# Impact of Surrogate Model Accuracy on Performance and Model Management Strategy in Surrogate-Assisted Evolutionary Algorithms

Yuki Hanawa<sup>a,\*</sup>, Tomohiro Harada<sup>b</sup>, Yukiya Miura<sup>c</sup>

<sup>a</sup> Graduate School of Systems Design, Tokyo Metropolitan University, Asahigaoka
 6-6, Hino, 191-0065, Tokyo, Japan
 <sup>b</sup> Graduate School of Science and Engineering, Saitama University, 255, Sakura-ku
 Shimo-okubo, Saitama, 338-8570, Saitama, Japan
 <sup>c</sup> Faculty of Systems Design, Tokyo Metropolitan University, Asahigaoka
 6-6, Hino, 191-0065, Tokyo, Japan

#### Abstract

Surrogate-assisted evolutionary algorithms (SAEAs) have been proposed to solve expensive optimization problems. Although SAEAs use surrogate models that approximate the evaluations of solutions using machine learning techniques, prior research has not adequately investigated the impact of surrogate model accuracy on search performance and model management strategy in SAEAs. This study analyzes how surrogate model accuracy affects search performance and model management strategies. For this purpose, we construct a pseudo-surrogate model with adjustable prediction accuracy to ensure fair comparisons across different model management strategies. We compared three model management strategies: (1) pre-selection (PS), (2) individual-based (IB), and (3) generation-based (GB) on standard benchmark problems with a baseline model that does not use surrogates. The experimental results reveal that a higher surrogate model accuracy improves the search performance. However, the impact varies according to the strategy used. Specifically, PS demonstrates a clear trend of improved performance as the estimation accuracy increases, whereas IB and GB exhibit robust performance when the accuracy surpasses a certain threshold. In model strategy

Email addresses: hanawa-yuki@ed.tmu.ac.jp (Yuki Hanawa), tharada@mail.saitama-u.ac.jp (Tomohiro Harada), miura@tmu.ac.jp (Yukiya Miura)

<sup>\*</sup>Corresponding author

comparisons, GB exhibits superior performance across a broad range of prediction accuracies, IB outperforms it at lower accuracies, and PS outperforms it at higher accuracies. The findings of this study clarify guidelines for selecting appropriate model management strategies based on the surrogate model accuracy.

Keywords: surrogate-assisted evolutionary algorithms, prediction accuracy, performance analysis, model management strategy

#### 1. Introduction

Optimization is the process of maximizing or minimizing an objective (evaluation) function under specific constraints. Optimization problems are expressed using the following mathematical formula:

minimize 
$$f(\mathbf{x})$$
  
subject to  $g_i(\mathbf{x}) \leq 0$   $(i = 1, ..., n)$   
 $h_i(\mathbf{x}) = 0$   $(j = 1, ..., p)$ 

where  $\boldsymbol{x}$  represents the design variables, f is the objective function, and  $g_i$  and  $h_j$  are the inequality and equality constraint functions, respectively, and n and p indicate the number of each type of constraint.

Evolutionary algorithms (EAs) are a class of metaheuristics that solve optimization problems by mimicking the mechanism of natural evolution to search for optimal solutions [10, 15]. EAs have demonstrated high search performance, particularly in high-dimensional spaces and problems with complex constraints. However, a challenge arises when the computational cost of the evaluation function is high (e.g., in problems requiring simulations or complex numerical computations). These problems are known as expensive optimization problems (EOPs) [13], which require extensive computation time because of the large number of evaluations required during the optimization process.

Surrogate-assisted evolutionary algorithms (SAEAs) [3, 5, 6] have attracted attention for addressing this challenge. SAEAs replace expensive solution evaluations with low-cost approximation models (surrogate models) using machine learning (ML) techniques. By employing surrogate models, SAEAs could achieve acceptable optimization results within a realistic execution time, leveraging low-cost approximated evaluations with surrogate models to explore solutions based on past histories.

Owing to the use of surrogate models, their prediction accuracy of surrogate models affects the search performance of SAEAs [20]. Specifically, an accurate surrogate model allows for more precise predictions of evaluation values, reduces the number of actual evaluation functions, and shortens the overall execution time. However, less accurate surrogate models might lead to erroneous search directions, yet they could enhance search performance by facilitating escape from the local optima.

From this perspective, the research questions addressed in this study are:

**RQ1:** How does the prediction accuracy of surrogate models impact the search performance of SAEAs?

**RQ2:** Do different SAEA model management strategies vary in their sensitivity to surrogate model accuracy?

**RQ3:** Could the optimal SAEA model management strategy be determined based on surrogate model accuracy?

To answer these research questions, we constructed a pseudo-surrogate model with an adjustable prediction accuracy using actual evaluation functions. We compared the search performance of SAEAs using pseudo-surrogate models with varying accuracies.

This study addresses three SAEA model management strategies: (1) preselection (PS), (2) individual-based (IB), and (3) generation-based (GB). We compared the combinations of these three strategies and pseudo-surrogate models of varying accuracy on CEC 2015 benchmark problems [1] to evaluate the search performance under a limited evaluation budget. In this study, we aim to provide guidelines for selecting appropriate model management strategies based on surrogate model accuracy, thereby enhancing the optimization performance of SAEAs.

In our previous study, [2] conducted a preliminary analysis of the relationship between the prediction accuracy of surrogate models and the SAEA strategies. However, this early work used only two strategies (PS and IB) with different surrogate representations, which restricted the direct comparison of model management strategies under consistent accuracy metrics. In contrast, this study incorporated a new SAEA strategy, GB, in addition to PS and IB, and used a common pseudo-surrogate model. This improvement enabled consistent comparisons across all strategies, allowing us to answer the research questions more precisely and comprehensively.

#### 2. SAEAs

Over the past several decades, SAEAs employing various surrogate models have been proposed [3, 5, 6]. This section provides an overview of SAEAs. Then, we explain three representative model management strategies: PS, IB, and GB, based on the classifications provided in previous studies [19, 4].

# 2.1. Overview of SAEAs

SAEA is a method that reduces computational costs by using surrogate models, which are generally constructed using ML techniques (e.g., the radial basis function, Kriging model, and neural networks). Specifically, during the optimization process, SAEA constructs a surrogate model from the information on solution evaluations obtained from the past search history. Then, SAEA uses the constructed surrogate model to predict the fitness values of the solutions and determine the solutions to be evaluated using an actual expensive evaluation function. Only solutions predicted to be promising undergo a detailed evaluation with the actual evaluation function; otherwise, less promising solutions are discarded. This approach enables the omission of actual evaluations for less promising solutions, thereby reducing the overall computational costs.

