Received: 22 November 2021 | Accepted: 28 December 2021

DOI: 10.1002/int.22816

RESEARCH ARTICLE

WILEY

Artificial bee colony algorithm with bi-coordinate systems for global numerical optimization

Xinyu Zhou | **Junhong Huang** | **Hao Tang** | **Mingwen Wang**

School of Computer and Information Engineering, Jiangxi Normal University, Nanchang, China

Correspondence
Mingwen Wang, School of Computer and Information Engineering, Jiangxi Normal University, 330022 Nanchang, China.
Email: mwwang@jxnu.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 61966019, 61876074, 61866017; Natural Science Foundation of Jiangxi Province, Grant/Award Number: 20192BAB207030; Postgraduate Innovation Fund Project of Jiangxi Provincial Department of Education, Grant/Award Number: YC2021-S307; Graduate Innovation Fund project of Jiangxi Normal University, Grant/Award Number: YJS2020023

Abstract

As an effective global optimization technique, artificial bee colony (ABC) algorithm has become one of the hottest research topics in the fields of evolutionary algorithms. However, the solution search equation is not rotationally invariant, which causes the problem that the performance of ABC is sensitive to the coordinate system. Although many improved ABC variants have been developed, they rarely considered the problem. Hence, to solve the problem, we propose a new ABC variant with bi-coordinate systems (BSABC), including the original coordinate system and the eigen coordinate system. The two coordinate systems own different characteristics: (1) the former one aims to maintain the population diversity, and (2) the latter one is to adapt the search to the fitness landscape of the problems. Based on the characteristics, in the BSABC, the two coordinate systems are used in the employed bee phase and onlooker bee phase, respectively. Meanwhile, two new solution search equations are designed by utilizing the elite information, and they are respectively performed in the two coordinate systems to further improve the algorithm performance. As another contribution of this study, in the scout bee phase, the multivariate Gaussian distribution is constructed to replace the original method to generate offspring, which is helpful to save the search experience. The performance of the BSABC is verified by extensive experiments on the CEC2013 test suite and one real-world optimization problem, and four well-established ABC variants and three other evolutionary algorithms are included in the performance comparison. The comparison results confirm that the BSABC shows competitive performance by achieving better results on the majority of test functions.

KEYWORDS

artificial bee colony, bi-coordinate systems, eigen coordinate system, elite information, multivariate Gaussian distribution

1 | INTRODUCTION

In recent years, as an effective technique to solve the global optimization problems, evolutionary algorithms (EAs) have been playing an important role in the fields of academic and industry. Because many classic optimization techniques, like the gradient-based methods,[1] cannot meet the requirements of optimizing the problems with complex features, such as nonconvex, multimodal, or even nondifferentiable. In contrast, EAs can be used for almost all of the problems without difficulty, which may be one of the most attractive advantages of EAs.[2–4] Being inspired by the biological evolution phenomenon, that is, the survival of the fittest, EAs are comprised of various paradigms with different metaphors,[5,6] such as the genetic algorithm (GA),[7] particle swarm optimization algorithm (PSO),[8] differential evolution algorithm (DE),[9] and artificial bee colony algorithm (ABC).[10] For instance, the GA simulates the mutation and crossover behavior of biological genes,[11] and the PSO imitates the flying and preying behavior of bird flocks.[12] In addition, some new paradigms have been recently proposed, such as the remora optimization algorithm (ROA),[13] African vultures optimization algorithm (AVOA),[14] and whale optimization algorithm (WOA).[15]

As one of the most popular paradigms of EAs, the ABC has attracted much attention in the community of EAs, which was first proposed by Karaboga in 2005.[16] The metaphor of the ABC derives from the intelligent foraging behavior of honey bees in nature, and the process of optimizing the problems by the ABC corresponds to the process of maximizing the nectar amount by honey bees. In comparison with other paradigms of EAs, one of the most salient features of the ABC may be that it owns fewer control parameters without sacrificing the performance,[17] which will be described in the next Section2.1 about the basic ABC. Until now, due to its simple algorithmic structure yet good performance, the ABC can be successfully used to solve a sequence of real-world optimization problems, such as portfolio selection model,[18] feature selection problem,[19] robot path planning,[20] and image segmentation.[21] Recently, however, some studies have pointed out that the ABC still shows unsatisfactory performance in solving some complex problems,[22,23] since it performs strong explorative capability while weak exploitative capability. The reported reason is that the solution search equation, as the main way to generate offspring, only includes a randomly selected individual (food source) but no other elite individuals, so the ABC performs slow convergence speed and low convergence accuracy.[24]

Hence, many modifications have been introduced into the basic ABC for gaining better exploitation,[25–28] and most of them share the similar idea of utilizing the global best individual or some elite individuals. There is no doubt that the reported experimental results verified the feasibility of the idea, that is, the performance of ABC can be effectively enhanced by utilizing the elite information. However, in our opinion, there still exists another important reason for the poor performance of ABC in solving complex problems, especially for the problems with high variable correlation. That is, the solution search equation is sensitive to rotation on the coordinate system. In other words, the solution search equation is not rotationally invariant. With this deficiency, the search behavior of ABC cannot well adapt to different problem landscapes, and as a result, the search efficiency is limited by the coordinate system. In fact, the similar problem is also encountered in other paradigms of EAs, especially for the DE.[29] Fortunately, the problem has been well investigated for these paradigms, and some available solutions have been provided. For example, Guo and Yang[30] proposed an eigenvector-based crossover operator for the DE (DE/eig). In the DE/eig, the eigenvector information of the covariance matrix of population is utilized to rotate the original coordinate system, which can make the crossover operator of DE rotationally invariant. In contrast, in the context of ABC, the deficiency of the rotational variability of the solution search equation is almost ignored, even though numerous ABC variants have been presented in recent years.

Based on the above observations, in this study, we are motivated to solve the problem of the rotational variability for a higher effective ABC, and propose a new ABC variant with bi-coordinate systems, named BSABC for short. In the BSABC, the introduced modifications mainly involve two aspects. As for the first modification, the eigen coordinate system is established as an alternative to the original coordinate system, by making use of the covariance matrix of the population. In the eigen coordinate system, it is quite possible that the offspring tend to correspond with the problem landscape, which is helpful to guide the population evolution towards the promising search areas. It is worth noting that the original coordinate system is also retained in the BSABC to avoid losing population diversity, and the two coordinate systems are implemented at different stages during the evolution. In terms of the second modification, in the scout bee phase, to prevent the scout bee losing the already obtained search experience, a new method is presented to generate offspring by sampling from the multivariate Gaussian distribution containing the elite information. Moreover, two new solution search equations are designed based on the idea of utilizing the elite information, with the aim of enhancing the exploitative capability. To validate the effectiveness of the BSABC, extensive experiments are carried out on the CEC2013 test suite and one real-world optimization problem, and four well-established ABC variants and three other paradigms of EAs are included in the performance comparison. The comparison results confirm the competitive performance of the BSABC, especially for the problems with high variable correlation.

In brief, the main contributions of this study can be summarized as follows:

- The method of bi-coordinate systems is proposed, including the eigen coordinate system and original coordinate system. For the eigen coordinate system, it is established to solve the problem that the solution search equation of ABC is not rotationally invariant. Meanwhile, the original coordinate system is retained for maintaining the population diversity.
- In the scout bee phase, to save the search experience, an effective restart method based on the multivariate Gaussian distribution is proposed. As for the multivariate Gaussian distribution, it is constructed by using the elite individuals of the population, which is helpful to inherit useful information to save the search experience.
- Extensive experiments on benchmark functions and a real-world optimization problem are adopted to verify the effectiveness of our work. The performance comparisons confirm that

our work has very competitive performance, it not only does well on the benchmark functions, but also on the real-world optimization problem.

The rest of this study is organized as follows. In Section 2, we briefly introduce the basic ABC and some related works on improved ABC variants. Next, in Section 3, the proposed BSABC is presented in details, and the corresponding pseudocode of the BSABC is given as well. In Section 4, the experiments and related discussions are given. Last, this study is concluded in Section 5.

2 | BASIC ABC AND RELATED WORKS

2.1 | Basic ABC

In the basic ABC, the honey bees are classified into three categories according to the specific responsibilities,[16] that is, the employed bees, onlooker bees and scout bees. The employed bees, as the first kind of honey bees, are responsible for generating offspring within the entire search space. To some extent, the search behavior of the employed bees can be considered as playing the role of global search. Hence, the employed bees are expected to perform strong explorative capability. In contrast, the responsibility of the onlooker bees is different from that of the employed bees. Generally, the explorative and exploitative capabilities are the two essential elements for the EAs, and a well performed EA should have a good balance between these two capabilities. So, the onlooker bees mainly concentrate on generate offspring within some promising search areas rather than the entire search space, which exhibits strong exploitative capability and has the same role of local search. In fact, to better locate the promising search areas, the onlooker bees first receive the information about the food sources from the employed bees before generating offspring. As for the scout bee, in the basic ABC, the purpose of them is not to generate offspring but to prevent population stagnation, and the reinitialization method is used to restart the abandoned food source which cannot be continuously improved for a predefined number of times. According to the number of the categories of the honey bees, the main procedure of the basic ABC is also comprised of three phases, that is, the employed bee phase, onlooker bee phase, and scout bee phase.

Before starting the main procedure, the population initialization phase is first triggered in the basic ABC. In this phase, SN food sources (individuals) are randomly generated to form the initial population, and each food source can be considered as a candidate solution to the problem at hand. Let $X_i = \{x_{i,1}, x_{i,2}, ..., x_{i,D}\}$ be the ith food source, which is produced by the following Equation (1).

$$x_{i,j} = x_j^{min} + rand_{i,j}[0, 1] \cdot \left(x_j^{max} - x_j^{min}\right) \tag{1}$$

where $x_{i,j}$ denotes the jth decision variable of the ith food source X_i, $i \in \{1, 2, ..., SN\}, j \in \{1, 2, ..., D\}$, and D denotes the dimension size of the problem. x_j^{min} and x_j^{max} represent the lower and upper boundary of the jth decision variable, respectively, and the coefficient $rand_{i,j}[0, 1]$ is a random number in the range of [0, 1]. After that, the basic ABC enters into the main procedure, which is composed of the above mentioned three phases, and they are briefly introduced as follows.

Employed bee phase—In this phase, all of the employed bees are responsible for generating offspring within the entire search space, and each employed bee corresponds to a food source, which implies that the number of employed bees is equal to that of the food sources, namely *SN*. Based on the positions of the food sources, the offspring is generated via the solution search equation listed in Equation (2).

$$v_{i,j} = x_{i,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j}) \tag{2}$$

where $V_i = \{v_{i,1}, v_{i,2}, \ldots, v_{i,D}\}$ is the offspring with respect to the food source X_i. The food source X_k is randomly selected from the population, which has to be different with X_i. The coefficient $\phi_{i,j}$ is a uniformly distributed random number within the range $[-1, 1]$. Note that, in the basic ABC, only the randomly chosen decision variable $x_{i,j}$ is changed by the solution search equation to generate offspring. In other words, the food source X_i is the same with its offspring V_i expect for the jth decision variable. Besides, to evaluate the quality of a food source, the following Equation (3) is used to calculate the fitness value.

$$fit(X_i) = \begin{cases} \dfrac{1}{1 + f(X_i)} & f(X_i) \geq 0 \\ 1 + abs(f(X_i)) & f(X_i) < 0 \end{cases} \tag{3}$$

where $fit(X_i)$ is the fitness value of X_i, $f(X_i)$ is the objective function value, and the symbol $abs(\cdot)$ denotes the absolute value. After that, to push the population towards a better evolution direction, the greed selection method is applied to X_i and V_i, and the one with the better fitness value survives into the next iteration.

Onlooker bee phase—After the employed bee phase, the onlooker bee phase is triggered accordingly. In this phase, the onlooker bees first receive the information about food sources from the employed bees, which is about the nectar amount and positions of food sources. Then, to perform fine search within some promising areas, the onlooker bees only select some good food sources for generating offspring. In general, the better the quality of a food source is, the higher the probability of being selected by the onlooker bees. The following Equation (4) shows how to calculate the selection probability p_i for the food source X_i.

$$p_i = \frac{fit(X_i)}{\sum_{j=1}^{SN} fit(X_j)} \tag{4}$$

Note that a food source may be selected for more than one time, if it has good quality. Note that the number of onlookers equals to that of employed bees. To generate offspring, the onlooker bees use the same solution search equation listed in Equation (2) as well. Next, the greed selection method is employed again for the food source and its offspring.

Scout bee phase—As the last phase of the basic ABC, the main purpose of the scout bee phase is not to generate offspring while to prevent population stagnation. So, if a food source has not been consecutively updated for more than a predefined number of times, called *limit*, then it is considered to be exhausted and a randomly generated food source will substitute it. To this end, a counter *trial* is defined for each food source, and its value will be changed in the above two phases. In other words, if the food source X_i has not been improved in either the employed bee phase or the onlooker bee phase, the associated counter $trial_i$ will be increased by one; otherwise, $trial_i$ will be reset to zero, once X_i gets improvement. In terms of the number of control parameters, in the three phases of the basic ABC, the parameter *limit* is the only single one requiring a predefined value, except for the common parameter of population size *SN*. Therefore, it is clear that the ABC has a very simple algorithmic structure, which may be one of the most salient features in comparison with other paradigms of EAs.

2.2 | Related works

To improve the performance of ABC for handling with complex problems, in recent years, various attempts have been made to design more effective ABC variants, in which how to enhance the exploitation of the solution search equation with elite information is the main way.[25–28] Besides, based on the idea of multiple strategies, many ABC variants have been proposed through integrating multiple strategies with complementary features.[31–35] Except for the two categories, most of the remaining works fall into the last category, namely hybridization with other auxiliary techniques.[36–39] In this section, the related works of the three categories are briefly introduced.

(1) Introduction of new solution search equations

As mentioned above, in the basic ABC, the solution search equation shows the drawback of strong exploitation but weak exploitation, which makes the ABC suffers from slow convergence speed and low convergence accuracy. A natural way to this drawback is to utilize the global best individual or some elite individuals, namely the elite information, to provide promising search directions. Following this idea, a number of ABC variants have been presented. For instances, Zhu and Kwong[25] proposed a gbest-guided ABC variant (GABC) by introducing the global best individual into the solution search equation, in which the valuable information of the global best individual is very helpful to guide the search. It is worth noting that the inspiration of the GABC derives from the PSO, and the new solution search equation is similar with the particle velocity update formula of the PSO. With the same idea of utilizing the elite information, Zhou et al.[26] presented a Gaussian bare-bones ABC variant (GBABC) based on the global best individual. In the GBABC, the Gaussian distribution with the elite information replaces the original solution search equation to generate offspring. Instead of only utilizing the global best individual, Kong et al.[27] designed a new solution search equation through using a group of elite individuals in their proposed ECABC variant. In the new solution search equation, the center point of the group of elite individuals is computed as the target individual, which can effectively locate promising search areas. Very recently, Zhou et al.[28] proposed the ABC variant with multielite guidance (MGABC), which is based on the group of elite individuals as well. In the MGABC, the elite individual, randomly chosen from the elite group consisting of some elite individuals, is utilized in the solution search equation. Meanwhile, a new control parameter, called *MR*, is introduced to update more decision variables for the offspring. In fact, the parameter *MR* plays the similar role with the crossover operator.

(2) Ensemble of multiple solution search equations

Although the new solution search equations with the elite information have been shown promising performance, they may not be always suitable for different optimization tasks, since different problems usually favor the solution search equations with different search capabilities. So, the ABC variants with a single solution search equation still require improvements, and the topic of how to integrate multiple solution search equations has aroused extensive interests in the community of ABC. In fact, the ABC variants with multiple solution search equations mainly involve two essential questions: (a) what kinds of solution search equations should be chosen? (b) how to determine which of the solution search equations are suitable for different evolution stages? For example, Wang et al.[31] presented a novel ensemble ABC variant (MEABC) based on the idea of integrating multiple solution search equations. In the MEABC, three solution search equations with distinct features are chosen, and all the three equations coexist throughout the search process and compete to generate offspring. Similarly, Kiran et al.[32] proposed the ABC variant with variable search strategy (ABCVSS) by simultaneously employing five different solution search equations. In the ABCVSS, to determine the selection probability of each solution search equation, a historical experience-based strategy selection mechanism is designed, which is inspired from the adaptive selection mechanism of the SaDE.[33] Being similar with the ABCVSS, in the MuABC, Gao et al.[34] developed an adaptive selection mechanism to choose the most suitable strategies based on the previous search experience. The MuABC uses three solution search equations to generate offspring as well. Recently, to reduce the computational burden of performing multiple solution search equations, Gao et al.[35] introduced the Parzen window method into the ABC, which results in a novel ABC variant called ABCPW. In the ABCPW, three different solution search equations are used to produce offspring simultaneously, and this implies that each food source corresponds to three offspring individuals. Then, the Parzen window method is used to estimate the quality of each offspring individual instead of using the original fitness function, and the best one among the three offspring individuals could survive into the next iteration. Note that, in fact, the ABCPW employs the idea of cheap surrogate model.

