IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 21, 2018, accepted July 18, 2018, date of publication August 6, 2018, date of current version September 5, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2861760

Elitism and Distance Strategy for Selection of Evolutionary Algorithms

HAIMING DU[1], ZAICHAO WANG[1], WEI ZHAN[2], AND JINYI GUO[3]
[1]School of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[2]School of Computer Science, Yangtze University, Jingzhou 434023, China
[3]School of Computer Science, China University of Geosciences (Wuhan), Wuhan 430074, China

Corresponding author: Wei Zhan (zhanwei814@yangtzeu.edu.cn)

This work was supported in part by the Zhengzhou University of Light Industry through the Doctoral Fund under Grant 2013BSJJ026, in part by the Backbone Teacher Fund under Grant 2013XGGJS011, in part by the Science and Technology Fund of Henan Provincial Education Department, China, under Grant 14B510022, in part by the Project for Foundations and Frontiers Research of Henan Province, China, under Grant 112300410144, in part by the Young Talent Project of Hubei Provence, Department of Education, China, under Grant Q20131209, and in part by the National Natural Science Foundation of China under Grant 51541408.

ABSTRACT Evolutionary algorithms (EAs) have been applied successfully in many fields. However, EAs cannot find an optimal solution on many occasions because the balance between exploration and exploitation is lost in runs. So far, tricking the balance is an important research topic in the field of evolutionary computation. Elitism strategy is a typical scheme applied in selection for the above purpose and can be widely used in different EAs. In this paper, we propose elitism and distance strategy based on the elitism strategy. According to our strategy, elites are still kept in selection for reducing genetic drift. Meanwhile, the individual among candidates for selection having the longest distance to each elite is also kept for maintaining diversity. We carry out experiments based on not only a genetic algorithm for the traveling salesman problem but also two differential evolution algorithms, DE/rand/2/bin and CoBiDE. Experimental results show that adding our strategy in all generations can significantly improve solutions of the genetic algorithm for the traveling salesman problem. Moreover, calling our strategy at a low probability can significantly improve solutions of DE/rand/2/bin, while calling the strategy based on our proposed adaptive scheme can statistically improve solutions of CoBiDE, a state-of-the-art differential evolution algorithm.

INDEX TERMS Elitism, distance, evolutionary algorithm, diversity.

I. INTRODUCTION

Evolutionary algorithms (EAs) include genetic algorithm, genetic programming, evolutionary programming, evolution strategy, differential evolution, etc. Such optimization techniques have distinct advantages as below [1].

- Concept simplicity
- Broad applicability
- Potential to use knowledge and hybridize with other methods
- Prone to parallelism
- Robust to dynamic changes
- Capability for self-optimization
- Able to solve problems that have no known solutions

EAs are based on populations consisting of μ-encoded tentative solutions, individuals. In each generation, individuals are manipulated competitively by some variation operators, mutation, crossover, selection, etc. In detail, new individuals, offspring, are produced by mutation and crossover based on parents, the individuals of the current generation. In selection, parents and offspring are all candidates of the next generation. After generations of evolution, satisfactory solutions may be obtained.

In fact, evolutionary algorithms needs to address the exploration and exploitation of a search space in runs. Exploration is the process of visiting entirely new regions of a search space, whilst exploitation is the process of visiting entirely new regions of a search space within the neighborhood of previously visited points [2]. On many occasions, an optimal solution cannot be found by an EA because the balance between exploration and exploitation is lost in runs. In short, tricking the balance in evolutionary algorithms is important.

Compared to crossover or mutation, selection is not closely related to chromosome representation. Thus, it is possible that a scheme applied in selection can be widely used in different EAs. Elitism strategy is a typical example.

Elitism strategy provides a means for reducing genetic drift by ensuring that the most fitting individuals among candidates for selection, elites, are allowed to copy their traits to the next generation [3]. In evolution, such a strategy can add selective pressure and improve convergence speed [4]. Thus, it is usually employed for exploitation [5]. Elitism strategy has been used widely in different EAs. In most differential evolution (DE) algorithms, elitism strategy is employed by default. Also, recent applications of elitism strategy can be found in [6]–[10].

Now that elitism strategy has been widely used in EAs, an improved strategy based on it taking exploration into consideration may be a better method for maintaining the balance between exploration and exploitation. Based on the above motivation, we propose elitism and distance strategy. According to our strategy, elites are still kept for reducing genetic drift. Meanwhile, candidates for selection having the longest distance to each elite is also kept for maintaining diversity.

Three EAs are involved in our experiments. The first EA is a genetic algorithm (GA) for the Traveling Salesman Problem (TSP) showing severe lack of diversity in runs. In this situation, our strategy is called in every generations for maintaining diversity. The second one is a basic Differential Evolution (DE) algorithm, DE/rand/2/bin. Its runs for different functions show different diversity changing trend. In this situation, our strategy is called at a probability for fine tuning the ratio of exploration and exploitation. More importantly, the third EA is a state-of-the-art DE algorithm, CoBiDE [11]. According to a phenomenon reflected by the diversity changing trend and the fitness changing trend in runs, we propose an adaptive scheme for calling our strategy. The results of our experiments show that the proposed strategy can help to improve solutions.

The rest of this paper is organized as follows. Section II introduces existing approaches for the balance between exploration and exploitation used in selection as related work. Section III gives our proposed elitism and distance strategy. Section IV goes our experiment. Finally, a conclusion and a prospect are dealt with in Section V.

II. RELATED WORKS

So far, some approaches for the balance between exploration and exploitation in evolutionary computation have been proposed in literature. Among them, the approaches embedded in selection can be classified into simple approaches and complex ones. Simple approaches are mainly based on fitness, while complex ones consider distance or something else besides fitness. Details are given below.

