IEICE Proceeding Series

Investigation of Artificial Bee Colony Algorithm with Two Kinds of Colonies

Yudai Shirasaki, Yoko Uwate, Yoshifumi Nishio

Vol. 1 pp. 223-226
Publication Date: 2014/03/17
Online ISSN: 2188-5079

2012 International Symposium on Nonlinear Theory and its Applications
NOLTA2012, Palma, Majorca, Spain, October 22-26, 2012

Investigation of Artificial Bee Colony Algorithm with Two Kinds of Colonies

Yudai Shirasaki†, Yoko Uwate† and Yoshifumi Nishio†

†Dept. of Electrical and Electronic Eng.,
Tokushima University,
Tokushima 770–8506, Japan
phone: +81–88–656–7470, fax: +81–88–656–7471
Email: { shirasaki, uwate, nishio } @ee.tokushima-u.ac.jp

Abstract—Artificial Bee Colony (ABC) is well known as optimization algorithm based on a particular intelligent foraging behavior of honeybee swarms. In this study, We propose Artificial Bee Colony having Two Kinds of Colonies (ABCTKC). It has different characteristic each group. We investigate the characteristic and the performance of ABCTKC by four benchmark functions and confirm that this new searching strategy has high speed convergence.

1. Introduction

Karaboga has proposed an Artificial Bee Colony (ABC) [1] [2] algorithm in 2005. The ABC is known as one of the optimization techniques used for optimization problems such as a linear programming, and this is an evolutionary algorithm to simulate the intelligent foraging behavior of honeybee swarms [3] [4]. The ABC algorithm is proposed that base on real bee behavior and consist of three kinds of honeybee, employed bee, onlooker bee and scout bee. Employed bee has the role of searching and searches wide area. Onlooker bee has the role of development and searches small area depending on employed bee searching informations. Scout bee that fell into local minima can escape from local minima by scout bee searching. ABC can obtain high efficiency that combine these bees searching. However, it has a disadvantage that convergence of the entire colony is slow.

In the previous study, we proposed Artificial Bee Colony having Twin Colonies (ABCTC) [5]. ABCTC divides one colony to two. In order to search effectively, we propose the exchange of employed bee between colonies as new strategy of ABCTC. It has high speed convergence in optimal solution because it can efficiently search having the exchange. The strategy has the effect having decrease search for local minima in the exchange and can easy to escape from local minima.

In this study, we propose Artificial Bee Colony having Two Kind of Colonies (ABCTKC). In ABCTKC, it has different characteristics each colony. By the interaction of different characteristics colonies, ABCTKC is able to explore more effectively than ABC and ABCTC. First, we research the effect of the information sharing and present the utility of ABCTKC. By computer simulations, we confirm that ABCTKC can converge with higher speed than the conventional ABC by avoiding the local minimum problem.

2. ABC with Two Kinds of Colonies

2.1. Behavior of real bees

One of the behavior of honeybee has divide into colony. When the number of bee of a colony became many numbers, the production efficiency of the colony decreases. In order to clear up this problem, honeybee divide into two colonies. From these, honeybee can keep constant efficiency for searching.

2.2. Process of ABCTKC algorithm

The ABCTKC has two different characteristic colonies. Two colonies are set to $Colony1$ and $Colony2$, respectively. $Colony1$ has the same characteristic as ABC and ABCTC, $Colony2$ has different characteristic from $Colony1$. In $Colony2$, employed bee search new solution using the current best result of employed bee in $Colony1$ and the average of whole position information in $Colony2$. The algorithm requires several parameters to be set; the number of bees is M, employed bees M_g is 50% and onlooker bees M_l is 50%. Initialization count is t_{Ri}. We explain the process of ABCTKC algorithm in detail as follows.

Step 1 (Initialization):
Let a generation step $t = 0$, the converge step $t_{Ri} = 0$ and the exchange step t_e. After the employed bees divided two colonies, disposes the employed bee $i = (1, 2, ..., M_e)$ at random.

Step 2 (Search by employed bee):
$Colony1$:
Employed bees search new food source each

group. In order to search better food source, the employed bee uses the following expression (1):

$$X_{id}(t+1) = X_{id}(t) + \Phi_{id}(X_{id}(t) - X_{kd}(t)) \quad (1)$$

where $\boldsymbol{X}_i(t) = (x_{i1}, x_{i2}, ..., x_{id})$ is position informations in employed bees. $k \in \{1, 2, ..., M_e\}$ is randomly chosen indexes for each groups. Although k is determined randomly, it has to be different from i. Φ_{id} is a random number between [−1,1]. d is dimension parameters.

