IOP Conf. Series: Materials Science and Engineering **864** (2020) 012065 doi:10.1088/1757-899X/864/1/012065

A New Hybrid Algorithm Based on ABC and PSO for Function Optimization

Chang-Feng Chen[1], Azlan Mohd Zain[2], Li-Ping Mo[1] and Kai-Qing Zhou[1]

[1]College of Information Science and Engineering, Jishou University, Jishou, Hunan, China

[2] UTM Big Data Centre, Ibnu Sina Institute for Scientific and Industrial Research, University Teknologi Malaysia, Johor, Malaysia

Email:740839301@qq.com

Abstract. Artificial bee colony algorithm (ABC) and particle swarm optimization (PSO) are both famous optimization algorithms that have been successfully applied to various optimization problems, especially in function optimization. Those two algorithms have been attracting more and more research interest because of their efficiency and simplicity. However, PSO has poor exploration capabilities and thus is easy to fall into the local optimum; Likewise, ABC has low convergence speed. To address these shortcomings, firstly, we improved the ABC with the combination of greedy selection and crossover, secondly, a sine-cosine method will be used to help PSO jump into local optimal. Finally, a new hybrid algorithm based on improved ABC and PSO are proposed. Moreover, four functions are used to verify the effectiveness of the proposed algorithm, and the results show that, compared with other well-known algorithms, ABC-PSO is more efficient, faster and more robust in function optimization.

1. Introduction

Optimization problem tend to be more and more complex as the development of science and technology. Traditional optimization methods seem to be inefficient and cannot meet the requirements of solving some practical problems. Therefore, the development and research of different optimization methods has attracted increasing interest from researchers in the past two decades. For example, nature-inspired algorithm have been successfully used to solve optimization problems in real life, including particle swarm optimization(PSO)[1], artificial bee colony algorithm(ABC)[2], ant colony algorithm(ACO), genetic algorithm(GA) and differential evolution(DE)[3].

Those algorithms has attracted growing attention due to their flexible structure and easy application. Among existing nature-inspired algorithm, ABC and PSO have shown considerable success in solving numerical optimization problem. Nevertheless, according to "no free lunch" theorem [4], no algorithm can achieve all the advantages. ABC and PSO also face some challenge.

The rest of the paper is organized as follows: section 2 briefly describes the original ABC and PSO algorithms. The detailed process of hybridization of ABC-PSO are explained in section 3[5]. Section 4 presents the simulation experiments and compares the results of other optimization algorithm with the original algorithms and some variants of ABC and PSO. Finally, the conclusion are summarized and the future work are presented in section 5.

2. ABC and PSO algorithm

2.1. ABC algorithm

ABC is a nature-inspired swarm intelligence based metaheuristic algorithm developed by Karaboga[2], which mimics the foraging behaviour of honey bees. ABC has the advantages of robustness, effectiveness, and easy implementation[6]. The colony of bees in ABC can be classified as three groups: employed bees, onlookers and scouts. The process of ABC are described as follows:

Step1: All bees are scouts bees, and they know nothing about the nectar sources, so those scouts bees will search the solution space randomly, and get the initial solutions. This process can be described as equation (1):

$$X_{ij} = X_j^{\min} + rand(\mathrm{X}_j^{\max} - \mathrm{X}_j^{\min}) \tag{1}$$

$$new\_X_j^i = X_j^i + \phi_j^i(X_j^i - X_j^i) \tag{2}$$

Where $i \in \{1,2,\ldots,\mathrm{SN}\}, j \in \{1,2,\ldots,\mathrm{D}\}$; SN describes the number of nectar sources; d is the number of dimensions; rand is a uniform random value between 0 and 1[7].

Step2: According to the fitness of solutions produced in step 1. The scout bees will be divided into two parts,To be specific, those bees with high nectar sources will be transformed into employed bees, and the rest will become onlooker bees. The employed bees will search new food in their neighbour area. The process can be explained as equation (2):

Where $k \in \{1,2,\ldots,\mathrm{SN}\}$, is[8] selected randomly from the population and must be different with i; ϕ is randomly produced from [-1,1].