- Simple approaches
  As above mentioned, elitism strategy is a widely used method. In rank selection, the selected probability of candidates are given based on their rank of fitness [12]. In some EAs, such as the series of DE [13], each offspring only competes with one of its parents. Matsui [14] proposed correlative family-based selection. Here, among two parents and their two offspring, the most befitting individual and the individual having the maximum Hamming distance to the former are selected. Hutter and Legg [15] proposed that candidates are divided into several classes according to fitness, and each class has an equal opportunity of survival. Chen *et al.* [16] proposed a selection scheme which deliberately selects candidates with relatively low fitness. Inspired by simulated annealing, Mori *et al.* [17] proposed thermodynamic genetic algorithm (TDGA) borrowing the concepts of temperature and entropy.
- Complex Approaches
  Shimodaira [18] proposed diversity control genetic algorithm (DCGA). In this algorithm, higher survival probability is given to candidates with a greater distance to the most befitting individual. Bersano-Begey [19] proposed an evaluation function, which keeps track of how many individuals have solved a particular fitness case, to detect when population has locked in on a partial solution. Chaiyaratana *et al.* [20] modified the DCGA. Then, the survival probability of candidates depends on similarity at the phenotype level. Wong *et al.* [21] repelled population from the most befitting candidate. Two diversity maintenance algorithms, repelling and lazy repelling, were proposed. In the repelling algorithm, the fitness value is added diversity factor to encourage candidates with rare alleles. The lazy repelling algorithm reduces the computational cost of repelling by decreasing the frequency of considering diversity in evaluations. McGinley *et al.* [22] introduced healthy population diversity as feedback to adaptively control selective pressure through tournament size. Adra and Fleming [23] employed a diversity indicator I_s to activate a diversity mechanism. If $I_s < 1$, the binary tournament selection with random tie breaking is inactive. Nagata and Kobayashi [24] developed selection rules based on the selection model of the TDGA. These selection rules require a negligible computational cost and need not adjust parameters. Weise *et al.* [25] introduced frequency fitness for rating how often individuals appear in population. In detail, a frequency fitness assignment process as an addition to the fitness function minimized the reappearance of individuals. Segura *et al.* [26] proposed a replacement strategy for selection based on the idea of transforming a single-objective problem into a multi-objective one. In detail, the distance from the current candidate to the closest surviving individual is considered as an explicit objective. Cuevas *et al.* [27] proposed selection based on the Golden Section (GS), one of the most famous patterns present in nature. In this approach, population is segmented into several groups. Each group involves a certain number of individuals and a probability to be selected, which are determined according to the GS proportion.

III. OUR PROPOSED SCHEME

Elitism strategy in selection always keeps the most befitting candidates and is helpful for exploitation. In practice, a small value, such as one, is often set as the degree of elitism strategy, number of elites. One of the most significant advantages of elitism strategy is that it can be widely used. However, although elitism strategy can always significantly improve convergence speed, it cannot always be benefit for the balance between exploration and exploitation because it one-sidedly emphasizes exploitation. Therefore, a measure for enhancing exploration may be required to cooperate with elitism strategy.

Diversity shows the difference among individuals. If diversity is excessively low, crossover can hardly produce offspring different with parents. Then, little possibility to obtain better offspring than parents is left. In many occasions, it is necessary that diversity is maintained beyond a level. Now that elites are always kept in population under the control of elitism strategy, keeping individuals far from elites at the same time is a feasible way to maintaining diversity without making excess degeneracy.

Based on the above discussion, we propose elitism and distance strategy. Selection procedure with this strategy is given in the Algorithm 1. On one hand, our scheme keeps

Algorithm 1 The Procedure of Selection With our Elitism and Distance Strategy

Input:
  CS_t: the candidate set for selection of the t^{th} generation
  D_e: the number of elites, the degree of elites
  D_d: the number of the individuals having the longest distance to each elite
Output:
  P_{t+1}: the population of the $t+1^{th}$ generation

for $i = 1$ to D_e **do**
  Find the ith elite E_i
  for $j = 1$ to D_d **do**
    Find the individual having the longest distance to E_i, $L_{i,j}$
  end for
end for
Select candidates including all the found individual to P_{t+1}

elites among parents and offspring as elitism strategy does. On the other hand, candidates having the longest distance to one of the elites are also kept. It can be seen that, based on elitism strategy, the extra time complexity of our strategy based on elitism strategy is $O(D_e \cdot D_d \cdot (n-1))$. Thus, the smaller value is assigned to D_e and D_d, the lower the extra time complexity goes. As mentioned above, it is very often that $D_e = 1$. Similarly, D_d can be set 1 for decreasing the extra time complexity. In this case, the extra time complexity is just $O(n-1)$.

IV. EXPERIMENTAL STUDY

Three EAs, the GA for the TSP, DE/rand/2/bin and CoBiDE are involved in our experiments. We observe the diversity changing trend during runs of the former two EAs. Further, we observe not only the diversity changing trend but also the fitness changing trend of CoBiDE. Based on our analysis on observations, we chose to call our elitism and distance strategy for the algorithms in different manners.

A. THE INVOLVED EAS

The flow of the GA for the TSP is given in Algorithm 2. Five datasets from TSPLIB [28] which are difficult for this

Algorithm 2 The GA for the TSP

Input:
  NP: the population size
  MAX_GEN: the maximum number of generations
  p: the only parameter to control both crossover and mutation
Output:
  P_{MAX_GEN}: the final population

Generate the initial population, $P_0 = \{I_1, \ldots, I_{NP}\}$, randomly
for $g = 0 : MAX\_GEN$ **do**
  for $i = 1 : NP$ **do**
    Product a copy of I_i, x, and randomly select a city, c, in the chromosome of x
    repeat
      if $RAND\_float(0, 1) < p$ **then**
        Randomly select a city other than c, c', in the chromosome of x
      else
        $r = RAND\_int(1, NP)$
        In the chromosome of I_r, let the city next to c be c'
      end if
      if c and c' are adjacent in the chromosome of x **then**
        Break the loop
      end if
      Inverse the segment of the chromosome of x from the city next to c to c' and obtain an offspring of I_i, O_i
      c=c'
    until No city in the chromosome of x has never been considered as c
  end for
  Execute tournament selection from all parents and all offspring
end for

algorithm to obtain an optimal solution, rat195, krob200, pr226, a280 and pr439, are used in our experiment on the TSP GA.