*Colony*2:

In *Colony*2, the search method of employed bee is different from *Colony*1. Employed bee uses the following expression (2):

$$\begin{aligned} X_{id}(t+1) =_{id} (t) + rand[0,1](X_{id}(t) - X_{best,d}(t)) \\ - rand[0,1](X_{id}(t) - X_{ave,d}(t)) \end{aligned} \quad (2)$$

where $X_{best,d}(t)$ is the current best position of *Colony*1 and $X_{ave,d}(t)$ is the average position of whole *Colony*2.

Step 3 (Evaluation of employed bee): Evaluate $\boldsymbol{X}_i(t)$ and comparison of the current employed bee fitness value $f(\boldsymbol{X}_i(t))$.
If $f(\boldsymbol{X}_i(t)) \leq f(\boldsymbol{X}_i(t+1))$: Let $t_{Ri} = t_{Ri} + 1$.
If $f(\boldsymbol{X}_i(t)) > f(\boldsymbol{X}_i(t+1))$: Update the position information on each $\boldsymbol{X}_i$, and $t_{Ri} = 0$.

Step 4 (Search by onlooker bee): Onlooker bee searches new food sources depending on evaluation values of employed bees. As the evaluation value of food source increases, the probability chosen as onlooker bee increases. The chosen probability P_i is calculated by the following expression (3):

$$P_i = \frac{f(X_i)}{\sum_{n=1}^{M_g} f(X_n)} \quad (3)$$

Onlooker bee searching methods is the same as employed, it shown following expression (4):

$$Y_{jd}(t) = X_{id}(t) + \Phi_{id}(X_{id}(t) - X_{kd}(t)) \quad (4)$$

where $\boldsymbol{Y}_j(t) = (x_{i1}, x_{i2}, ..., x_{id})$ is position informations in onlooker bees.

Step 5 (Evaluation of onlooker bee):
If $f(\boldsymbol{X}_i(t)) \leq f(\boldsymbol{Y}_j(t))$: Let $t_{Ri} = t_{Ri} + 1$.
If $f(\boldsymbol{X}_i(t)) > f(\boldsymbol{Y}_j(t))$: Update the position information on each employed bee $\boldsymbol{X}_i$ to onlooker bee $\boldsymbol{Y}_j$, and $t_{Ri} = 0$.

Step 6 (Search by scout bee):
If T_{limit}, initialize $\boldsymbol{X}_i$ and $t_{Ri} = 0$. New $\boldsymbol{X}_i$ is searched by scout bee and is calculated by the following expression (5):

$$X_{id}(t+1) = X_{min} + rand[0,1](X_{max} - X_{min}) \quad (5)$$

where T_{limit} shows generation until carry out initialize $\boldsymbol{X}_i$ and is calculated by the multiplication of generation number T and dimension parameter d.

Step 7 (Exchange of employed bee): This Step 7 is new step in ABCTC and ABCTKC.
If $T_e \geq t_e$: Exchange employed bee between colonies and $t_e = 0$. The exchanged bee is determined by the provability of inverse proportional to t_{Ri}. In other words, the larger t_{Ri}, the higher is the likelihood of that employed bee to be selected. The selection probability E_i is calculated by the following expression (6):

$$E_i = \frac{(\frac{1}{T_{Ri}})^3}{\sum_{n=1}^{M_g} (\frac{1}{T_{Ri}})^3} \quad (6)$$

Step 8 Let $t = t + 1$, go back to [Step 2] and repeat until $T = t$.

3. Simulation

3.1. Benchmark Functions

In order to evaluate the performance of ABCTKC, we apply ABCTKC to some benchmark problems. The problems have the optimum (minimum) values of $f(x)$ in the algorithm. We use the following four benchmarks.

1. Sphere function :

$$f_1(x) = \sum_{d=1}^{d} x_d^2, \quad (7)$$

where $x \in [-5.12, 5.12]^d$ and the optimum solution x^* are all $[0, 0, ..., 0]$.

2. Rosenbrock function :

$$f_2(x) = \sum_{d=1}^{d-1} (100(x_{d+1}^2 - x_d)^2 + (x_d - 1)^2), \quad (8)$$

where $x \in [-2.048, 2.048]^d$ and the optimum solution x^* are all $[1, 1, ..., 1]$.

3. Rastrigin function :

$$f_3(x) = 10d + \sum_{d=1}^{d} (x_d^2 - 10\cos(2\pi x_d)), \quad (9)$$

where $x \in [-5.12, 5.12]^d$ and the optimum solution x^* are all $[0, 0, ..., 0]$.

4. Griewank function :

$$f_4(x) = \sum_{d=1}^{d} \frac{x_d^2}{4000} + \prod_{d=1}^{d} \cos(\frac{x_d}{\sqrt{d}}) + 1, \quad (10)$$

where $x \in [-512, 512]^d$ and the optimum solution x^* are all $[0, 0, ..., 0]$.

