

## An enhanced Kalman filtering and historical learning mechanism driven estimation of distribution algorithm

Ningning Zhu<sup>a</sup>, Fuqing Zhao<sup>a,\*</sup>, Ling Wang<sup>b</sup>, Chenxin Dong<sup>c</sup>

<sup>a</sup> School of Computer and Communication Technology, Lanzhou University of Technology, Lanzhou 730050, China

<sup>b</sup> Department of Automation, Tsinghua University, Beijing, 10084, China

<sup>c</sup> School of Mechanical and Automotive Engineering, Qingdao Hengxing University of Science and Technology, Qingdao 266100, China



### ARTICLE INFO

#### Keywords:

Estimation of distribution algorithm  
Kalman filtering  
Historical learning mechanism  
Adaptive adjustment strategy  
Elite strategy

### ABSTRACT

As a representative evolutionary algorithm based on probabilistic models, the estimation of distribution algorithm (EDA) is widely applied in complex continuous optimization problems based on remarkable characteristics of modeling with macro-dominant information. However, the success of EDA depends on the quality of dominant solutions, modeling, sampling methods, and the efficiency of searching. An enhanced Kalman filtering and historical learning mechanism-driven EDA (KFHLEDA) is proposed to adjust the search direction and enlarge the search range of classical EDA in this paper. The enhanced Kalman filtering is designed in allusion to specific problems during the search through the prediction, observation, and the first and second revision stages. A historical archive is integrated into KFHLEDA to store the elite individuals with specific knowledge and diverse solutions from Kalman filtering. The elite strategy is embedded in the revision improvement matrix to revise modeling data, which is fed back to the probabilistic model through the historical learning mechanism with previous promising solutions to estimate the covariance matrix. The population adaptive adjustment strategy is introduced to reduce the number of invalid iterations. The effectiveness of the proposed KFHLEDA is proved through theoretical analysis. The evaluation results on benchmark functions of the CEC 2017 test suit validate that the KFHLEDA is efficient and competitive compared with fifteen classical metaheuristic algorithms and state-of-the-art EDA variants.

### 1. Introduction

Complex continuous optimization problems exist in practical industrial production and manufacturing systems with various optimal configurations aiming at minimizing objective functions [1]. However, it is extremely difficult to accurately establish mathematical models in several subsistent optimization scenarios, and the computational complexity increases exponentially with the dimension of the problem scale [2]. Moreover, branch and bound [3], linear programming [4], and traditional mathematical methods are unable to solve such problems efficiently due to non-separability, rotation invariance, non-linearity, and other characteristics of the variables, especially for the black-box optimization without analyzable objectives and gradient information [5,6].

Meta-heuristic [7], a typical population-based intelligent optimization algorithm [8,9], overcomes the defects of traditional mathematical methods and achieves remarkable success due to its strong versatility.

The optimization solutions for the optimization objectives are robust [10]. In 1996, H.Mühlenbein and G.Paaß first proposed EDA as a typical meta-heuristic method based on estimation and sampling of the statistical probabilistic model [11]. In contrast to the other meta-heuristic methods, such as differential evolution (DE) [12,13] and genetic algorithm (GA) [14,15], that generate new solutions through crossover and mutation, the EDA combines GA and statistical learning approaches to select the high-quality individuals for modeling and estimating the distribution at the next generation to guide the population close to the potential region [16].

EDA establishes the mathematical model at the macro level of evolution, which provides a meaningful tool for complex optimization problems. Effective information is extracted and utilized in the evolutionary process based on specific structural characteristics. The relationship between variables is described through a probability model, which is valid for settling the high-dimensional, non-linear, and variable coupling optimization problems [17,18]. Diverse solution-generation

\* Corresponding author.

E-mail addresses: [307516638@qq.com](mailto:307516638@qq.com) (N. Zhu), [zhaofq@lut.edu.cn](mailto:zhaofq@lut.edu.cn) (F. Zhao), [wangling@tsinghua.edu.cn](mailto:wangling@tsinghua.edu.cn) (L. Wang), [deyangxuanyi@hotmail.com](mailto:deyangxuanyi@hotmail.com) (C. Dong).

approaches arise from the utilization of various probabilistic models. Common models include histogram models [19,20], Cauchy models [21], and Gaussian models [22]. Gaussian EDA (GEDA) is commonly applied in continuous optimization problems, and classified into three categories according to the complexity and variable dependency. The first category is the variable independent EDA [23], which includes the univariate marginal distribution algorithm (UMDAc) [24], and population-based incremental learning (PBIL) [25]. The second category is the bivariate correlated EDA represented by the bivariate marginal distribution algorithm (BMDA) [26,27]. The third category is the multivariate correlated EDA, which includes the estimation of multivariate normal algorithm (EMNA) [28] and the estimation of Gaussian networks algorithm (EGNA) [29].

Multivariate correlated EDA shows outstanding performance on most problems. Both EMNA and EGNA are based on single peaks [30]. The EMNA adopts maximum likelihood to estimate the mean and covariance matrix of multivariate Gaussian distribution at each generation in the process of evolution [31]. The EMNA instantly describes the distribution of high-quality solutions through the feedback of valuable information between variables and the utilization of problem characteristics. Therefore, the EMNA with immense potential for further improvement is employed in this paper. The EGNA is based on a Gaussian graph model where directed edges represent the relationship between variables. The EGNA reconstructs the Gaussian graph network according to the current population to represent the solution space and requires a long time to learn from the complex multi-peak and non-linear optimization problems [29]. The iterated density estimation algorithm (IDEA), proposed by Bosman [32], was employed to optimize problems with complex shapes. However, the relationship between variables is considered inadequate.

Despite exhibiting promising performance on a wide range of problems by capturing the characteristics of the optimization process, the EMNA still suffers from some limitations. First, the EMNA constantly estimates the distribution of the solutions for the next generation based on the high-quality solutions of the current generation. This approach results in selected solutions being located near the current mean, and population diversity is lost [33]. Therefore, designing an appropriate mechanism to learn from both historical and current information is indispensable to estimating the variance trend of new promising solutions. Furthermore, the main search direction of EMNA is perpendicular to the direction of fitness improvement, which affects its performance [34]. Several studies reveal that the search scope and direction affect the efficiency and accuracy of EMNA [35,36]. A large search scope contributes to producing plentiful promising solutions with high accuracy. The parallel direction of search and fitness improvement is conducive to search efficiency.

This paper proposes a novel approach to solve the above-mentioned problems simultaneously. Moreover, the study theoretically analyzes the improvement with the search scope and direction of EMNA by the proposed method. Kalman filtering, a digital processing technology in a communication system, is introduced. Based on the specific problem characteristics, the corresponding learning mechanisms and adjustment strategies combined with Kalman filtering are embedded in the framework of EMNA. An enhanced Kalman filtering and historical learning mechanism driven EDA (KFHLEDA) is proposed to improve search efficiency and solution accuracy while addressing the limitations of diversity loss, small search scope, and poor search direction. The effectiveness of the proposed KFHLEDA is verified through both theoretical analysis and experimental results. The main contributions are summarized as follows.

- A novel EMNA model is introduced based on an enhanced Kalman filtering mechanism designed for specific problem characteristics. The filtering process contains prediction, observation, and revision. The proposed KFHLEDA modifies the search center and improves the

search scope and direction. The revision stage is further divided into the first and second revisions to prevent population stagnation.

- The elite strategy is integrated into the revision improvement matrix to accelerate the process of evolution. The population adaptive adjustment strategy is employed to reduce the invalid search and achieve a good balance between exploration and exploitation.
- The KFHLEDA utilizes Kalman filtering to acquire enhanced information, which is fed back to the probabilistic model through the historical learning mechanism with previous promising solutions to estimate the covariance matrix, resulting in increased population diversity and further improvements in the quality of solutions.

The remainder of this paper is organized as follows. A literature review is presented in Section 2. The proposed KFHLEDA is described in Section 3. The experimental results of KFHLEDA and the comparison algorithms are presented in Section 4. The conclusions are drawn in Section 5.

## 2. Literature review

As a classical probabilistic model for GEDA, the multivariate model considers the dependency of multiple variables [37]. Representative algorithms, such as EMNA [31], EGNA [29], IDEA [38], and Bayesian optimization algorithm (BOA) [39], have demonstrated superior optimal effects on most problems. However, there are some limitations, such as fast diversity loss, poor evolution direction, and small search scope, so the search efficiency and solution accuracy are affected [34, 35]. Several studies propose various variants to overcome the limitations and exploit potential multivariate models.

Population diversity has a significant impact on the performance of EDA. Various studies demonstrate the effectiveness of variance adaptation and scaling methods to address diversity loss. Grahl and Bosman et al. presented a correlation-triggered adaptive variance scaling mechanism and incorporated it into IDEA to enlarge or shrink the variances based on the generated solutions [40]. Later, they proposed another approach named standard deviation ratio to deal with this structure [41]. Wagner et al. designed a novel EDA through the eigenspace and reconstructed the covariance matrix for the eigenvector with the smallest eigenvalue in the eigenspace [42]. In [43], an adaptive covariance scaling estimation of distribution algorithm (ACSEDA) was developed to compute the covariance based on various high-quality solutions, and an adaptive selection strategy and a covariance scaling method were established to estimate the mean and covariance, respectively. In [44], a random walk mutation was introduced into the collaborative frame of DE and EDA to settle the non-linear equations systems to retain the population diversity.

The archive technology is introduced to store a variety of promising information in optimization algorithms to maintain population diversity. The high-quality solutions in the archive have a high probability of being close to the global optimum. The historical learning mechanism utilizes effective historical and current information to influence the subsequent evolutionary process. The historical archive was employed by Zhang et al. [45], and a new differential evolution algorithm, called JADE, was proposed to implement a new mutation strategy with an optional external archive utilizing historical data to provide information for evolution direction. In [46], a multivariate  $t$  distribution, archive, and mutation-based EDA was introduced to store the promising solutions for modeling. Liang et al. [36] incorporated a historical archive into EDA to guide the population to move toward promising regions. The number of individuals was reduced while the population diversity was maintained. In [47], a neighborhood-based EDA was proposed to encourage population diversity and search efficiency by using a neighborhood sampling strategy and diverse coverage schemes. Inspired by this, the historical learning mechanism is integrated into the proposed KFHLEDA.

Various studies reveal that the search direction and scope confine the

search efficiency and solution accuracy of GEDA. The main search direction becomes orthogonal to that of the fitness improvement. The rapid decline of variance generates a sharp decrease in search scope, and premature convergence occurs in the evolutionary process [34]. In [48], a novel approach was proposed to estimate probability density by adjusting the search behavior and implementing a reflecting sampling mechanism to further enhance the search efficiency. Hansen et al. [49] presented a rank- $\mu$ -update operator for the covariance with the weighted dominant solutions to increase the variance along the gradient direction and explored a surrogate fitness model in the continuous optimization problems. Bosman et al. [50] proposed an anticipated mean shift strategy to shift several solutions along a certain direction, and the main search direction was altered. A novel algorithm, named AMaLGaM-IDEA, was developed by incorporating anticipated mean shift, adaptive variance scaling, and standard deviation ratio.

Various progressive mechanisms have been introduced to EDA models to improve solution accuracy. Doerr et al. [51] introduced a long history of samples into the EDA framework and updated the probabilistic model according to the information that it classified. The time cost was reduced and the accuracy was improved. In [52], an estimation of distribution algorithm based on an approximation of the Boltzmann distribution was proposed to balance the time cost and solution accuracy. Lin et al. [53] presented a new EDA that utilized the normalized mutual information in conjunction with an updated sampling mechanism, and opposition-based learning scheme to generate high-quality solutions. In [54], a co-evolutionary algorithm based on EDA and DE was incorporated with the random forest and fitness landscape to guide the search behavior and visualize the exact distribution of solutions in complex optimization problems.

The selection of population size is an important metric for EDA and other meta-heuristic algorithms [55–57]. Appropriate population size affects the exploration and exploitation of the algorithms and conduces to saving computing sources. A large population size is beneficial to exploration, but the convergence speed is slow, and considerable computing resources are occupied. A small population size accelerates the convergence speed, but it is easy to fall into the local optimum. Various population adjustment strategies have been applied in EDA, and good results have been obtained. Auger et al. [56] introduced a covariance matrix adaptive evolution strategy (CMA-ES) by increasing the population size to restart the population and improve the search characteristic. Tanabe et al. [57] addressed an adaptive DE utilizing the success-history-based parameter adaptation with linear population size reduction which makes the population size decline linearly over time. Bujok et al. [58] presented eigen crossover based on a cooperative algorithm with CMA-ES and three DE variants, and a linear population-size reduction mechanism was employed by the cooperative model to balance exploration and exploitation.

Diverse GEDA variants have achieved promising effects in multi-

#### Algorithm 1

##### Kalman filtering

---

```

1      Set the initial state variable and revision improvement matrix.
2      nifes = 1.
3      While nifes ≤ max_nifes do
4          Perform prediction:
5              Obtain the predicted status  $x^{pre}$  and revision improvement matrix  $v$ .
6          Perform observation:
7              Obtain the observed status  $x^{obs}$ .
8          Perform revision:
9              Calculate the gain vector  $g$ .
10             Update the revised status  $x^{rev}$ .
11             Update revision improvement matrix  $v$ .
12             nifes = nifes + 1.
13     End while

```

---

where  $nifes$  and  $max\_nifes$  represent the evaluated number and the maximum evaluated number of the fitness values.

modal optimization problems. Wang et al. [59] proposed an adaptive estimation distribution distributed differential evolution (AEDDDE) algorithm, where every individual formed its niche to reach the global optimum, and multiple parameter-free niches coevolved with master-slave mechanisms to solve the distributed multi-modal problems. Yang et al. [60] designed a maintaining and processing sub-model method integrated with GEDA to explore the potential regions and accelerate the evolution process for multimodal problems. In [61], a new EDA that incorporated a dynamic cluster sizing strategy and utilized both Gaussian and Cauchy distributions was presented to balance exploration and exploitation. Moreover, an adaptive local search was adopted to improve the solution accuracy.

The EDA demonstrates splendid performance in solving continuous optimization problems due to its exceptional self-learning ability [62]. The EDA has also been applied to a variety of practical problems. Shao et al. [63] presented a Pareto-based EDA to solve the distributed no-wait flow shop scheduling problem with sequence-dependent setup time. In [64], a knowledge-based reinforcement learning and EDA for a flexible job shop scheduling problem that involved time-of-use electricity price constraints was presented. Ceberio et al. [65] proposed a distance-based ranking model EDA with the generalized Mallows model to explore the promising regions for the flow-shop scheduling problem. Zhang et al. [66] presented a three-dimensional matrix-cube-based EDA for the distributed assembly permutation flow-shop scheduling problem to minimize the maximum completion time.

To sum up, various studies have addressed the limitations of EDA by focusing on specific aspects such as learning mechanisms, adjustment strategies, or other meta-heuristics. This paper proposes a novel EDA that utilizes Kalman filtering and historical learning mechanisms to simultaneously improve the population diversity, enlarge the search range, and adjust the search direction of traditional EMNA in continuous optimization problems.

### 3. KFHEDA

#### 3.1. Kalman filtering

Kalman filtering [67] is an algorithm that utilizes the state equation to optimally estimate the state of a system through input and output observations. All past inputs and disturbances are represented by a set of minimal parameters. The state determines the entire behavior of the system along with future inputs and disturbances. Assuming proper statistical properties of the disturbance and observation error, the estimated value of the real state is acquired by processing the observed variables.

Gaussian distribution is adopted in this paper. When both the observation data and the state obey Gaussian distribution, the mean and covariance are calculated by Kalman recursive equations according to the properties of Kalman filtering. The updating process of conditional probability density is conducted by the minimum variance estimation. State estimation is a crucial part of Kalman filtering. Generally, quantitative inference based on the observed variables is regarded as an estimation problem. Especially, the estimation of dynamic behavior contributes to predicting the current state.

Kalman filtering is particularly well-suited for continuously changing dynamic systems [68]. It is due to its ability to track the real state to provide a reasonable prediction of the next direction for the system. The predicted and observed values are calculated according to Eqs. (1) and (2), respectively.

$$x(k) = Ax(k-1) + B\theta(k) + w(k) \quad (1)$$

$$y(k) = Hx(k-1) + v(k) \quad (2)$$

where  $x(k-1)$  and  $x(k)$  represent the predicted values at the previous and current time, respectively,  $\theta(k)$  is a real number as the current input,

**Algorithm 2**

EMNA with the historical learning mechanism

---

<b>Input:</b>	$\tau, NP$
<b>Output:</b>	$NEW\_POP$
1	Initialize all parameters.
2	Generate the initial population.
3	<b>While</b> $n_{fes} \leq max\_n_{fes}$ <b>do</b>
4	Evaluate the fitness function.
5	Select the dominant individuals by truncation selection with $\tau$ and place them into the archive.
6	Estimate $\mu$ and $C$ .
7	Sample the probabilistic model to generate the new dominant individuals placed into the archive.
8	Merge dominant individuals in the archive to form $NEW\_POP$ .
9	$n_{fes} = n_{fes} + 1$ .
10	<b>End while</b>

---

where  $NEW\_POP$  represents the generated new population,  $\tau$  is the selection rate, and  $NP$  is the population size.

$w(k)$  represents the prediction error matrix,  $y(k)$  represents the observed value at the current time,  $v(k)$  represents the noise component during observation,  $A$  is the state transition matrix from the previous to the current state,  $B$  is the observation coefficient matrix, and  $H$  is the coefficient matrix of input variables.

Kalman filtering is well-suited for linear Gaussian systems and is applied to certain dynamic systems with uncertain information, including continuous systems. It can also be employed in complex non-linear systems in virtue of non-linear mapping techniques [69]. In this paper, the difference in the function fitness is introduced in the gain coefficient and the non-linear relationship is simulated.

Kalman filtering with the operation of prediction, observation, and revision is introduced into the EMNA. The process of Kalman filtering is shown in [Algorithm 1](#).

### 3.2. EMNA with historical learning mechanism

The EMNA utilizes the dominant information in the solutions generated at the current generation to estimate the probabilistic models, which is crucial to the effectiveness of the algorithm. The introduction of a historical learning mechanism enables the Gaussian model to predict new solutions by utilizing historical information rather than by relying on the limited current information. The Gaussian probability density function is expressed as [Eq. \(3\)](#).

$$G_{(\mu, C)}(x) = \frac{(2\pi)^{-n/2}}{(\det C)^{1/2}} \exp(-\frac{1}{2}(x - \mu)^T C^{-1} (x - \mu)) \quad (3)$$

where  $x$  is an  $n$ -dimensional random vector,  $\mu$  represents the mean, and  $C$  is the covariance matrix.

The  $\mu$  and  $C$  contain  $n$  and  $0.5 * (n^2 + n)$  estimated parameters, respectively, which are estimated according to the maximum likelihood estimation based on the selected solutions at the current generation, as shown in [Eqs. \(4\)-\(5\)](#).

The historical learning mechanism is further emphasized by introducing the archive technique to update the mean and covariance matrix. The information from the historical learning mechanism is employed to estimate the mean and covariance matrix to revise the modeling data, which further improves the solution accuracy.

$$\mu(k) = \frac{1}{s} \sum_{i=1}^s x_i(k) \quad (4)$$

$$C(k) = \frac{1}{s} \sum_{i=1}^s (x_i(k) - \mu(k))(x_i(k) - \mu(k))^T \quad (5)$$

where  $x_i(k)$  represents the  $i$ th solution in the history archive,  $POP = \{x_1(k), x_2(k), \dots, x_s(k)\}$ ,  $POP$  represents the population, and  $s$  represents

**Algorithm 3**

KFHLEDA

---

<b>Input:</b>	$\tau, NP_{max}, max\_n_{fes}, h, D, NP_{min}$
<b>Output:</b>	The best individual
1	Initialize the population.
2	$n_{fes} = 1$ ;
3	Select the $\tau NP$ individuals with small fitness values as the modeling population $POP^{mod}$ .
4	Find the best individual.
5	Establish the model according to <a href="#">Eqs. (4)</a> and <a href="#">(5)</a> .
6	Sample and obtain $(1 - \tau) NP$ observed individuals $POP^{obs}$ .
7	Merge $POP^{mod}$ and $POP^{obs}$ to form $NEW\_POP$ .
8	$POP^{obs}$ is regarded as $POP^{rev}$ at the first generation.
9	Calculate $\pi$ according to <a href="#">Eq. (7)</a> .
10	$n_{fes} = 2$ ;
11	<b>While</b> $n_{fes} \leq max\_n_{fes}$ <b>do</b>
12	Perform Kalman filtering to obtain $POP^{pre}$ .
13	Find $x_{pbest}$ in the archive.
14	If the conditions for the first revision are satisfied <b>do</b>
15	Perform the first revision.
16	<b>End</b>
17	Perform the second revision.
18	Select $\tau * NP$ individuals of $POP^{pre}$ , and $POP^{rev}(n_{fes} - 1)$ in the archive $h$ as $POP^{mod}$ .
19	Establish the model according to <a href="#">Eqs. (12)-(13)</a> .
20	Calculate the population size of every generation according to <a href="#">Eq. (6)</a> .
21	Sample and obtain $(1 - \tau) NP$ $POP^{obs}$ .
22	Merge $POP^{mod}$ and $POP^{obs}$ .
23	Find the best individual.
24	Calculate $g$ and $\pi$ .
25	Obtain $POP^{rev}$ .
26	$n_{fes} = n_{fes} + 1$ .
27	<b>End while</b>

---

where  $\tau$  represents the truncated selection rate of selected dominant individuals,  $D$  is the dimension,  $h$  represents the size of the historical archive,  $POP^{pre}$ ,  $POP^{mod}$ ,  $POP^{obs}$ , and  $POP^{rev}$  represent the predicted, modeled, observed, and revised populations,  $\pi$  represents the revision improvement matrix and  $g$  is the gain coefficient of revision improvement.

the number of the selected solutions in the archive.

