An Enhanced Artificial Bee Colony Algorithm for Constraint Optimization

Zhen Wang and Xiangyu Kong

Abstract—**As the complexity of constraint optimization problem constructed based on the engineering problem, it is usually hard to solve. So as to solve the constrained optimization problem effectively, an enhanced artificial bee colony algorithm is put forward in this article. At the employed bee stage, the proposed algorithm introduces a new search equation and constraint handling strategy by identifying the current state of the population, and guide the population to access the feasible area quickly. Moreover, at the onlooker bee stage, the global optimal solution is used to lead the individuals to search the feasible region deeply, which improves the development capability of algorithm. The experiments are operated on 20 test functions in CEC 2006 and 2 realistic engineering optimization problems. All results confirm that the use of proposed algorithm for solving constraint optimization is valid.**

Index Terms—**constraint optimization, artificial bee colony algorithm, feasibility rules, best-guided searching**

I. Introduction

THE constraint optimization problems are widely exist in our daily life and engineering optimization, such as reducer design, vehicle routing problem, p-hub allocation problem, portfolio optimization and consumption prediction et. al. [1-5]. For these kinds of problems, we usually construct its mathematical model based on the original settings, and provide the optimal scheme for decision makers' selection over hundreds of considered solutions by the optimization methods. In mathematics, the general constraint optimization can be formed as:

$$\begin{aligned} &\min \quad f(x) \\ &s.t. \quad \begin{cases} g_i(x) \le 0, & i = 1,...,k \\ h_j(x) = 0, & j = k+1,...,m \\ l \le x \le u. \end{cases} \end{aligned} \tag{1}$$

where $g(\cdot)$ represents the inequality constraints and $h(\cdot)$ represent the equality constraints.

The intelligent algorithms have been spread used in constraint optimization problems because of their advantages, such as no need for gradient information of the problem, lower demand of the initialization points, and so on.[6] Artificial bee colony (ABC) algorithm is a swarm intelligent optimal algorithm invented by Karaboga[7] on 2005 based on bionics, which has much strength so as lesser parameters, concise structure, better robustness, and so forth. However, for solving constraint optimization problems, the ABC shows some issues. Over the years, researches have done numerous works on improving ABC and applying ABC, which has become a researching tend. In 2017, Liang et al. [8] proposed an improving artificial bee colony algorithm to conquer the premature convergence. In the article, the exploratory ability and exploitative ability are balanced based on the sorting selection method and best-so-far individual guided. Aiming at the precocity problem of GABC algorithm, Bansal et al. [9] invented an improved GABC algorithm (MGABC) by combining the concept based on individual movement fitness probability, which improved the searching stages of employed bees and onlooker bees. Gao et al. [2] designed a new artificial bee colony mechanism (LL-ABC) by combining direction learning and elite learning. The usage of LL-ABC on fuzzy portfolio optimization problem embodied the excellent optimal capability for complex problems. Li et al. [10] used two combination strategies and different renewal mechanisms to balance the exploratory and exploitative capabilities of the algorithm on employed bees and onlooker bees stages, and proposed an improved artificial bee colony algorithm. Aiming at such defects as weak exploratory ability, slow convergence and precocity problem, Chen et al. [11] proposed an extreme individual-guided ABC algorithm. In the article, the global extreme value and local extreme value are used to guide the algorithm searching, which can avoid algorithm premature.

Previous studies have shown that for solving constrained optimization problems, researchers usually use feasibility rules to retain better individuals. When updating individuals in the employed bee phase, most of the improved algorithms only use one strategy to adapt and adjust the individual position to get closer to the optimal solution, such as those in literature [8]-[13]. However, this single searching strategy is hard to precisely identify the current condition of the swarm and make corresponding adjustments. Therefore, there is still a large room for algorithm improvement at the convergence accuracy and maintaining diversity. To achieve the above purpose, an enhanced artificial bee colony (EABC) algorithm is proposed for constraint optimization .

Followings are the main contributions of this article. Firstly, two new searching equations are designed for different purpose. Secondly, the new searching mechanism is introduced to equilibrate the exploratory and exploitative capabilities. Finally, the feasible rules are used for comparing the quality of alternative solutions. Experiments

Manuscript received July 21, 2023; revised December 1, 2023.

This work was supported by the National Social Science Foundation of China under Grant 20BTJ026.

Zhen Wang is a professor of the Institute of Mathematics and Statistics, Xianyang Normal University, Shaanxi, 712000, China. (corresponding author. e-mail: jen715@163.com).

Xiangyu Kong is a professor of the Institute of Mathematics and Statistics, Xianyang Normal University, Shaanxi, 712000, China. (e-mail: kxywz08@163.com).

are conducted on CEC 2006 test functions and realistic engineering optimization problems.