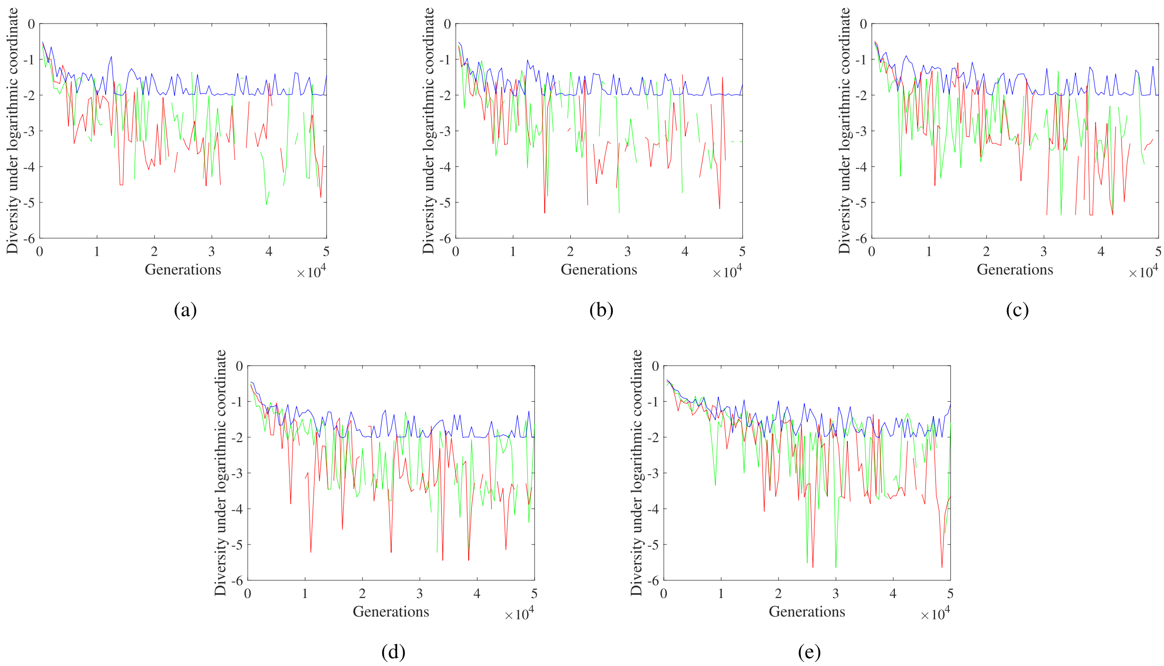

FIGURE 1. **The average diversity during runs of the TSP GA. The green polyline denotes the average diversity without any strategies. The red polyline denotes the average diversity with elitism strategy. The blue polyline denotes the average diversity with our strategy. a: rat195. b: krob200. c: pr226. d: a280 e: pr439.**

DE/rand/2/bin belongs to one of standard DE algorithms and can be easily found in literature. For example, [29] introduced this DE algorithm. Hence, the pseudo code of this algorithm is omitted here. Also, the pseudo code of CoBiDE can be found in [11] and is omitted here. The 25 benchmark functions developed for the 2005 Congress on Evolutionary Computation (CEC) special session [30] are used in the experiments on these two DE algorithms.

B. EXPERIMENT ON THE GA FOR THE TSP

In this experiment, four versions of the TSP GA are executed by us. Except the original TSP GA without either elitism strategy or our strategy, the algorithm with elitism strategy and the algorithm with our strategy, we execute the TSP GA with another strategy named elite and scum strategy in which elites and the worst individuals in fitness are kept for the next generation as a reference object for our proposed strategy. Beside elites, the individuals who have the largest difference on fitness with each elite are remained according to the elite and scum strategy, while the individuals who have the largest difference on distance with each elite are kept according to our proposed strategy. In other words, our strategy focus on genotype difference between individuals, while its reference object focus on phenotype difference. Elite and scum strategy need a parameter D_s to given the number of inferior individuals. Similarly, if $D_s = 1$, the extra time complexity is minimized.

Each version of algorithm is executed 30 times for the five datasets, respectively. $D_e = 1$ for all the strategies, while $D_d = 1$ and $D_s = 1$ for our strategy and elite and scum strategy, respectively. The settings of the TSP GA are listed in TABLE 1 where *NP* denotes population size, *Max_GEN* is the maximum generations and p is the parameter mentioned in Algorithm 2.

TABLE 1. **The settings of the TSP GA.**

NP	100
Max_GEN	5.0E+04
p	0.02

FIGURE 1 gives the diversity change trend during runs of three versions of the algorithm, the version without any strategies, the version with elitism strategy and the version with our proposed strategy. Nevertheless, the diversity change trend of the version with elite and scum strategy is not given. Reasons are given as follow. Firstly, according to TABLE 2, elite and scum strategy cannot significantly improve solutions based on elitism strategy. Moreover, this strategy is not a pointed measure for keeping diversity and just a reference object of our proposed strategy. In each subfigure of FIGURE 1, the average diversity in the 30 runs for a dataset of the three versions of the algorithm is plotted at intervals of 500 generations.

It can be viewed from FIGURE 1 that, in runs of the original version, the diversity changing trend show a similarity for different datasets. In detail, diversity goes down with sharp fluctuations in whole runs. At the bottom of some later fluctuations, diversity value goes to zero. Although elitism scheme

