Historical and Heuristic-Based Adaptive Differential Evolution

Xiao-Fang Liu, *Student Member, IEEE*, Zhi-Hui Zhan, *Member, IEEE*, Ying Lin, *Member, IEEE*, Wei-Neng Chen, *Member, IEEE*, Yue-Jiao Gong, *Member, IEEE*, Tian-Long Gu, Hua-Qiang Yuan, and Jun Zhang, *Fellow, IEEE*

***Abstract*—As the mutation strategy and algorithmic parameters in differential evolution (DE) are sensitive to the problems being solved, a hot research topic is to adaptively control the strategy and parameters according to the requirements of the problem. In the literature, most adaptive DE use either historical experiences of the population or heuristic information of the individuals to promote adaptation. In this paper, we develop a novel variant of adaptive DE, utilizing both the historical experience and heuristic information for the adaptation. In this novel historical and heuristic DE (HHDE), each individual dynamically adjusts its mutation strategy and associated parameters not only by learning from previous successful experience of the whole population, but also according to heuristic information related with its own current state. These help the algorithm select a more suitable mutation strategy and determine better parameters for each individual in different evolutionary stages. The performance of the proposed HHDE is extensively evaluated on 30 benchmark functions with different dimensions. Experimental results confirm the competitiveness of the proposed algorithm to a number of DE variants.**

***Index Terms*—Differential evolution (DE), heuristic information, historical experience, mutation strategy selection, parameter adaptation.**

I. Introduction

Differential evolution (DE) is a simple and efficient global optimizer developed by Storn and Price [1]. DE has been successfully applied to many real-world optimization problems [2]–[8]. DE utilizes differences between individuals to drive the evolution iteratively via the mutation, crossover, and selection operators. Among these operators, mutation strategy is the most sensitive because different mutation strategies present different characteristics and are suitable for different evolutionary stages on different problems [9]. Moreover, DE uses parameters scale factor (F) and crossover rate (CR) to control the evolutionary process. Researches have shown that different parameter settings are required when dealing with different problems or being in different evolutionary stages [10]. Therefore, during recent years, researches on mutation strategies and parameter settings of DE have attracted an increasing interest, with a number of adaptive/self-adaptive DE variants being proposed.

In the literature, two types of mechanisms are often used for mutation strategy and parameter adaptation.

1) Historical experience-based mechanism (HEM), where the mutation strategy and parameters are set according to the historical successful experience of the population. In the evolutionary process, individuals are assigned with different mutation strategies and parameters, attempting to produce diverse offspring. The successful attempts indicate "*Good*" mutation strategies and parameters that can produce better individual survival in the next generation. So these *Good* historical experiences are regarded as learning sources to guide the selection of mutation strategies and parameters. In the literatures, many adaptation methods use HEM for parameter control [11]–[16]. In these DEs, the successful parameters in the previous generation are directly propagated to next generation [11] or used as learning exemplars to generate new parameters [17]. The HEM is also adopted widely for mutation strategy selection. The main idea is to adaptively update the selection probability of strategies according to their success history [18]–[20]. The strategy with higher success rate (SR) is regarded as more suitable for the problem and is assigned more computational resource [21] or selected with higher probability [22].
2) Heuristic information-based mechanism (HIM), where the mutation strategy and parameters are set for each individual according to the heuristic information based on its current state, such as the current position, the corresponding fitness value, and/or the ranking information.

Manuscript received January 30, 2018; revised April 30, 2018; accepted June 28, 2018. Date of publication August 8, 2018; date of current version November 19, 2019. This work was supported in part by the National Natural Science Foundations of China under Grant 61772207 and Grant 61332002, in part by the Natural Science Foundations of Guangdong Province for Distinguished Young Scholars under Grant 2014A030306038, in part by the Project for Pearl River New Star in Science and Technology under Grant 201506010047, in part by GDUPS in 2016, and in part by the Fundamental Research Funds for the Central Universities. This paper was recommended by Associate Editor J.-H. Chou. *(Corresponding authors: Zhi-Hui Zhan; Jun Zhang.)*

X.-F. Liu, Z.-H. Zhan, Y. Lin, W.-N. Chen, Y.-J. Gong, and J. Zhang are with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China, and also with the Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, South China University of Technology, Guangzhou 510006, China (e-mail: zhanapollo@163.com; junzhang@ieee.org).

T.-L. Gu is with the School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin 541004, China.

H.-Q. Yuan is with the School of Computer Science and Network Security, Dongguan University of Technology, Dongguan 523808, China.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author. This includes experimental results and figures not included in the paper. This material is 558 KB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2855155

For example, Ghosh *et al.* [23] proposed an improved DE which assigns control parameters for each individual based on the fitness value. Tang *et al.* [24] proposed an individual-dependent DE (IDE), taking the difference of different individuals into consideration when generate the parameters and determine the strategy.

Note that some other similar classification terms, such as success-based/observation-based [27], are also used to classify adaptive DEs from the perspectives of population success and population distribution. However, the HEM/HIM terms used in this paper focus more on whether the adaptation is controlled by macro-level historical experiences from the population or micro-level heuristic information from the individual.

Both the HEM and HIM for mutation strategy and parameter adaptation show good performance when compared to the classical DE. In the HEM, individuals are regarded as equal, and their strategies and parameters are assigned according to the same "distribution" learning from the same historical experiences. Alternatively, in the HIM, the differences of individuals are used as heuristic information for adaptation. In the literature, the HEM and HIM are often utilized in simple and separate ways. However, it may be efficient to make comprehensive utilization of them both. On the one hand, the population historical experiences are useful to describe the "shape" of search space that can provide an experienced guidance. On the other hand, the individual heuristic information can yield the difference between individuals and reflect their various search requirements for strategy and parameters. The combination of them can help the individual itself find promising strategy and parameters that are suitable for the problem as well as for the individual. This has attracted researchers' attentions. Efforts have already been seen in combining HEM and HIM for parameter adaptation [25], [26]. Zhou *et al.* [25] proposed to generate *CR* values in a two-step process. Multiple *CR* values are generated by a distribution according to the population's historical experience first and then are assigned to the individuals according to their fitness values. However, the historical experience and heuristic information are utilized separately. Jia *et al.* [26] proposed to generate parameters for each individual according to its own past parameters and fitness value. However, it does not consider the cooperation among individuals' experiences. For strategy adaption, to our best knowledge, there is still few even no works that adopt both of HEM and HIM. Therefore, how to utilize both the population's historical experience and individual's current heuristic information in a more direct and effective way to select promising mutation strategy and parameters is an important and challenging research issue.

How to represent the information of HEM and HIM, and how to combine them together to generate parameter/strategy are two crucial issues. For parameters, they are real values in continuous domain. Therefore, it is natural to calculate the HEM and HIM as continuous values and adopt weight sum method to combine them to generate new parameters. However, for strategy, it is in discrete domain to select one strategy. Therefore, special design is required to determine the values of HEM and HIM and then combine them together to construct a probabilistic model for strategy selection.

In this paper, we propose a novel historical experience and heuristic information-based adaptive DE algorithm, termed HHDE. In HHDE, the mutation strategy and parameters for each individual are adaptively controlled by both the historical success experience of the whole population and the heuristic information related to individual's own current state. Therefore, HHDE combines the advantages of both HEM and HIM. Specially, the new HHDE includes a historical-heuristic-based mutation (HHM) strategy and a historical-heuristic-based parameter (HHP) setting. In the HHM, a mutation strategy selection probabilistic model associated with historical SR of different mutation strategies (can be regarded as an HEM) and heuristic information representing the preference of the individual "state" to the mutation strategy characteristic (can be regarded as an HIM) is designed for each individual to match the problem as well as the individual. The HHP first ranks the individuals based on their fitness values and then determines the parameters for each individual by considering its ranking information (can be regarded as an HIM) and by learning from successful parameters of the population (can be regarded as an HEM). In our HHDE, the HEM and HIM are combined in a direct way to construct a distribution for strategy selection or parameter determination for each individual. In this way, the population historical experience is adapted to the individual itself. To our best knowledge, this is an innovative attempt in the DE community that proposes to use both the population historical experience and the individual heuristic information for both strategy and parameter adaptation. To evaluate the efficiency of the proposed algorithm, the performance of HHDE on 30 benchmark functions from CEC 2014 with 10-D, 30-D, and 50-D scales [29] is compared with other state-of-the-art DE algorithms and some top methods in CEC 2014 and CEC 2016 competition.

The rest of this paper is organized as follows. To set the scene, Section II describes the basic procedure of DE and its improved variants in strategy and parameter. Section III develops the new HHDE algorithm in detail. Experimental results are presented in Section IV. Finally, the conclusions are drawn in Section V.

II. DE and Its Variants

A. DE Algorithm

DE is a population-based stochastic search algorithm for global optimization. In a D-dimensional space, each individual i has a position vector $\boldsymbol{x}_{i,g} = (x_{1,i,g}, x_{2,i,g}, \dots, x_{D,i,g})$, mutation scale factor F_i, and crossover rate CR_i, where $i = 1, 2, \dots, Np$, Np is the population size, g is the current generation index. In the beginning, all the individuals are randomly initialized as a position in the searching space $x_{j,\min} \le x_{j,i,0} \le x_{j,\max}$ for $j = 1, 2, \dots, D$. Then DE repeatedly performs three operations: 1) mutation; 2) crossover; and 3) selection to generate a new population.

