

## Research Article

# A Tristage Adaptive Biased Learning for Artificial Bee Colony

Qiaoyong Jiang <sup>1,2</sup>, Yueqi Ma,<sup>1</sup> Yanyan Lin <sup>3</sup>, Jianan Cui,<sup>1</sup> Xinjia Liu,<sup>1</sup> Yali Wu,<sup>4</sup>  
and Lei Wang<sup>1</sup>

<sup>1</sup>College of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China

<sup>2</sup>Shaanxi Key Laboratory of Network Computing and Security Technology, Xi'an University of Technology, Xi'an 710048, China

<sup>3</sup>College of Information Engineering, Xi'an University, Xi'an 710065, China

<sup>4</sup>College of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China

Correspondence should be addressed to Qiaoyong Jiang; [jiangqiaoyong@126.com](mailto:jiangqiaoyong@126.com) and Yanyan Lin; [linyy1027@126.com](mailto:linyy1027@126.com)

Received 1 August 2021; Accepted 3 September 2021; Published 23 September 2021

Academic Editor: Shi Cheng

Copyright © 2021 Qiaoyong Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent ten years, artificial bee colony (ABC) has attracted more and more attention, and many state-of-the-art ABC variants (ABCs) have been developed by introducing different biased information to the search equations. However, the same biased information is employed in employed bee and onlooker bee phases, which will cause over exploitation and lead to premature convergence. To overcome this limit, an effective framework with tristage adaptive biased learning is proposed for existing ABCs (TABL + ABCs). In TABL + ABCs, the search direction in the employed bee stage is guided by learning the ranking biased information of the parent food sources, while in the onlooker bee stage, the search direction is determined by extracting the biased information of population distribution. Moreover, a deletion-restart learning strategy is designed in scout bee stage to prevent the potential risk of population stagnation. Systematic experiment results conducted on CEC2014 competition benchmark suite show that proposed TABL + ABCs perform better than recently published AEL + ABCs and ACoS + ABCs.

## 1. Introduction

Optimization has long been a basic research topic, which attracts an ever-increasing interest from scientific research to engineering practice due to its obvious application potential in almost all real-world systems. Without losing generality, a box-constrained optimization problem can be modelled as follows:

$$\begin{aligned} & \min f(x), \\ & \text{s.t. } x = (x_1, x_2, \dots, x_D), \end{aligned} \quad (1)$$

where  $f(x)$  is a real-value objective function and  $x$  is a candidate solution in search space  $\Omega = \prod_{i=1}^D [L_i, U_i]$ , where  $U_i$  and  $L_i$  are the upper and lower boundaries for the  $i$ th dimension.

To better achieve the goal of global optimization, various optimization technologies have been developed, mainly including mathematical programming and evolutionary

computing (EC). Compared with mathematical programming, EC has its own advantages in the weak assumption of mathematical properties for optimized problems and high probability to find a global optimal solution. In the past thirty years, a variety of EC algorithms have been proposed, such as genetic algorithm (GA) [1], differential evolution (DE) [2], particle swarm optimization (PSO) [3], and artificial bee colony (ABC) [4]. As a relatively new evolutionary optimization algorithm, ABC has become popular in EC community due to its simple concept and easy implementation yet effectiveness.

ABC developed by Karaboga is a population-based stochastic algorithm which simulates the foraging behavior of honey bees. A recent study has shown that ABC performs a good performance on many box-constrained continuous optimization problems [5–9], but its slow convergence rate has been widely criticized. The main reason is that the solution search strategy of ABC is good at exploration but poor at exploitation. Therefore, how to properly balance the

exploration and exploitation during the search process is the core idea of improvement works of ABC. For this purpose, a lot of improvement works have been developed in the last ten years. In the following, a brief survey about ABC is reviewed from four aspects.

*1.1. Search Equations with Biased Information.* In the original ABC, one parent of the solution search equation is the current target vector, and the other is randomly chosen from the population. Therefore, one of the reasons for the slow convergence of ABC is that the search equation has no biased information. To overcome this limitation, many researchers have proposed various solution search equations with biased information [10–17]. For example, Cui et al. proposed an adaptive ABC (ARABC) [18], in which all parent food sources were chosen based on their rankings. Xiang et al. put forward a gravity model-based ABC (ABCG) [19]. In ABCG, an attractive force model was designed to select a better neighbor of the current target vector. Kumar and Mishra proposed a covariance-based guided ABC (CABC) [20], in which the covariance information was embedded to accelerate the convergence. Aslan et al. designed an improved qABC (iqABC) [21] to balance the search ability. Ji et al. proposed a scale-free ABC (SFABC) [22], in which the topology information of a scale-free network introduced into ABC. Awadallah et al. introduced natural selection methods for ABC (NSM + ABCs) [23]. Bajer and Zoric proposed an improved ABC based on diversity refining (DRABC) [24]. In DRABC, the solution search equation was modified by introducing the information of the top population members to increase the exploitation ability.

*1.2. Ensemble of Multiple Search Strategies.* In the original ABC, only one solution search equation is used as the update rule during the search process, which is considered to be very difficult to adapt to optimization problems with different characteristics. To enhance the robustness of ABC, many researchers make efforts to realize an ensemble of multiple search strategies [25–34]. For instance, Song et al. designed a two-strategy adaptive ABC (TSaABC) [35]. In TSaABC, the two search strategies were dynamically adjusted by success rate to balance the exploration and exploitation ability. Chen et al. developed a self-adaptive differential ABC (sdABC) [36]. In sdABC, three differential search strategies were combined into the framework of ABC, and the selection probability of each strategy was adjusted adaptively. Zhou et al. presented a multicolony ABC (IDABC) [37], in which the whole colony was divided into superior, mid, and inferior subcolonies, and three search strategies with different biases were assigned into the corresponding subcolony for different roles. Gao et al. introduced a Parzen window-based ABC (ABCPW) [38], in which three search strategies with different characteristics were estimated by PW and the best one was used to generate offspring. Yavuz and Aydin designed a self-adaptive search equation-based ABC (SSEABC) [39] which integrated three local search strategies into the ABC, and each strategy was

selected by competitive learning. Very recently, Song et al. put forward a multistrategy fusion ABC (MFABC) [40], in which the search strategy with high exploration was inherited and the other two search strategies were selected adaptively according to the evolution ratio.

*1.3. Hybridization of ABCs and Other Metaheuristic Algorithms.* Combining the advantages of different algorithms is an effective way to design high-performance algorithms. According to this idea, many researchers have proposed various hybrid algorithms based on ABCs. For example, Gao et al. proposed an enhanced ABC through DE (DGABC) [41] which combined the good features of DE and GABC to accelerate the convergence. Liang et al. employed adaptive differential operator to remedy the limitation of slow convergence speed of ABC (ABCADe) [42]. Ghanem and Jantan mixed the ABC with monarch butterfly optimization (HAM) [43], in which the modified butterfly adjusting operator was used as the search equation of employed bees. Wang and Yi developed a hybridization algorithm of krill herd (KH) and ABC (KHABC) [44], in which the found optimal solutions by KHABC were considered as a neighbor food source for onlooker bees. Chen et al. proposed a hybrid ABC (TLABC) [45] by combining the teaching-learning-based optimization with the ABC. Very recently, Chen et al. further presented a mixed ABC based on fireworks explosion (FW-ABC) [46]. In FW-ABC, the fireworks explosion search was implemented to find better solutions after three bee search stages.

*1.4. Combination of ABCs and Local Search Techniques.* The success of memetic framework has confirmed the effectiveness of the combination of metaheuristic algorithms and local search techniques, which drives many researchers to design various enhanced ABCs with local search strategies. For example, Kang et al. presented a memetic ABC based on the Rosenbrock approach (RABC) [47]. In RABC, the Rosenbrock rotational direction was used to improve the quality of the optimal solution. Kang et al. further proposed another memetic ABC (HABC) [48], in which the Hooke–Jeeves method was carried out to improve the local search capability. Gao et al. introduced two other memetic ABCs, denoted as PABC [49] and OL-ABC [50], respectively. In PABC and OL-ABC, Powell's method and orthogonal crossover operator were employed to enhance their exploitation ability.

Based on the previously reported experimental results, we note that most ABCs are very efficient for separable functions, but they still suffer from low convergence rate for nonseparable functions. Therefore, there is much room for improvement of existing ABCs in terms of nonseparable functions. Nevertheless, not much effort on nonseparable functions for ABCs has been done so far. In our recently published work [51, 52], an adaptive encoding learning was introduced to improve the performance of existing ABCs (AEL + ABCs) on complex nonseparable functions. Different from most existing ABCs based on the individual biased information to guide search, the AEL method guides the

population to converge to the promising region by extracting the population distribution information. It is therefore natural to combine both types of biased information into existing ABCs for further improving their performance on complex nonseparable functions. To achieve this goal, we propose a tristage adaptive biased learning framework for ABCs (TABL + ABCs), where the main new contributions are summarized as follows:

- (i) A tristage adaptive biased learning is designed for existing ABCs, in which the search direction in the employed bee stage is guided by the ranking biased information of the parent food sources, while in the onlooker bee stage, the search direction is determined by extracting the biased information of population distribution. Both types of biased information are adaptively learned from the feedback information of the population evolution.
- (ii) A deletion-restart learning strategy is designed in scout bee stage to prevent the potential risk of population stagnation.
- (iii) The proposed TABL framework is applied to several existing ABCs. Systematic experiment results conducted on CEC2014 competition benchmark suite show that proposed TABL + ABCs has better optimization performance than the corresponding ABCs, recently published AEL + ABCs and ACoS + ABCs.

In the rest of the paper, we first describe the original ABC in Section 2. Section 3 develops a tristage adaptive biased learning for ABCs. Section 4 designs a set of experiments to verify the advantages of TABL + ABCs, and some discusses are made in Section 5. The proposed algorithm is applied to big optimization problems in Section 6. Finally, Section 7 concludes the entire work and identifies some future studies.

## 2. Artificial Bee Colony

The ABC is a population-based intelligent optimization technique. It is inspired by the foraging behavior of bees and their cooperative foraging process is transformed into a search mechanism of optimal solution. In ABC, the location of a food source is hypothesized as a candidate solution, and its fitness is measured by the amount of nectar. During the cooperative foraging, employed bees are employed to explore the food sources and pass on their quality information to onlooker bees through waggle dancing. Based on the obtained quality information, onlooker bees select some better food sources for further exploitation. If the quality of some food source is not improved over a predefined cycle (limit), the food source is considered as exhausted and its associated employed bee is transformed into a scout bee to explore other food source in a random way. Algorithm 1 is the main framework of ABC and its key steps are introduced as follows:

- (i) *Initialization*. ABC randomly creates  $SN$  food sources  $x = \{x_1, x_2, \dots, x_N\}$  by the following equation:

$$x_{i,j} = L_j + \text{rand}_{i,j}[0, 1] \cdot (U_j - L_j), \quad (2)$$

where  $i \in \{1, 2, \dots, SN\}$ ,  $j \in \{1, 2, \dots, D\}$ .  $\text{rand}_{i,j}[0, 1]$  is a random number between 0 and 1. The quality information of each food source is measured by the objective function  $f(\cdot)$ . The smaller the  $f(\cdot)$  is, the better the solution quality is.

- (ii) *Employed Bee Phase*. After initialization, each employed bee begins to visit its associated food source  $x_i$  in an attempt to explore a better one  $v_i$ . The process is performed by equation (3), which updates a single component of  $x_i$  as the linear combination:

$$v_{i,j} = x_{i,j} + \phi \cdot (x_{i,j} - x_{k,j}), \quad (3)$$

where  $\phi \in [-1, 1]$ ,  $k \in \{1, 2, \dots, SN\} \setminus \{i\}$  and  $j \in \{1, 2, \dots, D\}$ . The better one between  $v_i$  and  $x_i$  in terms of quality will survive to the next generation.

- (iii) *Onlooker Bee Phase*. Based on the quality information of the food sources, each onlooker bee will fly to a better food source  $x_i$  for further exploitation in terms of the selection probability which is calculated by the following equations:

$$p_i = \frac{\text{fit}_i}{\sum_{i=1}^{SN} \text{fit}_i}, \quad (4)$$

$$\text{fit}_i = \begin{cases} \frac{1}{1 + |f_i|}, & \text{if } f_i \geq 0, \\ 1 + |f_i|, & \text{otherwise,} \end{cases} \quad (5)$$

where  $\text{fit}_i$  is the fitness value of the  $i$ th solution. Obviously, solutions with larger fitness values have more opportunity to be selected.

- (iv) *Scout Bee Phase*. During the search process, each update of a particular food source in the previous two phases is tracked. When the number of unsuccessful update attempts exceeds the threshold *limit*, the food source exceeding the threshold by the largest amount denoted as equation (6) is replaced by equation (2):

$$\text{arg max}_{i \in \{1, 2, \dots, SN\}} \{i: n_i \geq \text{limit}\}, \quad (6)$$

where  $n_i$  is the number of consecutive unsuccessful update associated with the  $i$ th food source  $x_i$ .

## 3. Proposed TABL + ABCs

*3.1. Motivation*. The AEL framework introduced in [51] is a very promising method. However, we also notice that the AEL framework is not always effective or even negative for ABCs on some multimode problems. Figure 1 shows that the AEL strategy can effectively accelerate the convergence speed but significantly reduce the convergence accuracy. The

```

//Initialization
Randomly create  $SN$  food sources  $\mathbf{x}_i, i = 1, 2, \dots, SN$  by Equation (2);
Evaluate the quality information of each food source  $f(\mathbf{x}_i), i = 1, 2, \dots, SN$ ;
while Termination condition is not met do
  //Employed bee phase
  for  $i = 1, 2, \dots, SN$  do
    Create a new food source  $\mathbf{v}_i$  near the  $\mathbf{x}_i$  by Equation (3);
    Evaluate the quality information of  $\mathbf{v}_i$ ;
    if  $f(\mathbf{v}_i) < f(\mathbf{x}_i)$  then
       $\mathbf{x}_i = \mathbf{v}_i$ ;
       $n_i = 0$ ;
    else
       $n_i = n_i + 1$ ;
    end
  end
  //Onlooker bee phase
  Calculate the selection probability  $p_i$  for each onlooker bee by Equation (4);
   $t = 0, i = 1$ ;
  while  $t \leq SN$  do
    if  $\text{ran } d[0, 1] < p_i$  then
      Create a new food source  $\mathbf{v}_i$  near the  $\mathbf{x}_i$  by Equation (3);
      Evaluate the quality information of  $\mathbf{v}_i$ ;
      if  $f(\mathbf{v}_i) < f(\mathbf{x}_i)$  then
         $\mathbf{x}_i = \mathbf{v}_i$ ;
         $n_i = 0$ ;
      else
         $n_i = n_i + 1$ ;
      end
       $t = t + 1$ ;
    end
    if  $i == SN$  then
       $i = 1$ ;
    end
  end
  //Scout bee phase
  if  $\max(n_i) > \text{limit}$  then
    Reinitialize  $\mathbf{x}_i$  by Equation (2);
     $n_i = 0$ ;
  end
end

```

ALGORITHM 1: The main flowchart of ABC

reason is that the information of population distribution is used in two search stages may increase the risk of falling into local optimum for some complex multimode problems. In addition, the experimental results of [52] have been clearly shown that most existing ABCs are only efficient for separable functions, while the convergence rate of these algorithms is still very poor when working with nonseparable functions.

Motivated by the above two aspects, an effective framework with tristage adaptive biased learning is proposed for existing ABCs (TABL + ABCs) to improve their performance on nonseparable problems, and its main framework is described in Algorithm 2. In the following, we will elaborate on the TABL + ABCs framework.