# 2.2. PS SAEA

The PS strategy uses a surrogate model to select superior individuals before performing actual evaluations. In general, the PS strategy generates several offspring. Then, the surrogate model selects the most promising individual. Subsequently, the selected individuals undergo an actual evaluation. This approach allows superior individuals to be retained more easily, thereby reducing computational costs. Examples of SAEAs using PS include SAC-CDE [21] and CHDE+ELDR [7] for constrained optimization and KTA2 [16] and MCEA/D [17] for multi-objective optimization.

PS could employ several types of surrogate models, including absolute, rank-based, and classification-based surrogates. In a PS with an absolute evaluation surrogate model, multiple offspring candidates are generated, and their evaluation values are predicted. Then, the superior solutions are preselected and evaluated. In a PS with a rank-based surrogate model, the rankings of the generated offspring are predicted, and those with higher ranks are preselected for evaluation. In PS with a classification-based surrogate model, the superiority or inferiority of the solutions compared to their parent models

**Algorithm 1** A pseudocode of the pre-selection strategy SAEA (PS) used in this study [19]

```
1: Create the initial sample of 5 \times d individuals using Latin Hypercube
   Sampling (LHS) [9]
2: Evaluate all initial individuals and store them in an archive \mathcal{A}
3: Set FE = 5 \times d
 4: Select the best N individuals from \mathcal{A} for the population P
 5: while FE < maxFE do
      Perform crossover and mutation operators to generate N offspring
6:
      for each offspring of f do
 7:
        Find its parent par as a reference individual
8:
9:
        Build a surrogate S using A
        Predict a label whether of f is better than par using S
10:
        if predictive label is true then
11:
           Evaluate the offspring with an actual evaluation function
12:
           Add all actual evaluated offspring to \mathcal{A}
13:
           FE = FE + 1
14:
           if of f is superior to par then
15:
             Replace par with of f in the population P
16:
           end if
17:
        end if
18:
      end for
19:
20: end while
```

is learned. If the offspring are predicted to be superior, they are evaluated; otherwise, they are rejected without actual evaluation. If the actual evaluation is better than that of the parent, the parent replaces it; otherwise, the parent remains in the next generation.

Algorithm 1 presents the pseudocode of PS with the classification-based surrogate model employed in this study. Where d denoted the dimensions of the design variables, N denoted population size, and maxFE denoted the maximum number of evaluations. All evaluated individuals were stored in an archive  $\mathcal{A}$  and used for surrogate construction. In PS, offspring were generated through crossover and mutation operators (Line 6). Subsequently, a classification-based surrogate model  $\mathcal{S}$  was constructed for each offspring using its parent as a reference individual (Lines 8–9). Then, the surrogate model predicted the superiority of the offspring over its parent individual

(Line 10). If it was predicted to be superior, the offspring was evaluated using the actual evaluation function (Line 12). After the actual evaluation was confirmed, it was compared with its parent and replaced if it was superior to its parent (Lines 14–16). Conversely, if offspring were predicted to be inferior, they were rejected without applying an actual evaluation function.

#### 2.3. IB SAEA

The IB strategy uses a surrogate model to determine whether each individual offspring should undergo actual evaluation. Generally, the IB strategy selects offspring that are predicted to be the most promising based on the evaluation values predicted by the surrogate model. Then, the selected offspring were evaluated using an actual evaluation function. This approach reduces the number of evaluations by excluding individuals who are unlikely to perform well. Examples of SAEAs that use IB include GPEME [8], VESAEA [18], and RFMOISR [11].

IB could be combined with absolute fitness and rank-based surrogates. In IB with an absolute fitness surrogate model, the evaluation values of the newly generated offspring are predicted, and the offspring with higher predicted values are evaluated. Finally, the next population was selected based on actual and predicted evaluation values. IB with a rank surrogate model predicts ranking within the offspring population and performs actual evaluations for higher-ranked individuals until the correlation with the training data reaches a certain level. Once sufficient correlation is achieved, the top-ranked individuals are selected as the next generation's population.

Algorithm 2 presents the pseudocode for IB used in this study. In IB, offspring are generated through crossover and mutation and are sorted using a surrogate model (Line 8). Then, the top  $r_{sm} \times N$  (0 <  $r_{sm}$  < 1) individuals are re-evaluated using the actual evaluation function, and the top N individuals from both parents and offspring are selected for the next generation's population (Lines 8–11). During this process, unevaluated individuals were compared based on their predicted evaluation values.

# 2.4. GB SAEA

The GB strategy [5] evolved the population using only a surrogate model for specific generations, without using an actual evaluation function during this period. After completing a set number of generations, the most promising individual in the final population was evaluated using the evaluation function. Examples of SAEAs using GB include SAHO [14], LSA-FIDE [23],

**Algorithm 2** A pseudocode of the individual-based strategy (IB) used in this study [19]

- 1: Create the initial sample of  $5 \times d$  individuals using Latin Hypercube Sampling (LHS)
- 2: Evaluate all initial individuals and store them in an archive  $\mathcal{A}$
- 3: Set  $FE = 5 \times d$
- 4: Select the best N individuals from  $\mathcal{A}$  for the population P
- 5: while FE < maxFE do
- 6: Perform crossover and mutation operators to generate N offspring
- 7: Build a surrogate S using A
- 8: Sort the parent and offspring individuals using  $\mathcal{S}$
- 9: Evaluate  $r_{sm} \times N$  best predicted individuals with actual function
- 10:  $FE = FE + r_{sm} \times N$
- 11: Select N best individuals from parents and offspring for the next generation
- 12: end while

and GORS-SSLPSO [22]. GB could be combined with two types of surrogates: absolute fitness- and rank-based. With either type of surrogate, the search continues for a certain number of generations while performing parent and survival selection based on surrogate-based fitness values.

Algorithm 3 shows the pseudocode for GB, which is addressed in this study. Here, gen represents the generation count within the loop, and maxGen represents the number of generations required to evolve the population using only a surrogate model. In GB, candidate solutions are evaluated for a certain number of generations using only the surrogate while repeating crossover and mutation operators (Lines 7–13). After a specific number of generations, the individual with the best fitness value predicted by the surrogate model underwent an actual evaluation, and the evaluation count was updated (Lines 14–16). The surrogate was retrained using an archive that included the newly evaluated individual.