The rest of the paper is organized as follows. Section 2 describes the proposed EABC in detail. In Section 3, experiments are presented and the results are discussed. Finally, a conclusion is provided.

II. THE ENHANCED ARTIFICIAL BEE COLONY ALGORITHM

Artificial bee colony Algorithm (ABC) is a swarm intelligent optimization algorithm on the basis of bionics [7]. In ABC, the artificial bees imitate the movements of the honey bees. According to the division of labor, the artificial colony is consist of employed bees, onlooker bees and scout bees. Through a series of searching in the feasible area, food sources with the largest amount of nectar are found. The ABC has the merit of parallel computation, fast convergence , few control parameters, simple principle, etc. It can effectively solve highly complex nonlinear optimization problems, and has been widely concerned by researchers. This article presents an enhanced artificial bee colony algorithm (EABC) for constraint optimization. In EABC, two different search equations and selection strategies are proposed for the searching stages. Specific ideas and processes are as follows.

A. Initialization

The random initialization is put into use in EABC to generate the initial population as follows:

$$x_{i,j} = x_{min}^{j} + rand\left(0,1\right)\left(x_{max}^{j} - x_{min}^{j}\right),\ \ i = 1,2,...,SN, \quad (2)$$

where SN denotes the quantity of food sources, D is the problem dimension, x_{max}^{j} and x_{min}^{j} is the upper bound and lower bound of the search space.

B. Constraint handling method

Different from unconstrained problems, constraint problems need to explore the optimum solution in the feasible region limited by constraints, which increases the difficulty of the search process. Aiming at promoting the searching capability of the algorithm, the constraint handling method is also an important part of the algorithm. Feasibility rule is extensively used in designing intelligent algorithm for constraint optimizations because of its simplicity. In this article, feasibility rule is introduced to select the candidate solutions. The feasibility rule is defined as follows.

Definition 1: (Feasibility Rule) [13] If x_i superior x_j, one of the following conditions must be met:

1. $conV_i = conV_j = 0$ and $f_i < f_j$;
2. $conV_i = 0$ and $conV_j > 0$;
3. $conV_i > 0$, $conV_j > 0$ and $conV_i < conV_j$;

where $conV_i$ and $conV_j$ respectively represent the degree of constraint violation of individuals, and f_i and f_j respectively represent the objective function values.

C. Employed bee stage

At the employed bee stage, EABC algorithm uses a new strategy to develop the search space, namely:

$$v_{i,j} = x_{r_1,j} + F_1 * (x_{best,j} - x_{i,j}) + F_2 * (x_{r_1,j} - x_{r_2,j}), \quad (3)$$

where x_{best} is the best individual at present, $j \in \{1,2,...,D\}$, $r_1, r_2 \in \{1,2,...,SN\}$, $r_1 \neq r_2 \neq i$, F_1 is a random number in the range $[0,1]$, F_2 is $A * rand$, A is a random number generated by a normal distribution.

D. Onlooker bee stage

At the onlooker bee stage, artificial bees will further exploit the better food source on the basis of food sources states provided by the employed bees. In the algorithm, the location of food source mined by the onlooker bees is determined by selecting probability. The original artificial bee colony algorithm adopts a binary method to calculate the selection probability. At the onlooker bee stage, the probability of the infeasible individual being selected is less than 0.5, and the probability of the feasible individual being selected is greater than 0.5. This probabilistic calculation method reduces the probability of the infeasible solution being selected, so that the optimal infeasible solution at the onlooker bee stage will be eliminated in the following bee stage. Therefore, formula (4) is used in this article to calculate the selection probability of the onlooker bees, making full use of objective function values and the degree of constraint violation.

$$Prob_i = 0.9 * \frac{fit_i}{max(fit_i)} + 0.1, \quad (4)$$

where $fit_i = f1_i + conV1_i$, $conV1_i = \frac{1}{conV_i + 1}$, $f1_i = \begin{cases} \frac{1}{1+f_i}, & if\ f_i \geq 0 \\ 1+|f_i|, & if\ f_i < 0 \end{cases}$, f_i is the objective function value.

Since the population has been evaluated at the employed bee stage, and appropriate strategies have been selected to explore nectar sources. At this stage, the population has been guided to search in a helpful direction, the convergence accuracy needs to be appropriately improved. In view of this, the EABC algorithm uses formula (5) to search the new food sources at the onlooker bee stage,

$$v_{i,j} = x_{r_1,j} + F_1 * (x_{best,j} - x_{r_1,j}) + F_2 * (x_{r_1,j} - x_{best,j}), \quad (5)$$

where x_{best} is the best individual at present, $j \in \{1,2,...,D\}$, $r_1 \in \{1,2,...,SN\}$, $r_1 \neq i$, F_1 is a random number in the range $[0,1]$, F_2 is $A * rand$, A is a random number generated by a normal distribution. Since formula (5) can be simplified into

$$v_{i,j} = x_{r_1,j} + (F_1 - F_2) * (x_{best,j} - x_{r_1,j}),$$

the new candidate solution generated by formula (5) is more likely to move towards the current optimal individual, which can guide the population towards the optimal area to accelerate the convergence accuracy of the algorithm.