**3.2. Rank-Based Biased Learning.** As discussed in introduction, the imbalance between exploration and exploitation is the focus of the argument about ABC. One of the reasons for this phenomenon is the lack of biased information in the solution search equation. In order to reverse this phenomenon, many improved ABCs have been proposed by introducing various biased information into solution search equations. Most of them are inspired by the natural phenomenon that good individuals always contain good biased information and thus they always have more chances to generate good offspring. Different from the guidance mechanism based on single individual biased information, the AEL method extracted the biased information of population distribution to guide the search. As discussed in Section 3.1, this method may also

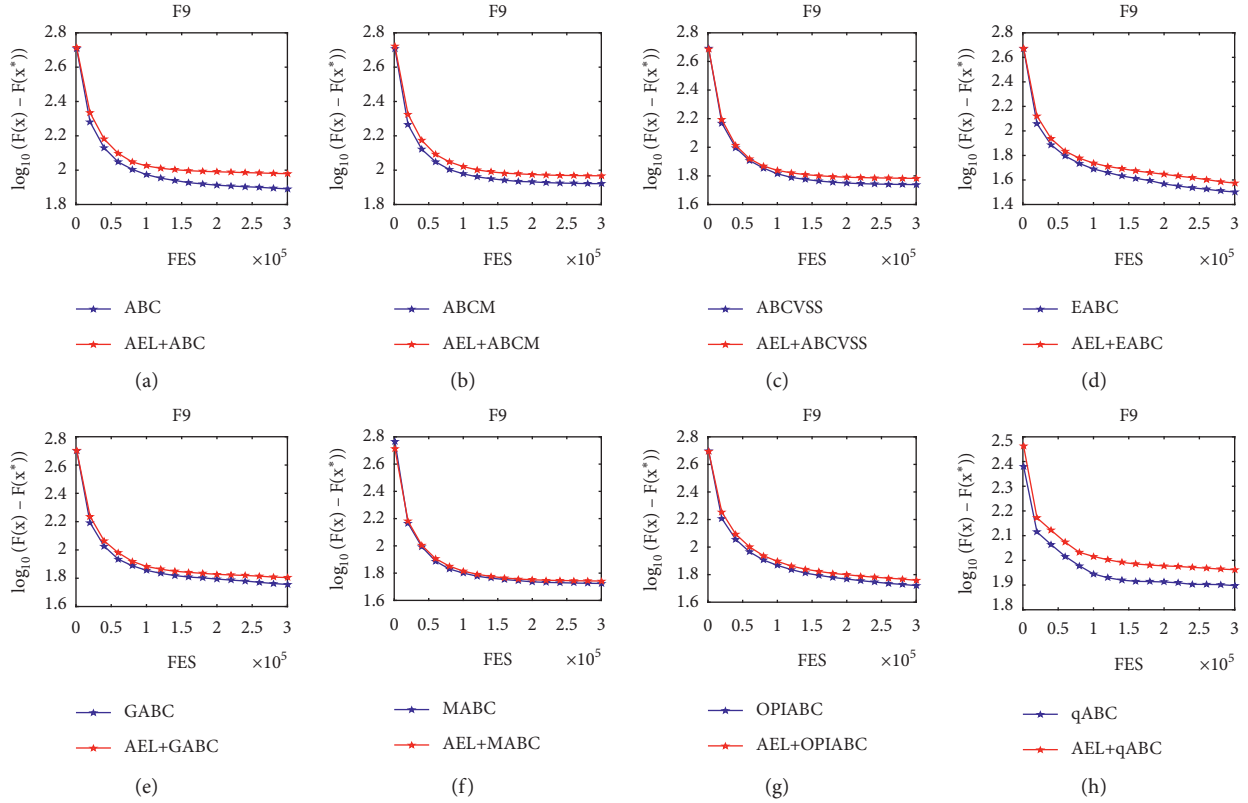


FIGURE 1: The convergence curves of AEL + ABCs vs. ABCs on F9 with 30D.

increase the risk of falling into local optimum for some complex multimode problems. The reason is that the same biased information is employed in two search phases, which will cause over exploitation and lead to premature convergence.

To reduce the over exploitation caused by the population distribution information, the rank-based biased information instead of it is introduced into the solution search equation for guiding the search of employed bees. The main process is described lines 8-18 of Algorithms 2 and 3. The modified solution search equation with the rank-based biased learning is designed as follows:

$$v_{i,j} = x_{i,j} + \phi \cdot (x_{i,j} - \tilde{x}_{k,j}). \quad (7)$$

It is obvious that the only difference between equations (3) and (7) is that the parent food source  $x_k$  of equations (3) is selected with the same probability from the population, while the parent food source  $\tilde{x}_k$  in equation (7) is selected based on its ranking among the population. According to the linear ranking model [18], the selection probability  $\text{Prob}_k$  of the  $k$ th food source is calculated as

$$\text{Prob}_k = \frac{R_k}{\text{SN}}, \quad (8)$$

where  $R_k$  is the linear ranking of  $k$ th food source among the population by fitness value. Apparently, the higher the ranking is, the greater the selection probability is. Compared with equation (3), equation (7) can make each target individual converge to the global optimal position faster by

learning the biased information from higher ranking solutions. Figure 2 shows this process. In an ideal state, the solution  $P$  can reach the global optimal point  $P^*$  of the quadratic function by only two-step search ( $P \rightarrow P_1 \rightarrow P^*$  or  $P \rightarrow P_2 \rightarrow P^*$ ) and achieve the global best of the rotated quadratic function by only three-step search ( $P \rightarrow P_1 \rightarrow P_2 \rightarrow P^*$  or  $P \rightarrow P_3 \rightarrow P_4 \rightarrow P^*$ ).

In addition, it is noteworthy that the target vector  $x_i$  in equation (7) is still selected from the population in turn, while in [18], it is selected based on the selection probability. The reason is that the target vectors have been proportionally selected by equation (4) in the onlooker bee phase, and the similar biased information adopted in the employed bee phase may cause over exploitation and premature convergence. To further confirm it, Figure 3 shows the situation based on a typical multimodal problem, where the higher rank solutions are distributed near the local optimal solution, and the lower ranking solutions are distributed near the global optimal solution. In this case, ABC is not sufficient to yield any progress and may be further intensified into local convergence since these higher rank solutions are frequently selected for search.

**3.3. Population Distribution Information-Based Biased Learning.** For a separable function, the original coordinate system  $x_1 - o - x_2$  is the best choice for ABC. This is because the original coordinate system can maximize the improvement interval such as Figure 4(a). When the fitness



```

//Initialization
Randomly create  $SN^{(max)}$  food sources  $\mathbf{x}_i, i = 1, 2, \dots, SN^{(max)}$  by Equation (2);
Evaluate the quality information of each food source  $f(\mathbf{x}_i), i = 1, 2, \dots, SN^{(max)}$ ;
Set  $G = 0, SN^{(0)} = SN^{(max)}$ ;
while Termination condition is not met do
     $f_1 = \min(f(\mathbf{x}))$ ;
    //Employed bee phase
    Assign a linear ranking for each food source  $R_i, i = 1, 2, \dots, SN^{(G)}$ ;
    Calculate the selection probability for each food source  $Prob_i, i = 1, 2, \dots, SN^{(G)}$  by Equation (10);
    for  $i = 1, 2, \dots, SN^{(G)}$  do
        Select the parent food source index  $k$  in Equation (9) by  $LRS(i, SN, \mathbf{Prob})$ ;
        Create a new food source  $\mathbf{v}_i$  near the  $\mathbf{x}_i$  by Equation (9);
        Evaluate the quality information of  $\mathbf{v}_i$ ;
        if  $f(\mathbf{v}_i) < f(\mathbf{x}_i)$  then
             $\mathbf{x}_i = \mathbf{v}_i$ ;
        end
    end
    //Onlooker bee phase
    Compute the covariance matrix  $cov(\mathbf{x}_{1:SN^{(G)}/2})$  of the top  $SN^{(G)}$  food sources;
    Get new coordinate system  $\mathbf{B}$  by the Eigen decomposition of  $cov(\mathbf{x}_{1:SN^{(G)}/2})$ ;
    Calculate the selection probability  $p_i$  for each onlooker bee by Equation (4);
     $t = 0, i = 1$ ;
    while  $t \leq SN$  do
        if  $ran\ d[0, 1] < p_i$  then
            Randomly create a food source  $\mathbf{x}_k$  different from  $\mathbf{x}_i$ ;
            Create a new food source  $\mathbf{v}_i$  near the  $\mathbf{x}_i$  by  $\mathbf{B}$ ;
            Evaluate the quality information of  $\mathbf{v}_i$ ;
            if  $f(\mathbf{v}_i) < f(\mathbf{x}_i)$  then
                 $\mathbf{x}_i = \mathbf{v}_i$ ;
            end
        end
         $t = t + 1$ ;
    end
    if  $i == SN$  then
         $i = 1$ ;
    end
    //Scout bee phase
     $f_2 = \min(f(\mathbf{x}))$ ;
    if  $|f_2 - f_1| < 0.1$  then
        Replace the worst-ranking food source by the new food source generated by "DE/rand/1";
    else
        Calculate  $SN^{(G+1)}$  according to Equation (11);
        Delete  $SN^{(G)} - SN^{(G+1)}$  worst-ranking food sources from the population;
    end
     $G = G + 1$ ;
end

```

ALGORITHM 2: The main flowchart of TABL + ABC

```

Randomly select  $k \in [1, SN]$ ;
while  $ran\ d[0, 1] > Prob_k || k == i$  do
    Randomly select  $k \in [1, SN]$ ;
end
Return  $k$ 

```

ALGORITHM 3:  $LRS(i, SN, \mathbf{Prob})$ .

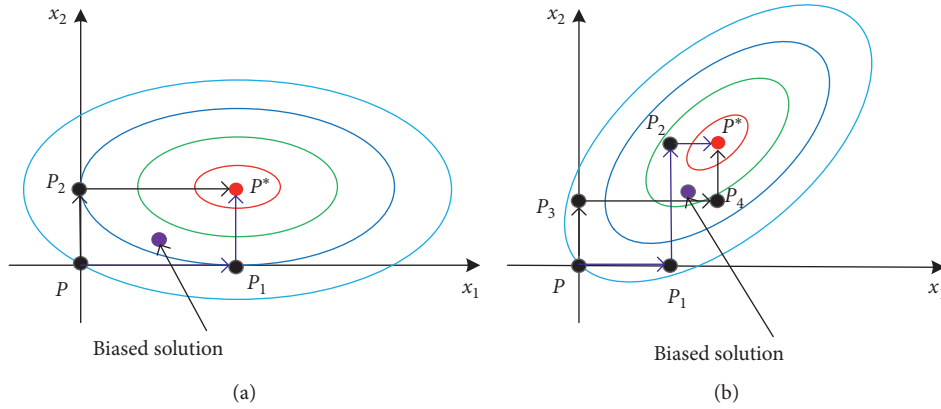


FIGURE 2: The search process of the solution  $P$  by learning the biased information for the quadratic function (a) and the rotated quadratic function (b), respectively.

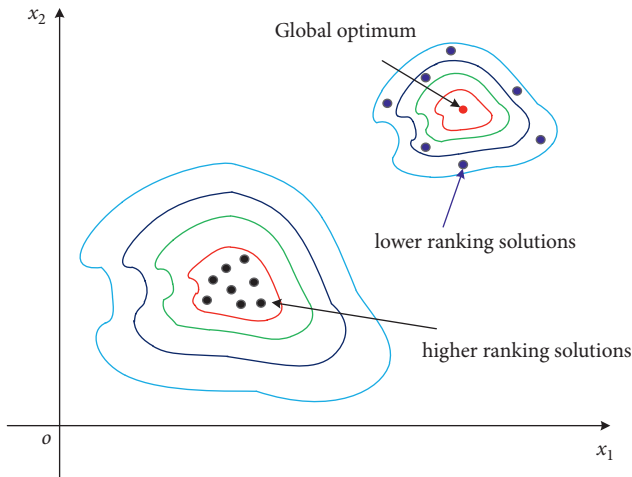


FIGURE 3: The population distribution for a typical multimodal problem.

landscape of Figure 4(a) is rotated, however, the improvement interval under the original coordinate system such as Figure 4(b) is shrunk rapidly, which severely limits the search ability of ABC. Therefore, increasing the improvement interval of variables by turning the coordinate system is an effective way to enhance the optimization performance of ABC on nonseparable problems. According to this, an adaptive Eigen coordinate system  $x'_1 - o - x'_2$  (i.e., **B**) was built by learning the population distribution information. A more detailed introduction can be found in [51].

It is widely known that the covariance matrix of population can release variable correlation to a certain extent. Therefore, the obtained Eigen coordinate system can effectively increase the improvement interval of variables and thus improve the search ability. Figure 5 shows a simple example to illustrate how the Eigen coordinate system increases the improvement interval of variables. In addition, the search direction guided by Eigen coordinate system is based on the biased information of population distribution instead of individual rank-based biased information.

Therefore, the Eigen coordinate system is served as the guiding mechanism in onlooker bee phase, which is conducive to the realization of the complementary advantages of two kinds of biased learning.

**3.4. Deletion-Restart Learning Strategy.** In addition to slow convergence for nonseparable problems, ABC is also prone to fall into local optimum for complex multimodal problems. Figure 6 shows the situation for a typical multimodal problem, where the local optimum position is not distributed on a grid that is aligned with the coordinate system. It is assumed that the population has converged to the vicinity of the lower-left local optimum. In this case, ABC is not sufficient to yield any progress. In other words, the global optimum cannot be reached by modifying one decision variable per food source position.

To detect the state evolution in real time, the improvement of the optimal value of two adjacent generations is identified. If the improvement value is larger than a predefined threshold (e.g., 0.1), the particles should pay more attention exploitation to accelerate the convergence. In this case,  $SN^{(G)} - SN^{(G+1)}$  worst-ranking food sources need be deleted based on the linear decreasing strategy of population size defined in equation (9); otherwise, the particles should pay more attention exploration to prevent the potential risks of stagnation as shown in Figure 6. In this situation, intervention of multivariable perturbation strategy is very useful for jumping out of local optimum. Based on this consideration, the “DE/rand/1” mutation strategy is adopted to create a new food source position to replace the worst one. The reason for “DE/rand/1” mutation strategy is the rotation-invariant property, while the solution search equation in ABC as well as the random initialization operator is rotation-variant. Therefore, the combination of two operators with different properties can achieve complementary advantages:

$$SN^{(G+1)} = \text{round}\left(SN^{(\max)} - \frac{\text{FES}}{\text{MaxFES}} \cdot (SN^{(\max)} - SN^{(\min)})\right), \quad (9)$$

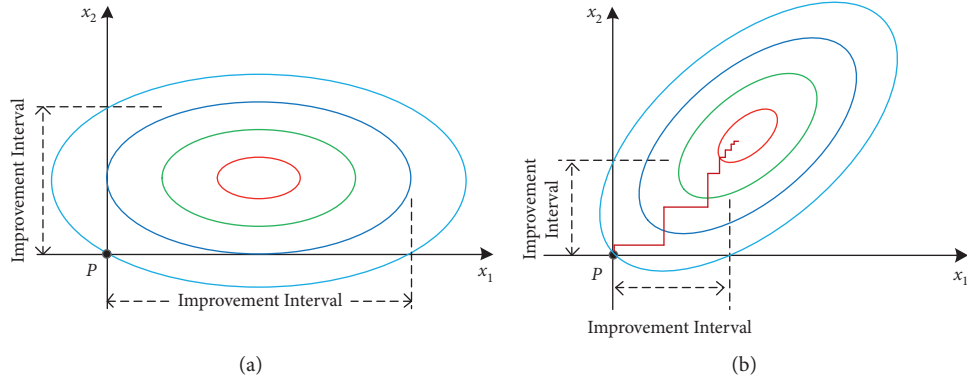


FIGURE 4: (a) Fitness landscape of the quadratic function and (b) fitness landscape of the rotated quadratic function [52].

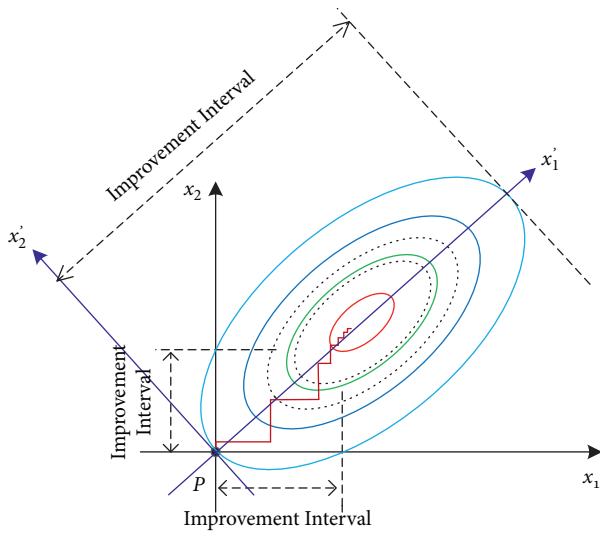


FIGURE 5: Increasing the improvement interval of variables by turning the coordinate system [52].

where  $SN^{(\max)} = 4D$  and  $SN^{(\min)} = 4$ . Whenever  $SN^{(G+1)} < SN^{(G)}$ , the  $SN^{(G)} - SN^{(G+1)}$  worst-ranking food sources are deleted from the population.

#### 4. Experimental Studies

For experimental comparison, 30 CEC2014 competition benchmarks (denoted as F1–F30) with 30 dimensions (30D) are employed for performance test. These competition benchmarks are nonseparable and can be further subdivided into unimodal functions (F1–F3), simple multimodal functions (F4–F16), hybrid functions (F17–F22), and composition functions (F23–F30). A more detailed description about them can be found in [53].

In addition, we performed 51 independent runs for each algorithm on each test problem, which is based on a PC with Intel Core i7-4790 CPU @ 3.60 GHz, 8 GB RAM, and 64 bit Windows 7 OS. In each run, the absolute error  $F(x_{\text{best}}) - F(x^*)$  was recorded when the maximum number of function evaluations  $10000 * D$  was met, where  $x_{\text{best}}$  and  $x^*$  represent the best solution found by some algorithm and the optimal solution of a test problem, respectively. The mean error

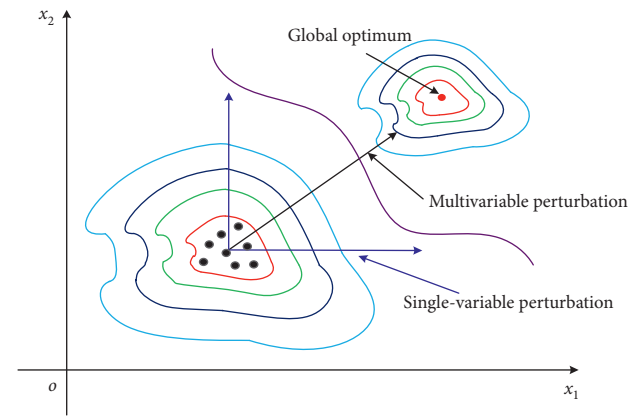


FIGURE 6: The local optimum situation for a typical multimodal problem [52].

values and their standard deviation are considered for performance assessment. Furthermore, the Wilcoxon rank sum test was introduced to test the statistical significance of experimental results between pair of algorithms, and the Friedman test was used to compute the average rankings of all compared algorithms [54].

