

RESEARCH ARTICLE

Efficient Surrogate Model Assisted Estimation of Distribution Algorithm for Expensive Optimization

JIN SHANG¹, GUIYING LI², HAO HAO³, AND YUFANG ZHANG¹¹Control Technology Institute, Wuxi Institute of Technology, Wuxi 214121, China²School of Mechanical & Electrical Engineering, Heilongjiang University, Harbin 150080, China³Institute of Natural Science, Shanghai Jiao Tong University, Shanghai 200240, China

Corresponding author: Guiying Li (liguiying@hlju.edu.cn)

This work was supported by Jiangsu Province University Science and Technology Innovation Team Project.

ABSTRACT In recent years, several surrogate assisted evolutionary algorithms (SAEAs) have been proposed to solve expensive optimization problems. These problems lack explicit expressions and are characterized by high invocation costs. SAEAs leverage surrogate models to accelerate convergence towards the optimal region and reduce the number of function evaluations. While Gaussian Processes (GPs) are widely used due to their robustness and capability of providing uncertainty estimates, their applicability becomes limited in scenarios involving a large number of samples or high-dimensional spaces. This is due to their cubic time complexity in relation to the number of samples, which results in prohibitive computational demands for large-scale problems. To address the challenge, this work presents an efficient surrogate model-assisted estimation of the distribution algorithm (ESAEDA). This method employs a random forest as a surrogate model and combines it with a GP-hedge acquisition strategy to ensure the efficiency and accuracy of model-assisted selection. An improved EDA model called the variable-width histogram model with some unevaluated solutions is used to generate new solutions. To demonstrate the benefits of the proposed method, we compared ESAEDA with several state-of-the-art surrogate-assisted evaluation algorithms and the Bayesian optimization method. Experimental results demonstrate the superiority of the proposed algorithm over these comparison algorithms for two well-known test suites.

INDEX TERMS Artificial intelligence, optimization algorithm, random forest.

I. INTRODUCTION

In some real-world applications, the box-constrained problem is computationally expensive and is considered a black-box function because its inner workings are difficult to trace and it can only be accessed based on its inputs and outputs. The expensive evaluation cost, coupled with the lack of a closed-form objective function, poses significant challenges to traditional numerical and heuristic optimization methods.

Classical optimization methods, such as linear programming and convex optimization, may not always be suitable for addressing the aforementioned problems due to their

reliance on favorable function properties [1]. As a result, there has been growing interest in the use of derivative-free or heuristic optimization methods [2], [3], [4], which do not rely on strong assumptions or closed-form dependencies of the optimization problem. Among these methods, evolutionary algorithms (EAs) show promise due to their trial-and-error strategies, but their effectiveness heavily relies on fitness values (objective values) to guide the search [5]. Therefore, frequent evaluations of the objective function pose challenges in practice, especially in expensive black-box optimization problems [6], [7].

In recent years, Surrogate-Assisted Evolutionary Algorithms (SAEAs) have emerged as a promising approach to tackling expensive optimization problems. In SAEAs, the

The associate editor coordinating the review of this manuscript and approving it for publication was Qiang Li.

surrogate model replaces the expensive real function evaluations. Since surrogate modeling and prediction require much less computational budget than directly using the expensive real black-box function, the approach can significantly reduce the optimization cost. SAEAs consist of two significant operators, namely the reproduction operator and the model-assisted selection operator. The former is responsible for exploring the search space, while the latter is accountable for selecting promising offspring solutions. The two operators alternate their execution, driving the population towards the optimal region.

In current research on SAEAs, a significant amount of work has been focused on studying model-assisted operators. Based on the type of model, this can be divided into three categories. The first category is regression-based SAEAs, which use a model to map a decision variable vector to its corresponding objective value. Popular surrogate models used for this type of SAEA include polynomial regression [8], radial basis function (RBF) networks [9], and Gaussian processes (GPs) [10]. The second category is classification-based SAEAs, where solutions are assigned labels based on their quality. Surrogate models used in this paradigm include support vector machines [11], artificial neural networks [12], and fuzzy K-nearest neighbors [2]. The third category is relation-based SAEAs. Loshchilov et al. [13] argue that comparison-based optimization algorithms like evolutionary algorithms require surrogate models that are also comparison-based. This is because the relational model is used to learn the comparative relationships between solutions. While supervised learning methods can be used to learn these relationships, the key difference is that the relations between solutions (i.e., superiority or inferiority) are learned as features, rather than features of a single solution. For example, in [13] and [14], surrogate models are used to predict the ranks of solutions. In another study [15], a convolutional neural network (CNN) is proposed to approximate the difference in fitness function values between two solutions, rather than directly approximating the fitness of a solution. Recently, Hao et al. [5], [16] proposed a series of methods to learn the relationship between pairs of solutions.

Among the various surrogate models highlighted in the literature, GPs have emerged as the preferred surrogate model for SAEAs due to their reliable modeling capabilities and accurate estimation of model response uncertainty. The combination of Gaussian Processes with acquisition functions, also known as model management strategies [17], allows for effective balancing of exploration and exploitation trade-offs, enabling SAEAs to exhibit powerful global optimization abilities when tackling expensive black-box optimization problems. However, due to the high computational cost of GPs modeling, which is scaled cubically concerning the number of data points [18], it is not suitable for medium-to-large scale problems, which has limited the development of SAEAs.

In terms of reproduction operators, current SAEAs employ methods such as genetic algorithms (GA) [12], differential evolution (DE) [19], and particle swarm optimization (PSO)

[20] as operators for producing new solutions. These operators have all demonstrated good global search capabilities in SAEAs. However, another type of population-based solution generation operator has not been widely used in SAEAs, namely the distribution estimation algorithm (EDA) [21]. This method models the current population and generates new solutions through sampling. By utilizing global information to guide population search, it exhibits stronger global optimization capabilities, potentially enhancing the search efficiency of SAEAs.

Cai et al. [22] explore event-triggered consensus in multi-agent systems, highlighting efficiency in communication and computation, crucial for cooperative control in distributed systems. He et al. [23] introduce the Dynamic Opinion Maximization Framework (DOMF) for social networks, addressing the dynamic nature of user opinions and challenges like non-monotonicity and non-submodularity. They present a blockchain-based solution for secure data offloading in healthcare, combining blockchain technology with deep reinforcement learning for enhanced data privacy and computational efficiency [24]. Cai et al. [25] focus on adaptive control in heterogeneous systems, improving real-time response and synchronization, essential for managing multi-agent system behaviors. Zhong et al. [26] introduce a surrogate ensemble-assisted hyper-heuristic algorithm designed to tackle expensive optimization problems. It combines multiple surrogate models to estimate the fitness of potential solutions, enhancing the efficiency and accuracy of finding optimal solutions. This method is particularly useful in scenarios where traditional optimization techniques fall short due to computational expense or complexity. Pan et al. [27] present a hybrid optimization algorithm that integrates surrogate models with evolutionary strategies. It aims to reduce computational costs while solving expensive optimization challenges, demonstrating significant improvements in both speed and solution quality. The approach provides a promising direction for addressing complex problems in various engineering fields. Chen et al. [28] detail a surrogate model-assisted algorithm for multi-objective optimization with an application in wind farm layout design. By incorporating Sparse Gaussian Process models, the algorithm achieves high accuracy and computational efficiency, showcasing its capability to enhance wind farm performance and its applicability to real-world engineering problems. He et al. [29] present a reinforcement-learning-based approach for competitive opinion maximization in signed social networks. The authors develop a two-phase model incorporating an activated dynamic opinion model and a reinforcement-learning-based seeding process to identify and influence key individuals in social networks, optimizing the spread of opinions against competitive opinions. The results confirm the method's effectiveness in achieving superior opinion propagation across various datasets.

In SA-EDA [30], a GPs assisted EDA algorithm is proposed, which utilizes a variable-width histogram (VWH) to represent the distribution of the population. Due to its

focus on the potentially optimal regions, this VWH exhibits good performance on expensive black-box problems. However, due to the use of Gaussian processes as the surrogate model, the algorithm still faces challenges related to the curse of dimensionality and high computational complexity. To address these issues, this paper proposes an efficient surrogate assisted EDA, called ESAEDA. The main contributions of this algorithm are as follows:

- We introduce an efficient surrogate model-assisted estimation of distribution algorithm (ESAEDA) that employs a random forest as the surrogate model. This choice is pivotal as it significantly reduces the training budget compared to traditional Gaussian processes, thereby enhancing computational efficiency without compromising accuracy.
- ESAEDA incorporates multiple acquisition strategies to optimize model-assisted selection. This method leverages the predictive power of the surrogate model to select promising solutions from a pool of candidates, thereby reducing the number of expensive function evaluations needed.
- A distinctive feature of our approach is the incorporation of unevaluated solutions in the updating of the probability model. This technique allows for a richer exploration of the solution space, potentially uncovering superior solutions that might not be immediately evident through evaluated solutions alone.
- To improve the search capability on low-dimensional problems, we integrate the Powell method, a derivative-free optimization technique. This integration is specifically tailored to enhance the exploration capabilities of ESAEDA, particularly in scenarios where traditional methods may falter due to limited solution evaluations.