The historical archive is used to store the promising solutions from Kalman filtering, so the abundant hidden information is learned by the proposed algorithm. [Algorithm 2](#) illustrates the operation of EMNA with the historical learning mechanism.

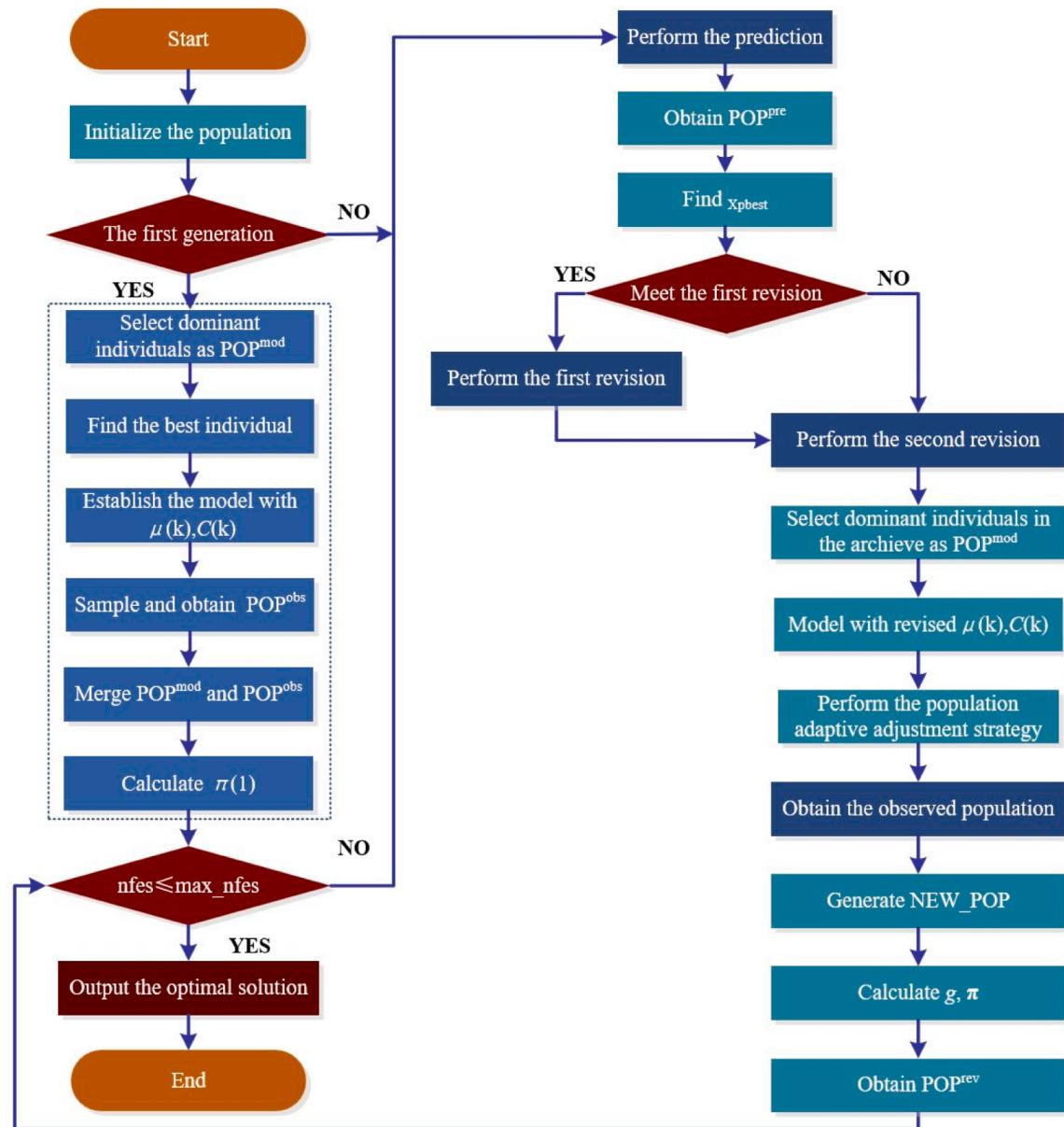
### 3.3. Population adaptive adjustment strategy

The EDA is a population-based algorithm, and the size of the population has a vital impact on its performance. A large search space avoids getting stuck in a local optimum, hence, exploration is crucial in the early evolution. The exploitation is emphasized at the later stage. A large population size is beneficial to population diversity and exploration, while a small one contributes to exploitation. The decrease in the population size accelerates the convergence speed and achieves the optimal solution appropriately. Compared to the conventional approach with a fixed population size, a population adaptive adjustment strategy makes a better trade-off between exploration and exploitation. The strategy utilizes the specific characteristics of the population at different stages to adaptively adjust the population size with the iteration process, which avoids the waste of computing resources. The adaptive adjustment strategy is shown in [Eq. \(6\)](#).

$$NP(k+1) = round\left(NP_{max} - \frac{n_{fes}}{max\_n_{fes}} (NP_{max} - NP_{min})\right) \quad (6)$$

where  $NP_{max}$  and  $NP_{min}$  represent the maximum and the minimum numbers of the population,  $NP(k+1)$  is the population size at the  $(k+1)$  generation and  $round$  represents a function.

The population size is gradually decreased from the initial maximum



**Fig. 1.** The flow chart of KFHLEDA.

value to a minimum value by the linear reduction during the iteration process. The maximum and minimum of the population size are critical. An excessive value leads to slow convergence, which wastes computing resources, while a small one weakens exploration. With a view to the characteristics of EMNA, the covariance matrix contains  $0.5 * (n^2 + n)$  estimated parameters.  $NP_{min}$  is set as  $0.5 * (n^2 + n)$ , and  $NP_{max}$  is obtained by parameter analysis in Section 4.2.

### 3.4. The proposed KFHLEDA

Kalman filtering has been employed in various communication systems. The optimization problem is abstracted as a system, and Kalman filtering is typically employed to solve the ideal state of the system. In this paper, the idea of the system is integrated with EMNA and specific problem characteristics. The Kalman filtering aims to predict the probabilistic model by promising solutions with diverse characteristics. The sampled solutions are applied to observation. Inspired by the filtering technique, the individuals are revised by prediction and observation, which in turn affects the modeling for the next generation. Transferring

it to the optimization problems is a new attempt.

The pseudocode of the proposed KFHLEDA is shown in Algorithm 3.

The probabilistic model of traditional EMNA does not utilize the previous information. The dominant solutions from the prediction, observation, and revision stages of Kalman filtering are placed into the historical archive to discover the potential regions in the search space. The enhanced information in the archive is fed back to the probabilistic model through the historical learning mechanism. Due to the positive impact of the elite strategy on solving various optimization problems [70,71], the elite strategy is integrated into the revision improvement matrix to accelerate the search speed during the whole process of Kalman filtering. The flow chart is summarized in Fig. 1, and the detailed steps are described in the following sections.

Kalman filtering and archive technology utilizes the state information of the previous generation to predict the current state. The revised solution is generated by the predicted and the observed values in the archive, which affects the solutions at the next generation by turns. Appropriate revisions lead the solutions to move to the search center with a high probability. The filtering aims to improve search efficiency

and solution accuracy by adjusting the search direction and scope.

### 3.4.1. Initialization

The concept of Kalman filtering requires the availability of prior information. In the proposed KFHLEDA, the solutions of the first generation are separately updated with the prior information.

The dominant individuals selected by the truncation selection are taken as  $POP^{mod}$ , and the best individual  $x_{pbest}$  is found. The model is established as Eqs. (3)-(4). After modeling and sampling,  $POP^{obs}$  is obtained.  $NEW\_POP$  at the first generation is constructed with  $NEW\_POP_{(1)} = (POP^{mod}, POP^{obs})$ . The revision improvement matrix at the first generation  $\pi(1)$  is calculated according to Eq. (7) and added to the revision stage as an index of prediction accuracy.

$$\pi(1) = \begin{bmatrix} \pi_1(1) \\ \vdots \\ \pi_m(1) \end{bmatrix} = \begin{bmatrix} x_{pbest}(1) - x_1^{obs}(1) \\ \vdots \\ x_{pbest}(1) - x_m^{obs}(1) \end{bmatrix} \quad (7)$$

where  $x_i^{obs}(1) = (x_1^{obs}(1), \dots, x_m^{obs}(1))$ ,  $i = 1, \dots, m$  represents the selected individuals from the observed population at the first generation,  $m = \tau NP(k)$ ,  $\tau$  is obtained based on the parameter analysis in Section 4.2, and  $NP(k)$  is the population size of the  $k$ th generation, which is calculated by the population adaptive adjustment strategy discussed in Section 3.3.

### 3.4.2. The prediction and observation stages

The predicted population, denoted as  $POP^{pre}(k)$ , is generated by predicting the relevant information based on the previous population. The  $x_{pbest}$  is the best individual from the archive.

The prediction process is shown in Eq. (8).

$$x_i^{pre}(k) = Mx_{pbest}(k-1) + N\pi_i(k-1) \quad (8)$$

where  $x_i^{pre}(k)$  is the  $i$ th predicted individual of the  $k$ th generation,  $x_{pbest}(k-1)$  is the best individual of the  $(k-1)$ th generation,  $N$  is a diagonal matrix with the diagonal elements randomly from the uniform distribution  $U[0, 1]$ , and  $\pi_i(k-1)$  is the revision improvement of the  $i$ th individual at the  $(k-1)$ th generation. The revision improvement matrix  $\pi(k)$  is calculated using Eq. (9).

$$\pi(k) = \begin{bmatrix} \pi_1(k) \\ \vdots \\ \pi_m(k) \end{bmatrix} = \begin{bmatrix} x_{pbest}(k) - x_1^{obs}(k) \\ \vdots \\ x_{pbest}(k) - x_m^{obs}(k) \end{bmatrix} \quad (9)$$

In the proposed KFHLEDA, the dominant individuals from  $POP^{pre}(k)$  and  $POP^{pre}(k-1)$  are used for modeling. The model is established according to Eqs. (12)-(13) and the observed solutions for the observation phase are generated after sampling.

### 3.4.3. The revision procedure

In contrast to the conventional Kalman filtering, the proposed KFHLEDA divides the revision stage into two procedures, namely, the first revision and the second revision in allusion to the specific optimization problems.

#### (1) The first revision

The first revision is performed when at least one of the following two conditions is satisfied during the evolution process.

**Condition 1.** The covariance matrix is not positive definite.

**Condition 2.** The difference between the maximum and minimum fitness values is less than  $10^{-4}$ .

The specific revision method is described as follows. The dominant

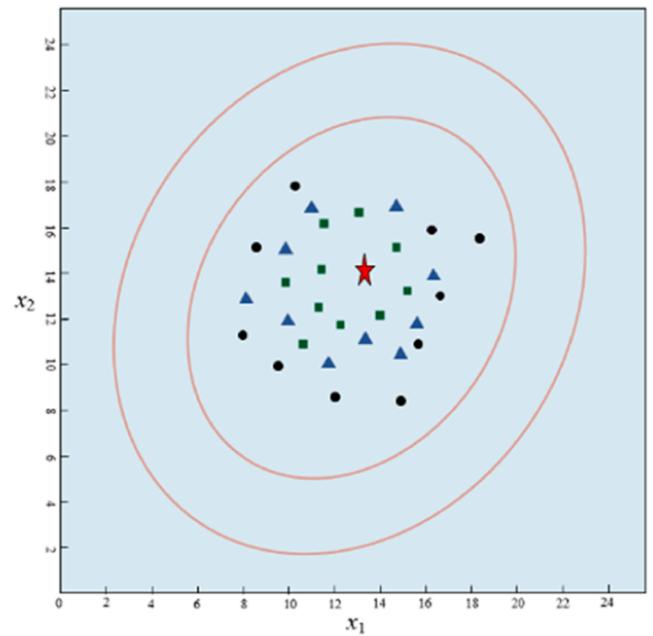


Fig. 2. The composition of the solutions after Kalman filtering.

individuals in the archive are selected for modeling with the selection rate  $\tau$ , which affects the evolution of the next generation. When any condition occurs,  $\tau$  changes with a fixed value, which is set to 10% according to [71]. The  $\tau$  for the predicted population increases by 10% when  $\tau$  for other population decreases by 10%. Similarly, if no better solutions are generated for two consecutive generations,  $\tau$  is adjusted in the same manner. If the new percentage exceeds 100%, the algorithm subtracts 100% and retains the remaining percentage. When the percentage is less than 0, 100% is added. For example, the proportion of the predicted population is 95%, and the other is 5%. When any condition of the first revision is satisfied,  $\tau$  for the predicted population is calculated as  $(95\%+10\%)-100\% = 5\%$ . For the other population,  $\tau$  is  $(5\% -10\%) + 100\% = 95\%$ .

#### (2) The second revision

Dominant individuals of  $POP^{pre}(k)$  and  $POP^{pre}(k-1)$  are selected from the historical archive. After the modeling and sampling are implemented,  $POP^{obs}(k)$  is acquired.  $NEW\_POP$  is generated by combining  $POP^{mod}(k)$  and  $POP^{obs}(k)$ . The best individual is found.

The search process of EMNA is typically non-linear. The gain coefficient and the revision improvement matrix are employed to perform the necessary non-linear transformation for Kalman filtering. The gain coefficient  $g_i(k)$  is shown in Eq. (10).

$$g_i(k) = \frac{\sum_{i=1}^m \pi_i(k-1)}{\sum_{i=1}^m \pi_i(k-1) \sum_{i=1}^m (f(x_{pbest}(k-1)) - f(x_i^{obs}(k-1))) / m} \quad (10)$$

where  $f(x_{pbest}(k-1))$ , and  $f(x_i^{obs}(k-1))$  are the fitness values of the best individual and the  $i$ th observed individual at the  $(k-1)$ th generation, respectively.

The  $\pi_i(k-1)$  is introduced into  $g_i(k)$ . The fitness difference between the best and the observed individuals at the previous generation is taken as the weighted coefficient in the revised population, which is generated according to Eq. (11).

$$x_i^{res}(k) = x_i^{pre}(k) + g_i(k) \left( f(x_{pbest}^{pre}(k)) - f(x_i^{pre}(k)) \right) \pi_i(k), \quad i = 1, \dots, m \quad (11)$$

where  $f(x_{pbest}^{pre}(k))$  and  $f(x_i^{pre}(k))$  represent the best and the  $i$ th individuals from the prediction population at the  $k$ th generation, respectively.

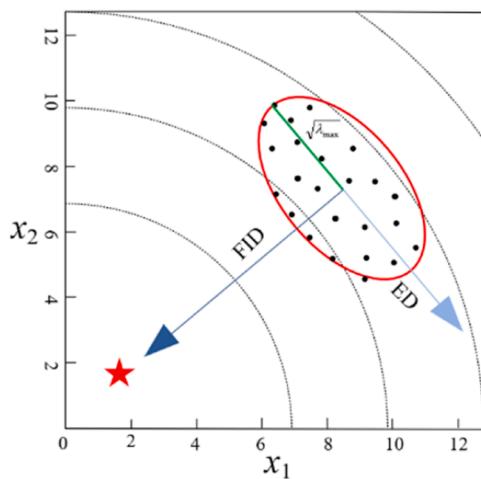


Fig. 3. The search behavior of the original EMNA.

The revision operations are executed in the direction of fitness improvement. The revised solutions are recycled for modeling at the next generation, which improves both population diversity and solution accuracy. Fig. 2 shows the composition of the solutions obtained through Kalman filtering. The red and black dots represent the optimal and the current solutions, while the blue triangles and the green squares are the predicted and the revised solutions, which are closer to the optimal solution than the ones before Kalman filtering.

The revised solutions are concentrated in the potential regions. The elite strategy guides the population to approach the direction of the optimal solution. The proposed KFHLEDA utilizes the specific knowledge embodied by the dominant and elite individuals. A faster search is achieved.

According to the definition of mean in Eq. (4), the improved mean value after revision, denoted by  $\tilde{\mu}(k)$ , is calculated as Eq. (12). The proposed KFHLEDA incorporates the weighted fitness of the elite individuals in the archive to modify the mean to enhance the influence of the dominant information.

$$\tilde{\mu}(k) = \frac{1}{s} \sum_{i=1}^s (x_i(k) + g_i(k)(f(x_{pbest}(k)) - f(x_i(k)))\pi_i(k)) \quad (12)$$

$$\begin{aligned} \tilde{C}(k) &= \frac{1}{s} \sum_{i=1}^s (x_i(k) - \tilde{\mu}(k)) = \frac{1}{s} \sum_{i=1}^s (x_i(k) - \mu(k) - \Delta(k))(x_i(k) - \mu(k) - \Delta(k))^T \\ &= \frac{1}{s} \sum_{i=1}^s (x_i(k) - \mu(k))(x_i(k) - \mu(k))^T - \frac{1}{s} \sum_{i=1}^s (x_i(k) - \mu(k))(\Delta(k))^T \\ &\quad - \frac{1}{s} \sum_{i=1}^s \Delta(k)(x_i(k) - \mu(k))^T + \Delta(k)(\Delta(k))^T = \frac{1}{s} \sum_{i=1}^s (x_i(k) - \mu(k))(x_i(k) - \mu(k))^T \\ &\quad - \frac{1}{s} \sum_{i=1}^s x_i(k)(\Delta(k))^T + \mu(k)(\Delta(k))^T - \frac{1}{s} \sum_{i=1}^s \Delta(k)(x_i(k))^T + \Delta(k)(\mu(k))^T + \Delta(k)(\Delta(k))^T \\ &= C(k) + \Delta(k)(\Delta(k))^T \end{aligned} \quad (15)$$

The mean after revision explores more potential solution spaces than that without revision. The improved covariance matrix, denoted by  $\tilde{C}(k)$ , is obtained as shown in Eq. (13).

$$\tilde{C}(k) = \frac{1}{s} \sum_{i=1}^s (x_i(k) - \tilde{\mu}(k))(x_i(k) - \tilde{\mu}(k))^T \quad (13)$$

The revision improvement of the mean, represented by  $\Delta(k)$ , is

shown in Eq. (14).

$$\Delta(k) = \tilde{\mu}(k) - \mu(k) \quad (14)$$

After the operations of enhanced Kalman filtering, the new mean is adjusted to explore more potential solution spaces, which leads to a more prospective search direction and enables KFHLEDA to achieve promising solutions along the direction of fitness improvement. The revision is guided by elite solutions, which speed up the search and reduce the time consumption caused by random searches.

In Figs. 3–4, the red and black dots represent the optimal and current solutions, the green squares are the revision solutions, FID represents the direction of fitness improvement, and ED represents the evolution direction, which is the direction of the principal axis of the probability density ellipse. In the traditional EMNA, ED is perpendicular to FID and the search scope is small, as shown in Fig. 3. In the process of Kalman filtering, the prediction, observation, and revision operations guided by the elite strategy are embedded into the EMNA framework. ED becomes parallel to FID and the search scope turns extensive, as shown in Fig. 4. The adjustments made through the Kalman filtering enable KFHLEDA to modify the search direction and scope. The search efficiency and solution accuracy are improved. Section 3.5 conducts a detailed theoretical analysis to prove the above conclusions.

Moreover, a historical archive is integrated into KFHLEDA to make full use of the promising information from various types of individuals, which promotes population diversity. To address the specific search problem, the Kalman filtering revision procedure is divided into the first and the second revisions, which improves the accuracy of the algorithm.

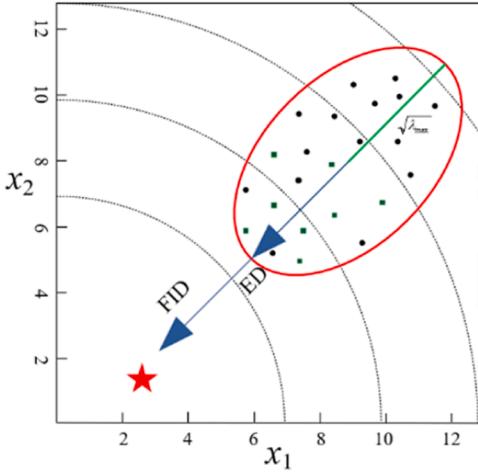
### 3.5. Theoretical analysis

The covariance matrix determines the search scope and direction [35]. Fig. 3 indicates that the individuals generated by the traditional EMNA evolve in an unfavorable direction that is perpendicular to the direction of fitness improvement. After applying the enhanced Kalman filtering, more potential revision solutions are achieved. The revised mean and covariance matrix results in the principal axis of a probability density ellipsoid being oriented towards the direction of  $\Delta(k) = \tilde{\mu}(k) - \mu(k)$ , which is the direction of fitness improvement.

Eq. (15) illustrates that the covariance matrix  $\tilde{C}(k)$  improved by enhanced Kalman filtering is the rank-one correction of the original covariance matrix  $C(k)$ .