# 3. Pseudo-surrogate Models

To enable performance analysis of SAEAs using surrogate models with arbitrary prediction accuracy, this study proposes a pseudo-surrogate model. The pseudo-surrogate model determines its predictions by directly using ac**Algorithm 3** A pseudocode of the generation-based strategy (GB) used in this study [6]

- 1: Generate the initial sample of  $5 \times d$  individuals using Latin Hypercube Sampling (LHS)
- 2: Evaluate all initial individuals and store them in an archive  $\mathcal{A}$
- 3: Set  $FE = 5 \times d$
- 4: Select the best N individuals from  $\mathcal{A}$  for the population P
- 5: while FE < maxFE do
- 6: gen = 0
- 7: while gen < maxGen do
- 8: Perform crossover and mutation operators to generate N offspring
- 9: Build a surrogate S using A
- 10: Sort the parent and offspring individuals using  $\mathcal{S}$
- 11: qen = qen + 1
- 12: Select N best individuals from parents and offspring for the next generation
- 13: end while
- 14: Evaluate best predicted individuals with an actual function
- 15: Add an actual evaluated individual to  $\mathcal{A}$
- 16: FE = FE + 1
- 17: Select N best individuals from parents and offspring for the next generation
- 18: end while

tual evaluation values rather than by learning from past history. The pseudosurrogate model allows for adjustable prediction accuracy. To enable adjustment of surrogate model accuracy in PS, IB, and GB, the actual evaluation function was used directly to replicate the surrogate models artificially. Note that for all models, the actual evaluations used in the pseudo-surrogate are not included in the number of actual evaluations (FE) within the algorithm.

The pseudocode for the pseudo-surrogate model is presented in Algorithm 4. In Algorithm 4, sp indicated the surrogate prediction accuracy and was an adjustable parameter. rand(0,1) returned a random value in the range [0,1]. In the pseudo-surrogate model, two solutions,  $x_1$  and  $x_2$ , were evaluated and compared (Lines 1–2). First, the actual superiority was set to the variable label and reversed with a certain probability of (1-sp) to artificially replicate the prediction error (Lines 3–5).

**Algorithm 4** A pseudo-surrogate model to compare two individuals  $x_1$  and  $x_2$ 

```
1: Evaluate x_1 and x_2 with an actual evaluation function
```

```
2: label \leftarrow (f(x_1) < f(x_2))
```

- 3: **if** rand(0,1) < (1-sp) **then**
- 4: Flip *label*
- 5: end if
- 6: return label

In the PS strategy, the pseudo-surrogate model shown in Algorithm 4 is used in Lines 9–10 of Algorithm 1, which simply replaces a surrogate model S.

For the IB and GB strategies, a pseudo-surrogate model was used to sort the populations. Algorithm 5 is a sorting algorithm that uses a pseudo-surrogate model. In Algorithm 5, P denoted the population to be sorted. This sorting algorithm was based on bubble sorting, in which a comparison of two individuals was performed as follows:

- If both individuals were actually evaluated, their actual evaluations were compared without error (Lines 4–7).
- If either individual was unevaluated, they were compared based on the predictions of the pseudo-surrogate model with a prediction accuracy of sp (Lines 8–12).

Note that the actual evaluation values used in this sort exclude those evaluated specifically for the pseudo-surrogate model and refer only to those obtained during the algorithm execution procedure. Algorithm 5 is used in Lines 7–8 of Algorithm 2 for IB, while it is used in Lines 9–10 of Algorithm 3 for GB.

The use of this pseudo-surrogate model enables the reproduction of SAEA behavior with a surrogate model of arbitrary accuracy. In addition, it ensures fair comparisons between model management strategies because all strategies use the same pseudo-surrogate model (see Algorithm 3).

# 4. Experiments

To investigate the impact of the surrogate model accuracy on the search performance of SAEAs, we conducted experiments using a pseudo-surrogate

# Algorithm 5 Sorting algorithm with the pseudo-surrogate model

```
1: n = |P|
2: for i = 1 to n do
3:
      for j = 1 to n - i do
        if Both individuals evaluated with an actual fitness function then
4:
           if f(P_i) > f(P_{i+1}) then
5:
             swap P_i and P_{i+1}
6:
           end if
 7:
        else
8:
           if The pseudo-surrogate model (Algorithm 4) returns false then
9:
             swap P_j and P_{j+1}
10:
11:
           end if
        end if
12:
      end for
13:
14: end for
```

model. The following subsection explains the compared SAEA strategies and presents the experimental setup. Finally, the benchmark functions used in the experiment are detailed.

# 4.1. Compared Strategies

For the SAEA methods, this experiment employed three model management strategies: PS, IB, and GB, with the pseudo-surrogate model explained in the previous section. In addition, a method without a surrogate model (NoS) was compared as a baseline. Algorithm 6 presented the pseudocode for the NoS. The flow of the NoS differed based on the compared algorithms. When comparing NoS with PS (Lines 7–14), all the generated offspring were always evaluated with an actual evaluation. If an offspring was superior to its parent, it replaced the parent individual for use in the next generation's population. However, when comparing NoS with IB and GB (Lines 16–19), all generated offspring were evaluated using an actual evaluation function, and the top N individuals from both parents and offspring were selected for the next generation. In addition to not using a surrogate, other procedures were the same as those used for PS, IB, and GB.

# 4.2. Test Problems

This experiment used a variety of test problems from the CEC2015 competition [1]; in particular, f1, f2, f4, f8, f13, and f15 were used. Table 1

**Algorithm 6** A pseudocode for a method without a surrogate model (NoS) used in the experiment [19]

```
1: Create the initial sample of 5 \times d individuals using Latin Hypercube
   Sampling (LHS)
2: Evaluate all initial individuals
3: Set FE = 5 \times d
 4: Select the best N individuals from the sample for the population P
5: while FE < \max FE do
      Perform crossover and mutation operators to generate N offspring
      /***PS***/
7:
     for each offspring of f do
8:
9:
        Find its parent par as the reference individual
        Evaluate the offspring
10:
        if f(off) < f(par) then
11:
          Replace par with of f in P
12:
        end if
13:
     end for
14:
15:
      /***IB and GB***/
16:
17:
      Evaluate all offspring
18:
     Select N best individuals from parents and offspring for the next gen-
     eration
19: end while
```

presents the fitness landscapes, optimal values, and dimensions. The objective functions f1 and f2 were unimodal, f4 and f8 were multimodal, and f13 and f15 combined the features of both the unimodal and multimodal functions. In this experiment, the dimensionality of the design variables was set to 10 and 30, respectively.