Section II provides an overview of related works. Section III presents a detailed description of the proposed method. Section IV empirically studies the new method with other comparisons on two widely used test suites. Finally, Section V concludes this paper and discusses some future works.

II. RELATED WORK

A. ESTIMATION OF DISTRIBUTION ALGORITHM

Estimation of distribution algorithms (EDAs) is a distinct evolutionary computation paradigm that employs probabilistic models to guide population search, unlike traditional genetic operators such as crossover or mutation. In EDAs, variable linkages are typically represented using Bayesian networks, which are learned from data using statistical and machine learning techniques. For EDAs to generate high quality samples, it is imperative to have a suitable Bayesian network model. Based on the variable linkages, Bayesian network models in EDAs can be classified into three categories [3]: univariate models, bivariate models, and multivariate models. Univariate histogram marginal models are a class of simple univariate histogram models that model

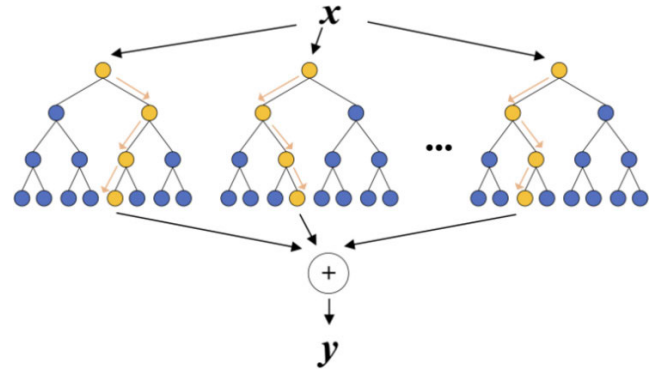


FIGURE 1. Illustration of random forest.

the distribution of the population using histograms. Decision variables are assumed to be independent and the joint probability of the variables is fully factorized as:

$$P(\mathbf{x}) = \prod_{j=1}^n P(x_j) \quad (1)$$

where x_j is the j th variable. In modeling, the data is divided into several bins according to a certain method, and each bin represents a probability interval. The probability weight of each bin is calculated based on the ratio of the number of samples it contains to the total number of samples. Four types of histogram models can be used to model for each dimension [3], [31], [32], including equal-width histogram (EWH) models, equal-height histogram (EHH) models, cluster-based histogram (CBH) models, and variable-width histogram (VWH) models. Among them, VWH models pay more attention to the potential optimal region in an EDA with cheap and expensive local search methods (EDALS) [3], providing stronger convergence speed and therefore serving as the basic model for the generation operator in this paper.

B. SURROGATE MODEL

This study centers on the application of surrogate models in machine learning, with particular emphasis on random forests as a viable option due to their lower computational complexity in comparison to Gaussian processes. Figure 1 illustrates a schematic diagram of the random forest algorithm, which involves averaging multiple decision trees to decrease variances in a given dataset. The decision trees are trained on various subsets of the data, enabling robust and accurate predictions. For dataset $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, the bagging method is repeated (B times) to sample from \mathcal{D} with put-backs and then train the tree model on these samples. After training, the prediction of a testing point \mathbf{x} can be achieved by averaging the predictions of all individual regression trees on \mathbf{x} . The predicted value is defined as follows.

$$\hat{f}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B f_b(\mathbf{x}) \quad (2)$$

The predictions of all regression trees on \mathbf{x} can be used as an estimate of the uncertainty.

$$\sigma(\mathbf{x}) = \sqrt{\frac{1}{B} \sum_{b=1}^B (f_b(\mathbf{x}) - f'')^2} \quad (3)$$

Random forests have gained widespread popularity in various practical fields due to their robust data fitting capabilities and relatively lower computational complexity. As a result, RF has been increasingly utilized as a surrogate model in various areas such as trauma systems [33], deep neural network optimization [34], and combinatorial optimization [35]. In this study, we utilize RF to fit black-box objective functions and reduce computational costs, with the ultimate goal of assisting the EDA in optimization.

III. PROPOSED METHOD

A. ALGORITHM FRAMEWORK

ESAEDA is presented in Algorithm 1. Its components are explained as follows.

1) INITIALIZATION

N initial solutions are obtained by sampling from the search space $\Pi_{i=1}^n [a_i, b_i]$ using the Latin hypercube sampling method (LHS) [36]. Each of these solutions undergoes evaluation by the real expensive function $f(\cdot)$ and is subsequently stored in the archive \mathcal{A} . The count of evaluated solutions, denoted by fes , is updated accordingly. Based on the current population \mathcal{P} , a probability distribution model $P_{VWH}(X)$ is constructed [3].

2) STOPPING CONDITION

The termination criterion is defined as the number of real function evaluations exceeding the pre-specified maximum limit, denoted by FE_{max} .

3) SURROGATE MODEL TRAINING

The surrogate model \mathcal{M} is trained using the archive \mathcal{A} . Given the low complexity of the random forest, all solutions that stored in the archive are utilized to train the model.

4) GENERATE SOLUTIONS

The new N solutions are composed of two parts, where Q_I is generated by sampling from the $P_{VWH}(X)$, while Q_{II} is produced by local search on the current population \mathcal{P} . According to [3], this approach combines both global and local population information to enhance the quality of the offspring solutions.

5) SURROGATE ASSISTED SELECTION

The surrogate model \mathcal{M} aids in two tasks. Firstly, utilizing a multiacquisition strategy based on the predicted value $\hat{f}(\mathbf{x})$ and uncertainty $\sigma(\mathbf{x})$ to select the optimal solution Q_{best} . Secondly, relying on the model predicted values, the top $N/2$ solutions are chosen as the unevaluated good solutions.

Algorithm 1 Pseudocode of Main Framework

Input: N (population size); FE_{max} (maximum number of FEs);
 n' (threshold for using the Powell method).
// Initialization
1 $\mathcal{P} \leftarrow \text{LHS}(N)$.
2 $\mathcal{A} \leftarrow \mathcal{P}$.
3 $fes \leftarrow N$.
4 $P_{VWH}(X) \leftarrow \text{EDA}(\mathcal{P})$.
5 **while** $fes \leftarrow N \leq FE_{max}$ **do**
// Surrogate model training
6 $\mathcal{M} \leftarrow \text{Training}(\mathcal{A})$.
// Generate new solutions
7 $Q_I \leftarrow \text{Sample}(P_{VWH}(X))$.
8 $Q_{II} \leftarrow \text{localsearch}(\mathcal{P})$.
9 $Q \leftarrow \text{Combination}(Q_I, Q_{II})$.
// Surrogate assisted selection
10 $Q_{best}, Q' \leftarrow \text{SA_Selection}(Q, \mathcal{M})$.
// Update archive
11 $\mathcal{A} \leftarrow \mathcal{A} \cup \text{Evaluation}(Q_{best})$.
12 $fes \leftarrow fes + 1$.
// Improve solution by Powell method
13 **if** *improving* and $n \leq n'$ **then**
14 $Q \leftarrow \text{Powell}(\mathcal{A})$.
15 $\mathcal{A} \leftarrow \mathcal{A} \cup Q$.
16 $fes \leftarrow fes + |Q|$.
17 **end**
// Selection for next generation
18 $\mathcal{P} \leftarrow \mathcal{A}_{1:N}$.
// Update probability model
19 $P(X) \leftarrow \text{Model}(\mathcal{P}, Q')$.
20 **end**
Output: \mathcal{A}_{best} (the optimum solution).

6) UPDATE ARCHIVE

The current best solution Q_{best} is evaluated using the real fitness function and stored in the archive \mathcal{A} .

7) IMPROVE SOLUTION BY POWELL METHOD

Powell method [37] is a derivative-free optimization algorithm that can effectively improve the convergence of our algorithm. Due to the significant fitness evaluation budget required by this method, which increases with the search dimension, we have devised a simple condition to activate it. Specifically, the Powell method will be used once when the search space dimension is no greater than 20 and there is no improvement in the \mathcal{A} continuity for 10 iterations. All solutions evaluated during this process will be saved in \mathcal{A} .

8) SELECT SOLUTIONS FOR NEXT GENERATION

The next population \mathcal{P} is formed by selecting the optimal N solutions based on the objective value from \mathcal{A} .

9) UPDATE PROBABILITY MODEL

The new EDA model will be updated, and the population will be derived from two parts. One part is the population \mathcal{P} selected based on the real function values, while the other part is the population \mathcal{Q}' selected based on the surrogate model. This approach can further accelerate the convergence of the population towards the optimal region in next iteration.

