Applied Soft Computing 59 (2017) 340–362

Contents lists available at ScienceDirect

Applied Soft Computing

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

A modified competitive swarm optimizer for large scale optimization problems

Prabhujit Mohapatra*, Kedar Nath Das, Santanu Roy

Department of Mathematics, National Institute of Technology Silchar, Silchar, Assam, India

ARTICLE INFO

Article history:
Received 2 August 2016
Received in revised form 9 May 2017
Accepted 30 May 2017

Keywords:
Particle swarm optimization
Competitive swarm optimizer
Large scale global optimization
Evolutionary algorithms
Swarm intelligence

ABSTRACT

In the recent literature a popular algorithm namely 'Competitive Swarm Optimizer (CSO)' has been proposed for solving unconstrained optimization problems that updates only half of the population in each iteration. A modified CSO (MCSO) is being proposed in this paper where two thirds of the population swarms are being updated by a tri-competitive criterion unlike CSO. A small change in CSO makes a huge difference in the solution quality. The basic idea behind the proposition is to maintain a higher rate of exploration to the search space with a faster rate of convergence. The proposed MCSO is applied to solve the standard CEC2008 and CEC2013 large scale unconstrained benchmark optimization problems. The empirical results and statistical analysis confirm the better overall performance of MCSO over many other state-of-the-art meta-heuristics, including CSO. In order to confirm the superiority further, a real life problem namely 'sampling-based image matting problem' is solved. Considering the winners of CEC 2008 and 2013, MCSO attains the second best position in the competition.

1. Introduction

Particle Swarm Optimization (PSO) is one of the powerful and effective swarm intelligence techniques [1] introduced by Kennedy and Eberhart in 1995 [2] for solving optimization problems. It is inspired by the social behavior of bird flocking or fish schooling. Due to easy understanding and ease implementation, PSO has witnessed a rapid growth in popularity over last few decades [3–8]. In PSO, the best position that has ever been found by each particle in the swarm is termed as personal best (*pbest*); whereas the best position that has been found by the whole swarm is known as global best (*gbest*). The particles learn from the *pbest* and *gbest* positions, during simulation to approach towards global optimum. It is clear from the literature that the performance of PSO becomes unsatisfactory in solving high dimensional, non-separable and multi-modal problems [9–11,14]. These weaknesses are attributed as premature convergence which usually occurs in PSO [12]. In order to enhance the search capability and hence the performance of the PSO, several variants have been proposed over the time. These variants are primarily classified into different categories such as adaptive control strategy of parameters in PSO [13–17], hybrid PSO algorithms [18–20], topological structures in neighborhood control strategy of PSO [21–23], Multi-swarm PSO [24–26] etc. Unfortunately, in most of these PSO variants, the introduction of new mechanisms and operators often increases the computational complexity. Another major issue is related to the strong influence of the *gbest* position on the convergence speed, which is basically responsible for the premature convergence. To overcome this issue, Liang [23] proposed a new PSO variant without the *gbest* term and the update strategy relies only on the *pbest* position. An alternate way to address this issue is to get freed of both the *pbest* and *gbest* factors. In 2013, the first attempt was made with a multi-swarm framework based on a feedback mechanism [27], where particles are updated by a pairwise competition between particles of two different swarms. In second attempt, a social learning mechanism [28] is introduced where each particle learn from any better particles in the swarm. Similar mechanisms have been employed by many other researchers [29–31]. This concept of competitive mechanisms mainly results with two consequences. First as a convergence strategy, the weak solutions get a chance to learn from the stronger ones of the other swarm and the second as a mutation strategy, the strong individuals self-motivated by the previous experiences to produce better solutions. These strategies work together to catch hold of a good balance between exploration and exploitation. Following this idea, another algorithm called competitive swarm optimizer (CSO) [32] is proposed that uses the competitive mechanism between particles within a single swarm.

* Corresponding author.
E-mail addresses: prabhujit.mohapatra@gmail.com (P. Mohapatra), kedar.iitr@gmail.com (K. Nath Das), santanuroy79@yahoo.in (S. Roy).

http://dx.doi.org/10.1016/j.asoc.2017.05.060
1568-4946/

After each pair-wise competition, the loser particle learns from the winner particle instead of from *pbest* or *gbest*. Surprisingly, it outperforms quite a good number of meta-heuristics without the involvement of the *pbest*/*gbest*. The concept of CSO algorithm is very simple, yet powerful to solve high dimensional large scale optimization problems.

Though the CSO algorithm has established a number of success mile stones in solving complex optimization problems, the better accuracy in the solution quality and higher rate of convergence are yet to be addressed. Therefore in this paper, a modified CSO is being proposed. It is worth noting that CSO uses the concept of pair-wise competition to update half of the swarm in each iteration during simulation. Unlike CSO, the modified version uses tri-competitive mechanism to update two thirds of the swarm each time. The prime idea is to allow the swarm of the population for a larger rate of exploration those results in a faster rate of convergence. The paper is organized as follows. In Section 2, the related works to large scale optimization problem are being reviewed. The motivation behind the work and the proposed algorithm are presented in Section 3. In Section 4, the experimental studies are carried out. A real world application of the proposed algorithm has been presented in Section 5. Finally, the conclusion is drawn in Section 6.