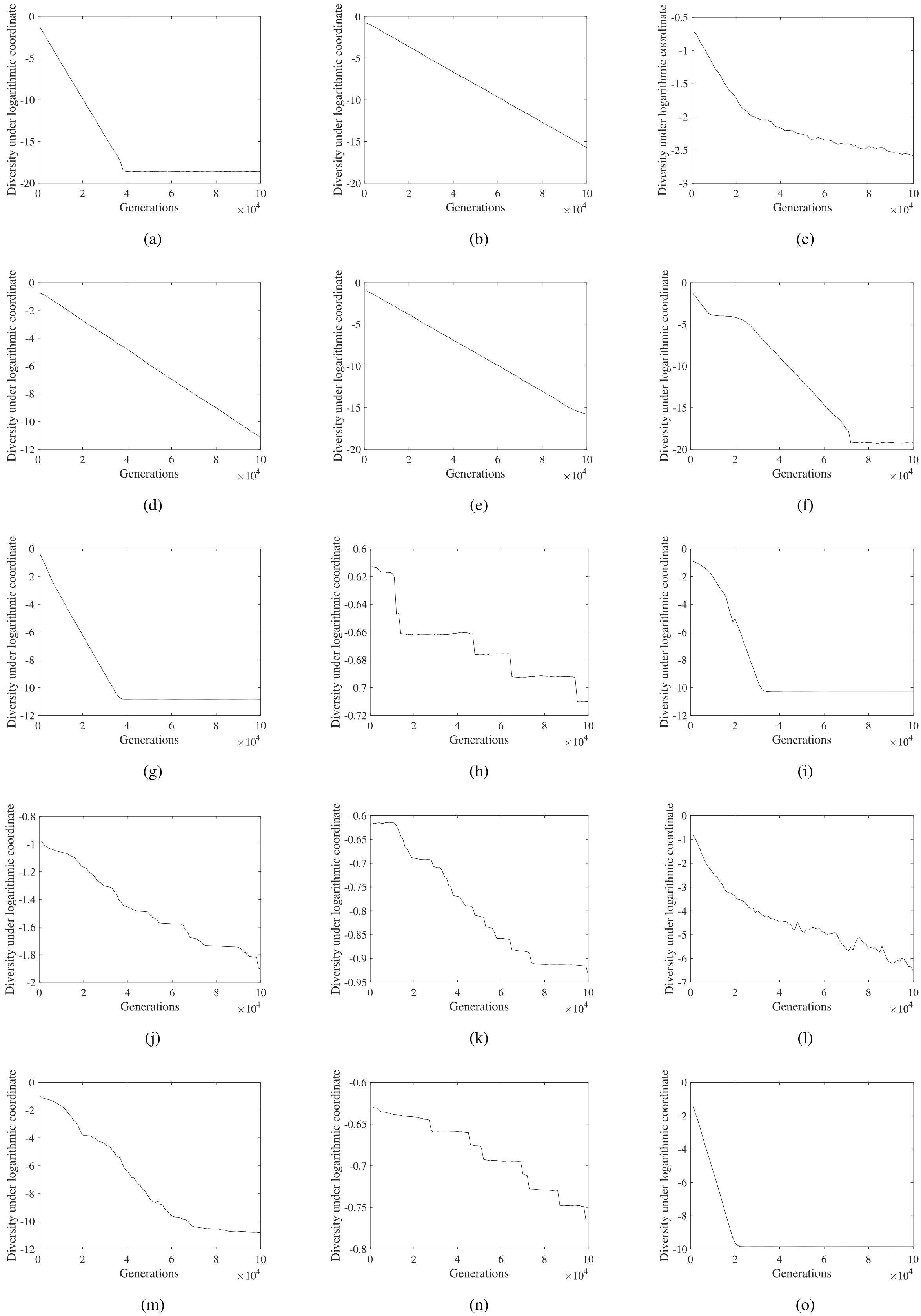

FIGURE 2. The average diversity during runs of the DE/rand/2/bin (Part 1). a: F1. b: F2. c: F3. d: F4. e: F5. f: F6. g: F7. h: F8. i: F9. j: F10. k: F11. l: F12. m: F13. n: F14. o: F15.

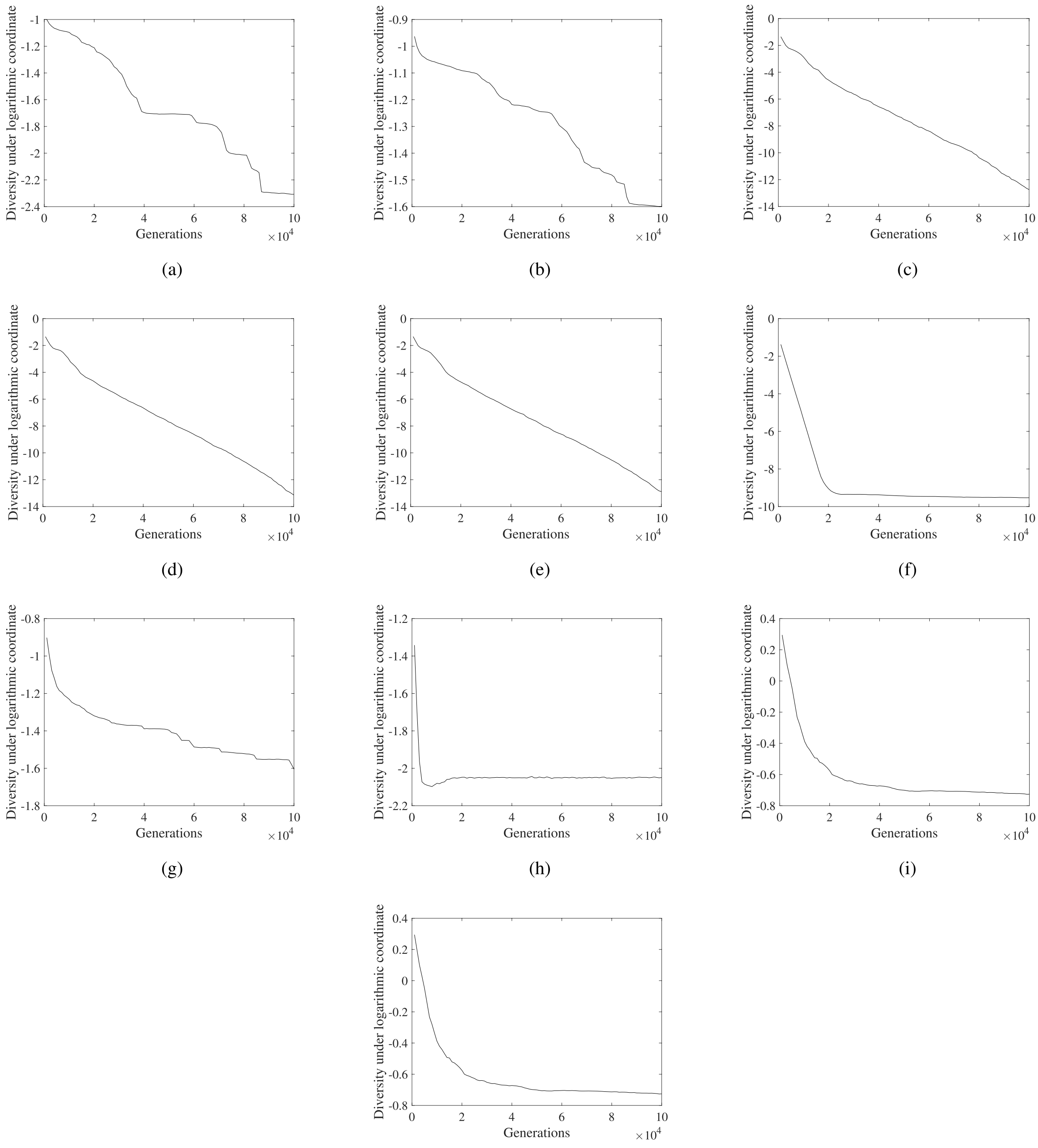

FIGURE 2. ***(Continued.)*** **The average diversity during runs of DE/rand/2/bin (Part 2). a: F16. b: F17. c: F18. d: F19. e: F20. f: F21. g: F22. h: F23. i: F24. j: F25.**

is used, diversity still goes down with sharp fluctuations for the all datasets. However, when our scheme is employed, diversity is maintained much higher during runs.