*4.1. Comparison between TABL and AEL Framework.* The aim of this section is to compare TABL with the AEL framework by applying them to the eight ABCs above. The resultant methods are denoted as TABL+ABCs and AEL+ABCs, respectively. For fair comparison, TABL+ABCs and AEL+ABCs adopt the same parameter settings shown in Table 1, while the AEL+ABCs have an additional learning period LP which is set to 50 suggested in [51]. Tables 2 and 3 summarize the experimental results and some important observations can be made as follows:

- (i) TABL+ABCs perform better than AEL+ABCs for 20, 22, 23, 20, 22, 27, 25, 27 out of 30 test functions with 30D, respectively. However, AEL+ABCs cannot surpass the TABL+ABCs on more than seven test functions.
- (ii) Compared with AEL+ABCs, TABL+ABCs can achieve significant performance improvement on all



TABLE 1: The parameter settings of eight compared ABCs.

Public parameter settings	Private parameter settings
Population size: $SN = 2D$	ABC: limit = 100, GABC: limit = 100, $C = 1.5$
Termination condition: MaxFEs = 10000 * ?	ABCM: limit = $SN * D$ , $M = 2$ ; MABC: $K = 301$
Number of independent runs: 51	ABCVSS: limit = $SN * D$ ; OPIABC: limit = 200
Dimension: ? = 30	EABC: limit = 200, $\mu = 0.3$ , $\gamma = 0.3$ ; OPIABC: $SN * D$

TABLE 2: The statistical results (mean (Std)) of AEL+ABC vs. TABL+ABC, AEL+ABCM vs. TABL+ABCM, AEL+ABCVSS vs. TABL+ABCVSS, and AEL+EABC vs. TABL+EABC over 51 independent runs on the CEC2014 benchmarks with 30D.

Function	AEL+ABC	TABL+ABC	AEL+ABCM	TABL+ABCM	AEL+ABCVSS	TABL+ABCVSS	AEL+EABC	TABL+EABC
F1	2.65E+04‡ (2.24E+04)	<b>6.53E-02</b> (1.28E-01)	2.72E+04‡ (3.18E+04)	<b>8.57E-02</b> (1.64E-01)	5.62E+04‡ (3.22E+04)	<b>6.63E-01</b> (1.79E+00)	1.60E+05‡ (9.83E+04)	<b>1.06E+00</b> (1.87E+00)
F2	<b>1.14E-09</b> † (1.16E-09)	1.79E-07 (5.12E-07)	<b>4.08E-11</b> † (2.41E-10)	4.85E-07 (3.12E-06)	<b>2.79E-14</b> † (3.98E-15)	6.52E-14 (9.73E-14)	1.47E+02‡ (9.52E+02)	<b>2.10E-13</b> (9.93E-13)
F3	3.37E-08§ (1.65E-07)	<b>2.41E-10</b> (1.05E-09)	<b>8.58E-14</b> † (2.87E-14)	7.23E-12 (2.58E-11)	<b>5.68E-14</b> † (0.00E+00)	7.91E-14 (4.56E-14)	5.35E-14‡ (1.35E-14)	<b>2.90E-14</b> (2.87E-14)
F4	1.96E+01‡ (2.69E+01)	<b>8.28E+00</b> (2.11E+01)	1.62E+01‡ (2.69E+01)	<b>1.11E+01</b> (2.35E+01)	1.45E+01‡ (2.64E+01)	<b>6.98E+00</b> (1.25E+01)	<b>1.37E+01</b> § (2.59E+01)	1.42E+01 (2.60E+01)
F5	2.02E+01‡ (4.17E-02)	<b>2.00 E+01</b> (1.42E-04)	2.01E+01‡ (1.69E-02)	<b>2.00E+01</b> (1.66E-03)	2.02E+01‡ (2.47E-02)	<b>2.00E+01</b> (4.75E-05)	2.02E+01‡ (5.25E-02)	<b>2.00E+01</b> (5.33E-04)
F6	1.44E+01‡ (1.35 E+00)	<b>1.06E+01</b> (1.90E+00)	1.46E+01‡ (1.48E+00)	<b>1.11E+01</b> (1.60E+00)	1.36E+01‡ (1.33E+00)	<b>9.13E+00</b> (1.84E+00)	1.18E+01‡ (1.56E+00)	<b>6.25 E+00</b> (2.20E+00)
F7	3.15E-06‡ (4.67E-06)	<b>3.39E-11</b> (5.59E-11)	3.15E-06‡ (8.57E-06)	<b>3.87E-11</b> (1.63E-10)	2.74E-06‡ (2.94E-06)	<b>1.81E-13</b> (1.02E-13)	1.08E-07‡ (2.86E-07)	<b>1.07E-13</b> (4.78E-14)
F8	<b>1.89E-13</b> † (5.41E-14)	4.66E-08 (2.29E-07)	<b>1.09E-13</b> † (2.23E-14)	1.95E-02 (1.48E-01)	1.18E-13‡ (2.23E-14)	<b>7.36E-14</b> (5.49E-14)	1.00E-13‡ (3.70E-14)	<b>8.92E-15</b> (3.09E-14)
F9	9.45E+01‡ (1.58E+01)	<b>8.20E+01</b> (1.36E+01)	9.30E+01‡ (1.62E+01)	<b>8.58E+01</b> (1.21E+01)	6.29E+01§ (8.73E+00)	<b>6.01E+01</b> (9.14E+00)	<b>3.80E+01</b> † (8.07E+00)	4.28E+01 (6.67E+00)
F10	<b>1.24E+00</b> † (7.36E-01)	4.25E+00 (1.33E+00)	<b>1.67E+00</b> † (9.63E-01)	4.14E+00 (1.49E+00)	<b>2.77E-02</b> † (2.69E-02)	4.67E-02 (4.09E-02)	<b>3.83E-01</b> † (5.01E-01)	4.22E-01 (4.05E-01)
F11	1.95E+03‡ (2.21E+02)	<b>1.83E+03</b> (2.02E+02)	2.00E+03‡ (2.34E+02)	<b>1.80E+03</b> (2.25E+02)	1.87E+03‡ (3.10E+02)	<b>1.74E+03</b> (2.54E+02)	1.74E+03‡ (3.24E+02)	<b>1.50E+03</b> (2.48E+02)
F12	1.99E-01‡ (3.77E-02)	<b>9.96E-02</b> (2.46E-02)	1.63E-01‡ (2.31E-02)	<b>9.39E-02</b> (2.38E-02)	1.91E-01‡ (3.25E-02)	<b>8.29E-02</b> (2.45E-02)	2.36E-01‡ (6.40E-02)	<b>1.02E-01</b> (3.31E-02)
F13	2.22E-01‡ (2.59E-02)	<b>2.02E-01</b> (2.68E-02)	2.17E-01‡ (2.75E-02)	<b>1.90E-01</b> (2.74E-02)	2.53E-01‡ (3.19E-02)	<b>2.19E-01</b> (3.35E-02)	2.30E-01‡ (3.26E-02)	<b>2.15E-01</b> (3.40E-02)
F14	1.75E-01§ (1.94E-02)	<b>1.69E-01</b> (2.11E-02)	1.85E-01§ (2.02E-02)	<b>1.84E-01</b> (2.28E-02)	1.94E-01‡ (2.70E-02)	<b>1.75E-01</b> (2.68E-02)	<b>1.71E-01</b> † (2.19E-02)	1.85E-01 (2.52E-02)
F15	8.04E+00‡ (1.19E+00)	<b>4.96E+00</b> (9.35E-01)	7.48E+00‡ (1.03E+00)	<b>4.97E+00</b> (8.34E-01)	5.82E+00‡ (9.94E-01)	<b>3.67E+00</b> (7.56E-01)	4.09E+00‡ (8.33E-01)	<b>3.35E+00</b> (5.89E-01)
F16	9.88E+00§ (3.55E-01)	<b>9.74E+00</b> (3.61E-01)	<b>9.72E+00</b> § (4.47E-01)	9.79E+00 (3.98E-01)	9.66E+00‡ (3.32E-01)	<b>9.50 E+00</b> (3.43E-01)	8.95E+00§ (6.28E-01)	<b>8.93E+00</b> (4.65E-01)
F17	5.34E+02‡ (1.54E+02)	<b>3.78E+02</b> (1.51E+02)	5.19E+02‡ (1.30E+02)	<b>3.93E+02</b> (1.35E+02)	4.37E+02‡ (1.38E+02)	<b>3.00E+02</b> (1.47E+02)	5.30E+02‡ (2.14E+02)	<b>2.43 E+02</b> (1.18E+02)
F18	2.38E+01‡ (6.11E+00)	<b>1.85E+01</b> (5.14E+00)	2.44E+01‡ (7.11E+00)	<b>1.58E+01</b> (5.17E+00)	1.72E+01‡ (6.06E+00)	<b>1.24E+01</b> (3.68E+00)	2.69E+01‡ (1.05E+01)	<b>8.90E+00</b> (2.68E+00)
F19	6.68E+00‡ (6.53E-01)	<b>4.79E+00</b> (8.05E-01)	6.56E+00‡ (7.78E-01)	<b>4.85E+00</b> (6.62E-01)	6.34E+00‡ (5.72E-01)	<b>4.01E+00</b> (3.54E-01)	5.46E+00‡ (9.09E-01)	<b>3.77E+00</b> (6.94E-01)
F20	1.94E+01‡ (5.53E+00)	<b>1.46E+01</b> (4.62E+00)	2.21E+01‡ (6.24E+00)	<b>1.49E+01</b> (4.90E+00)	1.35E+01‡ (3.70E+00)	<b>1.05E+01</b> (3.21E+00)	1.21E+01‡ (4.08E+00)	<b>7.47E+00</b> (2.14E+00)
F21	1.66E+02§ (8.05E+01)	<b>1.54E+02</b> (8.19E+01)	1.76E+02§ (7.95E+01)	<b>1.60E+02</b> (7.68E+01)	1.65E+02‡ (9.29E+01)	<b>1.25E+02</b> (7.32E+01)	2.20E+02‡ (9.45E+01)	<b>1.00E+02</b> (7.45E+01)
F22	2.23E+02§ (8.38E+01)	<b>2.13E+02</b> (7.39E+01)	2.61E+02‡ (8.68E+01)	<b>2.09E+02</b> (8.03E+01)	2.28E+02‡ (8.73E+01)	<b>1.90E+02</b> (7.18E+01)	1.86E+02§ (7.43E+01)	<b>1.75E+02</b> (6.60E+01)
F23	3.15E+02‡ (1.14E-11)	<b>3.15 E+02</b> (1.56E-09)	3.15E+02‡ (3.76E-05)	<b>3.15E+02</b> (2.64E-08)	3.15E+02‡ (1.43E-11)	<b>3.15E+02</b> (5.94E-13)	3.15E+02‡ (7.31E-04)	<b>3.15E+02</b> (3.98E-13)
F24	2.27E+02‡ (1.12E+00)	<b>2.26E+02</b> (8.16E-01)	2.27E+02‡ (1.31E+00)	<b>2.26 E+02</b> (9.00E-01)	2.25E+02‡ (7.41E+00)	<b>2.25E+02</b> (7.30E-01)	<b>2.24E+02</b> † (8.27E-01)	<b>2.25E+02</b> (7.61E-01)
F25	2.03E+02‡ (2.84E-01)	<b>2.03E+02</b> (2.53E-01)	2.03E+02‡ (3.19E-01)	<b>2.03E+02</b> (2.53E-01)	2.03E+02‡ (3.19E-01)	<b>2.03E+02</b> (1.15E-01)	2.03E+02‡ (3.78E-01)	<b>2.03E+02</b> (7.56E-02)

TABLE 2: Continued.

Function	AEL + ABC	TABL + ABC	AEL + ABCM	TABL + ABCM	AEL + ABCVSS	TABL + ABCVSS	AEL + EABC	TABL + EABC
F26	1.00E+02‡ (4.21E−02)	<b>1.00E+02</b> (3.88E−02)	1.00E+02‡ (4.69E−02)	<b>1.00E+02</b> (3.73E−02)	1.00E+02‡ (4.97E−02)	<b>1.00E+02</b> (3.52E−02)	1.00E+02‡ (3.94E−02)	<b>1.00E+02</b> (3.80E−02)
F27	<b>3.93E+02</b> † (5.52E+01)	4.00E+02 (2.05E−01)	4.06E+02‡ (2.55E+00)	<b>4.00E+02</b> (1.34E−01)	4.05E+02‡ (2.70E+00)	<b>4.00E+02</b> (9.05E−02)	4.01E+02‡ (4.69E−01)	<b>4.00E+02</b> (1.49E−01)
F28	1.12E+03‡ (1.40E+02)	<b>9.26E+02</b> (3.63E+01)	1.25E+03‡ (2.03E+02)	<b>9.14E+02</b> (4.39E+01)	9.67E+02‡ (5.82E+01)	<b>8.74E+02</b> (3.98E+01)	8.59E+02§ (4.74E+01)	<b>8.44E+02</b> (2.42E+01)
F29	7.26E+02§ (5.81E+01)	<b>7.15E+02</b> (8.77E−01)	9.84E+02‡ (8.02E+01)	<b>7.06E+02</b> (6.90E+01)	7.15E+02‡ (8.53E−01)	<b>7.15E+02</b> (5.35E−01)	7.98E+02‡ (8.01E+01)	<b>7.15E+02</b> (1.43E+00)
F30	1.68E+03‡ (3.17E+02)	<b>7.35E+02</b> (1.63E+02)	1.96E+03‡ (3.75E+02)	<b>8.14E+02</b> (1.99E+02)	1.47E+03‡ (3.48E+02)	<b>7.34E+02</b> (2.53E+02)	1.46E+03‡ (3.71E+02)	<b>7.93E+02</b> (1.92E+02)
‡/§/†	20/6/4		23/3/4		26/1/3		22/4/4	

the three simple modal functions (F6, F7, and F11), three hybrid functions (F15 and F19–F20), and two composition functions (F28 and F30).

Combining the above experiment results, it is concluded that the proposed TABL framework is more effective than the AEL framework and its main advantage is the performance improvement of complex multimodal problems. It further shows that the proposed TABL approach is effective to fuse the advantages of the ranking biased information and the biased information of population distribution and thus improve optimization accuracy of multimodal problems due to the risk reduction of falling into local optimum. Moreover, as can be seen from the statistical results of the multiproblem Wilcoxon test of Table 4, TABL + ABCs obtain higher  $R^+$  values than  $R^-$  values in all cases, which further confirms that TABL + ABCs perform significantly better than the AEL + ABCs. Finally, the convergence curves of TABL + ABCs and AEL + ABCs on F1, F7, and F17 functions with 30D are shown in Figure 7.

**4.2. Comparison between TABL and ACoS Framework.** The main purpose of this subsection is to compare TABL with the ACoS framework [55] by applying them to the eight ABCs above. The resultant algorithms are denoted as TABL + ABCs and ACoS + ABCs, respectively. Different from the AEL framework, the ACoS framework adaptively tunes the coordinate systems using the cumulative population distribution information. Therefore, the ACoS framework can be considered as an improved version of the AEL framework. For fair comparison, TABL + ABCs and ACoS + ABCs adopt the same parameter settings shown in Table 1. The experimental results are summarized in Tables 5 and 6, and some important observations can be made as follows:

- (i) TABL + ABCs beat the ACoS + ABCs on 22, 20, 19, 19, 20, 17, 22, and 27 out of 30 test functions, respectively. However, ACoS + ABCs cannot surpass the TABL + ABCs on more than six test functions.
- (ii) Compared with ACoS + ABCs, TABL + ABCs can achieve significant performance improvement on all the three simple modal functions (F6, F7, and F11), two hybrid functions (F15–F16), and one composition function (F28).

Based on the above experiment results, it can be seen that the proposed TABL framework is more effective than the ACoS framework. The main advantage of TABL framework is that it can reduce the risk of falling into local optimum for most complex multimodal problems, which shows that the proposed TABL framework can achieve a better balance between exploration and exploitation by tristage adaptive biased learning. Moreover, as can be seen from the statistical results of the multiproblem Wilcoxon test of Table 7, TABL + ABCs obtain higher  $R^+$  values than  $R^-$  values in all cases, which further confirms that TABL + ABCs perform significantly better than the ACoS + ABCs for most cases. Finally, the convergence curves of TABL + ABCs and ACoS + ABCs on F1, F7, and F17 functions with 30D are shown in Figure 8.