All the functions have the dimension parameter d. In this study, d is set 20 or 50 to investigate the performance and the defaults of employed bee is same each algorithms. We carry out the simulation 100 times with $T = 10000$ generations.

3.2. Simulation Results

We investigate the performance of ABCTKC that compare with the conventional ABC by tour benchmark functions. In this study, we calculated the percentage of found optimal solution and the average of generation until found optimal solution. First, we investigate the influence of the exchange of employed bees between colonies for benchmark functions (see Fig. 1). The best I_t is shown in x-axis. The I_t shows interval timing that carry out exchange of employed bee between colonies and used to calculate the T_e. The frequency of exchange is calculated from multiplication of generation times and I_t. In the case of $I_t = 0.0001$, the exchange is carried out every generation. In the case of $I_t = 1$, the exchange is not done. The y-axis shows average evaluation value of the simulation 100 times. Next, we consider the number of exchanges of bees for one time. In this study, the maximum number of exchanges of bees is set from one point to five points. The exchanges are performed when the bee did not update evaluation value. From Fig. 1, we found that its optimal parameters are very similar in $d = 20$ and $d = 50$ each function. If we apply ABCTKC to new function in high dimension, we can use optimal parameters that were determined by low dimension. From this, we can find optimal parameters that are obtained in little calculation. In comparison with ABC, we determined optimal parameters of ABCTKC with reference to these investigations.

In Table 1, it is shown comparison results with ABC and ABCTKC in $d = 20$ and $d = 50$. In $d = 20$, ABCTKC is better results than ABC in Sphere, Rosenbrock and Griewank function. In Rastrigin function, because of the difference between MAX and MIN by ABCTKC's result, ABCTKC obtains better results with small number of trials. In $d = 50$, ABCTKC is better results than ABC in all functions. Particularly, ABCTKC is more effectively in Rastrigin and Griewank functions. From results, we found that ABCTKC is obtained better results when the number of exchange times in every generation was increased. So, it is necessary to investigate behaviors by increasing the maximum number of exchange. Further, we found that ABCTKC is effective in high dimensions, particularly, multimodal functions. This is considered to be because $Colony2$ having high neighborhood search ability acted commutatively well with $Colony1$. It was also good to that have an unbalanced ratio of information sharing between colonies.

4. Conclusion

In this study, we proposed Artificial Bee Colony having Two Kinds of Colonies (ABCTKC) to improve the convergence speed of the conventional ABC. We investigated the performance of ABCTKC by comparison with ABC. From results, we confirmed ABCTKC has high dependability of parameters. Its optimal parameters are very similar in $d = 20$ and $d = 50$ for each function. As results, we found that ABCTKC obtained better results when the number of exchange times in every generation was increased. In future works, we would like to perform closer parametric examination. We will analyze the behavior of ABCTKC by more multi colonies.

Acknowledgment

This work was partly supported by JSPS Grant-in-Aid for Young Scientists 23700269.

References

[1] D. Karaboga, "An idea based on honeybee swarm for numerical optimization," Technical Report TR06, Erciyes University, Computer Engineering Department, 2005.

[2] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," Applied Mathematics and Computation 214, pp. 108-132, Erciyes University, Computer Engineering Department, 2009.

[3] D.T. Pham, A. Ghanbarzaden, E. Koc, S. Otri, S. Rahim and M. Zaidi, "The bees algorithm," Technical Note, Manufacturing Engineering Centre, Cardiff University, UK, 2005.

[4] K. Guney and M. Onay, "Bees algorithm for interference suppression of linear antenna arrays by controlling the phase-only and both the amplitude and phase," Faculty of Engineering, Erciyes University, 2007.

[5] Y. Shirasaki, Y. Uwate and Y. Nishio, "Artificial bee colony having twin colonies," Proc. of Asia Pacific Conference on Circuits and Systems, 2012.(Being submitted.)

