Engineering Applications of Artificial Intelligence 37 (2015) 9–19

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Multi-strategy adaptive particle swarm optimization for numerical optimization

Kezong Tang [a,b,*], Zuoyong Li [c], Limin Luo [a], Bingxiang Liu [b]

[a] School of Computer Science and Engineering, Southeast University, Nanjing Jiangsu 210094, PR China
[b] Information Engineering Institute, Jingdezhen Ceramic Institute, Jingdezhen Jiangxi 333000, PR China
[c] Department of Computer Science, Minjiang University, Fuzhou Fujian 350108, PR China

ARTICLE INFO

Article history:
Received 7 February 2014
Received in revised form
7 July 2014
Accepted 1 August 2014
Available online 15 September 2014

Keywords:
Particle swarm optimization
Optimization problems
Diversity of population
Entropy
Image segmentation

ABSTRACT

To search the global optimum across the entire search space with a very fast convergence speed, we propose a multi-strategy adaptive particle swarm optimization (MAPSO). MAPSO develops an innovative strategy of diversity-measurement to evaluate the population distribution, and performs a real-time alternating strategy to determine one of two predefined evolutionary states, exploration and exploitation, in each iteration. During iterative optimization, MAPSO can dynamically control the inertia weight according to the diversity of particles. Moreover, MAPSO introduces an elitist learning strategy to enhance population diversity and to prevent the population from possibly falling into local optimal solutions. The elitist learning strategy not only acts on the globally best particle, but also on some special particles that are very near to the globally best particle. The aforementioned features of MAPSO have been comprehensively analyzed and tested on eight benchmark problems and a standard test image. Experimental results show that MAPSO can substantially enhance the ability of PSOs to jump out of the local optimal solutions and significantly improve the search efficiency and convergence speed.

1. Introduction

In real-world applications, there exist a lot of complicated optimization problems whose model might change along with environment. In the past decades, researchers have focused on solving complicated optimization problems in chemical industry production and bioengineering research, such as the economic dispatch problem, the minimum spanning tree problem, and the vehicle routing problem. To solve these optimization problems, some traditional optimization methods have emerged, such as the gradient-based method, Nelder–Mead's simplex method, and the quasi-Newton method. However, these methods usually lead to inaccurate and even infeasible solutions. This restricts their further application to some practical problems, as many practical problems are usually discontinuous and even non-differentiable at some points in the domain. Moreover, it is worth noting that there are no common methods for solving all complex problems in a similar context according to the non-free lunch theorem.

Apart from the aforementioned traditional methods, there are also some well-established and common heuristic population-based methods such as Genetic Algorithm (GA) (Holland, 1975), Ant Colony Algorithm (ACA) (Dorigo and Gambardella, 1997), Particle Swarm Optimization (PSO) (Kennedy and Eberhart, 1995), Bacterial Foraging Algorithm (BFA) (Passino, 2002), Differential Evolution (Storn and Price, 1997), and Artificial Neural Network (ANN) (Lu et al., 2003). These methods have their own characteristics, merits, and demerits. Among them, PSO is probably the simplest and the most effective, and a lot of research has been done on the optimization of complex system (Kathiravan and Ganguli (2007); Gudla and Ganguli, 2005; Samanta and Nataraj, 2009; Modares et al., 2010; Dutta et al.,2013; Lim and Isa, 2014).

As a well-known swarm intelligence method, PSO has been attracting considerable interest owing to its simple mechanism and easy implementation, as it requires only a few parameters. Compared with traditional methods, PSO has four notable advantages (Yang et al., 2007; Tang et al., 2011a): (1) PSO is conceptually simple and relatively easy to implement, which is mainly due to a simple search mechanism that mimics feeding behavior in birds flocking and fish schooling. (2) In a similar fashion to other population-based iterative algorithms, PSO is less sensitive to the characteristics of optimization problems, which means that PSO does not require complex functions to be continuous or even to be differentiable as required by traditional methods. (3) In most cases, PSO has the ability to jump out of the local optima because of the individual memory of the particles. Thus, the knowledge of good solutions is retained throughout the search process. (4) PSO can be programmed easily and is computationally inexpensive in terms of both memory capacity and running speed.

* Corresponding author. Tel.: +8613627987439.
E-mail address: tangkezong@126.com (K. Tang).