Step3: The fitness value of new nectar source and old one will be compared, and a greedy selection will be used to select the better one. The progress can be described as the following equation (3):

$$new\_X_i = \begin{cases} 1, f(new\_X_i) \geq f(X_i) \\ 0, f(new\_X_i) \leq f(X_i) \end{cases} \tag{3}$$

$$P_i = fit_i / \sum_{k=1}^{SN} fit_k \tag{4}$$

Step4: Employed bees will fly back to a specific dance area, and share nectar information with onlooker bees. Then, according to the probability value of generating equation (4), choose to follow a nectar source.

Compared with other well-known algorithms such as GA and DE, it shows competitive advantages in various numerical problems[9]. However, it still suffers from poor exploitation ability and low convergence speed while dealing with complex problems. To improve the exploitation ability of ABC, many researchers have proposed various strategies. For example, Alkm and Erdal (2017) developed a modified ABC algorithm (MABC) which has solution acceptance rule and probabilistic multisearch to address global optimization problem[10].

2.2. *PSO algorithm*

PSO is also a nature-inspired heuristic algorithm presented by Kennedy et al. in 1995, which simulate the social behaviour of bird flocking and fish schooling. Since PSO was proposed, it has been used to solve many practical problems such as engineering optimization, pattern recognition and automatic control [11].

Compared with other optimization algorithms, PSO can obtain more accurate results at a cheaper cost. PSO is a robust, straightforward and efficient algorithm, in which each particle represents a potential solution. Each particle consists of two vectors: a position vector and a velocity vector. During the optimization process, two vectors can be updated by the following equation (5) and (6):

$$v_{iD}(\mathrm{t}+1) = w.v_{tD}(\mathrm{t}) + \mathrm{c}_1.\mathrm{r}_1.(\mathrm{P}_{iD}(\mathrm{t}) - x_{iD}(\mathrm{t})) + \mathrm{c}_2.\mathrm{r}_2.(\mathrm{P}_{gD}(\mathrm{t}) - x_{iD}(\mathrm{t})) \tag{5}$$

$$x_{iD}(\mathrm{t}+1) = x_{iD}(\mathrm{t}) + v_{iD}(\mathrm{t}+1) \tag{6}$$

Where w is the inertial weight, c1 and c2 denote the accelerate coefficients, r1 and r2 are two random number in the range [0,1], $P_i = (\mathrm{p}_{i1}, \mathrm{p}_{i2}, \ldots, \mathrm{p}_{iD})$ and $P_g = (\mathrm{p}_{g1}, \mathrm{p}_{g2}, \ldots, \mathrm{p}_{gD})$ are prvious optimal position and global position respectively [12].

Even though, basic PSO also has some disadvantages. In this case, it tends to fall into a local optimum and is prone to premature convergence. Accordingly, some strategies have been designed to enhance the origianl PSO. For example, Wang, Zhang, Li and Lin (2017) propose a hybrid PSO using adaptive learning strategy. Simultaneously, Chen, Zhou and Liu (2017) use chaotic dynamic weight to improve PSO for numerical optimization.

3. Hybridization and improvements of ABC and PSO

As mentioned earlier, basic ABC has drawbacks of poor exploitation ability and low convergence rate, and original PSO has poor exploration ability and premature convergence. To avoid disadvantages of these two algorithm, three modifications are performed.

Firstly, after a new nectar source is generated, the greedy selection will be used to determine whether the original solution should be discarded. But sometimes poor solutions can also lead bees to unexpectedly good solutions. Consequently, ABC-PSO algorithm selects a predetermined number of nectar sources into a pool, and the nectar sources in the pool are randomly divided into several pairs. Each pair reproduces two children by crossover. Then the greedy selection is used to choose the better one. This process can be described as the following equation (7) and (8):

$$s_u = p \times p_{1u} + p \times p_{2u} \tag{7}$$

$$s_v = \frac{p_{1v} + p_{2v}}{| p_{1v} + p_{2v} |} | p_{1v} | \tag{8}$$

Where $u = v = (1, 2, ..., m/2)$, is the half of the predetermined number of nectar source. P_1 and P_2 are a pair of nectar source. S_u and S_v are the offspring of crossover.