1) Mutation: In each generation g, each individual i generates a mutant vector $\boldsymbol{v}_{i,g}$. The frequently used mutation strategies of DE can be named as "DE/-/k," in which the "−" indicates the base vector and k is the number of difference vectors used for constructing the moving direction [13]. The common mutation strategies are listed as follows:

DE/rand/1 [1]:

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{r1,g} + F_i \cdot \left(\boldsymbol{x}_{r2,g} - \boldsymbol{x}_{r3,g}\right). \tag{1}$$

DE/best/1 [47]:

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{\text{best},g} + F_i \cdot \left(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g}\right). \tag{2}$$

DE/current-to-best/1 [48]:

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{i,g} + F_i \cdot \left(\boldsymbol{x}_{\text{best},g} - \boldsymbol{x}_{i,g}\right) + F_i \cdot \left(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g}\right). \tag{3}$$

DE/current-to-rand/1 [49]:

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{i,g} + \lambda \cdot \left(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{i,g}\right) + F_i \cdot \left(\boldsymbol{x}_{r2,g} - \boldsymbol{x}_{r3,g}\right) \tag{4}$$

where r_1, r_2, and r_3 are distinct integers randomly selected from $\{1, 2, \ldots, Np\}$, which are all different from the index i. F_i is a positive real number of individual i for scaling the difference vectors, λ is a random value in the range of [0,1], and $\boldsymbol{x}_{\text{best},g}$ is the best individual at generation g.

2) Crossover: Usually, a crossover operation is performed following mutation, to form the offspring vector. The binomial crossover and exponential crossover are commonly used. In an exponential crossover, the trial vector $\boldsymbol{u}_{i,g} = (u_{1,i,g}, u_{2,i,g}, \ldots, u_{D,i,g})$ is formed by exchanging one part connected components of mutant vector with the target vector $\boldsymbol{x}_{i,g}$. The starting point and the component length of crossover are randomly generated [30], [31]. Differently, binomial crossover is performed on each dimension j if the randomly generated number is less than or equal to CR_i as

$$u_{j,i,g} = \begin{cases} v_{j,i,g}, & \text{if rand}(0,1) \le CR_i \text{ or } j = j_{\text{rand}} \\ x_{j,i,g}, & \text{otherwise} \end{cases} \tag{5}$$

where j_{rand} is a random integer in [1, D] to ensure that there is at least one component that is inherited from the mutant vector.

If the value of the trial vector violates boundary constraints, the value will be reset as

$$u_{j,i,g} = \begin{cases} \left(x_{j,i,g} + x_{j,\min}\right)/2, & \text{if } u_{j,i,g} < x_{j,\min} \\ \left(x_{j,i,g} + x_{j,\max}\right)/2, & \text{if } u_{j,i,g} > x_{j,\max}. \end{cases} \tag{6}$$

3) Selection: A vector with a better fitness value is selected from the trial vector and target vector to enter the next generation. If the objective is to find a minimum solution, the selection operation is as follows:

$$\boldsymbol{x}_{i,g+1} = \begin{cases} \boldsymbol{u}_{i,g}, & \text{if } f\left(\boldsymbol{u}_{i,g}\right) \le f\left(\boldsymbol{x}_{i,g}\right) \\ \boldsymbol{x}_{i,g}, & \text{otherwise.} \end{cases} \tag{7}$$

The performance of DE depends on the mutation operator and also the F and CR parameters. Thus, many DE variants have been proposed to improve mutation and parameters.

B. Mutation Strategy Variants

The mutation strategy mutates the corresponding parent with difference vectors, so the selection of the parents affects the performance of DE. Different methods are proposed to select the parents from: neighbor list constructed by different population topologies [12], archive with successful solutions [32], probability distribution based on proximity characteristics among the individuals [33] or fitness ranking [16], [34], [35], to improve DE.

Since mutation strategies are used to produce promising candidate solutions, other methods that do not use a scheme of "base vector + difference vectors" can also be utilized in DE. For example, opposition-based DE (ODE) employs opposition-based learning method to create new offspring to increase the chance of finding good solutions [36]. A stochastic ODE is further developed by using a beta distribution to increase the population diversity [37]. Gaussian sampling method is also used to generate the mutation vector in Gaussian bare-bones DE [38]. In addition, the combination of mutate operators in different algorithms is also promising. For example, the mutate operator of CMA-ES is combined with that of DE to enhance the explorative power of the algorithm [39].

As different mutation strategies have different characteristics, multiple strategies method is promising. Wang *et al.* [28] proposed a composite DE (CoDE), in which three strategies with randomly selected parameters generate three mutant vectors and only the best trial vector is compared with the target vector to determine which one to enter the next generation. Similarly, in [40], multiple strategies generate multiple solutions first and then a surrogate model is used for quality estimation to select one solution as the candidate. In [41], four mutation strategies are used to generate trial vectors. In [24], the individuals with different fitness values employed specific mutation strategy in different evolutionary stages.

Based on assembling multiple strategies, feedback from the evolutionary search can be used to dynamically change the strategy selection probability. Qin *et al.* [18] developed an adaptive DE (SaDE), wherein the mutation strategy is determined according to a probability model calculated by previous successful experiences. The strategy with higher SR in the last generation has more opportunities to be adopted. All the individuals share the same probability model. Fialho *et al.* [20] used fitness improvements to evaluate the effect of strategy and proposed probability matching and adaptive pursuit method to adaptive select mutation strategy. Gong *et al.* [19] employed the similar strategy selection approach according to the recent effect of the strategies. Mallipeddi *et al.* [9] proposed an ensemble of strategies and control parameter with DE (EPSDE), wherein a pool of strategies and multiple parameter values are combined to generate the offspring. The successful combinations are reserved and are more probable to be selected. Despite the progress so far, room exists to improve DE further.

C. Parameter Adaptation Variants

The jDE [11] directly propagates the successful parameter values of F and CR into the next generation to improve the performance of DE. DE-DPS [15] selects parameters from the parameter combination (Np, F, and CR) pool according to their SR in the past period. Indirect learning is another adaptation method. JADE [13], MDE_pBX [16], and SHADE [14] generate parameters F and CR according to a distribution with mean value learning from successful parameters in the previous evolution process. The DE variants mentioned above adaptively update parameters by learning from history of successful parameters in the previous evolution process. Differently, in IDE [24], parameters are assigned for each individual according to its fitness ranking.

III. HHDE

In this section, we first discuss the convergence features of DE with different mutation strategies and propose a new HHM strategy adaptation method. Then, we discuss the influence of the control parameters in DE and design a new HHP setting mechanism. Finally, we present the complete procedure of HHDE.

A. HHM for Mutation Strategy Adaptation

In mutation strategy adaptation, historical experience of the population is an important learning source. Moreover, different individuals in different states need different mutation strategies to improve themselves. Motivated by these considerations, we propose an HHM strategy that uses both the population historical experience and the individual heuristic information to select a suitable mutation strategy for each individual.

Different mutation strategies have different characteristics on exploitation and exploration abilities. The "DE/best/k" schemes search in a small neighborhood of the best solution and have a high convergence speed. They perform well on the unimodal functions but easily get trapped in local optima on multimodal functions [50]. Differently, "DE/current-to-/k" schemes move toward other individuals and have higher diversity and stronger exploration ability than DE/best/k [13]. "DE/rand/k" schemes select the base vector randomly from the current population and move in the solution space aimlessly. They always have strong exploration ability [15].

In the HHM, we maintain a strategy candidate pool including three mutation strategies.

Strategy 1 (DE/current-to-rand/1):

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{i,g} + \lambda \cdot \left(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{i,g}\right) + F_i \cdot \left(\boldsymbol{x}_{r2,g} - \boldsymbol{x}_{r3,g}\right) \tag{8}$$

where the value of λ is set as F_i to eliminate one additional parameter.

Strategy 2 (DE/current-to-pbest/1):

$$\begin{aligned} \boldsymbol{v}_{i,g} &= \boldsymbol{x}_{i,g} + F_i \cdot \left(\boldsymbol{x}_{p\text{best},g} - \boldsymbol{x}_{i,g}\right) + F_i \cdot \left(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g}\right) \\ & p \in [0, 0.1 \cdot Np]. \end{aligned} \tag{9}$$

Strategy 3 (DE/rand/1):

$$\boldsymbol{v}_{i,g} = \boldsymbol{x}_{r_1,g} + F_i \cdot \left(\boldsymbol{x}_{r2,g} - \boldsymbol{x}_{r3,g}\right), \ \ r_1 \in \left[0.1 \cdot Np, 0.5 \cdot Np\right] \tag{10}$$

where $\boldsymbol{x}_{r_1,g}$ is randomly selected between the top 10% and the top 50% individuals in the population. Note that the individuals in the population are sorted based on their fitness values before the mutation operation.

In the evolutionary process, for each individual, one strategy will be chosen from the candidate pool according to HHM that considers both HEM and HIM. First, the HEM learning in HHM means the historical SR of different strategies learning from the population history, where a more successful strategy has a higher chance to be selected. Second, the HIM learning in HHM means the preference of different strategies possessed by the individual according to its current state. The SR in the past history reflects the suitability of the strategy to the problem, while preference describes the matching degree between the characteristic of strategy and the current state of the individual. The combination of historical experience and heuristic information is described as follows.

1) HEM Learning in HHM: The historical SR τ of different strategies accumulates as the evolution promotes. The τ_j of mutation strategy j is calculated as

$$\tau_j = (1 - \varepsilon) \cdot \tau_j + \varepsilon \cdot \Delta\tau_j \tag{11}$$

$$\Delta\tau_j = \frac{S_{j,g}}{\sum_{k=1}^{K} S_{k,g}} \tag{12}$$

$$S_{k,g} = \frac{ns_{k,g}}{ns_{k,g} + nf_{k,g}} + \delta, \ \ (k = 1, 2, \ldots, K) \tag{13}$$

where $S_{k,g}$ is the SR of strategy k in generation g, $\mathrm{ns}_{k,g}$ is the number of successful individuals which employed strategy k in generation g, $nf_{k,g}$ is the number of failed individuals which employed strategy k in generation g, $\delta = 0.05$ is a constant to avoid the possible null SRs, K is the number of mutation strategies, and ε is used to balance the effect of previous and current experiences. The τ_j of each strategy j is initialized as $1/K$. In each generation, the τ value of the strategy with higher SR increases more. Thus, the more suitable strategy in the current evolutionary stage will be accumulated to a higher value.