The analysis of the following two mathematical lemmas illustrates that the estimation method of the enhanced Kalman filtering extends the search range and adjusts the search direction adaptively.

**Lemma 1.** If the selected set of samples yields  $\tilde{\mu}(k) \neq \mu(k)$ , the



**Fig. 4.** The search behavior after Kalman filtering.

probability density ellipsoid bound by the covariance matrix after Kalman filtering provides coverage that is greater than or equal to that without Kalman filtering.

**Proof.** In the Gaussian distribution, the principal axis direction of the probability density ellipsoid corresponds to the eigenvector of the covariance matrix, and the length of the half-axis is equal to the square root of the eigenvalue [72]. As the length of the principal axis increases, the coverage of the probability density ellipsoid expands accordingly. The eigenvalues of  $\tilde{C}(k)$  and  $C(k)$  are defined as  $\tilde{\lambda}_i$  and  $\lambda_i$ , respectively, where  $i = 1, 2, \dots, m$ . According to Eq. (5), it is easy to prove that the covariance matrix is positive semi-definite. Therefore,  $C(k)$  is decomposed into  $C(k) = \psi(k)\Lambda(k)\psi(k)^T$ , where  $\Lambda(k)$  is a diagonal matrix with the eigenvalues as the diagonal elements, and  $\psi(k)$  is an orthogonal matrix. From Eq. (14),  $\Delta(k) = \tilde{\mu}(k) - \mu(k)$ . When  $\varphi(k) = \psi(k)^T\Delta(k)$ , the following conclusion holds.

$$\begin{aligned} \psi(k)^T\tilde{C}(k)\psi(k) &= \psi(k)^T(C(k) + \Delta(k)\Delta(k)^T)\psi(k) \\ &= \psi(k)^TC(k)\psi(k) + \psi(k)^T\Delta(k)\Delta(k)^T\psi(k) \\ &= \psi(k)^TC(k)\psi(k) + (\psi(k)^T\Delta(k))(\psi(k)^T\Delta(k))^T \\ &= \Lambda(k) + \varphi(k)\varphi(k)^T \end{aligned} \quad (16)$$

Based on the conclusion of Lemma 1.4 in [73], the eigen polynomial of  $\Lambda(k) + \varphi(k)\varphi(k)^T$  is equal to that of  $\tilde{C}(k)$ .

$$\begin{aligned} p(\lambda) &= \prod_{i=1}^m (\lambda - \lambda_i) = \det(\lambda E - \Lambda(k) - \varphi(k)\varphi(k)^T) \\ &= \det(\lambda E - \Lambda(k))\det(E - (\lambda E - \Lambda(k))^{-1}\varphi(k)\varphi(k)^T) \\ &= \prod_{i=1}^m (\lambda - \lambda_i) \left( 1 - \sum_{j=1}^m \frac{\varphi_j^2}{\lambda - \lambda_j} \right) \\ &= \prod_{i=1}^m (\lambda - \lambda_i) - \sum_{j=1}^m \left( \varphi_j^2 \prod_{i=1, i \neq j}^m (\lambda - \lambda_i) \right) \end{aligned} \quad (17)$$

where  $E$  represents an identity matrix. When  $\lambda = 0$ , Eq. (17) is written as follows.

$$\prod_{i=1}^m \tilde{\lambda}_i = \sum_{i=1}^m \lambda_i + \sum_{j=1}^m \left( \varphi_j^2 \prod_{i=1, i \neq j}^m \lambda_i \right) > \prod_{i=1}^m \lambda_i \quad (18)$$

**Lemma 1** is proved.

According to [72], the length of the half-axis equals the square root of  $\lambda_i$ . Combined with the analysis of Lemma 1, the proposed KFHLEDA expands the coverage of the probability density ellipsoid compared to

the traditional EMNA, hence, the search scope of EMNA is expanded.

**Lemma 2.** For a selected set of samples, when  $\tilde{\mu}(k) \neq \mu(k)$ , the angle between  $\Delta(k)$  and the major axis of the probability density ellipsoid revised by Kalman filtering is less than or equal to that without Kalman filtering.

**Proof.** The eigenvectors corresponding to the maximum eigenvalues  $\tilde{\lambda}_m$  and  $\lambda_m$  of  $\Lambda(k) + \varphi(k)\varphi(k)^T$  and  $\Lambda(k)$  are defined as  $\tilde{\chi}(k)$  and  $\chi(k)$ , respectively. Based on Eq. (16) in Lemma 1,  $\Lambda(k) + \varphi(k)\varphi(k)^T$  and  $\Lambda(k)$  are orthogonal transformations of  $\tilde{C}(k)$  and  $C(k)$ , respectively. Thus, Lemma 2 is equivalent to certify that  $\angle(\varphi(k), \tilde{\chi}(k)) \leq \angle(\varphi(k), \chi(k))$ .

An acute angle is employed to represent the angle between the two vectors. Evidently,  $\varphi(k)^T\tilde{\chi}(k) \geq 0$ ,  $\varphi(k)^T\chi(k) \geq 0$ .

**Case 1.**  $\varphi(k)^T\chi(k) = 0$ ,  $\angle(\varphi(k), \chi(k)) = 90^\circ$ .  $\angle(\varphi(k), \tilde{\chi}(k)) \leq 90^\circ$ ,  $\angle(\varphi(k), \tilde{\chi}(k)) \leq$

$\angle(\varphi(k), \chi(k))$ . Lemma 2 is proven.

**Case 2.** If  $\varphi(k)^T\chi(k) \neq 0$ ,  $\tilde{\lambda}_m > \lambda_m$  is valid based on Eq. (18). According to the definition of eigenvalues, Eq. (19) is established.

$$(\Lambda(k) + \varphi(k)\varphi(k)^T)\tilde{\chi}(k) = \tilde{\lambda}_m\tilde{\chi}(k) \quad (19)$$

If  $\varphi(k)^T\tilde{\chi}(k) = 0$  in Eq. (19),  $\Lambda(k)\tilde{\chi}(k) = \tilde{\lambda}_m\tilde{\chi}(k)$ .  $\tilde{\lambda}_m$  is the eigenvalue of  $\Lambda(k)$ , which is a contrary conclusion.  $\varphi(k)^T\tilde{\chi}(k) \neq 0$ . Eq. (19) is converted to Eq. (20).

$$\tilde{\chi}(k) = (\tilde{\lambda}_m E - \Lambda(k))^{-1}\varphi(k)\varphi(k)^T\tilde{\chi}(k) \quad (20)$$

The  $\lambda$  in Eq. (17) is replaced with  $\tilde{\lambda}_m$  as shown in Eq. (21).

$$p(\tilde{\lambda}_m) = \left( 1 - \sum_{i=1}^m \frac{\varphi_i^2}{\tilde{\lambda}_m - \lambda_i} \right) \prod_{i=1}^m (\tilde{\lambda}_m - \lambda_i) = 0 \quad (21)$$

Eq. (21) is transformed into Eq. (22).

$$\sum_{i=1}^m \frac{\varphi_i^2}{\tilde{\lambda}_m - \lambda_i} = 1 \quad (22)$$

Eq. (20) is obtained based on Eq. (18).

$$1 \geq \frac{\varphi_m^2}{\tilde{\lambda}_m - \lambda_m} > 0 \quad (23)$$

$$\begin{aligned} \cos \angle(\varphi(k), \tilde{\chi}(k)) &= \frac{\varphi(k)^T\tilde{\chi}(k)}{\|\varphi(k)\| \|\tilde{\chi}(k)\|} \\ &= \frac{\varphi(k)^T\tilde{\chi}(k)}{\|\varphi(k)\| \|(\tilde{\lambda}_m E - \Lambda(k))^{-1}\varphi(k)\| \|\varphi(k)^T\tilde{\chi}(k)\|} \\ &= \frac{1}{\|\varphi(k)\| \sqrt{\sum_{i=1}^m \frac{\varphi_i^2}{(\tilde{\lambda}_m - \lambda_i)^2}}} \\ &\geq \frac{1}{\|\varphi(k)\| \sqrt{\sum_{i=1}^m \frac{\varphi_i^2}{(\tilde{\lambda}_m - \lambda_i)^2}}} \sqrt{\frac{\varphi_m^2(k)}{(\tilde{\lambda}_m - \lambda_m)}} \\ &= \frac{|\varphi_m(k)|}{\|\varphi(k)\| \sqrt{(\tilde{\lambda}_m - \lambda_m) \sum_{i=1}^m \frac{\varphi_i^2}{(\tilde{\lambda}_m - \lambda_i)^2}}} \\ &\geq \frac{|\varphi_m(k)|}{\|\varphi(k)\| \sqrt{(\tilde{\lambda}_m - \lambda_m) \sum_{i=1}^m \frac{\varphi_i^2}{(\tilde{\lambda}_m - \lambda_m)(\tilde{\lambda}_m - \lambda_i)}}} \end{aligned} \quad (24)$$

**Table 1**

MANOVA results for parameter settings of KFHLEDA.

Source	Sum of squares	Degrees of freedom	Mean Square	F-ratio	p-value
$NP_{max}$	582.95	3	194.317	10.38	0.0001
$\tau$	107.71	3	35.902	1.92	0.1506
$h$	54.42	3	18.14	0.97	0.4219
$NP_{max} * \tau$	339.62	9	37.736	2.01	0.0771
$NP_{max} * h$	308.31	9	34.25	1.87	0.1203
$\tau * h$	711.11	9	79.013	4.22	0.0017
Error	505.66	27	18.728		
Total	3054.18	63			

$$\begin{aligned}
&= \frac{|\varphi_m(k)|}{\|\varphi(k)\| \sqrt{\sum_{i=1}^m \frac{\varphi_i^2}{(\lambda_m - \lambda_i)}}} \\
&= \frac{|\varphi_m(k)|}{\|\varphi(k)\| \times 1} \\
&= \frac{\varphi(k)^T \chi(k)}{\|\varphi(k)\| \|\chi(k)\|} \\
&= \cos\angle(\varphi(k), \chi(k)) \\
\angle(\varphi(k), \tilde{\chi}(k)) &\leq \angle(\varphi(k), \chi(k))
\end{aligned} \tag{25}$$

Based on the detailed analysis presented above, the conclusion in Eq. (25) confirms that the angle between  $\Delta(k)$  and the major axis of the probability density ellipsoid revised by Kalman filtering is less than or equal to that without Kalman filtering. This conclusion indicates that the direction of fitness improvement aligns with the evolutionary direction after the filtering.

#### 4. Experimental results and analysis

##### 4.1. Test suites

The KFHLEDA is tested in the IEEE CEC 2017 test suite to demonstrate and evaluate the effectiveness [74]. The algorithm is run on 29 benchmark test functions. The 29 functions in the test suite include two unimodal functions  $f_1, f_3$ , seven simple multimodal functions  $f_4 - f_{10}$ , ten hybrid functions  $f_{11} - f_{20}$ , and ten composition functions  $f_{21} - f_{30}$ . It is difficult to obtain the optimal solution due to the complex structural characteristics and multiple local optima of most functions.

The performance of the proposed KFHLEDA and comparison algorithms is evaluated by the experiments on 10, 30, 50, and 100 di-

mensions, denoted as 10D, 30D, 50D, and 100D, respectively. Each function is independently run 51 times.  $\max\_nfe = 10000 D$ , which is taken as the experimental termination condition for each run. The function error (the difference between the best solution and the known global optimal of the function by each run) is taken as an evaluation criterion. When the error is less than  $10^{-8}$ , it is set to 0. All the algorithms in this study are evaluated under the same experimental conditions and environments. The experiments are performed by MATLAB on a PC with a 3.4 GHz Intel(R) Core (TM) i7-6700 CPU, and 8 GB of RAM. The source code and material data can be downloaded at <https://github.com/Znnalgorithms/KFHLEDA.git>.

##### 4.2. Parameters analysis

The parameter setting affects the performance of the algorithms. The proposed KFHLEDA includes three parameters, which are the initial size of the population  $NP_{max}$ , the selection rate  $\tau$ , and the size of the historical archive  $h$ .

Selecting a suitable  $NP_{max}$  is crucial. A too-large value leads to slow convergence and wastes computing resources. While a too-small value reduces population diversity, exploration is weak. The  $\tau$  affects the selection of the dominant solutions and directly determines the quality of solutions at each generation. If  $\tau$  is small, the influence of the dominant population on subsequent iterations is weakened. It is easy to fall into the local optimum, which affects the solution accuracy. When  $\tau$  is too colossal, the useless solutions are introduced except for the superior ones.

The historical archive  $h$  determines the amount of historical information that is retained by the algorithm. The degree of learning the dominant solutions makes the acquisition depend on both the current and the historical information. If  $h$  is oversized, the search space contains superabundant inferior information. The convergence speed is retarded. Similarly, if  $h$  is undersize, the dominant information in the archive is hard to learn adequately. Therefore, selecting an appropriate value of  $h$  is critical for achieving high search efficiency.

Various combinations of the three parameters result in diverse performances. The differences are analyzed through the results in the Design of the Experiments (DOE) and Multifactor Analysis of Variance (MANOVA) to achieve a promising performance [75]. The values are set as  $NP_{max} = \{1000, 2000, 3000, 4000\}$ ,  $\tau = \{0.15, 0.25, 0.35, 0.45\}$ , and  $h = \{100, 200, 300, 400\}$ . The total number of the parameter combinations is  $4 \times 4 \times 4 = 64$ . Each combination is run on 29 functions until the termination condition is met. There are three basic assumptions in MANOVA including normality, variance homogeneity, and sample independence. The hypotheses are satisfied through the experiments. The number of samples in the experiments is large, and the

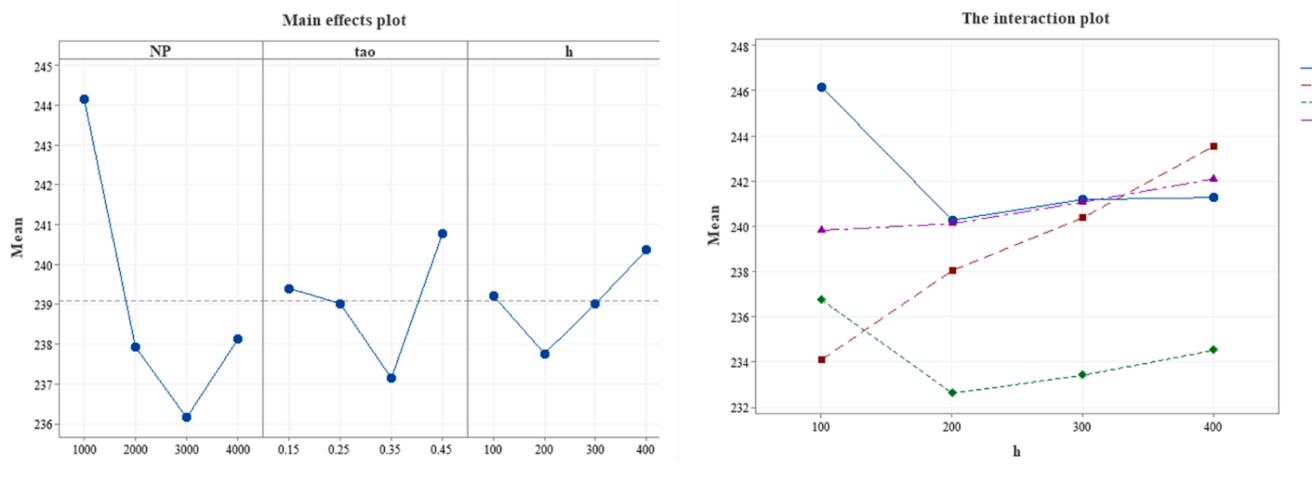


Fig. 5. Main effects plot and interaction plot of three parameters.

**Table 2**

The experimental results of KFHLEDA and other comparison algorithms (50D).