#### 4.3. Experimental Setup

The population size was set to 40. The extended intermediate crossover (EIX) [12] was employed as a crossover operator. First, the EIX selected two parent individuals from the population based on the crossover rate. Then, for each *i*-th variable of the offspring  $t_i$ , an arbitrary value  $\alpha_i$  was randomly chosen between  $\gamma$  and  $1+\gamma$ , and the *i*-th variable of the offspring was determined

Table 1: Details of benchmarks and optimal values [19]

Problem	Landscape	Optimal Value $(f^*)$	Dimensions		
f1	Unimodal	100			
f2	Ommodai	200			
f4	Circula Multimadal	400	10. 20		
f8	Simple Multimodal	800	10, 30		
f13	Composition Dynation	1300			
f15	Composition Function	1500			

using the following formula:

$$t_i = \alpha_i v_i + (1 - \alpha_i) w_i \tag{1}$$

where  $v_i$  and  $w_i$  represented the *i*th design variables of the two parent individuals. In this study,  $\gamma$  was set to 0.4, in accordance with [19]. As a mutation operator, we used uniform mutation within the upper and lower limits of the design variables. The crossover rate pc was set to 0.7, whereas the mutation rate pm was set to 0.3 following a previous study. For each method, the actual evaluation ratio rsm for IB was set as 0.5, whereas maxGen for GB was set as 30. The prediction accuracy of the pseudo-surrogate model was set as sp = 0.5, 0.6, 0.7, 0.8, 0.9, 1.0. When sp = 1.0, no prediction error occurred, and all comparison results were always correct.

In these experiments, the maximum number of evaluations was limited to 2000, within which the progression of the best-found solution during the search process was analyzed. We conducted 21 independent trials for each problem and calculated their averages.

#### 5. Results and Discussion

The experimental results are presented in this section. Section 5.1 analyzed the relationship between the search performance and prediction accuracy to answer RQ1 for each model management strategy. To answer RQ2, Section 5.2 compared the sensitivities of the three strategies to different surrogate model accuracies. Finally, to answer RQ3, Section 5.3 compared the search performance across the three strategies for each prediction accuracy and revealed the optimal strategy for each accuracy.

Experiments were conducted with dimensions of 10 and 30. However, the trends observed for the 10 and 30 dimensions were similar, and including both would make the study unnecessarily lengthy. Therefore, this study presented the results for 30 dimensions in the following subsection, owing to page limitations. The results for the 10 dimensions are provided in the Supplementary Material.

## 5.1. Relation between Search Performance and Prediction Accuracy

First, we analyzed the relationship between search performance and prediction accuracy for PS, IB, and GB.

Figures 1–3 show the results of solving 30-dimensional problems using PS, IB, and GB, respectively. The horizontal axis represented the number of actual evaluations, whereas the vertical axis represented the difference between the optimal objective function value and that obtained using the algorithms. Different colors indicated the difference in the accuracy of the pseudo-surrogate model, whereas the black line indicated the result of the NoS. In the following subsections, we discuss the results of each model management strategy and conclude with an analysis of the correlation between the model accuracy and search performance.

#### 5.1.1. PS

Figures 1a and 1b show the results for unimodal problems. For f1, the smallest objective function value was obtained with an accuracy of 1.0. In addition, after 200 evaluations, an accuracy of 1.0 constantly obtained the smallest objective function value. In addition, an accuracy of 0.5 was comparable to that of NoS, resulting in the worst search performance. Similarly, for f2, the search performance was the best, with an accuracy of 1.0. In addition, the accuracies of 0.5 and 0.6 were comparable to those of the NoS, resulting in the worst search performance.

Figures 1c and 1d show the results for multimodal problems. For f4, an accuracy of 1.0 achieved the smallest objective function value, and this trend was observed throughout the entire search process. In addition, an accuracy of 0.5 was comparable to that of the NoS, resulting in the poorest search performance. For f8, an accuracy of 1.0 achieved the smallest objective function value. In addition, an accuracy of 0.5 resulted in the lowest search performance.

Figures 1e and 1f present the composite function results. Similar to the other functions, for both f13 and f15, an accuracy of 1.0 achieved the small-

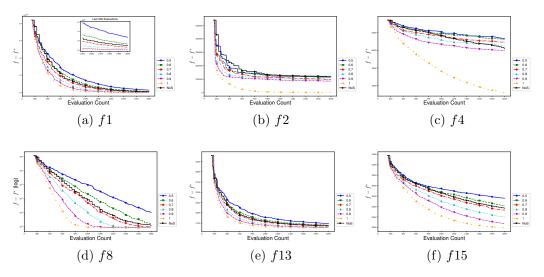


Figure 1: Transition of the difference between the optimal objective function value and the value obtained by algorithms when solving 30D problems using PS

est objective function value. In addition, an accuracy of 0.5, which was comparable to that of the NoS, led to the worst search results.

#### 5.1.2. IB

Figures 2a and 2b show the results for unimodal problems. For f1, the smallest objective function value was obtained with an accuracy of 1.0. In addition, with the exception of an accuracy of 0.8, higher accuracy indicated better search performance. However, for f2, the best performance was obtained with an accuracy of 0.9. However, an accuracy of 1.0

Figures 2c and 2d show the results for multimodal problems. For f4, an accuracy of 0.9 achieved the smallest objective function value, and this trend was observed throughout the entire search process. NoS exhibited the worst search performance. With a surrogate model, an accuracy of 0.5 showed the worst search performance. For f8, an accuracy of 1.0 achieved the smallest objective function value. In contrast, the NoS exhibited the worst search performance, followed by an accuracy of 0.5.

Figures 2e and 2f present the composite function results. For f13, an accuracy of 1.0 achieved the smallest objective function value, followed by an accuracy of 0.8. NoS exhibited the worst search performance. For f15, accuracies of 0.9 and 0.7 achieved small objective function values in that

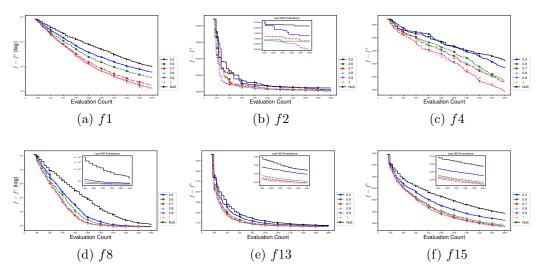


Figure 2: Transition of the difference between the optimal objective function value and the value obtained by algorithms when solving 30D problems using IB

order. In contrast, NoS demonstrated the poorest search performance, with an accuracy of 0.5 being better but still suboptimal.