In TABLE 2, the results are listed. In terms of the t-test with 95% confidence, significant difference exists in some cases of comparison. Details are given as below. In all cases, the results without any strategies are significantly worse than the results with elitism strategy. Moreover, the results with our strategy are statistical better than the results with elitism strategy in all cases. Although the results with the elite and scum strategy are significantly better than the results without any strategies, they are statistically worse than the results with our strategy in all cases. Meanwhile, the results with the elite and scum strategy show no significant difference with the results with elitism strategy.

The similarity in diversity changing trend in runs of the original GA for different datasets reflects that the trend is mainly decided by the combination of operators, parameter

TABLE 2. Results of the four versions of the GA for the TSP.

Dataset	Average (standard deviation)			
	Without any strategies	With elitism strategy	With elite and scum strategy	With our strategy
Rat195	5.0706E+03 (1.56E+02)	4.6539E+03 (1.20E+02)	4.6380E+03 (1.34E+02)	3.9492E+03 (1.04E+02)
Krob200	7.0117E+04 (3.39E+03)	6.4396E+04 (1.56E+03)	6.4218E+04 (1.59E+03)	5.2924E+04 (1.68E+03)
Pr226	2.6482E+05 (1.30E+04)	2.2525E+05 (1.03E+04)	2.2594E+05 (9.81E+03)	1.6817E+05 (9.30E+03)
A280	8.0521E+03 (2.14E+02)	7.3147E+03 (1.77E+02)	7.2881E+03 (1.82E+02)	5.9388E+03 (1.68E+02)
Pr439	4.8232E+05 (1.44E+04)	4.4108E+05 (1.06E+04)	4.4254E+03 (1.08E+02)	3.4681E+05 (1.24E+04)

settings, etc. In short, the improper combination of operators, parameter settings, etc leads to the severe imbalance between exploration and exploitation which appears as lack of diversity. Our strategy changes the combination to some extent. Such a change is beneficial to achieve the balance between exploration and exploitation.

A summary of this experiment can be given as below. Without any schemes, diversity always tends to be lack in runs of this TSP GA and even goes to zero from time to time. Elitism strategy hardly does good to maintain diversity. So does elite and scum strategy. However, our proposed strategy is conducive to the balance between exploration and exploitation with the phenomenon that diversity is maintained. As a result, our strategy leads to further improvement on solutions based on elitism strategy.

C. EXPERIMENT ON DE/RAND/2/BIN

Firstly, we run DE/rand/2/bin to observe the diversity changing trend during runs. DE/rand/2/bin settings are listed in TABLE 3. FIGURE 2 give the diversity change trend during runs in DE/rand/2/bin.

TABLE 3. The Settings of DE/rand/2/bin.

NP	60
Max_GEN	1.0E+05
F	0.90
CR	0.90

For each function, the average diversity in the 30 runs is plotted at intervals of 1000 generations in each subfigure of FIGURE 2.

According to FIGURE 2, the diversity changing trend during runs of the DE/rand/2/bin shows quite distinct for different functions. In detail, the trend for F1, F7, F9, F15, F21 and F23-F25 shows constantly decrease in the early stage of runs and becomes a flat line in the late stage. Moreover, the trend for F2-F5 and F18-F20 shows constantly decrease during the whole course. In addition, the trend for F6, F8, F10-F14, F16-F17 and F22 shows decrease period and flat line alternately. In short, for different functions, the diversity changing trend during runs of the DE/rand/2/bin shows different features which can be divided into at least three types roughly. These differences in the trend demonstrate that the combination of operators, parameter settings, etc cannot be the main cause of the trend. Instead, the fitness landscapes of each function becomes the more important cause.

Now that the DE/rand/2/bin's combination of operators, parameter settings, etc does not lead to a significant imbalance between exploration and exploitation, the selection manner of this algorithm need not change too much. Hence, we plan to call our elitism and distance strategy at a small rate for giving an unpredicted surviving chance to some individuals far from elites in selection as an attempt to improve solutions.

Since elitism strategy has been used in selection of DE/rand/2/bin by default, just three versions of this algorithm, with elitism strategy, with elite and scum strategy and with our strategy, are compared in this experiment. In fact, the version with elitism strategy is the original version. The different versions of the algorithm are executed 30 times for the 25 benchmark functions, respectively. For all the strategies, $D_e = 1$. In addition, $D_d = 1$ for our strategy, while $D_s = 1$ for elite and scum strategy. We just set only one value, 0.0017, for both the probability of applying elite and scum strategy and that of applying our strategy.

In TABLE 4, results are listed. For each function, the result of the algorithm with elite and scum strategy and the result of the algorithm with our strategy is given in italics if significantly different in terms of the Wilcoxon's rank sum test at a 0.05 significance level exists when compared with the result of the original version of the DE/rand/2/bin.

According to TABLE 4, there is only one case existing significant difference in the comparison between the original version and the version with elite and scum strategy. In detail, for F25, the results of the latter version are significant worse than the results of the former version. In the comparison between the original version and the version with our strategy, it can be seen that significant difference exists for eight functions, F3, F5-F6, F8, F10, F12-F13 and F18. In detail, for F3, F5-F6, F8, F10 and F12, the version with our strategy statistically wins. Meanwhile, in the case for F13, our version significantly loses.

A summary of this experiment goes follow. Although the combination of operators, parameter settings, etc is befitting, the different fitness landscapes of difficult tasks make different phenomena of the imbalance between exploration and exploitation. Consequently, in many cases, solutions of DE/rand/2/bin require be improved. Calling our strategy at a low probability is a useful method to improve solutions by fine tuning the exploration and exploitation ratio, while calling elite and scum strategy leads no significant improvement. In this experiment, we just set one value 0.0017 for the probability and significantly improve solutions of eight

TABLE 4. Results of three versions of the DE/rand/2/bin.