2. Large scale optimization problems

Most real-world optimization problems deal with a large number of decision variables, which are popularly known as Large Scale Global Optimization (LSGO) problems. In general, these problems become hard to handle as they are mostly complex due to presence of higher multimodality and large number of decision variables. Often the neighborhood search space becomes so narrow that it becomes very difficult to locate the global optimal solution. Since last few decades, several population based meta-heuristics have been proposed to solve them. However, these algorithms suffer with lots of jolts such as rapid deterioration of performances and exponential increase in computational complexity during simulation. Hence solving LSGO becomes a challenge in different fields of science and engineering. Over the time, quite a large number of algorithms are proposed in the literature to handle them. Majorly, such algorithms can be categorized in two ways based on the decomposition of the problem dimensions as described below.

The first type is 'Decomposition based algorithms'. They are also known as Cooperative Coevolution (CC) algorithms, where the high dimensional problems are decomposed into low dimensional subproblems. This concept is first introduced by Potter and De Jong [33] in 1994. Each subcomponent here undergoes a traditional optimization algorithm for a predefined number of generations in a round-robin strategy. Then the solution from each subcomponent is merged to form the n-dimensional solution. Yang et al. [34] incorporated a DE- based CC method called DECC-G [35] which uses the concept of random grouping of decision variables to solve LSGO problems of 500 and 1000 dimensions. Later, it has been improved in multilevel CC algorithm (MLCC) [36], uses a decomposer pool which employs dynamic group size of variables based on the past performance of the decomposer. With a gradual improvement, similar algorithms namely CCPSO2 [26] and CC-CMA-ES [37] are being proposed.

The second type is the 'Non-Decomposition based algorithms', where it avoids the divide-and-conquer strategy and rather applies different effective strategies to enhance the performance of the algorithms. These methods are primarily classified as local search based [38,39] evolutionary computation based [40,41] and swarm intelligence based approaches [42]. The algorithm of motivation of the proposed work namely CSO [32] and the proposed work MCSO both lie under the swarm intelligence approach.

Table 1
Swarm diameter comparison between MCSO and CSO.

Dimension	Iterations	Separable Function (f_1)		Non-Separable Function (f_2)	
		CSO	MCSO	CSO	MCSO
100D	1	9.96E+02	9.66E+02	1.00E+03	9.88E+02
	500	7.07E+02	1.00E-01	8.62E+02	6.41E+01
	1000	5.93E+02	1.42E-05	9.78E+02	7.69E+00
500D	1	2.02E+03	1.96E+03	2.01E+03	2.01E+03
	500	1.83E+03	5.98E+01	2.32E+03	8.99E+02
	1000	1.26E+03	2.88E+00	6.36E+02	6.74E+02
1000D	1	2.76E+03	2.77E+03	2.80E+03	2.77E+03
	500	2.69E+03	1.42E+02	2.78E+03	1.18E+03
	1000	2.47E+03	1.05E+01	1.96E+03	1.05E+03

3. Motivation and proposition

CSO is one such recently proposed popular algorithm in the literature that uses pair-wise competitive scenario [32]. An attempt is made in this paper to improve the working mechanism of CSO resulting in possible improvement in the solution quality. The major motivations behind the proposition are:-

i In CSO, half of the swarms are updated in each of the iteration. Therefore rest half particles in the swarm deprives of migrating towards the good solutions. As a result, the existence of high diversity in the swarm is unavoidable. In the proposed algorithm, 2/3rd of the swarms are allowed to participate in the process of up-gradation that results in a higher rate of convergence. Subsequently, the rest 1/3rd of population passes directly to the next generation and thus they fulfill the necessity of swarm diversity. More clearly, in maintaining the diversity in the swarm. To measure the swarm diversity as well as the convergence, the swarm diameter [44] is calculated as follows and are compared in Table 1 using the following formula.

$$|D| = \max_{(i \neq j) \in [1,|S|]} \left(\sqrt{\sum_{k=1}^{I} \left(x_{ik} - x_{jk}\right)^2} \right)$$

where $|S|$ is the swarm size, I is the problem dimension and x_{ik}, x_{jk} is the k^{th} dimension of the i^{th} and j^{th} particle position respectively. The large value of the swarm diameter ($|D|$) signifies high particle dispersion and smaller value signifies convergence. From Table 1, it is observed that the swarm diameter of the MCSO algorithm remains smaller always than CSO for separable and non-separable functions as well.

ii The motivation behind the division of the population into three swarms is to increase the convergence speed. The bi-population concept in CSO, allows half of the population to update i.e. the losers only. But the tri-population concept in the proposed method will allow two thirds of the population (superior loser and inferior loser) to update. Hence more swarms are converging towards the good solutions. However, the higher breakup of the population is not recommended due to failure of neighborhood topology [45,46], according to which a tri-population concept is considered in this research due the large population size (i.e. 200 or more). In addition, as reported in [47–50], the tri-population breakup also maintains a proper balance between exploration and exploitation. It also helps to increase the diversity in the population [51]. Hence, the tri-break up of the population is chosen in the proposed algorithm.