The ESAEDA algorithm encompasses several principal components: initialization, training of the surrogate model, generation of new solutions, surrogate-assisted selection, archive updates, and updates to the probability model. The initialization phase predominantly utilizes Latin hypercube sampling, manifesting a time complexity of $\mathcal{O}(n)$, where N signifies the size of the population. In the training of the surrogate model, a Random Forest method is employed, which incurs an approximate time complexity of $\mathcal{O}(MN \log(N))$, M being the number of trees involved. The generation of new solutions hinges on the probability distribution model of the current population, entailing a time complexity of $\mathcal{O}(ND)$, with D representing the dimensionality of the problem at hand. For the surrogate-assisted selection phase, the time complexity scales as $\mathcal{O}(MND)$. Meanwhile, the tasks of updating the archive and the probability model are marked by high efficiency, each showcasing a time complexity of $\mathcal{O}(N)$. These integral steps collectively constitute the computational backbone of the ESAEDA algorithm as outlined in this study.

Algorithm 1 and SAEDA [30] share some similar processes, but to improve algorithm efficiency, we have the following main differences: 1) utilizing a random forest as a surrogate model to reduce modeling overhead, 2) using the Powell method to compensate for the degradation of search ability in low-dimensional spaces (compared to SAEDA), and 3) using unevaluated solutions to accelerate the search speed of the algorithm. In the following sections, we will provide further details regarding the new solution generation and offspring solutions selection techniques employed in our proposed algorithm.

B. NEW SOLUTION GENERATION

In EAs, the quality of newly generated solutions directly affects the efficiency of the algorithm search. In this work, new solutions are generated through a combination of EDA and local search, and the Powell method is used to further improve the quality of low-dimensional spaces.

In this study, we utilize a variable-width histogram (VWH) model, as proposed by Zhou et al. [3], to serve as a probabilistic model. This approach focuses on identifying and exploiting promising regions within the search space, which is essential for achieving improved performance on expensive optimization problems. Rather than utilizing traditional crossover and mutation operators, the new population \mathcal{Q}_I is generated by sampling from the probability distribution $P(X)$ of the current population \mathcal{P} . The VWH model represents a univariate model, where the n dimensions of the solution vector $\mathbf{x} = (x_1, x_2, \dots, x_n)$ are assumed to be independent

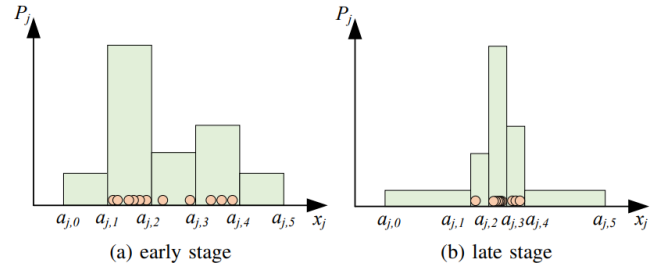


FIGURE 2. Illustration of VWH model for population on early and late search stage.

of each other. The model can be expressed mathematically using (2), where $P_j(x_j)$ denotes the marginal probability function of the j^{th} variable x_j .

In modeling, to partition the search space $[a_j, b_j]$ of the j^{th} variable, a total of M bins are created, denoted by $[a_{j,m}, a_{j,m} + 1)$, $m = 0, 1, \dots, M - 2$ and $[a_{j,M-1}, a_{j,M}]$, where $a_{j,0} = a_j$ and $a_{j,M} = b_j$. The bins from the 2nd to the $(M - 1)^{th}$ are of equal width, focusing on the distribution of the current population. The size of each bin is determined based on the number of solutions falling in the current interval. The first and last two bins of the search space without solutions still have a very low probability to avoid premature convergence. Figure 2 illustrates the basic idea of VWH with different population distributions. The probability defined by $\prod_{j=1}^n [a_{j,1}, a_{j,M-1}]$ is higher than other no-solutions space, which focuses more on the promising search areas, especially when the population converges to a small area (see Figure 2 (b)).

To generate a new solution \mathbf{x} , the probability distribution $P(X)$ is used to sample each component $x_j, j = 1, \dots, n$ independently. For each component, a bin is selected based on its corresponding probability function $P_j(x_j)$. Then, a value is randomly sampled within the boundaries of the selected bin to obtain the current x_j . This process is repeated for each component to obtain a complete solution \mathbf{x} . A variable-width histogram (VWH) model, which partitions the search space into bins, is used as the probability distribution model. The size of each bin is determined by the number of solutions falling within that interval, with the bins in the middle focusing on the distribution of the current population. This approach focuses more on promising search areas and improves the performance of the SAEAs on expensive problems. In addition to using evaluated solutions in the archive to update the VWH model, this work also uses unevaluated solutions \mathcal{Q}' selected by the surrogate model to update the probability model, which improves the quality of newly sampled solutions and makes the VWH model work better in SAEAs.

Another part of the solutions will be generated by local search (Algorithm 1 line 8). This is achieved by using the method proposed in EDALS [3], which constructs a lightweight quadratic approximation model based on the parent population \mathcal{P} , and then generates \mathcal{Q}_{II} based on the

quadratic approximation model. Finally, Q_I and Q_{II} are randomly combined with a certain probability, following the ratio of 0.8 for Q_I and 0.2 for Q_{II} as proposed in [3] and [30]. Local search serves as a supplement to the EDA search.

In ESAEDA, we utilized RF as a surrogate model to improve the algorithm execution speed. However, surpassing the Gaussian process assisted EDA (SA-EDA) [30] in low-dimensional space became a challenging task. Therefore, we applied the Powell method (Algorithm 1 line 8) [37] to further enhance the quality of solutions. Due to the consumption of the evaluation budget required for this method and its increase with the dimensionality of the search space, we restricted its use to low-dimensional cases.

C. OFFSPRING SELECTION

This subsection outlines the methodology employed for training and implementing a surrogate model for selecting candidate solutions from the current offspring population Q in the context of ESAEDA. The surrogate model is used to replace the expensive function, as outlined in (3). The prediction result obtained by multiple trees is utilized to estimate the modeling uncertainty of the current point, as denoted in (4). To determine the final Q_{best} solution, multiple acquisition functions' selection results are aggregated using the GP-Hedge method [38]. Additionally, top $N/2$ population Q' is selected based on the predicted value $\hat{f}(\cdot)$ by RF, which is subsequently used to update the VWH model.

In this work, acquisition functions are utilized to balance exploitation and exploration in the search process by selecting a promising solution Q_{best} based on the mean (μ or \hat{f}) and variance (σ) of the surrogate model predictions. To enhance the robustness of surrogate-assisted selection, the GP-hedge method is used to dynamically select a strategy from the three typical acquisition functions: expected improvement (EI), probability improvement (PI), and lower confidence bound (LCB), formulated in (5), (6), and (7), respectively. Here, \mathbf{x}^* represents the current best solution in the archive, $\Phi(\cdot)$ is the standard normal distribution probability density function, and β balances the expectation and variance in LCB. In the t -th generation, \mathbf{x}_{EI}^* , \mathbf{x}_{PI}^* , and \mathbf{x}_{LCB}^* denote the optimal solutions that satisfy $\mathbf{x}_{EI}^* = \arg \max(EI(\mathbf{x}))$, $\mathbf{x}_{PI}^* = \arg \max(PI(\mathbf{x}))$, and $\mathbf{x}_{LCB}^* = \arg \min(LCB(\mathbf{x}))$, where $\mathbf{x} \in Q$. GP-hedge selects a solution as Q_{best} based on the probability of the gains obtained by the three strategies in the previous $t-1$ generations. When $t = 1$, a solution is randomly selected from the three candidates. The details of calculating the gains can be found in GP-Hedge.

$$EI(\mathbf{x}) = (\mu(\mathbf{x}) - \mathbf{x}^*)\Phi\left(\frac{\mu(\mathbf{x}) - \mathbf{x}^*}{\sigma(\mathbf{x})}\right) + \sigma(\mathbf{x})\Phi\left(\frac{\mu(\mathbf{x}) - \mathbf{x}^*}{\sigma(\mathbf{x})}\right) \quad (4)$$

$$PI(\mathbf{x}) = \Phi\left(\frac{\mathbf{x}^* - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right) \quad (5)$$

$$LCU(\mathbf{x}) = -(\mu(\mathbf{x}) - \sqrt{\beta}\sigma(\mathbf{x})) \quad (6)$$

Moreover, using the approximate value \hat{f} provided by the surrogate model, the new population Q is sorted in ascending order, and the top half of solutions are selected as the promising candidates without real evaluation to update the VWH model.

