



Bi-indicator driven surrogate-assisted multi-objective evolutionary algorithms for computationally expensive problems

Wenxin Wang¹ · Huachao Dong¹ · Peng Wang¹ · Jiangtao Shen¹

Received: 12 September 2022 / Accepted: 3 January 2023 / Published online: 9 February 2023
© The Author(s) 2023

Abstract

This paper presents a bi-indicator-based surrogate-assisted evolutionary algorithm (BISAEA) for multi-objective optimization problems (MOPs) with computationally expensive objectives. In BISAEA, a Pareto-based bi-indicator strategy is proposed based on convergence and diversity indicators, where a nondominated sorting approach is adopted to carry out two-objective optimization (convergence and diversity indicators) problems. The radius-based function (RBF) models are used to approximate the objective values. In addition, the proposed algorithm adopts a one-by-one selection strategy to obtain promising samples from new samples for evaluating the true objectives by their angles and Pareto dominance relationship with real non-dominated solutions to improve the diversity. After the comparison with four state-of-the-art surrogate-assisted evolutionary algorithms and three evolutionary algorithms on 76 widely used benchmark problems, BISAEA shows high efficiency and a good balance between convergence and diversity. Finally, BISAEA is applied to the multidisciplinary optimization of blend-wing-body underwater gliders with 30 decision variables and three objectives, and the results demonstrate that BISAEA has superior performance on computationally expensive engineering problems.

Keywords Expensive multi-objective optimization · Pareto-based bi-indicator · One-by-one selection · Radial basis function

Introduction

Real-life engineering problems often need to simultaneously optimize multiple conflicting objectives [1], which are named multi-objective optimization problems (MOPs). A minimization MOP can be defined as follows [2]:

$$\begin{aligned} \min F(\mathbf{X}) &= (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x}))^T \\ \text{s.t. } L_i &\leq x_i \leq U_i \quad i \in \{1, 2, \dots, d\} \end{aligned} \quad (1)$$

where f_1, f_2, \dots, f_M are M objective functions to be optimized, L_i and U_i are the lower and upper boundaries of x_i . $\mathbf{x} = \{x_1, x_2, \dots, x_d\}$, and d is the number of decision variables of the optimization problem.

As an important method to solve MOPs, multi-objective optimization evolutionary algorithms (MOEAs) have been developed rapidly in recent 20 years. MOEAs search for a set of solutions to represent the whole Pareto front (PF).

Commonly used MOEAs can be roughly divided into three categories: indicator-based methods [3], dominance-based methods [4], and decomposition-based methods [5, 6].

It is worth mentioning that MOEAs require a large number of function evaluations (NFEs) to obtain the true PF. However, in some engineering optimization problems, the fitness functions are quite time-consuming simulations, such as Computational fluid dynamics (CFD) and Finite element analysis (FEA). To reduce the overall NFEs, the mainstream is to adopt surrogate models for approximation of expensive physical models of expensive optimization [7]. Various surrogates have been used to solve real engineering optimization problems [8–10], such as polynomial response surface (PRS) [11], Kriging [12], neural network (NN) [13], radius-based function (RBF) [14] and so on. Surrogate-assisted evolutionary algorithms (SAEAs) are proposed to handle single-objective optimization using classification or regression-based fitness approximation, which have been adopted successfully in engineering optimization e.g., a novel evolutionary sampling optimization method (ESAO) [15], surrogate-assisted grey wolf optimization (SAGWO) [16] and surrogate-assisted teaching and learning optimization (SATLBO) [17].

✉ Huachao Dong
hdong@nwpu.edu.cn

¹ School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an 710068, China

Inspired by these studies, numerous SAEAs for expensive multi-objective optimization are proposed in the past decades. Many surrogate models are used to approximate the function values in SAEAs, which can be roughly divided into two categories. In the first category, the surrogate models are used to approximate the objective functions, scalarized functions, and so on. The Kriging-assisted reference vector guided evolutionary algorithm (K-RVEA) [18] and expected improvement (EI) matrix-based infill criteria MOEA [19] for expensive multi-objective optimization are representative methods. Liu et al. [20] suggested a reference vector-assisted adaptive Kriging model management strategy (RVMM) and Song et al. [21] developed a Kriging-assisted two-archive evolutionary algorithm (KTA2) for surrogate-assisted many-objective optimization. The Kriging models are also used in MOEA-based decomposition (MOEA/D-EGO) [22], MOEA/D-EGO applies a fuzzy clustering-based modeling method in the decision space to build many local surrogate models for each objective. In addition, an efficient dropout neural network-assisted indicator-based MOEA with reference point adaptation (EDN-ARMOEA) [23] and a surrogate-assisted particle swarm optimization algorithm with an adaptive dropout mechanism (ADSAPSO) [24] was developed using the neural network. Meanwhile, the bagging technology of the surrogate models is also adopted to approximate function values, such as a heterogeneous ensemble-based infill criterion for MOEA (HeE-MOEA) [25].