Fun	jSO	LSHADE-SPACMA	LSHADE-cnEpSin	NL-SHADE-RSP	PID-DE	EA4eig	LSHADE	APBIL	CMA-ES	EGNA	EMNA	EDA2	ACSEDA	EMSM	AEDDDE	KFHLEDA	
F1	Mean	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	5.85E+03	<b>0.00E+00</b>	<b>0.00E+00</b>	1.49E+06	<b>0.00E+00</b>	1.60E+06	4.49E+10	<b>0.00E+00</b>	<b>0.00E+00</b>	9.03E+08	1.63E-02	<b>0.00E+00</b>
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	6.49E+03	0.00E+00	0.00E+00	1.65E+06	0.00E+00	6.74E+05	4.60E+09	0.00E+00	0.00E+00	5.53E+08	3.37E-03	0.00E+00
F3	Mean	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	1.73E+05	<b>0.00E+00</b>	<b>0.00E+00</b>	5.20E+04	<b>0.00E+00</b>	1.51E+05	9.66E+04	<b>0.00E+00</b>	<b>0.00E+00</b>	3.18E+04	2.13E-07	<b>0.00E+00</b>
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.57E+04	0.00E+00	0.00E+00	1.12E+04	0.00E+00	1.11E+04	1.20E+04	0.00E+00	0.00E+00	7.48E+03	0.00E+00	0.00E+00
F4	Mean	6.65E+01	3.10E+01	5.74E+01	2.69E+01	1.03E+02	5.48E+01	9.72E+01	2.65E+02	<b>0.00E+00</b>	1.92E+02	9.63E+03	8.60E+01	2.15E+02	4.95E+02	6.49E+00	1.45E+01
	Std	5.06E+01	3.48E+01	4.72E+01	6.69E+00	5.71E+01	4.06E+01	4.45E+01	3.19E+01	0.00E+00	5.76E+01	1.59E+03	5.09E+01	3.24E+01	1.63E+02	4.27E+01	6.57E+00
F5	Mean	1.51E+01	5.58E+00	2.63E+01	1.32E+02	3.69E+02	7.25E+01	1.19E+01	5.34E+01	1.33E+01	8.64E+00	2.51E+02	2.59E+01	3.65E+00	2.37E+02	<b>6.94E-01</b>	3.53E+00
	Std	5.77E+00	2.30E+00	6.46E+00	2.18E+01	1.56E+01	1.65E+01	2.43E+00	1.47E+01	3.32E+00	6.38E+00	1.96E+01	5.62E+00	2.41E+00	3.58E+01	1.04E+00	1.49E+00
F6	Mean	9.56E-06	<b>0.00E+00</b>	6.40E-07	<b>0.00E+00</b>	1.04E-04	1.18E-06	4.09E-05	2.88E-07	3.95E-00	3.56E-01	3.11E+01	2.28E-03	<b>0.00E+00</b>	4.34E-01	7.15E-04	5.52E-06
	Std	1.07E-06	0.00E+00	7.15E-07	0.00E+00	4.25E-05	1.89E-06	2.08E-04	5.24E-07	1.53E+01	5.05E+00	5.65E+00	9.41E-03	0.00E+00	4.93E+00	1.17E-04	1.73E-06
F7	Mean	1.48E+02	5.67E+01	7.71E+01	1.56E+02	4.22E+02	1.22E+02	6.35E+01	8.53E+01	6.19E+01	7.08E+01	4.51E+02	1.00E+02	5.66E+01	2.48E+02	3.27E+02	<b>5.57E+01</b>
	Std	4.27E+01	8.44E-01	6.13E+00	2.49E+01	1.24E+01	1.19E+01	1.62E+00	7.30E+00	2.34E+00	6.05E+00	9.42E+01	1.18E+01	1.38E+00	5.99E+01	1.51E+01	1.24E+00
F8	Mean	1.56E+01	5.72E+00	2.66E+01	1.36E+02	3.66E+02	6.79E+01	1.21E+01	5.22E+01	1.19E+01	8.81E+00	2.66E+02	2.45E+01	4.29E+00	2.37E+02	3.25E+00	<b>2.71E+00</b>
	Std	7.02E+00	2.25E+00	7.22E+00	2.18E+01	1.30E+01	1.53E+01	2.29E+00	1.09E+01	2.31E+00	7.78E+00	2.26E+01	7.75E+00	3.25E+00	3.60E+01	1.89E+00	1.61E+00
F9	Mean	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	1.10E-01	4.43E-01	<b>0.00E+00</b>	7.35E+00	1.22E+04	8.62E-01	9.64E+03	<b>0.00E+00</b>	<b>0.00E+00</b>	5.21E+03	1.05E-07	<b>0.00E+00</b>
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.72E+00	4.14E-01	6.42E-01	0.00E+00	5.98E+00	8.68E+01	6.51E-01	1.90E+03	0.00E+00	0.00E+00	1.13E+03	0.00E+00
F10	Mean	8.94E+03	3.64E+03	3.20E+03	3.00E+03	1.34E+04	3.80E+03	3.14E+03	3.51E+03	3.71E+03	1.42E+04	1.20E+04	1.02E+03	5.58E+02	5.68E+03	3.98E+03	<b>2.06E+02</b>
	Std	5.70E+02	5.85E+02	2.97E+02	4.87E+02	4.11E+02	4.66E+02	2.68E+02	7.09E+02	9.70E+02	4.13E+02	5.79E+02	3.62E+02	3.44E+02	6.78E+02	8.74E+02	1.71E-02
F11	Mean	2.67E+01	3.13E+01	2.19E+01	1.36E+04	1.57E+02	5.79E+01	4.73E+01	1.66E+03	1.13E+02	2.69E+03	6.09E+03	3.03E+01	6.68E+01	5.20E+02	2.02E+01	<b>1.91E+01</b>
	Std	3.66E+00	4.60E+00	1.74E+00	7.56E+03	1.39E+01	1.62E+01	8.52E+00	9.88E+02	3.31E+01	9.74E+02	1.52E+03	4.18E+00	3.12E+01	1.51E+02	1.20E+00	9.21E-01
F12	Mean	1.67E+03	1.61E+03	1.39E+03	1.37E+04	1.72E+05	3.38E+04	2.33E+03	7.57E+06	2.63E+03	8.85E+05	2.38E+10	1.61E+03	<b>4.54E+02</b>	8.60E+07	1.31E+03	4.62E+02
	Std	4.89E+02	4.96E+02	3.31E+02	6.72E+03	1.18E+05	9.41E+04	4.84E+02	4.22E+06	4.04E+02	1.20E+05	4.19E+09	4.39E+02	1.67E+02	1.22E+08	4.73E+02	1.94E+02
F13	Mean	3.66E+01	3.55E+01	6.67E+01	4.92E+01	1.09E+03	8.92E+01	5.75E+01	2.65E+03	7.21E+01	1.30E+05	9.03E+09	7.27E+01	8.43E+01	1.83E+04	1.32E+02	<b>1.17E+01</b>
	Std	1.83E+01	1.44E+01	3.55E+01	1.85E+01	1.98E+03	3.78E+01	3.55E+01	1.90E+03	4.93E+01	1.27E+04	3.30E+09	3.46E+01	4.58E+01	1.16E+04	4.25E+01	1.44E+01
F14	Mean	2.42E+01	2.92E+01	2.67E+01	3.71E+01	1.40E+02	5.74E+01	2.94E+01	9.92E+05	2.74E+02	2.23E+04	5.36E+06	2.34E+01	2.14E+01	1.52E+04	2.05E+01	<b>1.89E+01</b>
	Std	1.75E+00	3.32E+00	2.34E+00	9.15E+00	8.05E+00	1.59E+01	3.06E+00	6.34E+05	9.02E+01	1.23E+04	5.36E+06	1.84E+00	1.50E+00	1.15E+04	2.90E+00	5.85E+00
F15	Mean	2.40E+01	3.12E+01	2.50E+01	1.20E+03	1.27E+02	5.96E+01	4.03E+01	3.89E+03	4.15E+02	2.01E+04	1.48E+07	2.37E+01	2.60E+01	2.19E+03	<b>1.78E+01</b>	1.79E+01
	Std	2.42E+00	5.63E+00	3.19E+00	3.64E+02	1.04E+01	2.55E+01	9.26E+00	2.43E+03	1.17E+02	1.59E+03	4.36E+07	3.16E+00	4.15E+00	1.30E+03	1.15E+00	1.45E+00
F16	Mean	7.57E+02	3.90E+02	2.97E+02	6.64E+02	2.98E+03	7.00E+02	3.82E+02	9.23E+02	2.00E+02	7.32E+02	7.89E+02	1.72E+02	1.28E+02	1.69E+03	1.32E+02	<b>1.27E+02</b>
	Std	1.41E+02	1.87E+02	1.15E+02	1.60E+02	2.54E+02	2.23E+02	1.50E+02	3.08E+02	9.96E+01	1.33E+02	1.75E+02	8.60E+01	1.89E+00	4.74E+02	1.01E+01	1.81E+00
F17	Mean	4.97E+02	2.77E+02	2.21E+02	3.72E+02	1.76E+03	4.50E+02	2.53E+02	5.30E+02	3.39E+02	3.57E+02	9.96E+02	<b>6.98E+01</b>	1.33E+02	1.37E+03	2.34E+02	1.03E+02
	Std	1.04E+02	8.62E+01	8.06E+01	1.15E+02	2.41E+02	1.34E+02	7.71E+01	2.02E+02	1.35E+02	3.02E+02	4.23E+02	6.59E+01	8.71E+01	2.98E+02	1.03E+02	6.77E+01
F18	Mean	2.46E+01	3.08E+01	2.40E+01	3.18E+01	1.20E+03	4.84E+01	4.13E+01	2.73E+06	2.50E+02	1.53E+05	3.09E+06	2.19E+01	<b>2.10E+01</b>	7.62E+04	2.13E+01	<b>2.10E+01</b>
	Std	2.05E+00	4.71E+00	1.90E+00	6.25E+00	1.08E+03	1.58E+01	1.37E+01	1.42E+06	9.44E+01	5.21E+04	3.14E+06	1.10E+00	1.84E+01	4.63E+04	1.03E-01	2.83E-01
F19	Mean	1.53E+01	2.12E+01	1.72E+01	1.69E+01	7.29E+01	2.28E+01	2.58E+01	1.32E+04	1.40E+02	9.12E+04	4.14E+06	1.04E+01	7.58E+00	3.45E+03	1.08E+01	<b>6.67E+00</b>
	Std	3.51E+00	3.55E+00	2.61E+00	4.09E+00	4.43E+00	5.79E+00	3.68E+03	5.99E+01	4.25E+03	1.13E+07	2.37E+00	2.08E+00	4.13E+03	1.69E+00	1.18E+00	
F20	Mean	3.74E+02	1.93E+02	1.08E+02	2.58E+02	1.60E+03	2.24E+02	1.52E+02	3.37E+02	1.55E+03	4.86E+02	6.33E+02	2.49E+01	3.56E+01	1.01E+03	8.12E+01	2.47E+01
	Std	7.09E+01	1.25E+02	1.95E+01	9.70E+01	2.54E+02	1.17E+02	5.28E+01	2.42E+02	2.23E+02	2.46E+02	3.19E+02	3.34E+00	1.73E+01	1.93E+02	2.95E+01	3.66E+00
F21	Mean	2.18E+02	2.15E+02	2.27E+02	3.05E+02	5.67E+02	2.71E+02	2.13E+02	2.50E+02	2.14E+02	3.71E+02	5.04E+02	2.25E+02	2.06E+02	5.23E+02	2.06E+02	
	Std	4.63E+00	9.68E+00	7.14E+00	9.37E+01	1.42E+01	1.41E+01	2.39E+00	1.29E+01	4.09E+00	9.43E+00	2.62E+01	5.90E+00	2.91E+00	6.22E+01	1.78E+00	1.86E+00
F22	Mean	3.45E+03	1.33E+03	1.67E+03	6.41E+02	1.33E+04	3.76E+03	2.12E+03	2.53E+03	6.72E+02	3.25E+02	4.02E+03	1.24E+02	2.80E+02	3.71E+03	<b>1.00E+02</b>	<b>1.00E+02</b>
	Std	4.30E+03	1.88E+03	1.69E+03	1.42E+03	1.93E+03	1.89E+03	1.78E+03	2.07E+03	3.44E+02	1.74E+03	6.69E+02	1.68E+02	3.15E+02	3.23E+03	1.69E-04	0.00E+00
F23	Mean	4.44E+02	4.40E+02	5.26E+02	7.90E+02	5.01E+02	4.30E+02	5.06E+02	4.40E+02	8.97E+02	1.14E+03	4.37E+02	4.26E+02	1.30E+03	4.20E+02	<b>3.98E+02</b>	
	Std	8.20E+00	6.88E+00	8.21E+00	8.98E+01	1.33E+01	1.74E+01	4.26E+00	2.11E+01	1.12E+01	2.30E+01	7.67E+01	1.46E+01	1.20E+01	2.12E+02	7.91E+00	6.63E+00
F24	Mean	5.12E+02	5.14E+02	5.14E+02	5.01E+02	8.61E+02	5.67E+02	5.07E+02	5.66E+02	5.11E+02	1.08E+03	1.86E+03	5.11E+02	4.92E+02	1.20E+03	5.02E+02	<b>4.55E+02</b>
	Std	4.67E+00	6.53E+00	6.75E+00	1.85E+02	1.33E+01	1.71E+01	2.74E+00	1.83E+01	4.11E+00	1.38E+02	2.12E+02	1.50E				

**Table 3**

The experimental results of KFHLEDA and other comparison algorithms (100D).

Fun	jSO	LSHADE-SPACMA	LSHADE-cnEpSin	NL-SHADE-RSP	PID-DE	EA4eig	LSHADE	APBIL	CMA-ES	EGNA	EMNA	EDA2	ACSEDA	EMSM	AEDDDE	KFHLEDA		
F1	Mean	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	6.63E+03	7.01E-02	<b>0.00E+00</b>	8.69E+07	<b>0.00E+00</b>	2.43E+07	1.57E+11	2.67E+09	6.35E-02	1.57E+10	6.26E+01	<b>0.00E+00</b>	
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	8.40E+03	1.73E-01	0.00E+00	1.20E+08	0.00E+00	1.26E+07	8.10E+09	2.05E+09	6.50E-03	5.34E+09	1.53E+01	0.00E+00	
F3	Mean	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	2.37E+01	5.54E+05	7.58E-07	8.45E-07	2.30E+05	<b>0.00E+00</b>	1.68E+06	2.54E+05	1.93E+04	9.65E+03	1.10E+00	1.96E+04	5.41E-05	0.00E+00
	Std	1.00E-07	0.00E+00	0.00E+00	3.33E+01	4.74E+04	9.72E-07	1.17E-06	2.53E+04	0.00E+00	2.56E+05	1.93E+04	2.77E+04	1.58E-01	1.40E+05	3.24E-04	<b>0.00E+00</b>	
F4	Mean	1.94E+02	1.99E+02	1.97E+02	<b>1.25E+01</b>	2.32E+02	1.49E+02	1.96E+02	6.49E+02	1.73E+01	3.91E+03	3.66E+04	4.02E+02	4.45E+02	2.90E+03	2.21E+02	1.89E+02	
	Std	1.75E+01	1.02E+01	8.85E+00	2.46E+01	3.14E+01	9.35E+01	9.64E+00	2.53E+04	5.50E-01	7.98E+02	4.61E+03	7.53E+01	2.73E+01	8.32E+02	2.42E+00	4.21E+00	
F5	Mean	3.22E+01	1.19E+01	5.60E+01	4.84E+02	8.42E+02	2.15E+02	3.92E+01	1.57E+02	3.45E+01	4.53E+01	9.16E+02	3.03E+02	2.85E+01	6.29E+02	6.20E+01	<b>1.09E+01</b>	
	Std	4.96E+00	3.07E+00	6.42E+00	7.36E+01	2.73E+01	4.58E+01	5.59E+00	3.70E+01	5.59E+00	6.51E+00	4.40E+01	3.74E+01	1.05E+01	4.42E+01	7.75E+01	3.78E+00	
F6	Mean	1.58E-03	<b>0.00E+00</b>	5.95E-05	<b>0.00E+00</b>	6.51E-03	3.16E-04	7.56E-03	6.45E-03	2.46E+01	9.37E-01	5.60E+01	9.86E+00	1.01E-03	5.22E+01	7.32E-03	3.27E-04	
	Std	3.59E-04	0.00E+00	2.19E-05	0.00E+00	1.91E-02	1.94E-04	4.50E-03	9.12E-03	2.94E+01	2.03E-02	6.80E+00	2.51E+00	4.73E-05	3.12E+00	6.98E-04	7.86E-04	
F7	Mean	3.50E+02	<b>1.12E+02</b>	1.62E+02	5.24E+02	9.60E+02	3.41E+02	3.39E+02	2.03E+02	1.33E+02	5.16E+02	1.73E+03	6.36E+02	4.83E+02	1.20E+03	8.07E+02	6.11E+02	
	Std	3.02E+01	1.59E+00	5.75E+00	7.93E+01	2.90E+01	4.46E+01	3.82E+00	2.17E+01	4.68E+00	5.82E+00	8.87E+01	7.64E+01	1.18E+02	1.84E+02	1.74E+01	3.54E+01	
F8	Mean	3.03E+01	<b>1.11E+01</b>	5.53E+01	5.10E+02	8.41E+02	2.23E+02	3.67E+01	1.71E+02	3.41E+01	6.76E+01	1.01E+03	3.12E+02	2.19E+01	6.97E+02	4.50E+01	1.46E+01	
	Std	5.29E+00	2.94E+00	9.73E+00	7.71E+01	1.89E+01	4.04E+01	4.37E+00	3.16E+01	5.35E+00	4.46E+00	4.08E+01	3.75E+01	5.55E+01	7.40E+01	7.38E+01	5.00E+00	
F9	Mean	2.02E-02	<b>0.00E+00</b>	<b>0.00E+00</b>	1.29E+04	1.18E+01	2.91E+01	4.24E+01	3.29E+02	2.21E+04	4.96E+00	3.84E+04	1.30E+03	4.55E-07	1.53E+04	3.63E-04	1.27E-04	
	Std	3.81E-02	0.00E+00	0.00E+00	2.65E+03	1.73E+01	5.84E+01	4.20E-01	1.02E+02	9.08E+01	4.31E-01	3.27E+03	8.06E+02	0.00E+00	1.56E+03	5.35E-05	2.35E-05	
F10	Mean	2.30E+04	1.02E+04	1.02E+04	9.05E+03	3.02E+04	1.14E+04	1.04E+04	9.47E+03	3.42E+04	2.55E+04	8.17E+03	5.88E+03	1.31E+04	1.94E+04	<b>5.31E+03</b>		
	Std	7.71E+02	9.35E+02	5.22E+02	1.34E+03	5.57E+02	8.84E+02	5.43E+02	1.41E+03	1.52E+03	1.37E+03	4.31E+03	1.03E+03	4.02E+03	7.64E+02	1.02E+03	1.82E-03	
F11	Mean	1.08E+02	4.68E+01	5.28E+01	3.54E+05	7.00E+02	3.32E+02	4.33E+02	2.66E+04	1.08E+03	8.83E+02	7.80E+04	5.91E+02	2.39E+02	1.39E+04	1.29E+02	<b>3.64E+01</b>	
	Std	3.41E+01	1.68E+01	3.46E+01	1.14E+05	8.63E+01	9.81E+01	1.11E+02	4.92E+03	3.79E+02	8.73E+01	8.91E+03	1.22E+02	1.06E+02	4.93E+03	5.61E+01	2.52E+01	
F12	Mean	1.83E+04	4.76E+03	4.50E+03	1.09E+05	9.15E+05	1.08E+05	2.17E+04	1.26E+08	5.36E+03	2.14E+06	8.84E+10	9.53E+05	2.66E+03	1.84E+09	7.60E+03	<b>2.34E+03</b>	
	Std	8.96E+03	7.62E+02	8.16E+02	3.71E+04	3.90E+05	4.88E+04	7.38E+03	2.92E+07	6.75E+02	4.14E+05	8.51E+09	8.48E+05	3.68E+02	8.19E+08	4.60E+02	5.12E+02	
F13	Mean	1.42E+02	1.28E+02	<b>1.25E+02</b>	6.90E+02	6.26E+03	1.70E+03	5.18E+02	1.83E+05	4.01E+03	7.03E+05	1.51E+10	4.54E+03	3.09E+02	3.78E+04	1.56E+02	2.32E+02	
	Std	4.16E+01	3.33E+01	4.11E+01	1.70E+02	6.45E+03	1.73E+03	2.46E+02	2.92E+07	8.88E+02	2.71E+04	2.03E+09	1.19E+03	8.55E+01	2.25E+04	3.83E+01	3.11E+01	
F14	Mean	6.52E+01	7.59E+01	5.12E+01	5.67E+03	1.53E+04	2.70E+02	2.58E+02	6.25E+06	3.72E+02	8.55E+05	1.66E+06	8.91E+01	2.87E+01	3.28E+05	7.29E+01	<b>2.35E+01</b>	
	Std	1.02E+01	1.22E+01	6.03E+00	4.45E+03	2.14E+04	9.22E+01	3.16E+01	2.76E+06	1.35E+02	2.23E+04	9.49E+05	2.03E+01	2.53E+01	1.29E+05	5.77E+01	1.68E+00	
F15	Mean	1.64E+02	1.15E+02	9.25E+01	3.13E+04	5.92E+03	6.28E+02	2.47E+02	1.31E+03	4.13E+02	4.47E+07	4.60E+09	4.46E+02	9.76E+01	6.66E+03	1.71E+02	<b>7.31E+01</b>	
	Std	1.02E+01	3.80E+01	3.26E+01	1.38E+04	7.50E+03	4.52E+02	4.61E+01	4.46E+02	8.31E+01	2.73E+04	1.17E+09	7.60E+01	3.14E+01	6.78E+03	4.64E+01	2.62E+01	
F16	Mean	2.84E+03	1.42E+03	1.19E+03	2.17E+03	7.92E+03	2.54E+03	1.67E+03	3.16E+03	4.45E+02	2.41E+03	5.31E+03	1.01E+03	2.94E+02	4.63E+03	1.08E+03	<b>1.40E+02</b>	
	Std	4.78E+02	4.67E+02	2.25E+02	3.52E+02	3.23E+02	3.29E+02	2.60E+02	7.02E+02	2.93E+02	4.68E+02	5.22E+02	2.97E+02	1.83E+02	7.31E+02	4.80E+02	1.21E+02	
F17	Mean	1.86E+03	1.11E+03	9.49E+02	1.46E+03	5.03E+03	1.66E+03	1.10E+03	1.97E+03	1.26E+03	2.52E+04	3.99E+04	4.62E+02	5.49E+02	3.78E+03	1.19E+03	<b>3.61E+02</b>	
	Std	2.57E+02	4.20E+02	1.96E+02	2.84E+02	2.31E+02	2.64E+02	2.21E+02	4.45E+02	3.60E+02	4.59E+02	3.81E+04	1.65E+02	2.74E+02	5.80E+02	3.68E+02	7.67E+01	
F18	Mean	1.87E+02	1.34E+02	7.13E+01	3.81E+04	2.33E+05	3.46E+03	2.42E+02	4.97E+06	2.25E+02	3.74E+05	8.94E+05	2.84E+02	2.38E+01	3.72E+05	3.52E+01	<b>2.32E+01</b>	
	Std	3.71E+01	3.42E+01	2.25E+01	1.45E+04	1.21E+05	1.85E+03	4.76E+01	1.93E+06	4.59E+01	1.68E+04	5.03E+05	5.63E+01	2.05E+00	1.46E+05	1.39E+01	1.13E+00	
F19	Mean	1.06E+02	7.11E+01	5.53E+01	7.30E+01	8.63E+03	1.90E+02	1.68E+02	7.52E+04	3.10E+02	1.52E+08	4.18E+09	1.56E+02	3.91E+01	8.55E+04	7.40E+01	<b>3.36E+01</b>	
	Std	2.04E+01	9.43E+00	6.20E+00	1.81E+01	1.02E+04	1.17E+02	2.26E+01	2.91E+05	7.75E+01	5.64E+07	1.09E+09	3.44E+01	1.54E+01	2.57E+01	3.41E+00	1.00E+00	
F20	Mean	2.16E+03	1.43E+03	1.13E+03	1.41E+03	4.96E+03	1.43E+03	1.54E+03	1.64E+03	3.82E+03	3.13E+03	2.03E+03	3.72E+02	5.99E+02	2.96E+03	1.11E+03	<b>4.15E-02</b>	
	Std	2.07E+02	3.41E+02	1.57E+02	2.61E+02	1.86E+02	2.49E+02	1.93E+02	4.69E+02	3.29E+02	5.03E+02	1.27E+03	8.02E+01	2.96E+02	4.40E+02	2.46E+02	1.01E+02	
F21	Mean	2.58E+02	2.41E+02	2.77E+02	6.96E+02	1.07E+03	8.56E+02	2.57E+02	3.82E+02	2.58E+02	3.76E+02	1.60E+03	5.28E+02	4.22E+02	1.45E+03	2.96E+02	<b>2.32E+02</b>	
	Std	7.17E+00	4.80E+00	5.88E+00	1.01E+02	2.29E+01	8.06E+01	7.21E+00	3.08E+01	4.85E+00	6.12E+00	7.54E+01	3.77E+01	2.15E+02	2.05E+02	6.95E+01	4.39E+00	
F22	Mean	2.32E+04	1.02E+04	1.04E+04	1.04E+04	3.08E+04	9.74E+03	1.12E+04	1.16E+04	1.73E+03	7.51E+03	2.03E+04	6.21E+03	2.57E+03	1.67E+04	1.15E+04	<b>1.25E+03</b>	
	Std	9.16E+02	8.34E+02	6.65E+02	1.52E+03	5.70E+02	1.04E+03	5.89E+02	2.97E+03	8.00E+02	8.16E+02	3.65E+03	4.52E+03	2.29E+03	2.70E+03	7.84E+03	1.64E+03	
F23	Mean	5.77E+02	5.82E+02	5.97E+02	7.46E+02	1.33E+03	7.56E+03	5.68E+02	8.13E+02	5.67E+02	1.47E+03	2.55E+03	9.57E+02	9.29E+02	1.82E+03	5.42E+02	<b>5.37E+02</b>	
	Std	9.65E+00	7.90E+00	8.32E+00	1.07E+02	4.78E+01	6.12E+01	9.87E+00	5.79E+01	1.23E+01	5.13E+01	1.40E+02	5.43E+01	1.48E+02	2.04E+02	1.15E+01	1.10E+01	
F24	Mean	9.13E+02	9.19E+02	9.18E+02	1.00E+03	1.70E+03	1.32E+03	9.09E+02	1.10E+03	8.96E+02	6.72E+03	7.40E+03	1.48E+03	9.62E+02	3.64E+03	8.93E+02	<b>3.50E+02</b>	
	Std	1.07E+01	1.93E+01	1.28E+01	1.45E+02	4.01E+01	1.11E+02	7.56E+00	5.28E+01	5.50E+00	7.12E+02	4.77E+02	1.03E+02	2.03E+02	6.70E+02			

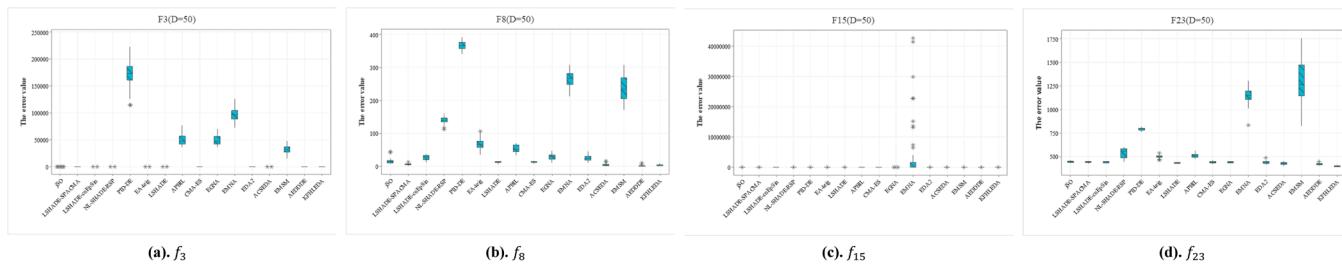


Fig. 6. Box plots of four typical functions (50D).

normality is guaranteed according to the central limit theorem. The individuals are randomly generated and the null hypothesis is rejected based on the Leneve test of SPSS. The validity of variance homogeneity is confirmed.

