Engineering Applications of Artificial Intelligence 70 (2018) 159–169

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Particle swarm optimizer with crossover operation

Yonggang Chen [a,b], Lixiang Li [c,*], Jinghua Xiao [d], Yixian Yang [c], Jun Liang [c], Tao Li [e]

[a] Information Engineering College, Henan University of Science and Technology, Luoyang 471023, China
[b] School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
[c] Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[d] State Key Laboratory of Information Photonics and Optical Communications, School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
[e] College of Computer Science, Sichuan University, Sichuan 610065, China

A R T I C L E I N F O

Keywords:
Swarm intelligence
Particle swarm optimization
Crossover operation
Global optimization

A B S T R A C T

A particle swarm optimization algorithm with crossover operation (PSOCO) is proposed. In the proposed PSOCO, two different crossover operations are employed in order to breed promising exemplars. By performing crossover on the personal historical best position of each particle, the effective guiding exemplars are constructed and they maintain a good diversity. In turn, these high quality exemplars are used to guide the evolution of particles. PSOCO is two-layer particle swarm optimization with positive feedback mechanism. In order to test the performance of PSOCO, we use a set of widely used benchmark functions. The experimental results demonstrate that the proposed PSOCO is a competitive optimizer in terms of both solution quality and efficiency.

1. Introduction

Lots of control and decision problems can be transformed into optimization problems (Chang, 2009; Chang and Shih, 2010; Pan et al., 2008; Jarboui et al., 2008). The optimization method and its application in the research of complex system is a challenging subject. With the development of technology, many problems encountered in engineering practice are becoming more and more complex. Some traditional computing methods are used to solve these problems. However, these methods often have the problems of high computational complexity and require optimization problems to be represented by continuous and differentiable function. Due to the limitations of traditional algorithms, lots of metaheuristic algorithm are designed to find satisfactory and approximate solutions for hard optimization problems.

In the research of nature-inspired metaheuristic optimization algorithms, the research of swarm intelligence algorithms is one of the most active fields. Some of the well known swarm intelligence algorithms are developed during the last two decades. Popular swarm intelligence optimizers include particle swarm optimization (PSO) (Eberhart and Kennedy, 1995; Kenndy and Eberhart, 1995), ant colony optimization (Yao et al., 2015), chaotic ant swarm algorithm (Li et al., 2006), biogeography-based optimization (Simon, 2008), etc. Many practices and experiments have shown that swarm intelligence algorithms are more effective and efficient in solving hard optimization problems (Zhang et al., 2010; Peng et al., 2013; Cai et al., 2012).

As a population-based and global optimizer, PSO was initially proposed by James Kennedy and Russell Eberhart in 1995. PSO was inspired by emulating the foraging behaviors of birds and fish schooling. Compared with other swarm intelligence algorithms, PSO is easy to be implemented and it has a good performance in solving many real world problems (Zhang et al., 2015). Owing to its simple concept, PSO has been the most popular and well-known algorithm. So far, PSO has been successfully applied in solving lots of scientific and engineering problems (Lu et al., 2015; Shen et al., 2014; Lin et al., 2016; Haddar et al., 2016; Gaxiola et al., 2016). However, lots of experiments have shown that traditional PSO algorithms cannot guarantee to find the global optimal solution and are easy to fall into local optima. In order to improve the performance of PSO, lots of PSO variants have been proposed. These variants can be mainly classified into the following two categories.

The first method is based on self-change strategy. Shi and Eberhart proposed that the inertia weight with decreased linearly was a good choice in 1998 (Shi and Eberhart, 1998). The inertia weight with fuzzy control has been proposed further (Shi and Eberhart, 2001). Other control strategies for parameters embedded in the particle motion equation have also been proposed (Taherkhani and Safabakhsh, 2016; Guo et al., 2006; Ratnaweera et al., 2004; Ardizzon et al., 2015; Olivas et

* Corresponding author.
E-mail address: lixiang@bupt.edu.cn (L. Li).

https://doi.org/10.1016/j.engappai.2018.01.009
Received 26 September 2016; Received in revised form 5 April 2017; Accepted 20 January 2018

al., 2016). The main role of these additional parameters is to adjust the particle's flight. In PSO, it is a good idea that new topological structures are adopted. Some other topologies have been proposed (Kennedy, 1999). In 2004, Mendes et al. proposed a fully informed PSO (FIPS) (Mendes et al., 2004). In FIPS, the historical best positions of several neighbors were used to adjust the position of each particle. A new updating model of velocity was proposed in the comprehensive learning PSO (CLPSO) (Liang et al., 2006). In CLPSO, the velocity updating rule of each particle was based on different historical personal best positions.

The second method is to combine other strategies in PSO (Liu et al., 2015; Zhao et al., 2016; Wu et al., 2014; Li et al., 2015a; Tang and Fang, 2015; Samma et al., 2016; Zhang et al., 2014; Wang et al., 2016; Valdez et al., 2017). Based on orthogonal experimental design (OED), Zhan et al. proposed orthogonal learning PSO (OLPSO) (Zhan et al., 2011). Because OED can mine for useful information, an effective guidance exemplar can be constructed by orthogonal learning (OL) strategy. In this way, particle swarm can fly to promising position. The experiments showed that the OL strategy could give PSO more robust and reliable in numerical optimization problems. In ALC–PSO (Chen et al., 2013), Chen et al. introduced the aging mechanism into PSO. This mechanism was used to change the historically best position of the entire swarm. The simulation results showed that this method enhanced the optimization performance of PSO. Social learning mechanisms was introduced into PSO by Cheng and Jin (2015). In this version, other better particles based on the current swarm were used to guide each particle to fly. The experiment results showed this idea was more successful. Using multi-swarm technique is also a good method (Xu et al., 2015; Chen, 2011; Gülcü and Kodaz, 2015). Because of the information exchange among different groups, multi-swarm technology can better balance the exploiting and exploration ability of the algorithm. Liang and Suganthan designed a dynamic multiswarm PSO (DMS-PSO) (Liang and Suganthan, 2005). In DMS-PSO, because these small swarms were regrouped continuously, the whole swarm maintained better diversity. A multi-layer PSO (MLPSO) was proposed by Wang et al. (2014). The MLPSO can avoid lapsing into the local optimum by increasing the number of the swarm layers. Multi-layer strategy can be regarded as a multi-swarm technique.