(3) Hybridization with other auxiliary techniques

Except for the above two categories of ABC variants, the hybrid ABC variants also have been shown competitive performance, which mainly focus on how to combine other auxiliary techniques to improve the performance of the ABC. For example, to meet the requirements of some applications with very limited data storage, Banitalebi et al.[36] proposed an enhanced compact ABC variant (EcABC) through combining the ABC with the EDA framework. The virtual population in the EcABC is expressed through the EDA framework by the probability density functions, which is able to make full use of the memory saving ability of EDA. Being similar with the idea of combining two paradigms of EAs in the EcABC, Jadon et al.[37] developed a hybrid ABC variant with the DE (HABCDE), in which the onlooker bee phase is implemented by the DE operators, while the other components are kept the same with the ABC. The reported experimental results verified that the HABCDE shows better performance than both of the ABC and DE. To enhance the exploitation of ABC, Zhou et al.[38] introduced a neighborhood search operator into the ABC (MABC-NS), and the introduced operator is executed after the main procedure of ABC with the aim of performing fine search. Recently, Wang et al.[39] proposed a new ABC variant (NSABC) with the concept of neighborhood selection, which aims to overcome the shortcoming of the probability selection mechanism in the onlooker bee phase. In the new neighborhood selection strategy, all the food sources are first collected to form a ring topology, and then a neighborhood radius is defined for each food source to determine the available neighbor range. Compared with the original probability selection mechanism, the new neighborhood selection strategy can effectively enhance the comprehensive performance of ABC.

3 | OUR APPROACH

3.1 | Motivation

In the basic ABC, as the main way to generate offspring, the solution search equation has significant impact on the performance of ABC. However, the solution search equation suffers from several drawbacks resulting in the poor performance on some complex problems. One widely recognized issue is that the solution search equation does well in exploration but weakly in exploitation, which causes the problems of slow convergence speed and low convergence accuracy. To address the issue, in recent years, researchers have proposed various solutions among which the ones with the idea of utilizing elite information have been shown effectiveness. To some extent, the issue of weak exploitation has been well addressed. However, the other issue that the solution search equation is sensitive to rotation on the coordinate system is almost ignored, and only a few works have been dedicated to solve it. To be specific, in the solution search equation, the offspring is obtained by changing one decision variable of the target individual, which makes the search dimension-wise and more suitable for separable problems. But for the nonseparable problems, that is, correlations exist among decision variables, the dimension-wise search pattern is no longer effective. So, the issue still limits the performance of ABC, and more efforts should be paid to the issue.

To better explain the issue, an intuitional example, named Example A, is given on the two-dimensional rotated ellipsoidal function. The definition of this function is listed as follows, where the origin (0,0) is the global optimum solution.

- Rotated ellipsoidal function: $f(X) = \sum_{i=1}^{2}(\sum_{j=1}^{i}X_i)^2, X \in [-6, 6]^2, f(0, 0) = 0$.

In the Example A, two different cases are used to demonstrate the issue. In the first case, an offspring is generated in the original coordinate system, while in the second case the other offspring is generated in the rotated coordinate system. If the two offspring do not match with each other, the issue is demonstrated that the solution search equation is not rotationally invariant. To have a fair comparison, the two cases are under the same conditions to generate offspring according to Equation (2), which are listed as follows: the second decision variable, namely x_2, is the selected dimension, the corresponding coefficient $\phi_{i,2}$ is 0.5, and the two parent individuals are $X_i = (1, 4)$ and $X_k = (5, 2)$. Moreover, this example is illustrated in Figure 1, and the two cases are shown in the subfigures, respectively.

For the first case, in the original coordinate system, $X_i = (1, 4)$ is the target individual to produce the offspring V_{i1}. Under the above conditions, the position of V_{i1} can be calculated, namely $V_{i1} = (1, 5)$. Figure 1A shows the specific positions of the involved individuals: the two red points denote the parent individuals X_i and X_k, and the green point represents the offspring V_{i1}. Besides, the contour lines and the global best solution (the red pentagon X^*) of the function are depicted as well. As seen, V_{i1} locates on the extension of the borders of the rectangle formed by X_i and X_k, which implies that there exists one identical dimension between V_{i1} and X_i.

In the second case, we rotate the coordinate system 90° clockwise to obtain a rotated coordinate system. In the rotated coordinate system, both X_i and X_k are used again to generate the offspring under the same conditions. Note that the positions of X_i and X_k are accordingly changed with the rotation, but their relative locations remain unchanged. Figure 1B shows the new positions of X_i and X_k, which are denoted as $\tilde{X}_i$ and $\tilde{X}_k$ (red points), respectively. With respect to the target individual $\tilde{X}_i$, the offspring $\tilde{V}_{i2}$ can be generated through Equation (2), which is represented by the blue point. Being similar with the first case, $\tilde{V}_{i2}$ is also locates on the

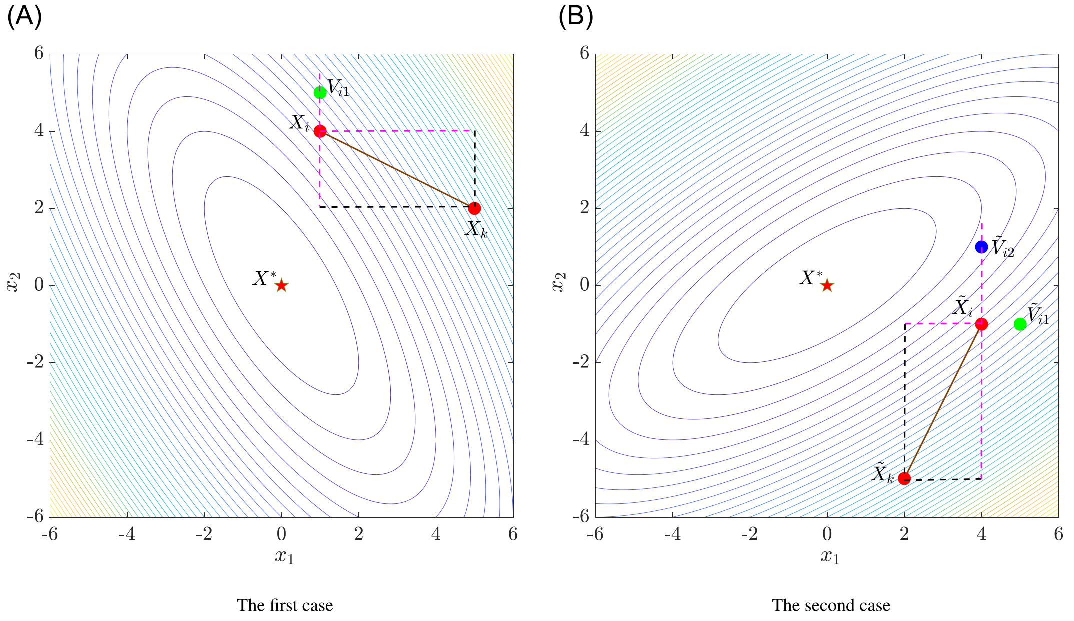

FIGURE 1 Illustration of the Example A. (A) The first case; (B) The second case [Color figure can be viewed at wileyonlinelibrary.com]

extension of the borders of the rectangle formed by $\tilde{X}_i$ and $\tilde{X}_k$. For convenience, the new position of V_{i1} in the rotated coordinate system is depicted in Figure 1B, namely the green point $\tilde{V}_{i1}$. It is clear that $\tilde{V}_{i1}$ and $\tilde{V}_{i2}$ do not match with each other, which implies that the solution search equation is not rotationally invariant.

In fact, in the search space, the rotation on the coordinate system can be treated as a transformation between two individuals, and the transformation can be expressed by a matrix. For instance, in the above Example A, the matrix $R = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ can express the clockwise rotation of 90°, since the rotation matrix can be given by the matrix $\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$, where θ is the rotation angle. So, for an arbitrary individual, the relationship between the positions in the two coordinate systems satisfy the following requirement:

$$\tilde{X} = X \cdot R \tag{5}$$

where $\tilde{X}$ denotes the position in the rotated coordinate system, X is the position in the original coordinate system, and R is the matrix rotating the coordinate system. According to Equation (5), it is not difficult to calculate the new position in the rotated coordinate system. Hence, $\tilde{X}_i = (4, -1)$, $\tilde{X}_k = (2, -5)$, and $\tilde{V}_{i1} = (5, -1)$. Furthermore, with the positions of $\tilde{X}_k$ and $\tilde{V}_{i1}$, $\tilde{V}_{i2} = (4, 1)$ can be obtained through Equation (2). As seen, the two offsprings $\tilde{V}_{i1}$ and $\tilde{V}_{i2}$ have distinct positions.

It is also interesting to observe that $\tilde{V}_{i2}$ has better quality than $\tilde{V}_{i1}$, since the contour line covered by $\tilde{V}_{i2}$ is closer to the global optimum solution X^*. So, if we rotate the rotated coordinate system back to the original, namely from Figure 1B to Figure 1A, the quality of $\tilde{V}_{i2}$ is still better than that of $\tilde{V}_{i1}$, because the relative location of $\tilde{V}_{i1}$ and $\tilde{V}_{i2}$ remains unchanged, which is the same with that of X_i and X_k. To better verify this, in the original coordinate system, we can compare the fitness values between $\tilde{V}_{i1}$ and $\tilde{V}_{i2}$. Hence, to obtain the position of $\tilde{V}_{i2}$ in the original coordinate system, the following Equation (6) can be derived from Equation (5), where R' is the inverse matrix of R.

$$X = \tilde{X} \cdot R' \tag{6}$$

According to the above Equation (6), the position of $\tilde{V}_{i2}$ in the original coordinate system can be obtained, denoted as $V_{i2} = (-1, 4)$. With the position of $V_{i1} = (1, 5)$, the fitness values of $\tilde{V}_{i1}$ and $\tilde{V}_{i2}$ can be calculated, namely $f(V_{i1}) = 37$ and $f(V_{i2}) = 10$. It is clear that V_{i2} is better than V_{i1}, which implies that, to some extent, the hardness of solving a nonseparable problem may be relieved by rotating the coordinate system. Further, to solve the deficiency of rotational variability associated with the solution search equation, we are motivated to investigate how to use the method of rotating the coordinate system for a higher effective ABC variant.

3.2 | Details of BSABC

3.2.1 | ABC with bi-coordinate systems

Through the Example A, we can learn that the method of rotating the coordinate system does work for the ABC, but how to obtain an appropriate rotation matrix R remains a challenge. In fact, in other paradigms of EAs, such as the DE, the method of rotating coordinate system has been well studied. In the DE, the crossover operator also shows the deficiency of rotational variability, which is similar with the solution search equation of the ABC. Thus, to solve the deficiency of the DE, the method of rotating the coordinate system is also used by some researchers, they rotate the original coordinate system to the eigen coordinate system. For instance, Guo et al.[30] proposed an eigenvector-based crossover operator for DE (DE/eig), in which the binomial crossover operator is not only implemented in the original coordinate system but also in the eigen coordinate system. The reported extensive experiments have verified the effectiveness of DE/eig. Similarly, Wang et al.[40] presented a novel DE variant based on covariance matrix learning, called CoBiDE. In the CoBiDE, the offspring population is generated in either the original coordinate system or the eigen coordinate system, and the probability of using the respective coordinate system is predefined.

Inspired by these previous studies, in this study, we also rotate the original coordinate system to the eigen coordinate system for the ABC, namely the solution search equation is implemented in the eigen coordinate system. Meanwhile, the original coordinate system is retained for taking advantages of the bi-coordinate systems, and the resulting ABC variant is called BSABC. To better explain the BSABC, the following three parts are described. (1) The first part is to present the construction process of the eigen coordinate system, including how to obtain the rotation matrix R. (2) In the second part, the Example B is given to demonstrate the different search behaviors of the solution search equation in the two coordinate systems. (3) The last part describes how to use the bi-coordinate systems for the ABC.

(1) First part

As mentioned above, the rotation of the coordinate system can be expressed by a matrix. So, how to obtain the rotation matrix R is the key to construct the eigen coordinate system. The following two steps are given to obtain the rotation matrix R.

Step 1: Select parts of individuals from the population to calculate the covariance matrix C, which can represent the search distribution of the population to some extent. Equations (7) and (8) show the details of obtaining C, where M is the mean vector of the N individuals, and $(\cdot)^T$ denotes a transpose operation.

$$M = \frac{\sum_{i=1}^{N} X_i}{N-1} \tag{7}$$

$$C = \frac{\sum_{i=1}^{N} (X_i - M)(X_i - M)^{\mathrm{T}}}{N-1} \tag{8}$$

Step 2: To obtain the rotation matrix R, the covariance matrix C should be eigen-decomposed according to the following Equation (9), where R denotes the rotation matrix, and R' is the inverse matrix of R.[41]

$$C = R \wedge^2 R' \tag{9}$$

As for the matrix R, it is a $D \times D$ square matrix whose columns are mutually orthogonal, and each column is an eigenvector of C which represents an axis of the eigen coordinate system.[42] The matrix R' represents the transformation from the eigen coordinate system to the original coordinate system, and $\wedge$ is a diagonal matrix composed of eigen values. With the matrixes R and R', the positions of an arbitrary individual in the two coordinate systems can be switched by using Equations (5) and (6).

(2) Second part

In the Example A, we have shown that the method of rotating the coordinate system is effective in solving the deficiency of rotational variability associated with the solution search equation, to some extent. To further capture the different search behaviors of the solution search equation in the two coordinate systems, the Example B is given here. In the Example B, we conduct two cases based on the rotated high conditioned elliptic function defined in the CEC2013 test suite.[43] In the two cases, the basic ABC is used to optimize the function with dimension size $D = 2$, and its parameter settings include: $SN = 50$ and $limit = 200$. In the first case, all offspring individuals are generated in the original coordinate system, while in the second case that they are generated in the eigen coordinate system. The distributions of the offspring population during the search process can be used to observe the differences between the search behaviors in the two coordinate systems. So, we record the positions of all individuals over different iterations, that is, the 1st, 10th, 20th, and 40th iterations.

Figure 2 shows the positions of all individuals in the two cases on the landscape contour, where the red points denote the individuals in the Case 1, and the blue points are the individuals in the Case 2. It is clear that the individuals of the two cases scatter evenly in the search space at the first iteration, but the gap between the two cases gradually expands as the iteration increases. As for the Case 1, the red points are spread out on a wide range. But, in contrast, regarding the Case 2, the blue points locate along an oblique line and are closer to the valley area, namely they tend to correspond with the problem landscape. Besides, the blue points converge faster than the red points. To some extent, the Example B shows the fact that the search behavior in the eigen coordinate system seems to be more exploitative, since the problems with high variable correlation can be transformed into pseudo separable by the method of rotating the coordinate system.

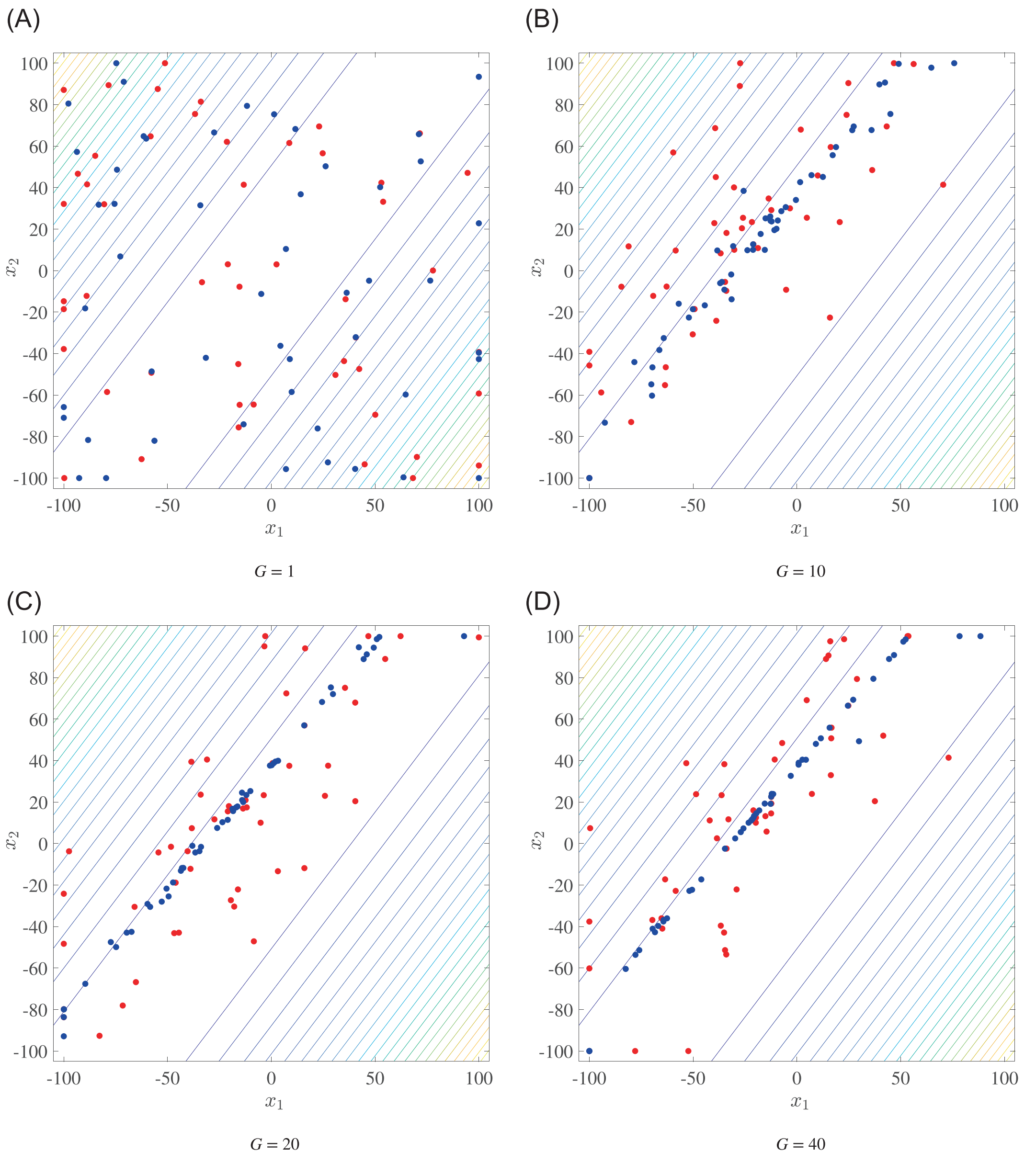

FIGURE 2 Illustration of the Example B (A) $G = 1$; (B) $G = 10$; (C) $G = 20$; (D) $G = 40$ [Color figure can be viewed at wileyonlinelibrary.com]

(3) Third part

Through the Example B, we can observe that the solution search equation has different search behaviors in the two coordinate systems. So, to take advantages of the two coordinate systems, both of them are used in the search process of BSABC with the aim of meeting the intrinsic requirement of ABC. For the original coordinate system, it is only used in the employed bee phase, since the employed bees are responsible for exploring the entire search space to generate offspring individuals, which favor the exploration-biased search behavior. Unlike the original coordinate system, the eigen coordinate system is only used in the onlooker bee phase, because the onlooker bees have the responsibility of performing fine search within some promising areas, which prefer the exploitation-biased search behavior. In summary, in terms of the algorithm framework, the BSABC uses the eigen coordinate system in the onlooker bee phase, which is different from the basic ABC. To better describe the details about how to use the eigen coordinate system in the onlooker bee phase, the following two steps are given.