2) HIM Design in HHM: We use the ranking of the fitness to classify the individuals into three different states.

1) *Superior State (S):* The top one-third individuals with best fitness values.
2) *Medium State (M):* One-third individuals with medium fitness values.
3) *Inferior State (I):* The last one-third individuals with worst fitness values.

For individuals in S, they are so close to the best solutions and easy to find proximity by learning from the better solutions. Therefore, the "DE/current-to-rand/1" strategy that explores promising areas around is more suitable to avoid premature convergence. For individuals in M, they have great potential to improve by learning from better solutions and require a strategy with adequate convergence speed and strong exploitation capability. Thus, the "DE/current-to-$p$best/1" is more suitable. For individuals in I, they can learn from "second" better (not the top but better than solutions in I) individual to obtain better solutions to get closer to good solution as well as explore new area to increase population diversity. The "DE/rand/1" strategy has strong exploration ability [15]. Employing this strategy, the search is able to cover the neighborhood areas of potentially better solutions to drive the population forward steadily and reliably.

Thus, the heuristic information η_{ij} of individual i for mutation strategy j can be described as

$$\eta_{ij} = \begin{cases} 0.5, & \text{if } \left(1 \le r_i \le \frac{1}{3}Np \text{ and } j = 1\right) \text{ or} \\ & \left(\frac{1}{3}Np < r_i \le \frac{2}{3}Np \text{ and } j = 2\right) \text{ or} \\ & \left(\frac{2}{3}Np < r_i \le Np \text{ and } j = 3\right) \\ 0.25, & \text{otherwise} \end{cases} \tag{14}$$

where $1 \le i \le Np$, $1 \le j \le 3$, and r_i is the rank of individual i in the current population according to the fitness value (from best to worst). That is, strategy 1 (DE/current-to-rand/1) is more suitable for individuals in state S, strategy 2 (DE/current-to-$p$best/1) is more suitable for individuals in state M, while strategy 3 (DE/rand/1) is more suitable for individuals in state I. Therefore, these heuristic information values are set to 0.5, while all the other values are set to 0.25.

3) Combine HEM and HIM in HHM: With respect to each individual, one mutation strategy is selected from the candidate pool according to the probability learning from historical SR and the heuristic information. The individual i selects a mutation strategy k_i by applying the rule given by

$$k_i = \begin{cases} \operatorname{argmax}_{j=1,\dots,K}\{\tau_j \cdot \eta_{ij}\}, & \text{if } q \le q_0 \\ T, & \text{otherwise} \end{cases} \tag{15}$$

where q is a random number uniformly distributed in [0,1], and q_0 is used to balance the importance of experience and heuristic versus exploration. If $q \le q_0$, then the "best" strategy with maximal $\tau_j \cdot \eta_{ij}$ value is chosen, otherwise, a strategy T is chosen according to the roulette wheel selection by probability distribution

$$p_{ij} = \frac{\tau_j \cdot \eta_{ij}}{\sum_{k=1,\dots,K} \tau_k \cdot \eta_{ik}}. \tag{16}$$

This way, the individual favors the strategy with a higher SR and a larger heuristic value that combines both HEM and HIM.

B. HHP for Parameter Adaptation

The scale factor F is the step size of movement, and the crossover rate CR influences the variation of the new generated trial vector. The parameters are problem dependent [13]. On the one hand, the previous successful parameters tent to generate individuals that are more likely to survive and thus they are believed to be more suitable for this problem. On the other hand, different individuals with different solution quality requires different parameters. For good solutions, better offspring individuals are probable to be obtained by exploiting the neighborhood with small F value and introducing less mutant components with a small CR value. For poor solutions, more promising offspring individuals are likely to be obtained by exploring the new area with a large value of F and adding more mutant elements with a large value of CR [24].

Motivated by these two influences, the HHP is shown as (17) and (18) by the consideration of both population experience (i.e., HEM) and individual information (i.e., HIM) to determine the parameters. For each individual i, the parameters are generated by Gaussian distribution with standard deviation specified to 0.1. The mean values of Gaussian distribution for F_i and CR_i are set to the weighted sum of population success parameter experience (i.e., the HEM μ_F and μ_{CR}) and individual ranking information (i.e., the HIM r_i).

It should be noted that, we also record the F and CR values that individual i succeeded in the latest time as pF_i and pCR_i, respectively. On the one hand, these can be regarded as some kind of historical experiences for this individual. On the other hand, since the current search area is near to the recently visited areas, they may have quite similar landscape. Therefore, these can also be regarded as some kind of heuristic information related to the search state. Hence, we also let the parameters F_i and CR_i set as pF_i and pCR_i controlled by probability p_0 as shown in (17) and (18)

$$F_i = \begin{cases} \text{Gaussian}\left((1-\alpha)\cdot\mu_F + \alpha\cdot\frac{r_i}{NP}, 0.1\right), & \text{if rand}(0,1) \le p_0 \\ pF_i, & \text{otherwise} \end{cases} \tag{17}$$

$$CR_i = \begin{cases} \text{Gaussian}\left((1-\alpha)\cdot\mu_{CR} + \alpha\cdot\frac{r_i}{NP}, 0.1\right), & \text{if rand}(0,1) \le p_0 \\ pCR_i, & \text{otherwise} \end{cases} \tag{18}$$

where p_0 is a parameter ($0 \le p_0 \le 1$), and the balance factor α increases over generations as

$$\alpha = 0.9 - 0.9 \cdot 10^{-5 \cdot g/G_{\max}} \tag{19}$$

where g is the generation number and $G_{\max}$ is the maximum number of generation. In the early stage, the individuals start without obvious difference, and the guidance of population experience helps accelerate convergence. Later, the difference of evolution ability among individuals appears. Different parameter requirements in different individuals also emerge. Thus, the heuristic information (i.e., the r_i) plays a dominant role and allows the algorithm to jump out of the population experience and also adaptively for each individual. Note that the parameter control in JADE [13] is a special case of HHP, where $\alpha = 0$ and $p_0 = 1$. JADE uses only one distribution constructed by population experience to generate parameters for all individuals. In contrast, HHDE further introduces individual's own heuristic information and creates different distributions for different individuals. The population's experience is therefore effectively utilized to adapt the individuals.

The last note of (17) and (18) is that the μ_F and μ_{CR} are calculated based on the successful parameters of the population in the last generation. The successful parameters are stored in the set S_F and S_{CR}, and they are used for calculating μ_F and μ_{CR} as (20) and (21), respectively. If the S_F and S_{CR} are empty, the μ_F and μ_{CR} remain unchanged. To consider the contribution of different parameters, the fitness improvement is also introduced as a weight as (22) in the calculation of μ_F and μ_{CR}

$$\mu_F = (1-C)\cdot\mu_F + C\cdot\frac{\sum_{F_i\in S_F} w_i F_i^2}{\sum_{F_i\in S_F} w_i F_i} \tag{20}$$

$$\mu_{CR} = (1-C)\cdot\mu_{CR} + C\cdot\sum_{CR_i\in S_{CR}} w_i CR_i \tag{21}$$

$$w_i = \frac{\Delta f_i}{\sum_{F_k\in S_F}\Delta f_k} \tag{22}$$

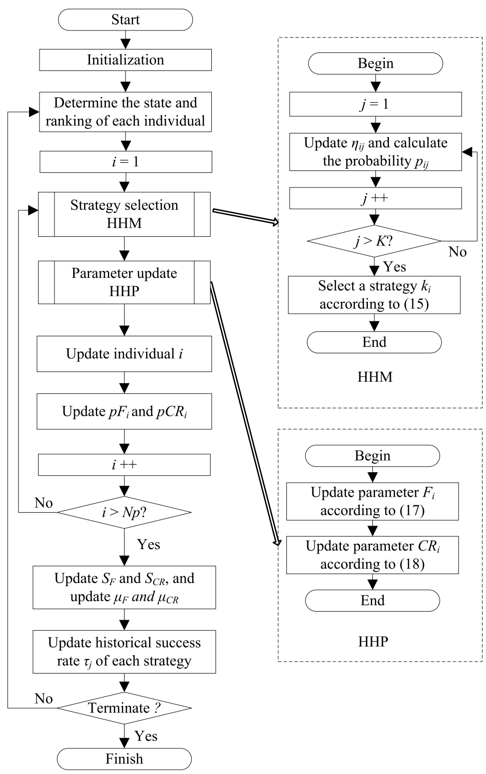

Fig. 1. Flowchart of HHDE.

where C is set as 0.1, μ_F and μ_{CR} are initialized as 0.5, and Δf_i represents the fitness improvement of individual i.

C. Complete Procedure of HHDE

Combining the HHM and HHP with DE, the HHDE is presented with the flowchart in Fig. 1. The complete procedure is described as follow.

Step 1 (Initialization): Set the parameters population size Np, and maximum generation number $G_{\max}$. Randomly initialize the population. Set the generation $g = 1$.

Step 2 (Mutation): For each individual, calculate the historical SR and heuristic value of each strategy, and select a strategy according to HHM as (15). Update the scale factor F of each individual according to HHP as (17). Then generate mutant individuals by mutation operation. Handle the corresponding components if the mutant individual violates the boundary constraints.