E. Scout bee stage

At the scout bee stage, if the mining degree of a food source reaches the limit, the original ABC algorithm adopts formula (2) to reinitialize the food source, and the EABC adopts this strategy at the scout bee stage.

The followings are the specific steps of the EABC algorithm:

Algorithm 1: Enhanced artificial bee colony algorithm

Step 1: Set iteration $t = 1$, $max\,Gen$ is maximum number of iterations, initialize the population P_t, and maximum number of mining $Limit = SN \times D$;

Step 2: At the employed bee phase, formula (3) and feasibility rule are used to update the population;

Step 3: Calculate the probability $Prob_i$ of each food source according to formula (4);

Step 4: According to $Prob_i$, the onlooker bees select the food sources and search for new food sources by using formula (5);

Step 5: Use the feasibility rule to retain better individuals between the old and new food sources and renew the population P_t;

Step 6: If there is a food source whose mining times are greater than Limit, a new food source is randomly initialized according to formula (2);

Step 7: Determine whether $t \le max\,Gen$ is satisfied. If yes, $t = t + 1$, go to Step 2; otherwise, output the global optimal solution.

TABLE I
EXPERIMENT RESULTS ON CEC2006.

Functions	Algorithm	min	mean	std.	percentage
F1	ABC	-15	-15	0	1
	GABC	-15	-14.99999962	2.05949E-06	0.975
	MeanABC	-14.60116768	-9.609462611	1.498354923	1
	EABC	-15	-14.99989527	0.000392573	1
F2	ABC	-0.646003938	-0.613646728	0.016199418	1
	GABC	-0.715581085	-0.652614205	0.027076951	1
	MeanABC	-0.627490106	-0.586430411	0.021544179	1
	EABC	-0.711639376	-0.649463822	0.027937429	1
F3	ABC	-0.590009223	-0.059530055	0.110435876	0.975
	GABC	-0.09288926	-0.017616455	0.025938814	0.975
	MeanABC	-0.20887784	-0.021259585	0.041281664	0.95
	EABC	-0.311082044	-0.084903194	0.071693898	0.95
F4	ABC	-30506.96741	-30167.40814	207.1946942	1
	GABC	-30616.05945	-30424.43277	116.7392656	1
	MeanABC	-30479.01015	-30286.91483	120.6085578	0.975
	EABC	-30571.47762	-30190.44159	153.7944662	1
F5	ABC	5125.029604	5422.783134	335.9758408	0
	GABC	5127.726747	5515.504623	337.8399549	0
	MeanABC	4946.358599	5417.602095	327.9686703	0
	EABC	5122.908779	5390.921283	297.7285091	0
F6	ABC	-6961.127751	-6957.078734	2.709971443	0.375
	GABC	-6947.3022	-6810.671831	96.47398356	0.325
	MeanABC	-7928.998188	-4610.221794	2830.430693	0.025
	EABC	-7261.019056	-6850.484617	109.6304398	0
F7	ABC	26.31511506	33.6674739	7.429609215	0.95
	GABC	26.77771611	31.82191887	4.025300628	0.95
	MeanABC	28.10517116	38.54938767	17.75635483	1
	EABC	25.44638357	30.00462391	3.128214224	0.95
F8	ABC	-0.095825041	-0.09360233	0.012174272	1
	GABC	-0.095825041	-0.095825041	2.64068E-17	1
	MeanABC	-0.095824987	-0.095823336	1.8621E-06	0.95
	EABC	-0.095825041	-0.095824941	4.03331E-07	1
F9	ABC	683.7097733	689.0311593	6.802499912	1
	GABC	684.3399483	686.6898661	2.575599239	1
	MeanABC	684.457946	685.2933018	0.867560623	0.95
	EABC	682.6793653	686.3113628	3.707781001	1
F10	ABC	5628.136014	8642.429034	1238.717509	0
	GABC	3895.164401	8560.360207	1772.529434	0
	MeanABC	6481.251411	8982.974594	1526.314851	0
	EABC	4985.906329	7600.46479	2049.495587	0

TABLE II
EXPERIMENT RESULTS ON CEC2006.