#### 5.1.3. GB

Figures 3a and 3b show the search results for unimodal problems. For f1, the smallest objective function value was obtained with an accuracy of 1.0. Next, an accuracy of 0.8 had good search performance. In addition, the search performance with an accuracy of 0.5 was worse than that of NoS. However, for f2, the search performance was the best, with an accuracy of 0.8, and the second-best performance was achieved with an accuracy of 0.9. Furthermore, although it had an accuracy of 0.5, it could search for a better objective function value than the NoS at the maximum number of evaluations, and the NoS outperformed an accuracy of 0.5, up to approximately 700 evaluations.

Figures 3c and 3d show the results for multimodal problems. For f4, an accuracy of 0.9 achieved the smallest objective function value. In addition, the search performance with an accuracy of 0.5 was worse than that of NoS. For f8, an accuracy of 0.9 achieved the smallest objective function value. An accuracy of 0.5 outperformed NoS to about 300 evaluations, but after that, NoS exceeded the search performance with an accuracy of 0.5.

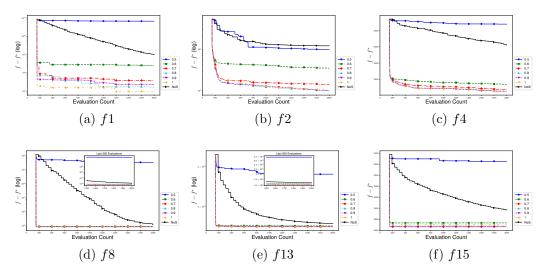


Figure 3: Transition of the difference between the optimal objective function value and the value obtained by algorithms when solving 30D problems using GB

Figures 3e and 3f present the composite function results. For f13, an accuracy of 0.9 achieved the smallest objective function value, followed by an accuracy of 1.0. For f15, an accuracy of 0.9 achieved small values. For f13 and f15, an accuracy of 0.5 was better than NoS until about 200 evaluations, but after that, NoS outperformed an accuracy of 0.5.

#### 5.1.4. Correlation between Accuracy and Search Performance

Table 2 shows Kendall's rank correlation coefficient between the accuracy of the pseudo-surrogate model and the objective function value after 2,000 evaluations. The coefficient ranged from -1 to 1, where a value close to -1 indicated a negative correlation, which meant that a higher prediction accuracy reduced search performance in our experiment. However, a value close to one indicated a positive correlation, where higher accuracy resulted in better search performance. A coefficient of zero implied that there was no correlation.

For PS, the results showed that higher accuracy led to better search performance across all functions in both 10 and 30 dimensions. For IB in 10 dimensions, the results on composite functions showed positive coefficients above 0.7, whereas those on unimodal and multimodal functions had coefficients below 0.6, indicating weak positive correlations. However, in 30

Table 2: Kendall's rank correlation coefficient between the accuracy of the pseudosurrogate model and the objective function value after 2,000 evaluations

		f1	f2	f4	f8	f13	f15
PS .	10D	1.00	1.00	1.00	1.00	1.00	1.00
				1.00			
IB	10D	0.47	0.33	0.60	0.33	0.93	0.73
				0.87			
GB .	10D	0.73	0.87	0.89	0.65	0.97	1.00
				0.87			

dimensions, although f15 showed a low coefficient of 0.33, the other cases demonstrated correlations greater than 0.6. For GB in 10 dimensions, although f8 showed a low value of 0.65, compared with other GB cases in 10 dimensions, positive correlations were observed overall. In 30 dimensions, while f15 showed a low coefficient of 0.47 compared with the other functions, the other functions exhibited positive correlations above 0.6.

#### 5.1.5. Summary

From these results, we summarized this tendency as follows:

- When using PS, the accuracy of surrogate models and search performance were fully correlated, and higher prediction accuracy consistently led to better performance.
- When using IB or GB, the accuracy and search performance showed a positive correlation. However, the highest accuracy did not always result in the best search performance, and lower accuracies might outperform higher accuracies.

From Algorithm 1, we could observe that in PS, offspring inferior to their parents were not selected for the next generation because if an inferior offspring was predicted to be superior to its parent, the actual evaluation revealed it to be inferior; therefore, it was rejected. Therefore, the low accuracy of PS did not contribute to increased diversity.

Conversely, from Algorithms 2 and 3, individuals with inferior evaluation values might be carried over to the next generation during sorting in IB and

GB. This was because in Algorithm 5, when one of the two individuals had not been evaluated with an actual evaluation function, it could be rearranged in the wrong order based on the surrogate accuracy. Owing to this characteristic, low accuracy might lead to increased diversity, potentially resulting in better search performance compared to higher accuracies.

## 5.2. Sensitivity of Model Management Strategies to Prediction Accuracy

This section analyzed the sensitivity of the model management strategies to the prediction accuracy. Specifically, we confirmed a significant difference between the different prediction accuracies for each strategy by using Tukey's HSD test.

Figures 4–6 show the results of Tukey's HSD test between the objective function values at the maximum evaluations for each accuracy level across the 21 trials using PS, IB, and GB, respectively. Red indicated a significant difference between Groups 1 and 2, whereas blue indicated no significant difference.

The following subsections discuss the results of each model's management strategy.

#### 5.2.1. PS

For f1, f8, and f13, there was no significant difference between the accuracy of 0.5 and NoS, but significant differences could be confirmed between other accuracy pairs. For f2, no significant differences could be confirmed between the accuracies of 0.5 and 0.6, 0.5 and NoS, 0.6 and NoS, and 0.7 and 0.8. However, significant differences were confirmed when surrogates with higher accuracies were used. For f4, there were no significant differences between the accuracies of 0.5 and 0.6, and 0.5 and NoS, but significant differences were found in other pairs. For f15, all accuracy levels obtained significantly different results.

As a general trend, no significant difference existed between the results of the NoS and an accuracy level of 0.5, whereas the other combinations tended to show significant differences. This observation suggested that PS was an accuracy-sensitive strategy, implying that its search performance was influenced by the accuracy of the surrogate model used.