http://dx.doi.org/10.1016/j.engappai.2014.08.002
0952-1976/

Despite PSO having been successfully applied to some complex problems such as the vehicle routing problem, the economic dispatch problem, the power allocation problem, or to problems having irregular, noisy, and multimodal characteristics, there are still some problems with running PSO. For instance, sometimes PSO might fall into local optimal solutions because of the faster loss of diversity on some optimization problems. When entering into a location near the optimal solution, the loss of diversity is too fast for the entire population to guarantee convergence to the optimal solution. Hence, the convergence speed can descend dramatically in the later stages of evolution. To overcome such problems, various attempts have been made to enhance the performance of PSO, such as by adjusting the inertia weight, the variation of neighbor topology, or the dynamic regulation of diversity. Tang and Wu (2013) used a neighboring function criterion to maintain diversity among population embers. Sun et al. (2011) proposed a novel variant of quantum-behaved particle swarm optimization (QPSO) algorithm with the local attractor point subject to a Gaussian probability distribution (GAQPSO). Zhao (2010) presented a perturbed particle-swarm algorithm based on a new particle-updating strategy, derived from the concept of perturbed global best particle, to deal with the problem of premature convergence and diversity maintenance within the swarm. Zhan et al. (2009) presented an adaptive PSO. By evaluating the population distribution and particle fitness, a real-time evolutionary state estimation procedure is performed to identify one of four predefined evolutionary states, and an elitist learning strategy is performed when the evolutionary state is classified as the convergence state. Experimental results show that APSO substantially enhances the performance of the PSO paradigm in terms of convergence speed, global optimality, solution accuracy, and algorithm reliability. Du and Li (2008) divided all particles into two parts. Then, two new strategies (Gaussian local search and differential mutation) are introduced into the two parts, respectively. Experimental results show that the two mechanisms can enhance the convergence ability of PSO, while the searching area of the particle population can be extended to avoid being trapped at the local optimum. Carlisle and Dozier (2002) used a multi-niche crowding mechanism to maintain population diversity throughout the run, and the spread-out population was used to enhance the adaptability of the algorithm to dynamic environments. Blackwell and Bentley (2002) proposed an atom analogy structure PSO optimizer, which makes use of the Coulomb force to increase diversity. Additionally, a multi-population technique has also been used to enhance the population diversity to respond promptly to the changed environment (Blackwell and Branke, 2004, 2006; Li et al., 2006; Parrott and Li, 2004).

Among the studies on PSO, there are two most important and appealing objectives, i.e., accelerating convergence speed and avoiding the local optima. Adaptive control of inertia weight and the adjustment of diversity have become the most promising approaches. In this paper, we propose a novel multi-strategy adaptive PSO (MAPSO). MAPSO first adopts a novel evaluation strategy of diversity to adjust the population distribution in search phases. Then, MAPSO uses the proposed evaluation strategy of diversity to dynamically change inertia weight, achieving a good balance between exploration and exploitation. Finally, MAPSO utilizes an elitist learning strategy (*ELS*) to enhance the population diversity and prevent search into a local region of optimum. This expands the search area of solutions at the next stage. *ELS* not only acts on the globally best particle, but also on some special particles that are very near to the globally best particle. Experimental results show that MAPSO is not only easy to implement, but also computationally efficient for complex multidimensional problems and better than the other compared algorithms in terms of quality of solution.

The rest of this paper is organized as follows: Section 2 describes the basic model of PSO. The MAPSO algorithm is proposed in Section 3 through the development of a novel evaluation strategy of diversity and adaptive adjustment on the inertia weight and an elitist learning strategy. Section 4 experimentally compares MAPSO with other PSO algorithms using a set of benchmark functions. Section 5 concludes with a brief summary of the paper and some paths for future research.

2. PSO scheme

Particle swarm optimization, introduced by Kennedy and Eberhart (1995), is one of the most important stochastic global optimization techniques. It mimics the process of predation in birds flocking or fish schooling and starts with artificial particles (represented by a group of "birds"). PSO tries to evolve those particles that are fitter and, by applying a particle updating strategy (measurement of velocity and displacement), it attempts to move to a new position that is better than the previous one in the search space. In spite of the diversity of PSO schemes, most of them are based on the same iterative procedure.

As a stochastic population-based optimization method, PSO is similar to a "black box", completely independent from the characteristic of the optimized problem. Fig. 1 describes the classic PSO flow chart. An initial population of particles is generated randomly, where each particle is represented as a potential solution. Each of these particles is evaluated in terms of a certain "fitness function" that can guide particles to search for globally optimal solutions. Kennedy's three basic updating operations (extremum, position, velocity) are the main components to improve the PSO's performance. Updating the extremum includes the operations of individual extremum "*pbest*" and global extremum "*gbest*" in each iteration. The *pbest* is the position with the best fitness found so far for the *i*th particle (memorized by every particle). The *gbest* is the best position in a neighborhood, where typical topologies (Ghosh et al., 2009) are the fully connected structure, the ring structure, the Von Neumann structure, and so on. Then, the velocity $v_{i,d}$ and position $x_{i,d}$ of the particle i on the dth dimension

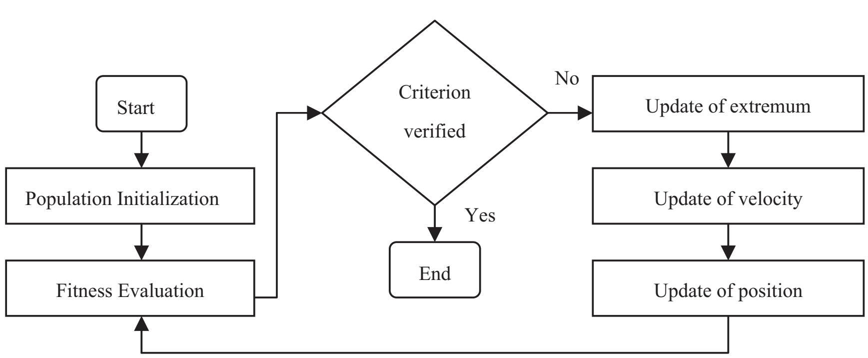

Fig. 1. PSO flow chart.