Functions	Algorithm	min	mean	std.	percentage
F11	ABC	0.74992655	0.75923819	0.021724187	0.9
	GABC	0.749903455	0.76585815	0.041807736	0.65
	MeanABC	0.749954975	0.776212786	0.038981228	0.8
	EABC	0.749917081	0.780297968	0.055018172	0.5
F12	ABC	-1	-1	0	1
	GABC	-1	-1	0	1
	MeanABC	-0.999999971	-0.999999621	3.37917E-07	0.95
	EABC	-1	-1	2.79805E-15	1
F13	ABC	0.290792533	0.839021006	0.194152936	0
	GABC	0.37849493	1.175924583	0.8055184	0
	MeanABC	0.090489761	0.970441725	0.445505751	0
	EABC	0.149313826	1.056095851	0.536182902	0
F14	ABC	-47.61472719	-46.25513131	1.017069026	0.55
	GABC	-47.37511958	-41.53137321	3.120628161	0.05
	MeanABC	-46.82811442	-42.10266229	2.568299208	0.15
	EABC	-46.28354972	-41.31442953	2.003954498	0
F15	ABC	962.3370273	968.4688316	2.775663213	0
	GABC	961.7264801	966.3695504	3.480862038	0.3
	MeanABC	961.7452931	966.152455	3.512712161	0
	EABC	961.7259691	966.0132959	3.259843681	0
F16	ABC	-1.748738457	-1.534270028	0.117783593	0.4
	GABC	-1.879053669	-1.669571305	0.101057065	0.85
	MeanABC	-1.870162938	-1.739555583	0.099645857	0.65
	EABC	-1.704550388	-1.446152514	0.148954532	0.6
F17	ABC	8862.462257	9010.034359	115.7186684	0
	GABC	8931.517948	9076.897368	119.7109363	0
	MeanABC	8859.481674	9051.602936	133.0629706	0
	EABC	8867.435393	9057.550607	146.5766826	0
F18	ABC	-0.865102466	-0.830800645	0.045320387	1
	GABC	-0.861389539	-0.848636539	0.014696944	0.9
	MeanABC	-0.862264097	-0.850364786	0.010396336	0.95
	EABC	-0.865332183	-0.817846078	0.082177042	0.9
F19	ABC	107.1861538	276.9514147	61.27641133	1
	GABC	84.84198214	226.0313504	89.81549696	1
	MeanABC	74.2412803	247.1260725	110.0114538	1
	EABC	90.70467128	262.3001588	62.54022744	1
F20	ABC	0.102684427	0.196736857	0.054698577	0
	GABC	0.142827152	0.234744656	0.064206718	0
	MeanABC	0.159032878	0.229444283	0.058794356	0
	EABC	0.158387092	0.218730168	0.048494933	0

III. NUMERICAL EXPERIMENTS

So as to verify the optimization capability of EABC, this section performs experiments on CEC 2006 test functions and two engineering optimization problems. The details of the CEC 2006 test functions show in literature [14], and the details of two engineering optimization problems will describe in the followings.

A. *Experiments on CEC 2006 test functions*

For the fair comparison to other algorithms, the population size sets $SN = 40$, the maximum number of iterations set as $maxGen = 3000$, and maximum mining degree is $Limit = SN * D$ in this section. Under the same conditions, each test function is independently run for 30 times. The EABC is contrasted with other three ABC variant algorithm, which is original ABC[7], GABC[15] and MeanABC[16]. The comparison results are provided in Table I and II, where min and mean respectively represent the best value and average value obtained by the algorithm in the optimization process, percentage and std. represent the proportion of feasible solutions and standard deviation in Table I and II.

From Table I and II, the average optimization values of EABC on the test functions are smaller than or equal to the average value of the other three ABC variant algorithms, and the values of standard deviation of EABC are much smaller, which prove the EABC has high convergence accuracy and maintains good stability.

Fig. 1 and Fig. 2 are comparison diagram of convergence curves on some test functions. From the diagram, we can see that compared with the original ABC, GABC and MeanABC, the EABC can find better individuals and has higher convergence accuracy.

It can be seen from the above analysis that compared with other ABC variants, EABC has strong competitiveness in solving constraint optimization problems.

B. Experiments on engineering problems

For the further verification of the EABC optimum capability for constraint optimization problems, the algorithm is applied to two engineering problems: welded beam design problem (WBD) and cantilever beam design problem.