#### 5.2.2. IB

For f1, there were no significant differences between the accuracies of 0.7 to 1.0, but there were significant differences between the other accuracy

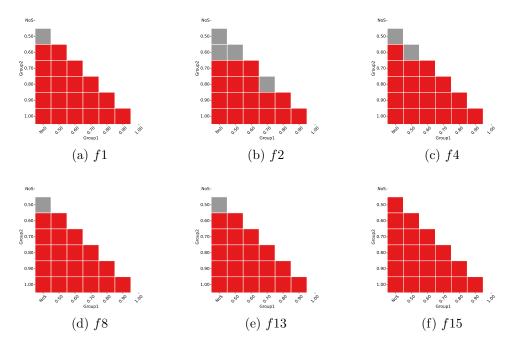


Figure 4: Heatmap of Tukey's HSD test results between accuracies when searching 30D problems using PS

levels. For f2, significant differences only existed between the NoS and accuracies of 0.6 or higher, with no significant differences between other accuracy levels. For f4, there were no significant differences between the NoS and an accuracy of 0.5, between accuracies of 0.6 to 1.0 (excluding 0.9), and between accuracies of 0.9 and 1.0. For f8, there were no significant differences between the accuracies of 0.6 to 1.0 and between 0.5 and 0.6. For f13, there were no significant differences between the accuracy of 0.5 and NoS. For f15, there were no significant differences between the accuracy of 0.5 and NoS. For f15, there were no significant differences between the accuracies of 0.6 to 0.9, 0.6, and 1.0, and 0.8 and 1.0.

These results indicated that the search performance of IB was robust, showing no significant difference when the surrogate accuracy was about 0.7 or higher. However, the performance declined when the accuracy decreased below 0.6.

#### 5.2.3. GB

For f1, there were no significant differences between accuracies of 0.7 and 0.9, as well as between accuracies ranging from 0.8 to 1.0, but significant

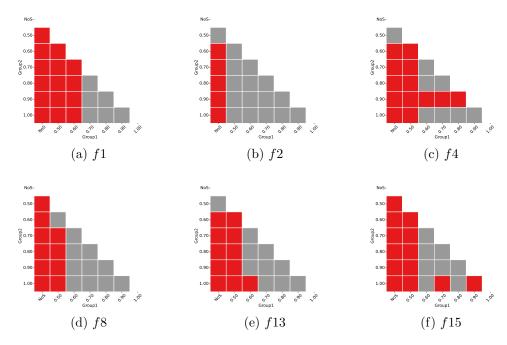


Figure 5: Heatmap of Tukey's HSD test results between accuracies when searching 30D problems using IB

differences were found across other accuracies. For f2, accuracies between 0.8 and 1.0 showed no significant difference, while other accuracies showed significant differences. For f4, a significant difference was only observed between the accuracies of 0.9 and 1.0. For f8, no significant differences were found between the accuracies of 0.7 and 1.0. For f13, accuracies of 0.8 to 1.0 exhibited no significant differences. Finally, for f15, no significant differences were found between the accuracies of 0.6, 1.0, 0.7, 0.9, and 0.8 and 1.0.

Overall, there was a tendency for no significant difference to be observed among the search results when the accuracy was high. Specifically, the search performance of GB was robust when the accuracy was 0.8 or higher. In contrast, with an accuracy of 0.7 or lower, significant differences emerged, indicating that the search performance was more sensitive to low-accuracy surrogate models.

# 5.3. Comparison of Different SAEA Strategies Across the Same Accuracy

This section compares the search performance of the different model management strategies for the same accuracy. Specifically, we performed the

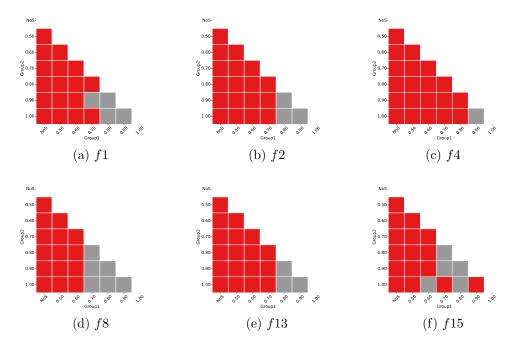


Figure 6: Heatmap of Tukey's HSD test results between accuracies when searching 30D problems using GB

Mann-Whitney U-test to compare the final results of PS, IB, and GB with the same accuracy.

Figure 7 shows the comparison results for each pair of model-management strategies. For the A versus B comparisons, red indicates that A has a better search performance than B, blue indicates worse performance, and gray indicates no significant difference.

For the comparison of PS and IB, Figures 7a and 7d show that IB outperformed PS in both 10 and 30 dimensions when accuracy was low, whereas PS tended to perform better at high accuracy. This suggested that PS with a low-accuracy surrogate rejects offspring individuals that were better than their parents and should be selected for the next generation based on surrogate prediction, which negatively affected search performance.

For the comparison of PS and GB, Figures 7b and 7e show that when the accuracy was 0.5, PS outperformed GB in all cases, except for f2 in 30 dimensions. However, when the accuracy was between 0.6 and 0.9, the GB either performed better or showed no significant difference. When the

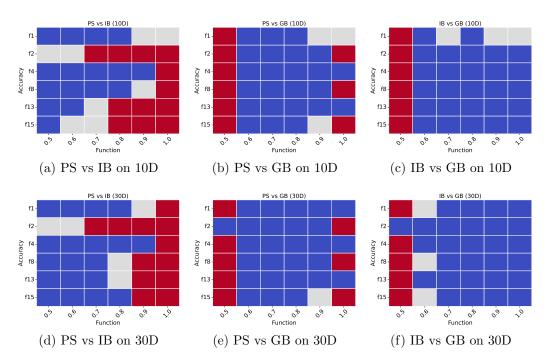


Figure 7: Performance comparisons of PS, IB, and GB across the same accuracy using the Mann-Whitney U-test

accuracy was 1.0, PS showed better results for functions f2, f8, and f15, where GB was better or did not differ from other functions.

For the comparison of IB and GB, Figures 7c and 7f show that IB outperformed GB in all cases, except for f2 in 30 dimensions, with an accuracy of 0.5. However, GB tended to perform better at an accuracy of 0.6 or higher.

The comparison of GB and other strategies with an accuracy of 0.5 showed that GB had significantly worse search performance. This was attributed to GB using surrogate evaluations more frequently than IB or PS. The repeated use of incorrect predictions misled the search direction of the GB, and the final promising solutions obtained were of lower quality. However, when the accuracy was 0.6 or higher, the increased number of generations enabled better search performance.

Based on these results, IB was superior when the accuracy was 0.5, whereas GB was superior at an accuracy of 0.6 or higher. However, when the accuracy reached 1.0, PS demonstrated superior performance.