Step 3 (Crossover): For each individual, a trial one is generated by adopting binomial crossover operation on the parent and mutant. The CR values of each individual are generated by HHP as (18).

Step 4 (Selection): Select a better one between the parent and the trial individual to enter the next iteration. Record the successful parameters and update the values of μ_F and μ_{CR} defined in (20) and (21).

Step 5 (Termination Detection): When the maximum number of generations is reached, the algorithm terminates. Otherwise, set $g = g + 1$ and return to step 2 for the next generation.

TABLE I
COMPARISON ALGORITHMS

DE variants	jDE [11]	self-adaptive parameters
	SaDE [18]	self-adaptive parameter and mutation strategies selection learning from the history experience
	JADE [13]	adaptive parameters and extern archive
	CoDE [28]	random parameters and random mutation strategy
	EPSDE [9]	ensemble of mutation strategies and parameters
	SHADE [14]	success-history based scheme
	DPS_DE [15]	dynamic parameters selection
	IDE [24]	individual dependent parameters and mutation strategy
Top methods in CEC competition	L-SHADE [43]	CEC 2014 winner, improved SHADE [14] with a population size linear reduction
	UMOEAs [44]	United multi-operator evolutionary algorithms with genetic algorithm, DE, and evolution strategy
	MVMO-SH [45]	Mean-variance mapping optimization

IV. EXPERIMENTS AND COMPARISONS

In this section, experiments are carried out to evaluate the HHDE performance. We employ the widely used benchmark suite in CEC 2014 competition [29]. These minimization optimization problems own different characteristics. F_1–F_3 are nonseparable unimodal functions, F_4–F_{16} are simple multimodal functions, F_{17}–F_{22} are hybrid functions in which different subcomponents of the variables have different properties, and F_{23}–F_{30} are composition functions which merge the properties of the subfunctions.

The performance of the algorithm is evaluated by the fitness error value between the obtained best solution and the optimum. The results are reported based on 30 independent runs. The Wilcoxon signed rank test with a significance level of 5% is also performed to obtain a meaningful statistical comparison [42]. Three symbols "+, −, =" indicate that HHDE displays significantly better, worse than, or equal to the competitor, respectively. Due to space limitation, only the statistical results are presented in this paper, while the detailed results are presented in the supplementary file.

The dimensions of the problems are 10, 30, and 50, and hence the corresponding population sizes for HHDE are set increasingly to 50, 75, and 100, respectively. In this paper, the fixed parameters are set as $q_0 = 0.2$, $p_0 = 0.7$, $\varepsilon = 0.4$, and $C = 0.1$ and are also investigated in this section. To be fair and reliable, the compared algorithms adopt the parameter configurations in their original papers. The maximum function evaluations (FEs) is set as $D \times 10\,000$ (D is the dimension of the problem).

A. Comparison With State-of-the-Art DE Algorithms

In this section, we compare the proposed HHDE with eight state-of-the-art DE variants in Table I. These algorithms adopt different strategy and parameter selection mechanisms and are commonly used for comparison.

The detailed results of these DE variants on 10-D, 30-D, and 50-D problems are reported in Table S.I–S.III, in the supplementary material. The comparisons are summarized in Table II

TABLE II
COMPARISON RESULTS OF HHDE WITH STATE-OF-THE-ART DE VARIANTS ON CEC 2014

Algorithms	10-*D* +	10-*D* =	10-*D* −	30-*D* +	30-*D* =	30-*D* −	50-*D* +	50-*D* =	50-*D* −
jDE	20	7	3	26	4	0	21	3	6
SaDE	21	8	1	25	5	0	23	5	2
JADE	19	10	1	20	5	5	15	11	4
CoDE	10	13	7	17	9	4	14	10	6
EPSDE	17	10	3	27	2	1	28	2	0
SHADE	12	16	2	22	2	6	19	5	6
DE-DPS	16	10	4	19	9	2	20	5	5
IDE	15	13	2	22	7	1	24	3	3

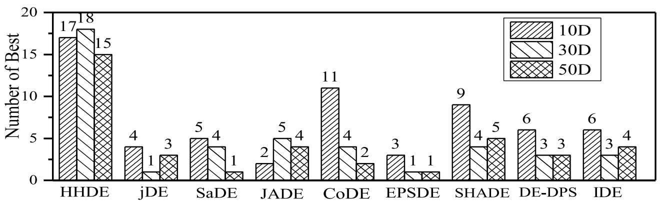

Fig. 2. Number of best cases obtained by each algorithm on 10-D, 30-D, and 50-D problems in the comparison with state-of-the-art DE algorithms.

and the number of best cases obtained by each algorithm is depicted in Fig. 2. It can be observed that on 10-D problems, HHDE performs the best for 17 out of the 30 functions, obtaining significantly better solutions than jDE, SaDE, JADE, CoDE, EPSDE, SHADE, DPS_DE, and IDE on 20, 21, 19, 10, 17, 12, 16, and 15 cases, while significantly worse results on only 3, 1, 1, 7, 3, 2, 4, and 2 cases, respectively. HHDE performs the best among all the competitors. On the 30-D and 50-D problems, the superiority of HHDE to the other variants is more significant. The proposed algorithm performs the best on 18 and 15 cases for 30-D and 50-D problems, respectively, while these numbers for other competitors are smaller than five functions. Moreover, the number of significantly better results is much larger than that of significantly worse results, when comparing HHDE with other variants. For convenience of illustration, we plot the curves of the convergence of HHDE and DE variants on 30-D problems F_5, F_6, F_{17}, F_{22}, F_{29}, and F_{30} in Fig. S.1, in the supplementary material. The curves suggest that the HHDE converges faster compared with other algorithms.

To further test the performance of HHDE, we compare HHDE with four recent algorithms with multiple strategies and parameter adaptation control, CSM-JADE [40], SaJADE [22], MPEDE [21], and JADE_sort [25], on 30-D problems. Among them, CSM-JADE ensembles multiple strategies, SaJADE uses historical information for strategy adaptation, MPEDE automatically adjusts the computation assignment for different strategies, and JADE_sort uses population historical information and individual heuristic information in a stepwise manner for parameter adaptation. The detailed results are presented in Table S.IV, in the supplementary material. From Table S.IV, in the supplementary material, we can see that HHDE obtains the best values on 13 functions, which is much larger than the number of best cases that the other four algorithms can obtain. That is, CSM-JADE, SaJADE, MPEDE, and JADE_sort can achieve the best values on only 2, 7, 4, and 8 functions, respectively. According to the significance test, HHDE performs better than CSM-JADE, SaJADE, MPEDE, and JADE_sort on 23, 19, 15, and 17 cases, while worse on only 7, 6, 6, and 9 cases, respectively. Thus, HHDE generally outperforms the compared algorithms.

B. Comparison With Top Methods in CEC Competition

Top methods in CEC 2014 competition described in Table I are compared with HHDE in this section. The codes are downloaded from http://www.ntu.edu.sg/home/epnsugan/. The winner, L-SHADE [43], is an improved version of SHADE [14] with linear population size reduction. UMOEAs is a united multioperation algorithm using genetic algorithm, DE, and evolution strategy [44]. MVMO-SH evolves each particle according to its own search experience and implements mutation by mapping function [45]. The detailed results are reported in Table S.V, in the supplementary material, where the best and second best results of each function are marked in **boldface** and *italic*, respectively. The results show that L-SHADE and HHDE can obtain the best or second best results on not less than 20 out of 30 functions, while UMOEAs and MVMO-SH can only obtain on 12 and 7 functions, respectively. Therefore, HHDE and L-SHADE outperform UMOEAs and MVMO-SH. Specially, on unimodal functions, UMOEAs performs the best and HHDE performs well on F_2 and F_3 except F_1. On simple multimodal functions, L-SHADE performs the best on seven functions while HHDE performs the best on three functions. It is interesting to observe that the best results obtained by HHDE on these three functions are all the global optima (i.e., 0) of F_6, F_7, and F_8. More significantly, only HHDE can find the global optima on F_6 and F_8, while all the other three algorithms fail. On hybrid functions, L-SHADE gets the best values except F_{19}. HHDE achieves the best value on F_{19} and slightly worse results on other functions. However, HHDE shows comparable performance with L-SHADE on composition functions and even outperforms L-SHADE on the very complex composition functions F_{28}, F_{29}, and F_{30}. Therefore, HHDE presents competitive performance with L-SHADE, UMOEAs, and MVMO-SH, even though they may have different advantages on different problems.