IV. EXPERIMENTAL COMPARISON

To evaluate the effectiveness of the proposed methods, experiments were conducted on the LZG [27] and YLL [39] test suites in both 20 and 50 dimensions, along with some state-of-the-art methods, including surrogate model assisted algorithms, Bayesian optimization algorithms, and general evolutionary algorithms. The study included ablation experiments on two crucial components of the ESAEDA algorithm, called the Powell method and the VWH model with unevaluated solutions. Additionally, the runtime performance of ESAEDA is compared with some GPs assisted algorithms.

A. EXPERIMENTAL SETTINGS

1) TEST SUITES

For the empirical study, two test suites were used. The first was the LZG test suite, which includes four test functions, namely Ellipsoid, Rosenbrock, Ackley, and Griewank. These functions have unimodal, gully, and multimodal landscapes. The second test suite used was the YLL test suite, which contains functions $f1-f4$ with unimodal landscapes, $f5, f8-f13$ with multimodal landscapes, $f6$ with a step landscape, and $f7$ with random noise. The problems in both test suites were evaluated in dimensions $n = 20$ for small-scale and $n = 50$ for median-scale.

2) ALGORITHM IN STUDY

Six algorithms are chosen for empirical study, namely CMA-ES [4], FCPS-CoDE [2], EDALS [3], SAMSO [28], Skopt, GPME [27], and SA-EDA [30]. These algorithms can be divided into three categories:

- Basic EDA: CMA-ES and EDALS are two general EDAs, that are not specially designed for expensive optimization.
- Bayesian optimization: Skopt is an efficient global optimization algorithm in the Bayesian optimization (BO) framework. It applies GPs as the surrogate model.
- Surrogate assisted EAs: FCPS-CoDE uses a fuzzy K nearest neighbor based classification model to evaluate candidate solutions. GPME uses GPs to evaluate candidate solutions. SAMSO is a surrogate assigned particle swarm optimization with RBF. SA-EDA is a GPs assisted EDA.

3) PARAMETER SETTINGS

In the empirical study, we use the recommended parameters in the original literature for each algorithm for a fair comparison, and the details are given below.

TABLE 1. Statistics of median results obtained by nine comparison algorithms on LZG and YLL test suites with n = 20.

Problem	p-value	CMA-ES	FCPS-CoDE	EDALS	Skopt	GPEME	SAMSO	SA-EDA	ESAEDA*	ESAEDA**
LZG01	7.15e-51	1.50e+02[9](-) (4.25e+01)	1.30e+02[8](-) (3.13e+01)	7.17e+01[7](-) (1.56e+01)	1.81e+00[4](-) (2.29e+00)	3.20e-01[3](-) (1.77e-01)	1.87e+01[5](-) (2.52e+01)	2.34e-02[2](-) (8.71e-03)	3.78e+01[6](-) (9.94e+00)	1.17e-30[1] (8.16e-31)
LZG02	1.31e-41	3.03e+02[7](-) (7.50e+01)	3.22e+02[8](-) (1.05e+02)	2.37e+02[6](-) (4.02e+01)	6.45e+02[9](-) (1.62e+03)	1.27e+02[4](e) (4.32e+01)	3.57e+01[1](+) (2.47e+01)	7.07e+01[2](+) (3.09e+01)	1.76e+02[5](-) (3.59e+01)	1.16e+02[3] (2.34e+01)
LZG03	3.67e-45	1.59e+01[8](-) (1.28e+00)	1.48e+01[6](-) (1.00e+00)	1.33e+01[5](-) (7.37e-01)	1.49e+01[7](-) (6.97e+00)	3.77e+00[3](-) (5.52e+01)	1.83e+01[9](-) (1.33e+00)	3.35e+00[2](-) (3.84e-01)	7.48e+00[4](-) (4.18e+00)	3.03e-14[1] (7.71e-15)
LZG04	5.29e-51	5.09e+01[8](-) (1.06e+01)	5.46e+01[9](-) (1.33e+01)	2.96e+01[7](-) (7.62e+00)	1.19e+00[4](-) (3.04e-01)	1.17e+00[3](-) (8.76e-02)	2.06e+01[6](-) (1.33e+01)	1.00e+00[2](-) (1.45e-02)	5.89e+00[5](-) (4.26e+00)	2.99e-01[1] (2.10e-01)
YLLF01	2.40e-50	6.32e+03[9](-) (1.98e+03)	5.37e+03[8](-) (1.86e+03)	3.17e+03[7](-) (5.92e+02)	2.07e+01[4](-) (5.61e+01)	1.67e+01[3](-) (1.09e+01)	6.73e+02[5](-) (6.43e+02)	2.52e+00[2](-) (5.62e+00)	8.39e+02[6](-) (9.27e+02)	8.34e-29[1] (2.13e-29)
YLLF02	7.83e-48	1.15e+02[7](-) (2.20e+02)	2.41e+01[3](-) (3.67e+00)	2.67e+01[4](-) (3.93e+00)	8.08e+16[9](-) (2.18e+17)	5.03e+02[8](-) (1.04e+03)	3.27e+01[5](-) (1.72e+01)	3.61e+01[6](-) (6.07e+00)	1.40e+01[2](-) (1.43e+00)	2.14e-14[1] (2.70e-15)
YLLF03	9.84e-25	2.09e+04[5](-) (5.68e+03)	1.26e+04[3](-) (3.48e+03)	2.32e+04[6](-) (4.21e+03)	5.29e+04[9](-) (3.37e+04)	2.99e+04[8](-) (9.39e+03)	1.71e+04[4](e) (1.38e+04)	2.57e+04[7](-) (6.26e+03)	1.12e+04[2](e) (4.99e+03)	9.74e+03[1] (5.95e+03)
YLLF04	3.21e-33	4.03e+01[5](+) (1.25e+01)	4.17e+01[6](+) (6.55e+00)	3.28e+01[4](+) (3.26e+00)	7.86e+01[9](-) (3.14e+01)	3.09e+01[3](+) (8.94e+00)	7.06e+01[8](e) (7.40e+00)	2.47e+01[1](+) (5.58e+00)	2.69e+01[2](+) (1.98e+01)	7.00e+01[7] (1.05e+01)
YLLF05	9.73e-44	3.11e+06[8](-) (1.43e+06)	4.71e+06[9](-) (2.84e+06)	1.13e+06[6](-) (5.76e+05)	1.25e+06[7](-) (1.27e+06)	3.05e+05[5](-) (1.60e+05)	1.12e+05[3](-) (8.27e+04)	7.26e+04[2](-) (4.21e+04)	2.66e+05[4](e) (7.04e+05)	2.40e+04[1] (2.36e+04)
YLLF06	7.13e-47	5.79e+03[8](e) (1.20e+03)	6.26e+03[9](e) (1.28e+03)	3.20e+03[6](e) (5.04e+02)	2.11e+01[2](+) (5.38e+01)	2.32e+01[3](+) (1.06e+01)	8.59e+02[4](+) (1.24e+03)	3.43e+00[1](+) (1.26e+00)	1.09e+03[5](+) (1.79e+03)	5.39e+03[7] (4.16e+03)
YLLF07	3.93e-45	1.62e+00[6](-) (6.91e-01)	2.09e+00[7](-) (9.17e-01)	6.67e-01[4](-) (2.75e-01)	3.31e+00[8](-) (8.29e+00)	2.90e-01[2](+) (9.39e-02)	3.23e-01[3](+) (1.56e-01)	1.71e-01[1](+) (6.09e-02)	5.51e+00[9](-) (1.80e+00)	8.51e-01[5] (2.09e-01)
YLLF08	3.25e-46	5.71e+03[8](-) (3.16e+02)	5.52e+03[6](-) (2.88e+02)	4.66e+03[4](-) (3.32e+02)	7.45e+03[9](-) (1.05e+03)	4.70e+03[5](-) (7.05e+02)	5.67e+03[7](-) (2.83e+02)	3.82e+03[2](-) (4.92e+02)	4.45e+03[3](-) (3.76e+02)	2.36e+03[1] (2.57e+01)
YLLF09	1.41e-39	1.84e+02[8](-) (2.29e+01)	1.02e+02[2](-) (2.09e+01)	1.57e+02[7](-) (1.31e+01)	2.56e+02[9](-) (5.09e+01)	1.34e+02[5](-) (2.06e+01)	1.16e+02[3](-) (5.14e+01)	1.34e+02[4](-) (1.43e+01)	1.37e+02[6](-) (1.57e+01)	0.00e+00[1] (0.00e+00)
YLLF10	9.13e-51	1.61e+01[8](-) (1.90e+00)	1.45e+01[7](-) (1.34e+00)	1.33e+01[6](-) (9.54e-01)	5.36e+00[5](-) (4.86e+00)	1.23e+00[3](-) (5.35e-01)	1.99e+01[9](-) (2.80e-01)	6.46e-01[2](-) (2.26e-01)	3.63e+00[4](-) (5.17e-01)	7.55e-15[1] (0.00e+00)
YLLF11	1.41e-49	5.48e+01[9](-) (1.16e+01)	4.90e+01[8](-) (1.24e+01)	3.14e+01[7](-) (7.17e+00)	1.07e+00[3](-) (6.58e-02)	1.17e+00[4](-) (9.31e-02)	2.98e+01[6](-) (2.00e+01)	1.00e+00[2](-) (2.78e-02)	1.09e+01[5](-) (1.28e+01)	3.27e-01[1] (2.58e-01)
YLLF12	9.19e-44	6.01e+05[7](-) (7.89e+05)	2.46e+06[9](-) (3.04e+06)	6.09e+04[4](-) (8.76e+04)	1.13e+06[8](-) (9.82e+05)	4.29e+05[6](-) (5.69e+05)	1.23e+02[2](-) (5.49e+02)	2.07e+04[3](-) (1.08e+05)	1.33e+05[5](-) (6.57e+05)	1.33e-01[1] (1.53e-01)
YLLF13	1.07e-41	5.47e+06[3](+) (2.96e+06)	1.08e+07[4](+) (8.07e+06)	1.10e+06[2](+) (5.84e+05)	1.14e+12[8](e) (3.99e+12)	4.76e+10[6](e) (3.08e+10)	6.82e+04[1](+) (1.94e+05)	1.01e+10[5](e) (8.08e+09)	7.44e+10[7](e) (3.42e+11)	1.31e+12[9] (2.42e+12)
mean rank		7.24	6.59	5.41	6.71	4.35	4.76	2.71	4.71	2.53
+ / - / ≈		2/14/1	2/14/1	3/13/1	1/15/1	3/12/2	4/11/2	4/12/1	2/12/3	