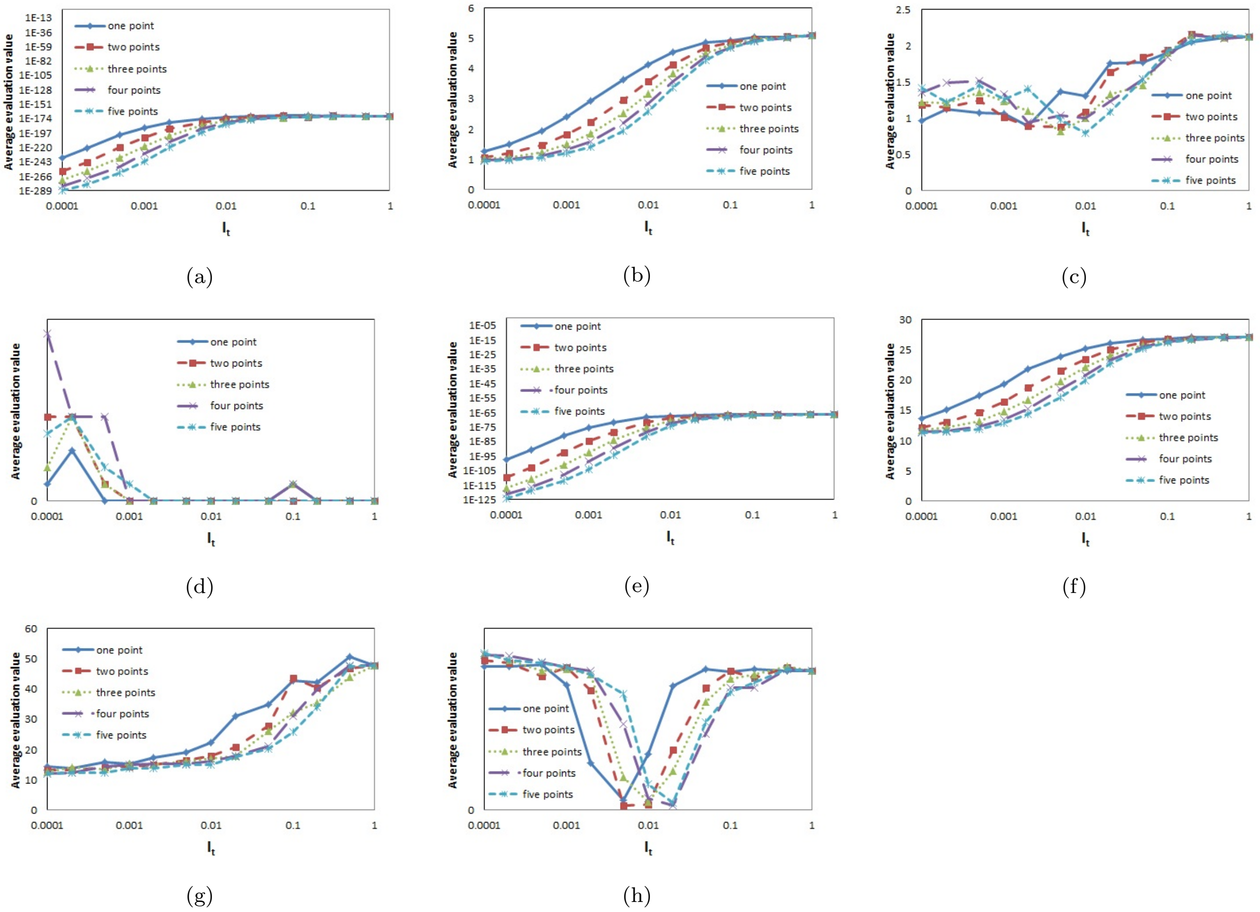

Figure 1: Performance of ABCTKC as changing parameter. (a) Sphere in $d = 20$. (b) Rosenbrock in $d = 20$. (c) Rastrigin in $d = 20$. (d) Griewank in $d = 20$. (e) Sphere in $d = 100$. (f) Rosenbrock in $d = 100$. (g) Rastrigin in $d = 100$. (h) Griewank in $d = 100$.

Table 1: Comparison of the performance of ABC and ABCTKC for $d = 20$.

Function	Algorithm	Ave	Min	Max	Optimal solution found [%]	Generation ave
Sphere	ABC	8.71E−255	4.34E−261	4.65E−253	NULL	NULL
	ABCTKC	**4.33E−289**	4.82E−300	3.21E−251	NULL	NULL
Rosenbrock	ABC	1.29	0.78	2.01	NULL	NULL
	ABCTKC	**0.94**	0.47	1.42	NULL	NULL
Rastrigin	ABC	**0.25**	0	18.7	86	7710
	ABCTKC	0.8	0	3.19	30	8130
Griewank	ABC	0	0	0	100	916
	ABCTKC	0	0	0	100	1053

Table 2: Comparison of the performance of ABC and ABCTKC for $d = 50$.

Function	Algorithm	Ave	Min	Max	Optimal solution found [%]	Generation ave
Sphere	ABC	9.77E−97	1.21E−98	1.19E−95	NULL	NULL
	ABCTKC	**6.34E-125**	5.34E−129	3.86E−123	NULL	NULL
Rosenbrock	ABC	12.83	8.38	19.75	NULL	NULL
	ABCTKC	**11.4**	8.06	15.8	NULL	NULL
Rastrigin	ABC	254.5	203.2	295.3	NULL	NULL
	ABCTKC	**12.1**	4.97	20.5	NULL	NULL
Griewank	ABC	1.09E−16	0	1.11E−16	2	8929
	ABCTKC	**3.33E-18**	0	1.11E−16	97	5890