**4.3. The Average Ranking of All Algorithms.** The previous comparison of experimental results is based on the Wilcoxon test, which is only applicable to the comparison of a pair of algorithms. To compare the performance of the algorithm more comprehensively, Table 8 gives the average rankings of all compared algorithms based on the Friedman test, and some important observations can be made as follows:

- (i) From Table 8, we can see that the overall rankings of TABL + ABCs are obviously better than AEL + ABCs and ACoS + ABCs, which further verifies the effectiveness of proposed TABL framework.
- (ii) Under the TABL framework, the performance of different algorithms is obviously different, which implies that how to design the matching operators with this framework is one of the key factors affecting the performance of the algorithm.
- (iii) The TABL + qABC ranks the first in all twenty-four comparison algorithms by calculating the sum of the average rankings for 30D.

**4.4. Comparison of Computation Complexity.** To compare the computation complexity of AEL + ABCs, ACoS + ABCs, and TABL + ABCs, we take the AEL + ABC, ACoS + ABC, and TABL + ABC as a comparative case. The computational

TABLE 3: The statistical results (mean (Std)) of AEL + GABC vs. TABL + GABC, AEL + MABC vs. TABL + MABC, AEL + OPIABC vs. TABL + OPIABC, and AEL + qABC vs. TABL + qABC over 51 independent runs on the CEC2014 benchmarks with 30D.

Function	AEL + ABC	TABL + ABC	AEL + ABCM	TABL + ABCM	AEL + ABCVSS	TABL + ABCVSS	AEL + EABC	TABL + EABC
F1	2.65E+04‡ (2.24E+04)	<b>6.53E-02</b> (1.28E-01)	2.72E+04‡ (3.18E+04)	<b>8.57E-02</b> (1.64E-01)	5.62E+04‡ (3.22E+04)	<b>6.63E-01</b> (1.79E+00)	1.60E+05‡ (9.83E+04)	<b>1.06E+00</b> (1.87E+00)
F2	<b>1.14E-09</b> † (1.16E-09)	1.79E-07 (5.12E-07)	<b>4.08E-11</b> † (2.41E-10)	4.85E-07 (3.12E-06)	<b>2.79E-14</b> † (3.98E-15)	6.52E-14 (9.73E-14)	1.47E+02‡ (9.52E+02)	<b>2.10E-13</b> (9.93E-13)
F3	3.37E-08§ (1.65E-07)	<b>2.41E-10</b> (1.05E-09)	<b>8.58E-14</b> † (2.87E-14)	7.23E-12 (2.58E-11)	<b>5.68E-14</b> † (0.00E+00)	7.91E-14 (4.56E-14)	5.35E-14‡ (1.35E-14)	<b>2.90E-14</b> (2.87E-14)
F4	1.96E+01‡ (2.69E+01)	<b>8.28E+00</b> (2.11E+01)	1.62E+01‡ (2.69E+01)	<b>1.11E+01</b> (2.35E+01)	1.45E+01‡ (2.64E+01)	<b>6.98E+00</b> (1.25E+01)	<b>1.37E+01</b> § (2.59E+01)	1.42E+01 (2.60E+01)
F5	2.02E+01‡ (4.17E-02)	<b>2.00E+01</b> (1.42E-04)	2.01E+01‡ (1.69E-02)	<b>2.00E+01</b> (1.66E-03)	2.02E+01‡ (2.47E-02)	<b>2.00E+01</b> (4.75E-05)	2.02E+01‡ (5.25E-02)	<b>2.00E+01</b> (5.33E-04)
F6	1.44E+01‡ (1.35E+00)	<b>1.06E+01</b> (1.90E+00)	1.46E+01‡ (1.48E+00)	<b>1.11E+01</b> (1.60E+00)	1.36E+01‡ (1.33E+00)	<b>9.13E+00</b> (1.84E+00)	1.18E+01‡ (1.56E+00)	<b>6.25E+00</b> (2.20E+00)
F7	3.15E-06‡ (4.67E-06)	<b>3.39E-11</b> (5.59E-11)	3.15E-06‡ (8.57E-06)	<b>3.87E-11</b> (1.63E-10)	2.74E-06‡ (2.94E-06)	<b>1.81E-13</b> (1.02E-13)	1.08E-07‡ (2.86E-07)	<b>1.07E-13</b> (4.78E-14)
F8	<b>1.89E-13</b> † (5.41E-14)	4.66E-08 (2.29E-07)	<b>1.09E-13</b> † (2.23E-14)	1.95E-02 (1.48E-01)	1.18E-03‡ (2.23E-14)	<b>7.36E-14</b> (5.49E-14)	1.00E-13‡ (3.70E-14)	<b>8.92E-15</b> (3.09E-14)
F9	9.45E+01‡ (1.58E+01)	<b>8.20E+01</b> (1.36E+01)	9.30E+01‡ (1.62E+01)	<b>8.58E+01</b> (1.21E+01)	6.29E+01§ (8.73E+00)	<b>6.01E+01</b> (9.14E+00)	<b>3.80E+01</b> † (8.07E+00)	4.28E+01 (6.67E+00)
F10	<b>1.24E+00</b> † (7.36E-01)	4.25E+00 (1.33E+00)	<b>1.67E+00</b> † (9.63E-01)	4.14E+00 (1.49E+00)	<b>2.77E+02</b> † (2.69E-02)	4.67E-02 (4.09E-02)	<b>3.83E-01</b> † (5.01E-01)	4.22E-01 (4.05E-01)
F11	1.95E+03‡ (2.21E+02)	<b>1.83E+03</b> (2.02E+02)	2.00E+03‡ (2.34E+02)	<b>1.80E+03</b> (2.25E+02)	1.87E+03‡ (3.10E+02)	<b>1.74E+03</b> (2.54E+02)	1.74E+03‡ (3.24E+02)	<b>1.50E+03</b> (2.48E+02)
F12	1.99E-01‡ (3.77E-02)	<b>9.96E-02</b> (2.46E-02)	1.63E-01‡ (2.31E-02)	<b>9.39E-02</b> (2.38E-02)	1.91E-01‡ (3.25E-02)	<b>8.29E-02</b> (2.45E-02)	2.36E-01‡ (6.40E-02)	<b>1.02E-01</b> (3.31E-02)
F13	2.22E-01‡ (2.59E-02)	<b>2.02E-01</b> (2.68E-02)	2.17E-01‡ (2.75E-02)	<b>1.90E-01</b> (2.74E-02)	2.53E-01‡ (3.19E-02)	<b>2.19E-01</b> (3.35E-02)	2.30E-01‡ (3.26E-02)	<b>2.15E-01</b> (3.40E-02)
F14	1.75E-01§ (1.94E-02)	<b>1.69E-01</b> (2.11E-02)	1.85E-01§ (2.02E-02)	<b>1.84E-01</b> (2.28E-02)	1.94E-01‡ (2.70E-02)	<b>1.75E-01</b> (2.68E-02)	<b>1.71E-01</b> † (2.19E-02)	1.85E-01 (2.52E-02)
F15	8.04E+00‡ (1.19E+00)	<b>4.96E+00</b> (9.35E-01)	7.48E+00‡ (1.03E+00)	<b>4.97E+00</b> (8.34E-01)	5.82E+00‡ (9.94E-01)	<b>3.67E+00</b> (7.56E-01)	4.09E+00‡ (8.33E-01)	<b>3.35E+00</b> (5.89E-01)
F16	9.88E+00§ (3.55E-01)	<b>9.74E+00</b> (3.61E-01)	<b>9.72E+00</b> § (4.47E-01)	9.79E+00 (3.98E-01)	9.66E+00‡ (3.32E-01)	<b>9.50E+00</b> (3.43E-01)	8.95E+00§ (6.28E-01)	<b>8.93E+00</b> (4.65E-01)
F17	5.34E+02‡ (1.54E+02)	<b>3.78E+02</b> (1.51E+02)	5.19E+02‡ (1.30E+02)	<b>3.93E+02</b> (1.35E+02)	4.37E+02‡ (1.38E+02)	<b>3.00E+02</b> (1.47E+02)	5.30E+02‡ (2.14E+02)	<b>2.43E+02</b> (1.18E+02)
F18	2.38E+01‡ (6.11E+00)	<b>1.85E+01</b> (5.14E+00)	2.44E+01‡ (7.11E+00)	<b>1.58E+01</b> (5.17E+00)	1.72E+01‡ (6.06E+00)	<b>1.24E+01</b> (3.68E+00)	2.69E+01‡ (1.05E+01)	<b>8.90E+00</b> (2.68E+00)
F19	6.68E+00‡ (6.53E-01)	<b>4.79E+00</b> (8.05E-01)	6.56E+00‡ (7.78E-01)	<b>4.85E+00</b> (6.62E-01)	6.34E+00‡ (5.72E-01)	<b>4.01E+00</b> (3.54E-01)	5.46E+00‡ (9.09E-01)	<b>3.77E+00</b> (6.94E-01)
F20	1.94E+01‡ (5.53E+00)	<b>1.46E+01</b> (4.62E+00)	2.21E+01‡ (6.24E+00)	<b>1.49E+01</b> (4.90E+00)	1.35E+01‡ (3.70E+00)	<b>1.05E+01</b> (3.21E+00)	1.21E+01‡ (4.08E+00)	<b>7.47E+00</b> (2.14E+00)
F21	1.66E+02§ (8.05E+01)	<b>1.54E+02</b> (8.19E+01)	1.76E+02§ (7.95E+01)	<b>1.60E+02</b> (7.68E+01)	1.65E+02‡ (9.29E+01)	<b>1.25E+02</b> (7.32E+01)	2.20E+02‡ (9.45E+01)	<b>1.00E+02</b> (7.45E+01)
F22	2.23E+02§ (8.38E+01)	<b>2.13E+02</b> (7.39E+01)	2.61E+02‡ (8.68E+01)	<b>2.09E+02</b> (8.03E+01)	2.28E+02‡ (8.73E+01)	<b>1.90E+02</b> (7.18E+01)	1.86E+02§ (7.43E+01)	<b>1.75E+02</b> (6.60E+01)
F23	3.15E+02‡ (1.14E-11)	<b>3.15E+02</b> (1.56E-09)	3.15E+02‡ (3.76E-05)	<b>3.15E+02</b> (2.64E-08)	3.15E+02‡ (1.43E-11)	<b>3.15E+02</b> (5.94E-13)	3.15E+02‡ (7.31E-04)	<b>3.15E+02</b> (3.98E-13)
F24	2.27E+02‡ (1.12E+00)	<b>2.26E+02</b> (8.16E-01)	2.27E+02‡ (1.31E+00)	<b>2.26E+02</b> (9.00E-01)	2.25E+02‡ (7.41E+00)	<b>2.25E+02</b> (7.30E-01)	<b>2.24E+02</b> † (8.27E-01)	2.25E+02 (7.61E-01)
F25	2.03E+02‡ (2.84E-01)	<b>2.03E+02</b> (2.53E-01)	2.03E+02‡ (3.19E-01)	<b>2.03E+02</b> (2.53E-01)	2.03E+02‡ (3.19E-01)	<b>2.03E+02</b> (1.15E-01)	2.03E+02‡ (3.78E-01)	<b>2.03E+02</b> (7.56E-02)
F26	1.00E+02‡ (4.21E-02)	<b>1.00E+02</b> (3.88E-02)	1.00E+02‡ (4.69E-02)	<b>1.00E+02</b> (3.73E-02)	1.00E+02‡ (4.97E-02)	<b>1.00E+02</b> (3.52E-02)	1.00E+02‡ (3.94E-02)	<b>1.00E+02</b> (3.80E-02)
F27	<b>3.93E+02</b> † (5.52E+01)	4.00E+02 (2.05E-01)	4.06E+02‡ (2.55E+00)	<b>4.00E+02</b> (1.34E-01)	4.05E+02‡ (2.70E+00)	<b>4.00E+02</b> (9.05E-02)	4.01E+02‡ (4.69E-01)	<b>4.00E+02</b> (1.49E-01)
F28	1.12E+03‡ (1.40E+02)	<b>9.26E+02</b> (3.63E+01)	1.25E+03‡ (2.03E+02)	<b>9.14E+02</b> (4.39E+01)	9.67E+02‡ (5.82E+01)	<b>8.74E+02</b> (3.98E+01)	8.59E+02§ (4.74E+01)	<b>8.44E+02</b> (2.42E+01)
F29	7.26E+02§ (5.81E+01)	<b>7.15E+02</b> (8.77E-01)	9.84E+02‡ (8.02E+01)	<b>7.06E+02</b> (6.90E+01)	7.15E+02‡ (8.53E-01)	<b>7.15E+02</b> (5.35E-01)	7.98E+02‡ (8.01E+01)	<b>7.15E+02</b> (1.43E+00)
F30	1.68E+03‡ (3.17E+02)	<b>7.35E+02</b> (1.63E+02)	1.96E+03‡ (3.75E+02)	<b>8.14E+02</b> (1.99E+02)	1.47E+03‡ (3.48E+02)	<b>7.34E+02</b> (2.53E+02)	1.46E+03‡ (3.71E+02)	<b>7.93E+02</b> (1.92E+02)
‡/§/†	20/6/4		23/3/4		26/1/3		22/4/4	

TABLE 4: The statistical results of the multiple-problem Wilcoxon test for TABL + ABCs vs. AEL + ABCs.

Algorithms at 30D 0.05	$R^+$	$R^-$	$p$ value
TABL + ABC vs. AEL + ABC Yes	429.0	36.0	$5.10E-05$
TABL + ABCM vs. AEL + ABCM Yes	393.0	42.0	$1.41E-04$
TABL + ABCVSS vs. AEL + ABCVSS Yes	350.0	85.0	$3.83E-03$
TABL + EABC vs. AEL + EABC Yes	368.5	66.5	$1.05E-03$
TABL + GABC vs. AEL + GABC Yes	367.0	68.0	$1.11E-03$
TABL + MABC vs. AEL + MABC Yes	440.0	25.0	$1.90E-05$
TABL + OPIABC vs. AEL + OPIABC Yes	415.0	20.0	$1.80E-05$
TABL + qABC vs. AEL + qABC Yes	430.0	35.0	$4.75E-05$

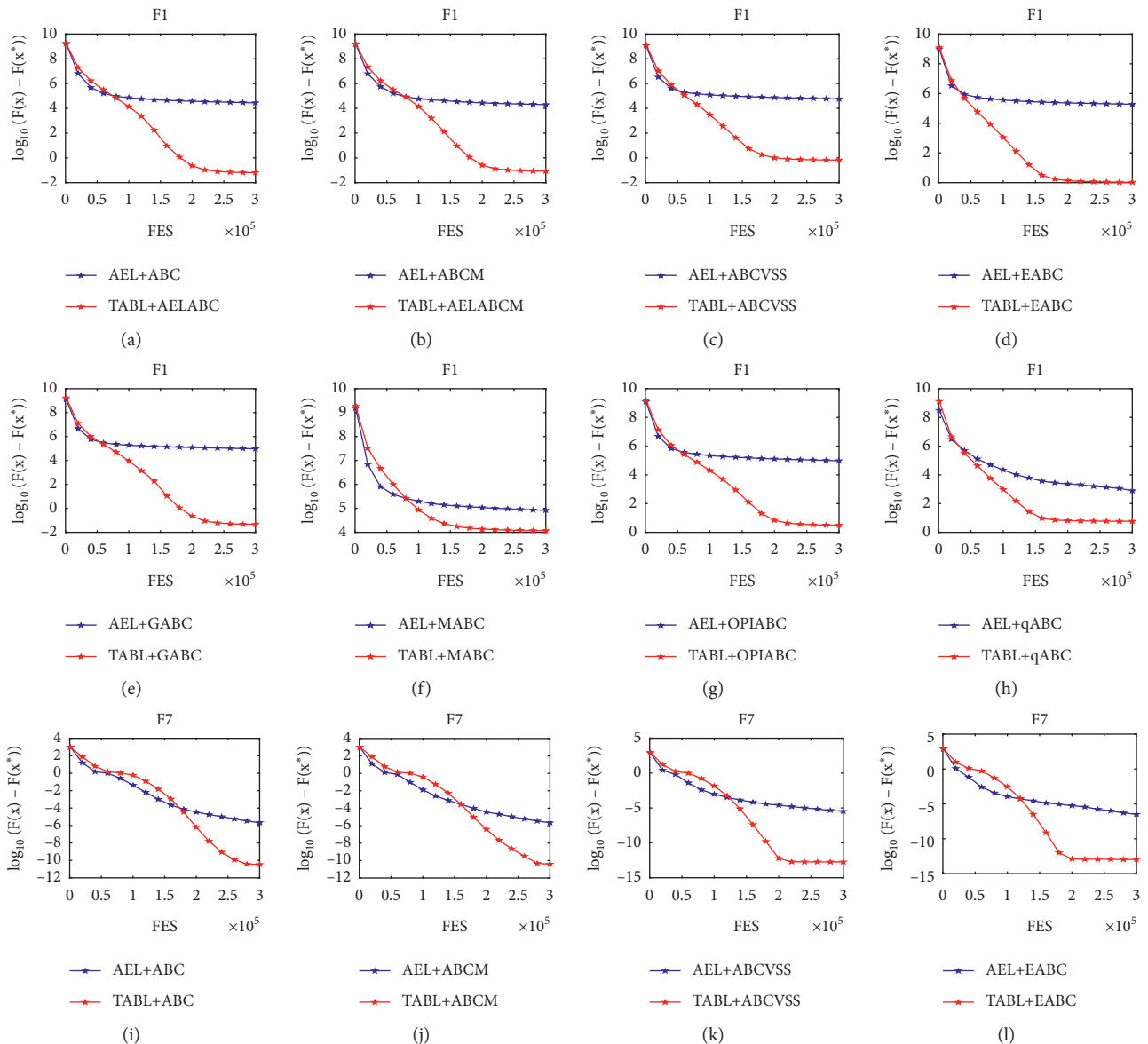


FIGURE 7: Continued.