As an improved version of L-SHADE with ensemble sinusoidal parameter adaptation (EpSin), LSHADE-EpSin [46], takes the joint-winner in the CEC 2016 competition. Except parameter control, LSHADE-EpSin also integrates with linear population size reduction and a local search. To further test the performance of the strategy and parameter control mechanisms, we compare LHHDE with L-SHADE and LSHADE-EpSin, where LHHDE is a variant of HHDE embedded the linear population size reduction and local search in LSHADE-EpSin. Herein, we do not directly compare HHDE with LSHADE-EpSin, so as to avoid the influences of the population size setting and local search. The detailed results are reported in Table S.VI, in the supplementary material. We can see that on the first 16 unimodal and simple multimodal functions, LSHADE-EpSin and L-SHADE performs better

TABLE III
COMPARISON RESULTS OF STATISTICALLY SIGNIFICANT DIFFERENCES OF HHDE WITH ITS VARIANTS ON 30-D CEC 2014 BENCHMARK SET

Fun	HHDE vs Its Variants																													
	Parameter setting						Mutation strategy selection									Heuristic information design in HHM														
	noHEM-P			noHIM-P			noHEM-M			noHIM-M			noHHM			HHDE-231			HHDE-213			HHDE-321			HHDE-312			HHDE-132		
	+	=	−	+	=	−	+	=	−	+	=	−	+	=	−	+	=	−	+	=	−	+	=	−	+	=	−	+	=	−
F_1-F_3	1	2	0	2	0	1	0	3	0	0	3	0	3	0	0	0	3	0	0	3	0	0	3	0	0	3	0	0	3	0
F_4-F_{16}	9	4	0	8	3	2	4	9	0	5	6	2	11	1	1	5	8	0	3	10	0	6	6	1	5	5	3	3	9	1
F_{17}-F_{22}	4	1	1	5	1	0	2	3	1	3	2	1	5	1	0	4	2	0	3	3	0	1	4	1	2	4	0	1	4	1
F_{23}-F_{30}	3	3	2	6	2	0	1	6	1	5	3	0	4	2	2	3	4	1	1	6	1	5	3	0	3	5	0	3	5	0
total	17	10	3	21	6	3	7	21	2	13	14	3	23	4	3	12	17	1	7	22	1	12	16	2	10	17	3	7	21	2

than LHHDE. However, on the last 14 hybrid and composition functions, LHHDE shows competitive performance with LSHADE-EpSin and L-SHADE. Both LHHDE and LSHADE-EpSin obtain the best or second best results on 11 out of these 14 functions while L-SHADE gets on only five functions. Moreover, according to the significance test, LHHDE performs better than L-SHADE and LSHADE-EpSin on 8 and 2 out of these complex functions, while worse on only 1 and 2 functions, respectively.

C. Evaluation on Different Components of HHDE

The mutation strategy selection and parameter setting are two important components of HHDE. In this section, we identify the benefit of the two components by comparing the performance of HHDE with its variants. The variants are named as noa-b (noa is "noHIM" or "noHEM," and *b* is "*P*" or "*M*"), where *P* and *M* mean variants for parameter or mutation strategy selection, while noHIM and noHEM mean that there is no heuristic information mechanism or historical experience mechanism in the selection of b, respectively.

Due to space limit, the detailed results are reported in Table S.VII, in the supplementary material and the statistical significance results are summarized in Table III.

1) Effect of Parameter Setting HHP: We consider two variants of HHDE, i.e., noHIM-P and noHEM-P to denote the HHDE variants which generate the parameters only according to the HEM (set the value $\alpha = 0$) learning from historical experience or the HIM (set the value $p_0 = 1.0$ and $\alpha = 1$) with heuristic information HHP.

From Table S.VII, in the supplementary material and Table III, we can see that HHDE yields significantly better results on 21 and 17 functions than noHIM-P and noHEM-P, respectively, which are much larger than the number of significantly worse results (three functions). The convergence curves on F_2, F_5, F_{17}, and F_{30} are plotted in Fig. S.2, in the supplementary material. We can see that noHIM-P always converges faster than noHEM-P. It is interesting to find that noHIM-P always performs well on simple unimodal and multimodal functions (F_1 to F_{16}), e.g., significantly outperforms HHDE on unimodal F_1, and multimodal F_{10} and F_{11}. The noHEM-P always performs well on complex hybrid and composition functions (F_{17} to F_{30}), e.g., significantly outperforms HHDE on hybrid F_{19} and composition F_{23} and F_{25}. This phenomenon may be due to that HEM implies the characteristic of the problem and is suitable for simple functions which need certain parameters for the whole population. However, complex problems F_{17} to F_{30} have different properties in different subcomponents. The individuals in different states require quite different parameters for the evolution of different subcomponents. Exactly, the heuristic information helps adaptively match the individual's state. Overall, HHDE outperforms both noHIM-P and noHEM-P on the four kinds of problems, which verifies the benefit of the combination of the learning of the success history (i.e., HEM) and current state of each individual (e.g., HIM).

2) Effect of Mutation Strategy Selection HHM: To verify the effect of the HHM strategy, we denote the HHDE variants without HIM design ($\eta_{ij} = 1$) in HHM, without HEM learning ($\tau_j = 1$) in HHM, and without HHM strategy ($\eta_{ij} = 1$, $\tau_j = 1$) as noHIM-M, noHEM-M, and noHHM, respectively.

From Table III, HHDE performs significantly better on 7, 13, and 23 cases but only significantly worse on 2, 3, and 3 cases than noHEM-M, noHIM-M, and noHHM, respectively. The noHHM performs the worst, which shows that the proposed HHM mutation strategy selection plays a very significant role in HHDE. For the unimodal functions F_1–F_3, noHIM-M, and noHEM-M obtain results without significant difference between HHDE, while noHHM obtains significantly worse results. For the multimodal functions F_4–F_{16}, it is clear that HHDE outperforms the other variants with significant difference in 4, 5, and 11 cases, respectively. HHDE achieves similar performance to noHEM-M and noHIM-M in most cases. On the hybrid functions F_{17}–F_{22}, HHDE performs significantly better than noHEM-M, noHIM-M, and noHHM on 2, 3, and 5 functions, respectively. On the composition functions F_{23}–F_{30}, these figures are 1, 5, and 4, respectively. Overall, noHEM-M performs better in two cases on the complex problem F_{17}–F_{30} but none on simple functions, while the noHIM-M gets better results on two simple multimodal functions. It can be seen that the heuristic information is crucial for solving the hybrid and composition problems. It may be the reason that the heuristic information makes the population evolve in a more directional and ordered way and maintains the population diversity. The historical experience and heuristic information in HHM strategy are both essential to improve the performance of HHDE.

3) Effect of Heuristic Information Design in HHM: For a further study of HHM, the variants with different heuristic information design are performed for comparison. To abbreviate the name of variants, we denote the variants as HHDE-$n_1n_2n_3$ ($n_1n_2n_3$ are a permutation of number 1, 2, 3), which means that the individuals in state S prefer mutation strategy n_1 and its corresponding heuristic information is set as 0.5 for n_1 but 0.25 for n_2 and n_3; the individuals in states M and I set the heuristic information as 0.5 for mutation strategy n_2 and n_3, respectively, and 0.25 for the other strategies.

The five variants are termed as: 1) HHDE-231; 2) HHDE-213; 3) HHDE-321; 4) HHDE-312; and 5) HHDE-132.

Table III shows that the design of the heuristic information in HHDE does make the search more effective. HHDE performs significantly better than HHDE-231, HHDE-213, HHDE-321, HHDE-312, and HHDE-132 on 12, 7, 12, 10, and 7 functions while worse on only 1, 1, 2, 3, 2 functions, respectively. On the unimodal functions, all the algorithms perform similarly. However, HHDE performs significantly better than the variants on simple multimodal, hybrid, and composition functions. This should be the reason that the design of heuristic information "123" can increase the population diversity. The current-to-*p*best/1 and rand/1 learn from different "better" individuals in the population; so the adoption of these two strategies in states M and I drive the whole population forwards to avoid big difference between superior and inferior individuals. As illustrated in Fig. S.2, in the supplementary material, different heuristic information design methods have different convergence features on different kinds of problems. HHDE-213 performs better than HHDE-312 on F_5 and F_{30} but worse on F_{17}. HHDE-321 yields the worst solutions on F_5 and F_{30} while HHDE-231 performs the worst on F_{17}. It can be observed that HHDE-$n_1n_2$2 and HHDE-$n_1n_2$3 variants often achieve better solutions than HHDE-$n_1n_2$1 on most cases, which shows that the adoption of strategy with good convergence for inferior individuals is benefit to the evolution of HHDE.

D. Discussion on HHDE

In this section, we discuss the evolution behavior of the HHDE. We take four 30-D functions from four types of problems as examples: 1) a unimodal function F_2; 2) a multimodal function F_5; 3) a hybrid function F_{17}; and 4) a composition function F_{30}. It should be noted that the curves are plotted according to the same run that is randomly selected from the 30 runs unless otherwise specified. We do not use error curves based on 30 runs in order to present an entire evolution behavior of HHDE with internal logic, since curves of 30 runs are messy.

1) Evolution of μ_F and μ_{CR} in HHDE: For each individual, the HHP setting assigns parameters for each individual by "shifting" the historical successful parameters to adapt to the current state of the individual, such that each individual tends to be assigned parameters matching the problem as well as the state of itself. To observe the changing trends of the suitable parameters, the curves of μ_F and μ_{CR} defined as (20) and (21) in HHDE, JADE, and IDE are plotted in Fig. 3. In Fig. 3(a) for F_2, μ_F and μ_{CR} of JADE kept constant after obvious initial changes due to the stagnation of the whole population, and IDE was almost constant after 150 000 FEs while the value of HHDE moved slowly toward the increasing direction after the early drastic change. Similar situation can be observed on F_5. The CR value of JADE decreased to 0.2 and maintained steady later; the F and CR values of IDE appeared a similar changing trend that they both decreased early but increased in the later stage; while μ_F and μ_{CR} of HHDE waved in the decreasing direction to small values. For hybrid function F_{17},

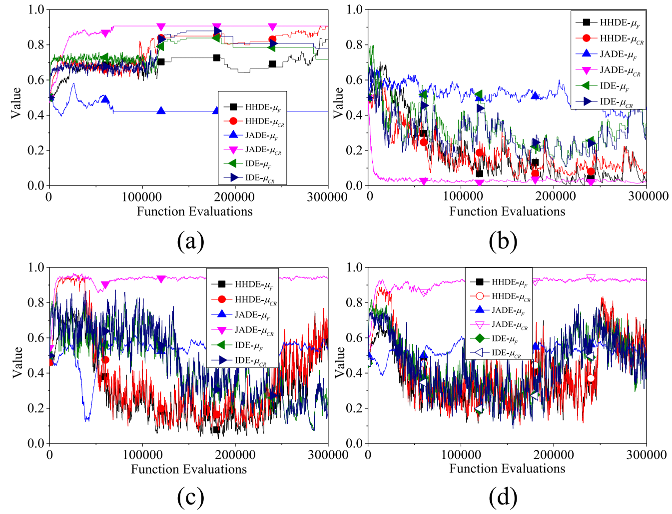

Fig. 3. Evolutions of μ_F and μ_{CR} in HHDE, JADE, and IDE in the optimization. (a) F_2, (b) F_5, (c) F_{17}, and (d) F_{30} ($D = 30$).