In the second category, the classifiers are adopted as surrogate models. A classification-based SAEA (CSEA) [26] uses feedforward neural networks (FNNs) to predict the dominant relationship between candidate solutions and reference solutions. Zhang et al. [27] suggest a classification-based preselection for the multi-objective evolutionary algorithm (CPS-MOEA).

Significantly, some of the above-mentioned surrogate-assisted MOEAs use the indicator to improve their performance, such as RVMM [20], KTA2 [21], EDN-ARMOEA [23], and so on. To take full advantage of indicators, we propose a bi-indicator-based surrogate-assisted MOEA (BISAEA), where the RBF models are adopted to approximate the expensive function evaluation, a Pareto-based bi-indicator (convergence indicator, CI and diversity indicator, DI) strategy is proposed to transform the MOPs into a bi-objective (CI and DI) optimization problem, and a one-by-one selection strategy is adopted to get expected samples for re-evaluation. To verify the effectiveness of BISAEA, it is compared with four state-of-the-art SAEAs and three MOEAs on 76 widely used benchmark problems and applied to the multidisciplinary optimization design of Blend-Wing-Body Underwater Gliders (BWBUGs). The contributions of this work are summarized as follows:

1. A Pareto-based bi-indicator strategy is designed to obtain new samples by using approximate objective values. A convergence indicator and diversity indicator are calculated by approximating objective values, and which are taken as the bi-objective optimization to select new samples based on Pareto sorting.
2. The one-by-one selection strategy is adopted to select several promising samples for re-evaluation. The candidate samples are selected by evaluating the angle of their approximate function values and exiting advantage samples successively.
3. The performance of BISAEA is evaluated on 76 benchmark functions and multidisciplinary design optimization of BWBUGs with three objectives, the results show that BISAEA outperforms the comparison algorithms.

The remainder of this article is organized as follows. In “[Related work](#)”, we review the related work on indicator and RBF models. The details of the proposed BISAEA are presented in “[The proposed algorithm](#)”, and the results of experiments on mathematical cases are shown in “[Empirical studies](#)”. The engineering applications are reported in “[Application to engineering problem](#)”, and “[Conclusions](#)” concludes the article and draws the future work.

Related work

Indicator-based MOEAs

As an important component of MOEAs, indicator-based MOEAs adopt the evaluation indicator to measure the performance of the solutions to obtain the final optimal solutions. Several indicators have been proposed, such as hypervolume [3, 28], IGD-NS [29], R^2 [31, 32], I_{ε^+} [32–35] and others [36]. In this article, I_{ε^+} is adopted as a convergence indicator, and I_{ε^+} reflects the smallest adjustment that may be made to allow one solution set to marginally outperform another for each objective. It might be characterized as follows:

$$I_{\varepsilon^+}(X_1, X_2) = \min \{ \varepsilon | \forall x_2 \in X_2, \exists x_1 \in X_1 : f_i(x_1) - \varepsilon \leq f_i(x_2), \text{ for } i = 1, 2, \dots, M \} \quad (2)$$

where X_1 and X_2 are two solutions sets and M is the number of objectives. I_{ε^+} could be defined as follows:

$$\mathbf{I}(\mathbf{x}_1, \mathbf{x}_2) = \max(\mathbf{F}(\mathbf{x}_1) - \mathbf{F}(\mathbf{x}_2)) \quad (3)$$

where $\mathbf{F}(\mathbf{x}_1)$ and $\mathbf{F}(\mathbf{x}_2)$ are objective values of \mathbf{x}_1 and \mathbf{x}_2 . The maximum indicator value for all $\mathbf{x}_i \in X$ will be evaluated as a scalar indicator by Eq. (4).

$$c(\mathbf{x}_j) = \max_{\mathbf{x}_i \in X} |\mathbf{I}(\mathbf{x}_i, \mathbf{x}_j)| \quad (4)$$

Finally, the fitness values can be computed for all $\mathbf{x}_i \in X$ in the following equation:

$$Fit(\mathbf{x}_i) = \sum_{\mathbf{x}_j \in X \setminus \{\mathbf{x}_i\}} -e^{-I(\mathbf{x}_i, \mathbf{x}_j)/(c(\mathbf{x}_j) \cdot K)} \tag{5}$$

where K is the scaling factor.