Step 1: Randomly select $SN/2$ individuals from the population to calculate the rotation matrix R through Equation (9). Calculate the new positions of all parent individuals in the eigen coordinate system according to Equation (5), and generate the offspring $\tilde{V}_i$ through the solution search equation.

Step 2: Rotate the eigen coordinate system back to the original coordinate system, and calculate the position of $\tilde{V}_i$ in the original coordinate system through Equation (6). With the position of $\tilde{V}_i$, calculate the fitness value of $\tilde{V}_i$, and select the better one between $\tilde{V}_i$ and its parent individual.

To the best of our knowledge, the AELABC[44] was the first attempt to combine the two coordinate systems. In the AELABC, the two coordinate systems are used simultaneously in the employed bee phase and the onlooker bee phase, and an adaptive selection mechanism is designed to determine the selection probabilities of the two coordinate systems, which depends on the historical performance of each coordinate system. Compared with the AELABC, the BSABC differs from it in the following major aspects.

(a) The usage scenarios of the two coordinate systems are different. In the AELABC, the two coordinate systems are used synchronously, but each coordinate system is selected by the adaptive selection mechanism. On the contrary, in the BSABC, the eigen coordinate system is only implemented in the onlooker bee phase, and the original coordinate system is still kept in the employed bee phase. Because different honey bees have different responsibilities, and they favor different search behaviors. Hence the usage scenario in the BSABC is more suitable for the intrinsic requirement of the ABC, which also has the advantages of simpler structure but better performance. This point will be verified by the performance comparison between the BSABC and AELABC in the later Section 4.4.

(b) The methods of selecting the individuals used to calculate the covariance matrix C are different. In the AELABC, the best $SN/2$ individuals are selected to calculate the covariance matrix C. In contrast, in the BSABC, the randomly selected $SN/2$ individuals are used. As mentioned above, the covariance matrix C can represent the search distribution of the population to some extent. Therefore, the search distribution in the AELABC may be biased due to the attraction of the best individuals, whereas that in the BSABC would be more reasonable. In the later Section 4.4, both of the two selection methods are separately introduced into the BSABC to make a contrast, and the comparison results confirm the random selection method is more effective.

3.2.2 | New solution search equations with elite information

In the basic ABC, the solution search equation showed in Equation (2), as the major way to generate offspring, significantly affects the algorithm performance. However, as previously mentioned, the solution search equation does well in exploration but poorly in exploitation. So, it is necessary to design a more effective solution search equation to balance the capabilities between exploration and exploitation. In then above Section 2.2, we have reviewed one category of ABC variants, which mainly focuses on the introduction of new solution search equations with elite information.[25–28] For instance, in the GABC,[25] the global best individual is added into the solution search equation as an extra term. In the ECABC,[26] a new solution search equation based on a group of elite individuals is designed. Similarly, being inspired by these works, we also design two new solution search equations with elite information. But, it should be noted that the purpose is to further improve the performance of ABC, while not be another contribution of our approach.

As for the employed bee phase, it has been showed in Section 2.1 that the employed bees have the responsibility of exploring the entire search space to generate offspring. Therefore, for the employed bees, its search behavior should be emphasized on exploration but not be as strong as the original solution search equation. For this aim, based on the conception of elite group, the following Equation (10) is designed for the employed bee phase.

$$v_{i,j} = x_{e,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j}) \tag{10}$$

where $X_e = (x_{e,1}, x_{e,2}, \ldots, x_{e,D})$ is an elite individual, which is randomly selected from the elite group. Note that the elite group is constructed before the execution of the employed bee phase, which consists of the top $q \cdot SN$ individuals of the population according to the fitness value. For the control parameter $q \in (0, 1)$, it can control the size of the elite group, which has significant effect on the performance of the solution search equation. If the value of q is near to 0, X_e can be considered as the global best individual; however, if the value of q tends to be 1, X_e will become a ordinary individual rather than an elite individual. In the BSABC, we recommend that the value of q is set to 0.1, since it can maximize the algorithm performance. Note that, the sensitivity test for q setting will be done in Section 4.3.

In contrast to the employed bee phase, in the onlooker bee phase, the onlooker bees are responsible for selecting some promising areas to conduct a fine search. In other words, the search behavior of the onlooker bees is expected to concentrate on exploitation. In the GABC, the exploitative capability is enhanced by using the information from the global best individual, and its solution search equation is listed as follows:

$$v_{i,j} = x_{i,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j}) + \psi_{i,j} \cdot (x_{gb,j} - x_{i,j}) \tag{11}$$

where X_{gb} denotes the global best individual of the whole population, and $\psi_{i,j}$ is set to 1.5. This solution search equation has been verified to be very effective. So, in the BSABC, we design the following Equation (12) for the onlooker bee phase, which derives from the above Equation (11).

$$v_{i,j} = x_{gb,j} + \phi_{i,j} \cdot (x_{i,j} - x_{k,j}) + \psi_{i,j} \cdot (x_{gb,j} - x_{i,j}) \tag{12}$$

where $\psi_{i,j}$ is a uniformly distributed random number within range $[-0.5, 0.5]$. Note that, compared with Equation (11), the first term of the solution search equation is different, and the coefficient of the last term $\psi_{i,j}$ ranges from -0.5 to 0.5 for reducing disturbance strength.

In conclusion, to keep a better balance between the exploration and exploitation, we design two new solution search equations with elite information based on the inspiration from the previous studies. According to the division of labor of the honey bees, the two solution search equations are used in the employed bee phase and onlooker bee phase, respectively. It is necessary to note again that the two solution search equations are not the contribution of our work, since the main focus of our work is on how to use the bi-coordinate systems to solve the problem of rotational variability for the ABC, rather than designing new solution search equations.

3.2.3 | Modified scout bee phase

In the basic ABC, if a food source X_i cannot be continuously improved for at least *limit* times, it will be considered as an abandoned food source, and a new food source should be generated to substitute it. In fact, the abandoned food source could be interpreted as a signal that the population may be trapped into a local optimum and the problem of population stagnation has been triggered accordingly. To solve this problem, in the basic ABC, the random restart method is used, which reinitializes the abandoned food source according to Equation (1). It is not difficult to note that this method plays the same role with the procedure of the population initialization. In essence, from the perspective of sampling, the random restart method can be considered as a uniform sampling of the search space. Although it is effective to break population stagnation, it may also cause the side effect that the already obtained search experience would be lost. Because the abandoned food source typically locates in a promising search area, which can reflect the search distribution of the population to a certain extent.

Algorithm 1 Pseudocode of the BSABC

```
Generate SN food sources {X_i | i = 1, 2, ..., SN} as the initial population according to the Eq. (1);
while FEs ≤ MaxFEs do
    // Employed bee phase
    for i = 1 → SN do
        Randomly choose an elite solution X_e from the elite group;
        In the original coordinate system, generate a new food source V_i for the parent food source X_i according to the Eq. (10);
        Evaluate fitness value for V_i and set FEs=FEs+1;
        if fit(X_i) < fit(V_i) then
            Replace X_i with V_i and set trial_i = 0;
        else
            Keep X_i to be the new food source and set trial_i=trial_i + 1;
        end if
    end for
    // Onlooker bee phase
    Calculate the rotation matrix R by randomly selecting SN/2 individuals from the population according to the Eq. (9);
    Calculate the new position of each food source in the eigen coordinate system according to the Eq. (5);
    Calculate the selection probability p_i for each food source according to the Eq. (4);
    Set t = 0 and i = 0;
    while t < SN do
        if rand(0, 1) < p_i then
            Generate a new food source Ṽ_i in the eigen coordinate system for the selected parent food source X_i according to the Eq. (12);
            For Ṽ_i, calculate its position in the original coordinate system, denoted as V_i, according to the Eq. (6);
            Evaluate the fitness value for V_i and set FEs=FEs+1;
            if fit(X_i) < fit(V_i) then
                Replace X_i with V_i and set trial_i = 0;
            else
                Keep X_i to be the new food source and set trial_i=trial_i + 1;
            end if
            Set t = t + 1;
        end if
        Set i = (i + 1)%SN;
    end while
    // Scout bee phase
    Calculate the mean vector M and the covariance matrix C through the Eqs. (7) and (8);
    if max(trial_i) > limit then
        Generate a new food source to replace X_i according to the Eq. (13);
        Set FEs=FEs+1 and trial_i = 0;
    end if
end while
```

To void the above side effect, in the BSABC, a multivariate Gaussian distribution is constructed to restart the abandoned food source instead of using the random restart method. To save the search experience, the multivariate Gaussian distribution is based on the elite individuals of the population, which is listed as follows.[45]

$$X = \mathcal{N}(M, C) \tag{13}$$

where X is the new food source to substitute the abandoned food source, $\mathcal{N}(\cdot)$ denotes the multivariate Gaussian distribution with mean vector M and the covariance matrix C.[45] As seen, X is sampled from $\mathcal{N}(\cdot)$, and its position is determined by M and C. As mentioned above, the elite individuals of the population are used to construct $\mathcal{N}(\cdot)$ for keeping the search experience. To this aim, the top $SN/2$ individuals are used, and the center point of them determines the position of the mean vector M. As for the covariance matrix C, it can represent the search distribution of the population to some extent. So, in this scenario, the new food source X can inherit useful information from the elite individuals, and the search experience would be saved.

In addition, for the scout bee phase, the parameter *limit* controls its execution frequency and has important effect on the algorithm performance. If the value of *limit* is set to be very small, it is possible that the frequency of the food sources being abandoned would be great. Although this is helpful to avoid being trapped into local optimal, it easily loses the already obtained search experience to a certain extent. However, if the value of *limit* is too large, the food sources may have little chance to be abandoned, and the risk of population stagnation will increase, then the scout bee phase becomes meaningless. The specific setting for *limit* will be discussed in Section 4.3.

3.3 | Framework of BSABC

In this section, to better describe the complete procedure of the BSABC, its pseudocode is given in the Algorithm 1. In there, *FEs* denotes the used number of fitness function evaluations, and *MaxFEs* is the maximal value of *FEs*, which is also the termination condition to stop the algorithm. As seen, in comparison with the basic ABC, there are mainly two modifications for the BSABC. First, the bi-coordinate systems are used instead of only using the original coordinate system. Second, a modified scout bee phase is proposed, which employs the multivariate Gaussian distribution[45] to generate offspring for saving the search experience.

4 | EXPERIMENTAL STUDY

To verify the effectiveness of the BSABC, a comprehensive set of experiments is conducted on the CEC2013 test suite and a real-world optimization problem, which is divided into the following five parts. (1) Effectiveness verifications of the BSABC, including the method of bi-coordinate systems and the modified scout bee phase. (2) Sensitivity analysis of the control parameters, including the parameters *limit* and q. (3) Performance comparison with other advanced ABC variants. (4) Performance comparison with other EAs. (5) Performance comparison on a real-world optimization problem.

4.1 | Test suite and parameter settings

In the experiments, the CEC2013 test suite is used, which is also widely used by other researchers. The benchmark functions of the test suite can be divided into three categories according to the function characteristics: (1) F01–F05 are unimodal functions, (2) F06–F20 are basic multimodal functions, and (3) F21–F28 are composition functions. All the benchmark functions share the same search range: $[-100, 100]^D$, where D denotes the dimension size of the functions. In the following experiments, the test dimension size includes two cases: $D = 30$ and 50. In addition, the configurations of the computational platform are listed as follows:

- Operating system: Windows 10 (×64);
- CPU: Inter Core i7-8700 (3.2 GHz);
- RAM: 16 GB;
- Programming language: Matlab;
- Compiler: MATLAB R2021a.

As for the parameter settings, there are two cases to be considered. On the one hand, for the common parameters, the maximal number of fitness function evaluations (*MaxFEs*) is set to $10000 \cdot D$ according to the suggestions from the test suite designers,[43] and the number of food sources *SN* is set to 50. On the other hand, for the specific parameters in the BSABC, the parameter q of controlling the size of the elite group is set to 0.1, and the parameter *limit* equals to 50. To reduce the impact of randomness as much as possibly, the algorithm is independently run 30 times for each benchmark function, and the average result is recorded as the final result.

4.2 | Effectiveness verifications of the BSABC

In this study, two modifications are proposed for the BSABC, namely the method of bi-coordinate systems, and the restart method based on the multivariate Gaussian distribution[45] for the scout bee phase. In this section, both of the two modifications are verified to check whether they indeed work for improving the performance of the BSABC.

4.2.1 | Effectiveness verification of the bi-coordinate systems

In this experiment, to verify the effectiveness of the bi-coordinate systems of the BSABC, we construct three comparative algorithms, which are listed as follows.

- $BSABC_{EO}$: The eigen coordinate system is implemented in the employed bee phase, while the original coordinate system is in the onlooker bee phase.
- $BSABC_{EE}$: The eigen coordinate system is implemented in both the employed bee phase and onlooker bee phase.
- $BSABC_{OO}$: The original coordinate system is implemented in both the employed bee phase and onlooker bee phase.

As for the first comparative algorithm, namely the $BSABC_{EO}$, the usage scenarios of the two coordinate systems are switched in comparison with the BSABC. Regarding to the two later comparative algorithms, that is, the BSABC_{EE} and BSABC_{OO}, both of them only use a single coordinate system through out the entire procedure. Note that the three algorithms have the same components with the BSABC except for the coordinate system, which is helpful to conduct a fair comparison. So, by comparing the performance of the three comparative algorithms, it is not difficult to verify whether the two coordinate systems have positive effects on the performance of the BSABC. The test dimension size of the CEC2013 test suite is set to $D = 30$, and the other parameter settings are kept the same as in the above Section 4.1. Besides, to make the comparison results sound statistically, the nonparametric Wilcoxon's rank sum test is used at the significance level $\alpha = 0.05$.

Table 1 presents the comparison results of the four involved algorithms, and the best results are marked in boldface. The results of the Wilcoxon's rank sum test are summarized in the last row of Table 1, where the symbols "+", "=", and "−" indicate that the result of the comparative algorithm is better than, similar to, and worse than that of the BSABC. It can be clearly observed from Table 1 that, on the whole, the performance of the BSABC is on the first place among the four involved algorithms by achieving the best results on most of the benchmark functions. To be specific, in comparison with the BSABC_{EO}, the BSABC outperforms it on nine functions and only loses on two functions. The comparison results between the BSABC and BSABC_{EO} can confirm that the usage scenario of the two coordinate systems in the BSABC is more effective than that in the BSABC_{EO}. The reason is that the responsibilities of different phases in the ABC have been considered in designing the usage scenario of the two coordinate systems. So, we can say that the original coordinate system is more suitable for the employed bee phase, while the eigen coordinate system is more fit for the onlooker bee phase.