In this experiment, it remained unclear exactly where the advantage

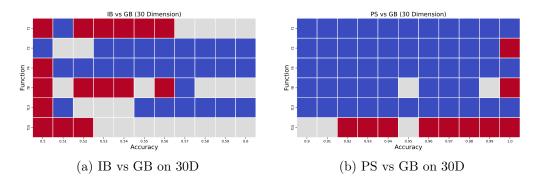


Figure 8: Performance comparisons of IB and GB (sp = [0.5, 0.6]) and PS and GB (sp = [0.9, 1.0]) using the Mann-Whitney U-test

shifted between the 0.5 and 0.6 accuracy and between the 0.9 and 1.0 accuracy. In addition, we conducted further experiments at 0.01 intervals between 0.5 and 0.6 and 0.9 and 1.0 accuracy, followed by significance testing.

The results of the Mann-Whitney U-test comparing IB and GB between accuracies of 0.5 and 0.6 (see Figure 8a). In Figure 8a, for the function f1, no significant difference was observed from an accuracy of 0.57 or higher. For f2, the GB became superior at 0.53 or higher. In f4, GB showed superiority from 0.51 or higher, and for f8, GB outperformed it at 0.57, but no significant differences were found thereafter. For f13, GB became superior with an accuracy of 0.55 and above, whereas for f15, no significant difference was observed at 0.53 or higher. Thus, GB should be selected when a surrogate accuracy of 0.57 or more could be ensured. Otherwise, IB was recommended when the accuracy was 0.56 or less.

Figure 8b presents the statistical test results comparing PS and GB for surrogate accuracy levels ranging from 0.9 to 1.0. In Figure 8b, for f1, GB consistently outperformed it; for f2, PS became superior only at an accuracy of 1.0. For f4, GB was always superior, whereas for f8, no significant difference was observed at 0.95 and 0.99, and PS became superior at 1.0. For f13, GB consistently outperformed, whereas for f15, no significant difference was found in the accuracies of 0.90, 0.91, and 0.95, with PS outperforming the other methods. Therefore, except in the case of f15, GB should be used unless an exact accuracy of 1.0 could be guaranteed.

Based on the above results, we could recommend guidelines for model management strategy selection.

- Use IB when the accuracy was between 0.5 and 0.56
- Use GB when the accuracy was between 0.57 and 0.99
- Use PS when the accuracy was 1.0

#### 6. Conclusion

This study aimed to analyze the impact of surrogate model prediction accuracy on SAEA search performance and its model management strategy. To this end, we constructed a pseudo-surrogate model that could set an arbitrary accuracy using actual evaluation function values. To answer our research questions, we conducted experiments comparing three SAEA strategies (PS, IB, and GB) and pseudo-surrogate models with various prediction accuracies ranging from 0.5 to 1.0. We used six CEC2015 benchmark problems (unimodal, multimodal, and their composites) with 10 and 30 dimensions and analyzed the progression of minimum values discovered during the search process with a maximum of 2,000 actual evaluations.

For RQ1, we found that increasing the accuracy of surrogate models improved search performance, but the impact varied depending on the search strategy (PS, IB, or GB).

Specifically, for PS, the experimental results showed a better search performance with a higher prediction accuracy. When the accuracy was 0.5, the performance was almost equivalent to that without the surrogate. However, when the accuracy was 0.6 or higher, they outperformed the results without a surrogate, and using more accurate models appeared beneficial in PS. For IB, positive correlations between accuracy and search performance were observed in all experimental results. However, the best performance was not always achieved when the accuracy was 1.0, though a certain level of accuracy was sufficient. In particular, for f15, the search performance decreased when the accuracy reached 1.0. For GB, while searches with an accuracy of 0.5 and NoS showed little progress, significant performance improvements were observed with an accuracy of 0.6 or higher. Moreover, except for f4, there were no significant differences in the search performance for accuracies of 0.8 or higher.

Regarding RQ2, we confirmed the common tendency that a higher surrogate model accuracy enhanced search performance and that its impact varied depending on the model management strategy.

In PS, no significant difference in search performance was observed between NoS and a surrogate model accuracy of 0.5, whereas search performance was improved by increasing surrogate model accuracy. In IB, differences were observed between the lower and higher accuracies. However, there was no significant difference in performance across the accuracies of 0.7 or higher. This suggested that the performance of IB stabilized beyond a certain accuracy threshold. Similarly, in GB, significant differences were observed between lower and higher accuracies, but no significant differences were found across accuracies of 0.8 or higher, suggesting that further accuracy improvements contributed minimally to the search performance beyond this threshold.

Finally, for RQ3, when the surrogate prediction accuracy was 0.56 or lower, it was preferable to use IB, whereas GB was recommended for accuracies of 0.57 or higher. However, if an exact accuracy of 1.0 could be ensured, using PS could be advantageous.

In our analysis, the impact of surrogate prediction accuracy on search performance and model management strategies in SAEA was examined under the assumption that the surrogate model's accuracy remained constant and uniformly distributed across the search space. However, in practical SAEA scenarios, the accuracy tended to improve with generational progress, and local accuracy variations arose because of biases in the training samples. These factors were not replicated in the pseudo-surrogate model, and future studies should address these limitations. Other future work includes analyzing the relationship between the prediction accuracy and search performance in SAEAs requiring multiple surrogate models (e.g., constrained optimization problems and multi-objective optimization problems). In addition, we explored methods for switching search strategies during the optimization process based on the recommendations obtained in this study.

#### CRediT authorship contribution statement

Yuki Hanawa: Writing – original draft, Investigation, Methodology, Formal analysis, Software, Visualization. Tomohiro Harada: Writing – review & editing, Conceptualization, Funding acquisition, Methodology, Project administration. Yukiya Miura: Resources, Supervision.

## Acknowledgments

This research was supported by a Japan Society for the Promotion of Science Grant-in-Aid for Young Scientists (Grant Number 21K17826).

# Declaration of competing interest

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

# Data availability

No relevant data were used in the research described in this study.