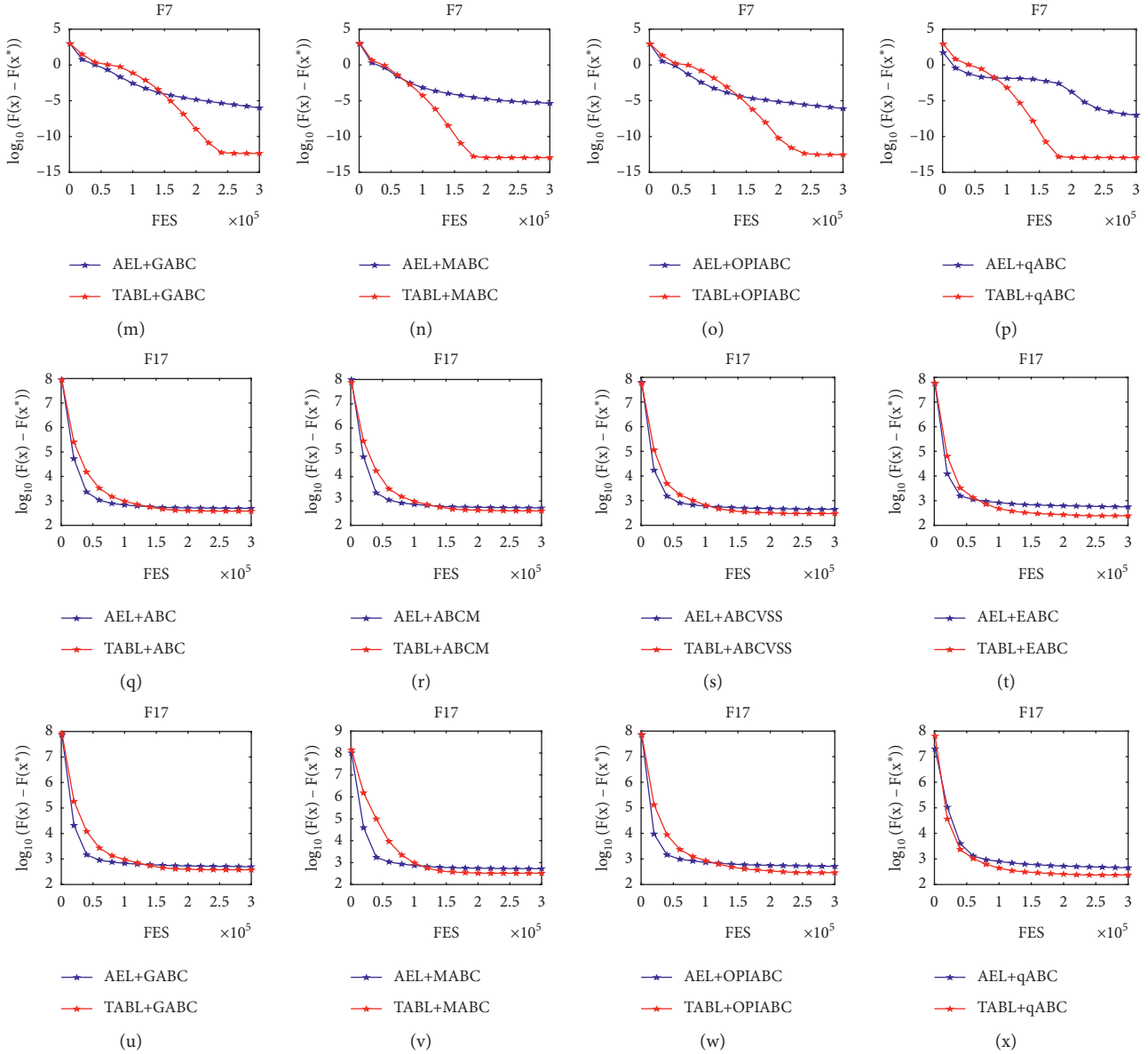


FIGURE 7: The mean function error values versus numbers of function evaluations for eight groups of AEL + ABCs vs. TABL + ABCs over 51 independent runs on F1, F7, and F17 with 30D, respectively.

costs of ABC mainly include the following three parts: (1) initialization phase ( $C_{IP}$ ), (2) employed bee phase ( $C_{EBP}$ ), and (3) onlooker bee phase ( $C_{OBP}$ ). Therefore, the total worst computation complexity of ABC in one generation is

$$\begin{aligned} C_{ABC} &= C_{IP} + C_{EBP} + C_{OBP} \\ &= O(SN \times D) + O(SN \times D) + O(SN \times D) \quad (10) \\ &= O(SN \times D). \end{aligned}$$

The computational costs of TABL + ABC mainly include the following three parts: (1) ABC ( $C_{ABC}$ ), (2) sorting fitness values ( $C_{SFV}$ ), and (3) Eigen decomposition of covariance

matrix ( $C_{CMED}$ ). Therefore, the total worst computation complexity of TABL + ABC in one generation is

$$\begin{aligned} C_{TABL+ABC} &= C_{ABC} + C_{SFV} + C_{CMDE} \\ &= O(SN \times D) + O(SN^2) + O(D^3). \quad (11) \end{aligned}$$

According to the above discussion, the main difference between TABL + ABC, AEL + ABC, and ACoS + ABC includes two aspects: (1) the AEL + ABC employed the covariance matrix, while the ACoS + ABC introduced the cumulative covariance matrix; (2) the TABL + ABC introduced the multivariable perturbation strategy as the scout bee phase. As the Eigen decomposition process of the cumulative covariance



TABLE 5: The statistical results (mean (Std)) of ACoS+ ABC vs. TABL+ ABC, ACoS+ ABCM vs. TABL+ ABCM, ACoS+ ABCVSS vs. TABL+ ABCVSS, and ACoS+ EABC vs. TABL+ EABC over 51 independent runs on the CEC2014 benchmarks with 30D.

Function	ACoS+ ABC	TABL+ ABC	ACoS+ ABCM	TABL+ ABCM	ACoS+ ABCVSS	TABL+ ABCVSS	ACoS+ EABC	TABL+ EABC
F1	3.10E+00‡ (1.27E+01)	6.53E-02 (1.28E-01)	2.39E+00‡ (1.39E+01)	8.57E-02 (1.64E-01)	1.31E+04‡ (3.33E+04)	6.63E-01 (1.79E+00)	6.70E+03‡ (1.52E+04)	1.06E+00 (1.87E+00)
F2	1.18E-08§ (4.33E-08)	1.79E-07 (5.12E-07)	8.18E-11† (1.56E-10)	4.85E-07 (3.12E-06)	1.11E-14† (1.40E-14)	6.52E-14 (9.73E-14)	5.57E-16† (3.98E-15)	2.10E-13 (9.93E-13)
F3	8.02E-14† (2.83E-14)	2.41E-10 (1.05E-09)	4.90E-14† (1.98E-14)	7.23E-12 (2.58E-11)	0.00E+00† (0.00E+00)	7.91E-14 (4.56E-14)	0.00E+00† (0.00E+00)	2.90E-14 (2.87E-14)
F4	1.86E+01‡ (2.91E+01)	8.28E+00 (2.11E+01)	1.71E+01‡ (3.04E+01)	1.11E+01 (2.35E+01)	8.22E+00‡ (2.00E+01)	6.98E+00 (1.35E+01)	3.51E+00† (1.37E+01)	1.42E+01 (2.60E+01)
F5	2.02E+01‡ (3.21E-02)	2.00E+01 (1.42E-04)	2.01E+01‡ (3.84E-02)	2.00E+01 (1.66E-03)	2.02E+01‡ (4.17E-02)	2.00E+01 (4.75E-05)	2.03E+01‡ (4.74E-02)	2.00E+01 (5.33E-04)
F6	1.44E+01‡ (1.55E+00)	1.06E+01 (1.90E+00)	1.54E+01‡ (1.54E+00)	1.11E+01 (1.60E+00)	1.34E+01‡ (1.29E+00)	9.13E+00 (1.84E+00)	1.36E+01‡ (1.24E+00)	6.25E+00 (2.20E+00)
F7	1.44E-06‡ (2.32E-06)	3.39E-11 (5.59E-11)	8.50E-07‡ (1.75E-06)	3.87E-11 (1.63E-10)	4.16E-06‡ (6.07E-06)	1.81E-13 (1.02E-13)	1.27E-05‡ (1.73E-05)	1.07E-13 (4.78E-14)
F8	1.14E-13† (0.00E+00)	4.66E-08 (2.29E-07)	1.83E-13† (6.05E-14)	1.95E-02 (2.05E-01)	0.00E+00† (0.00E+00)	7.36E-14 (5.49E-14)	0.00E+00† (0.00E+00)	8.92E-15 (3.09E-14)
F9	9.44E+01‡ (2.54E+01)	8.20E+01 (1.36E+01)	1.02E+02‡ (2.04E+01)	8.58E+01 (1.21E+01)	5.65E+01† (7.66E+00)	6.01E+01 (9.32E+00)	4.36E+01§ (6.09E+00)	4.28E+01 (3.55E+00)
F10	1.56E+00† (7.15E-01)	4.25E+00 (1.33E+00)	3.14E+00† (1.17E+00)	4.14E+00 (1.49E+00)	2.04E-02† (2.18E-02)	4.67E-02 (4.09E-02)	2.22E-01† (1.72E-01)	4.22E-01 (4.05E-01)
F11	1.97E+03‡ (2.61E+02)	1.83E+03 (2.02E+02)	2.18E+03‡ (3.53E+02)	1.80E+03 (2.25E+02)	1.78E+02§ (2.59E+02)	1.74E+03 (2.54E+02)	1.84E+03‡ (2.77E+02)	1.50E+03 (2.48E+02)
F12	1.99E-01‡ (3.57E-02)	9.96E-02 (2.35E-02)	2.23E-01‡ (3.93E-02)	9.39E-02 (2.23E-02)	2.73E-01‡ (5.11E-02)	8.29E-02 (2.65E-02)	3.30E-01‡ (6.47E-02)	1.02E-01 (3.31E-02)
F13	2.46E-01‡ (2.36E-02)	2.02E-01 (2.68E-02)	2.44E-01‡ (2.75E-02)	1.90E-01 (2.74E-02)	2.69E-01‡ (3.22E-02)	2.19E-01 (3.35E-02)	2.95E-01‡ (3.29E-02)	2.15E-01 (3.40E-02)
F14	1.78E-01§ (2.28E-02)	1.69E-01 (2.11E-02)	1.78E-01§ (2.27E-02)	1.84E-01 (2.28E-02)	1.93E-01‡ (1.96E-02)	1.75E-01 (2.68E-02)	1.90E-01§ (3.28E-02)	1.85E-01 (2.52E-02)
F15	7.66E+00‡ (1.03E+00)	4.96E+00 (9.35E-01)	7.90E+00‡ (1.41E+00)	4.97E+00 (8.34E-01)	6.19E+00‡ (8.71E-01)	3.67E+00 (7.56E-01)	6.38E+00‡ (8.62E-01)	3.35E+00 (5.89E-01)
F16	9.97E+00‡ (3.57E-01)	9.74E+00 (3.61E-01)	1.03E+01‡ (3.80E-01)	9.79E+00 (3.98E-01)	9.60E+00§ (4.17E-01)	9.50E+00 (3.43E-01)	9.43E+00‡ (3.87E-01)	8.93E+00 (3.76E-01)
F17	4.49E+02‡ (1.26E+02)	3.78E+02 (1.51E+02)	4.42E+02§ (1.32E+02)	3.93E+02 (1.56E+02)	3.62E+02‡ (1.61E+02)	3.00E+02 (2.56E+02)	3.51E+02‡ (1.47E+02)	2.43E+02 (1.18E+02)
F18	2.11E+01‡ (4.97E+00)	1.85E+01 (5.14E+00)	1.80E+01‡ (4.37E+00)	1.58E+01 (5.17E+00)	1.25E+01§ (3.36E+00)	1.24E+01 (3.68E+00)	1.24E+01‡ (3.02E+00)	8.90E+00 (2.68E+00)
F19	6.74E+00‡ (6.02E-01)	4.79E+00 (5.68E-01)	7.15E+00‡ (6.10E-01)	4.85E+00 (6.62E-01)	6.37E+00‡ (6.06E-01)	4.01E+00 (6.23E-01)	6.18E+00‡ (6.32E-01)	3.77E+00 (6.94E-01)
F20	1.95E+01‡ (5.20E+00)	1.46E+01 (4.62E+00)	1.87E+01‡ (4.46E+00)	1.49E+01 (4.90E+00)	1.29E+01‡ (3.67E+00)	1.05E+01 (2.57E+00)	1.28E+01‡ (2.85E+00)	7.47E+00 (2.14E+00)
F21	1.65E+02§ (6.92E+01)	1.54E+02 (8.19E+01)	1.59E+02§ (7.30E+01)	1.60E+02 (7.68E+01)	1.27E+02§ (7.61E+01)	1.25E+02 (7.32E+01)	1.12E+02§ (8.12E+01)	1.00E+02 (7.68E+01)
F22	2.25E+02§ (8.54E+01)	2.13E+02 (7.39E+01)	2.23E+02§ (8.06E+01)	2.09E+02 (8.03E+01)	2.01E+02§ (7.70E+01)	1.90E+02 (7.18E+01)	1.74E+02§ (6.19E+01)	1.75E+02 (6.60E+01)
F23	3.15E+02‡ (5.22E-13)	3.15E+02 (1.56E-09)	3.15E+02‡ (4.50E-13)	3.15E+02 (2.64E-08)	3.15E+02‡ (4.02E-13)	3.15E+02 (5.94E-13)	3.15E+02§ (7.20E-09)	3.15E+02 (3.98E-13)
F24	2.26E+02‡ (3.96E+00)	2.26E+02 (6.78E-01)	2.27E+02‡ (2.90E+00)	2.26E+02 (9.00E-01)	2.25E+02‡ (5.20E+00)	2.25E+02 (7.30E-01)	2.26E+02‡ (6.37E-01)	2.25E+02 (7.61E-01)
F25	2.03E+02‡ (3.05E-01)	2.03E+02 (2.53E-01)	2.03E+02‡ (3.41E-01)	2.03E+02 (2.53E-01)	2.03E+02‡ (1.44E-01)	2.03E+02 (1.15E-01)	2.03E+02‡ (1.19E-01)	2.03E+02 (7.83E-02)
F26	1.00E+02‡ (4.53E-02)	1.00E+02 (3.88E-02)	1.00E+02‡ (3.47E-02)	1.00E+02 (3.73E-02)	1.00E+02‡ (4.20E-02)	1.00E+02 (3.52E-02)	1.00E+02‡ (2.76E-02)	1.00E+02 (3.80E-02)
F27	4.01E+02‡ (8.62E-01)	4.00E+02 (2.54E-01)	3.93E+02† (5.42E+01)	4.00E+02 (1.34E-01)	4.02E+02‡ (1.55E+00)	4.00E+02 (9.05E-02)	4.00E+02‡ (8.06E-01)	4.00E+02 (1.49E-01)
F28	1.09E+03‡ (7.54E+01)	9.26E+02 (3.63E+01)	1.22E+03‡ (1.74E+02)	9.14E+02 (2.56E+01)	9.20E+02‡ (3.19E+01)	8.74E+02 (3.98E+01)	8.62E+02‡ (2.42E+01)	8.44E+02 (2.42E+01)
F29	7.15E+02‡ (9.38E-01)	7.15E+02 (8.77E-01)	7.15E+02§ (5.20E-01)	7.06E+02 (6.90E+01)	7.15E+02‡ (1.05E+00)	7.15E+02 (8.62E-01)	7.15E+02‡ (1.31E+00)	7.15E+02 (1.4300)
F30	1.27E+03‡ (2.38E+02)	7.35E+02 (1.63E+02)	1.39E+03‡ (2.27E+02)	8.14E+02 (1.99E+02)	1.32E+03‡ (2.19E+02)	7.34E+02 (1.77E+02)	1.24E+03‡ (2.13E+02)	7.93E+02 (1.92E+02)
‡/§/†	22/5/3		20/5/5		19/6/5		20/5/5	

TABLE 6: The statistical results (mean (std)) of ACoS + GABC vs. TABL + GABC, ACoS + MABC vs. TABL + MABC, ACoS + OPIABC vs. TABL + OPIABC, and ACoS + qABC vs. TABL + qABC over 51 independent runs on the CEC2014 benchmarks with 30D.