(1) Cantilever beam design problem

In this problem, the target is to select the proper parameters to make sure the weight of the cantilever beam is as small as possible. Therefore, the cantilever beam design problem can be formulated as:

$$\begin{aligned} \min \quad & f(x) = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5) \\ s.t. \quad & g_1(x) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} \le 1 \\ & 0.01 \le x_i \le 100, \quad i = 1,2,3,4,5 \end{aligned}$$

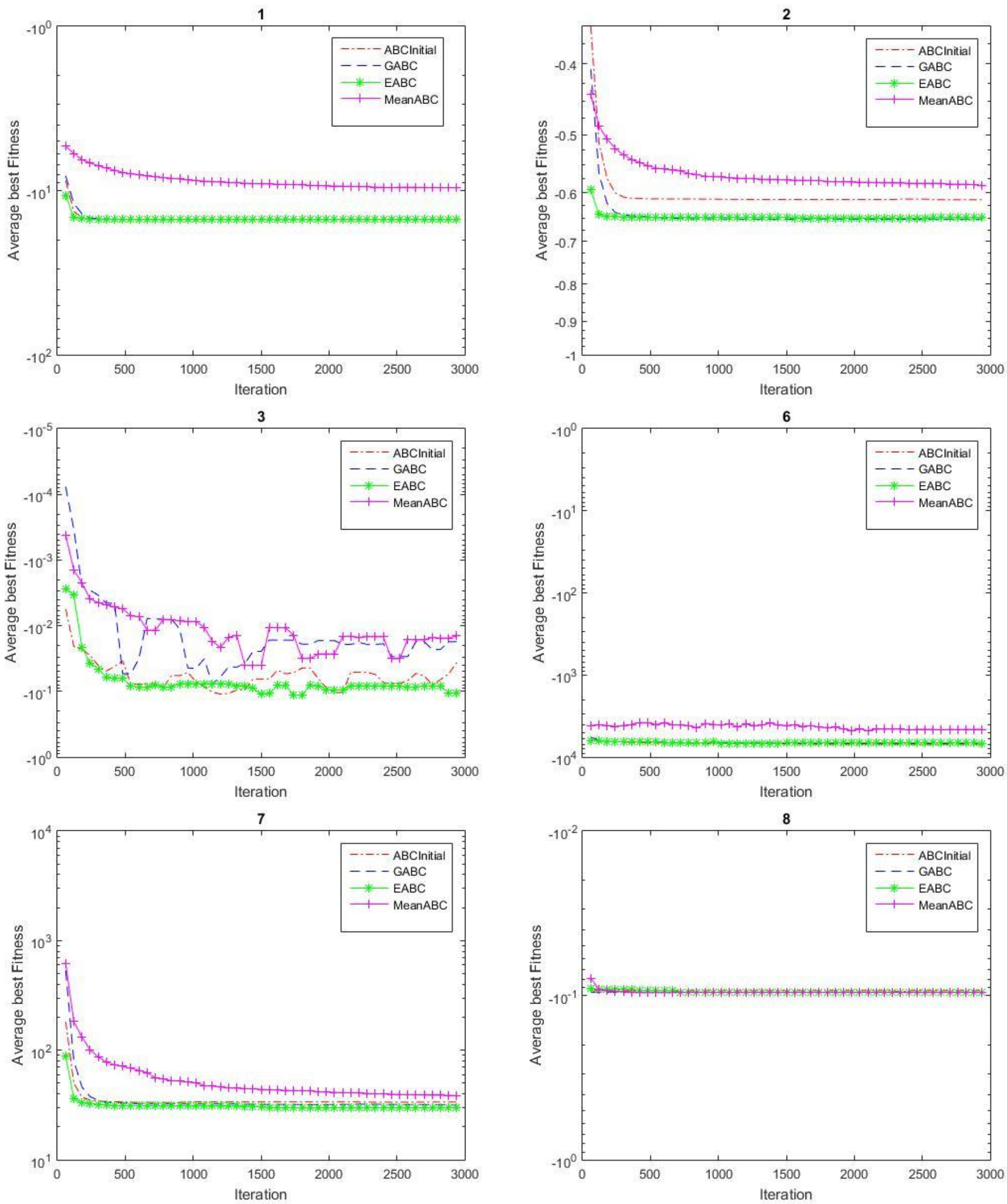

Fig. 1. Some convergence curves of EABC and other three ABC variant algorithms.