Meanwhile, the minimum angle is used as a diversity indicator, which was used in relevant research [9]. As shown in Fig. 1, it is obvious that the new samples 3 and 4 have two angles (γ , γ_1 and λ , λ_1) with the reference points, the angle indicator between each new sample and the reference point is shown in Fig. 1b) based on the minimum angle.

as bi-objective optimization to enhance the convergence and diversity simultaneously in BISAEA.

Radial basis function

In this article, the RBF model [14] is used as the surrogate model. Studies [37] reveal that RBF can obtain more accurate approximations for high dimensional problems, and its modeling speed is fast compared with the Kriging model. Given the data points $\{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathfrak{R}^d, i = 1, 2, \dots, N\}$, the RBF surrogate is defined as follows:

Algorithm 1: Fitness Assignment of I_{ϵ^+}

Input: X , the decision variables of current solutions $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$; Y , the objective values of current

solutions $Y = \{y_1, \dots, y_N\}$; N , the size of current solutions; K , scaling factor;

Output: $\{Fit(\mathbf{x}_1), \dots, Fit(\mathbf{x}_N)\}$

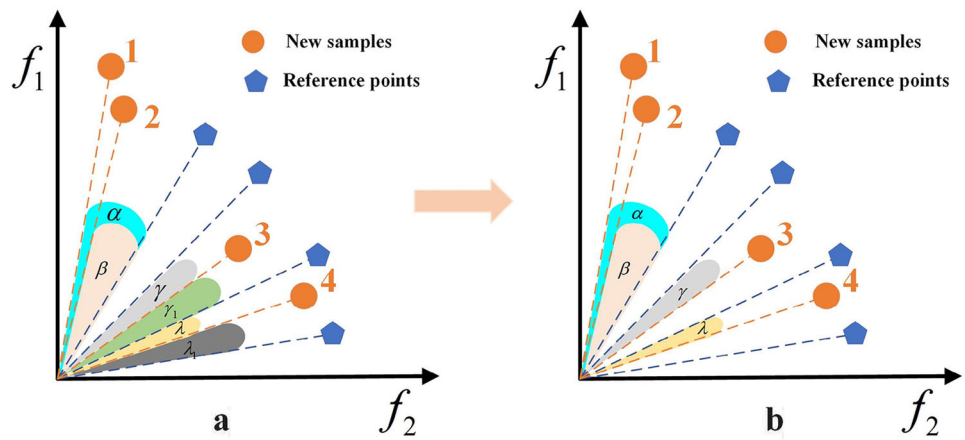
- 1: $Y_{\max} = \max(y_1, \dots, y_N)$;
- 2: $Y_{\min} = \min(y_1, \dots, y_N)$;
- 3: $Y_{nor} = \frac{Y - Y_{\min}}{Y_{\max} - Y_{\min}}$;/*Normalize the objective values*/
- 4: **For** $i = 1:N$
- 5: **For** $j = 1:N$
- 6: **If** $i \neq j$
- 7: $I(\mathbf{x}_i, \mathbf{x}_j) = \infty$;
- 8: **else**
- 9: $I(\mathbf{x}_i, \mathbf{x}_j) = \max(y_i - y_j)$;
- 10: **End If**
- 11: **End For**
- 12: **End For**

Different from the above indicator-based MOEAs, BISAEA adopts two indicators (CI and DI) as bi-objective to select the promising samples. Especially, the above MOEAs [28–36] use the indicators to select the expected samples to promote the convergence or diversity independently, the two indicators (bi-indicator, minimum angle and I_{ϵ^+}) are used

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^n \lambda_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) + p(\mathbf{x}), \mathbf{x} \in \mathfrak{R}^d \tag{6}$$

where $\|\mathbf{x} - \mathbf{x}_i\|$ is the Euclidean distance between the points \mathbf{x} and \mathbf{x}_i , $\phi(\cdot)$ is the basic function. Many forms of the basis