Function	Average (standard deviation)		
	With elitism strategy	With elite and scum strategy	With our strategy
F1	3.3658E-29 (6.27E-29)	6.2267E-29 (1.0581e-28)	7.4048e-29 (1.04e-28)
F2	7.2081E-26 (1.48E-25)	1.6269E-25 (5.4514E-25)	6.8901e-26 (1.22e-25)
F3	3.2650E+04 (2.36E+04)	3.2561E+04 (2.5916E+04)	*2.2436E+04 (1.81E+04)*
F4	2.1387E-16 (4.45E-16)	2.5166E-16 (6.0387E-16)	1.7715e-16 (5.31e-16)
F5	1.7144E-11 (1.08E-11)	1.7258E-11 (8.3954E-12)	*1.3597E-11 (5.08E-12)*
F6	1.4483E-25 (2.09E-25)	6.6147E-26 (8.1894E-26)	*3.6313E-26 (4.35E-26)*
F7	2.4653E-04 (1.35E-03)	2.4653E-04 (1.35E-03)	2.4653E-04 (1.35E-03)
F8	2.0853E+01 (5.92E-02)	2.0839E+01 (5.21E-02)	*2.0821E+01 (4.53E-02)*
F9	1.1989E+01 (3.71E+00)	1.1993E+01 (5.86E+00)	1.2837E+01 (4.16E+00)
F10	6.3955E+01 (5.76E+01)	7.8447E+01 (7.39E+01)	*3.6453E+01 (3.19E+01)*
F11	2.1671E+01 (1.58E+01)	2.0255E+01 (1.63E+01)	2.2237E+01 (1.60E+01)
F12	2.6565E+03 (2.81E+03)	2.6886E+03 (3.84E+03)	*1.5602E+03 (2.61E+03)*
F13	2.6096E+00 (7.02E-01)	2.4910E+00 (5.24E-01)	*2.8588E+00 (6.11E-01)*
F14	1.3017E+01 (1.58E-01)	1.3064E+01 (1.23E-01)	1.2910E+01 (3.02E-01)
F15	3.7333E+02 (6.91E+01)	3.9667E+02 (6.15E+01)	3.9672E+02 (4.14E+01)
F16	7.5446E+01 (4.65E+01)	8.7988E+01 (6.01E+01)	9.3304E+01 (6.51E+01)
F17	1.2296E+02 (8.85E+01)	1.7463E+02 (8.95E+02)	1.3649E+02 (8.51E+01)
F18	9.0352E+02 (2.04E-01)	9.0347E+02 (1.69E-01)	*9.0339E+02 (1.97E-01)*
F19	9.0340E+02 (1.51E-01)	9.0341E+02 (1.61E-01)	9.0347E+02 (1.69E-01)
F20	9.0338E+02 (1.61E-01)	9.0342E+02 (1.45E-01)	9.0336E+02 (2.10E-01)
F21	5.0000E+02 (0.00E+00)	5.0000E+02 (0.00E+00)	5.0000E+02 (0.00E+00)
F22	8.6495E+02 (9.78E+00)	8.6578E+02 (1.42E+01)	8.6203E+02 (1.49E+01)
F23	5.3416E+02 (4.02E-04)	5.3416E+02 (4.34E-04)	5.3416E+02 (5.18E-04)
F24	2.0000E+02 (0.00E+00)	2.0000E+02 (0.00E+00)	2.0000E+02 (0.00E+00)
F25	2.0892E+02 (2.73E-01)	*2.0904E+02 (2.81E-01)*	2.0889E+02 (2.14E-01)

functions out of 25 ones. Provided that more values are tested, solutions of more functions may be improved since functions are always different in fitness landscapes.

D. EXPERIMENT ON COBIDE

Firstly, we run CoBiDE to observe the diversity change trend during its runs. CoBiDE settings from [11] are listed in TABLE 5. We set *Max_GEN* larger than [11] but less than our experiment on DE/rand/2/bin. Reasons are as below. On one hand, enough generations are required for observing the diversity changing trend. On the other hand, runs for some functions have obtained an optimal in generations much less than *Max_GEN* set in our experiment on DE/rand/2/bin.

TABLE 5. Settings for the CoBiDE.

NP	60
Max_GEN	5.0E+04
pb	0.40
ps	0.50
Terminal criterion	50000 generations done

Enhancing CoBiDE is more difficult than improving the TSP GA and DE/rand/2/bin since CoBiDE is a state-of-the-art algorithm. Features in runs should be deeply investigated. In our plan, beside the diversity changing trend, the fitness changing trend requires be considered. In detail, we study not only the average diversity but also the average of the current best value in the 30 runs of original CoBiDE at intervals of 500 generations and include that runs of 10 functions have some common features. For the 10 functions, we plot both series of average values at intervals in FIGURE 3 to show the features. Each subfigure of this figure is for a function.

It can be seen from FIGURE 3 that, for F10-F14, F16-F19 and F22, improvement in fitness ceases after diversity goes lower than a value. Thus, we plan to call our strategy to improve diversity if the decrease of diversity is rapid. To improve diversity rapidly, the degree of our strategy D_d is set $NP-1$. Details are given in Algorithm 3. Here, NP is

Algorithm 3 The Adaptive Scheme for Calling our Strategy in CoBiDE

Input:
  i: the interval of generations;
  e: the number of power;
  $n = 1$
  Execute other steps for initialization
  for $g = 0 : MAX\_GEN$ **do**
    Execute steps other than selection in each generation
    if $g\%i = 0$ **then**
      Compute current diversity, d_n
      if $d_{n-1} >= d_n$ **then**
        $ir = \frac{d_{n-1} - d_n}{d_{n-1}}$
      else
        $ir = 0$
      end if
      $n = n + 1$
    end if
    Execute selection
    if $RAND\_float(0, 1) <= ir^e$ **then**
      Apply our strategy with $NP-1$ in the degree of elites in this generation
    end if
  end for