Secondly, in PSO algorithm, the new velocity and position are both generated from the previous local and global best positions, which will lead to the premature of the optimization algorithm.

IOP Conf. Series: Materials Science and Engineering **864** (2020) 012065 doi:10.1088/1757-899X/864/1/012065

Therefore, the generation of new velocities is adjusted by using sine and cosine algorithms to avoid them falling into a local optimal state.

$$v_{iD}(t+1)\begin{cases} v_{iD}(t+1)=w.v_{iD}(t)+c.\sin(r_1)|P_{iD}(t)-x_{iD}(t)|+c.\sin(r_1)|P_{gD}(t)-x_{iD}(t)|; & r_2<0.5 \\ v_{iD}(t+1)=w.v_{iD}(t)+c.\cos(r_1)|P_{iD}(t)-x_{iD}(t)|+c.\cos(r_1)|P_{gD}(t)-x_{iD}(t)|; & r2\geq 0.5 \end{cases} \tag{9}$$

Where c is a balance parameter which can be used to guide the moving direction of following particle. With the help of c, the algorithm can be developed and explored steadily. r_1 and r_2 are random numbers. r_1 and r_2 are random numbers. r_1 represents the distance between the current particle and the next particle; r_2 is used to decide whether to use sine or cosine. For stable development and exploration purposes, c should be changed automatically using the following equation (10):

$$r_1 = w - t.\frac{w}{T} \tag{10}$$

Where t is the current iteration, and T is the maximum cycle of solutions without improvement. Thirdly, the modified ABC and enhanced PSO are combined. The pseudo-code of ABC-PSO algorithm is as follows:

```
1. Parameters initialization, SN, limit, T, t, ite, m, pi, pg
2. While t<T : do
3.    for i=1,2, …, SN   do
4.       if   t<limit then
5.            if   ite<m then
6.                 PSO phase(equations (9) and (10))
7.                 ite=ite+1
8.            else
9.                 modified onlooker bees(equations (7), (8) and (3))
10.           end if
11.      else
12.           scout bees(equation (1))
13.    end if
14.   end for
15. end while
16. output best solution
```

IOP Conf. Series: Materials Science and Engineering **864** (2020) 012065 doi:10.1088/1757-899X/864/1/012065

4. Experiments

4.1. Test functions

In order to study the effectiveness and efficiency of ABC-PSO, it was compared with 8 algorithms (ABC, MABC, PSO, DE, GA, WOA(whale optimization agorithm), SFLA and GA-SFLA) of 4 classic benchmark functions. Descriptions of these functions are given in Table 1, where D is the dimensional size and Opt is the global optimum [13].

Table 1. Four classic benchmark functions in experiment [14]

Function	Formulation	interval	D	Opt
Sphere	$f_1(x)=\sum_{i=1}^{D}(x_i^2)$	[-100,100]	30	0
Rosenbrock	$f_2(x)=\sum_{i=1}^{D-1}(100(x_{i+1}-x_i^2)^2+(x_i-1)^2)$	[-100,100]	30	0
Rastrigin	$f_3(x)=\sum_{i=1}^{D}(x_i^2-10\cos(2\pi x_i)+10)$	[-100,100]	30	0
Griewank	$f_4(x)=\frac{1}{4000}\sum_{i=1}^{D}x_i^2-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})+1$	[-100,100]	30	0

4.2. Experiment results and discussions

The experimental results are recorded and listed in Table 2, including the mean fitness value and the standard deviation of the values (std). The best mean and standard deviation of the target values for each function are shown in bold.