# Declaration of generative AI and AI-assisted technologies

During the preparation of this work the authors used ChatGPT in order to improve language and readability. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

## References

- [1] Chen, Q., Liu, B., Zhang, Q., Liang, J.J., Suganthan, P.N., Qu, B., 2015. Problem definitions and evaluation criteria for cec 2015 special session on bound constrained single-objective computationally expensive numerical optimization. URL: https://api.semanticscholar.org/CorpusID: 61216678.
- [2] Hanawa, Y., Harada, T., Miura, Y., 2024. Analysis of the impact of prediction accuracy on search performance in surrogate-assisted evolutionary algorithms, in: 2024 IEEE Congress on Evolutionary Computation (CEC), pp. 1–8. doi:10.1109/CEC60901.2024.10611759.
- [3] He, C., Zhang, Y., Gong, D., Ji, X., 2023. A review of surrogate-assisted evolutionary algorithms for expensive optimization problems. Expert Systems with Applications 217, 119495. URL: https://www.sciencedirect.com/science/article/pii/S0957417422025143, doi:https://doi.org/10.1016/j.eswa.2022.119495.

- [4] Jin, Y., 2005. A comprehensive survey of fitness approximation in evolutionary computation. Soft Computing 9, 3–12. URL: https://doi.org/10.1007/s00500-003-0328-5, doi:10.1007/s00500-003-0328-5.
- [5] Jin, Y., 2011. Surrogate-assisted evolutionary computation: Recent advances and future challenges. Swarm and Evolutionary Computation 1, 61-70. URL: https://www.sciencedirect.com/science/article/pii/S2210650211000198, doi:https://doi.org/10.1016/j.swevo.2011.05.001.
- [6] Jin, Y., Wang, H., Sun, C., 2021. Data-Driven Evolutionary Optimization. Springer International Publishing. URL: https://doi.org/10.1007/978-3-030-74640-7.
- [7] Kano, H., Harada, T., Miura, Y., 2022. Differential evolution using surrogate model based on pairwise ranking estimation for constrained optimization problems, in: 2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS), pp. 1–6. doi:10.1109/SCISISIS55246.2022.10001982.
- [8] Liu, B., Zhang, Q., Gielen, G.G.E., 2014. A gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems. IEEE Transactions on Evolutionary Computation 18, 180–192. doi:10.1109/TEVC.2013.2248012.
- [9] McKay, M.D., Beckman, R.J., Conover, W.J., 1979. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21, 239–245. URL: http://www.jstor.org/stable/1268522.
- [10] Mezura-Montes, E., Coello Coello, C.A., 2011. Constraint-handling in nature-inspired numerical optimization: Past, present and future. Swarm and Evolutionary Computation 1, 173-194. URL: https://www.sciencedirect.com/science/article/pii/S2210650211000538, doi:https://doi.org/10.1016/j.swevo.2011.10.001.
- [11] Mezura-Montes, E., Coello Coello, C.A., Tun-Morales, E.I., 2004. Simple feasibility rules and differential evolution for constrained optimization, in: Monroy, R., Arroyo-Figueroa, G., Sucar, L.E., Sossa, H. (Eds.),

- MICAI 2004: Advances in Artificial Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 707–716.
- [12] Mühlenbein, H., Schlierkamp-Voosen, D., 1993. Predictive models for the breeder genetic algorithm i. continuous parameter optimization. Evolutionary Computation 1, 25–49. doi:10.1162/evco.1993.1.1.25.
- [13] Ong, Y.S., Nair, P.B., Keane, A.J., 2003. Evolutionary optimization of computationally expensive problems via surrogate modeling. AIAA Journal 41, 687–696. URL: https://doi.org/10.2514/2.1999, doi:10.2514/2.1999, arXiv:https://doi.org/10.2514/2.1999.
- [14] Pan, J.S., Liu, N., Chu, S.C., Lai, T., 2021. An efficient surrogate-assisted hybrid optimization algorithm for expensive optimization problems. Information Sciences 561, 304—325. URL: https://www.sciencedirect.com/science/article/pii/S0020025520311609, doi:https://doi.org/10.1016/j.ins.2020.11.056.
- [15] Slowik, A., Kwasnicka, H., 2020. Evolutionary algorithms and their applications to engineering problems. Neural Computing and Applications 32, 12363–12379. URL: https://doi.org/10.1007/ s00521-020-04832-8, doi:10.1007/s00521-020-04832-8.
- [16] Song, Z., Wang, H., He, C., Jin, Y., 2021. A kriging-assisted two-archive evolutionary algorithm for expensive many-objective optimization. IEEE Transactions on Evolutionary Computation 25, 1013–1027. doi:10.1109/TEVC.2021.3073648.
- [17] Sonoda, T., Nakata, M., 2022. Multiple classifiers-assisted evolutionary algorithm based on decomposition for high-dimensional multiobjective problems. IEEE Transactions on Evolutionary Computation 26, 1581–1595. doi:10.1109/TEVC.2022.3159000.
- [18] Tong, H., Huang, C., Liu, J., Yao, X., 2019. Voronoi-based efficient surrogate-assisted evolutionary algorithm for very expensive problems, in: 2019 IEEE Congress on Evolutionary Computation (CEC), pp. 1996– 2003. doi:10.1109/CEC.2019.8789910.

- [19] Tong, H., Huang, C., Minku, L.L., Yao, X., 2021. Surrogate models in evolutionary single-objective optimization: A new taxonomy and experimental study. Information Sciences 562, 414– 437. URL: https://www.sciencedirect.com/science/article/pii/ S0020025521002395, doi:https://doi.org/10.1016/j.ins.2021.03. 002.
- [20] Tsujino, K., Harada, T., Thawonmas, R., 2020. Analysis of relation between prediction accuracy of surrogate model and search performance on extreme learning machine assisted moea/d, in: 2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), pp. 820–825. doi:10.23919/SICE48898.2020.9240452.
- [21] Yang, Z., Qiu, H., Gao, L., Cai, X., Jiang, C., Chen, L., 2020. Surrogate-assisted classification-collaboration differential evolution for expensive constrained optimization problems. Information Sciences 508, 50–63. URL: https://www.sciencedirect.com/science/article/pii/S0020025519308047, doi:https://doi.org/10.1016/j.ins.2019.08.054.
- [22] Yu, H., Tan, Y., Sun, C., Zeng, J., 2019. A generation-based optimal restart strategy for surrogate-assisted social learning particle swarm optimization. Knowledge-Based Systems 163, 14–25. URL: https://www.sciencedirect.com/science/article/pii/S0950705118304064, doi:https://doi.org/10.1016/j.knosys.2018.08.010.
- [23] Yu, L., Ren, C., Meng, Z., 2024. A surrogate-assisted differential evolution with fitness-independent parameter adaptation for high-dimensional expensive optimization. Information Sciences 662, 120246. URL: https://www.sciencedirect.com/science/article/pii/S0020025524001592, doi:https://doi.org/10.1016/j.ins.2024.120246.

,