Expert Systems With Applications 83 (2017) 63–78

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Memory based Hybrid Dragonfly Algorithm for numerical optimization problems

Sree Ranjini K.S.[a,*], S. Murugan[b]

[a] Homi Bhabha National Institute, Indira Gandhi Centre for Atomic Research, Kalpakkam, India
[b] Remote handling, Irradiation & Robotics Division, Indira Gandhi Centre for Atomic Research, Kalpakkam, India

ARTICLE INFO

Article history:
Received 5 January 2017
Revised 27 February 2017
Accepted 17 April 2017
Available online 18 April 2017

Keywords:
Dragonfly algorithm
Particle Swarm Optimization
Hybridization
Benchmark functions
Engineering problems
Friedman's test

ABSTRACT

Dragonfly algorithm (DA) is a recently proposed optimization algorithm based on the static and dynamic swarming behaviour of dragonflies. Due to its simplicity and efficiency, DA has received interest of researchers from different fields. However, it lacks internal memory which may lead to its premature convergence to local optima. To overcome this drawback, we propose a novel Memory based Hybrid Dragonfly Algorithm (MHDA) for solving numerical optimization problems. The *pbest* and *gbest* concept of Particle Swarm optimization (PSO) is added to conventional DA to guide the search process for potential candidate solutions and PSO is then initialized with pbest of DA to further exploit the search space. The proposed method combines the exploration capability of DA and exploitation capability of PSO to achieve global optimal solutions. The efficiency of the MHDA is validated by testing on basic unconstrained benchmark functions and CEC 2014 test functions. A comparative performance analysis between MHDA and other powerful optimization algorithms have been carried out and significance of the results is proved by statistical methods. The results show that MHDA gives better performance than conventional DA and PSO. Moreover, it gives competitive results in terms of convergence, accuracy and search-ability when compared with the state-of-the-art algorithms. The efficacy of MHDA in solving real world problems is also explained with three engineering design problems.

1. Introduction

Optimization process have become an integral part of engineering and business problems. The purpose of the optimization can be for the maximization of efficiency, performance, productivity or social welfare. In real world, resources, time and money are always limited and hence there is an inevitable need for finding out solutions for optimal usage of these valuable resources under various constraints (Yang, 2014a). In recent years stochastic algorithms have been gaining significance in producing fast, low cost and robust solution to complex optimization problems (Dorigo & Thomas, 2004). Compared to conventional deterministic approach, they don't require any gradient information and are simple and easy to implement (Blum & Li, 2008). Among the stochastic optimization algorithms, swarm intelligence (SI) based optimization techniques have attracted the attention of researchers world wide. A swarm is characterized by a group of self-organized and decentralized system of non-complex individuals or agents interacting among themselves and with their environment for survival, hunting, navigation or foraging. It can be school of fish, flock of birds, colonies of ants etc. SI based algorithms models the collective behaviour of these individuals to solve complex optimization process. Even though as individuals, these agents have limited operational capability, they tend to outperform in accomplishing the desired task by interacting among themselves and with the environment using their own specific behavioural patterns. Literature review on the SI based optimization algorithms reveals their effectiveness in solving complex optimization problems in different fields of study. Ant colony optimization inspired by the foraging behaviour of the ants was found to be very effective in solving structural optimization problems (Luh & Lin, 2009), traffic area control problems (Sattari, Malakooti, Jalooli, & Noor, 2014) and also in the field of genomics (Greene, White, & Moore, 2008). Particle swarm algorithm (PSO) is well known optimization algorithm mimicking the social behaviour of bird flocking or fish schooling (Eberhart & Kennedy, 1995). The effectiveness of PSO in solving bi level programming problems (Kuo & Huang, 2009), electric power systems (AlRashidi & El-Hawary, 2009), offshore heavy oil reservoir (Wang & Qiu, 2013), and image processing (Omran, Engelbrecht, & Salman, 2006) is clearly explained in the literature. Bat

* Corresponding author.
E-mail addresses: srks@igcar.gov.in, sreeranjiniks53@gmail.com (S.R. K.S.), murugan@igcar.gov.in (S. Murugan).

http://dx.doi.org/10.1016/j.eswa.2017.04.033
0957-4174/

algorithm (Yang, 2010b), Firefly algorithm (Yang, 2009), Krill Herd (Gandomi & Alavi, 2012), Whale optimization algorithm (Mirjalili & Lewis, 2016), Grey wolf optimization (Mirjalili, Saremi, Mirjalili, & dos S. Coelho, 2016), Ageist Spider Monkey optimization (Sharma, Sharma, Panigrahi, Kiran, & Kumar, 2016), Moth search optimization (Wang, 2016), Competitive optimization algorithm (Kashani, Gandomi, & Mousavi, 2016)are some of the popular swarm based meta heuristic algorithms.