Table 2. Results of 9 algorithms over classic benchmark functions

Functions	F1		F2		F3		F4	
algorithms	mean	std	mean	std	mean	std	mean	std
ABC-PSO	**3.79e-24**	**1.62e-24**	**5.47e-20**	**1.71e-20**	**3.54e-22**	**3.26e-22**	**1.31e-23**	**4.36e-24**
ABC	2.8471e-16	2.4956e-17	1.3422e-18	1.1291e-18	6.63e-13	1.71e-12	8.4673e-20	1.2147e-19
MABC	7.6549e-18	5.4795e-18	3.1247e-19	2.4587e-18	1.5824e--20	3.9832e-21	3.2547e-22	1.9654e-22
PSO	0.531864	0.373814	44.2917	220.3997	39.5559	36.32899	0.0435754	0.0716879
DE	6.8475e-12	4.3647e-12	9.3129e-13	7.4735e-13	8.4217e-12	8.0326e-12	7.29147e-13	4.1395e-13
GA	1.9546	0.8361	0.6172	0.5048	0.3594	0.2907	0.9741	0.9796
SFLA	0.5821	0.4795	0.7709	0.6847	0.4009	0.3924	0.3279	0.4351
GA-SFLA	0.3462	0.3175	0.9876	0.9604	0.3102	0.3047	0.3107	0.4027
WOA	0.3514	0.2179	0.1217	0.0148	6.3126e-16	3.1423e-16	6.3479e-18	9.1475e-18

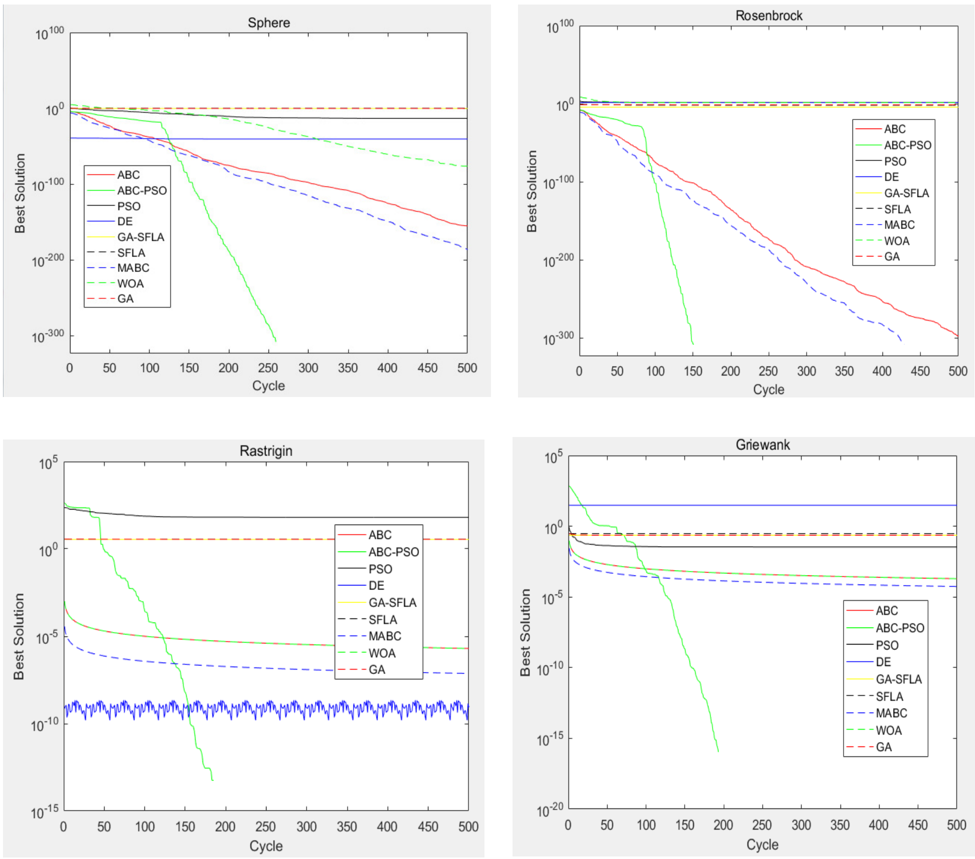

Figure 1. Convergence process of algorithms on four (f1-f4) benchmark functions.