Table 1 shows that the p-value of  $NP_{max}$  is less than 0.05, and the corresponding F-ratio is the largest. It indicates that  $NP_{max}$  has an important influence within the 95% confidence interval. The  $\tau$  and  $h$  are greater than 0.05, which means there is an interaction between them. The main effects plot and interaction plot in Fig. 5 show that the best and the worst results are obtained when  $NP_{max} = 3000$  and  $NP_{max} = 1000$ . The smaller  $NP_{max}$  is, the worse the exploration is. A large  $NP_{max}$  wastes computing resources. When  $\tau = 0.35$ , the minimum error value is achieved. Hence, the optimal combination of the three parameters is  $NP_{max} = 3000$ ,  $\tau = 0.35$ ,  $h = 200$ . The analysis above is consistent with the results shown in Fig. 5.

#### 4.3. Comparison with the state-of-the-art algorithms

In this study, the KFHLEDA is compared with fifteen classical and state-of-the-art algorithms. The classical model-based evolutionary strategies include APBIL [25], CMA-ES [49], EGNA [29], and EMNA [28]. The novel DE variants include LSHADE [57], jSO [76], PID-DE [77], LSHADE-cnEpSin [78], and NL-SHADE-RSP [55]. The state-of-the-art cooperative algorithms with DE and CMA-ES are EA4eig [58], and LSHADE-SPACMA [79]. The typical EMNA variants include EDA2 [36], ACSEDA [43], EMSM [80], and AEDDDE [59]. The corresponding parameters in comparison algorithms are set to be consistent with those in the original literature and are detailed in Table S1.

Tables S2-S3 (supplementary material) and Tables 2-3 compare the overall performance of KFHLEDA and comparison algorithms through the mean, standard deviation, and computing time on 10D, 30D, 50D, and 100D for 29 functions. The numbers in bold in each line indicate that the value on the corresponding function is minimal. The smaller these values are, the better the performance is. When the mean of two algorithms is the same, the standard deviation reflects the performance.

For the unimodal functions  $f_1$  and  $f_3$ , KFHLEDA, jSO, LSHADE-SPACMA, LSHADE-cnEpSin, and CMA-ES find the optimal solution on 10D, 30D, 50D, and 100D.

For the simple multimodal functions  $f_4 - f_{10}$ , KFHLEDA shows the obvious advantages. When the dimension is 10D, several algorithms are slightly better than the proposed KFHLEDA on  $f_4$ . The KFHLEDA ach-

ieves promising performance on  $f_6 - f_{10}$  compared with most algorithms. Several algorithms obtain similar results, such as LSHADE-SPACMA, LSHADE-cnEpSin, NL-SHADE-RSP, EA4eig, EGNA, ACSEDA, and AEDDDE on  $f_7$ . The EGNA is the best on  $f_5$ , followed by the NL-SHADE-RSP, AEDDDE, and KFHLEDA. When the dimension is 30D, CMA-ES performs best on  $f_4$ . The KFHLEDA obtains excellent results on  $f_6$ ,  $f_7$ , and  $f_9$ . The LSHADE, jSO, LSHADE-SPACMA, EDA2, and ACSEDA also reach the optimal solution on  $f_6$  and  $f_9$ . The AEDDDE is superior to the comparison algorithms on  $f_5$  and  $f_8$ . When the dimension is 50D, LSHADE-SPACMA, NL-SHADE-RSP, and ACSEDA outperform other algorithms on  $f_6$ , followed by LSHADE-cnEpSin, EA4eig, KFHLEDA and APBIL. The proposed KFHLEDA is better than other algorithms on  $f_7$ ,  $f_8$ , and  $f_9$ , while the LSHADE-SPACMA, LSHADE-cnEpSin, LSHADE, EGNA, and ACSEDA also achieve performance similar to the KFHLEDA. When the dimension is 100D, NL-SHADE-RSP is the best on  $f_4$ , and  $f_6$ . LSHADE-SPACMA is the best, followed by the KFHLEDA, which is superior to other algorithms on  $f_5$ , and  $f_{10}$ .

For the hybrid functions  $f_{11} - f_{20}$ , when the dimension is 10D, KFHLEDA achieves the best results compared with other algorithms on  $f_{11}$ , and  $f_{16}$ , and is preferable on  $f_{14}$ ,  $f_{15}$ , and  $f_{18}$ . Some other algorithms also obtain similar results. For example, the performance of jSO, LSHADE-SPACMA, LSHADE-cnEpSin, EA4eig, ACSEDA, and AEDDDE on  $f_{11}$  is equal to that of the proposed KFHLEDA. While EA4eig achieves the optimal solution on  $f_{11}, f_{14}, f_{19}$ , and  $f_{20}$ , the NL-SHADE-RSP also performs excellently on  $f_{14}, f_{19}$ , and  $f_{20}$ . When the dimension is 30D, the jSO is superior to others on  $f_{11}$ . The proposed KFHLEDA obtains superior performance on  $f_{13}, f_{15}, f_{16}, f_{17}$ , and  $f_{19}$ . While the NL-SHADE-RSP performs best on  $f_{18}$ , the LSHADE-cnEpSin, LSHADE, jSO, EA4eig, EMSM, EDA2, ACSEDA, and AEDDDE also achieve similar advantageous results on  $f_{18}$ . The EDA2 is the best on  $f_{14}$  and  $f_{20}$  compared with other algorithms. When the dimension is 50D, the KFHLEDA is superior to others on  $f_{11}, f_{13}, f_{14}, f_{16}, f_{18}, f_{19}$ , and  $f_{20}$ . However, the ACSEDA achieves equivalent results on  $f_{12}$  and  $f_{18}$ . The EDA2 performs best on  $f_{17}$ . When the dimension is 100D, the proposed KFHLEDA has more advantages over other algorithms on  $f_{11}, f_{12}, f_{14}, f_{15}, f_{16}, f_{18}$ , and  $f_{19}$ . While the AEDDDE, ACSEDA, and EDA2 also achieve better performance on  $f_{18}, f_{19}$ , and  $f_{20}$  than other algorithms.

For the composition functions  $f_{21} - f_{30}$ , most algorithms obtain similar results. However, the proposed KFHLEDA exhibits splendid performance on the whole. When the dimension is 10D, the KFHLEDA shows the best performance on  $f_{23}, f_{25}, f_{27}$ , and  $f_{28}$ . The PID-DE and NL-

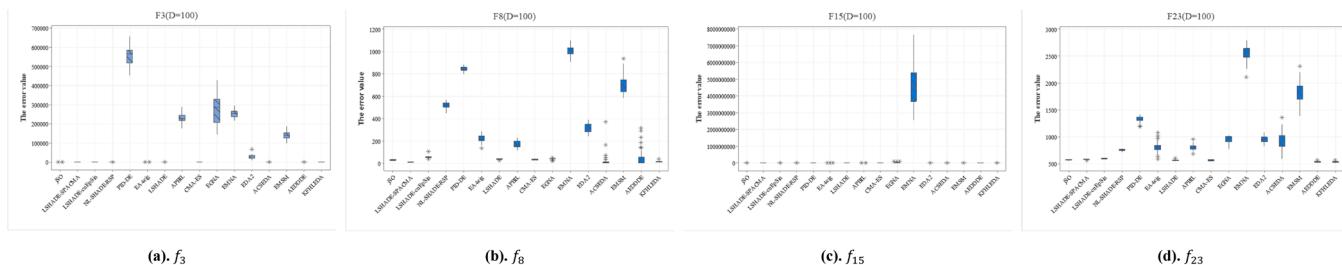


Fig. 7. Box plots of four typical functions (100D).

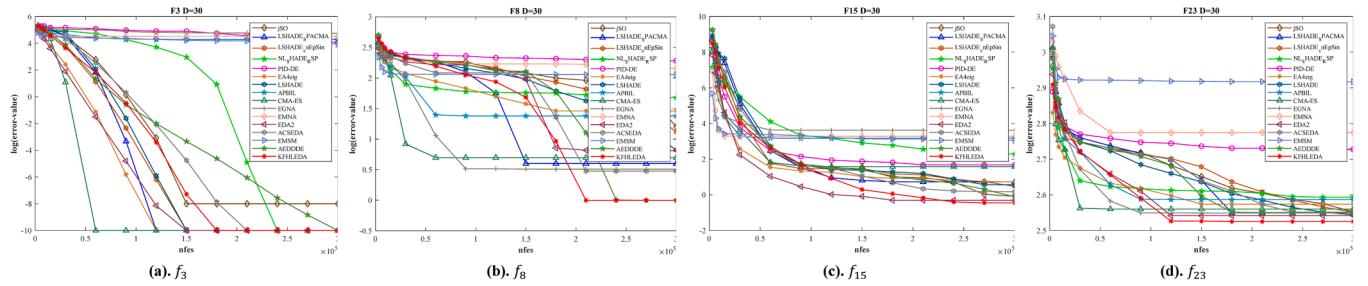


Fig. 8. Convergence curves of four typical functions (30D).

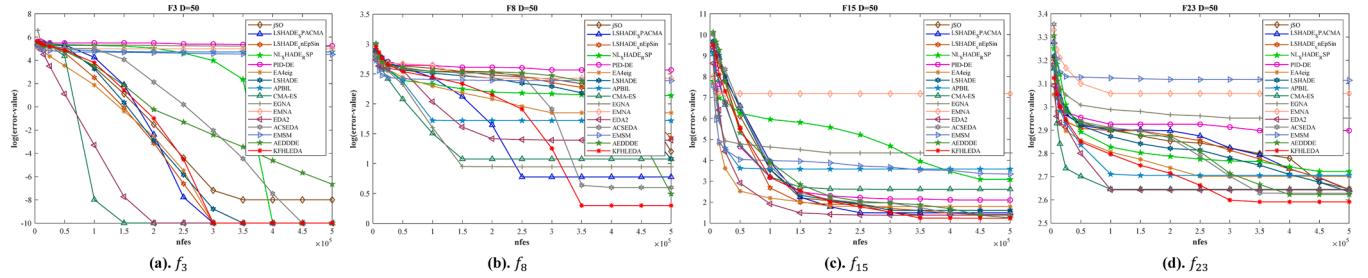


Fig. 9. Convergence curves of four typical functions (50D).

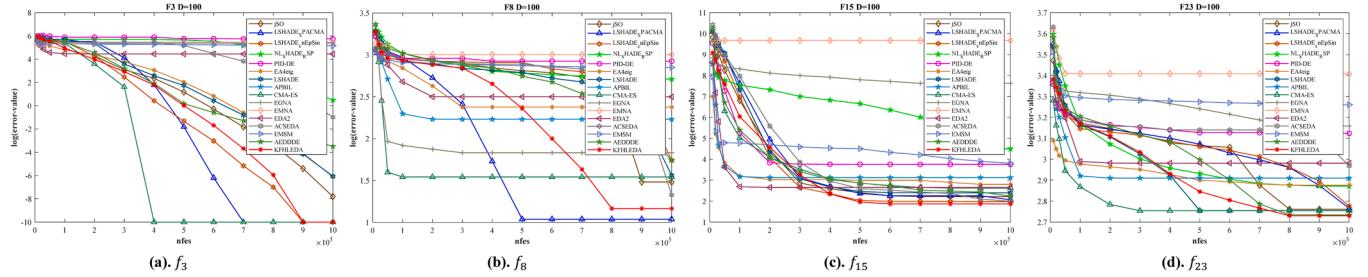


Fig. 10. Convergence curves of four typical functions (100D).

SHADE-RSP are superior to other algorithms on  $f_{22}$ , and  $f_{27}$ , respectively. The LSHADE-cnEpSin is the best on  $f_{21}$ , and  $f_{29}$ . When the dimension is 30D, the NL-SHADE-RSP achieves the optimal solution on  $f_{22}$ , and also shows outstanding performance on  $f_{24}$ ,  $f_{26}$ , and  $f_{29}$ . In addition, KFHLEDA also performs better than the other algorithms on  $f_{21}$ ,  $f_{23}$ , and  $f_{24}$ . The CMA-ES is superior to the other algorithms on  $f_{29}$ . When the dimension is 50D, the proposed KFHLEDA also demonstrates advantages on  $f_{22}$ ,  $f_{23}$ ,  $f_{24}$ , and  $f_{29}$  compared with the other algorithms. EA4eig achieves the best results on  $f_{27}$ , and  $f_{28}$ . When the dimension is 100D, the KFHLEDA outperforms other algorithms on  $f_{21}$ ,

$f_{22}$ ,  $f_{23}$ ,  $f_{24}$ ,  $f_{26}$ , and  $f_{29}$ .

The computing time of the sixteen algorithms is considered. The average computing time in milliseconds on 10D, 30D, 50D, and 100D is shown in Tables S2-S3 and Tables 2-3. According to the experimental results, the computing time of KFHLEDA is shorter than most algorithms. Especially, the computing time of ENGA is overlong because of the construction of the complex probability graph model through the finite sample set in the coupling relation of continuous variable space. The results verify that the learning of EGNA is of higher cost as the problem scale becomes larger.

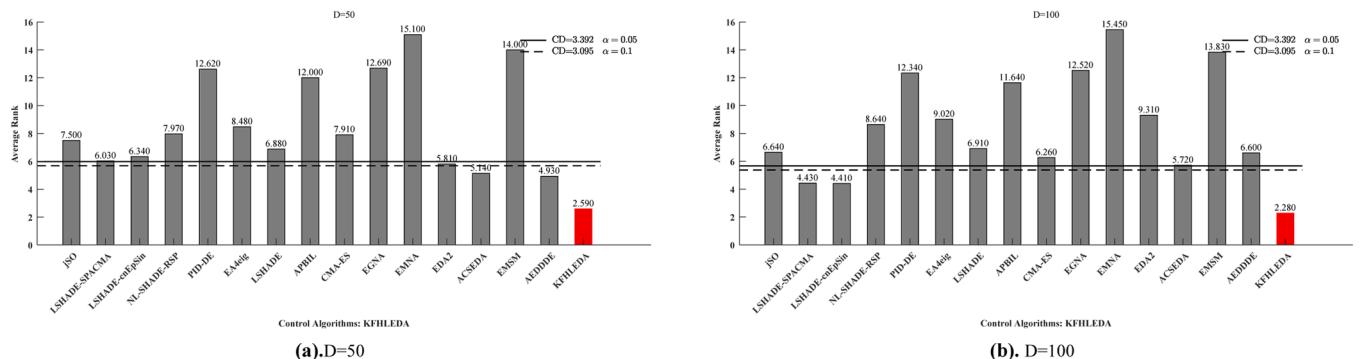


Fig. 11. Rankings obtained through the Friedman test.

**Table 4**

The mean rank of the algorithms (100 D).

Algorithms	Mean rank (Friedman test)	Mean rank (Friedman Aligned rank test)	Mean rank (Quade test)
jSO	6.67	212.88	6.35
LSHADE-SPACMA	4.48	185.07	4.63
LSHADE-cnEpSin	4.47	184.43	4.03
NL-SHADE-RSP	8.64	228.84	8.52
PID-DE	12.34	260.55	12.05
EA4eig	8.95	212.64	8.81
LSHADE	6.81	191.43	6.90
APBIL	11.64	255.33	12.56
CMA-ES	6.29	196.26	6.42
EGNA	12.52	301.97	13.17
EMNA	15.45	429.72	15.49
EDA2	9.31	196.48	9.36
ACSEDA	5.69	182.97	5.43
EMSM	13.83	305.21	13.71
AEDDDE	6.60	200.91	6.40
KFHLEDA	2.31	175.31	2.17

The stability of the proposed algorithm is demonstrated through box plots on all 29 functions. The experimental results are listed in Figs. S1-S6 (supplementary material) and Figs. 6-7 where the horizontal axis represents all the sixteen algorithms, and the vertical axis is the standard variance of the error values between the candidate and the optimal solutions. In Figs. S3-S4 and Figs. 6-7, one function is selected to represent each of the four function types, namely  $f_3$ ,  $f_8$ ,  $f_{15}$ , and  $f_{23}$ . Compared with other algorithms, the standard variance of KFHLEDA is less than that of most comparison algorithms. Therefore, the proposed KFHLEDA has the best overall stability and reveals the robustness in continuous real-valued optimization problems.

The convergence curves declare the convergence degree of all sixteen algorithms in Figs. S7-S11 (supplementary material) and Figs. 8-10, where the horizontal axis is the number of function evaluations, and the vertical axis is the logarithm of the error value. The red line represents the proposed KFHLEDA with the best convergence precision on most functions. The CEC 2017 test suit specifies that a convergence curve is drawn with specified 14 points. The results reveal that some algorithms converge quickly but with lower accuracy. The reason is that these algorithms fail to find the global optimal solution due to the limitation of the algorithm performance design. From the experimental data, some algorithms have stalled prematurely in the later iteration. The overall performance of the KFHLEDA is outstanding in most cases. While the proposed algorithm performs well on most functions, it shows poor performance on some others.

#### 4.4. The Friedman, Friedman Aligned rank, and Quade tests

The Friedman Test is a rank sum test between multiple samples that compares algorithms for significant differences. The test is employed to evaluate the performance of sixteen algorithms to obtain a statistical conclusion in this section. The critical difference ( $CD$ ) as a crucial indicator is calculated by  $CD = q_a \sqrt{\frac{k(k+1)}{6 \times fun}}$  where  $k$  is the total number of algorithms,  $fun$  is the number of functions, and  $q_a$  is the critical value. The effects of all algorithms are indicated in Fig. S12 (supplementary material) and Fig. 11 where the horizontal axis represents all algorithms, and the vertical axis represents an average rank. The solid and dotted lines are the CDs in the 90% and 95% confidence intervals, respectively. The bar of the proposed KFHLEDA is highlighted.

The ranking results of different dimensions reveal that KFHLEDA has the best ranking. The average rank of KFHLEDA is under the critical differences. There are significant differences between KFHLEDA and most algorithms except for jSO, LSHADE-SPACMA, LSHADE-cnEpSin, NL-SHADE-RSP, EA4eig, and LSHADE on 10D, EDA2, and AEDDDE on 30D, ACSEDA and AEDDDE on 50D, LSHADE-SPACMA and LSHADE-cnEpSin on 100D in 90% and 95% confidence intervals.

Friedman Aligned ranks test and Quade test are employed to further demonstrate the performance of algorithms. These two tests are methods of performing multiple comparisons. The experimental results between the proposed KFHLEDA and other comparison algorithms on 10D, 30D, 50D, and 100D are displayed in Tables S6-S8 (supplementary material) and Table 4. The results of the Friedman, Friedman Aligned, and Quade tests illustrate that the proposed KFHLEDA has a rank of 5.00, 193.24, and 5.62 on 10D, 3.22, 176.88, and 3.41 on 30D, 2.64, 176.22, 2.85 on 50D, 13.83, 305.21, 13.71 on 100D, respectively. From the experimental results, a significant difference exists between the KFHLEDA and most algorithms. The KFHLEDA is a superior algorithm.

#### 4.5. Post-hoc test

The statistical results of the Friedman test show whether there are significant differences among  $k$  algorithms. If differences exist, a post-hoc test is performed further, which reveals statistically detailed differences between algorithms. Common post-hoc test approaches include Nemenyi, Bonferroni-Dunn, Holm, Hochberg, Hommel, Holland, and Rom tests. The test principle is to compare the difference in average ranking between algorithms with  $CD$  to determine the significant difference.

The Bonferroni-Dunn, Holm, Hochberg, Hommel, Holland, and Rom methods are introduced to calculate a set of adjusted  $p$ -values for the experimental hypotheses. The KFHLEDA is taken as the control algorithm and compared with the remaining  $k - 1$  algorithms. The  $p$ -value is acquired from the Friedman test. When the values are less than  $10^{-6}$ ,

**Table 5**

Results of post-hoc test with KFHLEDA as control algorithm (100 D).