As for the other two comparative algorithms, namely the BSABC_{EE} and BSABC_{OO}, both of them are surpassed by the BSABC. Specifically, the BSABC defeats the BSABC_{EE} and BSABC_{OO} on 17 and nine functions, respectively. Hence, we can draw the conclusion that the BSABC with bi-coordinate systems performs much better than the competitors with just a single coordinate system. It is also interesting to note that, from the comparison results between the BSABC and BSABC_{EE}, the performance of the BSABC_{EE} is much deteriorated by using the eigen coordinate system on both the employed bee phase and onlooker bee phase simultaneously. The main reason is that the search behavior of the algorithm becomes overly greedy due to the exploitation-biased feature of the eigen coordinate system. So, it is an important task to choose the right usage scenarios for the two coordinate systems. The comparison results between the BSABC and BSABC_{OO} reveal that the introduction of the eigen coordinate system is indeed beneficial to boost the performance on the problems with high variable correlation, for example, the F01, F02, F04, and F10.

To simplify the overall performance comparison among the four involved algorithms, the Friedman test is used to obtain the average rankings which are shown in Figure 3. As seen, the BSABC ranks on the first place by getting the smallest ranking value, and the next places are obtained by the BSABC_{EO}, BSABC_{OO}, and BSABC_{EE}. In conclusion, the experiment carried out in this section confirms the effectiveness of the bi-coordinate systems, and their usage scenarios in the BSABC are better than other possible schemes.

4.2.2 | Effectiveness verification of the modified scout bee phase

In the scout bee phase of the BSABC, a new restart method based on the multivariate Gaussian distribution[45] is proposed to replace the original method, which generates offspring through

TABLE 1 Comparison results of the bi-coordinate systems

Func.	$BSABC_{EO}$	$BSABC_{EE}$	$BSABC_{OO}$	BSABC
F01	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00**
F02	1.29E+02±3.44E+02−	1.04E+01±2.75E+01−	4.16E+06±1.44E+06−	**1.37E−04±7.39E−04**
F03	5.99E+05±4.32E+05−	7.60E+05±8.27E+05−	1.92E+06±2.95E+06−	**2.69E+05±3.03E+05**
F04	1.29E−13±1.29E−13=	**3.78E−14±8.60E−14+**	4.73E+04±5.41E+03−	9.08E−14±1.13E−13
F05	6.46E−14±5.75E−14=	9.39E−08±1.88E−07−	7.22E−14±5.59E−14=	**5.32E−14±5.78E−14**
F06	1.03E+01±3.44E+00−	1.36E+01±1.23E+01−	1.85E+01±3.25E+00−	**8.61E+00±2.20E+00**
F07	2.60E+01±7.43E+00−	4.19E+01±1.23E+01−	2.66E+01±6.72E+00−	**2.24E+01±7.98E+00**
F08	2.10E+01±5.56E−02=	2.10E+01±5.71E−02=	**2.09E+01±5.71E−02=**	**2.09E+01±5.68E−02**
F09	1.26E+01±4.22E+00=	1.30E+01±2.58E+00=	1.50E+01±4.99E+00−	**1.24E+01±3.17E+00**
F10	4.15E−04±1.89E−03=	3.67E−05±1.23E−04=	3.16E−01±1.68E−01−	**3.64E−05±7.90E−05**
F11	**0.00E+00±0.00E+00=**	5.03E+01±1.25E+01−	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00**
F12	3.82E+01±7.85E+00−	4.22E+01±6.78E+00−	3.54E+01±8.82E+00=	**3.30E+01±9.89E+00**
F13	6.34E+01±1.38E+01−	7.79E+01±1.99E+01−	5.49E+01±1.68E+01=	**5.31E+01±1.41E+01**
F14	4.74E+01±6.03E+01+	3.62E+03±4.69E+02−	**4.13E−01±1.01E+00+**	5.78E+01±2.77E+01
F15	3.73E+03±4.14E+02=	**3.32E+03±5.62E+02+**	3.58E+03±6.61E+02=	3.65E+03±5.39E+02
F16	1.76E+00±2.67E−01=	**1.27E+00±2.43E−01+**	1.90E+00±2.72E−01=	1.78E+00±2.62E−01
F17	3.07E+01±1.27E−01−	8.20E+01±1.43E+01−	**3.04E+01±2.54E−02=**	**3.04E+01±1.45E−14**
F18	**8.51E+01±1.18E+01=**	8.65E+01±2.79E+01=	8.73E+01±1.34E+01=	8.91E+01±1.45E+01
F19	1.18E+00±4.42E−01+	3.76E+00±6.23E−01−	**5.04E−01±4.07E−01+**	1.95E+00±5.80E−01
F20	1.15E+01±2.05E+00=	1.23E+01±1.81E+00−	1.13E+01±2.08E+00=	**1.09E+01±1.56E+00**
F21	2.49E+02±6.13E+01=	3.00E+02±9.19E+01−	3.18E+02±9.96E+01−	**2.35E+02±5.31E+01**
F22	1.16E+02±6.91E+01−	3.27E+03±5.39E+02−	1.04E+02±3.73E+01=	**1.01E+02±4.09E+01**
F23	3.62E+03±5.98E+02=	3.57E+03±4.62E+02=	3.56E+03±5.94E+02=	**3.38E+03±5.95E+02**
F24	**2.13E+02±5.52E+00=**	2.21E+02±7.36E+00−	2.22E+02±8.37E+00−	**2.13E+02±6.85E+00**
F25	**2.49E+02±5.86E+00=**	2.59E+02±6.78E+00−	2.51E+02±6.91E+00=	**2.49E+02±4.54E+00**
F26	**2.00E+02±0.00E+00=**	**2.00E+02±0.00E+00=**	**2.00E+02±0.00E+00=**	**2.00E+02±0.00E+00**
F27	4.60E+02±8.96E+01−	5.04E+02±8.66E+01−	4.52E+02±9.27E+01=	**4.21E+02±7.95E+01**
F28	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00**
+/=/−	2/17/9	3/8/17	2/17/9	− −

Abbreviations: BSABC, our proposed approach; $BSABC_{EE}$, BSABC with the single eigen coordinate system; $BSABC_{EO}$, BSABC with switched usage scenario of the bi-coordinate systems; $BSABC_{OO}$, BSABC with the single original coordinate system.

sampling the constructed multivariate Gaussian distribution.[45] So, compared with the basic ABC, the scout bee phase is modified. To verify the effectiveness of the modified scout bee phase, in this section, we conduct two sets of experiments with different purposes. The first set of experiments is to check whether the proposed new method has advantages in comparison

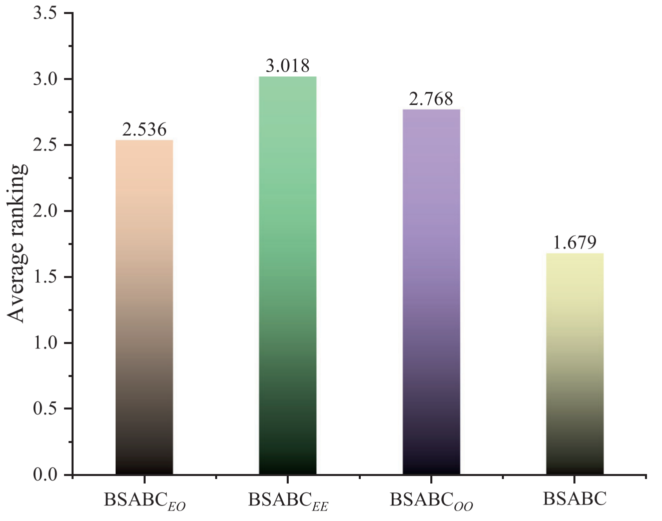

FIGURE 3 Average rankings of the BSABC$_{EO}$, BSABC$_{EE}$, BSABC$_{OO}$, and BSABC [Color figure can be viewed at wileyonlinelibrary.com]

with other methods of the previous related works. Unlike the first set of experiments, the second one aims to verify the effectiveness of the method of constructing the multivariate Gaussian distribution,[45] since there exist some other possible ways. Both sets of experiments are carried out on the CEC2013 test suite with $D = 30$, and the parameter settings are kept the same as in the above Section 4.1.

In the first set of experiments, three other different methods of the previous works are compared with our method, which include the original method,[46,47] the opposition-based learning strategy,[26,39] and the global neighborhood search operator.[38,48,49] Based on these three methods, we construct three corresponding comparative algorithms, which are listed as follows:

- BSABC$_{RND}$: BSABC with the random restart method.
- BSABC$_{OBL}$: BSABC with the opposition-based learning strategy.
- BSABC$_{GNS}$: BSABC with the global neighborhood search operator.

The first one is the BSABC$_{RND}$, which uses the original method in the scout bee phase. The latter two ones, namely the BSABC$_{OBL}$ and BSABC$_{GNS}$, use the the opposition-based learning strategy and the global neighborhood search operator, respectively. Note that the above three algorithms are the same with the BSABC except for the scout bee phase. So, in the first set of experiments, there are four involved algorithms in total.

Table 2 presents the comparison results of the four involved algorithms, and the best results are marked in boldface. Similarly, the Wilcoxon's rank sum test is used as well. As seen, the BSABC does not lose on any test function in comparison with the BSABC$_{RND}$ and BSABC$_{OBL}$. Hence, we can conclude that the proposed restart method can show much better performance than the original method and the opposition-based learning strategy. As for the BSABC$_{GNS}$, it achieves better results on two functions compared with the BSABC, but the BSABC defeats it on the other 18 functions. In brief, based on the comparison results, the high effectiveness of the proposed restart method can be verified. Figure 4 gives the average rankings of the above

TABLE 2 Comparison results of the modified scout bee phase in the the first set of experiments

Func.	BSABC$_{RND}$	BSABC$_{OBL}$	BSABC$_{GNS}$	BSABC
F01	5.91E−13±1.28E−13−	5.91E−13±1.13E−13−	5.08E−13±1.42E−13−	**0.00E+00±0.00E+00**
F02	1.66E+05±1.55E+05−	2.28E+05±1.75E+05−	2.90E+04±1.90E+04−	**1.37E−04±7.39E−04**
F03	3.13E+08±1.61E+08−	2.45E+08±1.74E+08−	1.51E+07±1.24E+07−	**2.69E+05±3.03E+05**
F04	1.58E+04±1.95E+04−	1.28E−01±8.34E−02−	6.46E−02±6.03E−02−	**9.08E−14±1.13E−13**
F05	7.05E−13±1.13E−13−	7.39E−13±1.14E−13−	7.31E−13±1.18E−13−	**5.32E−14±5.78E−14**
F06	1.22E+01±1.87E+00−	1.19E+01±1.96E+00−	9.39E+00±2.74E+00=	**8.61E+00±2.20E+00**
F07	9.43E+01±1.35E+01−	8.74E+01±1.04E+01−	4.78E+01±9.88E+00−	**2.24E+01±7.98E+00**
F08	**2.09E+01±5.56E−02=**	**2.09E+01±5.71E−02=**	**2.09E+01±6.21E−02=**	**2.09E+01±5.68E−02**
F09	2.88E+01±1.86E+00−	2.87E+01±1.95E+00−	1.81E+01±2.49E+00−	**1.24E+01±3.17E+00**
F10	1.24E−03±2.53E−03−	2.79E−03±4.56E−03−	4.38E−04±1.37E−03−	**3.64E−05±7.90E−05**
F11	2.18E−13±4.95E−14−	2.42E−13±4.70E−14−	2.25E−13±4.32E−14−	**0.00E+00±0.00E+00**
F12	1.49E+02±3.02E+01−	1.33E+02±2.75E+01−	7.55E+01±1.50E+01−	**3.30E+01±9.89E+00**
F13	1.92E+02±3.15E+01−	1.84E+02±2.05E+01−	9.73E+01±1.79E+01−	**5.31E+01±1.41E+01**
F14	2.60E+02±7.77E+01−	2.70E+02±9.06E+01−	6.55E+01±2.94E+01=	**5.78E+01±2.77E+01**
F15	5.20E+03±4.21E+02=	5.44E+03±4.60E+02=	**2.85E+03±5.46E+02+**	3.65E+03±5.39E+02
F16	2.28E+00±2.89E−01−	2.22E+00±3.37E−01−	**7.43E−01±2.17E−01+**	1.78E+00±2.62E−01
F17	3.48E+01±2.17E+00−	3.34E+01±1.99E+00−	**3.04E+01±1.83E−02=**	**3.04E+01±1.45E−14**
F18	2.84E+02±2.57E+01−	2.64E+02±2.32E+01−	1.34E+02±1.86E+01−	**8.91E+01±1.45E+01**
F19	5.47E+00±6.91E−01−	5.55E+00±8.38E−01−	2.96E+00±4.36E−01−	**1.95E+00±5.80E−01**
F20	1.28E+01±3.62E−01−	1.24E+01±5.42E−01−	1.24E+01±1.63E+00−	**1.09E+01±1.56E+00**
F21	2.40E+02±6.99E+01=	**2.32E+02±5.85E+01=**	2.51E+02±6.17E+01=	2.35E+02±5.31E+01
F22	2.66E+02±8.76E+01−	3.61E+02±1.08E+02−	1.03E+02±3.81E+01=	**1.01E+02±4.09E+01**
F23	5.99E+03±6.38E+02−	5.71E+03±4.26E+02−	3.82E+03±6.42E+02−	**3.38E+03±5.95E+02**
F24	2.77E+02±6.25E+00−	2.74E+02±6.44E+00−	2.26E+02±8.98E+00−	**2.13E+02±6.85E+00**
F25	2.94E+02±4.18E+00−	2.83E+02±7.12E+00−	2.72E+02±7.49E+00−	**2.49E+02±4.54E+00**
F26	2.01E+02±4.79E−01−	**2.00E+02±4.66E−01=**	**2.00E+02±0.00E+00=**	**2.00E+02±0.00E+00**
F27	**4.00E+02±0.00E+00=**	**4.00E+02±0.00E+00=**	5.69E+02±1.83E+02−	4.21E+02±7.95E+01
F28	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00**
+/=/−	0/5/23	0/6/22	2/8/18	– –

Abbreviations: BSABC, our proposed approach; BSABC$_{GNS}$, BSABC with global neighborhood search operator; BSABC$_{OBL}$, BSABC with opposition-based learning strategy; BSABC$_{RND}$, BSABC with random restart method.

algorithms through the Friedman test. It is clear that the first place is obtained by the BSABC, which further confirms the superior performance of the proposed restart method.

In the second set of experiments, the focus is to verify the effectiveness of the method of constructing the multivariate Gaussian distribution.[45] In our approach, we select the best

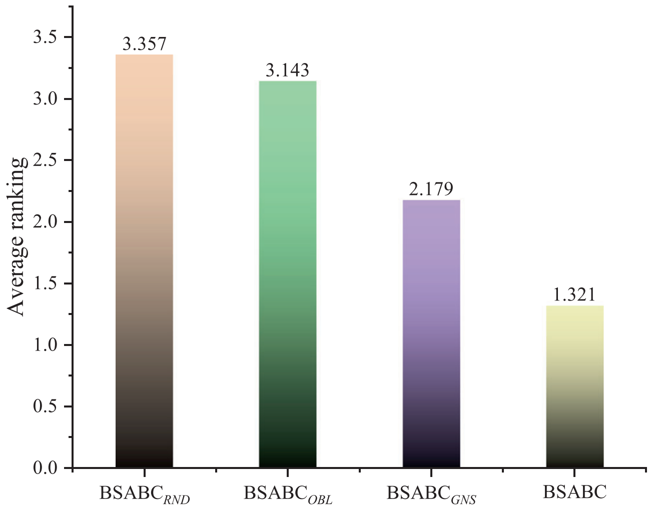

FIGURE 4 Average rankings of the BSABC$_{RND}$, BSABC$_{OBL}$, BSABC$_{GNS}$, and BSABC [Color figure can be viewed at wileyonlinelibrary.com]

$SN/2$ individuals of the population to design the restart method. However, in the above Section 3.2, the randomly selected $SN/2$ individuals are used to establish the eigen coordinate system. So, for these two operations, it is clear that we select different types of individuals to calculate the covariance matrix C, that is, the selection of individuals has preference. It is a natural idea to wonder how about switching the selection methods or there are some other possible selection options. To answer this, we construct the following three comparative algorithms.

- BSABC$_{er}$: the best $SN/2$ individuals are used for the eigen coordinate system, while the randomly selected $SN/2$ individuals are used for the restart method.
- BSABC$_{ee}$: the best $SN/2$ individuals of population are used for both the eigen coordinate system and the restart method.
- BSABC$_{rr}$: the randomly selected $SN/2$ individuals are used for both the eigen coordinate system and the restart method.

