Proceedings of the 36th Chinese Control Conference
July 26-28, 2017, Dalian, China

A Competitive Swarm Optimizer Integrated with Cauchy and Gaussian Mutation for Large Scale Optimization

Qiang Zhang, Hui Cheng, Zhencheng Ye, Zhenlei Wang

Key Laboratory of Advance Control and optimization for Chemical Processes, Ministry of Education,
East China University of Science and Technology, Shanghai 200237
E-mail: huihyva@ecust.edu.cn

Abstract: Competitive swarm optimizer (CSO) has shown promising results for solving large scale global optimization problems proposed recently. However, CSO shows insufficient exploitation of the population. In this paper, a competitive swarm optimizer integrated with Cauchy and Gaussian mutation (CGCSO) is proposed for large scale optimization. The new algorithm does not only update the losers' positions with the CSO method, but also update the winners' positions by Cauchy and Gaussian mutation to improve the exploitation capability of the population. Moreover, CGCSO utilizes the ring topology to enhance the swarm diversity and alleviate premature convergence. A comparative study between CGCSO and CSO evaluated on the CEC'08 benchmark functions has been carried out. The experimental results indicate that CGCSO performs better on the whole, especially on the non-separable functions and 500-dimensional problems.

Key Words: Competitive Swarm Optimizer, Cauchy Mutation, Neighborhood Topology, Particle Swarm Optimization, Large Scale Optimization

1. Introduction

Evolutionary optimization has achieved great success on many numerical and combinational problems in recent years [1]. Evolutionary algorithms (EAs) mainly include Genetic Algorithm (GA), Evolutionary Programming (EP) and Evolutionary Strategy (ES). Particle swarm optimization (PSO), introduced by Kennedy and Eberhart in 1995 [2], is one of the powerful evolutionary algorithm based on swarm intelligence.

PSO was derived from the study of social animals' behaviors such as bird flocking. Individuals of bird flocking learn from its neighbors and fly to positions which closer to the food, and the whole flocking seems to be controlled under a center. The study results indicate that complex behaviors are generated from interacted rules.

Due to its simplicity of program implement, PSO has attracted many researchers and been widely applied with success to many optimization problems over the past decades [3-4]. However, PSO performs very poor, with involving in complex optimization problems and high search dimensions [5]. In many cases, the growing dimensionality in an optimization problem will significantly increase the number of local optimums. PSO is difficult to escape from a local optimum due to the strong influence of global best position, which resulting in premature convergence [6].

A famous attempt to tackle high-dimensional optimization problems is the cooperative coevolution (CC) approach [7]. It decomposes a high-dimensional problem into sub-problems, and optimizes these sub-problems with a certain evolutionary algorithm individually. Although CC method has been verified to improve the performance of dealing with large scale optimizations, it also has some drawbacks [8].

1) Its performance is sensitive to the choice of decomposition strategy.
2) Its performance will be poor when the number of interacting variables grows. When the optimization problem is fully non-separable, its performance will only depend on which EA it adopts.
3) The computation complexity is high.

Apart from the method of decomposing optimization problems, designing an effective learning strategy, enhancing the swarm diversity and balancing exploration and exploitation also become an important approach to deal with large scale optimization problems. One representative algorithm is the competitive swarm optimizer (CSO) proposed by Cheng and Jin [11]. Experimental results showed that CSO outperformed overall the other PSO variants. However, CSO faces the problem of slow convergence, because it only updates a half of particles which lose the competition in each generation. A dynamic competitive swarm optimizer (DCSO) based on entropy was proposed in [12], where the diversity of the population is quantified by calculating the population entropy. DCSO utilizes the population entropy to divide the population into two sub-groups dynamically. The sub-group which has worse fitness learns from the better sub-group, and the better sub-group updates position and velocity according to itself. The method which decomposes the swarm based on the population entropy can effectively speed up convergence of CSO, but it does not consider to improve the search performance of CSO theoretically.

Building on the success of CSO, this paper proposed a competitive swarm optimizer integrated with Cauchy and Gaussian mutation (CGCSO). The new algorithm employed Cauchy and Gaussian mutation to improve exploitation capability of the population and help particles to escape from local optimum. First, CGCSO decomposes the population into many sub-groups and increases the number of particles in each competition to speed up convergence in the early stage. And then it utilizes a mutation operator based on

* This work is supported by National Natural Science Foundation of China (61590922, 61533003), Shanghai Natural Science Fund Program (16ZR1407300, 61403141).

Cauchy and Gaussian probability distribution to update the winner's position in each sub-group. Besides, the updating rules based on mutation operator use the ring topology to enhance the swarm diversity and search capability. Furthermore, experimental results indicate that CGCSO performs better than CSO on the whole, especially on non-separable functions and 500-dimensional problems.

The rest of this paper is organized as follows. Section 2 presents the CSO algorithm and makes a comparative analysis between CSO and canonical PSO. In section 3, the proposed CGCSO is presented in details. The experimental setup of CGCSO and the comparison with CSO on the CEC'08 benchmark functions are given in section 4. Section 5 summarizes this paper.