Algorithm	z-value	p-value	Bonferroni	Holm	Hochberg	Hommel	Holland	Rom
EMNA	10.51E+00	<b>0.00E+00</b>						
EMSM	9.21E+00	<b>0.00E+00</b>						
EGNA	8.16E+00	<b>0.00E+00</b>						
PID-DE	8.03E+00	<b>0.00E+00</b>						
APBIL	7.46E+00	<b>0.00E+00</b>						
EDA2	5.60E+00	<b>0.00E+00</b>						
EA4eig	5.31E+00	<b>0.00E+00</b>	<b>2.00E-06</b>	<b>1.00E-06</b>	<b>1.00E-06</b>	<b>1.00E-06</b>	<b>1.00E-06</b>	<b>1.00E-06</b>
NL-SHADE-RSP	5.06E+00	<b>0.00E+00</b>	<b>6.00E-06</b>	<b>3.00E-06</b>	<b>3.00E-06</b>	<b>3.00E-06</b>	<b>3.00E-06</b>	<b>3.00E-06</b>
LSHADE	3.60E+00	<b>3.19E-04</b>	<b>4.79E-03</b>	<b>2.24E-03</b>	<b>2.24E-03</b>	<b>1.79E-03</b>	<b>2.23E-03</b>	<b>2.13E-03</b>
jSO	3.49E+00	<b>4.85E-04</b>	<b>7.28E-03</b>	<b>2.91E-03</b>	<b>2.91E-03</b>	<b>2.43E-03</b>	<b>2.91E-03</b>	<b>2.77E-03</b>
AEDDDE	3.43E+00	<b>5.95E-04</b>	<b>8.93E-03</b>	<b>2.98E-03</b>	<b>2.98E-03</b>	<b>2.98E-03</b>	<b>2.97E-03</b>	<b>2.83E-03</b>
CMA-ES	3.19E+00	<b>1.45E-03</b>	<b>2.17E-02</b>	<b>5.78E-03</b>	<b>5.78E-03</b>	<b>5.78E-03</b>	<b>5.77E-03</b>	<b>5.51E-03</b>
ACSEDA	2.70E+00	<b>6.88E-03</b>	<b>1.03E-01</b>	<b>2.06E-02</b>	<b>2.06E-02</b>	<b>2.06E-02</b>	<b>2.05E-02</b>	<b>2.06E-02</b>
LSHADE-SPACMA	1.74E+00	8.23E-02	1.23E+00	1.65E-01	8.48E-02	8.48E-02	1.58E-01	8.48E-02
LSHADE-cnEpSin	1.72E+00	8.48E-02	1.27E+00	1.65E-01	8.48E-02	8.48E-02	1.58E-01	8.48E-02

**Table 6**

Results of 16 vs 16 comparisons on the benchmark (50D).

Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi
EMNA vs. KFHLEDA	9.97E+00	0.00E+00	0.00E+00	LSHADE- SPACMA vs. EGNA	5.20E+00	0.00E+00	2.40E- 05	LSHADE- cnEpSin vs. KFHLEDA	2.95E+00	3.17E- 03	3.80E-01
EMSM vs. KFHLEDA	9.09E+00	0.00E+00	0.00E+00	jSO vs. EMSM	5.19E+00	0.00E+00	2.60E- 05	EA4eig vs. APBIL	2.85E+00	4.31E- 03	5.17 E-01
EMNA vs. AEDDDE	8.37E+00	0.00E+00	0.00E+00	LSHADE- cnEpSin vs. EGNA	5.09E+00	0.00E+00	4.30E- 05	LSHADE- SPACMA vs. KFHLEDA	2.79E+00	5.34E- 03	6.41E-01
EGNA vs. KFHLEDA	8.04E+00	0.00E+00	0.00E+00	LSHADE- cnEpSin vs. PID-DE	5.03E+00	0.00E+00	5.80E- 05	NL-SHADE-RSP vs. AEDDDE	2.72E+00	6.60E- 03	7.91E-01
PID-DE vs. KFHLEDA	7.98E+00	0.00E+00	0.00E+00	APBIL vs. EDA2	4.90E+00	1.00E-06	1.10E- 04	CMA-ES vs. AEDDDE	2.63E+00	8.44E- 03	1.01E+00
EMNA vs. ACSEDA	7.90E+00	0.00E+00	0.00E+00	CMA-ES vs. EMSM	4.85E+00	1.00E-06	1.45E- 04	EDA2 vs. KFHLEDA	2.56E+00	1.03E- 02	1.24E+00
EMSM vs. AEDDDE	7.49E+00	0.00E+00	0.00E+00	NL-SHADE- RSP vs. EMSM	4.77E+00	2.00E-06	2.20E- 04	EA4eig vs. ACSEDA	2.55E+00	1.07E- 02	1.29E+00
APBIL vs. KFHLEDA	7.47E+00	0.00E+00	0.00E+00	LSHADE- SPACMA vs. APBIL	4.69E+00	3.00E-06	3.30E- 04	APBIL vs. EMNA	2.50E+00	1.26E- 02	1.51E+00
EMNA vs. EDA2	7.41E+00	0.00E+00	0.00E+00	EA4eig vs. KFHLEDA	4.62E+00	4.00E-06	4.61E- 04	jSO vs. AEDDDE	2.30E+00	2.13E- 02	2.55E+00
LSHADE- SPACMA vs. EMNA	7.18E+00	0.00E+00	0.00E+00	LSHADE vs. EGNA	4.62E+00	4.00E-06	4.61E- 04	NL-SHADE-RSP vs. ACSEDA	2.25E+00	2.46E- 02	2.95E+00
ACSEDA vs. EMSM	7.02E+00	0.00E+00	0.00E+00	PID-DE vs. LSHADE	4.56E+00	5.00E-06	6.01E- 04	CMA-ES vs. ACSEDA	2.17E+00	3.04E- 02	3.65E+00
LSHADE- cnEpSin vs. EMNA	7.02E+00	0.00E+00	0.00E+00	LSHADE- cnEpSin vs. APBIL	4.52E+00	6.00E-06	7.31E- 04	ACSEDA vs. KFHLEDA	2.07E+00	3.86E- 02	4.63E+00
LSHADE vs. EMNA	6.55E+00	0.00E+00	0.00E+00	EA4eig vs. EMSM	4.47E+00	8.00E-06	9.48E- 04	EA4eig vs. EDA2	2.05E+00	3.99E- 02	4.79E+00
EDA2 vs. EMSM	6.52E+00	0.00E+00	0.00E+00	NL-SHADE- RSP vs. KFHLEDA	4.32E+00	1.60E-05	1.90E- 03	PID-DE vs. EMNA	1.99E+00	4.71E- 02	5.65E+00
EGNA vs. AEDDDE	6.44E+00	0.00E+00	0.00E+00	CMA-ES vs. KFHLEDA	4.23E+00	2.30E-05	2.76E- 03	EGNA vs. EMNA	1.93E+00	5.35E- 02	6.42E+00
PID-DE vs. AEDDDE	6.38E+00	0.00E+00	0.00E+00	jSO vs. EGNA	4.14E+00	3.50E-05	4.22E- 03	LSHADE- SPACMA vs. EA4eig	1.83E+00	6.66E- 02	8.00E+00
LSHADE- SPACMA vs. EMSM	6.30E+00	0.00E+00	0.00E+00	jSO vs. PID-DE	4.08E+00	4.50E-05	5.36E- 03	jSO vs. ACSEDA	1.83E+00	6.66E- 02	8.00E+00
LSHADE- cnEpSin vs. EMSM	6.14E+00	0.00E+00	0.00E+00	LSHADE vs. APBIL	4.05E+00	5.00E-05	6.04E- 03	LSHADE vs. AEDDDE	1.82E+00	6.87E- 02	8.25E+00
jSO vs. EMNA	6.07E+00	0.00E+00	0.00E+00	jSO vs. KFHLEDA	3.90E+00	9.50E-05	1.14E- 02	NL-SHADE-RSP vs. EDA2	1.75E+00	7.99E- 02	9.59E+00
EGNA vs. ACSEDA	5.97E+00	0.00E+00	0.00E+00	CMA-ES vs. EGNA	3.81E+00	1.41E-04	1.69E- 02	LSHADE- cnEpSin vs. EA4eig	1.67E+00	9.52E- 02	1.14E+01
PID-DE vs. ACSEDA	5.92E+00	0.00E+00	0.00E+00	PID-DE vs. CMA-ES	3.75E+00	1.76E-04	2.11E- 02	CMA-ES vs. EDA2	1.67E+00	9.52E- 02	1.14E+01
APBIL vs. AEDDDE	5.87E+00	0.00E+00	1.00E-06	NL-SHADE- RSP vs. EGNA	3.72E+00	1.97E-04	2.36E- 02	APBIL vs. EMSM	1.61E+00	1.07E- 01	1.28E+01
CMA-ES vs. EMNA	5.74E+00	0.00E+00	1.00E-06	NL-SHADE- RSP vs. PID-DE	3.67E+00	2.44E-04	2.93E- 02	AEDDDE vs. KFHLEDA	1.60E+00	1.10E- 01	1.32E+01
LSHADE vs. EMSM	5.67E+00	0.00E+00	2.00E-06	jSO vs. APBIL	3.57E+00	3.55E-04	4.26 E- 02	LSHADE- SPACMA vs. NL- SHADE-RSP	1.53E+00	1.26E- 01	1.51E+01
NL-SHADE- RSP vs. EMNA	5.65E+00	0.00E+00	2.00E-06	LSHADE vs. KFHLEDA	3.42E+00	6.26E-04	7.52E-02	LSHADE- SPACMA vs. CMA-ES	1.45E+00	1.48E- 01	1.77E+01
EGNA vs. EDA2	5.47E+00	0.00E+00	5.00E-06	EA4eig vs. EGNA	3.42E+00	6.26E-04	7.52E-02	LSHADE- cnEpSin vs. NL- SHADE-RSP	17.3E+00	1.72E- 01	2.07E+01
PID-DE vs. EDA2	5.42E+00	0.00E+00	7.00E-06	PID-DE vs. EA4eig	3.36E+00	7.66E-04	9.19E-02	LSHADE- cnEpSin vs. AEDDDE	1.35E+00	1.77E- 01	2.12E+01
APBIL vs. ACSEDA	5.41E+00	0.00E+00	8.00E-06	APBIL vs. CMA-ES	3.24E+00	1.19E-03	1.43E-01	LSHADE vs. ACSEDA	1.35E+00	1.77E- 01	2.12E+01
EA4eig vs. EMNA	5.35E+00	0.00E+00	1.10E-05	NL-SHADE- RSP vs. APBIL	3.16E+00	1.59E-03	1.91E-01	jSO vs. EDA2	1.34E+00	1.81E- 01	2.17E+01

(continued on next page)

**Table 6 (continued)**

Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi
LSHADE- SPACMA vs. EGNA	5.25E+00	0.00E+00	1.80E-05	EA4eig vs. AEDDDE	3.02E+00	2.53E-03	3.03E-01				

they are set as 0. The test results are listed in Tables S9-S11 (supplementary material) and **Table 5**. The *p*-values and the adjusted *p*-values present differences between the comparison algorithms on 10D, 30D, 50D, and 100D in most situations.

The Nemenyi test compares the performance of *k* algorithms one-to-one to further statistically analyze the test results. All the comparison pairs are performed without a control algorithm. A  $16 \times 16$  comparison at 10D, 30D, 50D, and 100D are displayed in Tables S4-S5 (supplementary material) and **Tables 6-7**, respectively. The proposed KFHLEDA is superior to most comparison algorithms, which further shows its effectiveness.

These results demonstrate that the KFHLEDA outperforms most algorithms, which verifies that the enhanced Kalman filtering mechanism, history learning mechanism, and population adaptive adjustment strategy effectively improve the performance of the proposed algorithm. Kalman filtering feeds the enhanced information back to the EDA model through the history learning mechanism. The population adaptive adjustment strategy improves the quality of solutions and effectively balances exploration and exploitation. According to the No Free Lunch Theory [81], there is no single algorithm with optimal performance on all problems.

Compared to these classical and state-of-the-art algorithms, the proposed KFHLEDA demonstrates competitive performance due to its improved design based on problem characteristics, which include the enhanced Kalman filtering operation, history learning mechanism, and population adaptive adjustment strategy. The following experiments verify this point by components availability analysis in **Section 4.6**.

#### 4.6. Component availability analysis

The proposed KFHLEDA covers three important strategies: the Kalman filtering, the history learning mechanism, and the population adaptive adjustment strategy. This section evaluates the effectiveness of the three strategies.

Three variants of the KFHLEDA are adopted and tested against it on the CEC 2017 test suit to demonstrate the effectiveness of each strategy. The three variants are denoted as KFHLEDA1, KFHLEDA2, and KFHLEDA3, each representing KFHLEDA without the enhanced Kalman filtering, KFHLEDA without the history learning mechanism, and KFHLEDA without the population adaptive adjustment strategy, respectively. The three variants are detailed below.

- (1) **KFHLEDA1:** It excludes the application of enhanced Kalman filtering while preserving other operations of the KFHLEDA. Comparison with this variant demonstrates the effect of the enhanced Kalman filtering. The mean and variance of the probabilistic model are updated according to [Eqs. \(4\)-\(5\)](#), instead of [Eqs. \(12\)-\(13\)](#) based on the filtering operation, that is, the mean and variance are guided by the dominant information in the historical archive.
- (2) **KFHLEDA2:** This variant omits the history learning mechanism while retaining the other operations of the proposed KFHLEDA. It implies that the modification of the individuals by the enhanced Kalman filtering operation is based on the information obtained in the last operation rather than on the historical archive.
- (3) **KFHLEDA3:** This variant removes the population adaptive adjustment strategy while retaining the other operations of the KFHLEDA. Therefore, the population size is no longer updated

adaptively with each iteration but instead, set to a fixed value of 3000.

**Table 8** shows that the overall performance of the proposed KFHLEDA is optimal compared with the three variants. The KFHLEDA1 obtains worse results than KFHLEDA2 and KFHLEDA3, all of which are worse than the KFHLEDA. Among the three variants, the KFHLEDA1 performs the worst due to the absence of the Kalman filtering mechanism, while the KFHLEDA3 achieves the best results by utilizing Kalman filtering and the historical learning mechanism. These experimental results show the effectiveness and significance of the selected strategies for KFHLEDA. The mean values in **Table 8** are visualized in **Fig. 12**.

The population adaptive adjustment strategy contributes to exploration of the proposed KFHLEDA in the early stage to avoid premature convergence, and exploitation in the late stage to avoid ineffective searches and accelerate the evolution process. The historical learning mechanism helps the population explore the hidden information in the archive during the evolutionary process, and make it evolve towards the potential region. In the stages of the prediction, observation, and two revision operations during the enhanced Kalman filtering, the elite strategy is employed to guide the population to jump out of the local optimum.

#### 4.7. Analysis of the fitness landscape

The fitness landscape reproduces the evolution of the population in the search space and intuitively responds to the specific phenomena with the progress of iteration [82]. The fitness landscape is highly effective in analyzing the optimization effect of the algorithms on the functions in the test suit, especially for the complex, diverse, multi-modal, funnel-shaped features. The distribution of solutions is visualized. In view of the hybrid functions without a two-dimensional definition, the fitness landscape of seventeen functions is listed in **Figs. 13-15** and **Figs. S13-S15** (Supplementary materials), where the red dots represent the individuals involved in the evolution. In the unimodal functions  $f_1$  and  $f_3$ , the optimal position is found at express speed.  $f_4 - f_{10}$  are the multimodal functions with the rugged landscape and many local optimal solutions. The global optimal position is difficult to be discovered, particularly for composition functions  $f_{21} - f_{28}$  with varying basin shapes, making them difficult to optimize.

**Fig. 13** shows that the proposed KFHLEDA evolves rapidly on  $f_3$ , and finds the optimal solution, attributed to the rich population diversity and outstanding local search ability. While the CMA-ES performs poorly on asymmetrical function  $f_9$  with a huge number of local optima, the proposed KFHLEDA finds the optimal solution quickly as shown in **Fig. 14**. For the composition functions, such as  $f_{21}, f_{24}$ , and  $f_{26}$  with many peaks and valleys in **Fig. 15**, the proposed KFHLEDA performs well due to its splendid exploration and exploitation.

#### 5. Conclusion and future work

In this paper, a novel EDA named KFHLEDA that incorporates an enhanced Kalman filtering and historical learning mechanism and population adaptive adjustment strategy is proposed to solve continuous optimization problems. The proposed KFHLEDA enhances the conventional Kalman filtering operation. The prediction, observation, first revision, and second revision operations for the enhanced Kalman filtering are embedded into the EDA framework to address problem-

**Table 7**

Results of 16 vs 16 comparisons on the benchmark (100D).

Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi
EMNA vs. KFHLEDA	10.51E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	EA4eig vs. EMNA	5.20E+00	<b>0.00E+00</b>	<b>2.40E-05</b>	APBIL vs. EMNA	3.05E+00	<b>2.30E-03</b>	2.77E-01
EMSM vs. KFHLEDA	9.21E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	NL-SHADE-RSP vs. KFHLEDA	5.06E+00	<b>0.00E+00</b>	<b>5.00E-05</b>	NL-SHADE- RSP vs. PID-DE	2.96E+00	<b>3.02E-03</b>	3.63E-01
LSHADE- cnEpSin vs. EMNA	8.78E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	CMA-ES vs. EGNA	4.98E+00	<b>1.00E-06</b>	<b>7.70E-05</b>	EDA2 vs. ACSEDA	2.90E+00	<b>3.78E-03</b>	4.54E-01
LSHADE- SPACMA vs. EMNA	8.77E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	EMNA vs. EDA2	4.91E+00	<b>1.00E-06</b>	<b>1.10E-04</b>	EA4eig vs. EGNA	2.85E+00	<b>4.31E-03</b>	5.17E-01
EGNA vs. KFHLEDA	8.16E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	PID-DE vs. CMA-ES	4.84E+00	<b>1.00E-06</b>	<b>1.56E-04</b>	PID-DE vs. EA4eig	2.72E+00	<b>6.60E-03</b>	7.91E-01
PID-DE vs. KFHLEDA	8.03E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	APBIL vs. ACSEDA	4.76E+00	<b>2.00E-06</b>	<b>2.35E-04</b>	ACSEDA vs. KFHLEDA	2.70E+00	<b>6.88E-03</b>	8.25E-01
EMNA vs. ACSEDA	7.81E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	EGNA vs. AEDDDE	4.73E+00	<b>2.00E-06</b>	<b>2.69E-04</b>	EA4eig vs. ACSEDA	2.61E+00	<b>9.15E-03</b>	1.10E+00
LSHADE- cnEpSin vs. EMSM	7.49E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	jSO vs. EGNA	4.67E+00	<b>3.00E-06</b>	<b>3.53E-04</b>	EGNA vs. EDA2	2.56E+00	<b>1.03E-02</b>	1.24E+00
LSHADE- SPACMA vs. EMSM	7.47E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	PID-DE vs. AEDDDE	4.59E+00	<b>4.00E-06</b>	<b>5.27E-04</b>	PID-DE vs. EMNA	2.48E+00	<b>1.31E-02</b>	1.57E+00
APBIL vs. KFHLEDA	7.46E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	LSHADE vs. EGNA	4.56E+00	<b>5.00E-06</b>	<b>6.01E-04</b>	PID-DE vs. EDA2	2.43E+00	<b>1.52E-02</b>	1.83E+00
CMA-ES vs. EMNA	7.32E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	jSO vs. PID-DE	4.54E+00	<b>6.00E-06</b>	<b>6.85E-04</b>	CMA-ES vs. EDA2	2.41E+00	<b>1.58E-02</b>	1.90E+00
EMNA vs. AEDDDE	7.07E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	PID-DE vs. LSHADE	4.43E+00	<b>1.00E-05</b>	<b>1.15E-03</b>	NL-SHADE- RSP vs. APBIL	2.40E+00	<b>1.64E-02</b>	1.97E+00
jSO vs. EMNA	7.02E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	APBIL vs. CMA- ES	4.27E+00	<b>1.90E-05</b>	<b>2.30E-03</b>	NL-SHADE- RSP vs. ACSEDA	2.36E+00	<b>1.84E-02</b>	2.20E+00
LSHADE vs. EMNA	6.91E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	NL-SHADE-RSP vs. EMSM	4.15E+00	<b>3.30E-05</b>	<b>3.98E-03</b>	EGNA vs. EMNA	2.34E+00	<b>1.91E-02</b>	2.29E+00
ACSEDA vs. EMSM	6.51E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	APBIL vs. AEDDDE	4.03E+00	<b>5.70E-05</b>	<b>6.79E-03</b>	EDA2 vs. AEDDDE	2.17E+00	<b>3.04E-02</b>	3.65E+00
LSHADE- cnEpSin vs. EGNA	6.44E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	jSO vs. APBIL	3.97E+00	<b>7.10E-05</b>	<b>8.57E-03</b>	EA4eig vs. APBIL	2.15E+00	<b>3.15E-02</b>	3.77E+00
LSHADE- SPACMA vs. EGNA	6.43E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	EA4eig vs. EMSM	3.90E+00	<b>9.50E-05</b>	<b>1.14E-02</b>	EA4eig vs. CMA-ES	2.12E+00	<b>3.37E-02</b>	4.04E+00
LSHADE- cnEpSin vs. PID-DE	6.30E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	LSHADE- cnEpSin vs. EDA2	3.87E+00	<b>1.07E-04</b>	<b>1.28E-02</b>	jSO vs. EDA2	2.11E+00	<b>3.49E-02</b>	4.18E+00
LSHADE- SPACMA vs. PID-DE	6.29E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	LSHADE- SPACMA vs. EDA2	3.86E+00	<b>1.13E-04</b>	<b>1.35E-02</b>	LSHADE vs. EDA2	2.00E+00	<b>4.56E-02</b>	5.47E+00
CMA-ES vs. EMSM	6.03E+00	<b>0.00E+00</b>	<b>0.00E+00</b>	LSHADE vs. APBIL	3.86E+00	<b>1.13E-04</b>	<b>1.35E-02</b>	LSHADE- cnEpSin vs. LSHADE	1.88E+00	<b>6.07E-02</b>	7.29E+00
EMSM vs. AEDDDE	5.78E+00	<b>0.00E+00</b>	<b>1.00E-06</b>	EDA2 vs. EMSM	3.61E+00	<b>3.03E-04</b>	<b>3.63E-02</b>	NL-SHADE- RSP vs. CMA- ES	1.88E+00	<b>6.07E-02</b>	7.29E+00
LSHADE- cnEpSin vs. APBIL	5.74E+00	<b>0.00E+00</b>	<b>1.00E-06</b>	LSHADE vs. KFHLEDA	3.60E+00	<b>3.19E-04</b>	<b>3.83E-02</b>	EA4eig vs. AEDDDE	1.88E+00	<b>6.07E-02</b>	7.29E+00
LSHADE- SPACMA vs. APBIL	5.72E+00	<b>0.00E+00</b>	<b>1.00E-06</b>	LSHADE- cnEpSin vs. EA4eig	3.59E+00	<b>3.37E-04</b>	<b>4.04E-02</b>	LSHADE- SPACMA vs. LSHADE	1.86E+00	<b>6.27E-02</b>	7.52E+00
jSO vs. EMSM	5.72E+00	<b>0.00E+00</b>	<b>1.00E-06</b>	LSHADE- SPACMA vs. EA4eig	3.57E+00	<b>3.55E-04</b>	<b>4.26E-02</b>	APBIL vs. EDA2	1.86E+00	<b>6.27E-02</b>	7.52E+00
LSHADE vs. EMSM	5.61E+00	<b>0.00E+00</b>	<b>2.00E-06</b>	jSO vs. KFHLEDA	3.49E+00	<b>4.85E-04</b>	<b>5.82E-02</b>	jSO vs. EA4eig	1.82E+00	<b>6.87E-02</b>	8.25E+00
EDA2 vs. KFHLEDA	5.60E+00	<b>0.00E+00</b>	<b>3.00E-06</b>	AEDDDE vs. KFHLEDA	3.43E+00	<b>5.95E-04</b>	<b>7.15E-02</b>	jSO vs. LSHADE- cnEpSin	1.77E+00	<b>7.75E-02</b>	9.31E+00
EGNA vs. ACSEDA	5.46E+00	<b>0.00E+00</b>	<b>6.00E-06</b>	LSHADE- cnEpSin vs. NL- SHADE-RSP	3.34E+00	<b>8.46E-04</b>	<b>1.02E-01</b>	jSO vs. LSHADE- SPACMA	1.75E+00	<b>7.99E-02</b>	9.59E+00
NL-SHADE- RSP vs. EMNA	5.45E+00	<b>0.00E+00</b>	<b>6.00E-06</b>	LSHADE- SPACMA vs. NL-SHADE-RSP	3.32E+00	<b>8.89E-04</b>	<b>1.07E-01</b>	APBIL vs. EMSM	1.75E+00	<b>7.99E-02</b>	9.59E+00