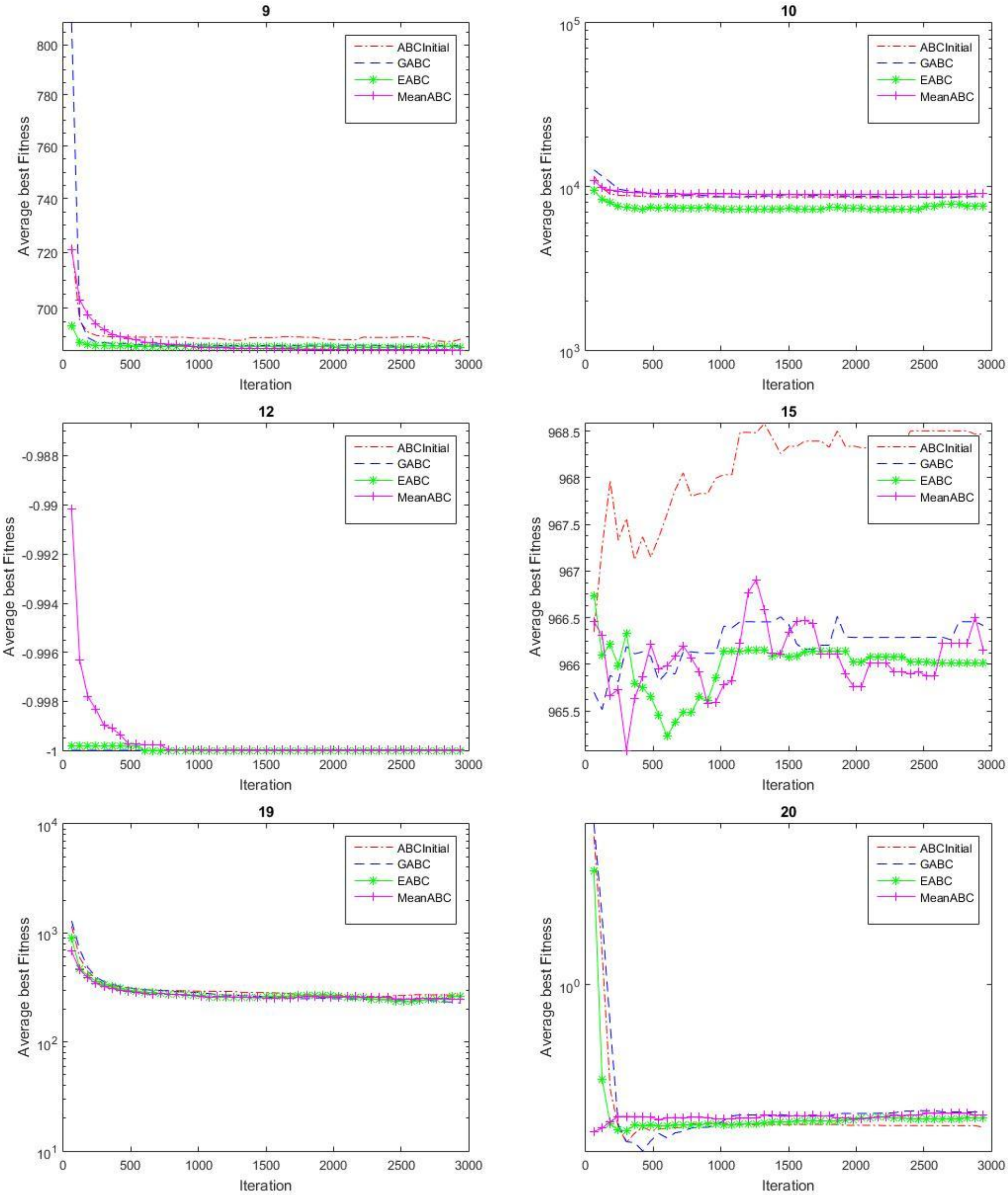

Fig. 2. Some convergence curves of EABC and other three ABC variant algorithms.

Fig. 3 displays the construction detail of this problem. ABC, GABC, MeanABC and EABC are compared on solving cantilever beam design problem.

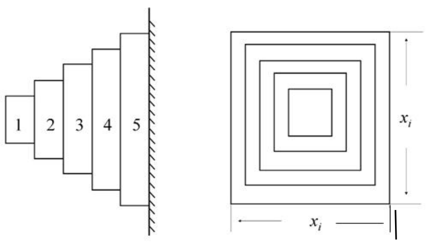

Fig. 3. *Cantilever beam design problem.*

TABLE III
EXPERIMENT RESULTS ON CANTILEVER BEAM DESIGN PROBLEM.

Algorithm	min	mean	max	std.
ABC	1.353687593	1.396972835	1.525259226	4.79E-02
GABC	1.362285153	1.379791392	1.395771502	1.17E-02
MeanABC	1.356431906	1.384081016	1.441050896	2.50E-02
EABC	1.345722893	1.366928895	1.392860994	1.62E-02

Table III lists the experiment results of four above algorithms. From Table III, either the min, mean, max value or standard deviation of 30 times runs, the EABC gets the minimum of all four algorithms. Furthermore, Fig. 4 shows the convergence curve of this problem, which also means the better performance of EABC.