- Termination condition: The maximum number of FEs is adopted as the termination condition, set at 500 for all instances.
- Population size: Set $N = 30$ for CMA-ES, EDALS and FCPS-CoDE. Set $N = 40$ for SAMSO (default set in PlatEMO [40]). Set $N = 50$ for GPEME, SA-EDA and ESAEDA.
- Parameters of compared algorithms: default setting according to the original version.

In addition, to demonstrate the effectiveness of the ESAEDA components, two variants were designed as follows:

- ESAEDA*: the Powell method was removed during the iterative process.
- ESAEDA**: the VWH model was updated without using unevaluated solutions.

Each algorithm is executed on each test instance for 30 independent runs to overcome randomness. The first step is to assess the independence of the results obtained by the algorithms on each test instance by the Friedman test [41]. Then the Wilcoxon rank sum test [42] is used to compare the results where '+', '-', and '~' in the tables indicate the value obtained by an algorithm is smaller than, greater than,

or similar to that obtained by the ESAEDA based version at a 0.05 significance level. The value in the brackets denotes the corresponding rank.

B. RESULTS AND ANALYSIS

Table 1 presents the statistical findings of 9 optimization algorithms evaluated on two test suites. The results are provided in terms of p-value by the Friedman test, mean ranks, and the corresponding Wilcoxon rank sum test. The p-value obtained from the Friedman test is much less than 0.05, indicating a significant difference between the results. The analysis indicates that ESAEDA achieves the best mean rank of 2.53, outperforming the BO algorithm Skopt, which achieves a mean rank of 6.71. The general evolution algorithms, CMA-ES and EDALS, achieve mean ranks of 7.24 and 3.41, respectively. Among the SAEAs, SA-EDA and GPEME attain the first and second mean ranks in the four comparison algorithms due to the Gaussian process, providing accurate modeling. However, the increased computational overhead associated with these approaches is discussed in the following subsection. The Wilcoxon rank sum test also supports the superiority of ESAEDA as it obtains better optimal solutions than other algorithms in most instances. Importantly, the

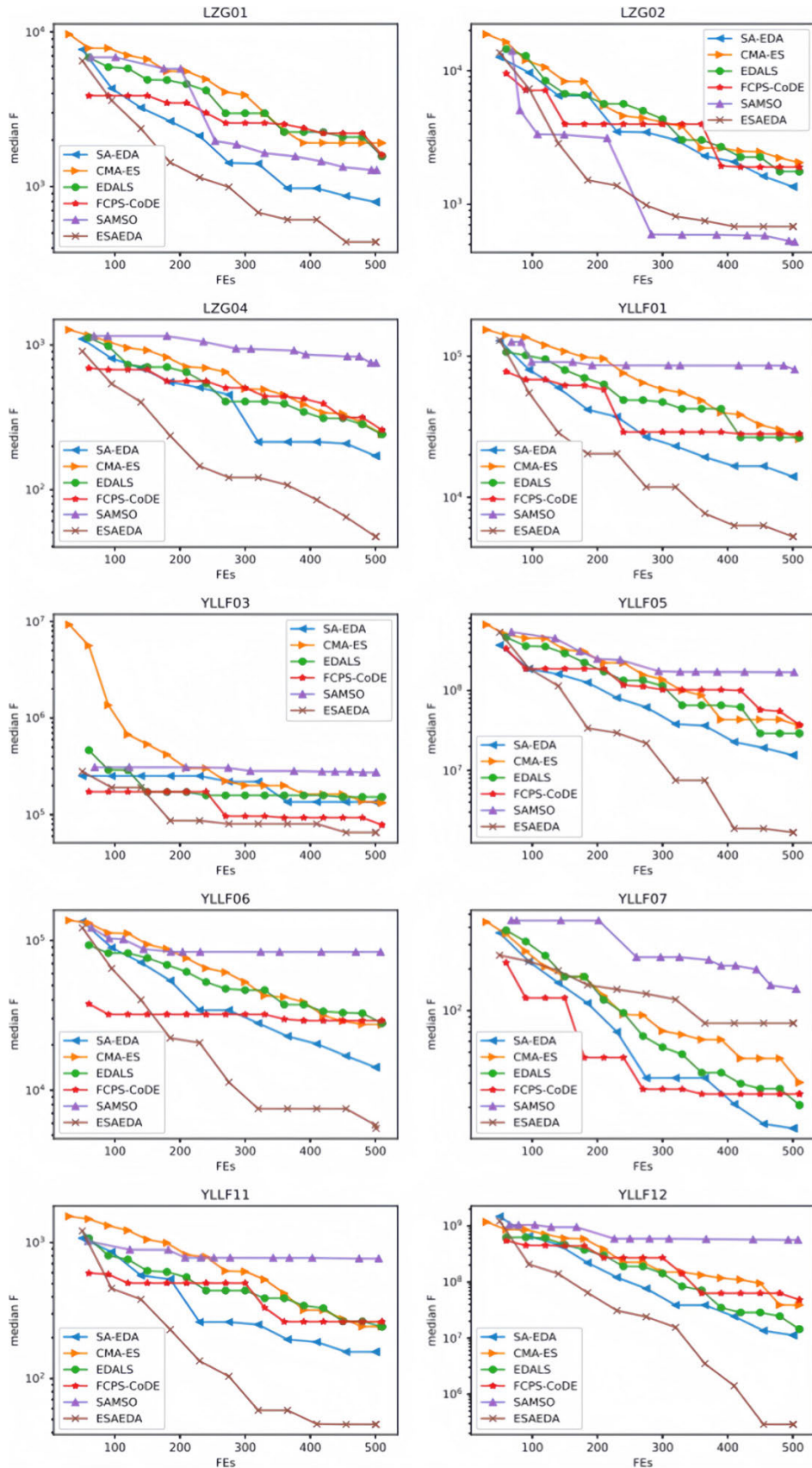


FIGURE 3. The median runtime performance of 6 algorithms on LZG and YLL test suites.

TABLE 2. Statistics of median results obtained by six comparison algorithms on LZG and YLL test suites with $n = 50$.

