Applied Soft Computing 35 (2015) 411–422

Contents lists available at ScienceDirect

Applied Soft Computing

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

Particle Swarm Optimization inspired by starling flock behavior

Nuttapong Netjinda, Tiranee Achalakul, Booncharoen Sirinaovakul *

Department of Computer Engineering, King Mongkut's University of Technology Thonburi, Bangkok 10140, Thailand

ARTICLE INFO

Article history:
Received 29 January 2014
Received in revised form 17 April 2015
Accepted 29 June 2015
Available online 6 July 2015

Keywords:
Particle Swarm Optimization
Swarm intelligence
Optimization
Collective behavior
Data clustering

ABSTRACT

Swarm intelligence is a meta-heuristic algorithm which is widely used nowadays for efficient solution of optimization problems. Particle Swarm Optimization (PSO) is one of the most popular types of swarm intelligence algorithm. This paper proposes a new Particle Swarm Optimization algorithm called *Starling PSO* based on the collective response of starlings. Although PSO performs well in many problems, algorithms in this category lack mechanisms which add diversity to exploration in the search process. Our proposed algorithm introduces a new mechanism into PSO to add diversity, a mechanism which is inspired by the collective response behavior of starlings. This mechanism consists of three major steps: initialization, which prepares alternative populations for the next steps; identifying seven nearest neighbors; and orientation change which adjusts velocity and position of particles based on those neighbors and selects the best alternative. Because of this collective response mechanism, the Starling PSO explores a wider area of the search space and thus avoids suboptimal solutions. We tested the algorithm with commonly used numerical benchmarking functions as well as applying it to a real world application involving data clustering. In these evaluations, we compared Starling PSO with a variety of state of the art algorithms. The results show that Starling PSO improves the performance of the original PSO and yields the optimal solution in many numerical benchmarking experiments. It also gives the best results in almost all clustering experiments.

1. Introduction

Swarm intelligence is a meta-heuristic algorithm which is widely used nowadays. The main characteristic of swarm intelligence is that high quality problem solutions emerge from the collaborative behavior of individuals within a swarm. In general, the meta-heuristic framework starts with an initial solution called the current solution and a set of neighborhood solutions, which are the candidate solutions. Each candidate solution in the neighborhood set is selected and evaluated. This evaluation involves calculating or estimating the performance of the candidate solution(s) and comparing them with the performance of the current solution and sometimes with each other. Based on this evaluation, if a candidate solution is better than the current solution, it will be accepted. If the current solution is better than the candidate solution, the candidate solution will be rejected. Each member of the swarm discovers and/or evaluates some candidates. The above process is repeated until the optimal solution is found [1].

To effectively discover optimal solutions, a meta-heuristic requires two mechanisms, exploration and exploitation. Exploration mechanisms help the algorithm discover candidate solutions from the whole search space rather than just specific regions. Exploitation focuses on retrieving the best solution from a specific area of the search space. In other words, exploration mechanisms identify possible alternative areas into the search space. Exploitation mechanisms prune to find the optimal solution in the explored area. If the best solution identified by the exploitation is unacceptable, the exploration mechanism finds another area in the search space to examine. An algorithm which provides good performance in both exploration and exploitation is more likely to reach the optimal solution in every kind of problem.

Popular swarm intelligence algorithms include Ant Colony Optimization (ACO) [2], Particle Swarm Optimization (PSO) [3] and Artificial Bee Colony algorithm (ABC) [4].

Ant Colony Optimization finds the optimal solution by exploring the candidate paths that ants travel from their nest to food sources. Each path represents a candidate solution. At the beginning, the chance that an ant will select any candidate path is equal for all paths. The ant colony exploits to find the solution by adding pheromone traces to the better candidate paths. The more pheromone deposited on a path, the higher the probability that other ants will select that path. After some iterations of route selection, the paths of all ants converge to the optimal candidate. ACO

* Corresponding author.
E-mail address: boon@kmutt.ac.th (B. Sirinaovakul).

http://dx.doi.org/10.1016/j.asoc.2015.06.052
1568-4946/

was originally proposed to solve combinatorial problems such as the traveling salesman problem.

Artificial Bee Colony (ABC) algorithm finds the optimal solution by sending employed bees to find food sources, which represent candidate solutions. The employed bees come back to report the nectar amount in the food sources they found to the onlooker bees in their hive. The higher the nectar mount, the higher the probability that the onlooker bees will prefer that food source. The onlooker bees investigate the selected food source as well as food sources in its neighborhood. Some poor food sources might be abandoned after some time. ABC also improves the exploration ability of the optimization algorithm by using scout bees. Scout bees introduce a mutation process to the algorithm by searching for new candidate food sources in previously unexplored areas of the search space.

Particle Swarm Optimization adds steps to update the current best solution into the meta-heuristic to increase the exploration ability in order to solve numerical problems. Instead of using individual ants to find the solution, Particle Swarm Optimization uses particles in the swarm moving around the solution space to find the global best position (or solution). The position of a particle represents an alternative solution in a multidimensional solution space. Each particle is assigned a velocity, which changes over time and which is used to update the particle's position over iterations until the optimal solution is found. However, because the exploration capability of this algorithm is limited, it sometimes fails to find the global best solution.