Function	ACoS + GABC	TABL + GABC	ACoS + MABC	TABL + MABC	ACoS + OPIABC	TABL + OPIABC	ACoS + qABC	TABL + qABC
F1	3.23E+02§ (1.65E+03)	<b>4.67E-02</b> (7.44E-02)	<b>6.07E+03†</b> (1.02E+04)	1.18E+04 (1.63E+04)	1.84E+04‡ (3.18E+04)	<b>3.17E+00</b> (8.25E+00)	6.66E+03‡ (7.54E+03)	<b>5.88E+00</b> (1.86E+01)
F2	<b>2.90E-14†</b> (3.98E-15)	1.73E-13 (4.39E-13)	2.62E-14§ (7.72E-15)	<b>2.45E-14</b> (9.88E-15)	<b>1.67E-15†</b> (6.75E-15)	4.12E-14 (1.91E-14)	4.06E+02‡ (1.78E+03)	<b>3.79E-14</b> (2.88E-14)
F3	<b>3.34E-14†</b> (2.83E-14)	7.69E-14 (3.38E-14)	5.24E-14‡ (1.54E-14)	<b>4.35E-14</b> (2.44E-14)	<b>2.23E-15†</b> (1.11E-14)	6.02E-14 (2.10E-14)	1.24E-07‡ (6.78E-07)	<b>2.56E-14</b> (2.86E-14)
F4	9.38E+00§ (2.19E+01)	<b>5.22E+00</b> (1.98E+01)	1.13E+01§ (1.89E+01)	<b>1.08E+01</b> (4.05E+01)	8.20E+00§ (1.95E+01)	<b>4.50E+00</b> (2.36E+01)	2.33E+01§ (5.12E+01)	1.18E+01 (3.25E+01)
F5	2.02E+01‡ (4.05E-02)	<b>2.00E+01</b> (8.36E-05)	2.02E+01‡ (3.10E-02)	<b>2.00E+01</b> (8.06E-04)	2.03E+01‡ (4.12E-02)	<b>2.00E+01</b> (1.77E-04)	2.02E+01‡ (4.03E-02)	<b>2.00E+01</b> (6.71E-04)
F6	1.36E+01‡ (1.43E+00)	<b>8.33E+00</b> (2.09E+00)	1.32E+01‡ (1.13E+00)	<b>7.16E+00</b> (2.37E+00)	1.26E+01‡ (1.66E+00)	<b>7.36E+00</b> (2.09E+00)	1.46E+01‡ (1.46E+00)	<b>6.85E+00</b> (1.81E+00)
F7	4.27E-07‡ (5.13E-07)	<b>4.37E-13</b> (3.19E-13)	1.26E-06‡ (1.70E-06)	<b>1.11E-13</b> (9.23E-14)	2.24E-07‡ (4.14E-07)	<b>2.96E-13</b> (2.88E-13)	7.83E-09‡ (1.95E-08)	<b>1.11E-13</b> (5.33E-14)
F8	<b>6.46E-14†</b> (5.69E-14)	1.25E-13 (4.10E-14)	1.05E-13‡ (3.09E-14)	<b>0.00E+00</b> (0.00E+00)	<b>0.00E+00†</b> (0.00E+00)	1.11E-14 (3.41E-14)	1.18E-13‡ (2.23E-14)	<b>2.23E-15</b> (1.59E-14)
F9	6.64E+01‡ (9.98E+00)	<b>5.41E+01</b> (8.05E+00)	6.23E+01‡ (9.65E+00)	<b>4.71E+01</b> (6.03E+00)	5.89E+01‡ (9.98E+00)	<b>4.92E+01</b> (8.76E+00)	9.51E+01‡ (1.30E+01)	<b>4.24E+01</b> (6.07E+00)
F10	<b>4.42E-01†</b> (3.81E-01)	1.64E+00 (1.06E+00)	<b>2.34E-01†</b> (4.05E-01)	4.59E-01 (8.20E-01)	6.82E-02‡ (3.44E-02)	<b>2.49E-02</b> (2.54E-02)	1.46E+00‡ (4.56E-01)	<b>1.47E-01</b> (4.70E-02)
F11	1.79E+03‡ (2.63E+02)	<b>1.63E+03</b> (2.27E+02)	1.80E+03‡ (2.74E+02)	<b>1.55E+03</b> (3.04E+02)	1.84E+03‡ (2.00E+02)	<b>1.69E+03</b> (2.39E+02)	2.02E+03‡ (2.69E+02)	<b>1.51E+03</b> (1.92E+02)
F12	1.95E-01‡ (3.84E-02)	<b>9.40E-02</b> (3.05E-02)	1.90E-01‡ (4.11E-02)	<b>7.73E-02</b> (2.89E-02)	2.93E-01‡ (5.12E-02)	<b>1.11E-01</b> (3.36E-02)	2.04E-01‡ (3.83E-02)	<b>1.09E-01</b> (4.56E-02)
F13	2.49E-01‡ (3.03E-02)	<b>2.32E-01</b> (3.45E-02)	2.43E-01§ (3.28E-02)	<b>2.31E-01</b> (4.05E-02)	2.59E-01‡ (3.55E-02)	<b>2.06E-01</b> (3.01E-02)	<b>1.89E-01†</b> (3.02E-02)	2.09E-01 (3.27E-02)
F14	<b>1.63E-01†</b> (2.37E-02)	1.84E-01 (2.50E-02)	<b>1.56E-01†</b> (3.05E-02)	2.20E-01 (2.82E-02)	<b>1.91E-01§</b> (2.56E-02)	1.96E-01 (2.10E-02)	<b>1.57E-01†</b> (1.82E-02)	1.76E-01 (2.39E-02)
F15	6.54E+00‡ (9.98E-01)	<b>4.22E+00</b> (8.05E-01)	5.82E+00‡ (9.55E-01)	<b>3.38E+00</b> (6.41E-01)	6.35E+00‡ (9.07E-01)	<b>4.16E+00</b> (7.83E-01)	6.39E+00‡ (1.15E+00)	<b>3.36E+00</b> (6.01E-01)
F16	9.61E+00‡ (5.57E-01)	<b>9.25E+00</b> (4.63E-01)	9.52E+00‡ (3.93E-01)	<b>8.93E+00</b> (5.41E-01)	9.80E+00‡ (3.26E-01)	<b>9.47E+00</b> (4.75E-01)	9.86E+00‡ (4.08E-01)	<b>8.78E+00</b> (5.46E-01)
F17	4.78E+02‡ (1.36E+02)	<b>3.77E+02</b> (1.70E+02)	4.22E+02‡ (2.06E+02)	<b>3.26E+02</b> (1.38E+02)	3.81E+02‡ (1.21E+02)	<b>2.90 E+02</b> (1.31E+02)	4.89E+02‡ (1.51E+02)	<b>2.35E+02</b> (2.04E+02)
F18	1.85E+01‡ (5.50E+00)	<b>1.32E+01</b> (3.90E+00)	2.01E+01‡ (8.72E+00)	<b>1.05E+01</b> (3.87E+00)	1.50E+01‡ (3.35E+00)	<b>1.05E+01</b> (2.66E+00)	2.45E+01‡ (6.36E+00)	<b>1.22E+01</b> (4.52E+00)
F19	6.48E+00‡ (7.25E-01)	<b>4.47E+00</b> (8.00E-01)	6.15E+00‡ (6.62E-01)	<b>3.74E+00</b> (6.61E-01)	6.33E+00‡ (5.44E-01)	<b>4.50E+00</b> (7.87E-01)	6.81E+00‡ (6.98E-01)	<b>3.94E+00</b> (5.80E-01)
F20	1.65E+01‡ (2.65E+00)	<b>1.05E+01</b> (3.38E+00)	1.48E+01‡ (5.60E+00)	<b>7.89E+00</b> (2.65E+00)	1.48E+01‡ (3.97E+00)	<b>9.89E+00</b> (3.13E+00)	1.98E+01‡ (5.93E+00)	<b>7.57E+00</b> (2.21E+00)
F21	1.61E+02§ (8.12E+01)	<b>1.36E+02</b> (9.63E+01)	1.58E+02§ (9.21E+01)	<b>1.47E+02</b> (5.65E+01)	1.41E+02‡ (7.17E+01)	<b>1.07 E+02</b> (6.89E+01)	1.77E+02‡ (8.60E+01)	<b>1.14E+02</b> (7.71E+01)
F22	2.02E+02§ (7.89E+01)	<b>1.75E+02</b> (9.62E+01)	1.91E+02‡ (8.25E+01)	<b>1.34E+02</b> (9.69E+01)	<b>1.93E+02§</b> (6.19E+01)	1.95E+02 (5.74E+01)	2.14E+02‡ (5.93E+01)	<b>1.72E+02</b> (6.42E+01)
F23	3.15E+02‡ (4.02E-13)	<b>3.15E+02</b> (5.03E-13)	3.15E+02§ (4.02E-13)	<b>3.15E+02</b> (3.98E-13)	3.15E+02§ (2.27E-10)	<b>3.15E+02</b> (3.98E-13)	3.15E+02‡ (5.57E-05)	<b>3.15E+02</b> (4.55E-13)
F24	2.26E+02‡ (4.96E+00)	<b>2.26E+02</b> (8.12E-01)	<b>2.24E+02§</b> (9.93E+00)	2.26E+02 (6.87E-01)	<b>2.22E+02†</b> (8.92E+00)	2.25E+02 (5.46E-01)	2.27E+02‡ (4.50E+00)	<b>2.25E+02</b> (9.02E-01)
F25	2.03E+02§ (1.75E-01)	<b>2.03E+02</b> (2.41E-01)	2.03E+02§ (2.35E-01)	<b>2.03E+02</b> (1.77E-01)	2.03E+02‡ (1.41E-01)	<b>2.03E+02</b> (2.02E-01)	2.03E+02‡ (3.47E-01)	<b>2.03E+02</b> (8.97E-02)
F26	1.00E+02‡ (4.18E-02)	<b>1.00E+02</b> (3.18E-02)	1.00E+02‡ (3.90E-02)	<b>1.00E+02</b> (3.14E-02)	1.00E+02‡ (3.82E-02)	<b>1.00E+02</b> (4.10E-02)	1.00E+02‡ (5.24E-02)	<b>1.00E+02</b> (2.69E-02)
F27	4.00E+02‡ (2.15E-01)	<b>4.00E+02</b> (6.54E-02)	<b>3.94E+02†</b> (5.53E+01)	4.00E+02 (1.95E-01)	4.01E+02‡ (1.34E+00)	<b>3.85E+02</b> (7.76E+01)	4.01E+02‡ (6.19E-01)	<b>4.00 E+02</b> (2.00E-01)
F28	9.18E+02‡ (6.00E+01)	<b>8.65E+02</b> (3.87E+01)	9.24E+02‡ (5.44E+01)	<b>8.42E+02</b> (2.69E+01)	9.09E+02‡ (4.24E+01)	<b>8.46E+02</b> (3.05E+01)	1.13E+03‡ (1.08E+02)	<b>8.43E+02</b> (1.88E+01)
F29	7.15E+02‡ (5.99E-01)	<b>7.05E+02</b> (6.65E+01)	<b>7.15E+02†</b> (1.43E+00)	7.41E+02 (3.65E+01)	7.16E+02‡ (1.02E+00)	<b>7.15E+02</b> (2.18E-01)	7.66E+02‡ (1.57E+02)	<b>7.15E+02</b> (1.18E+00)
F30	1.13E+03‡ (2.46E+02)	<b>8.13E+02</b> (1.88E+02)	1.15E+03§ (2.95E+02)	<b>1.08E+03</b> (2.39E+02)	1.22E+03‡ (2.01E+02)	<b>9.74E+02</b> (1.99E+02)	1.37E+03‡ (2.69E+02)	<b>8.03E+02</b> (1.96E+02)
‡/§/†	20/5/5		17/8/5		22/4/4		27/1/2	

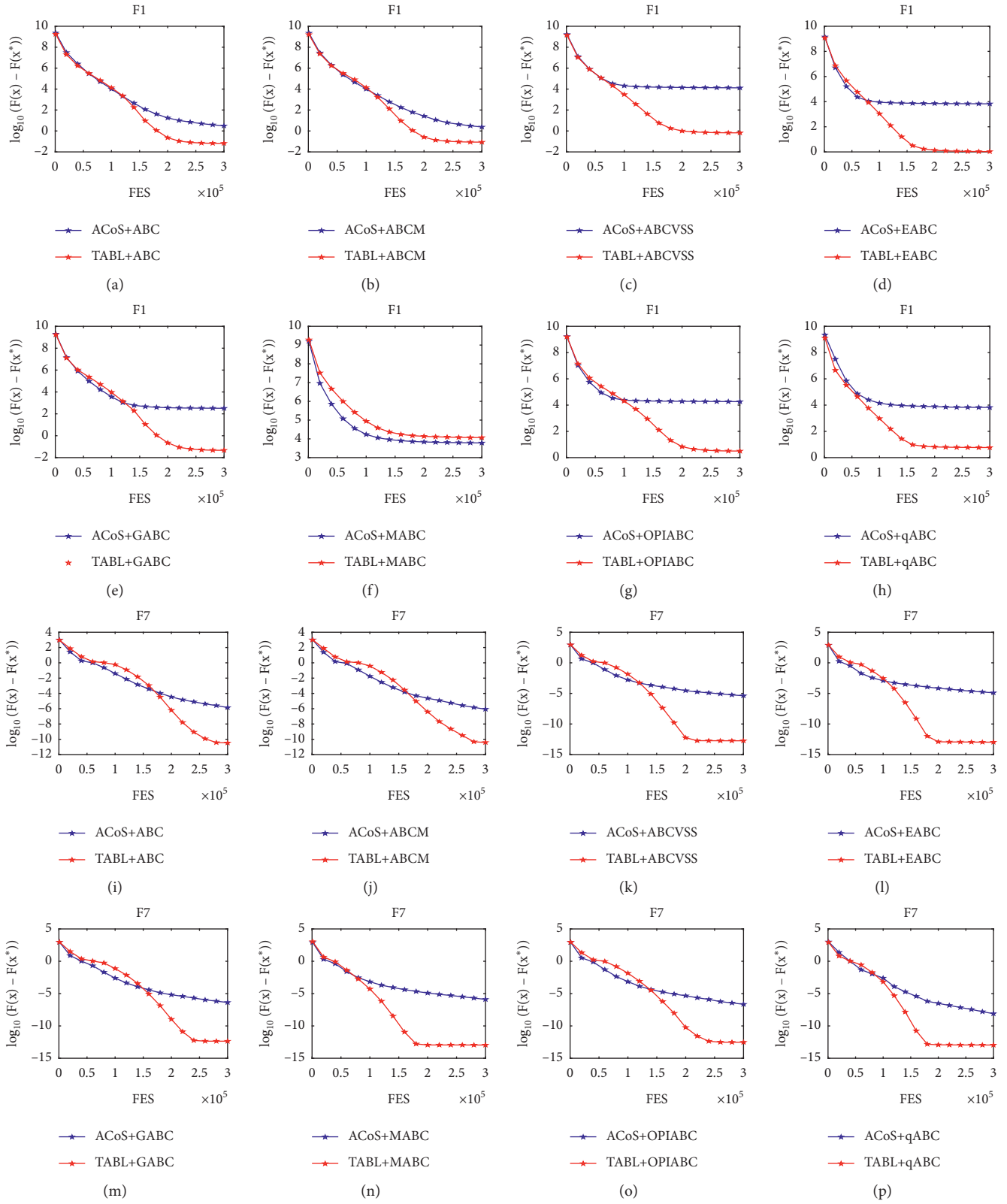


FIGURE 8: Continued.