Problem	p-value	CMA-ES	FCPS-CoDE	EDALS	SAMSO	SA-EDA	ESAEDA
LZG01	7.72e-18	1.91e+03[6](-) (2.77e+02)	1.61e+03[5](-) (3.39e+02)	1.52e+03[4](-) (2.23e+02)	1.31e+03[3](-) (1.13e+03)	8.08e+02[2](-) (1.72e+02)	4.44e+02[1] (6.10e+01)
LZG02	1.08e-11	2.09e+03[6](-) (4.12e+02)	1.90e+03[5](-) (5.11e+02)	1.78e+03[4](-) (2.84e+02)	1.58e+03[3](e) (1.70e+03)	1.33e+03[2](-) (2.86e+02)	7.15e+02[1] (1.43e+02)
LZG03	1.09e-23	1.85e+01[5](-) (9.68e-01)	1.75e+01[3](-) (6.00e-01)	1.76e+01[4](-) (4.24e-01)	1.86e+01[6](-) (1.18e+00)	1.56e+01[2](-) (6.55e-01)	1.01e+01[1] (1.54e+00)
LZG04	3.01e-17	2.38e+02[3](-) (3.80e+01)	2.69e+02[5](-) (6.72e+01)	2.41e+02[4](-) (2.86e+01)	6.19e+02[6](-) (3.66e+02)	1.73e+02[2](-) (2.23e+01)	5.22e+01[1] (2.29e+01)
YLLF01	1.19e-19	2.61e+04[3](-) (3.48e+03)	2.95e+04[5](-) (5.86e+03)	2.77e+04[4](-) (3.92e+03)	6.69e+04[6](-) (3.70e+04)	1.44e+04[2](-) (2.89e+03)	5.85e+03[1] (1.47e+00)
YLLF02	1.72e-28	8.62e+13[5](-) (3.40e+14)	8.92e+01[2](-) (8.24e+00)	2.79e+04[3](-) (9.04e+04)	1.05e+18[6](-) (3.77e+18)	1.64e+07[4](-) (5.33e+07)	4.32e+01[1] (4.28e+00)
YLLF03	2.05e-23	1.36e+05[4](-) (2.57e+04)	8.24e+04[2](-) (1.68e+04)	1.58e+05[5](-) (2.51e+04)	2.97e+05[6](-) (9.31e+04)	1.36e+05[3](-) (2.49e+04)	6.91e+04[1] (1.47e+04)
YLLF04	1.48e-20	9.16e+01[6](-) (1.26e+01)	5.88e+01[3](+) (5.58e+00)	5.80e+01[1](+) (2.78e+00)	8.74e+01[5](-) (8.06e+00)	5.85e+01[2](+) (2.72e+00)	6.94e+01[4] (1.57e+01)
YLLF05	1.58e-23	3.78e+07[4](-) (1.25e+07)	4.49e+07[5](-) (1.82e+07)	2.93e+07[3](-) (5.30e+06)	1.55e+08[6](-) (8.04e+07)	1.52e+07[2](-) (4.04e+06)	3.29e+06[1] (3.32e+06)
YLLF06	8.96e-24	2.82e+04[4](-) (5.82e+03)	2.92e+04[5](-) (7.35e+03)	2.74e+04[3](-) (4.41e+03)	7.82e+04[6](-) (3.06e+04)	1.50e+04[2](-) (3.03e+03)	5.81e+03[1] (1.98e+03)
YLLF07	2.76e-19	3.14e+01[4](+) (9.15e+00)	2.96e+01[3](+) (1.28e+01)	2.17e+01[2](+) (4.75e+00)	1.31e+02[6](-) (7.91e+01)	1.49e+01[1](+) (3.87e+00)	8.26e+01[5] (1.20e+01)
YLLF08	1.91e-19	1.65e+04[5](-) (5.98e+02)	1.63e+04[4](-) (6.11e+02)	1.51e+04[2](+) (5.84e+02)	1.66e+04[6](-) (6.62e+02)	1.38e+04[1](+) (8.40e+02)	1.60e+04[3] (4.64e+02)
YLLF09	1.63e-20	6.21e+02[6](-) (4.52e+01)	3.80e+02[1](+) (3.15e+01)	5.19e+02[5](-) (1.78e+01)	4.97e+02[4](e) (1.11e+02)	4.87e+02[3](e) (2.41e+01)	4.85e+02[2] (2.54e+01)
YLLF10	7.93e-25	1.89e+01[5](-) (1.45e+00)	1.72e+01[3](-) (8.67e-01)	1.75e+01[4](-) (3.88e-01)	1.94e+01[6](-) (2.95e+00)	5.25e+00[2](-) (2.93e-01)	4.51e+00[1] (2.93e-01)
YLLF11	4.20e-18	2.47e+02[4](-) (3.82e+01)	2.52e+02[5](-) (6.26e+01)	2.42e+02[3](-) (2.55e+01)	6.31e+02[6](-) (3.41e+02)	1.61e+02[2](-) (3.08e+01)	4.90e+01[1] (2.18e+01)
YLLF12	2.92e-20	4.07e+07[4](-) (2.32e+07)	5.44e+07[5](-) (3.49e+07)	1.64e+07[3](-) (5.98e+06)	5.10e+08[6](-) (3.16e+08)	1.17e+07[2](-) (4.77e+06)	1.77e+06[1] (5.01e+06)
YLLF13	1.88e-23	1.03e+08[2](+) (3.96e+07)	1.58e+08[3](+) (1.01e+08)	7.53e+07[1](+) (2.19e+07)	8.32e+08[4](+) (6.39e+08)	2.03e+12[6](-) (7.69e+11)	1.05e+12[5] (2.19e+12)
mean rank		4.47	3.76	3.24	5.35	2.35	1.82
+ / - / *		2/15/0	4/13/0	4/13/0	1/14/2	3/13/1	

Powell method applied in ESAEDA* effectively improves the performance of EDA algorithms in low-dimensional problems. The results of Table 1 indicate significant differences among the algorithms. ESAEDA consistently achieved the best mean rank, outperforming the Bayesian optimization algorithm Skopt and other general and surrogate-assisted evolutionary algorithms. Specifically, ESAEDA showcased superior convergence performance and optimal solution quality in most test instances, signifying its effectiveness over conventional and contemporary approaches.

Due to the limitation of Gaussian process-based methods in median dimension space, we selected a subset of algorithms for 50-dimensional problems, including CMA-ES, FCPS-CoDE, EDALS, SAMSO, and SA-EDA. As the Powell method does not apply to ESAEDA in 50 dimensions, the variant ESAEDA* was not included in the experiment. Table 2 presents the statistical results, indicating that ESAED outperforms SAMSO, a specifically designed method for large-scale expensive optimization problems, achieving the best mean of 1.82 under 50 problems. Furthermore, ESAEDA also achieved better results than GP assisted EDA, demonstrating its competitive performance. The Wilcoxon rank sum test supports these results, demonstrating ESAEDA's

superior performance in most instances compared to other algorithms. Figure 3 illustrates the convergence performance of all algorithms, with results consistent with those in Table 2. ESAEDA exhibits the best convergence performance on most test instances.

ESAEDA outperforms several state-of-the-art algorithms by demonstrating superior convergence performance on widely used test instances. It achieves significantly better median results compared to other algorithms. The algorithm exhibits remarkable computational efficiency, achieving 10-90 times efficiency improvements compared to Gaussian Process-assisted algorithms on problems of varying dimensions (20D and 50D). Besides demonstrating competitive performance in tackling expensive optimization problems, ESAEDA holds promise for extending its application to solve expensive multi-objective optimization problems, thereby presenting a versatile and practical approach for real-world scenarios.

The novel aspect of ESAEDA lies in its unique combination of a random forest surrogate model with the estimation of distribution algorithms, which is not commonly found in existing studies. This combination offers a new pathway to reduce computational costs while maintaining solution