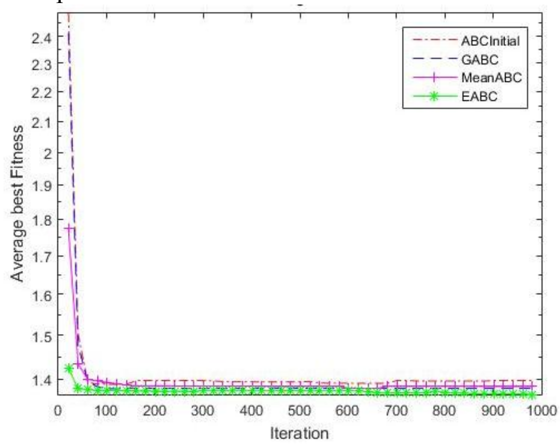

Fig. 4. Convergence curves for cantilever beam design problem.

(2) Welded beam design problem, WBD

In this problem, the main target is to minimize its cost under certain constraints. The four decision variables of this problem are given by $h(x_1)$, $l(x_2)$, $t(x_3)$ and $b(x_4)$. Then, the structure of welded beam design problem is shown in Fig. 5, and the problem is formulated as:

$$\min \quad f(x) = 1.1047x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$$

$$s.t. \quad g_1(x) = 0.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5 \le 0$$

$$g_2(x) = x_1 - x_4 \le 0$$

$$g_3(x) = 0.125 - x_1 \le 0$$

$$g_4(x) = \delta(x) - \delta_{\max} \le 0$$

$$g_5(x) = \sigma(x) - \sigma_{\max} \le 0$$

$$g_6(x) = \tau(x) - \tau_{\max} \le 0$$

$$g_7(x) = P - P_c(x) \le 0$$

$$0.1 \le x_1 \le 2.0$$

$$0.1 \le x_2 \le 10.0$$

$$0.1 \le x_3 \le 10.0$$

$$0.1 \le x_4 \le 2.0$$

where $\sigma(x) = \dfrac{6PL}{x_4x_3^2}$, $\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\dfrac{x_2}{2R} + (\tau'')^2}$, $\tau' = \dfrac{P}{\sqrt{2}x_1x_2}$, $\tau'' = \dfrac{MR}{J}$, $R = \sqrt{\dfrac{x_2^2}{4} + \left(\dfrac{x_1 + x_3}{2}\right)^2}$, $M = P\left(L + \dfrac{x_2}{2}\right)$, $J = 2\left\{\sqrt{2}x_1x_2\left[\dfrac{x_2^2}{12} + \left(\dfrac{x_1 + x_2}{2}\right)^2\right]\right\}$, $\delta(x) = \dfrac{4PL^3}{Ex_3^3x_4}$, $P_c = \dfrac{4.013\sqrt{x_3^3x_4/36}}{L^2}\left(1 - \dfrac{x_3}{2L}\sqrt{\dfrac{E}{4G}}\right)$, $P = 6000$, $G = 12\times10^6$, $E = 30\times10^6$, $L = 14$, $\delta_{\max} = 0.25$, $\sigma_{\max} = 30000$, $\tau_{\max} = 13600$.

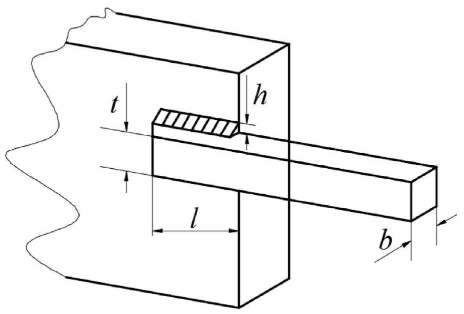

Fig. 5. *Welded beam design problem.*

The above problem is solved by ABC, GABC, MeanABC and EABC, and the experiment results are in Table IV. From Table IV the EABC is evidently superior to the other three ABC variants. Fig. 6 shows the convergence curve, and intuitively the convergence accuracy and speed of EABC are higher than other ABC variants.

TABLE IV
EXPERIMENT RESULTS ON WELDED BEAM DESIGN PROBLEM.

Algorithm	min	mean	max	std.
ABC	1.743678662	1.968862793	2.638985185	2.75E-01
GABC	1.758604004	1.92775213	2.417376518	1.94E-01
MeanABC	1.744021902	2.066846774	2.906530372	3.55E-01
EABC	1.740673629	1.911234217	2.240690755	1.36E-01

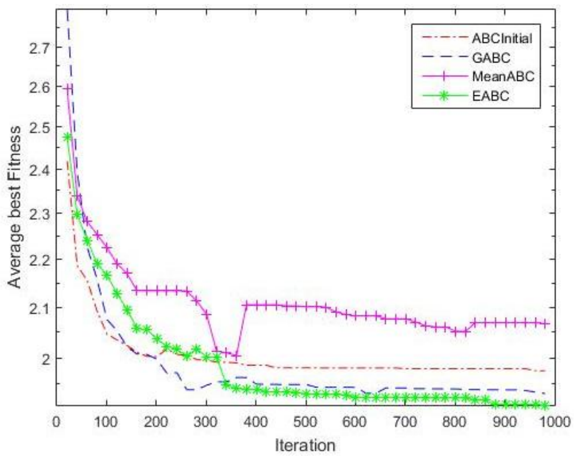

Fig. 6. Convergence curves for welded beam design problem.