μ_F, and μ_{CR} of HHDE evolved to different values for different subcomponents from range [0.5, 0.9] to [0.1, 0.3] and stayed in range [0.3, 0.7] at last. For F_{30} composed by three hybrid functions, the μ_F and μ_{CR} of HHDE and IDE were in a similar changing trend, which was consistent with their nonsignificantly difference results.

For simple problems F_2 and F_5, the suitable parameters increase or decrease in one direction. Differently, on complex problems F_{17} and F_{30}, parameters require various changing directions during the evolution process. The historical experience indicates the possible successful parameter direction while the introduction of individual characteristic increases the diversity of the parameters. The combination of these two learning mechanisms makes the HHDE able to jump out of the historical experience when it is not suitable for the current evolution stage, and converge to better solutions faster.

2) Parameter and Population Diversity in HHDE: The diversity is defined as the standard deviations of the elements in the population. The diversity of parameters (DP), individuals (DI), and fitness (DF) are calculated as

$$\mathrm{DP} = \frac{1}{NP}\sqrt{\sum_{i=1}^{NP}\left(\left\|F_i - \frac{1}{NP}\sum_{j=1}^{NP}F_i\right\|^2 + \left\|CR_i - \frac{1}{NP}\sum_{j=1}^{NP}CR_i\right\|^2\right)} \tag{23}$$

$$\mathrm{DI} = \frac{1}{NP}\sqrt{\sum_{i=1}^{NP}\left(\left\|x_i - \frac{1}{NP}\sum_{j=1}^{NP}x_i\right\|^2\right)} \tag{24}$$

$$\mathrm{DF} = \frac{1}{NP}\sqrt{\sum_{i=1}^{NP}\left(\left\|f_i - \frac{1}{NP}\sum_{j=1}^{NP}f_i\right\|^2\right)}. \tag{25}$$

The curves of DP, DI, and DF are plotted in Fig. 4. It can be observed that the DP value increases as evolution process promotes because heuristic information contributes more to the parameter setting in the later stages. The DI value decreased

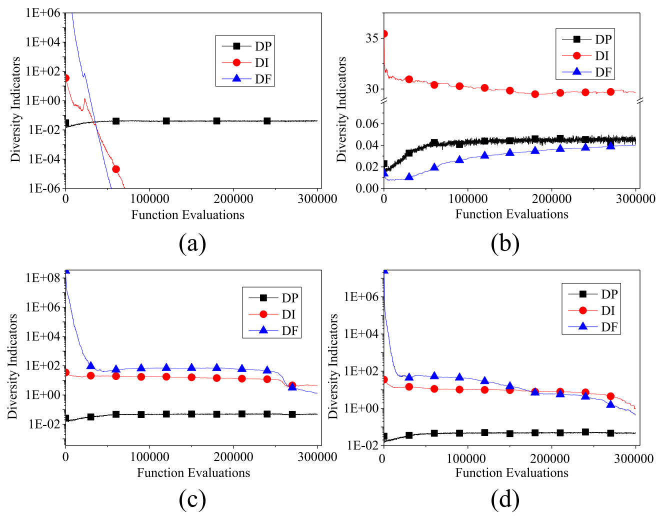

Fig. 4. Curves of the DP, DI, DF in HHDE on four 30-D problems. (a) F_2. (b) F_5. (c) F_{17}. (d) F_{30}.

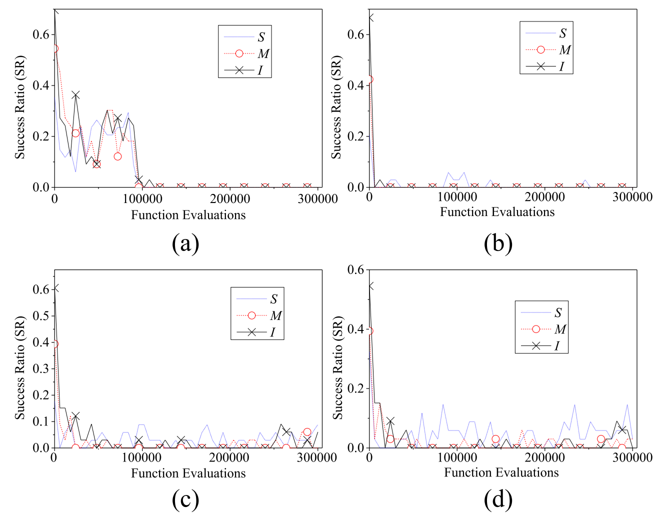

Fig. 6. SR of individuals in different states S, M, and I in HHDE on four 30-D problems. (a) F_2. (b) F_5. (c) F_{17}. (d) F_{30}.

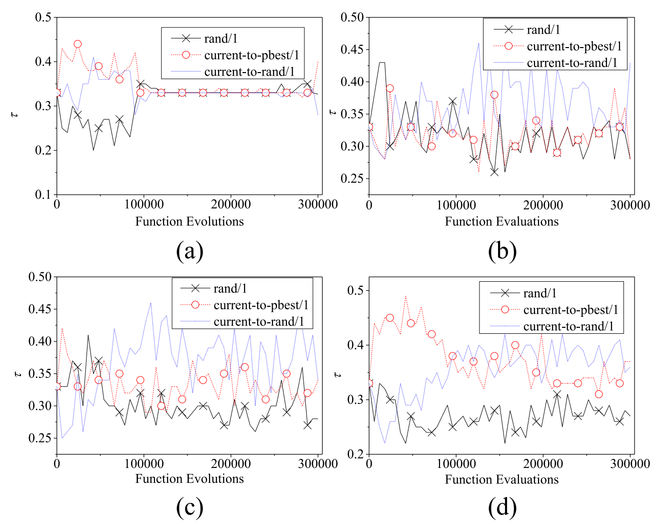

Fig. 5. Self-adaptation behavior of mutation strategy in HHDE in the optimization of (a) F_2, (b) F_5, (c) F_{17}, and (d) F_{30} ($D = 30$).

in early stage but reached stable value in the late stage on F_5, F_{17}, and F_{30} except for F_2 due to its convergence to optimal solution on F_2 (Fig. S.1, in the supplementary material). For F_5, the DF increased along with the evolution. For F_{17}, the DF decreased before 60 000 FEs but increased in the middle stages and decreased at last. In contrast, the DF dropped sharply early but reduced slowly later to about 1.0 on F_{30}, which was consistent with the variety of DI. This phenomenon is mainly caused by the effect of HHM strategy. The heuristic information considers the population distribution and current state of individuals, resulting in the diversity of the population. More discussion will be presented in the next section.

3) Self-Adaptation Behavior of Mutation Strategy in HHDE: To investigate the self-adaptation characteristic of mutation strategies, we plot their relative historical SR curves [τ_j defined in (11)] in Fig. 5. In Fig. 5(a) of F_2, the "current-to-*p*best" scheme occupied more proportion in the early stage because of its rapid convergence for unimodal functions while the τ value of strategy "rand/1" is smaller than 0.33 because of its bad performance. On the contrary, for F_5, the strategy rand/1 demonstrated a good performance at the beginning so as to achieve a high probability in 60 000 FEs but decreased under 0.33 in the later stage. After 60 000 FEs, the proportion of strategy "current-to-rand" slightly increased and occupied a higher probability because it found good solutions. For F_{17}, the strategies current-to-*p*best and current-to-rand performed better at the early stage of 60 000 FEs, and the current-to-rand represented a high proportion after 60 000 FEs. For F_{30}, the strategy current-to-*p*best occupied high proportion while the relative value of strategies current-to-rand and rand/1 were smaller than 0.33 at the beginning 90 000 FEs. After 90 000 FEs, the strategy current-to-rand occupied a higher proportion. The probability of different strategies is self-adaptive in different evolutionary stages for different problems.