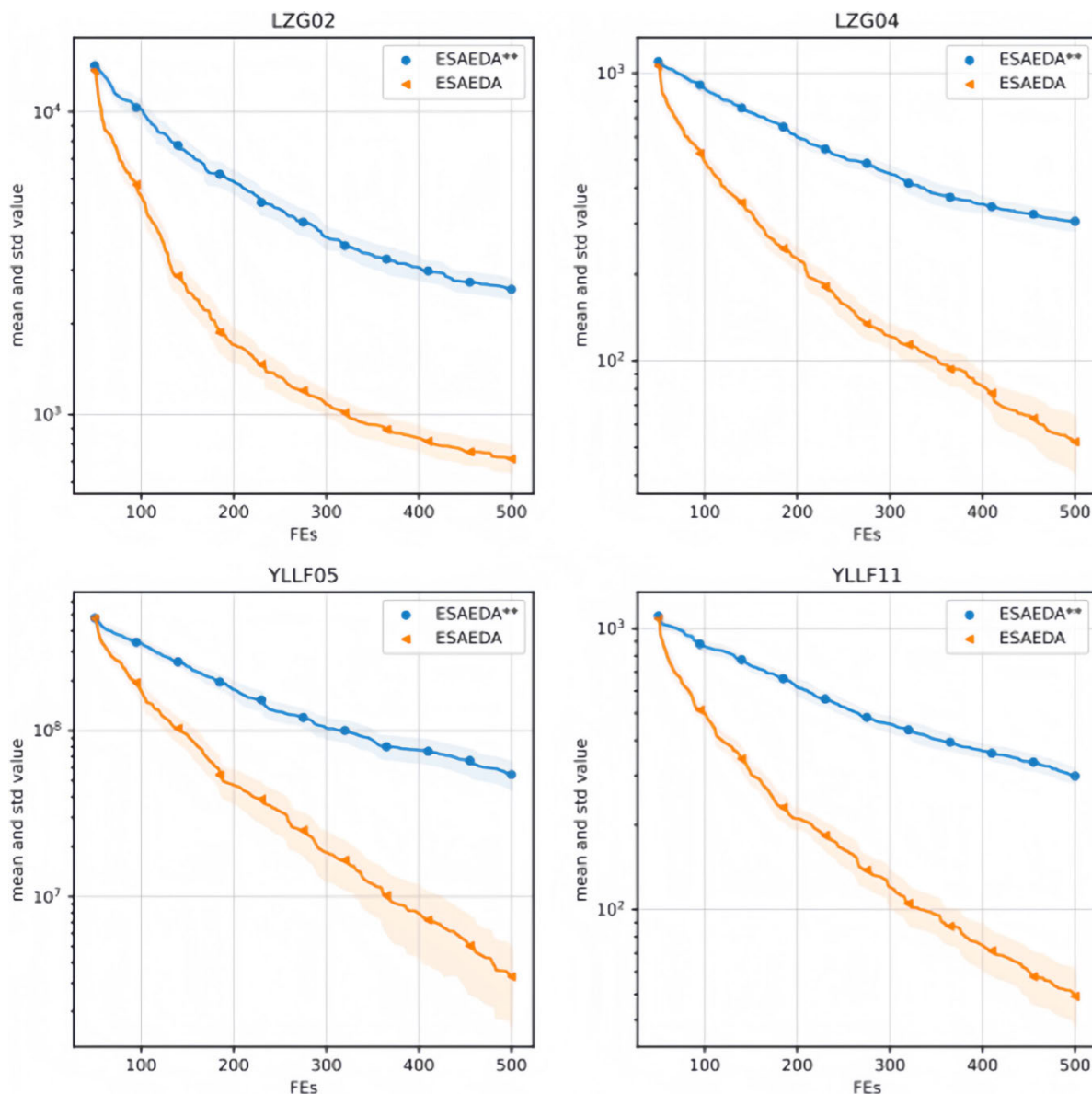


FIGURE 4. The runtime performance of ESAEDA and its variant on LZG and YLL test suites over 30 independent runs.

quality. Additionally, the innovative use of unevaluated solutions to update the probability model represents a significant departure from conventional approaches, aiming to accelerate convergence without excessive function evaluations.

C. ROLES OF ESAEDA

In this subsection, we will analyze the key characteristics of the ESAEDA algorithm. First, we will analyze the benefits of adding unevaluated solutions for the probabilistic model. Second, we will analyze the efficiency improvement of using RF as the surrogate model.

The importance of unevaluated solutions in the update of the probability model in ESAEDA is verified in Figure 4. The figure shows the convergence curves of ESAEDA and its variants on some test instances over 30 independent runs. It is

observed that the convergence of the algorithm can be further improved by incorporating unevaluated solutions in the VWH model updating. These solutions, although not evaluated by the real function, are still valuable as they are selected by the surrogate model. These solutions have a positive impact on the convergence of the population, thereby enhancing the quality of the offspring solutions.

Next, the running time of different algorithms on LZG01 is analyzed. Two Gaussian process based algorithms, SA-EDA and GPME, are selected as a comparator with ESAEDA. All algorithms are run on the same hardware platform (Apple M1pro and 32G Memory). We run each algorithm on the LZG01 under the same function evaluations (fes) for 10 independent runs. The mean CPU time obtained by three algorithms is recorded in Figure 5.

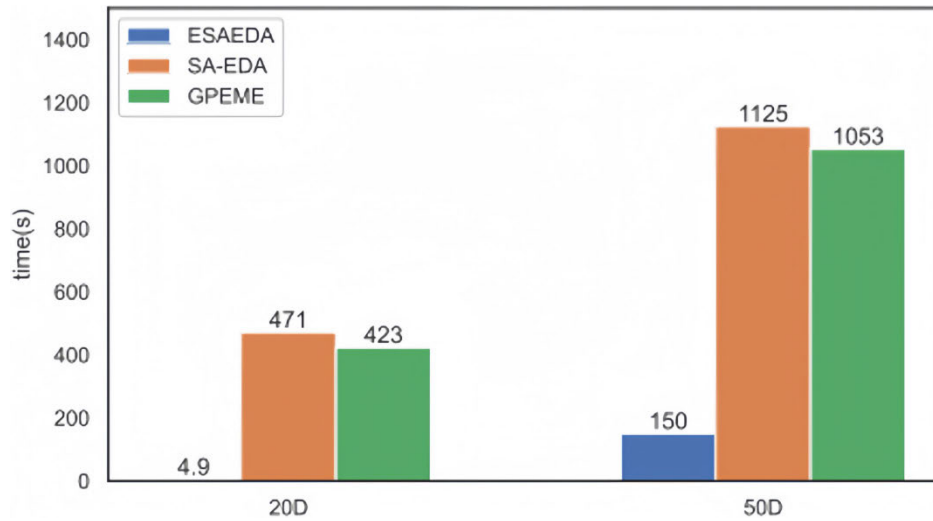


FIGURE 5. Bar plot of running time of ESAEDA, SA-EDA and GPEM on LZG01 under 500 fitness evaluations.

Using the computational lightweight surrogate model (For n samples, the complexity of RF is $\mathcal{O}(n \log(n))$ and the complexity of GPs is $\mathcal{O}(n^3)$), the ESAEDA has the lowest computational overhead among all algorithms tested, when executing the same number of iterations.

The proposed ESAEDA distinguishes itself with unique features. It utilizes RF as the surrogate model, significantly reducing computational demands compared to traditional Gaussian Process-based approaches. This adjustment leads to a more efficient resolution of expensive optimization problems. Additionally, ESAEDA employs unevaluated solutions to update its probabilistic model, thereby improving the generation of new solutions and accelerating convergence. Furthermore, the incorporation of the Powell method enhances search capabilities in low-dimensional spaces, addressing potential declines in search efficiency.

V. CONCLUSION

This paper proposes an efficient surrogate model-assisted estimation of distribution algorithm (ESAEDA) which aims to tackle expensive optimization problems. The use of random forest as a surrogate model instead of Gaussian processes results in significant reductions in computational overhead. The surrogate model works in conjunction with the GP-hedge strategy to identify promising solutions for real evaluations. The improved EDA with unevaluated is used to generate promising solutions. The VWH model focuses on promising areas and accelerating algorithm convergence. Additionally, the enhanced EDA with unevaluated samples is used to generate promising solutions, while the VWH model is utilized to focus on promising areas and accelerate the convergence of the algorithm. Local search and the Powell method are also incorporated to improve the quality of the population.

ESAEDA is benchmarked against several state-of-the-art algorithms on 17 widely used test instances, and the results

indicate that it performs well. Furthermore, the running time of the algorithm is analyzed, and the new method is shown to achieve 10-90 times efficiency improvements compared to Gaussian process assisted algorithms on 20D and 50D problems. This efficiency is a critical advantage in real-world applications where evaluation costs are high. The algorithm's design allows for faster convergence compared to traditional methods and some state-of-the-art algorithms, making it a viable option for expensive optimization problems. Extending ESAEDA to solve expensive multi-objective optimization problems holds promise, and its application to real-world problems is highly desirable.

It is worth noting that the Powell method has been significantly enhanced through integration with contemporary optimization technologies, such as cuckoo search, extending its applicability beyond traditional domains. This synergy has led to more accessible and efficient solutions for various nonlinear optimization issues, as illustrated by initiatives like PRIMA that aim to provide practical, derivative-free optimization frameworks. Moreover, the innovative application of the Powell method in evolutionary algorithms for hyperparameter optimization showcases its extended utility in the realm of machine learning and artificial intelligence. This adapted version, aimed at meeting the specific requirements of hyperparameter tuning in complex computational models, signifies the method's adaptability and potential. In future work, this enhanced Powell method could be applied within ESAEDA to further enhance the method's performance.

The proposed ESAEDA exhibits certain limitations, notably in handling high-dimensional problems due to its reliance on the Powell method, which is primarily effective in low-dimensional scenarios. To address this, future work could explore integrating techniques that enhance high-dimensional search capabilities. Additionally, ESAEDA's performance heavily depends on the accuracy of

the Random Forest surrogate model. An adaptive approach that selects the most suitable surrogate model based on current performance metrics could improve robustness and efficiency across diverse problem landscapes.