(continued on next page)

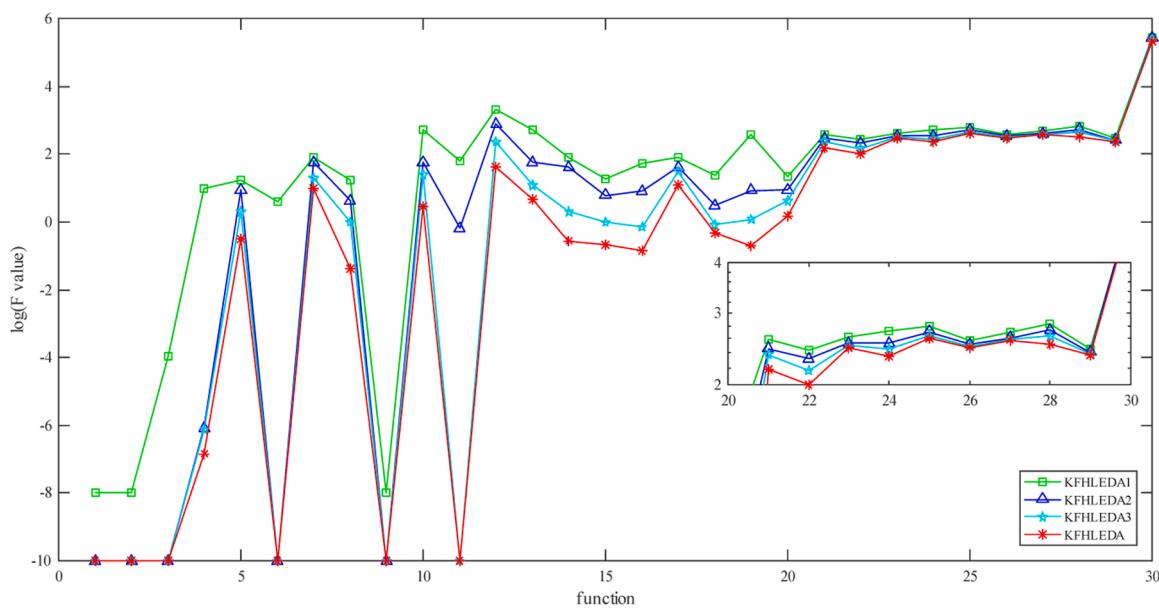
**Table 7 (continued)**

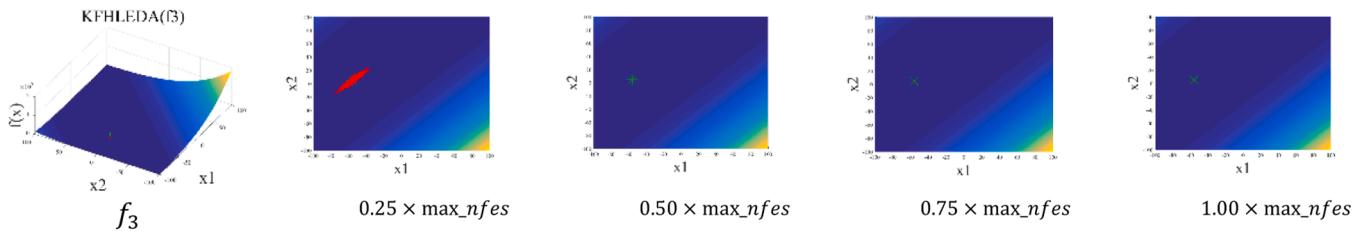
Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi	Algorithm1 VS Algorithm2	z-value	p-value	Nemenyi
PID-DE vs. ACSEDA	5.32E+00	<b>0.00E+00</b>	<b>1.20E-05</b>	CMA-ES vs. KFHLEDA	3.19E+00	<b>1.45E-03</b>	1.73E-01	LSHADE- SPACMA vs. KFHLEDA	1.74E+00	8.23E-02	9.88E+00
EA4eig vs. KFHLEDA	5.31E+00	<b>0.00E+00</b>	<b>1.30E-05</b>	NL-SHADE-RSP vs. EGNA	3.10E+00	<b>1.92E-03</b>	2.30E-01				

**Table 8**

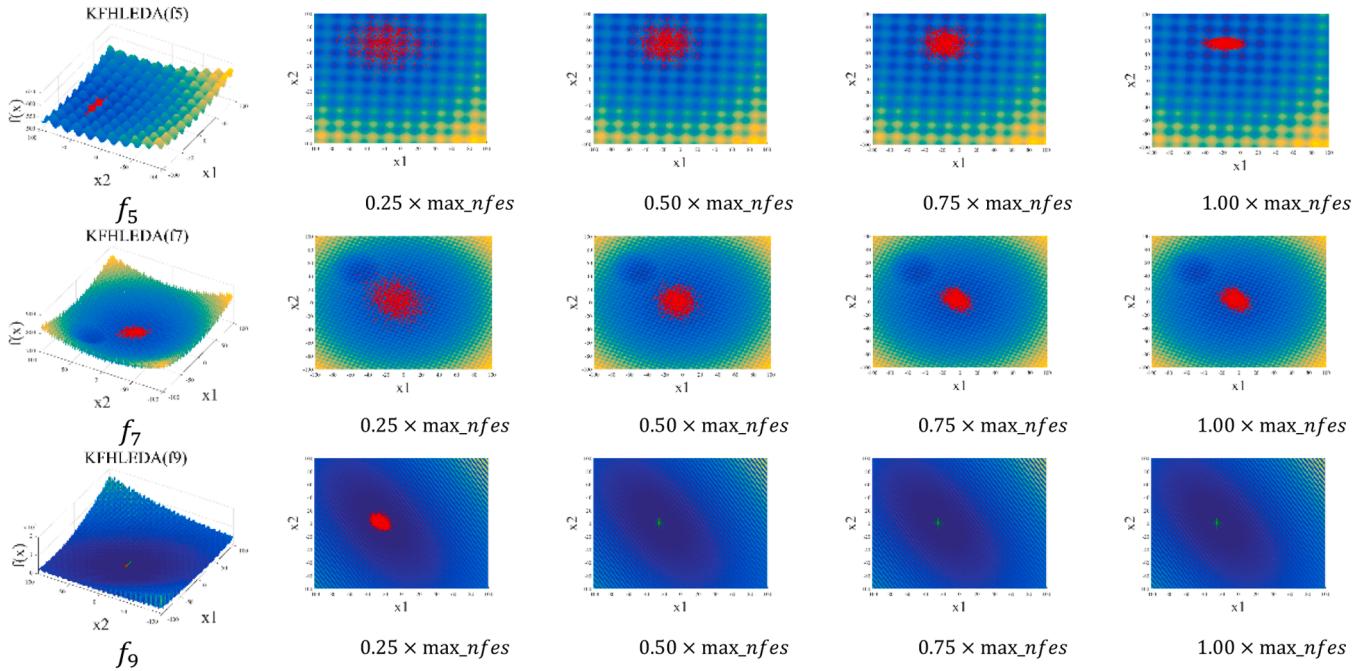
Comparison of results on 29 functions.

Fun	KFHLEDA mean	std	KFHLEDA1 mean	std	KFHLEDA2 mean	std	KFHLEDA3 mean	std
1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
3	0.00E+00	0.00E+00	1.14E-04	4.73E-05	0.00E+00	0.00E+00	0.00E+00	0.00E+00
4	1.44E-07	1.03E-06	9.69E+00	1.14E-02	8.47E-07	1.77E-07	7.35E-07	5.15E-06
5	3.12E-01	5.45E-01	1.72E+01	3.58E+00	8.62E+00	1.34E+00	1.99E+00	1.12E+00
6	0.00E+00	0.00E+00	3.82E+00	1.77E-02	0.00E+00	0.00E+00	0.00E+00	0.00E+00
7	1.01E+01	3.42E-01	8.25E+01	1.75E+01	5.62E+01	3.07E+00	1.99E+01	1.04E+00
8	4.15E-02	2.99E-01	1.75E+01	2.66E+00	4.19E+00	5.20E-01	9.95E-01	3.56E-01
9	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
10	2.90E+00	4.70E+00	5.12E+02	2.30E+02	5.72E+01	3.36E+01	2.51E+01	1.25E+01
11	0.00E+00	0.00E+00	6.20E+01	1.77E-01	6.71E-01	1.12E-01	0.00E+00	0.00E+00
12	4.14E+01	5.69E+01	2.14E+03	4.25E+02	7.64E+02	1.13E+02	2.31E+02	6.28E+01
13	4.77E+00	2.63E+00	5.23E+02	5.23E+01	5.73E+01	5.40E+00	1.21E+01	4.15E+00
14	2.82E-01	2.80E+00	8.22E+01	1.42E+01	4.17E+01	2.67E+00	2.00E+00	1.03E+00
15	2.08E-01	1.48E-01	1.92E+01	3.02E+00	6.14E+00	3.09E-01	9.89E-01	1.76E-01
16	1.44E-01	2.87E-01	5.56E+01	4.18E+00	8.15E+00	5.02E-01	7.14E-01	5.00E-01
17	1.24E+01	1.00E+01	7.76E+01	2.54E+01	4.16E+01	2.13E+01	3.11E+01	2.03E+01
18	4.77E-01	6.45E-02	2.31E+01	4.39E+00	3.04E+00	4.87E-01	8.20E-01	4.54E-01
19	2.03E-01	2.50E-01	3.65E+02	2.19E+01	8.48E+00	4.67E-01	1.21E+00	3.56E-01
20	1.54E+00	5.37E+00	2.22E+01	1.01E+01	8.97E+00	2.57E+00	4.13E+00	1.61E+00
21	1.55E+02	1.07E+01	3.84E+02	1.01E+02	2.86E+02	4.40E+01	2.38E+02	2.51E+01
22	1.00E+02	0.00E+00	2.74E+02	3.72E+01	2.08E+02	1.84E+01	1.47E+02	1.11E+01
23	2.97E+02	2.53E+01	4.22E+02	2.88E+02	3.43E+02	1.78E+02	3.15E+02	1.60E+02
24	2.28E+02	4.47E+00	5.11E+02	3.79E+01	3.41E+02	4.59E+00	2.81E+02	5.75E+00
25	3.98E+02	1.55E-02	6.18E+02	5.09E+00	5.03E+02	3.37E+00	4.41E+02	2.41E+00
26	3.00E+02	0.00E+00	3.68E+02	1.26E+01	3.34E+02	2.57E+00	3.12E+02	1.21E+00
27	3.81E+02	1.15E+00	4.89E+02	2.93E+01	4.13E+02	4.10E+00	3.92E+02	3.34E+00
28	3.30E+02	1.00E+02	6.72E+02	2.84E+02	5.43E+02	2.30E+02	4.34E+02	2.17E+02
29	2.32E+02	4.03E+00	2.86E+02	4.41E+01	2.57E+02	1.49E+01	2.42E+02	1.27E+01
30	2.13E+05	5.90E+05	2.83E+05	4.97E+05	2.57E+05	4.06E+05	2.38E+05	3.95E+05

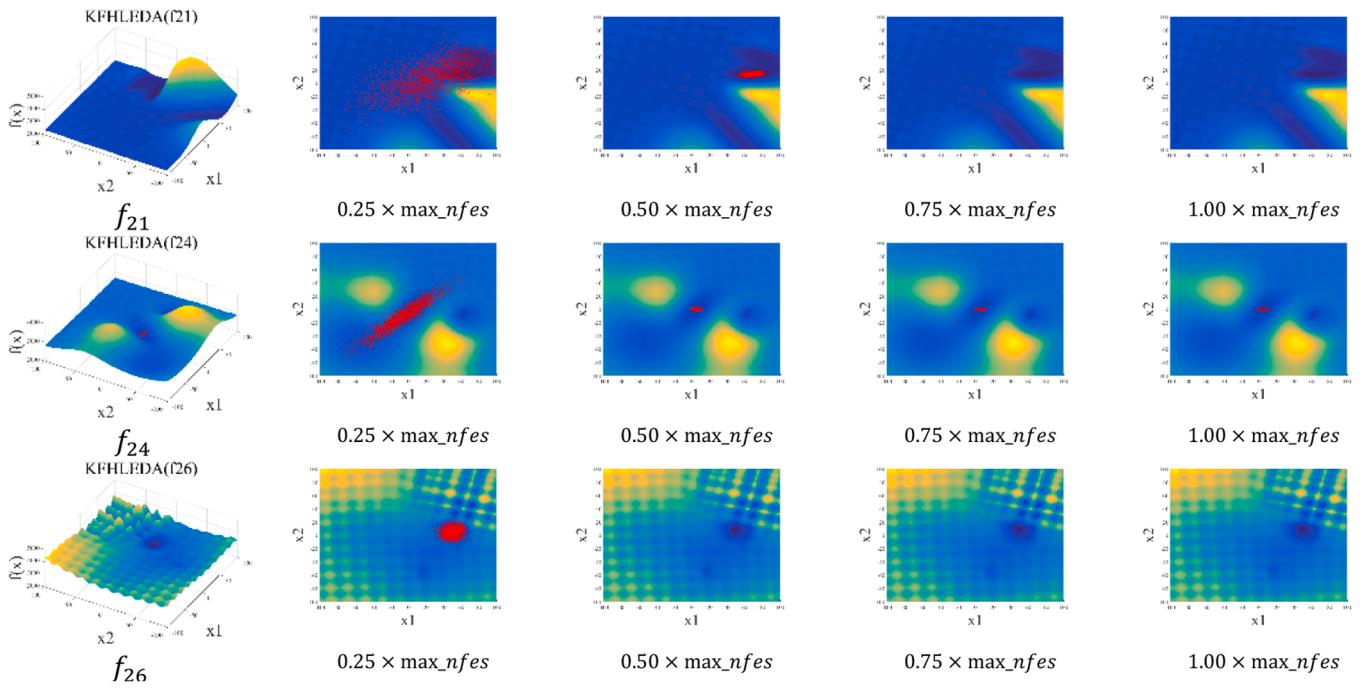
**Fig. 12.** The comparison results of KFHLEDA and the three variants on 29 functions.



**Fig. 13.** The fitness landscape of KFHLEDA on unimodal functions.



**Fig. 14.** The fitness landscape of KFHLEDA on simple multimodal functions.



**Fig. 15.** The fitness landscape of KFHLEDA on composition functions.

specific characteristics during the optimization process. The filtering mechanism is based on the elite strategy, which modifies the system gain to strengthen the filtering effect. The theoretical analysis and experiments demonstrate that the modified probabilistic model with enhanced Kalman filtering takes the revised mean as a new search center, and improves the direction of population evolution and the quality of solutions. These operations effectively improve the search efficiency of the algorithm.

The historical archive in the KFHLEDA incorporates the dominant information from the iterative process to guide population evolution, rather than solely relying on the current information. The premature convergence is avoided. Learning the information from dominant individuals modifies the mean and covariance matrix, and reduces the invalid evolutionary iterations. Moreover, the population adaptive adjustment strategy adjusts the population size to further balance exploration and exploitation during the evolution and ameliorates the overall efficiency of the KFHLEDA. The non-parametric tests show significant differences between the proposed KFHLEDA and most comparison algorithms. Furthermore, the fitness landscape completely reemerges the evolution process and the distribution of the entire population, which has guiding significance for the design of the algorithm in this study. The search behavior is visualized to show the effectiveness of the designed algorithm. The experimental results show that the KFHLEDA has superior robustness and stability on 29 functions in the CEC 2017 test suit with 10D, 30D, 50D, and 100D.

As the proposed algorithm achieves the desired results, future work will involve integrating the proposed KFHLEDA with other evolutionary algorithms and learning mechanisms for solving the constrained multi-objective continuous optimization problems. Moreover, the KFHLEDA will also be applied to solve practical problems in conjunction with hyper-heuristic algorithms, such as distributed no-wait flow shop scheduling problems with energy consumption constraints.

#### Credit authorship contribution statement

**Ningning Zhu:** Investigation, Software, Original draft, Experiments of the algorithms. **Fuqing Zhao:** Funding acquisition, Investigation, Supervision. **Ling Wang:** Methodology, Resources, Conceptualization. **Chenxin Dong:** Formal analysis, Review & editing, Visualization.

#### CRediT authorship contribution statement

**Ningning Zhu:** Writing – original draft, Software, Investigation, Data curation. **Fuqing Zhao:** Supervision, Investigation, Funding acquisition. **Ling Wang:** Resources, Methodology, Conceptualization. **Chenxin Dong:** Writing – review & editing, Visualization, Formal analysis.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### Data availability

I have shared the link to my code at my manuscript.

#### Acknowledgments

This work was financially supported by the Excellent Postgraduate Innovation Star Project of Gansu Provincial Education Department (2023CXZX-476). It was also supported by the National Natural Science Foundation of China under grant 62063021, the Key Program of National Natural Science Foundation of Gansu Province under Grant

23JYRA784, the High-level Foreign Experts Project of Gansu Province under grant 22JR10KA007, the Key Research Programs of Science and Technology Commission Foundation of Gansu Province (21YF5WA086), and Lanzhou Science Bureau project (2018-rc-98), respectively.

#### Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.swevo.2024.101502.