Fig. 1 The minimum angle with reference points



function can be used here. In this article, the cubic form ($\phi(r) = r^3$) is adopted because it was successfully employed in several surrogate-based algorithms [17, 38]. In addition, the weight vector $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_N)^T$ can be computed as follows:

$$\lambda = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y} \tag{7}$$

where $\mathbf{y} = (y_1, y_2, \dots, y_N)^T$ is the output vector and Φ is the following matrix:

$$\Phi = \begin{bmatrix} \phi(\mathbf{x}_1 - \mathbf{x}_1) & \phi(\mathbf{x}_1 - \mathbf{x}_2) & \dots & \phi(\mathbf{x}_1 - \mathbf{x}_N) \\ \phi(\mathbf{x}_2 - \mathbf{x}_1) & \phi(\mathbf{x}_2 - \mathbf{x}_2) & \dots & \phi(\mathbf{x}_2 - \mathbf{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\mathbf{x}_N - \mathbf{x}_1) & \phi(\mathbf{x}_N - \mathbf{x}_2) & \dots & \phi(\mathbf{x}_N - \mathbf{x}_N) \end{bmatrix} \tag{8}$$

$p(x)$ is a linear polynomial in d variables with $d + 1$ coefficients as in the formula:

$$p(x) = c_0 + c_1x_1 + c_2x_2 + \dots + c_dx_d = c^T \cdot x \tag{9}$$

The proposed algorithm

The framework of BISAEA

The framework of BISAEA is shown in Fig. 2, which can be divided into three parts, initialization, Pareto-based bi-indicator selection strategy, and one-by-one selection strategy. In the first part, the initial population and reference vector are obtained and the relevant parameters are set. In the

second part, a Pareto-based bi-indicator selection strategy is used to obtain better candidate samples. Variation operation is applied to produce offspring using the parent populations, and the RBF model is used to approximate the function values for each objective of offspring. The bi-indicator is calculated by the approximate function values and selecting better samples for the next operation. Finally, the best samples are selected by the one-by-one selection strategy to evaluate the real function.

The pseudocode of BISAEA is given in Algorithm 1, which can be divided into the following steps:

- (1) Initialization (Lines 1–4): The initial population P is generated using the Latin hypercube sampling (LHS) [39, 40]. The initial reference vector V_0 and the Pareto non-dominated samples P_{nd} are obtained. Besides, the parent population is determined.
- (2) Pareto-based bi-indicator selection strategy (Lines 5–14): Generating offspring from parent populations by crossover and mutation and approximating the objective values of the offspring by RBF model. The bi-indicator is evaluated using the approximate objective values of offspring and the non-dominated method is adopted to select better samples based on the bi-indicator.
- (3) The one-by-one selection strategy (Lines 15–27): The better samples are combined with P_{nd} , and a non-dominated sorting method is employed to obtain the new PFs of them, and the better samples are selected from new PFs. If the size of the better sample is larger than N_{max} , select the N_{max} best samples by Algorithm 4 and re-evaluate it to add into the database, otherwise re-evaluate X_{better} to add into the database.
- (4) Repeat (2)–(3) until the termination condition is met.

Algorithm 2: Pseudocode of BISAEA

Input: $MFEs$, maximum function evaluations; N , population size; K , the scaling factor; w_{\max} , the maximum number of iterations; N_{\max} , the maximum number of new samples for real function evaluation each time;

Output: P_{nd} , non-dominated solutions of the population;

*/*Initialization*/*

1: Form the population (P) based on LHS, and add the population into the database; */*the variables of the population (P) are X and the objective functions are Y */*

2: Initialize V_0 , a set of uniform unit reference vectors of size N ; $P_{nd} \leftarrow$ non-dominated solutions from P ;

3: Record the number of function evaluations as FES ;

4: $Dec \leftarrow X$; $Obj \leftarrow Y$; */* initialize parent populations*/*

*/*Main Loop*/*

While $FES < MFEs$ **do**

*/*Pareto-based bi-indicator selection strategy*/*

5: $model \leftarrow$ train RBF model for each objective by using X and Y ;

6: $w \leftarrow 1$; */*initialize the number of iterations*/*

7: **While** $w < w_{\max}$ **do**

8: $X_{new} \leftarrow$ generate offspring of Dec ;

9: $Y_{new} \leftarrow$ approximate objective value ($model$, Dec);

10: $Dec \leftarrow \{Dec, X_{new}\}$; */* combine parent and offspring populations*/*