For the above three algorithms, there is no difference with the BSABC except for the selection method. Hence, there are four involved algorithms in the performance comparison, and they have the same parameter settings as shown in Section 4.1. The experimental results are presented in Table 3. It can be seen that the BSABC achieves the best results on almost all the benchmark functions in comparison with the three algorithms. To be specific, compared with the BSABC$_{er}$, the BSABC can show better or similar performance on all the 28 functions, which verify that the switched selection methods do not work for the BSABC. In fact, as mentioned before, the selection of the best $SN/2$ individuals for the restart method is to save the search experience, since the individuals with better fitness values have relation with the promising search directions. In contrast, the selection of the random $SN/2$ individuals for the eigen coordinate system is to avoid making the search behavior being too greedy. The comparison results with the BSABC$_{ee}$ and BSABC$_{rr}$ confirm that only using the best $SN/2$

TABLE 3 Comparison results of the modified scout bee phase in the the second set of experiments

Func.	BSABC$_{er}$	BSABC$_{ee}$	BSABC$_{rr}$	BSABC
F01	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00=**	**0.00E+00±0.00E+00**
F02	6.13E+04±3.84E+04−	1.29E+05±8.40E+04−	5.11E−02±2.72E−01=	**1.37E−04±7.39E−04**
F03	6.60E+06±5.73E+06−	**1.15E+05±1.59E+05+**	4.67E+06±3.68E+06−	2.69E+05±3.03E+05
F04	2.13E+00±2.77E+00−	1.02E+01±2.32E+01−	1.44E−13±1.11E−13=	**9.08E−14±1.13E−13**
F05	6.46E−14±5.75E−14=	1.37E−13±4.60E−14−	**4.18E−14±5.59E−14=**	5.32E−14±5.78E−14
F06	9.68E+00±5.31E+00=	1.25E+01±4.01E+00−	9.29E+00±2.41E+00=	**8.61E+00±2.20E+00**
F07	2.26E+01±6.89E+00=	2.35E+01±8.51E+00=	2.35E+01±7.31E+00=	**2.24E+01±7.98E+00**
F08	**2.09E+01±7.28E−02=**	2.10E+01±6.82E−02=	2.10E+01±4.90E−02=	**2.09E+01±5.68E−02**
F09	1.95E+01±4.96E+00−	1.32E+01±5.21E+00=	1.64E+01±4.64E+00−	**1.24E+01±3.17E+00**
F10	4.69E−03±1.19E−02−	1.75E−02±2.56E−02−	4.14E−04±1.34E−03−	**3.64E−05±7.90E−05**
F11	1.17E−13±9.18E−14−	2.46E−14±2.86E−14−	4.75E−14±6.54E−14−	**0.00E+00±0.00E+00**
F12	5.75E+01±1.42E+01−	3.94E+01±1.02E+01−	5.85E+01±1.26E+01−	**3.30E+01±9.89E+00**
F13	6.97E+01±1.26E+01−	5.61E+01±1.83E+01=	7.51E+01±1.78E+01−	**5.31E+01±1.41E+01**
F14	5.31E+01±2.52E+01=	**5.17E+01±2.57E+01=**	7.13E+01±2.93E+01=	5.78E+01±2.77E+01
F15	3.97E+03±7.43E+02=	3.68E+03±6.08E+02=	4.04E+03±7.32E+02−	**3.65E+03±5.39E+02**
F16	1.77E+00±3.56E−01=	**1.69E+00±2.25E−01=**	1.85E+00±2.72E−01=	1.78E+00±2.62E−01
F17	**3.04E+01±1.83E−02=**	**3.04E+01±2.54E−02=**	**3.04E+01±3.79E−02=**	**3.04E+01±1.45E−14**
F18	1.13E+02±2.13E+01−	9.05E+01±1.24E+01=	1.14E+02±1.95E+01−	8.91E+01±1.45E+01
F19	**1.79E+00±5.15E−01=**	1.81E+00±5.49E−01=	2.20E+00±4.50E−01=	1.95E+00±5.80E−01
F20	**1.04E+01±7.64E−01=**	1.09E+01±1.57E+00=	1.12E+01±1.45E+00=	1.09E+01±1.56E+00
F21	3.25E+02±9.45E+01−	3.31E+02±8.87E+01−	2.41E+02±6.00E+01=	**2.35E+02±5.31E+01**
F22	1.05E+02±3.80E+01=	1.05E+02±4.57E+01=	1.14E+02±1.91E+01=	**1.01E+02±4.09E+01**
F23	4.14E+03±7.57E+02−	3.80E+03±7.59E+02−	4.42E+03±4.36E+02−	**3.38E+03±5.95E+02**
F24	2.14E+02±6.58E+00=	2.15E+02±6.35E+00=	2.15E+02±1.22E+01=	**2.13E+02±6.85E+00**
F25	2.59E+02±9.97E+00−	2.52E+02±6.58E+00−	2.54E+02±6.68E+00−	**2.49E+02±4.54E+00**
F26	**2.00E+02±0.00E+00=**	**2.00E+02±0.00E+00=**	**2.00E+02±0.00E+00=**	2.00E+02±0.00E+00
F27	5.42E+02±1.51E+02−	4.43E+02±9.21E+01=	4.74E+02±1.03E+02−	**4.21E+02±7.95E+01**
F28	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00=**	**3.00E+02±0.00E+00**
+/=/−	0/15/13	1/17/10	0/17/11	− −

Abbreviations: BSABC$_{er}$, BSABC with switched selection method; BSABC$_{ee}$, BSABC with the best $SN/2$ individuals; BSABC$_{rr}$, BSABC with the randomly selected $SN/2$ individuals; BSABC, our proposed approach.

individuals or the randomly selected $SN/2$ individuals is not good option for the BSABC. To summarize the overall comparison results, the average rankings of the four involved algorithms based on the Friedman test are illustrated in Figure 5. It is clear that the BSABC ranks on the first place.

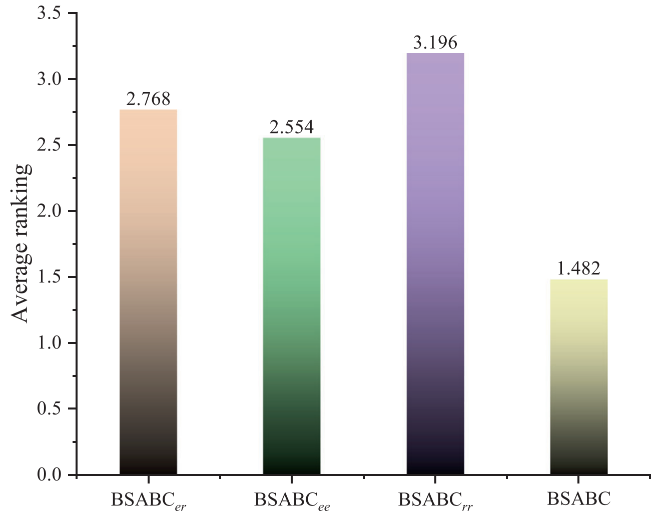

FIGURE 5 Average rankings of the BSABC$_{er}$, BSABC$_{ee}$, BSABC$_{rr}$, and BSABC [Color figure can be viewed at wileyonlinelibrary.com]

4.3 | Sensitivity tests for the control parameters

4.3.1 | Sensitivity test for the parameter *limit*

In the BSABC, the control parameter *limit* has an important effect on the performance of BSABC, which controls the frequency of triggering the scout bee phase. Therefore, in this section, we test the sensitivity of *limit* to the performance of BSABC, and further determine the most suitable parameter value to maximize the performance of BSABC. In most other ABC variants, *limit* is also an important control parameter, and its possible values often cover the following four different cases.

- $limit = 50$.[50,51]
- $limit = 100$ [39,52]
- $limit = 200$ [18,53]
- $limit = SN \cdot D$.[54,55]

So, all the above four values are test for the BSABC. The other parameter settings are kept the same as in Section 4.1. The comparison results are shown in Table 4.

As seen, the BSABC with different *limit* values shows various performance. On the whole, the BSABC with $limit = 50$ performs best by obtaining the best results on 19 out of 28 functions. To be specific, according to the categories of the functions, we can get the following conclusions. For the first five unimodal functions (F01–F05), $limit = 200$ obtains the best results on three functions, while $limit = 50$ and $limit = 100$ have similar performance by performing best on two functions. Regarding the middle 15 multimodal functions (F06–F20), $limit = 50$ gets the best results on 10 functions, which is much better than its other three competitors. Last, for the eight composite functions (F21–F28), $limit = 50$ still shows the best overall performance by achieving the best results on almost all the

TABLE 4 Comparison results of the BSABC with different *limit* values

Func.	*limit* = *SN·D*	*limit* = 200	*limit* = 100	*limit* = 50
F01	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F02	6.80E−07±1.62E−06	**1.54E−07±4.84E−07**	1.06E−06±3.80E−06	1.37E−04±7.39E−04
F03	7.35E+07±5.69E+07	5.63E+07±2.98E+07	1.54E+07±1.14E+07	**2.69E+05±3.03E+05**
F04	2.88E−13±1.68E−13	1.29E−13±1.76E−13	**8.32E−14±1.11E−13**	9.08E−14±1.13E−13
F05	1.10E−13±2.08E−14	**2.66E−14±4.90E−14**	4.18E−14±5.59E−14	5.32E−14±5.78E−14
F06	1.45E+01±5.88E+00	1.37E+01±3.62E+00	1.32E+01±4.45E+00	**8.61E+00±2.20E+00**
F07	8.70E+01±1.15E+01	4.63E+01±9.62E+00	2.92E+01±9.40E+00	**2.24E+01±7.98E+00**
F08	**2.09E+01±5.83E−02**	**2.10E+01±5.04E−02**	**2.09E+01±5.47E−02**	**2.09E+01±5.68E−02**
F09	2.74E+01±1.67E+00	2.66E+01±1.63E+00	2.49E+01±3.76E+00	**1.24E+01±3.17E+00**
F10	3.31E−04±1.37E−03	1.14E−03±2.67E−03	3.89E−04±1.35E−03	**3.64E−05±7.90E−05**
F11	5.68E−15±1.73E−14	**0.00E+00±0.00E+00**	1.89E−15±1.04E−14	**0.00E+00±0.00E+00**
F12	1.10E+02±1.77E+01	7.04E+01±1.22E+01	4.21E+01±8.11E+00	**3.30E+01±9.89E+00**
F13	1.56E+02±2.51E+01	9.55E+01±1.22E+01	6.61E+01±1.36E+01	**5.31E+01±1.41E+01**
F14	4.01E−01±8.30E−01	2.47E−01±3.35E−01	**2.13E−01±3.51E−01**	5.78E+01±2.77E+01
F15	3.53E+03±4.69E+02	**3.45E+03±5.18E+02**	3.78E+03±5.18E+02	3.65E+03±5.39E+02
F16	**1.25E+00±2.05E−01**	1.49E+00±2.78E−01	1.68E+00±2.81E−01	1.78E+00±2.62E−01
F17	**3.04E+01±1.45E−14**	**3.04E+01±1.45E−14**	**3.04E+01±1.83E−02**	**3.04E+01±1.45E−14**
F18	1.56E+02±1.49E+01	9.98E+01±1.41E+01	**8.41E+01±1.81E+01**	8.91E+01±1.45E+01
F19	**9.80E−01±1.90E−01**	1.14E+00±2.33E−01	1.37E+00±4.04E−01	1.95E+00±5.80E−01
F20	1.22E+01±4.01E−01	1.12E+01±9.77E−01	1.15E+01±1.75E+00	**1.09E+01±1.56E+00**
F21	2.49E+02±7.03E+01	2.39E+02±8.71E+01	2.62E+02±8.05E+01	**2.35E+02±5.31E+01**
F22	1.07E+02±4.34E+01	9.25E+01±3.84E+01	**8.66E+01±4.33E+01**	1.01E+02±4.09E+01
F23	4.64E+03±4.95E+02	3.99E+03±4.76E+02	3.67E+03±6.30E+02	**3.38E+03±5.95E+02**
F24	2.76E+02±5.45E+00	2.49E+02±1.18E+01	2.24E+02±8.20E+00	**2.13E+02±6.85E+00**
F25	2.96E+02±5.48E+00	2.86E+02±4.41E+00	2.66E+02±9.89E+00	**2.49E+02±4.54E+00**
F26	**2.00E+02±3.05E−01**	**2.00E+02±0.00E+00**	**2.00E+02±0.00E+00**	**2.00E+02±0.00E+00**
F27	7.01E+02±3.29E+02	7.25E+02±2.92E+02	5.50E+02±1.46E+02	**4.21E+02±7.95E+01**
F28	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**

functions except the function F22. Besides, the results of average rankings achieved by the Friedman test are presented in Figure 6, among which *limit* = 50 obtains the best performance, followed by *limit* = 100, 200 and $SN \cdot D$. Hence, according to the comparison results, the parameter value of *limit* is set to 50 for the BSABC.

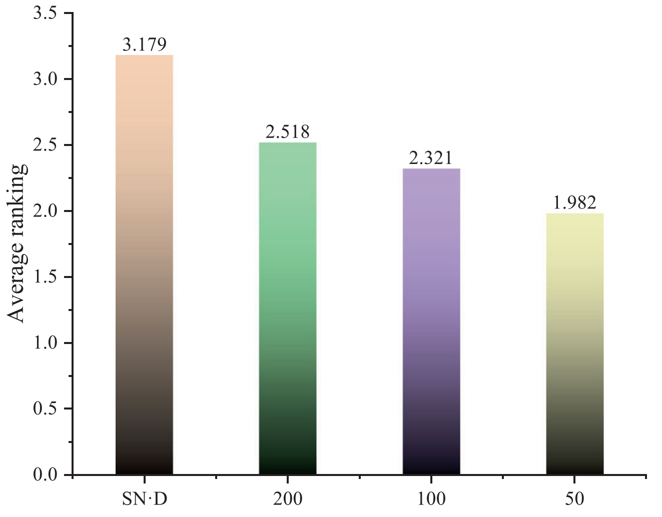

FIGURE 6 Average rankings of the BSABC with different *limit* values [Color figure can be viewed at wileyonlinelibrary.com]

4.3.2 | Sensitivity test for the parameter q

In Section 3.2.2, we introduce a control parameter q for the new solution search equation of the employed bee phase, namely Equation (10). The parameter q is to control the size of the elite group, which determines the number of elite individuals. The available parameter values of q typically fall into the range [0, 1], and its value affects the search behavior performed by the solution search equation. If the value of q is set to be very large or even close to 1, the elite group is potentially equivalent to the entire population, which implies that the elite individual X_e tends to be an ordinary individual rather than a "real" elite individual. As a result, the valuable information from the "real" elite individual may be lost for guiding the search. Conversely, if the value of q is very small, the number of elite individuals is also very limited, and the elite individual X_e has a high probability to equal to the global best individual. In this scenario, the search behavior performed by the solution search equation may be very greedy.

So, it is worth to investigate which of the values is more suitable for the parameter q. Note that q can take an arbitrary value from the range [0, 1]. In the experiment, we select nine representative values to test, that is, $q = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$, or 0.9. The other parameter settings of the BSABC are the same as shown in the Section 4.1. Table 5 presents the final comparison results of these different q values, and the best results are marked in boldface. To simplify the result presentations, in Table 5, only the results of five representative values are given, that is, $q = 0.1, 0.2, 0.3, 0.4$, and 0.5. As seen, on the whole, $q = 0.1$ performs the best by achieving the best results on 18 out of 28 test functions, and $q = 0.2$ obtains the second best performance. It can be observed from the comparison results that the parameter q favors a relative small value, which could offer a good balance between the exploration capability and exploitation capability. Figure 7 shows the average rankings achieved by the Friedman test, in which $q = 0.1$ ranks on the first place.