As can be seen from the table, the proposed ABC-PSO can obtain the best results in terms of mean fitness value and standard deviation of the values. In particular, it shows great improvements in all four benchmark functions compared to the original ABC and PSO.

Figure 1. shows the convergence graphs of 9 algorithms (ABC, MABC, PSO, DE, GA, SFLA, GA-SFLA, WOA, ABC-PSO) on four test functions. It can be seen from Figure 1. (Sphere) that ABC-PSO does not seem to have a great superiority in optimizing the function of the Sphere between 0th -125th. However, after 200th iteration, the results are much better than other algorithms and the convergence speed is faster than other algorithms as well. What is more, the performance of ABC-PSO on Rosenbrock function is similar to Sphere. As a consequence, the conclusion can be drawn that hybrid ABC can avoid the optimization process without falling into a local optimum, and it converges faster. Through the analysis of the optimization circles in Figure 1. (Rastrigin) and (Griewank), it can be found that ABC-PSO can achieve higher accuracy and fast convergence, which can confirm the conclusion obtained again.

5. Conclusion and future research

In order to overcome the respective shortcomings of ABC and PSO, while taking full use of their advantages in function optimization, this paper proposes a new hybrid algorithm based on the modified two original algorithms. By using this algorithm to optimize the four functions to test the efficiency of

ABC-PSO and compare it with other algorithms with good performance, the results show that ABC-PSO can achieve much better solution as well as fast convergence speed, which meet the expectations.

Future work may be to extend the research of the ABC-PSO algorithm to get rid of all random processes and simplify the simulation process. At the same time, the ABC-PSO algorithm will be used for optimization problems in the real world, such as the traveling salesman problem and image processing.

Acknowledgments

The work was supported by the National Natural Science Foundation of Hunan Province (No.2019JJ40234), Research Foundation of Education Bureau of Hunan Province (Nos.19A414 and 18B317), the Special Project on Language Application of Hunan Provincial Language Committee (No. XYJ2019GB09).

References

[1] Z. Li, W. Wang, Y. Yan, and Z. Li 2015 Expert Syst. Appl., vol. 42, no. 22, pp. 8881–8895
[2] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga 2014 Artif. Intell. Rev., vol. 42, no. 1, pp. 21–57
[3] S. Ghambari and A. Rahati 2018 Appl. Soft Comput. J., vol. 62, pp. 736–767
[4] Sarkheyli, A., Zain, A. M., & Sharif 2015 Soft Computing, 19(7), 2011-2038.
[5] Rahim, S. Z. A., Sharif, S., Zain, A. M., Nasir, S. M., & Mohd Saad 2016 Advances in polymer technology, 35(1)
[6] Zain, A. M., Haron, H., & Sharif 2012 International Journal of Production Research, 50(1), 191-213.
[7] X. Zhang, Q. Kang, and X. Wang 2019 Swarm Evol. Comput., vol. 49, no. October 2018, pp. 245–265
[8] D. Wu and J. Zheng 2012 vol.2012
[9] Zain, A. M., Haron, H., & Sharif 2011 International Journal of Computer Integrated Manufacturing, 24(6), 574-592.
[10] Deris, A. M., Zain, A. M., & Sallehuddin 2013 Meccanica, 48(8), 1937-1945.
[11] N Rayhana, M Fathullah, Z Shayfull, S M Nasir, M H M Hazwan, M Sazli and Z. R. Yahya 2017 AIP Conference Proceedings. 1885 020097
[12] L. Cui et al. 2017 vol. 414, pp. 53–67
[13] A. J. Yurtkuran and E. Emel 2016 vol. 2016
[14] Chang-feng Chen et al 2019 IOP Conf. Ser.: Mater. Sci. Eng