2. Competitive Swarm optimizer

CSO introduced the competitive mechanism of superiors survive in biology. It adopts a pairwise competition mechanism within one single swarm. CSO randomly pairwise grouping of the swarm at first, then two particles in one group compete with each other according to their fitness. As a result of each competition, the particle with better fitness denote as the winner, and the other denote as the loser. The loser updates its position and velocity by learning from the winner, and then all particles pass to the next generation.

In CSO, the M particles in the population are randomly divided into $M/2$ pairs, where M is the swarm size. The updating equations of the loser in each pair are given as follows.

$$\begin{aligned} v_l(t+1) = r_1(t)\cdot v_l(t) + r_2(t)\cdot(x_w(t)-x_l(t)) \\ +\varphi\cdot r_3(t)(\bar{x}(t)-x_l(t)) \end{aligned} \quad (1)$$

$$x_l(t+1) = x_l(t) + v_l(t+1) \quad (2)$$

Where $x_l(t)\in \mathrm{R}^D$ and $v_l(t)\in \mathrm{R}^D$ are the loser's position and velocity at generation t, D denotes as the dimensions of search space. $r_1(t)=(r_{1,1}, r_{1,2}, \cdots, r_{1,\mathrm{D}})$, $r_2(t)=(r_{2,1}, r_{2,2}, \cdots, r_{2,\mathrm{D}})$ and $r_3(t)=(r_{3,1}, r_{3,2}, \cdots, r_{3,\mathrm{D}})$ are three random vectors, in which each element is uniformly distributed in $[0,1]$. $\bar{x}(t)$ has two versions, one is the global version, which denotes as mean position of the whole population, and another is the local version where $\bar{x}(t)$ is the local mean position of predefined neighborhood particles of the winner. In general, $\bar{x}(t)$ denotes as the mean position of the population at generation t. φ is the control parameter that adjusts the influence of $\bar{x}(t)$.

Similar to the canonical PSO, CSO also keeps a simple principle and easy to be implemented in program. Nevertheless, CSO has some differences, for instance, it introduces a pairwise competition mechanism, there is no memory been used to memorize historical positions in it and the updating rule is different where it doesn't use the global best position ($gbest$) nor the personal best position ($pbest$).

However, there are some drawbacks of CSO. Due to it only updates a half of particles in each generation, the convergence speed is slow. Moreover, the exploitation of the population is not enough in CSO of which the learning strategy only depends on a half of particles to exploit in the search space. Our experimental studies shows that CSO has potential to improve performance for large scale optimization.

3. Algorithm

This section describes two techniques of Cauchy and Gaussian mutation and the ring topology, and presents the details of CGCSO.

3.1 Cauchy and Gaussian mutation

The typical using for Gaussian mutation could be used as updating rules in PSO [13], where there is no velocity used in. Fig. 1 shows the Cauchy and Gaussian density function curves. It indicates that Cauchy mutation is more likely to generate an offspring further away from its parent than Gaussian mutation due to its long flat tails [14]. Cauchy mutation is expected to have a higher probability of escaping from local optimum. As a result of this, in [14], Fast Evolutionary Program (FEP) utilizes Cauchy mutation instead of Gaussian mutation. But the ability of exploring in the search space is limited when using only a Gaussian distribution or a Cauchy distribution [10].

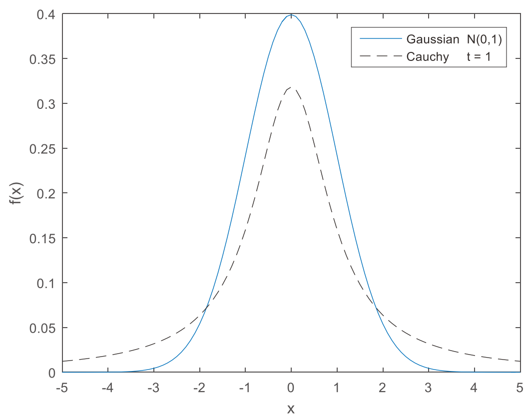

Fig. 1. Density functions of Cauchy and Gaussian

Inspired by [9-10], this paper combines Cauchy and Gaussian mutation to sample next point in the search space for the winner of each sub-group. The combination of Cauchy and Gaussian mutation can enhance diversity of the population and improve exploration performance for the winner in the early stage. In the late search stage, most particles in population are gathered in a small range when the losers are close to the winners, and the population falls into a local optimum. The winners' mutation can effectively improve exploitation capability and increase the probability that the population escape from the local optimum. Our experimental results also verified that Cauchy and Gaussian mutation can improve the search performance of CSO.

3.2 Ring Topology

Neighborhood topology is used to describe the relationship and interactive modes among particles, which controls the spreading speed of information in the population, and affects the exploitation performance and the convergence speed directly [15]. Many effective PSO variants with the motivation to enhance the diversity of the population were proposed based on the research of