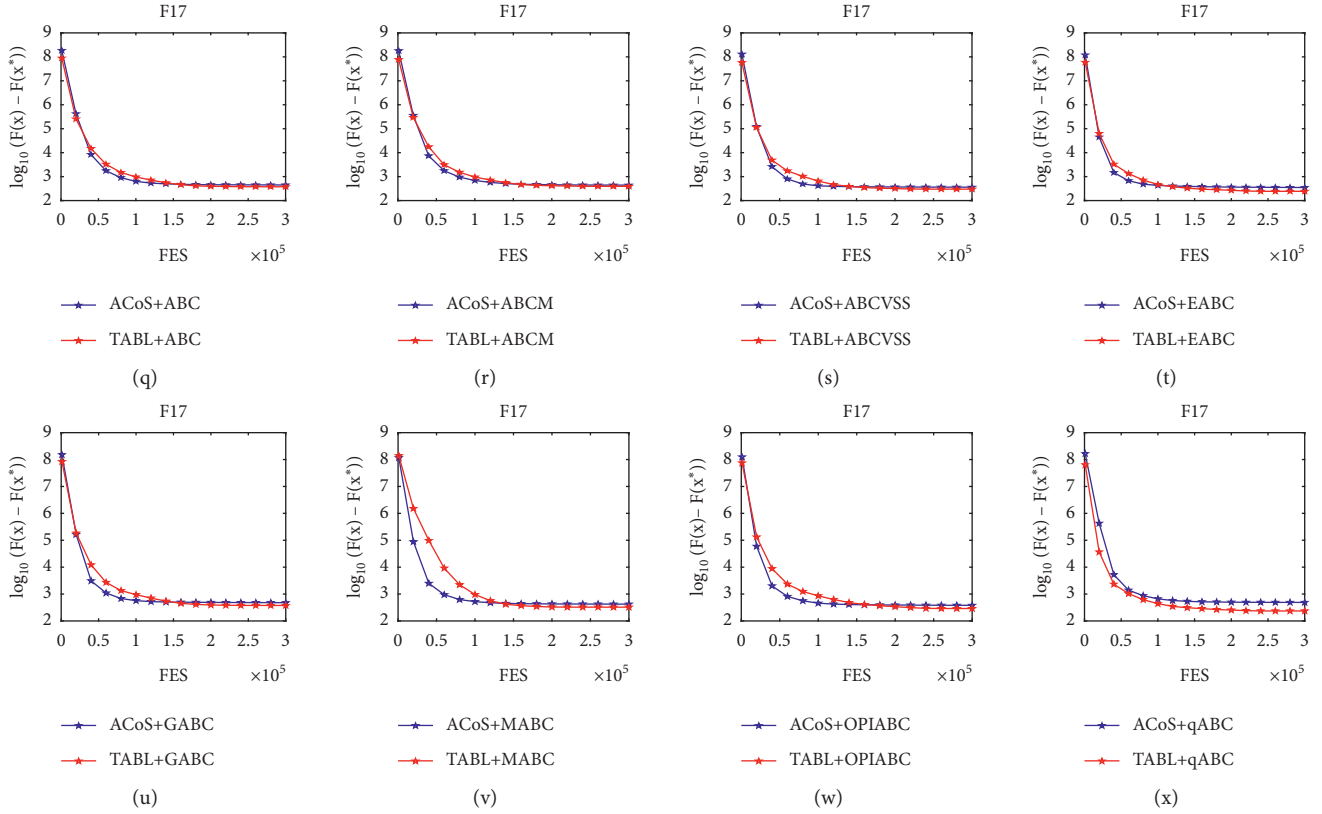


FIGURE 8: The mean function error values versus numbers of function evaluations for eight groups of ACoS + ABCs vs. TABL + ABCs over 51 independent runs on F1, F7, and F17 with 30D, respectively.

TABLE 7: The statistical results of the multiple-problem Wilcoxon test for TABL + ABCs vs. ACoS + ABCs.

Algorithms at 30D 0.05	$R^+$	$R^-$	$p$ value
TABL + ABC vs. ACoS + ABC	395.0	40.0	$1.19E-04$
Yes			
TABL + ABCM vs. ACoS + ABCM	366.0	69.0	$1.24E-03$
Yes			
TABL + ABCVSS vs. ACoS + ABCVSS	380.0	55.0	$3.94E-04$
Yes			
TABL + EABC vs. ACoS + EABC	361.5	73.5	$1.73E-03$
Yes			
TABL + GABC vs. ACoS + GABC	388.0	47.0	$2.18E-04$
Yes			
TABL + MABC vs. ACoS + MABC	331.5	103.5	$1.13E-02$
Yes			
TABL + OPIABC vs. ACoS + OPIABC	382.5	52.5	$3.45E-04$
Yes			
TABL + qABC vs. ACoS + qABC	420.5	14.5	$1.10E-04$
Yes			

matrix of ACoS + ABC is the same as the AEL + ABC, its computational complexity ( $C_{CCMED}$ ) is  $O(D^3)$ . In addition, the computational complexity of the multivariable perturbation strategy  $C_{MPS}$  is  $O(D)$ . Therefore, the total worst computation complexity of TABL + ABC in one generation is

$$\begin{aligned}
 C_{ABC} &= C_{ABC} + C_{SFV} + C_{CMDE} + C_{MPS} \\
 &= O(SN \times D) + O(SN^2) + O(D^3) \\
 &= O(SN \times D) + O(SN^2) + O(D^3) + O(D).
 \end{aligned}
 \tag{12}$$

TABLE 8: The average ranking of twenty-four algorithms based on the Friedman test, respectively.

Algorithms	Ranking
AEL + ABC	20.20
AEL + ABCM	20.65
AEL + ABCVSS	17.68
AEL + EABC	14.38
AEL + GABC	15.23
AEL + MABC	18.42
AEL + OPIABC	16.53
AEL + qABC	18.72
ACoS + ABC	19.52
ACoS + ABCM	19.35
ACoS + ABCVSS	14.68
ACoS + EABC	14.05
ACoS + GABC	15.25
ACoS + MABC	13.22
ACoS + OPIABC	15.25
ACoS + qABC	20.55
TABL + ABC	12.72
TABL + ABCM	13.13
TABL + ABCVSS	8.37
TABL + EABC	7.37
TABL + GABC	9.27
TABL + MABC	9.47
TABL + OPIABC	8.03
TABL + qABC	7.03

Based on the above discussion, the worst computation complexity of TABL+ABC is the same as that of the AEL+ABC and ACoS+ABC.

## 5. Discussion

In the previous experimental discussion, we have verified the effectiveness of the proposed TABL framework for ABCs. In this section, we further compare the proposed algorithms in terms of TABL+qABC with other state-of-the-art EAs. They are the CMM\_rcBBOg [56], NCS [57], TLBO [58], TSaABC [35], CSO [59], SLPSO [60], and CMA-ES [61]. The experimental platform for all compared algorithms is the same as that of TABL+qABC, and the parameter settings are suggested as the corresponding original literature studies. Table 9 summarizes the experimental results.

From Table 9, it can be seen that the TABL+qABC shows better performance than other seven compared algorithms. To be specific, the TABL+qABC outperforms the CMM\_rcBBOg, NCS, TLBO, TSaABC, CSO, SLPSO, and CMA-ES on 24, 26, 27, 25, 21, 24, and 24 functions, respectively. In contrast, the CMM\_rcBBOg, NCS, TLBO, TSaABC, CSO, SLPSO, and CMA-ES can beat the TABL+qABC on 3, 2, 2, 2, 8, 5, and 4 functions, respectively.

In addition, we note that the TABL+ABCs is not as good as CMA-ES in solving unimodal functions. The reason is that the covariance matrix of TABL+ABCs only utilizes the distribution information of the current population, while the CMA-ES employs the cumulative learning of population distribution information from the previous to current generations. Generally, the CMA-ES has faster guidance capability, which speeds up the convergence speed.

## 6. TABL + qABC for Big Optimization Problems

In this section, the proposed TABL+ABC algorithm is applied to big optimization problems. The results are compared with the other learning algorithms PSO, DE, and ABC.

*6.1. Big Optimization Problems.* In the optimization of big data 2015 competition [62], the big optimization problem is introduced. Next, we will briefly introduce this problem. Assume the dimension of matrix  $\mathbf{Y}$  is  $N \times M$ , where  $N$  and  $M$  are the number and length of interdependent time series, respectively. The same is true for matrix  $\mathbf{X}$ . A linear transformation matrix  $\mathbf{A}$  of  $N \times N$  is given. Then, we have

$$\mathbf{Y} = \mathbf{A}\mathbf{X}. \quad (13)$$

The main problem of big optimization problem is how to decompose the matrix  $\mathbf{X}$  into two matrices  $\mathbf{X}_1$  and  $\mathbf{X}_2$  as follows:

$$\begin{aligned} \mathbf{X} &= \mathbf{X}_1 + \mathbf{X}_2, \\ \mathbf{Y} &= \mathbf{A}\mathbf{X}_1 + \mathbf{A}\mathbf{X}_2. \end{aligned} \quad (14)$$

The Pearson correlation coefficient  $C$  between  $\mathbf{A}\mathbf{X}_1$  and  $\mathbf{A}\mathbf{X}_2$  is given as follows:

$$C = \frac{\text{cov}(\mathbf{Y}, \mathbf{A}\mathbf{X}_1)}{\delta(\mathbf{Y})\delta(\mathbf{A}\mathbf{X}_1)}, \quad (15)$$

where  $\text{cov}(\cdot)$  is the covariance matrix and  $\delta(\cdot)$  is the standard deviation.

The optimization objective can be defined as



TABLE 9: The statistical results (mean (std)) of CMM\_rcBBOg, NCS, TLBO, TSaABC, CSO, SLPSO, and CMA-ES over 51 independent runs on the CEC2014 benchmarks with 30D.

Function	TABL+qABC	CMM_rcBBOg	NCS	TLBO	TSaABC	CSO	SLPSO	CMA-ES
F1	5.88E+00 (1.86E+01)	1.50E+06‡ (1.13E+06)	7.86E+05‡ (2.98E+05)	2.88E+05‡ (3.85E+05)	1.80E+07‡ (6.00E+06)	5.19E+05‡ (2.22E+05)	3.87E+05‡ (2.41E+05)	1.25E-14† (4.62E-15)
F2	3.79E-14 (2.88E-14)	9.48E+03‡ (5.55E+03)	1.20E+02‡ (1.22E+02)	9.65E-01‡ (1.34E+00)	5.87E+02‡ (9.36E+02)	1.16E+04‡ (6.76E+03)	1.11E+04‡ (1.00E+04)	2.56E-14† (8.54E-15)
F3	2.56E-14 (2.86E-14)	6.31E+03‡ (5.14E+03)	9.09E+04‡ (1.65E+04)	5.59E+01‡ (7.67E+01)	7.66E+02‡ (7.78E+02)	7.17E+03‡ (5.37E+03)	6.71E+03‡ (5.44E+03)	5.57E-14† (1.79E-14)
F4	1.18E+01 (2.47E+01)	7.45E+01‡ (4.37E+01)	1.10E+01‡ (2.28E+01)	5.87E+01‡ (3.51E+01)	3.22E+01‡ (2.72E+01)	7.70E+01‡ (2.62E+01)	3.68E+01‡ (2.83E+01)	4.97E+00† (1.72E+01)
F5	2.00E+01 (6.71E-04)	2.00E+01§ (5.65E-05)	2.00E+01§ (1.25E-04)	2.09E+01‡ (5.79E-02)	2.03E+01‡ (5.23E-02)	2.09E+01‡ (5.04E-02)	2.09E+01‡ (5.38E-02)	2.00E+01§ (1.34E-05)
F6	6.85E+00 (1.81E+00)	6.68E+00† (2.67E+00)	1.20E+01‡ (1.61E+00)	1.55E+01‡ (2.31E+00)	1.27E+01‡ (1.54E+00)	8.05E-01† (9.67E-01)	9.42E-01† (1.19E+00)	4.16E+01‡ (1.07E+01)
F7	1.11E-13 (5.33E-14)	2.78E-02‡ (3.33E-02)	1.37E-02‡ (1.72E-02)	5.59E-02‡ (7.66E-02)	2.66E-06‡ (4.52E-06)	1.11E-14† (3.41E-14)	3.38E-04‡ (1.71E-03)	2.85E-03‡ (4.96E-03)
F8	2.23E-15 (1.59E-14)	3.16E+01‡ (8.74E+00)	9.35E+01‡ (1.72E+01)	6.75E+01‡ (1.26E+01)	8.69E-14‡ (4.87E-14)	1.13E+01‡ (2.64E+00)	1.73E+01‡ (4.73E+00)	4.32E+02‡ (7.82E+01)
F9	4.24E+01 (6.07E+00)	4.09E+01§ (1.15E+01)	9.98E+01‡ (1.66E+01)	7.34E+01‡ (1.54E+01)	3.92E+01§ (6.69E+00)	1.26E+01† (3.41E+00)	2.11E+01† (1.78E+01)	5.97E+02‡ (1.22E+02)
F10	1.47E-01 (4.70E-02)	1.21E+03‡ (4.30E+02)	2.22E+03‡ (2.56E+02)	1.58E+03‡ (3.99E+02)	2.77E-01‡ (2.59E-01)	1.08E+02‡ (1.26E+02)	3.55E+02‡ (2.28E+02)	5.05E+03‡ (8.26E+02)
F11	1.51E+03 (1.92E+02)	3.18E+03‡ (7.15E+02)	2.30E+03‡ (3.04E+02)	5.03E+03‡ (1.56E+03)	1.96E+03‡ (3.41E+02)	8.65E+02† (3.06E+02)	9.38E+02† (4.75E+02)	5.01E+03‡ (6.72E+02)
F12	1.09E-01 (3.39E-02)	9.73E-02† (6.67E-02)	6.39E-02† (1.83E-02)	2.45E+00‡ (2.27E-01)	3.77E-01‡ (7.04E-02)	1.69E-01‡ (4.52E-01)	2.22E+00‡ (5.60E-01)	2.86E-01‡ (3.06E-01)
F13	2.09E-01 (3.27E-02)	2.04E-01† (5.49E-02)	5.10E-01‡ (8.44E-02)	4.31E-01‡ (1.07E-01)	1.78E-01† (2.13E-02)	8.63E-02† (2.33E-02)	1.69E-01† (3.61E-02)	2.81E-01‡ (7.86E-02)
F14	1.76E-01 (2.39E-02)	2.78E-01‡ (5.14E-02)	2.21E-01‡ (3.21E-02)	2.71E-01‡ (8.61E-02)	2.15E-01‡ (1.77E-02)	3.82E-01‡ (4.70E-02)	3.94E-01‡ (7.05E-02)	3.98E-01‡ (1.88E-01)
F15	3.36E+00 (6.01E-01)	4.67E+00‡ (2.37E+00)	4.73E+00‡ (9.81E-01)	1.66E+01‡ (6.32E+00)	5.08E+00‡ (9.03E-01)	3.24E+00† (5.33E-01)	6.44E+00‡ (4.70E+00)	3.56E+00‡ (9.57E-01)
F16	8.78E+00 (5.46E-01)	1.22E+01‡ (5.32E-01)	1.20E+01‡ (4.49E-01)	1.13E+01‡ (4.89E-01)	9.56E+00‡ (4.42E-01)	7.88E+00† (9.40E-01)	1.21E+01‡ (2.91E-01)	1.43E+01‡ (4.42E-01)
F17	2.35E+02 (1.19E+02)	2.67E+05‡ (1.87E+05)	3.03E+04‡ (1.77E+04)	1.02E+05‡ (8.53E+04)	2.50E+06‡ (9.86E+05)	2.23E+05‡ (1.29E+05)	1.10E+05‡ (7.01E+04)	1.70E+03‡ (3.93E+02)
F18	1.22E+01 (4.52E+00)	8.88E+02‡ (1.15E+03)	7.36E+02‡ (4.92E+02)	2.87E+03‡ (5.08E+03)	5.51E+02‡ (5.99E+02)	7.31E+02‡ (8.25E+02)	1.41E+03‡ (1.76E+03)	1.55E+02‡ (5.20E+01)
F19	3.94E+00 (5.80E-01)	8.90E+00‡ (8.14E+00)	7.26E+00‡ (7.41E-01)	1.76E+01‡ (1.96E+01)	7.20E+00‡ (4.82E-01)	4.04E+00‡ (1.14E+00)	7.16E+00‡ (1.36E+00)	1.17E+01‡ (2.24E+00)
F20	7.57E+00 (2.21E+00)	1.63E+04‡ (8.23E+03)	1.67E+04‡ (8.05E+03)	3.83E+02‡ (1.40E+02)	2.92E+03‡ (1.47E+03)	1.47E+04‡ (7.21E+03)	2.20E+04‡ (1.17E+04)	2.75E+02‡ (1.10E+02)
F21	1.14E+02 (7.71E+01)	1.69E+05‡ (1.56E+05)	2.22E+04‡ (1.15E+04)	5.33E+04‡ (3.50E+04)	5.29E+05‡ (2.66E+05)	1.52E+05‡ (9.49E+04)	7.41E+04‡ (6.05E+04)	1.07E+03‡ (3.46E+02)
F22	1.72E+02 (6.42E+01)	3.50E+02‡ (1.55E+02)	2.59E+02‡ (1.08E+02)	2.70E+02‡ (9.84E+01)	2.86E+02‡ (1.02E+02)	1.77E+02‡ (6.44E+01)	1.81E+02‡ (1.26E+02)	3.89E+02‡ (2.54E+02)
F23	3.15E+02 (4.55E-13)	3.15E+02§ (1.32E-03)	3.16E+02‡ (1.00E-01)	3.15E+02§ (1.34E-11)	3.16E+02‡ (2.33E-01)	3.15E+02§ (1.11E-07)	3.15E+02§ (1.27E-12)	3.15E+02§ (5.08E-12)
F24	2.25E+02 (9.02E-01)	2.28E+02‡ (4.05E+00)	2.23E+02† (1.50E+01)	2.00E+02† (8.21E-04)	2.25E+02§ (3.78E+00)	2.26E+02‡ (3.29E+00)	2.31E+02‡ (6.53E+00)	2.55E+02‡ (1.55E+02)
F25	2.03E+02 (8.97E-02)	2.12E+02‡ (1.87E+00)	2.04E+02‡ (3.62E-01)	2.00E+02† (1.98E+00)	2.08E+02‡ (9.66E-01)	2.08E+02‡ (1.59E+00)	2.05E+02‡ (2.07E+00)	2.08E+02‡ (6.90E+00)
F26	1.00E+02 (2.69E-02)	1.10E+02‡ (3.00E+01)	1.01E+02‡ (9.61E-02)	1.20E+02‡ (3.99E+01)	1.00E+02§ (4.90E-02)	1.31E+02‡ (4.68E+01)	1.16E+02‡ (3.67E+01)	1.05E+02‡ (3.08E+01)
F27	4.00E+02 (2.00E-01)	4.31E+02‡ (5.07E+01)	4.02E+02‡ (3.28E-01)	5.62E+02‡ (1.75E+02)	4.15E+02‡ (4.38E+00)	3.66E+02† (5.31E+01)	3.93E+02† (5.62E+01)	4.72E+02‡ (1.11E+02)
F28	8.43E+02 (1.88E+01)	1.01E+03‡ (1.65E+02)	8.61E+02‡ (4.88E+01)	1.10E+03‡ (1.66E+02)	8.37E+02† (2.60E+01)	8.72E+02‡ (5.35E+01)	9.00E+02‡ (5.74E+01)	4.44E+03‡ (3.50E+03)
F29	7.15E+02 (1.18E+00)	1.47E+03‡ (2.50E+02)	4.05E+03‡ (1.47E+03)	9.42E+05‡ (2.89E+06)	1.63E+03‡ (2.42E+02)	1.42E+03‡ (2.87E+02)	1.65E+03‡ (5.57E+02)	4.91E+06‡ (5.60E+06)
F30	8.03E+02 (2.05E+02)	2.95E+03‡ (7.44E+02)	4.59E+03‡ (8.60E+02)	2.68E+03‡ (1.07E+03)	4.47E+03‡ (1.09E+03)	3.11E+03‡ (8.28E+02)	3.38E+03‡ (9.78E+02)	2.28E+03‡ (5.40E+02)
‡/§/†	--	24/3/3	26/2/2	27/1/2	25/3/2	21/1/8	24/1/5	24/2/4

TABLE 10: Description of the six datasets.