TABLE 5 Comparison results of the BSABC with different q values

Func.	q = 0.5	q = 0.4	q = 0.3	q = 0.2	q = 0.1
F01	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F02	2.70E+02±4.87E+02	1.20E+03±5.46E+03	**1.79E−06±5.58E−06**	8.85E+02±1.96E+03	1.37E−04±7.39E−04
F03	5.95E+05±7.62E+05	5.52E+05±6.70E+05	5.19E+05±5.23E+05	5.87E+05±1.36E+06	**2.69E+05±3.03E+05**
F04	8.02E−09±4.59E−09	7.65E−09±3.71E−09	1.14E−13±1.30E−13	9.21E−09±5.73E−09	**9.08E−14±1.13E−13**
F05	6.08E−14±5.78E−14	5.70E−14±5.80E−14	6.46E−14±5.75E−14	**4.94E−14±5.75E−14**	5.32E−14±5.78E−14
F06	1.27E+01±4.03E+00	1.18E+01±2.38E+00	9.62E+00±3.91E+00	1.17E+01±1.25E+00	**8.61E+00±2.20E+00**
F07	2.82E+01±9.32E+00	2.63E+01±8.66E+00	**2.12E+01±6.99E+00**	2.28E+01±8.89E+00	2.24E+01±7.98E+00
F08	2.10E+01±5.72E−02	2.10E+01±4.98E−02	2.10E+01±5.56E−02	**2.09E+01±8.58E−02**	**2.09E+01±5.68E−02**
F09	1.35E+01±4.16E+00	1.34E+01±4.70E+00	1.25E+01±3.23E+00	1.36E+01±4.86E+00	**1.24E+01±3.17E+00**
F10	3.24E−03±4.72E−03	4.51E−03±7.51E−03	**3.07E−05±3.78E−05**	2.79E−03±4.87E−03	3.64E−05±7.90E−05
F11	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F12	4.14E+01±8.98E+00	3.64E+01±8.41E+00	3.90E+01±8.98E+00	3.73E+01±7.15E+00	**3.30E+01±9.89E+00**
F13	5.17E+01±1.38E+01	5.14E+01±1.49E+01	5.71E+01±1.74E+01	**5.02E+01±1.34E+01**	5.31E+01±1.41E+01
F14	8.02E−01±1.58E+00	5.58E−02±2.09E−01	1.20E+02±3.49E+01	**1.39E−02±1.99E−02**	5.78E+01±2.77E+01
F15	4.04E+03±6.28E+02	3.91E+03±7.65E+02	3.80E+03±5.79E+02	3.66E+03±5.27E+02	**3.65E+03±5.39E+02**
F16	2.02E+00±2.51E−01	2.02E+00±2.43E−01	1.87E+00±2.88E−01	2.02E+00±2.33E−01	**1.78E+00±2.62E−01**
F17	**3.04E+01±1.45E−14**	**3.04E+01±1.45E−14**	**3.04E+01±1.45E−14**	**3.04E+01±1.45E−14**	**3.04E+01±1.45E−14**
F18	9.30E+01±1.35E+01	9.95E+01±1.40E+01	9.61E+01±1.49E+01	9.35E+01±1.38E+01	**8.91E+01±1.45E+01**
F19	1.12E+00±4.77E−01	1.21E+00±5.59E−01	2.20E+00±3.01E−01	**9.64E−01±4.99E−01**	1.95E+00±5.80E−01
F20	1.15E+01±1.92E+00	**1.04E+01±1.24E+00**	1.10E+01±1.46E+00	1.06E+01±1.83E+00	1.09E+01±1.56E+00
F21	2.63E+02±6.94E+01	2.49E+02±6.13E+01	**2.24E+02±5.83E+01**	2.51E+02±6.10E+01	2.35E+02±5.31E+01

TABLE 5 (Continued)

Func.	q = 0.5	q = 0.4	q = 0.3	q = 0.2	q = 0.1
F22	1.02E+02±3.64E+01	1.02E+02±2.54E+01	1.03E+02±2.63E+01	1.06E+02±3.28E+01	**1.01E+02±4.09E+01**
F23	3.89E+03±5.75E+02	3.86E+03±6.36E+02	3.90E+03±7.06E+02	3.62E+03±6.23E+02	**3.38E+03±5.95E+02**
F24	2.15E+02±7.87E+00	2.16E+02±6.93E+00	**2.13E+02±5.47E+00**	2.16E+02±8.81E+00	**2.13E+02±6.85E+00**
F25	2.51E+02±5.48E+00	2.52E+02±7.29E+00	2.51E+02±6.98E+00	2.51E+02±5.58E+00	**2.49E+02±4.54E+00**
F26	**2.00E+02±0.00E+00**	**2.00E+02±0.00E+00**	**2.00E+02±0.00E+00**	**2.00E+02±0.00E+00**	**2.00E+02±0.00E+00**
F27	4.73E+02±1.19E+02	4.17E+02±7.17E+01	4.30E+02±7.81E+01	**4.14E+02±5.88E+01**	4.21E+02±7.95E+01
F28	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**	**3.00E+02±0.00E+00**

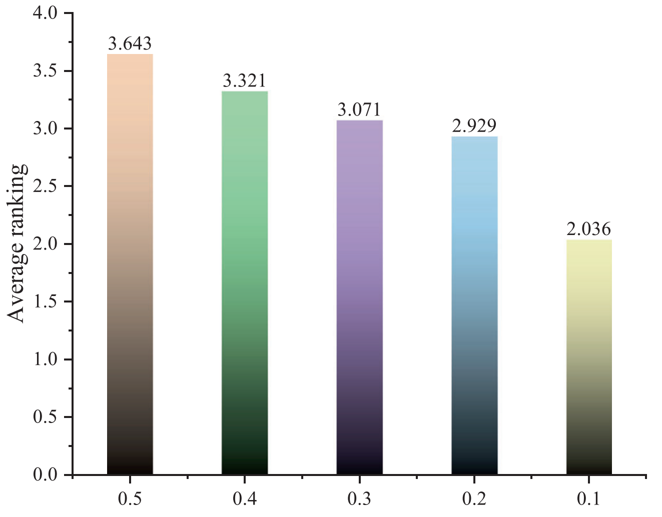

FIGURE 7 Average rankings of the BSABC with different q values [Color figure can be viewed at wileyonlinelibrary.com]

4.4 | Performance comparison with other advanced ABC variants

To investigate whether the performance of BSABC has advantages in comparison with other advanced ABC variants, in this section, four well-established ABC variants are included in the performance comparison with the BSABC. These four comparative ABC variants own different features, and they are listed as follows.

- AELABC[44]: ABC with adaptive encoding learning.
- ECABC[27]: ABC based on elite group guidance and combined breadth-depth search strategy.
- GBABC[26]: ABC with the Gaussian bare-bones search equation.
- NSABC[39]: ABC with a new neighborhood selection mechanism.

The AELABC[44] has a similar motivation with the BSABC, which also combines two coordinate systems and is introduced in the above Section 3.2.1. So, it is necessary to compare the BSABC with it. In the ECABC,[27] the main contribution is the modified solution search equation in which the concept of elite group is introduced. The comparison with the ECABC is to check whether the new solution search equation of the BSABC has advantages. As for the GBABC and NSABC, both of them have modified the scout bee phase. In the GBABC,[26] the opposition-based learning strategy is used in the scout bee phase. However, in the NSABC,[39] the global neighborhood search operator is employed. Hence, the main purpose of comparing with them is to verify the effectiveness of the modified scout bee phase of the BSABC. Beside, in the GBABC and NSABC, the solution search equations are also modified by using elite information. To have a fair comparison, all the four comparative algorithms share the same population size with the BSABC, that is, $SN = 50$. As for the other specific parameters of the four algorithms, they are set to the same values as in the original papers, and Table 6 shows the details of the parameter settings.

In this section, the test dimension sizes for the CEC2013 test suite include two cases: $D = 30$ and 50. Corresponding, the maximal number of fitness function evaluations ($MaxFEs$) is $MaxFEs = 10000 \cdot D$, which is kept the same with Section 4.1. As for the case of $D = 30$,

TABLE 6 Detailed parameter settings of four comparative algorithms

Algorithms	Parameter settings
AELABC	$limit = 100$, $LP = 50$
ECABC	$limit = SN \cdot D$, $p = 0.1$
GBABC	$limit = 100$, $CR = 0.3$
NSABC	$limit = 100$, $C = 1.5$, $k = 10$

Abbreviations: AELABC, ABC with adaptive encoding learning; ECABC, ABC based on elite group; GBABC, Gaussian barebones ABC; NSABC, ABC with neighborhood selection.

Table 7 shows the final results of the five involved algorithms, namely the AELABC,[44] ECABC,[27] GBABC,[26] NSABC,[39] and BSABC. According to the categories of the functions, we can make the following analyses for the comparison results. First, for the five unimodal functions (F01–F05), it is clear that the BSABC obtains the best results on all the five functions. In particular, the results on the first four functions are much better than its four competitors. Second, for the middle 15 multimodal functions, the BSABC performs the best on nine functions, while the number for the best performed competitor, namely the AELABC, is only five. Last, for the eight composite functions, both the BSABC and AELABC can get the best results on four functions. So, on the whole, the BSABC can surpass its four competitors by achieving the best results on 18 out 28 functions. The main reason behind the good performance of the BSABC includes two aspect. On the one hand, the benefits from that the BSABC owns two coordinate systems. As a result, the BSABC can perform versatile but complementary search behaviors. On the other hand, the modified scout bee phase can effectively avoid the problem of population stagnation, without sacrificing the search experience. What is more, the two new solution search equations based on elite information also play an important role. Figure 8 illustrates the convergence curves of the involved five algorithms on six representative functions, which shows that the BSABC can also provide fast convergence speed at the same time.

As for the case of $D = 50$, the comparison results of the above five algorithms are presented in Table 8. Note that the problem difficulty of the functions would increase with the development of the dimension size. Hence, the comparison results in this case are helpful to check the robustness of the BSABC. From Table 8, it can be seen that the overall comparison results in the case of $D = 50$ are very similar with those in the case of $D = 30$. So, we can draw a conclusion that the BSABC continually shows the best performance among the five algorithms. But it terms of the performance on different categories of the functions, it is impressive that the BSABC can display much better performance on the composite functions, since the number of functions on which the BSABC obtains the best results increases from four in the case of $D = 30$ to six in the case of $D = 50$. To sum up the comparison results in the two cases, Figure 9 shows the average rankings of all the five involved algorithms, based on the Friedman test. As seen, in these two cases, the BSABC ranks on the first place, which verifies the high effectiveness of the BSABC. For the competitors, AELABC is the second best one in the case of $D = 30$, while it is the NSABC for the case of $D = 50$.

4.5 | Performance comparison with other EAs

To further verify the performance of the BSABC, in this section, we carry out performance comparison between the BSABC and three other paradigms of EAs on the CEC2013 test suite

TABLE 7 Comparison results with other advanced ABC variants for $D = 30$

Func.	AELABC	ECABC	GBABC	NSABC	BSABC
F01	6.29E−13±1.14E−13−	1.59E−13±1.06E−13−	2.57E−13±7.88E−14−	7.57E−15±4.14E−14=	**0.00E+00±0.00E+00**
F02	7.47E+04±4.30E+04−	9.01E+06±2.78E+06−	2.41E+07±4.45E+06−	7.05E+06±2.57E+06−	**1.37E−04±7.39E−04**
F03	1.76E+08±1.21E+08−	1.60E+08±1.08E+08−	2.55E+08±1.79E+08−	7.29E+07±4.77E+07−	**2.69E+05±3.03E+05**
F04	2.52E+00±4.97E+00−	8.23E+04±9.86E+03−	4.92E+04±8.27E+03−	3.16E+04±5.51E+03−	**9.08E−14±1.13E−13**
F05	6.82E−13±9.87E−14−	1.82E−13±6.37E−14−1	2.23E−13±6.98E−14−	.10E−13±2.08E−14−	**5.32E−14±5.78E−14**
F06	1.23E+01±4.89E+00−	2.60E+01±2.08E+01−	1.69E+01±4.33E+00−	2.16E+01±8.10E+00−	**8.61E+00±2.20E+00**
F07	9.17E+01±1.06E+01−	9.29E+01±1.31E+01−	4.26E+01±1.21E+01−	5.59E+01±1.19E+01−	**2.24E+01±7.98E+00**
F08	**2.09E+01±6.48E−02=**	2.10E+01±5.63E−02=	**2.09E+01±5.68E−02=**	2.10E+01±6.26E−02=	**2.09E+01±5.68E−02**
F09	2.84E+01±1.65E+00−	2.93E+01±1.94E+00−	2.97E+01±1.10E+00−	2.91E+01±1.29E+00−	**1.24E+01±3.17E+00**
F10	**8.92E−06±2.15E−05+**	2.27E−01±7.44E−02−	4.48E−01±6.29E−01−	1.12E+00±2.37E−01−	3.64E−05±7.90E−05
F11	2.54E−13±4.65E−14−	1.70E−14±2.65E−14−	7.40E−14±2.67E−14−	3.32E−02±1.82E−01−	**0.00E+00±0.00E+00**
F12	2.23E+02±3.82E+01−	1.14E+02±1.94E+01−	1.85E+02±2.06E+01−	6.42E+01±1.22E+01−	**3.30E+01±9.89E+00**
F13	2.35E+02±2.83E+01−	1.71E+02±3.11E+01−	1.90E+02±1.25E+01−	9.11E+01±1.83E+01−	**5.31E+01±1.41E+01**
F14	4.63E+00±2.73E+00+	2.96E+00±3.10E+00+	1.50E+02±5.04E+01−	**2.67E+00±2.50E+00+**	5.78E+01±2.77E+01
F15	**3.56E+03±3.61E+02=**	3.68E+03±3.46E+02=	4.77E+03±2.89E+02−	4.19E+03±4.83E+02−	3.65E+03±5.39E+02
F16	**1.04E+00±1.40E−01+**	1.32E+00±2.57E−01+	1.88E+00±2.63E−01=	2.02E+00±2.54E−01−	1.78E+00±2.62E−01
F17	**3.00E+01±3.17E+00+**	3.05E+01±5.72E−02−	3.11E+01±2.31E−01−	3.05E+01±7.28E−02−	3.04E+01±1.45E−14
F18	2.11E+02±2.49E+01−	1.42E+02±2.00E+01−	1.68E+02±1.73E+01−	9.95E+01±1.32E+01−	**8.91E+01±1.45E+01**
F19	1.09E+00±2.97E−01+	5.25E−01±2.16E−01+	9.19E−01±4.69E−01+	**2.63E−01±1.65E−01+**	1.95E+00±5.80E−01
F20	1.43E+01±3.04E−01−	1.16E+01±4.01E−01−	1.16E+01±2.85E−01−	1.18E+01±2.79E−01−	**1.09E+01±1.56E+00**
F21	**1.75E+02±3.83E+01+**	3.05E+02±4.92E+01−	2.70E+02±4.65E+01−	3.47E+02±5.07E+01−	2.35E+02±5.31E+01

TABLE 7 (Continued)

Func.	AELABC	ECABC	GBABC	NSABC	BSABC
F22	1.02E+02±2.84E+01−	**9.16E+01±3.13E+01**+	1.21E+02±6.10E+01−	1.07E+02±6.60E+01−	1.01E+02±4.09E+01
F23	4.61E+03±2.99E+02−	4.76E+03±3.69E+02−	5.46E+03±3.76E+02−	4.63E+03±5.06E+02−	**3.38E+03±5.95E+02**
F24	2.84E+02±4.89E+00−	2.76E+02±6.14E+00−	2.37E+02±1.44E+01−	2.71E+02±1.10E+01−	**2.13E+02±6.85E+00**
F25	3.06E+02±5.68E+00−	3.06E+02±5.15E+00−	2.74E+02±1.04E+01−	2.77E+02±3.90E+00−	**2.49E+02±4.54E+00**
F26	**2.00E+02±0.00E+00**=	2.01E+02±5.04E−01−	**2.00E+02±3.05E−01**=	**2.00E+02±4.50E−01**=	**2.00E+02±0.00E+00**
F27	**4.00E+02±1.83E−01**=	7.48E+02±3.56E+02−	**4.00E+02±0.00E+00**=	8.14E+02±3.16E+02−	4.21E+02±7.95E+01
F28	**2.42E+02±7.82E+01**+	3.00E+02±1.64E+00=	3.00E+02±0.00E+00=	3.00E+02±0.00E+00=	3.00E+02±0.00E+00
+/=/−	7/4/17	4/3/21	1/5/22	2/4/22	− −

Abbreviations: AELABC, ABC with adaptive encoding learning; BSABC, our proposed approach; ECABC, ABC based on elite group; GBABC, Gaussian bare-bones ABC; NSABC, ABC with neighborhood selection.