Particle Swarm Optimization provides good exploitation performance. However, PSO is inferior in exploration and thus tends to converge prematurely to a local optimum, without considering more promising areas of the search space. Many researchers have introduced methods to improve the exploration mechanisms of PSO. Their work can be categorized into two groups: work that modifies the value of parameters in the PSO updating equation and work that introduces additional techniques to the original PSO process.

Van den Bergh and Engelbrecht [5] proposed an algorithm called Guaranteed Convergence PSO (GCPSO). In their algorithm, the global best solution is updated using a new equation to propose an alternative solution to the global best. The new algorithm tends to improve the performance of exploitation. However, the performance in exploration does not improve. Multi-Start PSO (MPSO) [6] is an enhancement of GCPSO. The process of MPSO is the same as GCPSO but the swarm is re-initialized if the solution converges to a local optimum. This modification improves the performance in exploration compared to GCPSO while maintaining the good performance in exploitation. However, performance of MPSO decreases when the dimensionality of the problem increases [7].

Some research has focused on improving exploration performance in PSO by adding communication among particles instead of communication regarding only global best solution. He [8] proposed PSO with Passive Congregation (PSOPC). The algorithm tries to introduce communication among individuals to avoid local optima. Velocity updating in PSOPC is based not only on personal best and global best but also on a randomly chosen particle position. Mendes [9] proposed Full Informed PSO (FIPSO). Each particle in FIPSO communicates not just with the particle whose position represents the global best but with all of its neighboring particles. The aforementioned methods can prevent PSO from converging prematurely but the exploitation performance sometimes decreases.

To improve the exploration performance while maintaining the exploitation performance of PSO, the concept of multi-swarm has been adoptedby many researchers. In species-based PSO (SPSO) [10] the population is separated into sub-swarms by similarity. The process of standard PSO is applied to each sub-swarm separately. Cooperative PSO (CPSO) [11] uses the multi-swarm concept in another manner. In CPSO, the solution vector is separated into sub-vectors. Each sub-swarm is responsible for finding the solution for each sub-vector. The solutions from each sub-swarm are then combined at the end.

The previously mentioned algorithms divide the population into sub-swarms but each sub-swarm works separately. There is no interaction between sub-swarms. To add the information sharing into the multi-swarm concept, Jie [12] developed a Knowledge-based Cooperative PSO (KCPSO). The information of the whole swarm and each sub-swarm is used to adapt the process by following a set of rules. Jiang [13] proposed Master-Slave Swarms Shuffling Evolution based on PSO (MSSE–PSO). The particles in MSSE–PSO are ranked by fitness of their current solution and are divided into sub-swarms. The sub-swarm with the best fitness is called the master swarm and the others are called slave swarms. A particle in master swarm takes information from the best position in the slave swarms to update its position.

Dividing the population into sub-swarms can add diversity to the population and thus improve the exploration performance. However, to maintain the exploitation performance in each sub-swarm, the algorithm needs a larger population size which increases the execution time of the algorithm.

Many recent studies focus on developing a new particle update equation. Xi [14] proposed Weighted Quantum-behaved PSO (WQPSO). The local attractor, which is the combination of a particle's own personal best, and global best, plus the mean of personal bests are used to update the particle's position. Each personal best is weighted according to the rank of its fitness compared to the other personal bests. Chuang [15] proposed Chaotic Catfish PSO (C-Catfish PSO). To accelerate the convergence, a logistic map is used instead of random numbers in the updating equation. C-Catfish also avoids local optima by randomly creating new particles to replace the worst 10% of the particle population. C-Catfish PSO performs best among the PSO-type algorithm on some specific benchmark functions.

Our research proposes a new variation of PSO. The objective is to increase PSO's capability in exploration without decreasing the capability in exploitation. We introduce Starling Particle Swarm Optimization (Starling PSO) which is a collective response process. The concept of applying collective behavior of starlings to PSO has been previously proposed in [16]. In their proposed method, the original velocity updating equation is replaced by new equation which mimics the collective behavior of starlings. In their experiments, the proposed algorithm performed better than the original PSO on unimodal benchmark functions. However, for multimodal benchmark functions, original PSO still performed better. As we have mentioned before, PSO is inferior in exploration in handling complex problems such as multimodal functions. In our work, we maintain the original velocity updating equation of PSO most of the time, except when stagnation occurs. In other words, when the global best stagnates in a local optimum, the position and velocity of particles are adjusted by using a mechanism similar to the collective behavior of starlings.

2. Original Particle Swarm Optimization overview

Particle Swarm Optimization (PSO) [3] [17] searches for a solution which is a multi-dimensional vector of real values. Each solution is modeled by the position of a particle in a multidimensional solution space. Each particle has its own velocity which it uses to update its position over iterations. Along the way, each particle remembers its personal best which is the best solution (i.e. location) which that particle has found so far. Among the local best solutions from all particles, the one with the highest fitness is remembered as the global best. The particles in the swarm move around the solution space by updating their positions and