So far, lots of crossover operators are proposed. The crossover operator has always been regarded as one of the main search operators in genetic algorithm, because it exploits the available information in previous samples to influence future searches (García-Martínez et al., 2008). The crossover operator can combine parts of good solution to form new potential solution (Singh et al., 2011). Due to its advantages, crossover operator can improve the performance of PSO. In this paper, the scheme based on two crossover operators is designed. PSOCO is different from other PSO algorithms with crossover operator in many aspects. Differential Evolutionary (DE) crossover operator and the modified velocity update strategy are adopted in PSOCO. A dynamic adjustment strategy is adopted to enhance exploitation capability. The main loop of the PSOCO is composed of two layers. In the top layer, two crossover operators are used to construct promising exemplar. Particles in the bottom layer are guided by exemplar. The PSOCO establishes a positive feed back to accelerate the population to find the optimum. Moreover, the two-layer evolution mechanism of the PSOCO can accelerate the convergence speed of the population.

The main contributions of this paper can be summarized as follow:

(1) In this paper, two crossover operators and the velocity update scheme with learning model are incorporated into the proposed approach.

(2) The role of crossover operation is investigated. The influences of some parameters are investigated. According to some experimental results, a better parameter configuration is found.

(3) A dynamic strategy is proposed. This strategy is simple and can enhance the exploitation ability of the proposed approach. Our experimental results show that PSOCO has better optimization performance on some benchmark functions.

The remainder of this paper is organized as follows. Section 2 introduces the basic PSO and gives a review about the crossover-based PSO algorithms. Section 3 presents the PSOCO in detail. In Section 4, the experimental results are discussed and analyzed between several state-of-the-art PSO algorithms and PSOCO. Some analysis of PSOCO is further discussed. Conclusions and outlook of work are given in Section 5.

2. Related works

2.1. The basic PSO theory

In PSO, a particle is used to represent the potential solution of the optimization problem. The whole particle swarm flies in the search space to search for the global optimum. In a D-dimensional hyperspace, the following two vectors are related to the ith particle ($i = 1, 2 \ldots N$), where N denotes the population size. One is the velocity vector $V_i = (v_{i1}, v_{i2}, \ldots, v_{id}, \ldots, v_{iD})$, and the other is the position vector $X_i = (x_{i1}, x_{i2}, \ldots, x_{id}, \ldots, x_{iD})$, where $d \in [1, D]$. Firstly, the velocity and the position of each particle are initialized by random vectors with the corresponding ranges. Two equations are used to adjust the position. The position of the particle on each dimension is updated using the following equations:

$$v_{id}^{t+1} = wv_{id}^{t} + c_1 rand_{1d}^{t}(P_{id}^{t} - x_{id}^{t}) + c_2 rand_{2d}^{t}(P_{gd}^{t} - x_{id}^{t}) \quad (1)$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \quad (2)$$

where w is named as inertia weight. c_1 and c_2 are called the acceleration coefficients. $rand_1$ and $rand_2$ are two uniformly distributed random numbers in range $[0, 1]$. d denotes one dimension of a particle. P_g denotes the best position that the whole swarm has found so far. In addition, the flying velocity is limited to a reasonable range. A positive value V_{MAXd} is used to clamp the updated velocity. If $|v_{id}|$ exceeds V_{MAXd}, then the velocity is set to $\text{sign}(v_{id})V_{MAXd}$.

2.2. Crossover-based PSO algorithms

Crossover operator can enhance the information sharing between particles and prevent the premature convergence of swarm. In the past ten years, lots of meaningful work has been done in this field. For example, the work in Pant et al. (2007) introduced the quadratic crossover operator in PSO. This nonlinear crossover operator made use of the three particles to produce a better particle in the search space. Ref. Wang et al. (2008) presented PSO with a novel multi-parent crossover operator. According to multi-parent crossover operator equation, the offspring still stayed in the linear space by the four different parents. In order to achieve a successful recombination, a selection operator was employed. In Zhang et al. (2013), the crossover operator was implemented by borrowing the concept of linear combination of two vectors, where one vector was the swarm member and the other vector was selected randomly from elite set. In Tawhid and Ali (2016), arithmetical crossover operator and Nelder–Mead method are used to solve global optimization problems. In Duong et al. (2010), a hybrid of PSO and a GA (HEA) is proposed. In HEA, the BLX-crossover is used to produce offspring. In Chen (2012), particle swarm optimization with *pbest* crossover is proposed. The algorithm can achieve large improvements on Black-Box Optimization Benchmarking functions by crossing the personal best positions. Ref. Miao et al. (2009) presents Dynamic PSO with arithmetic crossover (DP-PSO). Crossover operator was effective for high-dimensional multimodel benchmarks in DP-PSO. Hao et al. (2007) proposed a PSO based on crossover, where the arithmetic crossover operator is adopted in one dimension of two different individual best position. Xie and Yang (2010) developed a novel crossover operator for particle swarm algorithm, where a Laplace crossover operator was employed to generate good candidate solutions. In Engelbrecht (2016), an extensive review of PSO algorithms that make use of some form