The heuristic information redistributes the selection probability of different strategies for individuals in different states. To further observe the effect of the heuristic information, we plot the SR curves of individuals in states S, M, and I in Fig. 6. Note that, to simplify the description, we employ the state name to refer to the individuals in this state in the following. In Fig. 6(a) for F_2, each state had a similar SR, resulting in the advance of the entire population. The high SR value of state M in the early stage corresponds to the highest proportion of strategy current-to-*p*best that is preferred by the individuals in state M. For F_5 in Fig. 6(b), the SR of states M and I decreased to nearly zero from 0.42 and 0.66 in 6000 FEs, respectively, and remained as nearly zero after 30 000 FEs. The SR value of state S was small but large than zero in some stages responding to the highest proportion of strategy current-to-rand. The superior solutions improved but the inferior solutions almost stayed unchanged, so the population fitness diversity increased with the iteration continues as shown in Fig. 4(b). For hybrid function F_{17}, the SR of each state decreased to less than 0.1 after 30 000 FEs. The SR values of states M and I were larger than S in the 60 000 FEs due to the high proportion of strategies "current-to-*p*best/1" and "rand/1," so that the fitness diversity

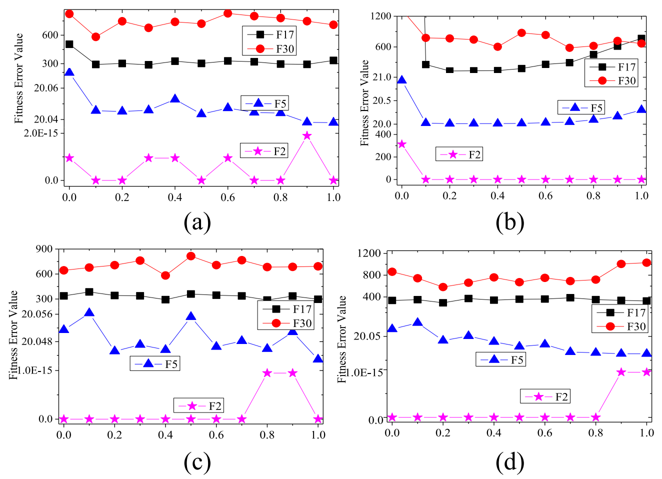

Fig. 7. Influence of the parameters on HHDE. (a) C. (b) p_0. (c) ε. (d) q_0.

value decreased as shown in Fig. 4(c). During about 60 000–240 000 FEs, the SR of S was larger than that of M and I, and agreed with the high proportion of strategy current-to-rand in Fig. 5(c), resulting in the increasing of the fitness diversity in Fig. 4(c). After 240 000 FEs, the SR of I and M was larger than S, and thus the DF decreased. The situation of F_{30} is similar. The SR values of M and I were larger than that of S in 60 000 FEs but reversed in the later stage. The diversity of fitness decreased as evolution proceeded. Even though the SR values of individuals in each state are small and not equal, it is the difference of the SR values that makes HHDE maintain population distribution and the diversity.

4) Parameter Investigation of HHDE: HHDE introduces its own parameters C, p_0, ε, and q_0. It is believed that these parameters are problem insensitive. To investigate the effect of parameters, the four parameters are set from 0 to 1.0 with a step length 0.1. In each parameter configuration, only one parameter is reset while all the other parameters values remain the same as in Section IV-A. The results of HHDE on F_2, F_5, F_{17}, and F_{30} are plotted in Fig. 7.

The investigation begins with C. From Fig. 7(a), we can see that HHDE with $C = 0.1$ performs well on four problems while $C = 0$ leads to a worse solution because of μ_F and μ_{CR} remain 0.5 and do not evolve with the process. For F_5 and F_{17}, HHDE has a similar performance when C is in range of [0.1, 1.0]. A smaller C performs better on F_{30}. Therefore, we set C as 0.1 in this paper.

The second parameter p_0 controls the probability of regenerating a new parameter. In Fig. 7(b), p_0 should not be too large, e.g., up to 0.7. For F_5 and F_{17}, the obtained results increase with the growth of p_0 from 0.7 to 1.0 while decreases in range of [0.1, 0.4] and achieves the best result at value of 0.7 for F_{30}. Moreover, the poor performance of HHDE when p_0 is 0 indicates that the HHP plays an important role in HHDE.

The next parameter tested is ε. The ε controls the updating ratio of newly SR of each mutation strategy in the historical experience evaluation as shown in (11). In Fig. 7(c), HHDE performs better when ε is in range of [0, 0.7] on F_2, [0.2, 0.4] on F_5, [0.3, 0.5] on F_{17}, and [0, 0.4] on F_{30}. A relatively small value is suitable. This indicates that a slow change of historical experience information τ_j is beneficial to mutation strategy selection. This avoids the misguidance of occasional success.

Finally, the parameter q_0 balances the greedy selection or probability selection of mutation strategy. The tendency of the curves in Fig. 7(d) indicates that it is better to use some moderate value around 0.2 for q_0 for better performance. Although the performance of HHDE decreases with the increase of q_0 on F_5, the obtained result is still good when $q_0 = 0.2$. Thus, we set q_0 as 0.2 in this paper.

V. Conclusion

As mutation strategy and parameter selection are influential to improving the performance of DE, many variants have been developed to adaptively update the mutation strategy and parameters using either historical experience from the population or heuristic information from the individuals. However, using both historical experience and heuristic information should provide a more effective improvement. This has motivated our development of the HHDE, a historical and heuristic adaptive DE algorithm. A mutation strategy is assigned to each individual according to a probability model learning from the population's historical experience and heuristic information related to the current state of the individual. The SRs of the strategies in the proceeding evolution process offer a guideline. The heuristic information redistributes the strategy selection probability in the population, so as to match the individual's current state. For the parameters adaptation in HHDE, the historical successful parameters and differences between individuals are both used to set parameters for each individual. Through learning from past successes, *good* parameters are propagated to the next generation. The introduction of the current state of individuals shifts the population experience to adapt to itself. In addition, it increases the DP in the whole population and allows the parameters to "jump" out of the past when the experience is not applicable. Through adjusting strategy and setting parameters from two perspectives, i.e., population experience at a macroscopic level (suitable assignments for the population according to the problem characteristic) and a current state of each individual at a micro level (a preferred choice of individuals for the evolution of the population), HHDE assigns suitable strategy and parameters to each individual in different evolution stages and thus promotes the evolution of population in a collaborative way.

The HHDE has been tested on the CEC 2014 benchmark suite. Experimental results show that the HHDE algorithm can evolve suitable strategies and parameter values during the evolutionary process. It is seen that the proposed algorithm achieves better performance compared with other classic and adaptive DE algorithms. The HHDE also exhibits competitive performance with top methods in CEC 2014 and CEC 2016 competition.

References

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

[2] P. Rakshit, A. Konar, S. Das, L. C. Jain, and A. K. Nagar, "Uncertainty management in differential evolution induced multiobjective optimization in presence of measurement noise," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 7, pp. 922–937, Jul. 2014.

[3] W. Yuan, Y. Liu, H. Wang, and Y. Cao, "A geometric structure-based particle swarm optimization algorithm for multiobjective problems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 9, pp. 2516–2537, Sep. 2017.

[4] S.-Y. Chen and M.-H. Song, "Energy-saving dynamic bias current control of active magnetic bearing positioning system using adaptive differential evolution," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2691304.

[5] S. Zemouri, S. Djahel, and J. Murphy, "An altruistic prediction-based congestion control for strict beaconing requirements in urban VANETs," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2759341.

[6] D. M. K. K. V. Rao and A. Ukil, "Modeling of room temperature dynamics for efficient building energy management," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2758766.

[7] Y. Fu, M. Ding, C. Zhou, and H. Hu, "Route planning for unmanned aerial vehicle (UAV) on the sea using hybrid differential evolution and quantum-behaved particle swarm optimization," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 6, pp. 1451–1465, Nov. 2013.

[8] P. Rakshit *et al.*, "Realization of an adaptive memetic algorithm using differential evolution and q-learning: A case study in multirobot path planning," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 4, pp. 814–831, Jul. 2013.

[9] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren, "Differential evolution algorithm with ensemble of parameters and mutation strategies," *Appl. Soft. Comput.*, vol. 11, no. 2, pp. 1679–1696, Mar. 2011.

[10] J. Ronkkonen, S. Kukkonen, and K. V. Price, "Real-parameter optimization with differential evolution," in *Proc. IEEE Congr. Evol. Comput.*, 2005, pp. 506–513.

[11] J. Brest, S. Greiner, B. Boškovic, M. Mernik, and V. Žumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 646–657, Dec. 2006.

[12] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, "Differential evolution using a neighborhood-based mutation operator," *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 526–553, Jun. 2009.

[13] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.

[14] R. Tanabe and A. Fukunaga, "Success-history based parameter adaptation for differential evolution," in *Proc. IEEE Congr. Evol. Comput.*, 2013, pp. 71–78.

[15] R. A. Sarker, S. M. Elsayed, and T. Ray, "Differential evolution with dynamic parameters selection for optimization problems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 5, pp. 689–707, Oct. 2014.

[16] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, "An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 482–500, Apr. 2012.

[17] Q. Fan and X. Yan, "Self-adaptive differential evolution algorithm with zoning evolution of control parameters and adaptive mutation strategies," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 219–232, Jan. 2016.

[18] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[19] W. Y. Gong, Á. Fialho, Z. H. Cai, and H. Li, "Adaptive strategy selection in differential evolution for numerical optimization: An empirical study," *Inf. Sci.*, vol. 181, no. 24, pp. 5364–5386, Dec. 2011.

[20] Á. Fialho, R. Ros, M. Schoenauer, and M. Sebag, "Comparison-based adaptive strategy selection with bandits in differential evolution," in *Proc. 11th Int. Conf. Parallel Problem Solving Nat. I*, 2010, pp. 194–203.

[21] G. Wu, R. Mallipeddi, P. N. Suganthan, R. Wang, and H. Chen, "Differential evolution with multi-population based ensemble of mutation strategies," *Inf. Sci.*, vol. 329, pp. 329–345, Feb. 2016.

[22] W. Gong, Z. Cai, C. X. Ling, and H. Li, "Enhanced differential evolution with adaptive strategies for numerical optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 2, pp. 397–413, Apr. 2011.

[23] A. Ghosh, S. Das, A. Chowdhury, and R. Giri, "An improved differential evolution algorithm with fitness-based adaptation of the control parameters," *Inf. Sci.*, vol. 181, no. 18, pp. 3749–3765, Sep. 2011.

[24] L. X. Tang, Y. Dong, and J. Y. Liu, "Differential evolution with an individual-dependent mechanism," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 560–574, Aug. 2015.