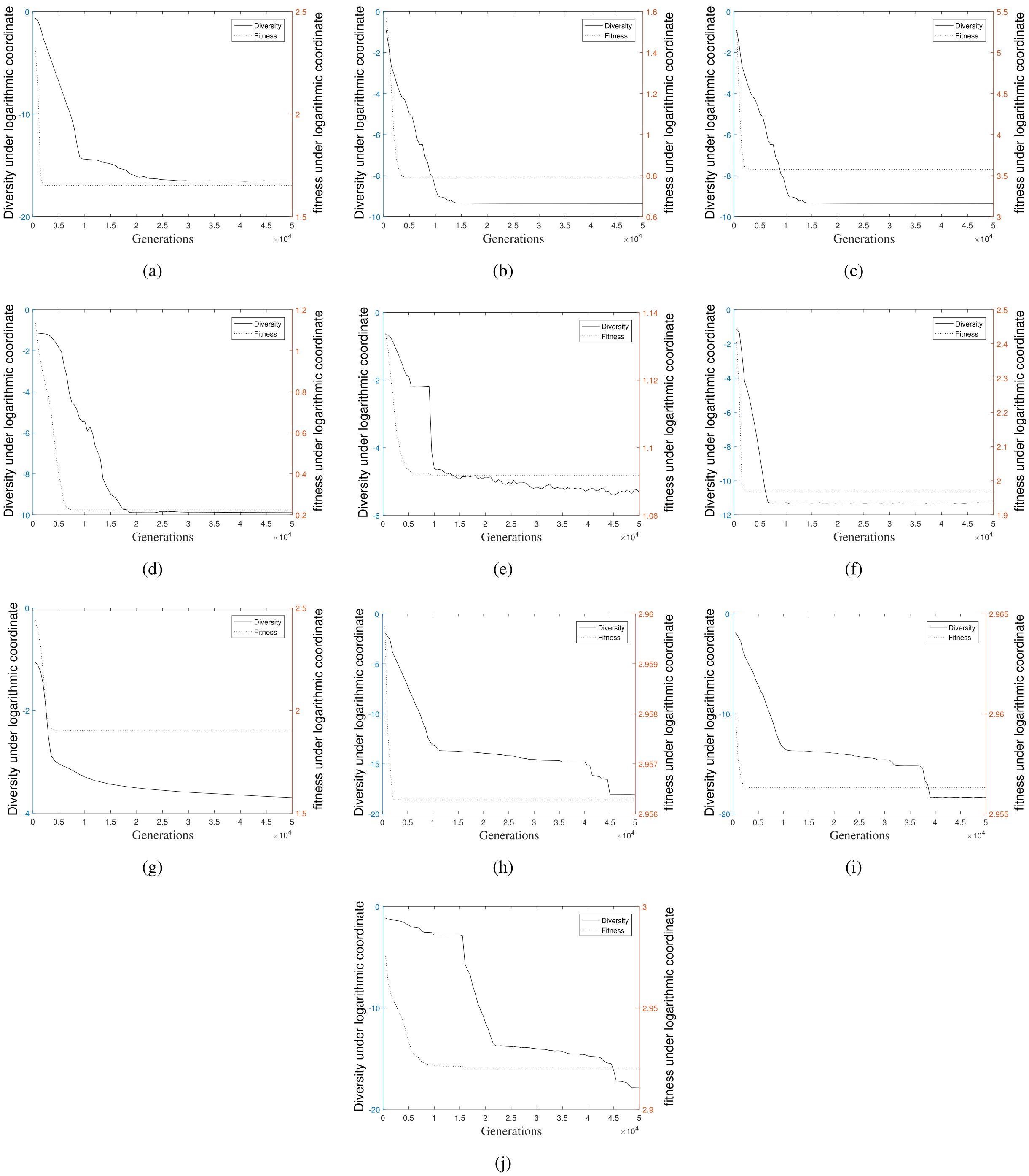

FIGURE 3. The average diversity during runs of CoBiDE. a: F10. b: F11. c: F12. d: F13. e: F14. f: F16. g: F17. h: F18. i: F19. j: F22. Solid represents diversity trend, while dashes denote fitness trend.

population size, and g denotes the number of generations. d_x represents diversity in the generation at the xth interval, and d_0 is the genotype diversity in the initial generation. According to Algorithm 3, the higher the decreasing ratio of genotype diversity during an interval, the higher the possibility of calling secondary selection in generations during the next interval. In this way, our strategy with the large degree is called frequently when diversity has reduced sharply.

We run CoBiDE with the proposed adaptive scheme 30 time for the 10 above-mentioned functions, respectively. In TABLE 6, the results of original CoBiDE and the results with our strategy are both listed. It can be seen that, for the 10 functions, our strategy under the control of the adaptive scheme improves solutions of these 10 functions.

A summary of this experiment goes follow. Although CoBiDE is a state-of-the-art DE algorithm, for certain

TABLE 6. The significant improvement in terms of Wilcoxon's rank sum test at a 0.05 significance level in the experiment on CoBiDE.

Function	Average (standard deviation)	
	With elitism strategy	With our strategy
F10	4.4942E+01 (1.51E+01)	3.5586E+01 (8.68E+00)
F11	6.1589E+00 (3.08E+00)	4.3288E+00 (1.93E+00)
F12	3.7778E+03 (4.28E+03)	1.4656E+03 (2.39E+03)
F13	1.6766E+00 (4.12E-01)	1.2939E+00 (2.17E-01)
F14	1.2356E+01 (4.32E-01)	1.1456E+01 (4.71E-01)
F16	9.2541E+01 (6.78E+01)	6.1822E+01 (8.56E+00)
F17	7.9355E+01 (2.75E+01)	6.7199E+01 (1.06E+01)
F18	9.0423E+02 (8.80E-01)	9.0418E+02 (3.23E-01)
F19	9.0429E+02 (1.10E+00)	9.0417E+02 (4.03E-01)
F22	8.3259E+02 (2.20E+01)	8.1663E+02 (0.00E+00)

functions, its solutions still need be improved. We find that, in runs of original CoBiDE for some functions, fitness cannot make a progress as soon as diversity is lower than a value. The above phenomenon reflect the imbalance between exploration and exploitation. Based on such an observation, we propose a scheme to call our strategy adaptively. Then, solutions of the 10 functions go significantly better since the exploration and exploitation balance is improved.

V. CONCLUSIONS

In this paper, we proposed elitism and distance strategy used in selection of EAs based on the widely used strategy, elitism strategy. In our strategy, beside the best individuals, individuals with longer distance to each of the best individuals are remained to the next generation. When an EA is serious lack of diversity in runs, our strategy is required to be called in every generations for the balance between exploration and exploitation. If runs of an EA for different tasks show different diversity changing trends, calling our strategy with a probability is beneficial for the balance. Then, solutions are improved. More importantly, for runs of a state-of-the-art EA for especially tasks, an adaptive scheme needs be designed after studying diversity changing trends to call our strategy for obtaining better solutions. In short, diversity changing trend in runs should be observed before calling this strategy accurately. Sometimes, adaptive schemes based on diversity need be proposed for calling this strategy.

Related researches are remained to be done. Firstly, elitism and distance strategy can be further studied for improving efficiency. More importantly, adaptive schemes for calling our strategy in more situations should be proposed. We will focus on these two aspects in the future.

REFERENCES

[1] D. B. Fogel, "The advantages of evolutionary computation," in *Proc. BCEC*, 1997, pp. 1–11.

[2] M. Črepinšek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: A survey," *ACM Comput. Surv.*, vol. 45, no. 3, p. 35, 2013.

[3] D. Dumitrescu, B. Lazzerini, L. C. Jain, and A. Dumitrescu, *Evolutionary Computation*. Boca Raton, FL, USA: CRC Press, 2000.

[4] C. W. Ahn and R. S. Ramakrishna, "Elitism-based compact genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 4, pp. 367–385, Aug. 2003.