No.	# of sources	# of artifact	Noise free	Dimensions
1	4	2	D2	1024
2	12	6	D4	3076
3	19	6	D12	4864

TABLE 11: Experimental results of four algorithms.

Problem	PSO	DE	ABC	TABL + qABC
D4	7.56E + 00‡ (4.06E - 01)	5.25E + 00‡ (8.02E - 01)	4.58E + 00‡ (2.36E - 02)	<b>4.05E + 00</b> (2.25E - 01)
D12	7.09E + 00‡ (2.23E - 01)	2.34E + 01‡ (3.25E - 01)	5.82E + 00‡ (1.42E - 01)	<b>5.72E + 00</b> (2.54E - 01)
D19	<b>7.82E + 00‡</b> (3.72E - 01)	2.34E + 01‡ (5.34E - 01)	9.23E + 00‡ (1.26E - 01)	8.95E + 00 (3.25E - 01)
‡/\$/†	2/0/1	3/0/0	3/0/0	--

$$\min f = \frac{1}{N \cdot M} \sum_{i,j} (X_{i,j} - X_1^{i,j})^2 + \frac{1}{N^2 - N} \sum_{i,j \neq i} C_{i,j}^2 + \frac{1}{N} \sum (1 - C_{i,i})^2. \quad (16)$$

6.2. *Experimental Results.* To verify the effectiveness of the proposed TABL + qABC algorithm, six datasets as test problems are introduced. They are shown in Table 10, and the experimental results are given in Table 11.

Table 11 shows that TABL + qABC achieves the best results on D4, D12, and D19 problems, which suggested that the TABL + qABC is more suitable for solving high-dimensional optimization problems.

## 7. Conclusions

In this paper, we have discussed the search behavior of ABC and confirmed that ABC always works perfectly for separable problems but suffers a drastic performance loss for nonseparable problems. Based on this analysis, a tristage adaptive biased learning framework is proposed to enhance the performance of ABCs on nonseparable problems, termed TABL + ABCs. In TABL + ABCs, the ranking biased information of the parent food sources is served as the search direction to accelerate the convergence speed in employed bee stage, while in onlooker bee stage, the search direction is guided by turning the coordinate system for increasing the improvement interval of variables. Moreover, a deletion-restart learning strategy is designed in scout bee stage to prevent the potential risk of population stagnation. Finally, we compared TABL + ABCs with AEL + AELs and ACoS + ABCs by applying them to 30 CEC2014 test problems with 30D, respectively. The experimental results show that TABL + ABCs perform better than the compared algorithms in most cases.

Although TABL + ABCs can effectively solve these benchmark problems with 30D, like many evolutionary algorithms, they still suffer from the curse of dimensionality. Therefore, it needs to study the TABL strategy in depth to

meet the requirements of large-scale optimization in the future. In addition, establishing a bridge between the benchmark problems and real-world application problems is of great significance to the generalization of TABL + ABCs.

## Data Availability

The data used to support the findings of this study are available from the first author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The authors wish to thank the partial support of the National Natural Science Foundation of China under grant 61803301 and 61773314, the Key Project of Shaanxi Key Research and Development Program under grant 2020ZDLGR07-06, and the Doctoral Foundation of Xi'an University of Technology (112-256081812).

## References

- [1] R. Salomon, "Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms," *Biosystems*, vol. 39, no. 3, pp. 263–278, 1996.
- [2] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [3] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.
- [4] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [5] X. Zhao, J. Ji, and X. Wang, "Dynamic brain functional parcellation via sliding window and artificial bee colony

- algorithm,” *Applied Intelligence*, vol. 49, no. 5, pp. 1748–1770, 2019.
- [6] Y. Zhao, H. Liu, and K. Gao, “An evacuation simulation method based on an improved artificial bee colony algorithm and a social force model,” *Applied Intelligence*, vol. 51, no. 1, pp. 100–123, 2021.
- [7] F. Boudardara and B. Gorkemli, “Solving artificial ant problem using two artificial bee colony programming versions,” *Applied Intelligence*, vol. 50, no. 11, pp. 3695–3717, 2020.
- [8] Y. Zhang, C. L. He, X. F. Song, and X. Y. Sun, “A multi-strategy integrated multi-objective artificial bee colony for unsupervised band selection of hyperspectral images,” *Swarm and Evolutionary Computation*, vol. 60, Article ID 100806, 2021.
- [9] S. Xiao, H. Wang, W. Wang, Z. Huang, X. Zhou, and M. Xu, “Artificial bee colony algorithm based on adaptive neighborhood search and Gaussian perturbation,” *Applied Soft Computing*, vol. 100, p. 106955, 2021.
- [10] G. Zhu and S. Kwong, “Gbest-guided artificial bee colony algorithm for numerical function optimization,” *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166–3173, 2010.
- [11] W.-f. Gao and S.-Y. Liu, “A modified artificial bee colony algorithm,” *Computers & Operations Research*, vol. 39, no. 3, pp. 687–697, 2012.
- [12] X. Zhang and S. Y. Yuen, “Improving artificial bee colony with one-position inheritance mechanism,” *Memetic Computing*, vol. 5, no. 3, pp. 187–211, 2013.
- [13] J. Luo, Q. Wang, and X. Xiao, “A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization,” *Applied Mathematics and Computation*, vol. 219, no. 20, pp. 10253–10262, 2013.
- [14] S. Das, S. Biswas, and S. Kundu, “Synergizing fitness learning with proximity-based food source selection in artificial bee colony algorithm for numerical optimization,” *Applied Soft Computing*, vol. 13, no. 12, pp. 4676–4694, 2013.
- [15] D. Karaboga and B. Gorkemli, “A quick artificial bee colony (qABC) algorithm and its performance on optimization problems,” *Applied Soft Computing*, vol. 23, pp. 227–238, 2014.
- [16] W. Gao, F. T. S. Chan, L. Huang, and S. Liu, “Bare bones artificial bee colony algorithm with parameter adaptation and fitness-based neighborhood,” *Information Sciences*, vol. 316, pp. 180–200, 2015.
- [17] X. Li and G. Yang, “Artificial bee colony algorithm with memory,” *Applied Soft Computing*, vol. 41, pp. 362–372, 2016.
- [18] L. Cui, G. Li, X. Wang et al., “A ranking-based adaptive artificial bee colony algorithm for global numerical optimization,” *Information Sciences*, vol. 417, pp. 169–185, 2017.
- [19] W.-l. Xiang, X.-l. Meng, Y.-z. Li, R.-c. He, and M.-q. An, “An improved artificial bee colony algorithm based on the gravity model,” *Information Sciences*, vol. 429, pp. 49–71, 2018.
- [20] D. Kumar and K. K. Mishra, “Co-variance guided artificial bee colony,” *Applied Soft Computing*, vol. 70, pp. 86–107, 2018.
- [21] S. Aslan, H. Badem, and D. Karaboga, “Improved quick artificial bee colony (iqABC) algorithm for global optimization,” *Soft Computing*, vol. 23, no. 24, pp. 13161–13182, 2019.
- [22] J. Ji, S. Song, C. Tang, S. Gao, Z. Tang, and Y. Todo, “An artificial bee colony algorithm search guided by scale-free networks,” *Information Sciences*, vol. 473, pp. 142–165, 2019.
- [23] M. A. Awadallah, M. A. Al-Betar, A. L. a. Bolaji, E. M. Alsukhni, and H. Al-Zoubi, “Natural selection methods for artificial bee colony with new versions of onlooker bee,” *Soft Computing*, vol. 23, no. 15, pp. 6455–6494, 2019.
- [24] D. Bajer and B. Zorić, “An effective refined artificial bee colony algorithm for numerical optimisation,” *Information Sciences*, vol. 504, pp. 221–275, 2019.
- [25] W.-f. Gao, S.-y. Liu, and L.-l. Huang, “Enhancing artificial bee colony algorithm using more information-based search equations,” *Information Sciences*, vol. 270, pp. 112–133, 2014.
- [26] S. Biswas, S. Das, S. Debchoudhury, and S. Kundu, “Co-evolving bee colonies by forager migration: a multi-swarm based artificial bee colony algorithm for global search space,” *Applied Mathematics and Computation*, vol. 232, pp. 216–234, 2014.
- [27] H. Wang, Z. Wu, S. Rahnamayan, H. Sun, Y. Liu, and J.-s. Pan, “Multi-strategy ensemble artificial bee colony algorithm,” *Information Sciences*, vol. 279, pp. 587–603, 2014.
- [28] M. S. Kiran and O. Flndlk, “A directed artificial bee colony algorithm,” *Applied Soft Computing*, vol. 26, pp. 454–462, 2015.
- [29] W.-f. Gao, L.-l. Huang, S.-y. Liu, F. T. S. Chan, C. Dai, and X. Shan, “Artificial bee colony algorithm with multiple search strategies,” *Applied Mathematics and Computation*, vol. 271, pp. 269–287, 2015.
- [30] M. S. Kiran, H. Hakli, M. Gunduz, and H. Uguz, “Artificial bee colony algorithm with variable search strategy for continuous optimization,” *Information Sciences*, vol. 300, pp. 140–157, 2015.
- [31] W. F. Gao, L. L. Huang, S. Y. Liu, and C. Dai, “Artificial bee colony algorithm based on information learning,” *IEEE Transactions on Cybernetics*, vol. 45, no. 12, pp. 2827–2839, 2015.
- [32] S. K. Nseef, S. Abdullah, A. Turkey, and G. Kendall, “An adaptive multi-population artificial bee colony algorithm for dynamic optimisation problems,” *Knowledge-Based Systems*, vol. 104, pp. 14–23, 2016.
- [33] F. Harfouchi and H. Habbi, “A cooperative learning artificial bee colony algorithm with multiple search mechanisms,” *International Journal of Hybrid Intelligent Systems*, vol. 13, no. 2, pp. 113–124, 2016.
- [34] F. Harfouchi, H. Habbi, C. Ozturk, and D. Karaboga, “Modified multiple search cooperative foraging strategy for improved artificial bee colony optimization with robustness analysis,” *Soft Computing*, vol. 22, no. 19, pp. 6371–6394, 2018.
- [35] X. Song, M. Zhao, Q. Yan, and S. Xing, “A high-efficiency adaptive artificial bee colony algorithm using two strategies for continuous optimization,” *Swarm and Evolutionary Computation*, vol. 50, p. 100549, 2019.
- [36] X. Chen, H. Tianfield, and K. Li, “Self-adaptive differential artificial bee colony algorithm for global optimization problems,” *Swarm and Evolutionary Computation*, vol. 45, pp. 70–91, 2019.
- [37] J. Zhou, X. Yao, F. T. S. Chan et al., “An individual dependent multi-colony artificial bee colony algorithm,” *Information Sciences*, vol. 485, pp. 114–140, 2019.
- [38] W. Gao, Z. Wei, Y. Luo, and J. Cao, “Artificial bee colony algorithm based on Parzen window method,” *Applied Soft Computing*, vol. 74, pp. 679–692, 2019.
- [39] G. Yavuz and D. Aydın, “Improved self-adaptive search equation-based artificial bee colony algorithm with competitive local search strategy,” *Swarm and Evolutionary Computation*, vol. 51, Article ID 100582, 2019.
- [40] X. Song, M. Zhao, and S. Xing, “A multi-strategy fusion artificial bee colony algorithm with small population,” *Expert Systems with Applications*, vol. 142, Article ID 112921, 2020.

- [41] W.-F. Gao, L.-L. Huang, J. Wang, S.-Y. Liu, and C.-D. Qin, "Enhanced artificial bee colony algorithm through differential evolution," *Applied Soft Computing*, vol. 48, pp. 137–150, 2016.
- [42] Z. Liang, K. Hu, Q. Zhu, and Z. Zhu, "An enhanced artificial bee colony algorithm with adaptive differential operators," *Applied Soft Computing*, vol. 58, pp. 480–494, 2017.
- [43] W. A. H. M. Ghanem and A. Jantan, "Hybridizing artificial bee colony with monarch butterfly optimization for numerical optimization problems," *Neural Computing & Applications*, vol. 30, no. 1, pp. 163–181, 2018.
- [44] H. Wang and J.-H. Yi, "An improved optimization method based on krill herd and artificial bee colony with information exchange," *Memetic Computing*, vol. 10, no. 2, pp. 177–198, 2018.
- [45] X. Chen, B. Xu, C. Mei, Y. Ding, and K. Li, "Teaching-learning-based artificial bee colony for solar photovoltaic parameter estimation," *Applied Energy*, vol. 212, pp. 1578–1588, 2018.
- [46] X. Chen, X. Wei, G. Yang, and W. Du, "Fireworks explosion based artificial bee colony for numerical optimization," *Knowledge-Based Systems*, vol. 188, p. 105002, 2020.
- [47] F. Kang, J. Li, and Z. Ma, "Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions," *Information Sciences*, vol. 181, no. 16, pp. 3508–3531, 2011.
- [48] F. Kang, J. Li, and H. Li, "Artificial bee colony algorithm and pattern search hybridized for global optimization," *Applied Soft Computing*, vol. 13, no. 4, pp. 1781–1791, 2013.
- [49] W.-F. Gao, S.-Y. Liu, and L.-L. Huang, "A novel artificial bee colony algorithm with Powell's method," *Applied Soft Computing*, vol. 13, no. 9, pp. 3763–3775, 2013.
- [50] W. F. Wei-feng Gao, S. Y. San-yang Liu, and L. L. Ling-ling Huang, "A novel artificial bee colony algorithm based on modified search equation and orthogonal learning," *IEEE Transactions on Cybernetics*, vol. 43, no. 3, pp. 1011–1024, 2013.
- [51] J. Yang, Q. Jiang, L. Wang et al., "An adaptive encoding learning for artificial bee colony algorithms," *Journal of Computational Science*, vol. 30, pp. 11–27, 2019.
- [52] Q. Y. Jiang, J. Cui, and Y. Ma, "Improved adaptive encoding learning for artificial bee colony algorithms," *Applied Intelligence*, 2021.
- [53] J. J. Liang, B. Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC2014 special session and competition on single objective real-parameter numerical optimization," Technical Report 201311, Computational Intelligence Zhenzhou University, Zhenzhou, China, 2013.
- [54] J. Alcalá-Fdez, L. Sánchez, S. García et al., "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Computing*, vol. 13, no. 3, pp. 307–318, 2009.
- [55] Z.-Z. Liu, Y. Wang, S. Yang, and K. Tang, "An adaptive framework to tune the coordinate systems in nature-inspired optimization algorithms," *IEEE Transactions on Cybernetics*, vol. 49, no. 4, pp. 1403–1416, 2019.
- [56] X. Chen, H. Tianfield, W. Du, and G. Liu, "Biogeography-based optimization with covariance matrix based migration," *Applied Soft Computing*, vol. 45, pp. 71–85, 2016.
- [57] K. Tang, P. Yang, and X. Yao, "Negatively correlated search," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 3, pp. 1–9, 2016.
- [58] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, 2011.
- [59] R. Ran Cheng and Y. C. Yaochu Jin, "A competitive swarm optimizer for large scale optimization," *IEEE Transactions on Cybernetics*, vol. 45, no. 2, pp. 191–204, 2015.
- [60] R. Cheng and Y. Jin, "A social learning particle swarm optimization algorithm for scalable optimization," *Information Science*, vol. 291, pp. 43–60, 2015.
- [61] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.
- [62] S. K. Goh, H. A. Abbass, and K. C. Tan, "Evolutionary big optimization (BigOpt) of signals," in *Proceedings of the IEEE Congress on Evolutionary Computation*, May 2015.