With development of numerous optimization algorithms, it is difficult to test and determine which algorithm is most suitable for solving a particular optimization problem. This is because most of the algorithms works on generalized concept and don't have domain knowledge specific to each problem. Hybridization process gains importance in this situation, as it combines the desirable properties of different approaches to mitigate their individual weaknesses (Thangaraj, Pant, Abraham, & Bouvry, 2011). Lesser computation, improvement of solution accuracy, enhancement of algorithm stability and the handling of searching convergence can be considered as targets of hybridization and improvement process. A number of hybridized versions of many conventional algorithms have evolved recently as a part of this process. They tend to show shows remarkable improved performance compared to their traditional counterparts. Nasir, Tokhi, and Ghani (2015) proposed adaptive chemotactic step size based bacterial foraging algorithm depending on individual bacterium fitness value, index of iteration and index of chemotaxis. An improved version of Differential Evolution (DE) algorithm combining different mutation operators and empowered by co-variance adaptation matrix evolution strategy algorithm as a local search is introduced by Elsayed, Sarker, and Essam (2013). Improved PSO based on adaptive inertial weight, introduced in the year 2011 (Nickabadi, Ebadzadeh, & Safabakhsh, 2011) and is found to be very effective in solving real engineering problems. Li, Wang, Yan, and Li (2015) proposed a hybrid PS-ABC combining the local search capabilities of PSO and global search capabilities of Artificial Bee Colony (ABC) algorithm and found that the hybrid algorithm is a better solution than the parent algorithms in terms of speed, convergence and robustness. Nabil (2016) investigated the performance of flower pollination algorithm incorporating colonal selection operator and validated the improved performance through standard benchmark functions. Garg (2016) proposed hybrid optimization algorithm based on Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) for solving constrained optimization problems. A hybrid ICE- SA algorithm based on Imperialist Competitive algorithm (ICE) and Simulated annealing (SA) is proposed for multi source multi product location-routing-inventory problem by Ghorbani and Jokar (2016).

The objective of the paper is to introduce a novel hybrid version of Dragonfly algorithm (DA) which is a recently evolved meta-heuristic optimization algorithm proposed by Mirjalili (2016). Because of its simple and easy implementation, DA has been used to solve many real world optimization problems such as extension of RFID network lifetime (Hema, Sankar, & Sandhya, 2016), range based wireless node equalization (Daely & Shin, 2016), threshold for multilevel segmentation of digital images (Sambandam & Jayaraman, 2016), and photo-voltaic systems (Raman, Raman, Manickam, & Ganesan, 2016). However DA does not keep track of its best positions in previous generations which limits its exploitation capability and causes premature convergence to local optima. Even though global search capability of DA is good with randomization and static swarm behaviour, the local search capability is limited which causes the solutions to get trapped in local optima. In order to overcome these shortcomings, we propose a novel Memory based Hybrid Dragonfly Algorithm (MHDA). The proposed MHDA works in two stages, in the first stage, memory element is incorporated in DA algorithm so as to store the coordinates in the solution space which are associated with the best solution (fitness) that has achieved so far by the dragonfly and in the second stage PSO is initialized with this best solutions for further exploitation. Thus exploration capability at the initial stages and exploitation capability at the later stages is guaranteed by iteration level hybridization process and ensures to obtain the global optimum with increased accuracy. The performance of MHDA is compared with other state-of-the-art algorithms on two benchmark function suites. Suite-I consist of benchmark functions commonly found in literature and Suite-II consist of CEC 2014 functions. The significance of the experimental results is proved by statistical analysis.

The paper is organized as follows. Section 2 describes the conventional DA and PSO algorithms. Section 3 describes about the proposed MHDA and its functioning. Section 4 describes the performance evaluation and detailed analysis of MHDA. Section 5 discusses the application of MHDA on engineering problems, comparison of its performance with other conventional algorithms and its statistical results. Finally, conclusion and future scope are described in Section 6.

2. Related work

2.1. DA

Dragonfly algorithm is inspired by the unique and superior swarming behaviour of dragonflies. The dragonfly swarms for hunting and migration. Hunting swarm behaviour which is otherwise known as static swarm behaviour is characterized by the formation of small group of dragonflies moving locally and abruptly changing the steps. Migratory swarm behaviour which is otherwise known as dynamic swarm is characterized by a massive number of dragonflies flying in one direction over long distances. Static Swarm and dynamic swarms represents exploitation and exploration capabilities of DA. The behaviour of dragonfly follows the principles of separation, alignment, cohesion, distraction from the enemies and attraction towards the food. Each dragon fly in the swarm corresponds to the solution in the search space. Swarm movement of dragonfly is determined by five different operators such as Separation, Alignment, Cohesion, Attraction towards food sources and distraction towards enemy sources (Reynolds, 1987). Separation (S_i) which refers to the static collision avoidance of individuals from other individuals in the neighbourhood. Alignment (A_i) refers to the velocity matching of individuals to other individuals in neighbourhood. Cohesion (C_i) refers to the tendency of individuals towards the center of the mass of the neighbourhood. Suitable weights are assigned to each operators and they are adaptively tuned to ensure the convergence of dragonflies towards optimal solution. The neighbouring radius of the dragonflies also increases as the process of optimization progresses. The mathematical implementation of DA can be explained as follows.

Consider population of dragonflies of size N. The position of *i*th dragonfly is given by Eq. (1)

$$X_i = (x_i^1, x_i^d \ldots, x_i^N) \tag{1}$$

where i = 1,2,3... N, x_i^dcorresponds to the position of the *i*th dragon fly in *d*th dimension of the search space and N is the number of search agents.

The fitness function is evaluated based on the initial position values which is randomly generated between the lower and upper bounds of the variables. The weights for separation (s), alignment (a), cohesion(c), food (f) and enemy (e) factors for each dragonfly is initialized randomly. For updating the position and velocity of dragonflies separation, alignment and cohesion coefficients are calculated using Eqs. (2)–(4)