#### References

- [1] Z.H. Zhan, L. Shi, K.C. Tan, J. Zhang, A survey on evolutionary computation for complex continuous optimization, *Artificial Intelligence Review* 55 (2022) 59–110, <https://doi.org/10.1007/s10462-021-10042-y>.
- [2] P. Bujok, J. Tvrdík, R. Poláková, Comparison of nature-inspired population-based algorithms on continuous optimisation problems, *Swarm and Evolutionary Computation* 50 (2019) 100490, <https://doi.org/10.1016/j.swevo.2019.01.006>.
- [3] D. Bhati, P. Singh, Branch and bound computational method for multi-objective linear fractional optimization problem, *Neural Computing & Applications* 28 (2017) 3341–3351, <https://doi.org/10.1007/s00521-016-2243-6>.
- [4] A.C. Luna, N.L. Diaz, Mixed integer linear programming based energy management system for hybrid PV wind battery microgrids: modeling, design, and experimental verification, *IEEE Transactions on Power Electronics* 32 (2017) 2769–2783, <https://doi.org/10.1109/TPEL.2016.2581021>.
- [5] M.N. Omidvar, X.D. Li, X. Yao, A review of population-based metaheuristics for large-scale black-box global optimization-Part I, *IEEE Transactions on Evolutionary Computation* 26 (2022) 802–822, <https://doi.org/10.1109/TEVC.2021.3130838>.
- [6] M.N. Omidvar, X.D. Li, X. Yao, A review of population-based metaheuristics for large-scale black-box global optimization-Part II, *IEEE Transactions on Evolutionary Computation* 26 (2022) 823–843, <https://doi.org/10.1109/TEVC.2021.3130835>.
- [7] L.L. Meng, K.Z. Gao, Y.P. Ren, B. Zhang, H.Y. Sang, C.Y. Zhang, Novel MILP and CP models for distributed hybrid flowshop scheduling problem with sequence-dependent setup times, *Swarm and Evolutionary Computation* 71 (2022) 101058, <https://doi.org/10.1016/j.swevo.2022.101058>.
- [8] K.Z. Gao, Z.G. Cao, L. Zhang, Z.H. Chen, Y.Y. Han, Q.K. Pan, A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems, *IEEE-CAA Journal of Automatica Sinica* 6 (2019) 904–916, <https://doi.org/10.1109/JAS.2019.1911540>.
- [9] L.L. Meng, C.Y. Zhang, B. Zhang, K.Z. Gao, Y.P. Ren, H.Y. Sang, MILP modeling and optimization of multi-objective flexible job shop scheduling problem with controllable processing times, *Swarm and Evolutionary Computation* 82 (2023) 101374, <https://doi.org/10.1016/j.swevo.2023.101374>.
- [10] E.H. Houssein, A.G. Gad, Y.M. Wazery, P.N. Suganthan, Task scheduling in cloud computing based on meta-heuristics: review, taxonomy, open challenges, and future trends, *Swarm and Evolutionary Computation* 62 (2021), <https://doi.org/10.1016/j.swevo.2021.100841>.
- [11] H. Mühlenbein, J. Bänsch, H.M. Voigt, From recombination of genes to the estimation of distributions II. Continuous parameters, *Lecture Notes in Computer Science* 1141 (1996) 188–197, [https://doi.org/10.1007/3-540-61723-X\\_983](https://doi.org/10.1007/3-540-61723-X_983).
- [12] A. Kumar, P.P. Biswas, P.N. Suganthan, Differential evolution with orthogonal array-based initialization and a novel selection strategy, *Swarm and Evolutionary Computation* 68 (2022), <https://doi.org/10.1016/j.swevo.2021.101010>.
- [13] R. Tanabe, A. Fukunaga, Reviewing and benchmarking parameter control methods in differential evolution, *IEEE Transactions on Cybernetics* 50 (2020) 1170–1184, <https://doi.org/10.1109/TCYB.2019.2892735>.
- [14] A. Konak, S. Kulturel-Konak, Regret-based nash equilibrium sorting genetic algorithm for combinatorial game theory problems with multiple players, *Evolutionary Computation* 30 (2022) 447–478, [https://doi.org/10.1162/evco\\_a\\_00308](https://doi.org/10.1162/evco_a_00308).
- [15] P.A. Grudniewski, A.J. Sobey, Behaviour of multi-level selection genetic algorithm (MLSGA) using different individual-level selection mechanisms, *Swarm and Evolutionary Computation* 44 (2019) 852–862, <https://doi.org/10.1016/j.swevo.2018.09.005>.
- [16] R. Santana, P. Larrañaga, J.A. Lozano, Protein folding in simplified models with estimation of distribution algorithms, *IEEE Transactions on Evolutionary Computation* 12 (2008) 418–438, <https://doi.org/10.1109/TEVC.2007.906095>.
- [17] J.A. Lozano, Q. Zhang, P. Larrañaga, Guest editorial: Special issue on evolutionary algorithms based on probabilistic models, *IEEE Transactions on Evolutionary Computation* 13 (2009) 1197–1198, <https://doi.org/10.1109/TEVC.2009.2028646>.
- [18] R. Armañanzas, I. Inza, R. Santana, Y. Saeys, J.L. Flores, J.A. Lozano, Y. Van de Peer, R. Blanco, V. Robles, C. Bielza, P. Larrañaga, A review of estimation of distribution algorithms in bioinformatics, *BioData Mining* 1 (2008) 1–12, <https://doi.org/10.1186/1756-0381-1-6>.
- [19] A.M. Zhou, J.Y. Sun, Q.F. Zhang, An estimation of distribution algorithm with cheap and expensive local search methods, *IEEE Transactions on Evolutionary Computation* 19 (2015) 807–822, <https://doi.org/10.1109/TEVC.2014.2387433>.
- [20] F. Wang, Y.X. Li, A.M. Zhou, K. Tang, An estimation of distribution algorithm for mixed-variable newsvendor problems, *IEEE Transactions on Evolutionary Computation* 24 (2020) 479–493, <https://doi.org/10.1109/TEVC.2019.2932624>.

- [21] M.L. Sanyang, A. Kaban, Multivariate cauchy EDA optimisation, Intelligent Data Engineering and Automated Learning - IDEAL 8669 (2014) 449–456, [https://doi.org/10.1007/978-3-319-10840-7\\_54](https://doi.org/10.1007/978-3-319-10840-7_54).
- [22] P. Larrañaga, Optimization in continuous domains by learning and simulation of Gaussian networks, in: Proceedings of the 2000 Genetic and Evolutionary Computation Conference Workshop Program, 2000.
- [23] J.A. Lozano, P. Larrañaga, I. Inza, E. Bengoetxea, Towards a new evolutionary computation advances on estimation of distribution algorithms, Springer, 2006.
- [24] C. Gonzalez, J.A. Lozano, P. Larrañaga, Mathematical modelling of UMDAc algorithm with tournament selection. Behaviour on linear and quadratic functions, International Journal of Approximate Reasoning 31 (2002) 313–340, [https://doi.org/10.1016/S0888-613X\(02\)00092-0](https://doi.org/10.1016/S0888-613X(02)00092-0).
- [25] K.A. Folly, A short survey on population-based incremental learning algorithm, in: 2011 IEEE symposium series on computational intelligence, 2019, pp. 1–6, <https://doi.org/10.1109/SSCI44817.2019.9002858>.
- [26] M. Pelikan, H. Muehlenbein, The bivariate marginal distribution algorithm, Advances in Soft Computing (1999) 521–535, [https://doi.org/10.1007/978-1-4471-0819-1\\_39](https://doi.org/10.1007/978-1-4471-0819-1_39).
- [27] M. Pelikan, H. Mühlenbein, Marginal distributions in evolutionary algorithms, in: Proceedings of the International Conference on Genetic Algorithms Mendel, Citeseer, 1998, pp. 90–95.
- [28] S. Muelas, A. Mendiburu, A. Latorre, J.M. Peña, Distributed estimation of distribution algorithms for continuous optimization: How does the exchanged information influence their behavior? Information Sciences 268 (2014) 231–254, <https://doi.org/10.1016/j.ins.2013.10.026>.
- [29] P. Larrañaga, J.A. Lozano, Estimation of distribution algorithms: A new tool for evolutionary computation, Springer, 2001.
- [30] C. Witt, Theory of estimation of distribution algorithms, in: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, 2020, pp. 1254–1282, <https://doi.org/10.1145/3377929.3389888>.
- [31] V.P. Soloviev, P. Larrañaga, C. Bielza, Estimation of distribution algorithms using Gaussian Bayesian networks to solve industrial optimization problems constrained by environment variables, Journal of Combinatorial Optimization 44 (2022) 1077–1098, <https://doi.org/10.1007/s10878-022-00879-6>.
- [32] P.A.N. Bosman, D. Thierens, Expanding from discrete to continuous estimation of distribution algorithms: The IDEA, Parallel Problem Solving from Nature PPSN VI, Springer, 2000, pp. 767–776, [https://doi.org/10.1007/3-540-45356-3\\_75](https://doi.org/10.1007/3-540-45356-3_75).
- [33] A. Shirazi, J. Ceberio, J.A. Lozano, EDA++: Estimation of distribution algorithms with feasibility conserving mechanisms for constrained continuous optimization, IEEE Transactions on Evolutionary Computation 26 (2022) 1144–1156, <https://doi.org/10.1109/TEVC.2022.3153933>.
- [34] P.A.N. Bosman, J. Grahl, D. Thierens, Enhancing the performance of maximum-likelihood Gaussian EDAs using anticipated mean shift, Parallel Problem Solving From Nature - PPSN X 5199 (2008) 133–143, [https://doi.org/10.1007/978-3-540-87700-4\\_14](https://doi.org/10.1007/978-3-540-87700-4_14).
- [35] Z.G. Ren, Y.S. Liang, L. Wang, A.M. Zhang, B. Pang, B.Y. Li, Anisotropic adaptive variance scaling for Gaussian estimation of distribution algorithm, Knowledge-Based Systems 146 (2018) 142–151, <https://doi.org/10.1016/j.knosys.2018.02.001>.
- [36] Y.S. Liang, Z.G. Ren, X.H. Yao, Z.R. Feng, A. Chen, W.H. Guo, Enhancing Gaussian estimation of distribution algorithm by exploiting evolution direction with archive, IEEE Transactions on Cybernetics 50 (2020) 140–152, <https://doi.org/10.1109/TCYB.2018.2869567>.
- [37] M. Pelikan, K. Sastry, E CantuPaz, Scalable optimization via probabilistic modeling, Studies in Computational Intelligence 33 (2006).
- [38] P.A.N. Bosman, D. Thierens, Continuous iterated density estimation evolutionary algorithms within the IDEA framework, (2000).
- [39] M. Pelikan, K. Sastry, E.D. Goldberg, Sporadic model building for efficiency enhancement of hierarchical BOA, in: Proceedings of the 2006 Genetic and Evolutionary Computation Conference Companion 2, 2006, p. 405.
- [40] J. Grahl, P.A.N. Bosman, F. Rothlauf, The correlation-triggered adaptive variance scaling IDEA, in: Proceedings of the 2006 Genetic and Evolutionary Computation Conference Companion, 2006, p. 397.
- [41] P.A.N. Bosman, J. Grahl, F. Rothlauf, SDR: A better trigger for adaptive variance scaling in normal EDAs, in: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, 2007, pp. 492–499.
- [42] M. Wagner, A. Auger, M. Schoenauer, EEDA : A new robust estimation of distribution algorithms, inria, 2006, p. 16. <http://inria.hal.science/inria-00070802>.
- [43] Q. Yang, Y. Li, X.D. Gao, Y.Y. Ma, Z.Y. Lu, S.W. Jeon, J. Zhang, An adaptive covariance scaling estimation of distribution algorithm, Mathematics 9 (2021), <https://doi.org/10.3390/math9243207>.
- [44] Z.W. Liao, W.Y. Gong, Z.H. Cai, L. Wang, Y. Wang, Random walk mutation-based DE with EDA for nonlinear equations systems, in: 2019 IEEE Congress on Evolutionary Computation, 2019, pp. 3118–3125, <https://doi.org/10.1109/CEC.2019.8790111>.
- [45] J. Zhang, S. Sanderson, JADE: Adaptive differential evolution with optional external archive, IEEE Transactions on Evolutionary Computation 13 (2009) 945–958, <https://doi.org/10.1109/TEVC.2009.2014613>.
- [46] B. Gao, I. Wood, TAM-EDA: Multivariate t distribution, archive and mutation based estimation of distribution algorithm, Anziam Journal 54 (2012) 720–746, <https://doi.org/10.21914/anziamj.v54i0.6365>.
- [47] Z.G. Chen, Y. Lin, Y.J. Gong, Z.H. Zhan, J. Zhang, Maximizing lifetime of range-adjustable wireless sensor networks: A neighborhood-based estimation of distribution algorithm, IEEE Transactions on Cybernetics 51 (2021) 5433–5444, <https://doi.org/10.1109/TCYB.2020.2977858>.
- [48] Z.G. Ren, C.L. He, D.X. Zhong, S.S. Huang, Y.S. Liang, Enhance continuous estimation of distribution algorithm by variance enlargement and reflecting sampling, in: Proceedings of the 2016 IEEE Congress on Evolutionary Computation, 2016, pp. 3441–3447, <https://doi.org/10.1109/CEC.2016.7744225>.
- [49] N. Hansen, A global surrogate assisted CMA-ES, in: Proceedings of the 2019 IEEE Congress on Evolutionary Computation, 2019, pp. 664–672, <https://doi.org/10.1145/3321707.3321842>.
- [50] P.A.N. Bosman, J. Grahl, D. Thierens, Benchmarking parameter-free AMaLGa on functions with and without noise, Evolutionary Computation 21 (2013) 445–469, [https://doi.org/10.1162/EVCO\\_a.00094](https://doi.org/10.1162/EVCO_a.00094).
- [51] B. Doerr, M.S. Krejca, Significance-based estimation of distribution algorithms, in: Proceedings of the 2018 IEEE Congress on Evolutionary Computation 7, 2018, pp. 1483–1490, <https://doi.org/10.1145/3205455>.
- [52] X.L. Liang, H.P. Chen, J.A. Lozano, A Boltzmann-based estimation of distribution algorithm for a general resource scheduling model, IEEE Transactions on Evolutionary Computation 19 (2015) 793–806, <https://doi.org/10.1109/TEVC.2014.2382135>.
- [53] Z.Y. Lin, Q. Su, G.B. Xie, NMIEDA: Estimation of distribution algorithm based on normalized mutual information, Concurrency and Computation-Practice & Experience 33 (2021) 1–16, <https://doi.org/10.1002/cpe.6074>.
- [54] F.Q. Zhao, B. Zhu, L. Wang, T. Xu, N.N. Zhu, J. Jonrinaldi, An offline learning co-evolutionary algorithm with problem-specific knowledge, Swarm and Evolutionary Computation 75 (2022) 101148, <https://doi.org/10.1016/j.swevo.2022.101148>.
- [55] V. Stanovov, S. Ahmedova, E. Semenkin, NL-SHADE-RSP algorithm with adaptive archive and selective pressure for CEC 2021 numerical optimization, in: Proceedings of the 2021 IEEE Congress on Evolutionary Computation, 2021, pp. 809–816, <https://doi.org/10.1109/CEC45853.2021.9504959>.
- [56] A. Auger, N. Hansen, A restart CMA evolution strategy with increasing population size, in: Proceedings of the 2005 IEEE Congress on Evolutionary Computation 2, 2005, pp. 1769–1776, <https://doi.org/10.1109/CEC.2005.1554902>.
- [57] R. Tanabe, A. Fukunaga, Improving the search performance of SHADE using linear population size reduction, in: Proceedings of the 2014 IEEE Congress on Evolutionary Computation 2, 2014, pp. 1659–1665, <https://doi.org/10.1109/CEC.2014.6900380>.
- [58] P. Bujok, P. Kolenovsky, Eigen crossover in cooperative model of evolutionary algorithms applied to CEC 2022 single objective numerical optimisation, in: Proceedings of the 2022 IEEE Congress on Evolutionary Computation, 2022, <https://doi.org/10.1109/CEC55065.2022.9870433>.
- [59] H. Zhao, Z.H. Zhan, J. Liu, Outlier aware differential evolution for multimodal optimization problems, SSRN Electronic Journal (2022) 1–12, <https://doi.org/10.2139/ssrn.4040685>.
- [60] P. Yang, K. Tang, F.X. Lu, Improving estimation of distribution algorithm on multimodal problems by detecting promising areas, IEEE Transactions on Cybernetics 45 (2015) 1438–1449, <https://doi.org/10.1109/TCYB.2014.2352411>.
- [61] Q. Yang, W.N. Chen, Y. Li, C.L.P. Chen, X.M. Xu, J. Zhang, Multimodal estimation of distribution algorithms, IEEE Transactions on Cybernetics 47 (2017) 636–650, <https://doi.org/10.1109/TCYB.2016.2523000>.
- [62] N.N. Zhu, F.Q. Zhao, J. Cao, A knowledge-driven co-evolutionary algorithm assisted by cross-regional interactive learning, Engineering Applications of Artificial Intelligence 126 (2023) 107017, <https://doi.org/10.1016/j.engappai.2023.107017>.
- [63] W.S. Shao, D.C. Pi, Z.S. Shao, A Pareto-based estimation of distribution algorithm for solving multiobjective distributed no-wait flow-shop scheduling problem with sequence-dependent setup time, IEEE Transactions on Automation Science and Engineering 16 (2019) 1–17, <https://doi.org/10.1109/TASE.2018.2886303>.
- [64] Y. Du, J.Q. Li, X.L. Chen, P.Y. Duan, Q.K. Pan, Knowledge-based reinforcement learning and estimation of distribution algorithm for flexible job shop scheduling problem, IEEE Transactions on Emerging Topics in Computational Intelligence (2022) 1–17, <https://doi.org/10.1109/TETCI.2022.3145706>.
- [65] J. Ceberio, E. Irurozki, A. Mendiburu, J.A. Lozano, A distance-based ranking model estimation of distribution algorithm for the flowshop scheduling problem, IEEE Transactions on Evolutionary Computation 18 (2014) 286–300, <https://doi.org/10.1109/TEVC.2013.2260548>.
- [66] Z.Q. Zhang, B. Qian, R. Hu, H.P. Jin, L. Wang, A matrix-cube-based estimation of distribution algorithm for the distributed assembly permutation flow-shop scheduling problem, Swarm and Evolutionary Computation 60 (2021) 100785, <https://doi.org/10.1016/j.swevo.2020.100785>.
- [67] R.E. Kalman, A new approach to linear filtering and prediction problems, Journal of Basic Engineering 82 (1960) 35–45, <https://doi.org/10.1115/1.3662552>.
- [68] W.C. Zhong, J. Liu, F. Liu, L.C. Jiao, Second order estimation of distribution algorithms based on Kalman Filter, Chinese Journal of Computers 27 (2004) 1272–1277.
- [69] M. Raitoharju, A.F. Garcia-Fernandez, R. Piche, Kullback-Leibler divergence approach to partitioned update Kalman filter, Signal Processing 130 (2017) 289–298, <https://doi.org/10.1016/j.sigpro.2016.07.007>.
- [70] Y.Y. Zhang, Elite archives-driven particle swarm optimization for large scale numerical optimization and its engineering applications, Swarm and Evolutionary Computation 76 (2023), <https://doi.org/10.1016/j.swevo.2022.101212>.
- [71] H.D. de Mello, L. Martí, A.V.A. da Cruz, M. Vellasco, Evolutionary algorithms and elliptical copulas applied to continuous optimization problems, Information Sciences 369 (2016) 419–440, <https://doi.org/10.1016/j.ins.2016.07.006>.
- [72] H. Zhao, Areas (volumes) of n-dimension ellipsoid by quadratic curve (surface) enclosed, Mathematics in Practice and Theory 43 (2013) 279–282.
- [73] Q.X. Yin, The inverse problem of rank -1 modification of real symmetric matrices, Journal of Nantong Institute of Technology (Natural Science) (2003).

- [74] N.H. Awad, M.Z. Ali, J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective bound constrained real-parameter numerical optimization, in: Technical Report, Singapore, (2016) 1–34.
- [75] M.J. Anderson, C.J.F. Ter Braak, Permutation tests for multi-factorial analysis of variance, *Journal of Statistical Computation and Simulation* 73 (2003) 85–113, <https://doi.org/10.1080/00949650215733>.
- [76] J. Brest, M.S. Maucec, B. Boskovic, Single objective real-parameter optimization: Algorithm jSO, in: Proceedings of the 2017 IEEE Congress on Evolutionary Computation, 2017, <https://doi.org/10.1109/CEC.2017.7969456>.
- [77] R. Jiang, R. Shankaran, S. Wang, T. Chao, A proportional, integral and derivative differential evolution algorithm for global optimization, *Expert Systems with Applications* 206 (2022) 1–29, <https://doi.org/10.1016/j.eswa.2022.117669>.
- [78] N.H. Awad, M.Z. Ali, P.N. Suganthan, Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems, in: Proceedings of the 2017 IEEE Congress on Evolutionary Computation, 2017, pp. 372–379, <https://doi.org/10.1109/CEC.2017.7969336>.
- [79] A.W. Mohamed, A.A. Hadi, A.M. Fattouh, K.M. Jambi, LSHADE with semi-parameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems, in: Proceedings of the 2017 IEEE Congress on Evolutionary Computation, 2017, pp. 145–152, <https://doi.org/10.1109/CEC.2017.7969307>.
- [80] Q.L. Dang, W.F. Gao, M.G. Gong, An efficient mixture sampling model for gaussian estimation of distribution algorithm, *Information Sciences* 608 (2022) 1157–1182, <https://doi.org/10.1016/j.ins.2022.07.016>.
- [81] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, *IEEE Transactions on Evolutionary Computation* 1 (1997) 67–82, <https://doi.org/10.1109/4235.585893>.
- [82] M.H. Tayarani-N, A. Prugel-Bennett, Anatomy of the fitness landscape for dense graph-colouring problem, *Swarm and Evolutionary Computation* 22 (2015) 47–65, <https://doi.org/10.1016/j.swevo.2015.01.005>.