REFERENCES

- [1] E. Polak, *Optimization: Algorithms and Consistent Approximations*. Springer, 2012. [Online]. Available: <https://link.springer.com/book/10.1007/978-1-4612-0663-7>
- [2] A. Zhou, J. Zhang, J. Sun, and G. Zhang, "Fuzzy-classification assisted selection preselection in evolutionary optimization," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 2403–2410.
- [3] A. Zhou, J. Sun, and Q. Zhang, "An estimation of distribution algorithm with cheap and expensive local search methods," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 807–822, Dec. 2015.
- [4] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, Jun. 2001.
- [5] H. Hao, J. Zhang, X. Lu, and A. Zhou, "Binary relation learning and classifying for preselection in evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 24, no. 6, pp. 1125–1139, Dec. 2020.
- [6] D. Lim, Y. Jin, Y.-S. Ong, and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 329–355, Jun. 2010.
- [7] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft Comput.*, vol. 9, no. 1, pp. 3–12, Jan. 2005.
- [8] Y. Lian and M.-S. Liou, "Multiobjective optimization using coupled response surface model and evolutionary algorithm," *AIAA J.*, vol. 43, no. 6, pp. 1316–1325, 2005.
- [9] C. Sun, Y. Jin, R. Cheng, J. Ding, and J. Zeng, "Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems," *IEEE Trans. Evol. Comput.*, vol. 21, no. 4, pp. 644–660, Aug. 2017.
- [10] L. Willmes, T. Back, Y. Jin, and B. Sendhoff, "Comparing neural networks and Kriging for fitness approximation in evolutionary optimization," in *Proc. Congr. Evol. Comput. (CEC)*, vol. 1, Dec. 2003, pp. 663–670.
- [11] H. Li, F. He, and X. Yan, "IBEA-SVM: An indicator-based evolutionary algorithm based on pre-selection with classification guided by SVM," *Appl. Math.-J. Chin. Universities*, vol. 34, no. 1, pp. 1–26, 2019.
- [12] L. Pan, C. He, Y. Tian, H. Wang, X. Zhang, and Y. Jin, "A classification-based surrogate-assisted evolutionary algorithm for expensive many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 74–88, Feb. 2019.
- [13] I. Loshchilov, M. Schoenauer, and M. Sebag, "Comparison-based optimizers need comparison-based surrogates," in *Proc. Int. Conf. Parallel Problem Solving Nature*. Berlin, Germany: Springer, 2010, pp. 364–373.
- [14] T. P. Runarsson, "Ordinal regression in evolutionary computation," in *Proc. Int. Conf. Parallel Problem Solving Nature*. Berlin, Germany: Springer, 2006, pp. 1048–1057.
- [15] A. Dushatskiy, A. M. Mendrik, T. Alderliesten, and P. A. Bosman, "Convolutional neural network surrogate-assisted GOMEA," in *Proc. Genetic Evol. Comput. Conf.*, 2019, pp. 753–761.
- [16] H. Hao, A. Zhou, and H. Zhang, "An approximated domination relationship based on binary classifiers for evolutionary multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2021, pp. 2427–2434.
- [17] Y. Jin, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm Evol. Comput.*, vol. 1, no. 2, pp. 61–70, Jun. 2011.
- [18] M. Wang, Z. Ma, Y. Wang, J. Liu, and J. Guo, "A multi-view convolutional neural network method combining attention mechanism for diagnosing autism spectrum disorder," *PLoS ONE*, vol. 18, no. 12, Dec. 2023, Art. no. e0295621.
- [19] B. Liu, Q. Zhang, and G. G. E. Gielen, "A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 2, pp. 180–192, Apr. 2014.
- [20] F. Li, X. Cai, L. Gao, and W. Shen, "A surrogate-assisted multiswarm optimization algorithm for high-dimensional computationally expensive problems," *IEEE Trans. Cybern.*, vol. 51, no. 3, pp. 1390–1402, Mar. 2021.
- [21] N. Hansen, "Towards a new evolutionary computation Advances in the estimation of distribution algorithms," in *Studies in Fuzziness and Soft Computing*, vol. 192. Springer, 2006, pp. 75–102. [Online]. Available: <https://link.springer.com/book/10.1007/3-540-32494-1>
- [22] Y. Cai, H. Zhang, Y. Liu, and Q. He, "Distributed bipartite finite-time event-triggered output consensus for heterogeneous linear multi-agent systems under directed signed communication topology," *Appl. Math. Comput.*, vol. 378, Aug. 2020, Art. no. 125162.
- [23] Q. He, H. Fang, J. Zhang, and X. Wang, "Dynamic opinion maximization in social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 350–361, Jan. 2023.
- [24] Q. He, Z. Feng, H. Fang, X. Wang, L. Zhao, Y. Yao, and K. Yu, "A blockchain-based scheme for secure data offloading in healthcare with deep reinforcement learning," *IEEE/ACM Trans. Netw.*, vol. 32, no. 1, pp. 65–80, Jun. 2024.
- [25] Y. Cai, H. Zhang, Y. Wang, Z. Gao, and Q. He, "Adaptive bipartite fixed-time time-varying output formation-containment tracking of heterogeneous linear multiagent systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 9, pp. 4688–4698, Sep. 2022.
- [26] R. Zhong, J. Yu, C. Zhang, and M. Munetomo, "Surrogate ensemble-assisted hyper-heuristic algorithm for expensive optimization problems," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, p. 169, 2023.
- [27] J.-S. Pan, N. Liu, S.-C. Chu, and T. Lai, "An efficient surrogate-assisted hybrid optimization algorithm for expensive optimization problems," *Inf. Sci.*, vol. 561, pp. 304–325, Jun. 2021.
- [28] Y. Chen, L. Wang, and H. Huang, "An effective surrogate model assisted algorithm for multi-objective optimization: Application to wind farm layout design," *Frontiers Energy Res.*, vol. 11, Sep. 2023, Art. no. 1239332.
- [29] Q. He, X. Wang, Y. Zhao, B. Yi, X. Lu, M. Yang, and M. Huang, "Reinforcement-learning-based competitive opinion maximization approach in signed social networks," *IEEE Trans. Computat. Social Syst.*, vol. 9, no. 5, pp. 1505–1514, Oct. 2022.
- [30] H. Hao, S. Wang, B. Li, and A. Zhou, "A surrogate model assisted estimation of distribution algorithm with multi-acquisition functions for expensive optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2022, pp. 1–8.
- [31] M. Pelikan, D. E. Goldberg, and S. Tsutsui, "Getting the best of both worlds: Discrete and continuous genetic and evolutionary algorithms in concert," *Inf. Sci.*, vol. 156, nos. 3–4, pp. 147–171, Nov. 2003.
- [32] P. Pošík, "On the use of probabilistic models and coordinate transforms in real-valued evolutionary algorithms," Ph.D. thesis, Czech Tech. Univ. Prague, Prague, Czech Republic, 2007.
- [33] H. Wang and Y. Jin, "A random forest-assisted evolutionary algorithm for data-driven constrained multiobjective combinatorial optimization of trauma systems," *IEEE Trans. Cybern.*, vol. 50, no. 2, pp. 536–549, Feb. 2020.
- [34] Y. Sun, H. Wang, B. Xue, Y. Jin, G. G. Yen, and M. Zhang, "Surrogate-assisted evolutionary deep learning using an end-to-end random forest-based performance predictor," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 350–364, Apr. 2020.
- [35] L. Han and H. Wang, "A random forest assisted evolutionary algorithm using competitive neighborhood search for expensive constrained combinatorial optimization," *Memetic Comput.*, vol. 13, no. 1, pp. 19–30, Mar. 2021.
- [36] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.
- [37] R. Fletcher, *Practical Methods of Optimization*. Hoboken, NJ, USA: Wiley, 2000.
- [38] M. Hoffman, E. Brochu, and N. De Freitas, "Portfolio allocation for Bayesian optimization," in *Proc. UAI*, 2011, pp. 327–336.
- [39] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, vol. 3, no. 2, pp. 82–102, Jul. 1999.
- [40] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [Educational Forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017.
- [41] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Stat. Assoc.*, vol. 32, no. 200, pp. 675–701, 1937.
- [42] M. Hollander, D. A. Wolfe, and E. Chicken, *Nonparametric Statistical Methods*. Hoboken, NJ, USA: Wiley, 2013.



JIN SHANG was born in Harbin, Heilongjiang, China, in 1974. He received the Ph.D. degree in control engineering from Harbin University of Science and Technology, in 2012. From 2014 to 2023, he was an Associate Professor with Wuxi Institute of Technology. His research interests include intelligent optimal and the study of multi-objective optimization algorithms.



HAO HAO received the B.Eng. degree in electrical engineering and automation from Heilongjiang University, Harbin, Heilongjiang, China, in 2017, and the Ph.D. degree from the School of Computer Science and Technology, East China Normal University, in 2022. Currently, he is conducting postdoctoral research with the Institute of Natural Science, Shanghai Jiao Tong University.



GUIYING LI was born in Jingle, Shanxi, China, in 1974. She received the Ph.D. degree from Northeast Forestry University, China. She is currently with the School of Mechanical and Electrical Engineering, Heilongjiang University. Her research interests include intelligent optimal and the study of multi-objective optimization algorithms.



YUFANG ZHANG received the Ph.D. degree from Harbin Engineering University. She is currently with the Department of Robotic Control System, Wuxi Institute of Technology, China. Her current research includes adaptive and nonlinear control in the field of industrial control.

...