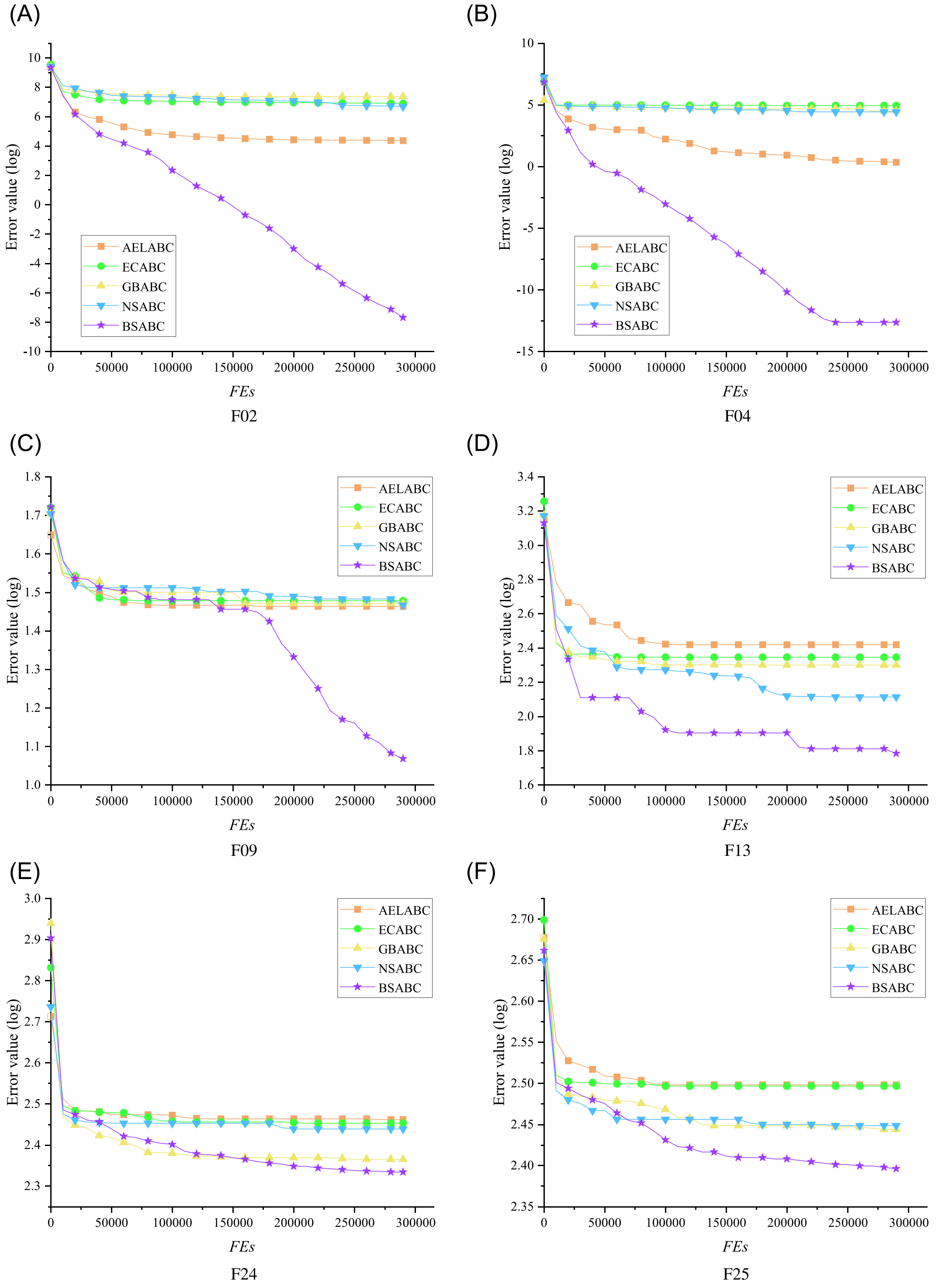

FIGURE 8 Convergence curves of the five involved algorithms on six representative functions for $D = 30$ (A) F02; (B) F04; (C) F09; (D) F13; (E) F24; (F) F25 [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 8 Comparison results with other advanced ABC variants for $D = 50$

Func.	AELABC	ECABC	GBABC	NSABC	BSABC
F01	1.41E−12±2.10E−13−	3.49E−13±1.16E−13−	4.63E−13±7.27E−14−	1.89E−13±8.60E−14−	**1.06E−13±1.15E−13**
F02	8.73E+05±3.57E+05−	1.45E+07±4.00E+06−	5.44E+07±6.36E+06−	1.54E+07±5.35E+06−	**9.80E+04±7.99E+04**
F03	5.98E+08±3.79E+08−	1.80E+09±1.34E+09−	5.86E+09±3.05E+09−	1.51E+09±1.06E+09−	**5.89E+06±6.43E+06**
F04	3.03E+02±2.49E+02−	1.48E+05±1.47E+04−	1.03E+05±1.16E+04−	6.85E+04±9.58E+03−	**1.43E+02±1.72E+02**
F05	1.67E−12±2.32E−13−	4.25E−13±5.13E−14−	4.40E−13±7.14E−14−	**1.14E−13±5.14E−29**+	1.29E−13±3.91E−14
F06	**4.15E+01±5.16E+00**+	4.55E+01±1.34E+00−	4.38E+01±2.78E+00−	4.21E+01±8.73E−01+	4.34E+01±7.23E−15
F07	1.28E+02±1.12E+01−	1.25E+02±1.27E+01−	7.58E+01±1.67E+01−	1.01E+02±1.36E+01−	**6.22E+01±9.22E+00**
F08	**2.11E+01±5.48E−02**=	**2.11E+01±5.04E−02**=	2.12E+01±5.07E−02−	**2.11E+01±5.47E−02**=	**2.11E+01±5.48E−02**
F09	5.65E+01±1.81E+00−	5.73E+01±1.96E+00−	5.83E+01±2.03E+00−	5.82E+01±2.39E+00−	**2.38E+01±3.33E+00**
F10	1.13E−02±1.11E−02−	3.99E−01±2.43E−01−	4.59E+00±4.90E+00−	1.73E+00±4.54E−01−	**1.87E−04±1.01E−03**
F11	7.28E−13±2.29E−13−	5.68E−14±2.57E−29=	1.39E−13±2.87E−14−	**4.73E−14±2.15E−14**=	5.11E−14±1.73E−14
F12	5.06E+02±4.56E+01−	3.57E+02±4.42E+01−	3.28E+02±1.20E+01−	1.64E+02±3.22E+01−	**1.11E+02±2.20E+01**
F13	5.07E+02±3.86E+01−	4.60E+02±5.04E+01−	3.27E+02±2.42E+01−	2.28E+02±3.07E+01−	**1.78E+02±3.05E+01**
F14	1.32E+01±3.49E+00+	**4.66E+00±2.52E+00**+	2.99E+02±1.42E+02=	1.09E+01±2.18E+01+	2.67E+02±1.11E+02
F15	**7.36E+03±4.96E+02**+	7.73E+03±6.34E+02=	9.88E+03±5.07E+02−	8.57E+03±8.37E+02=	8.03E+03±1.31E+03
F16	**1.29E+00±1.96E−01**+	1.79E+00±2.11E−01+	2.57E+00±2.78E−01−	2.75E+00±2.53E−01−	2.31E+00±3.72E−01
F17	5.12E+01±8.94E−02−	5.09E+01±1.32E−01−	5.27E+01±4.25E−01−	5.09E+01±9.71E−02−	**5.08E+01±1.83E−02**
F18	4.36E+02±4.57E+01−	3.55E+02±4.41E+01−	3.45E+02±4.00E+01−	1.86E+02±1.87E+01=	**1.75E+02±3.21E+01**
F19	1.66E+00±3.11E−01+	8.92E−01±1.96E−01+	1.85E+00±4.90E−01+	**6.26E−01±2.80E−01**+	3.92E+00±7.92E−01
F20	2.40E+01±6.34E−01−	2.08E+01±3.73E−01−	2.09E+01±2.68E−01−	2.10E+01±3.54E−01−	**1.91E+01±9.84E−01**
F21	1.98E+02±1.53E+01=	**3.04E+02±3.22E+01**+	2.13E+02±3.46E+01=	3.13E+02±4.34E+01+	3.86E+02±3.50E+02

(Continues)

TABLE 8 (Continued)

Func.	AELABC	ECABC	GBABC	NSABC	BSABC
F22	4.09E+01±1.07E+01−	2.90E+01±3.01E+01−	7.77E+01±7.51E+01−	6.31E+01±7.05E+01−	**2.87E+01±2.25E+01**
F23	9.32E+03±7.05E+02−	9.51E+03±9.15E+02−	1.11E+04±5.19E+02−	9.88E+03±1.04E+03−	**7.66E+03±9.64E+02**
F24	3.67E+02±9.77E+00−	3.57E+02±8.65E+00−	2.92E+02±2.37E+01−	3.51E+02±5.33E+00−	**2.53E+02±9.47E+00**
F25	4.16E+02±7.35E+00−	4.13E+02±8.04E+00−	3.47E+02±1.69E+01−	3.51E+02±6.15E+00−	**3.03E+02±9.78E+00**
F26	**2.00E+02±0.00E+00**=	2.02E+02±4.98E−01−	2.01E+02±4.07E−01−	2.01E+02±4.66E−01−	**2.00E+02±0.00E+00**
F27	1.11E+03±7.74E+02=	1.75E+03±4.62E+02−	**6.08E+02±3.98E+02**+	1.72E+03±3.56E+02−	8.93E+02±1.76E+02
F28	**4.00E+02±0.00E+00**=	**4.00E+02±0.00E+00**=	6.10E+02±7.98E+02=	**4.00E+02±0.00E+00**=	**4.00E+02±0.00E+00**
+/=/−	5/5/18	4/4/20	2/3/23	5/5/18	– –

Abbreviations: AELABC, ABC with adaptive encoding learning; BSABC, our proposed approach; ECABC, ABC based on elite group; GBABC, Gaussian bare-bones ABC; NSABC, ABC with neighborhood selection.

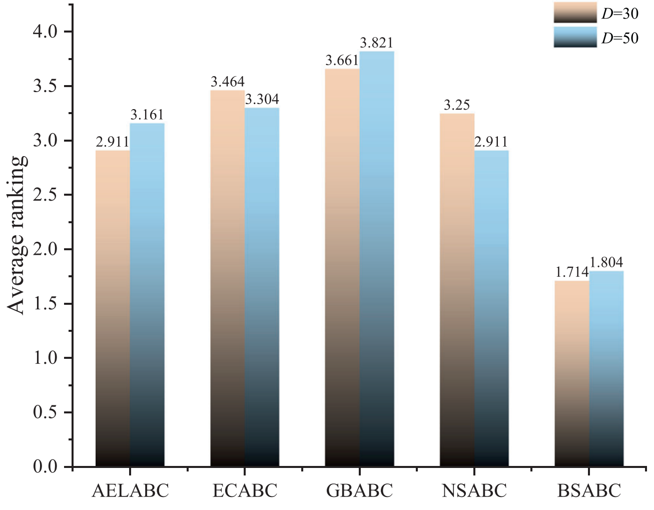

FIGURE 9 Average rankings of the AELABC, ECABC, GBABC, NSABC and BSABC for $D = 30$ and 50 [Color figure can be viewed at wileyonlinelibrary.com]

with $D = 30$. The three competitors cover the categories of evolution strategy (ES), DE, and PSO. The first competitor is a very classic ES variant and has promising performance. The latter two ones were proposed recently, which also display very competitive performance. They are listed as follows:

- CMA-ES[56]: Covariance matrix adaptation evolution strategy.
- CPI-DE/rand/1/bin[57]: A cumulative population distribution information based differential evolution.
- FMPSO[58]: A fitness-based multi-role particle swarm optimizer.

The reasons of selecting the above three competitors include two aspects. (1) Both the CMA-ES and CPI-DE/rand/1/bin use the similar techniques with the BSABC. So, it is necessary to investigate whether the BSABC has better performance than the CMA-ES and CPI-DE/rand/1/bin. As for the CMA-ES,[56] it generates offspring by sampling from a multivariate Gaussian distribution, which comprises the three main elements: mean vector of the search distribution, covariance matrix, and step-size. For the CPI-DE/rand/1/bin,[57] the crossover operator is implemented in both the original coordinate system and eigen coordinate system simultaneously, and two trial vectors are generated accordingly. (2) As one of the most popular paradigms of EAs, PSO has been widely used due to its attracting performance. So, it is interesting to investigate how about the performance comparison between the BSABC and a well-established PSO variant. Regarding the FMPSO,[58] three roles are defined for the particles based on their fitness values, and extensive experiments have verified that the FMPSO has very promising performance. So, it is worthy to select the above three algorithms as the competitors for the BSABC.

All the specific parameters of the algorithms are kept the same as the original references. For the algorithm stopping condition, *MaxFEs* is set to $10000 \cdot D$ for all the algorithms. Table 9 shows the comparison results of the involved four algorithms, in which the

TABLE 9 Comparison results with three other EAs for $D = 30$

Func.	CMA-ES	CPI-DE/rand/1/bin	FMPSO	BSABC
F01	4.24E−13±2.05E−13	2.27E−13±0.00E+00	**0.00E+00±0.00E+00**	**0.00E+00±0.00E+00**
F02	**3.79E−13±1.38E−13**	8.43E−03±3.52E−02	6.49E+06±5.73E+06	1.37E−04±7.39E−04
F03	**3.77E+03±2.06E+04**	6.73E+04±4.79E+04	2.64E+07±2.38E+07	2.69E+05±3.03E+05
F04	4.70E−13±2.06E−13	2.36E−08±1.49E−08	1.24E+04±5.27E+03	**9.08E−14±1.13E−13**
F05	1.14E−12±2.90E−12	2.30E−09±8.21E−10	**0.00E+00±0.00E+00**	5.32E−14±5.78E−14
F06	4.40E+00±1.00E+01	**1.51E−02±2.29E−02**	2.05E+01±1.26E+01	8.61E+00±2.20E+00
F07	1.25E+01±7.72E+00	**8.91E+00±3.84E+00**	3.65E+01±6.83E+00	2.24E+01±7.98E+00
F08	2.14E+01±1.88E−01	**2.09E+01±5.26E−02**	**2.09E+01±3.49E−02**	**2.09E+01±5.68E−02**
F09	4.39E+01±6.57E+00	3.91E+01±1.29E+00	2.15E+01±3.36E+00	**1.24E+01±3.17E+00**
F10	1.73E−02±1.47E−02	2.48E−04±1.35E−03	1.02E−01±4.02E−02	**3.64E−05±7.90E−05**
F11	1.30E+02±3.12E+02	1.21E+02±8.82E+00	5.74E+01±1.14E+01	**0.00E+00±0.00E+00**
F12	8.19E+02±9.47E+02	1.88E+02±1.28E+01	6.54E+01±1.90E+01	**3.30E+01±9.89E+00**
F13	1.42E+03±1.38E+03	1.89E+02±1.08E+01	1.45E+02±2.65E+01	**5.31E+01±1.41E+01**
F14	5.17E+03±8.18E+02	4.70E+03±2.12E+02	1.90E+03±4.45E+02	**5.78E+01±2.77E+01**
F15	5.32E+03±8.13E+02	7.28E+03±2.69E+02	**3.42E+03±4.10E+02**	3.65E+03±5.39E+02
F16	8.28E−02±4.80E−02	2.39E+00±3.45E−01	**1.35E+00±4.33E−01**	1.78E+00±2.62E−01
F17	3.62E+03±1.23E+03	1.90E+02±1.05E+01	1.02E+02±2.83E+01	**3.04E+01±1.45E−14**
F18	4.23E+03±6.08E+02	2.27E+02±8.90E+00	**8.02E+01±1.25E+01**	8.91E+01±1.45E+01
F19	3.53E+00±7.83E−01	1.57E+01±1.24E+00	5.10E+00±1.18E+00	**1.95E+00±5.80E−01**
F20	1.25E+01±9.09E−01	1.26E+01±2.95E−01	1.27E+01±4.78E−01	**1.09E+01±1.56E+00**
F21	3.10E+02±9.87E+01	2.85E+02±1.05E+02	2.40E+02±4.80E+01	**2.35E+02±5.31E+01**
F22	6.77E+03±1.12E+03	5.11E+03±3.10E+02	1.31E+02±3.65E+01	**1.01E+02±4.09E+01**
F23	6.75E+03±5.99E+02	7.31E+03±3.80E+02	3.98E+03±6.79E+02	**3.38E+03±5.95E+02**
F24	5.85E+02±5.83E+02	**2.09E+02±2.66E+00**	2.87E+02±7.74E+00	2.13E+02±6.85E+00
F25	4.02E+02±2.15E+02	2.72E+02±2.59E+01	2.98E+02±7.12E+00	**2.49E+02±4.54E+00**
F26	3.74E+02±3.31E+02	**2.00E+02±1.76E−09**	3.45E+02±2.97E+01	**2.00E+02±0.00E+00**
F27	5.99E+02±2.20E+02	9.75E+02±2.72E+02	6.04E+02±2.37E+02	**4.21E+02±7.95E+01**
F28	1.19E+03±2.48E+03	**3.00E+02±1.45E−06**	**3.00E+02±8.03E−11**	**3.00E+02±0.00E+00**

Abbreviations: BSABC, our proposed approach; CMA-ES, covariance matrix adaptation evolution strategy; CPI-DE/rand/1/bin, a cumulative population distribution information based DE; FMPSO, a fitness-based multi-role PSO.

results of FMPSO are derived from the original reference.[58] Similarly, for the sake of clarity, the best results are marked in boldface. As shown in Table 9, we can observe that the BSABC performs the best overall performance by obtaining the best results on 19 out of 28 functions. In details, for the unimodal functions (F01–F05), the BSABC, CMA-ES and FMPSO display similar performance, because all of them achieve the best results on two out

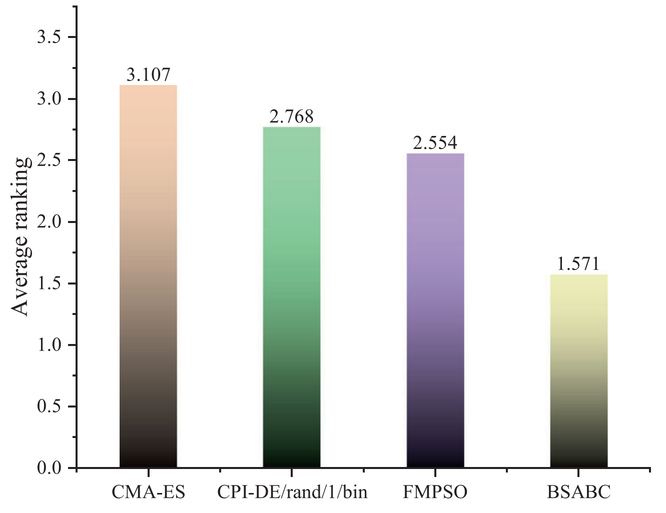

FIGURE 10 Average rankings of the CMA-ES, CPI-DE/rand/1/bin, FMPSO, and BSABC [Color figure can be viewed at wileyonlinelibrary.com]

of five functions. For the multimodal functions (F6–F20), the BSABC manifests the most favorable performance by providing the best results on 10 out of 15 functions, followed by the FMPSO, which obtains the best results on four functions. As for the last type of functions, namely the composition functions (F21–F28), the BSABC yields the best results on almost all the functions except the function F24. To simplify the overall performance comparison, according to the results of the Friedman test, Figure 10 shows the average rankings for the above involved algorithms. As seen, the BSABC ranks on the first place, followed by the FMPSO, CPI-DE/rand/1/bin, and CMA-ES.