IV. CONCLUSIONS

In this article, an enhanced artificial bee colony (EABC) algorithm is designed for constraint optimization. At the employed bee phase, the EABC uses the new search equation and selection mechanism to lead the colony to enter the feasible region from different directions, speeding up the algorithm convergence speed and avoiding the algorithm premature. At the onlooker bee stage, a new probabilistic calculation method is proposed, which enables the algorithm to identify the best individual effectively. Furthermore, a new best-guided search equation is also designed for enhancing the exploitative ability. The experimental results on CEC 2006 test function and two realistic engineering optimization problems show that EABC has fast convergence speed, high convergence accuracy and strong robustness.

REFERENCES

[1] Z. Y. Su, Q. X. Feng, Y. T. Wei and S. N. Yuan, “Improved self-adaptive constrained differential evolution algorithm,” *Microelectronics & Computer*, vol. 36, no. 9, pp. 30-37, 2019.

[2] W. Gao, H. Sheng, J. Wang and S. Wang, “Artificial Bee Colony Algorithm Based on Novel Mechanism for Fuzzy Portfolio Selection,” *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 5, pp. 966-978, 2019.

[3] Y. Hou, W. He, C. Wang and X. Ren, "A Two-stage Selection Hyper-heuristic Algorithm for the Capacitated Vehicle Routing Problem," *IAENG International Journal of Applied Mathematics*, vol. 52, no.4, pp. 979-988, 2022.

[4] Y. Xing, J. Wang, Y. Zheng and Y. Wang, "Particle Swarm Optimization Algorithm Based on Different Inertia Weights for Solving the P-Hub Allocation Problem," *IAENG International Journal of Applied Mathematics*, vol. 52, no.4, pp. 1025-1039, 2022.

[5] A. Wahba, R. El-khoribi and Shereen Taie, "A New Hybrid Model for Energy Consumption Prediction Based on Grey Wolf Optimization," *IAENG International Journal of Computer Science*, vol. 49, no.2, pp. 469-481, 2022.

[6] C. Wang, P. Shang and X. Wu, "Hierarchical Particle Swarm Optimization Based on Mean Value," *IAENG International Journal of Applied Mathematics*, vol. 53, no.2, pp. 695-703, 2023.

[7] D. Karaboga, *An idea based on honey bee swarm for numerical optimization*, Technical report-tr06, Kayseri, Turkey: Erciyes University, 2005.

[8] Y. Liang, Z. Wan and D. Fang, “An improved artificial bee colony algorithm for solving constrained optimization problems,” *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 3, pp. 739-754, 2017.

[9] J. C. Bansal, S. K. Joshi and H. Sharma, “Modified global best artificial bee colony for constrained optimization problems,” *Computers & Electrical Engineering*, vol. 2017, pp. 365-382, 2017.

[10] W. X. Li, L. Z. Liu, C. J. Dai and y. Li, “Improved artificial bee colony algorithm based on multiple group combination strategies,”

Journal of Computer Applications, vol. 41, no. 11, pp. 3113-3119, 2021.

[11] L. Chen and L. G. Wang. "Extremal-guided Artificial Bee Colony Algorithm," *Journal of Frontiers of Computer Science and Technology*, vol. 16, no. 11, pp. 14, 2022.

[12] Y. Wang and Z. Cai, "A Dynamic Hybrid Framework for Constrained Evolutionary Optimization," *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics: A Publication of the IEEE Systems Man & Cybernetics Society*, vol. 42, no. 1, pp. 203-217, 2012.

[13] Z. Y. Li, T. Huang, S. M. Chen and R. Li, "Overview of constrained optimization evolutionary algorithms," *Journal of Software*, vol. 28, no. 6, pp. 1529-1546, 2017.

[14] J. J. Liang, T. P. Runarsson, E. Mezura-montes, M. Clerc, P. N. Suganthan, C. A. Coello Coello and K. Deb, *Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization*, Technical Report, 2005, May, pp. 251-256.

[15] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166-3173, 2010.

[16] A. Alrosan, W. Alomoush, N. Norwawi, M. Alswaitti and S. N. Makhadmeh, "An improved artificial bee colony algorithm based on mean best-guided approach for continuous optimization problems and real brain MRI images segmentation," *Neural Computing & Applications*, vol. 33, no. 5, pp. 1-27, 2021.