[25] Y.-Z. Zhou, W.-C. Yi, L. Gao, and X.-Y. Li, "Adaptive differential evolution with sorting crossover rate for continuous optimization problems," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2742–2753, Sep. 2017, doi: 10.1109/TCYB.2017.2676882.

[26] L. Jia, W. Gong, and H. Wu, "An improved self-adaptive control parameter of differential evolution for global optimization," in *Proc. Int. Symp. Intell. Comput. Appl.*, 2009, pp. 215–224.

[27] T. Takahama and S. Sakai, "Efficient constrained optimization by the ε constrained rank-based differential evolution," in *Proc. IEEE Congr. Evol. Comput.*, 2012, pp. 1–8.

[28] Y. Wang, Z. X. Cai, and Q. F. Zhang, "Differential evolution with composite trial vector generation strategies and control parameters," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 55–66, Feb. 2011.

[29] J. J. Liang, B.-Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization," Comput. Intell. Lab., Zhengzhou Univ., Zhengzhou, China, and Nanyang Technol. Univ., Singapore, Rep. 201311, Dec. 2013.

[30] 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.

[31] X. Qiu, K. C. Tan, and J.-X. Xu, "Multiple exponential recombination for differential evolution," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 995–1006, Apr. 2017.

[32] S.-M. Guo, C.-C. Yang, P.-H. Hsu, and J. S.-H. Tsai, "Improving differential evolution with a successful-parent-selecting framework," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 717–730, Oct. 2015.

[33] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, "Enhancing differential evolution utilizing proximity-based mutation operators," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 99–119, Feb. 2011.

[34] W. Y. Gong and Z. H. Cai, "Differential evolution with ranking-based mutation operators," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2066–2081, Dec. 2013.

[35] J.-I. Kushida, A. Hara, and T. Takahama, "Rank-based differential evolution with multiple mutation strategies for large scale global optimization," in *Proc. IEEE Congr. Evol. Comput.*, 2015, pp. 353–360.

[36] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp. 64–79, Feb. 2008.

[37] S.-Y. Park and J.-J. Lee, "Stochastic opposition-based learning using a beta distribution in differential evolution," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2184–2194, Oct. 2016.

[38] H. Wang, S. Rahnamayan, H. Sun, and M. G. H. Omran, "Gaussian bare-bones differential evolution," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 634–647, Apr. 2013.

[39] S. Ghosh, S. Das, S. Roy, S. K. M. Islam, and P. N. Suganthan, "A differential covariance matrix adaptation evolutionary algorithm for real parameter optimization," *Inf. Sci.*, vol. 182, no. 1, pp. 199–219, Jan. 2012.

[40] W. Gong, A. Zhou, and Z. Cai, "A multioperator search strategy based on cheap surrogate models for evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 746–758, Oct. 2015.

[41] S. M. Elsayed, R. A. Sarker, and D. L. Essam, "Differential evolution with multiple strategies for solving CEC2011 real-world numerical optimization problems," in *Proc. IEEE Congr. Evol. Comput.*, 2011, pp. 1041–1048.

[42] G. W. Corder and D. I. Foreman, *Nonparametric Statistics for Non-Statisticians: A Step-by-Step Approach*. Hoboken, NJ, USA: Wiley, 2009.

[43] R. Tanabe and A. S. Fukunaga, "Improving the search performance of shade using linear population size reduction," in *Proc. IEEE Congr. Evol. Comput.*, 2014, pp. 1658–1665.

[44] S. M. Elsayed, R. A. Sarker, D. L. Essam, and N. M. Hamza, "Testing united multi-operator evolutionary algorithms on the CEC2014 real-parameter numerical optimization," in *Proc. IEEE Congr. Evol. Comput.*, 2014, pp. 1650–1657.

[45] I. Erlich, J. L. Rueda, S. Wildenhues, and F. Shewarega, "Evaluating the mean-variance mapping optimization on the IEEE-CEC 2014 test suite," in *Proc. IEEE Congr. Evol. Comput.*, 2014, pp. 1625–1632.

[46] N. H. Awad, M. Z. Ali, P. N. Suganthan, and R. G. Reynolds, "An ensemble sinusoidal parameter adaptation incorporated with L-SHADE for solving CEC2014 benchmark problems," in *Proc. IEEE Congr. Evol. Comput.*, 2016, pp. 2958–2965.

[47] R. Storn and K. Price, "Minimizing the real functions of the ICEC'96 contest by differential evolution," in *Proc. IEEE Int. Conf. Evol. Comput. (ICEC)*, 1996, pp. 842–844.

[48] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Heidelberg, Germany: Springer, 2006.

[49] K. V. Price, "An introduction to differential evolution," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. Maidenhead, U.K.: McGraw-Hill, 1998, pp. 79–108.

[50] E. Mezura-Montes, J. Velazquez-Reyes, and C. A. C. Coello, "Modified differential evolution for constrained optimization," in *Proc. IEEE Congr. Evol. Comput.*, Vancouver, BC, Canada, Jul. 2006, pp. 25–32.

Xiao-Fang Liu (S'14) received the B.S. degree in computer science from Sun Yat-sen University, Guangzhou, China, in 2015, where she is currently pursuing the Ph.D. degree in computer science.

She is currently a Research Fellow with the Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, South China University of Technology, Guangzhou. Her current research interests include artificial intelligence, evolutionary computation, swarm intelligence, and their applications in design and optimization, such as cloud computing resources scheduling.

Zhi-Hui Zhan (S'09–M'13) received the bachelor's and Ph.D. degrees in computer science from the Department of Computer Science, Sun Yat-sen University, Guangzhou, China, in 2007 and 2013, respectively.

He is currently the Changjiang Scholar Young Professor and the Pearl River Scholar Young Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou. His current research interests include evolutionary computation algorithms, swarm intelligence algorithms and their applications in real-world problems and in environments of cloud computing and big data.

Dr. Zhan was a recipient of the China Computer Federation Outstanding Dissertation Award and the IEEE CIS Outstanding Dissertation Award for his doctoral dissertation, the Natural Science Foundation for Distinguished Young Scientists of Guangdong Province, China, in 2014, the Pearl River New Star in Science and Technology Awrad in 2015, the Youth Talent in Science and Technology Innovation of Guangdong Province in 2016, and the Wu Wen Jun Artificial Intelligence Excellent Youth Award from the Chinese Association for Artificial Intelligence in 2017. He is listed as one of the Most Cited Chinese Researchers in Computer Science.

Ying Lin (M'12) received the Ph.D. degree in computer applied technology from Sun Yat-sen University, Guangzhou, China, in 2012.

She is currently an Assistant Professor with the Department of Psychology, Sun Yat-sen University. Her current research interests include computational intelligence and its applications in network analysis and cognitive diagnosis.

Wei-Neng Chen (S'07–M'12) received the bachelor's and Ph.D. degrees in computer science from Sun Yat-sen University, Guangzhou, China, in 2006 and 2012, respectively.

He is currently a Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou. He has published 50 papers in international journals and conferences. His current research interests include swarm intelligence algorithms and their applications on cloud computing, operations research, and software engineering.

Dr. Chen was a recipient of the IEEE Computational Intelligence Society Outstanding Dissertation Award in 2016 for his doctoral thesis, the National Science Fund for Excellent Young Scholar in 2016, and the Natural Science Fund for Distinguished Young Scientists of Guangdong Province, China, in 2015, the "Guangdong Special Support Program" for Outstanding Young Scientists in 2015, and the Pearl River New Star in Science and Technology in 2014.

Yue-Jiao Gong (M'15) received the B.S. and Ph.D. degrees in computer science from Sun Yat-sen University, Guangzhou, China, in 2010 and 2014, respectively.

From 2015 to 2016, she was a Post-Doctoral Research Fellow with the Department of Computer and Information Science, University of Macau, Macau, China. She is currently an Associate Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Her current research interests include evolutionary computation and machine learning methods, as well as their applications to intelligent transportation and smart city. She has published over 50 papers in the above areas.

Dr. Gong currently serves as a Reviewer for the IEEE TRANSACTION ON EVOLUTIONARY COMPUTATION, IEEE TRANSACTION ON NEURAL NETWORK AND LEARNING SYSTEMS, and IEEE TRANSACTION ON IMAGE PROCESSING.

Tian-Long Gu received the M.Eng. degree in computer science from Xidian University, Xi'an, China, in 1987, and the Ph.D. degree in computer science from Zhejiang University, Hangzhou, China, in 1996.

From 1998 to 2002, he was a Research Fellow with the School of Electrical and Computer Engineering, Curtin University of Technology, Perth, WA, Australia, and a Post-Doctoral Fellow with the School of Engineering, Murdoch University, Perth, WA, Australia. He is currently a Professor with the School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin, China. His current research interests include formal methods, data and knowledge engineering, software engineering, and information security protocol.

Hua-Qiang Yuan received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 1996.

He is currently a Professor with the School of Computer Science and Network Security, Dongguan University of Technology, Dongguan, China. His current research interests include computational intelligence and cyberspace security.

Jun Zhang (M'02–SM'08–F'17) received the Ph.D. degree in electrical engineering from the City University of Hong Kong, Hong Kong, in 2002.

He is currently a Changjiang Chair Professor with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His current research interests include computational intelligence, cloud computing, high performance computing, data mining, wireless sensor networks, operations research, and power electronic circuits. He has published over 100 technical papers in the above areas.

Dr. Zhang was a recipient of the China National Funds for Distinguished Young Scientists from the National Natural Science Foundation of China in 2011 and the First-Grade Award in Natural Science Research from the Ministry of Education, China, in 2009. He is currently an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, IEEE TRANSACTIONS ON CYBERNETICS, and IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.