Expert Systems With Applications 99 (2018) 56–70

Contents lists available at ScienceDirect

Expert Systems With Applications

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

ASCA-PSO: Adaptive sine cosine optimization algorithm integrated with particle swarm for pairwise local sequence alignment

Mohamed Issa[a,1], Aboul Ella Hassanien[b,1,*], Diego Oliva[c,1], Ahmed Helmi[a], Ibrahim Ziedan[a], Ahmed Alzohairy[d]

[a] Computer Engineering and Systems Department, Faculty of Engineering, Zagazig University, Egypt
[b] Faculty of Computers and Information, Cairo University, Egypt
[c] Departamento de Ciencias Computacionales, Universidad de Guadalajara, Mexico
[d] Genetic Department, Faculty of Agriculture, Zagazig University, Egypt

ARTICLE INFO

Article history:
Received 16 September 2017
Revised 12 January 2018
Accepted 13 January 2018

Keywords:
Sine cosine algorithm (SCA)
Particle swarm optimization (PSO)
Meta-heuristics algorithms
Pairwise local alignment
Longest consecutive substrings
Smith-Waterman alignment algorithm

ABSTRACT

The sine cosine algorithm (SCA), a recently proposed population-based optimization algorithm, is based on the use of sine and cosine trigonometric functions as operators to update the movements of the search agents. To optimize performance, different parameters on the SCA must be appropriately tuned. Setting such parameters is challenging because they permit the algorithm to escape from local optima and avoid premature convergence. The main drawback of the SCA is that the parameter setting only affects the exploitation of the prominent regions. However, the SCA has good exploration capabilities. This article presents an enhanced version of the SCA by merging it with particle swarm optimization (PSO). PSO exploits the search space better than the operators of the standard SCA. The proposed algorithm, called ASCA-PSO, has been tested over several unimodal and multimodal benchmark functions, which show its superiority over the SCA and other recent and standard meta-heuristic algorithms. Moreover, to verify the capabilities of the SCA, the SCA has been used to solve the real-world problem of a pairwise local alignment algorithm that tends to find the longest consecutive substrings between two biological sequences. Experimental results provide evidence of the good performance of the ASCA-PSO solutions in terms of accuracy and computational time.

1. Introduction

In the past two decades, nature-inspired optimization methods, also called meta-heuristic algorithms (MAs), have attracted the attention of researchers from a variety of fields (Boussaïd, Lepagnot, & Siarry, 2013). MAs search for optimal solutions based on a search strategy that imitates a natural behavior. In this sense, different metaphors are created in MAs, such as genetic algorithms (GAs) (Holland, 1992) and differential evolution (DE) (Storn & Price, 1997), which are based on evolutionary theory. Additionally, physically based algorithms include methods such as the sine cosine algorithm (SCA) (Mirjalili, 2016), the ions motion optimization (IMO) (Javidy, Hatamlou, & Mirjalili, 2015) and the gravitational search algorithm (GSA) (Rashedi, Nezamabadi-Pour, & Saryazdi, 2009). Another group of methods is based on insects and other animals and includes particle swarm optimization (PSO) (Kennedy, 1995), artificial bee colony (ABC) (Karaboga & Akay, 2009) and moth-flame optimization (MFO) (Mirjalili, 2015). Other algorithms imitate human concepts (or creations), such as the mine blast algorithm (MBA) (Sadollah, Bahreininejad, Eskandar, & Hamdi, 2013) and teaching learning-based algorithm (TLBO) (Rao, Savsani, & Vakharia, 2011).

Two contradictory factors must be considered in designing new MAs: the exploration of the search space (diversification) and the exploitation of prominent regions (intensification). Exploration is used to diversify the regions of the search space to ensure that all regions of the search space are evenly explored and that the search is not confined to a limited number of regions and, in addition, to avoid becoming trapped in local minima. Exploitation is the process of analyzing the bounded search area around the best solution in order to improve it. Balancing between exploration and exploitation is essential to enhancing the efficiency of a meta-heuristic algorithm.

In the related literature, a substantial number of meta-heuristics can be attributed to the no-free-lunch (NFL) theorem (Wolpert & Macready, 1997), which states that the success of an optimization technique in addressing a specific problem

* Corresponding author.
E-mail addresses: aboitcairo@fci-cu.edu.eg (A.E. Hassanien), doliva@ucm.es (D. Oliva).
[1] Scientific Research Group in Egypt, (SRGE) http://www.egyptscience.net.

https://doi.org/10.1016/j.eswa.2018.01.019
0957-4174/

does not guarantee success in different optimization problems with different natures and types. Hence, according to NFL, the research in meta-heuristics has three main directions: (1) the improvement of current methods, (2) the creation of new algorithms, and (3) the combination of different meta-heuristics. The first direction modifies the operators to enhance the performance of the existing approaches, such as chaotic maps (Wang, Guo, Gandomi, Hao, & Wang, 2014; Petrović, Mitić, Vuković, & Miljković, 2016), local searches (Cao, Li, & Chaovalitwongse, 2017; Premalatha & Natarajan, 2008) and evolutionary operators (Wang, Guo, Duan, Liu, & Wang, 2012; Wang, Deb, Gandomi, & Alavi, 2016). The second direction proposes new optimization mechanisms inspired by different behaviors, such as the slap swarm algorithm (SSA) (Mirjalili et al., 2017), whale optimization algorithm (WOA) (Mirjalili & Hatamlou, 2016) and multi-verse optimizer (MVO) (Mirjalili & Lewis, 2016). The most recent direction for meta-heuristics is hybridizing different optimization algorithms to benefit from each of their advantages (Garg, 2016; Güçyetmez & Çam 2016; Santra, Mukherjee, Sarker, & Chatterjee, 2016; Yang, Wang, Lin, & Chen, 2016).

