of a single P_g particle, an example set of multiple global best particles is employed to update the particles' position in ELPSO. Noel [28] hybridized the PSO with a gradient-based local search algorithm, to combine the strengths of stochastic and deterministic optimization schemes. Zhou et al. [25] introduced the Random Position PSO (RPPSO) by proposing a probability $P(\Delta f)$. A random position is used to guide the particle, if the randomly generated number is smaller than $P(\Delta f)$. Jin et al. [29] advocated to update the particles' velocities and positions in certain dimensions and thus proposed the PSO with dimension selection methods. A total of three approaches,