4.6 | Performance comparison on a real-world optimization problem

To evaluate the performance of the BSABC on the real-world optimization problems, in this section, we use the BSABC to address a real-world optimization problem, that is, the Parameter Estimation of Frequency Modulated Sound Wave (PEFMSW).[59] This problem is a complex engineering problem that often arises in many modern music systems, which has the goal to minimize the sum of square errors between the estimated sound and target sound. The mathematical expression of this problem is given in the following Equation (14).

$$f(\overrightarrow{X}) = \sum_{t=0}^{100} (S(t) - S_0(t))^2 \tag{14}$$

where t is the index of sound samples, and the optimal value of $f(\overrightarrow{X})$ is 0. $S(t)$ denotes the estimated sound, and $S_0(t)$ represents target sound. The two sounds are determined by the six real parameters $\overrightarrow{X} = \{\alpha_1, \omega_1, \alpha_2, \omega_2, \alpha_3, \omega_3\}$, and the specific formulas of $S(t)$ and $S_0(t)$ are listed in the Eqs. (15) and (16), respectively.

TABLE 10 Performance comparison on the PEFMSW problem

Algorithms	Mean	SD
AELABC	2.59E+00	3.93E+00
ECABC	8.42E+00	4.33E+00
GBABC	1.22E+00	3.01E+00
NSABC	3.59E+00	4.77E+00
BSABC	**2.81E−01**	**1.54E+00**

Abbreviations: AELABC, ABC with adaptive encoding learning; BSABC, our proposed approach; ECABC, ABC based on elite group; GBABC, Gaussian bare-bones ABC; NSABC, ABC with neighborhood selection.

$$S(t) = \alpha_1 \sin(\omega_1 t\theta + \alpha_2 \sin(\omega_2 t\theta + \alpha_3 \sin(\omega_3 t\theta))) \tag{15}$$

$$S_0(t) = 1.0 \sin(0.5t\theta + 1.5 \sin(4.8t\theta + 2.0 \sin(4.9t\theta))) \tag{16}$$

where $\theta = 2\pi/100$, and α_i and ω_i ($i \in 1, 2, 3$) both fall into the range of [−6.4, 6.35]. It is clear that the PEFMSW problem is a six dimensional and highly complex multimodal optimization problem, which has strong epistasis (interrelation among the variables).

In the experiment, the AELABC,[44] ECABC,[27] GBABC,[26] and NSABC,[39] as the competitors, are used to solve this problem as well. To have a fair performance comparison, the algorithm stopping condition *MaxFEs* is set to 200,000, and the results are averaged over 30 independent runs. Table 10 shows the mean and standard deviation values of the five algorithms, in which the best results are marked in boldface. It is clear that the result of the BSABC is better than its other four competitors. Moreover, the result accuracy is much higher. The main reason is that there exist strong correlations among the decision variables of the problem, which makes the problem hard to solve. So, with the assistance from the eigen coordinate system, the BSABC shows satisfactory performance. From this experiment, we can conclude that the BSABC not only does well on the benchmark functions, but also on the real-world optimization problem.

5 | CONCLUSIONS

In this study, we propose an improved ABC variant, called the BSABC, based on the bi-coordinate systems for global numerical optimization. The motivation behind the BSABC is different from the previous most ABC variants, which mainly focuses on how to solve the problem that the solution search equation is not rotationally invariant. Hence, in the BSABC, the eigen coordinate system is established to make the search adapt to the fitness landscape, which is beneficial to alleviate the difficulty of solving complex problems, especially for the problems with high variable correlation. Moreover, the original coordinate system is retained for encouraging search diversity. As another contribution of this study, to save the search experience, the scout bee phase is modified based on the multivariate Gaussian distribution.

To verify the effectiveness of the BSABC, extensive experiments are conducted on the widely used CEC2013 test suite and one real-world optimization problem, which cover five different sets as follows. The first set of experiments is to verify effectiveness of the proposed modifications, while the second one aims to check the parameter sensitivity of the new control parameters of the BSABC. In the third set of experiments, four related advanced ABC variants, namely the AELABC, ECABC, GBABC, and NSABC, are included in the performance comparison with the BSABC. The comparison results show that the BSABC can achieve much better performance on the rotated unimodal functions and mutlimodal functions. Meanwhile, the performance of the BSABC is consistently promising when the dimension size of the functions is increased. Hence, we can conclude that the introduction of the eigen coordinate system is indeed helpful to boost the performance of ABC. In addition, the fourth set of experiments further verify the performance of BSABC by the comparison with three other EAs, that is, CMA-ES, CPI-DE/rand/1/bin, and FMPSO. In the last set of experiments, a application in the PEFMSW problem, it has shown that BSABC not only does well CEC test suite, but also on the real-world optimization problem.

However, our work still has a shortcoming, since the covariance matrix used to establish the eigen coordinate system only reflects the correlation between any two decision variables, while not for multiple decision variables. So, in the future, it is worth to further research how to capture the correlation among multiple decision variables for a more reliable eigen coordinate system. In our opinion, there may exist two potential ways: (1) the Bayesian networks can be used to identify the relationship among multiple decision variables, and (2) the principal component analysis technique can be first used to distinguish the main variables, and then the correlation between any two main variables can be captured by the covariance matrix. Besides, we will extend the BSABC to solve some other challenging real-world optimization problems, such as the fuzzy portfolio selection problem, the multilevel image segmentation problem and the automatic data clustering problem.

ACKNOWLEDGMENTS

This study is supported by the National Natural Science Foundation of China (Nos. 61966019, 61876074 and 61866017), the Natural Science Foundation of Jiangxi Province (No. 20192BAB207030), the Postgraduate Innovation Fund Project of Jiangxi Provincial Department of Education (No. YC2021-S307), and the Graduate Innovation Fund project of Jiangxi Normal University (No. YJS2020023).

CONFLICT OF INTERESTS

The authors declare that there are no conflict of interests.

AUTHOR CONTRIBUTIONS

Xinyu Zhou: Conceptualization, Methodology, Writing—original draft, Writing—review and editing. **Junhong Huang**: Methodology, Software, Data curation, Writing—review and editing. **Hao Tang**: Formal analysis, Writing—review and editing. **Mingwen Wang**: Formal analysis, Writing—review and editing.

ORCID

Xinyu Zhou https://orcid.org/0000-0001-9443-5256
Junhong Huang https://orcid.org/0000-0002-2994-3331
Hao Tang https://orcid.org/0000-0002-8781-9558
Mingwen Wang https://orcid.org/0000-0002-8118-9612

REFERENCES

1. Li X, Shi J, Dong X, Yu J. A new conjugate gradient method based on quasi-newton equation for unconstrained optimization. *J Comput Appl Math.* 2019;350:372-379.

2. Ma L, Cheng S, Shi Y. Enhancing learning efficiency of brain storm optimization via orthogonal learning design. *IEEE Trans Syst, Man Cybern Syst.* 2021;51(11):6723-6742.
3. Son NN, Van Kien C, Anh HP. Hysteresis compensation and adaptive control based evolutionary neural networks for piezoelectric actuator. *Int J Intell Syst.* 2021;36(10):5472-5492.
4. Wang F, Li Y, Zhou A, Tang K. An estimation of distribution algorithm for mixed-variable news vendor problems. *IEEE Trans Evol Comput.* 2020;24(3):479-493.
5. Abdollahzadeh B, Soleimanian Gharehchopogh F, Mirjalili S. Artificial gorilla troops optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. *Int J Intell Syst.* 2021;36(10): 5887-5958.
6. Gu Q, Chen S, Jiang S, Xiong N. Improved strength Pareto evolutionary algorithm based on reference direction and coordinated selection strategy. Int J Intell Syst. 2021.
7. Zhang XY, Zhang J, Gong YJ, Zhan ZH, Chen WN, Li Y. Kuhn-Munkres parallel genetic algorithm for the set cover problem and its application to large-scale wireless sensor networks. *IEEE Trans Evol Comput.* 2015;20(5):695-710.
8. Wang F, Zhang H, Zhou A. A particle swarm optimization algorithm for mixed-variable optimization problems. *Swarm Evol Comput.* 2021;60:100808.
9. Chen ZG, Zhan ZH, Wang H, Zhang J. Distributed individuals for multiple peaks: a novel differential evolution for multimodal optimization problems. *IEEE Trans Evol Comput.* 2019;24(4):708-719.
10. Akay B, Karaboga D, Gorkemli B, Kaya E. A survey on the Artificial Bee Colony algorithm variants for binary, integer and mixed integer programming problems. *Appl. Soft Comput.* 2021;106:107351.
11. Abdel-Basset M, Mohamed R, Chakrabortty RK, Ryan MJ. IEGA: an improved elitism-based genetic algorithm for task scheduling problem in fog computing. Int. J. Intell. Syst. 2021.
12. Liu F, Huang H, Li B, Xi F. A parallel learning particle swarm optimizer for inverse kinematics of robotic manipulator. *Int J Intell Syst.* 2021;36(10):6101-6132.
13. Jia H, Peng X, Lang C. Remora optimization algorithm. *Expert Syst Appl.* 2021;185:115665.
14. Abdollahzadeh B, Gharehchopogh FS, Mirjalili S. African vultures optimization algorithm: a new nature-inspired metaheuristic algorithm for global optimization problems. *Comput Ind Eng.* 2021;158:107408.
15. Mirjalili S, Lewis A. The whale optimization algorithm. *Adv Eng Softw.* 2016;95:51-67.
16. Karaboga D. An idea based on honey bee swarm for numerical optimization. tech. rep., Technical report-tr06, Erciyes university, engineering faculty; 2005.
17. Karaboga D, Akay B. A comparative study of artificial bee colony algorithm. *Appl Math Comput.* 2009; 214(1):108-132.
18. Gao W, Sheng H, Wang J, Wang S. Artificial bee colony algorithm based on novel mechanism for fuzzy portfolio selection. *IEEE Trans Fuzzy Syst.* 2018;27(5):966-978.
19. Wang X, Zhang Y, Sun X, Wang Y, Du C. Multi-objective feature selection based on artificial bee colony: an acceleration approach with variable sample size. *Appl Soft Comput.* 2020;88:106041.
20. Liu H, Xu B, Lu D, Zhang G. A path planning approach for crowd evacuation in buildings based on improved artificial bee colony algorithm. *Appl Soft Comput.* 2018;68:360-376.
21. Gupta S, Deep K. Hybrid sine cosine artificial bee colony algorithm for global optimization and image segmentation. *Neural Comput Appl.* 2020;32(13):9521-9543.
22. Singh A, Deep K. Exploration-exploitation balance in Artificial Bee Colony algorithm: a critical analysis. *Soft Comput.* 2019;23(19):9525-9536.
23. Song X, Zhao M, Xing S. A multi-strategy fusion artificial bee colony algorithm with small population. *Expert Syst Appl.* 2020;142:112921.
24. Gao W, Gao S. A modified artificial bee colony algorithm. *Comput Oper Res.* 2012;39(3):687-697.
25. Zhu G, Kwong S. Gbest-guided artificial bee colony algorithm for numerical function optimization. *Appl Math Comput.* 2010;217(7):3166-3173.
26. Zhou X, Wu Z, Wang H, Rahnamayan S. Gaussian bare-bones artificial bee colony algorithm. *Soft Comput.* 2016;20(3):907-924.
27. Kong D, Chang T, Dai W, Wang Q, Sun H. An improved artificial bee colony algorithm based on elite group guidance and combined breadth-depth search strategy. *Inform Sci.* 2018;442:54-71.
28. Zhou X, Lu J, Huang J, Zhong M, Wang M. Enhancing artificial bee colony algorithm with multi-elite guidance. *Inform Sci.* 2021;543:242-258.

29. Noman N, Iba H. Accelerating differential evolution using an adaptive local search. *IEEE Trans Evol Comput.* 2008;12(1):107-125.
30. Guo SM, Yang CC. Enhancing differential evolution utilizing eigenvector-based crossover operator. *IEEE Trans Evol Comput.* 2014;19(1):31-49.
31. Wang H, Wu Z, Rahnamayan S, Sun H, Liu Y, Pan Js, Multi-strategy ensemble artificial bee colony algorithm. *Inform Sci.* 2014;279:587-603.
32. Kiran MS, Hakli H, Gunduz M, Uguz H. Artificial bee colony algorithm with variable search strategy for continuous optimization. *Inform Sci.* 2015;300:140-157.
33. Qin AK, Huang VL, Suganthan PN. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Trans Evol Comput.* 2008;13(2):398-417.
34. Gao W, Huang L, Liu S, Chan FT, Dai C, Shan X. Artificial bee colony algorithm with multiple search strategies. *Appl Math Comput*. 2015;271:269-287.
35. Gao W, Wei Z, Luo Y, Cao J. Artificial bee colony algorithm based on Parzen window method. *Appl Soft Comput.* 2019;74:679-692.
36. Banitalebi A, AbdAziz MI, Bahar A, Aziz ZA. Enhanced compact artificial bee colony. *Inform Sci*. 2015;298: 491-511.
37. Jadon SS, Tiwari R, Sharma H, Bansal JC. Hybrid artificial bee colony algorithm with differential evolution. *Appl Soft Comput.* 2017;58:11-24.
38. Zhou X, Wang H, Wang M, Wan J. Enhancing the modified artificial bee colony algorithm with neighborhood search. *Soft Comput.* 2017;21(10):2733-2743.
39. Wang H, Wang W, Xiao S, Cui Z, Xu M, Zhou X. Improving artificial bee colony algorithm using a new neighborhood selection mechanism. *Inform Sci.* 2020;527:227-240.
40. Wang Y, Li HX, Huang T, Li L. Differential evolution based on covariance matrix learning and bimodal distribution parameter setting. *Appl Soft Comput.* 2014;18:232-247.
41. Demmel J, Veselić K. Jacobi's method is more accurate than QR. *SIAM J Matrix Anal. A*. 1992;13(4): 1204-1245.
42. Strang G, Borre K. *Linear Algebra, Geodesy, and GPS*. Siam; 1997.
43. Liang J, Qu B, Suganthan P, Hernández-Díaz AG. Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report; 2013. 2013201212(34):281-295.
44. Yang J, Jiang Q, Wang L, et al. An adaptive encoding learning for artificial bee colony algorithms. *J Comput Sci-NETH.* 2019;30:11-27.
45. Do CB. *The multivariate Gaussian distribution*. Section Notes, Lecture on Machine Learning, CS; 2008:229.
46. Xiang Wl, Li Y, He R, Gao M, An M. A novel artificial bee colony algorithm based on the cosine similarity. *Comput Ind Eng*. 2018;115:54-68.
47. Hong PN, Ahn CW. Linkage artificial bee colony for solving linkage problems. *Expert Syst Appl.* 2016;61: 378-385.
48. Wang H, Sun H, Li C, Rahnamayan S, Pan J, Diversity enhanced particle swarm optimization with neighborhood search. *Inform Sci.* 2013;223:119-135.
49. Zhou X, Wu Y, Zhong M, Wang M. Artificial bee colony algorithm based on multiple neighborhood topologies. *Appl Soft Comput.* 2021;111:107697.
50. Peng H, Deng C, Wu Z. Best neighbor-guided artificial bee colony algorithm for continuous optimization problems. *Soft Comput.* 2019;23(18):8723-8740.
51. Sharma TK, Pant M. Shuffled artificial bee colony algorithm. *Soft Comput.* 2017;21(20):6085-6104.
52. Shao P, Yang L, Tan L, Li G, Peng H. Enhancing artificial bee colony algorithm using refraction principle. *Soft Comput.* 2020:1-16.
53. Gao WF, Huang LL, Liu SY, Dai C. Artificial bee colony algorithm based on information learning. *IEEE Trans Cybernet* 2015;45(12):2827-2839.
54. Cui L, Zhang K, Li G, et al. Modified Gbest-guided artificial bee colony algorithm with new probability model. *Soft Comput.* 2018;22(7):2217-2243.
55. Aslan S, Badem H, Karaboga D. Improved quick artificial bee colony (iqABC) algorithm for global optimization. *Soft Comput.* 2019;23(24):13161-13182.

56. Hansen N, Ostermeier A. Completely derandomized self-adaptation in evolution strategies. *Evol Comput.* 2001;9(2):159-195.
57. Wang Y, Liu ZZ, Li J, Li HX, Yen GG. Utilizing cumulative population distribution information in differential evolution. *Appl Soft Comput.* 2016; 48: 329-346.
58. Xia X, Xing Y, Wei B, et al. A fitness-based multi-role particle swarm optimization. *Swarm Evol Comput.* 2019;44:349-364.
59. Horner A, Beauchamp J, Haken L. Machine tongues XVI: genetic algorithms and their application to FM matching synthesis. *Comput Music J.* 1993;17(4):17-29.

How to cite this article: Zhou X, Huang J, Tang H, Wang M. Artificial bee colony algorithm with bi-coordinate systems for global numerical optimization. *Int J Intell Syst*. 2022;37:5816-5858. doi:10.1002/int.22816