One of the main problems of MAs is that they commonly have parameters that must be tuned according to the problems to be solved. In this context, the SCA possesses several parameters that must be tuned to maximize performance in the optimization process. Tuning these parameters is challenging, and if they are not correctly selected, the algorithms can become trapping in local optima or premature convergence. However, the main advantage of the SCA is its power of exploration of the search space. SCA has been used in applications and has efficient performance on problems such as handwritten Arabic text (Mudhsh, Xiong, El Aziz, Hassanien, & Duan, 2017), photovoltaic systems (Kumar, Hussain, Singh, & Panigrahi, 2017) and detection of galaxies using image retrieval (Abd ElAziz, Xiong, & Selim, 2017). However, according to the NFL theory, the SCA would not perform well on all optimization problems, including finding the longest consecutive substrings between two biological sequences. This article introduces an enhanced version of the SCA and merges it with PSO, a traditional optimization algorithm inspired by the behavior of flocking birds or schooling fish. The main advantages of PSO are robustness, the need for few parameters and the efficient exploitation of the search space.

Considering the above, performance of the SCA is improved by integrating the features of PSO to exploit the optimal solutions with the capabilities of the SCA to explore the entire search space. This hybridization provides a good balance between exploration and exploitation throughout the iterative process.

The proposed algorithm is called ASCA-PSO and consists of two layers: the bottom layer is responsible for exploration the search space, which is performed by the search agents of the SCA; the top layer is responsible for the exploitation of the best solution found by the bottom layer based on the search agents of the PSO. The proposed approach enables a good diversification of the population and preserves the best information on the position at each iteration.

To verify the performance of the proposed algorithm, it was tested over a set of mathematical optimization problems with different degrees of difficulty. Moreover, the local sequence alignment (LSA) problem is used as a case of study for testing the improved ASCA-PSO. The experimental results on both the mathematical and LSA problem provide evidence of the accuracy of the proposed method in complex optimization problems and show that it maintains balance with regard to the computational time.

The remainder of this paper is organized as follows. Section 2 describes the preliminaries for the standard SCA and PSO. The proposed technique is presented in Section 3. Section 4 describes the results of the proposed algorithm compared with the testing benchmark functions. Section 5 provides the fragmentation local sequence alignment technique based on a proposed algorithm for comparison with the SCA. Finally, Section 6 presents the conclusions.

2. Preliminaries

This section provides a brief explanation of the basic framework of PSO and the SCA along with some of the fundamental concepts.

2.1. Particle swarm optimization (PSO)

Swarm optimization algorithms are stochastic population-based search methods that mimic the behavior of fish schooling, birds flocking and other grouping behaviors. The search strategy of these algorithms is mainly based on global communications among the individuals of the population, where the particles adapt their movements toward the particle that finds the best solution. PSO generates a swarm of particles, and each particle has a position (a solution to the problem) in the search space. All particles tune their movements according to Eqs. (1) and (2) toward the particle (P^{gbest}) that has the best position (the global best solution) and the best personal position (P_i^{best}) for each particle pass during the previous iterations.

$$v_i(t+1) = w * v_i(t) + c_1\ rand\ (P^{\mathrm{best}}{}_i - P_i(t)) + c_2\ rand\ (P^{\mathrm{gbest}} - P_i(t)) \tag{1}$$

$$P_i(t+1) = P_i(t) + v_i(t+1) \tag{2}$$

Here, v_i is the velocity of the ith particle, c_1 is the best local position weight coefficient, and c_2 is the global best position weight coefficient. w is the inertia coefficient that controls the effect of the previous velocity on the new velocity. P_i is the position of particle i, t is the iteration number, and *rand* is a uniformly distributed random variable in the range (0–1). P_i^{best} is the best local position (solution) found by particle P_i, and P^{gbest} is the best solution found in the whole swarm.

PSO has several advantages, such as robustness and information interchange among particles, which provides a high probability of achieving a near-optimal solution with a reasonable convergence speed. PSO has been used to solve many optimization problems, such as the optimization of the parameters of electrical motors (Calvini, Carpita, Formentini, & Marchesoni, 2015), solar cell design (Khanna, Das, Bisht, & Singh, 2015) and surgical robot applications (Tuvayanond & Parnichkun, 2017). The steps of PSO are summarized in Algorithm 1.

The time complexity of a PSO is $O(T^*n^*c_{\mathrm{pso}})$, where n is the number of particles, c_{pso} is the time cost of updating the position of one particle and T is the number of iterations.

2.2. Sine-cosine optimization algorithm (SCA)

SCA is a population-based optimization algorithm that depends on sine and cosine operators for updating the movement of the

Algorithm 1 Particle swarm optimization.

1: Initialize a set of population solutions (P_i), initial velocity (v_i) and algorithm's parameters (c_1, c_2 and w)
2: **Repeat**
3: Evaluate the objective function based on population solutions
4: Update the best local solution for each particle (P_i^{best})
5: Update the best global solution over all particles (P_{gbest})
6: Update the next position of population solutions using Eqs. (1) and (2)
7: **Until** ($T <$ maximum number of iterations)
8: Return the best solution (P^{gbest}) obtained as the global optimum