[5] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*, vol. 7. Hoboken, NJ, USA: Wiley, 2000.

[6] N. Fang, J. Zhou, R. Zhang, Y. Liu, and Y. Zhang, "A hybrid of real coded genetic algorithm and artificial fish swarm algorithm for short-term optimal hydrothermal scheduling," *Int. J. Elect. Power Energy Syst.*, vol. 62, pp. 617–629, Nov. 2014.

[7] H.-X. Huang, J.-C. Li, and C.-L. Xiao, "A proposed iteration optimization approach integrating backpropagation neural network with genetic algorithm," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 146–155, 2015.

[8] I. J. Leno, S. S. Sankar, and S. Ponnambalam, "An elitist strategy genetic algorithm using simulated annealing algorithm as local search for facility layout design," *Int. J. Adv. Manuf. Technol.*, vol. 84, nos. 5–8, pp. 787–799, 2016.

[9] W. Zhang, W. Xu, G. Liu, and M. Gen, "An effective hybrid evolutionary algorithm for stochastic multiobjective assembly line balancing problem," *J. Intell. Manuf.*, vol. 28, no. 3, pp. 783–790, 2017.

[10] I. J. Leno, S. S. Sankar, and S. Ponnambalam, "MIP model and elitist strategy hybrid GA–SA algorithm for layout design," *J. Intell. Manuf.*, vol. 29, no. 2, pp. 369–387, 2018.

[11] Y. Wang, H.-X. Li, T. Huang, and L. Li, "Differential evolution based on covariance matrix learning and bimodal distribution parameter setting," *Appl. Soft Comput.*, vol. 18, pp. 232–247, May 2014.

[12] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, vol. 1. Springer, 1996.

[13] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[14] K. Matsui, "New selection method to improve the population diversity in genetic algorithms," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, vol. 1, Oct. 1999, pp. 625–630.

[15] M. Hutter and S. Legg, "Fitness uniform optimization," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 568–589, Oct. 2006.

[16] G. Chen, C. P. Low, and Z. Yang, "Preserving and exploiting genetic diversity in evolutionary programming algorithms," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 661–673, Jun. 2009.

[17] N. Mori, J. Yoshida, H. Tamaki, and H. Nishikawa, "A thermodynamical selection rule for the genetic algorithm," in *Proc. IEEE Int. Conf. Evol. Comput.*, vol. 1, Nov./Dec. 1995, p. 188.

[18] H. Shimodaira, "DCGA: A diversity control oriented genetic algorithm," in *Proc. 2nd Int. Conf. Genetic Algorithms Eng. Syst., Innov. Appl.*, Sep. 1997, pp. 367–374.

[19] T. Bersano-Begey, "Controlling exploration, diversity and escaping local optima in GP: Adapting weights of training sets to model resource consumption," in *Proc. Late Breaking Papers Genetic Program. Conf.*, 1997, pp. 7–10.

[20] N. Chaiyaratana, T. Piroonratana, and N. Sangkawelert, "Effects of diversity control in single-objective and multi-objective genetic algorithms," *J. Heuristics*, vol. 13, no. 1, pp. 1–34, 2007.

[21] Y.-Y. Wong, K.-H. Lee, K.-S. Leung, and C.-W. Ho, "A novel approach in parameter adaptation and diversity maintenance for genetic algorithms," *Soft Comput.*, vol. 7, no. 8, pp. 506–515, 2003.

[22] B. M. Ginley, J. Maher, C. O'Riordan, and F. Morgan, "Maintaining healthy population diversity using adaptive crossover, mutation, and selection," *IEEE Trans. Evol. Comput.*, vol. 15, no. 5, pp. 692–714, Oct. 2011.

[23] S. F. Adra and P. J. Fleming, "Diversity management in evolutionary many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 15, no. 2, pp. 183–195, Apr. 2011.

[24] Y. Nagata and S. Kobayashi, "A powerful genetic algorithm using edge assembly crossover for the traveling salesman problem," *INFORMS J. Comput.*, vol. 25, no. 2, pp. 346–363, 2013.

[25] T. Weise, M. Wan, P. Wang, K. Tang, A. Devert, and X. Yao, "Frequency fitness assignment," *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 226–243, Apr. 2014.

[26] C. Segura, C. A. C. Coello, E. Segredo, and A. H. Aguirre, "A novel diversity-based replacement strategy for evolutionary algorithms," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 3233–3246, Dec. 2016.

[27] E. Cuevas, L. Enríquez, D. Zaldívar, and M. Pérez-Cisneros, "A selection method for evolutionary algorithms based on the golden section," *Expert Syst. Appl.*, vol. 106, pp. 183–196, Sep. 2018.

[28] G. Reinelt, "TSPLIB—A traveling salesman problem library," *ORSA J. Comput.*, vol. 3, no. 4, pp. 376–384, 1991.

[29] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.

[30] P. N. Suganthan *et al.*, "Problem definitions and evaluation criteria for the cec 2005 special session on real-parameter optimization," KanGAL Rep. 2005005, 2005.

HAIMING DU was born in Lingbao, Henan, in 1977. He received the bachelor's degree in electric engineering from the Harbin University of Science and Technology, Harbin, China, in 2001, the master's degree in communication engineering from Harbin Engineering University, Harbin, in 2004, and the Ph.D. degree in electromagnetic and microwave technology from the Huazhong University of Science and Technology, Wuhan, China, in 2012.

He joined the Zhengzhou University of Light Industry in 2004, where he is currently a Teacher with the School of Electrical and Information Engineering. His research interests include not only signal processing in wireless communication and evolutionary but also geo-computation.

ZAICHAO WANG was born in Zhengzhou, Henan, in 1992. He received the bachelor's degree in electrical engineering and automatic from the Shangqiu College in 2015. He is currently pursuing the master's degree in instrumentation engineering with the School of Electrical and Information Engineering, Zhengzhou University of Light Industry.

WEI ZHAN received the Ph.D. degree in computer science from the China University of Geosciences in 2013. In 2015, he was a Visiting Scholar with the University of Missouri-Kansas City, USA. He is currently an Associate Professor with the School of Computer Science, Yangtze University, Jingzhou, Hubei, China. His main research interests focus on computer vision for big image data via deep learning algorithm, computer vision application development via Microsoft open source computer vision toolkit, and evolutionary computing and its application.

JINYI GUO was born in Hohhot, Inner Mongolia, China, in 1997. She is currently pursuing the bachelor's degree with the School of Computer Science, China University of Geosciences (Wuhan), China.