11: Select the samples by **Algorithm 3** with P_{nd} and V_0 ;

12: $w \leftarrow w + 1$;

13: $Dec \leftarrow X_{better}$; $Obj \leftarrow Y_{better}$;

14: **End while**

*/*One-by-one selection strategy*/*

15: $Y_{better} \leftarrow$ non-dominated solutions in $\{P_{nd_{obj}}, Y_{better}\}$; */* $P_{nd_{obj}}$ is objective values in P_{nd} */*

16: $Y_{better} \leftarrow \{Y_{better} / P_{nd_{obj}}\}$;

17: $X_{better} \leftarrow$ find the design variables corresponding to Y_{better} ;

18: **If** $|X_{better}| > N_{\max}$ **then**

19: $X_{best} \leftarrow$ select the best samples by using **Algorithm 4**;

20: **Else If**

21: $X_{best} \leftarrow X_{better}$;

22: **End If**

23: Re-evaluate X_{best} with the real functions and update $FES = FES + |X_{best}|$;

24: Add these samples to P and the database;

25: $P_{nd} \leftarrow$ non-dominated solutions from P ;

26: $Dec \leftarrow X$; $Obj \leftarrow Y$; */* X and Y are the variables and objective values of the population (P)*/*

27: **End while**

Pareto-based bi-indicator selection strategy

In BISAEA, a Pareto-based bi-indicator strategy is proposed to select better samples from offspring. This strategy includes two parts, i.e., bi-indicator calculation and selection based on the Pareto relationship.

In the first part (Lines 1–8), we adopt the non-dominated method to select the new PFs of offspring and their parent populations. The new PFs (X_{better} and Y_{better}) are selected using the I_{g^+} and the minimum angle. It should be noted that the objective values (Y_{new} and Obj) are the approximate values of the RBF model.

their parent samples are taken as a combination, and the non-dominated sorting method is employed to obtain the new PFs and the candidate samples are selected from new PFs.

To select the samples with better diversity and convergence, the Pareto-based bi-indicator strategy is proposed. The main idea of the Pareto-based bi-indicator strategy is to formulate the selected samples as the MOP, where convergence and diversity indicators are two objectives to be optimized. It is worth noting that the convergence and diversity indicators are evaluated by the approximate objective values of the RBF model in BISAEA.

After obtaining two indicators, the Pareto-based bi-indicator strategy can be written as:

Algorithm 3: Pareto-based bi-indicator

Input: X , design variable of the database; Y , objective values of the database; P_{nd} ; the non-dominated samples of the database; V_0 , the reference vector; w_{max} , the maximum number of iterations; K , the scaling factor; X_{new} , design variable of sample points; Y_{new} , approximate objective values of sample points; Dec, Obj ; the variables of population and the objective functions of parent populations;

Output: X_{better} the better samples points; Y_{better} , approximate objective values of better samples;

- 1: $Y_{better} \leftarrow$ non-dominated solutions in $\{Y_{new}; Obj\}$;
 - 2: $X_{better} \leftarrow$ find the design variables corresponding to Y_{better} ;
 - 3: $Pndobj_{max} = \max(Pnd1_{obj}, \dots, PndN_{obj})$; /* $Pnd1_{obj}$ and $PndN_{obj}$ are the objective values of the first and last non-dominated samples*/
 - 4: $Pndobj_{min} = \min(Pnd1_{obj}, \dots, PndN_{obj})$;
 - 5: $V \leftarrow V_0 \bullet (Pndobj_{max} - Pndobj_{min}) + Pndobj_{min}$;
 - 6: $CI \leftarrow$ calculate the I_{g^+} (convergence indicator);
 - 7: $MA \leftarrow$ calculate the minimum angle (diversity indicator) with V ;
 - 8: $DI \leftarrow -MA$;
 - 9: $ND \leftarrow$ non-dominated solutions in $\{CI, DI\}$;
 - 10: $X_{better}, Y_{better} \leftarrow$ find the design variables and objective values corresponding to ND ;
-

In the second part (Lines 9–10), the convergence indicator (I_{g^+}) and the diversity indicator (The opposite of the minimum angle) is adopted as the bi-objective, and the non-dominated method is used to obtain the PFs of them. In this way, the selected samples will have a good convergence and diversity at the same time.

The details of the Pareto-based bi-indicator selection strategy can be found in Figs. 3 and 4. The offspring samples and

$$\begin{aligned}
 \min G(x) &= (g_1(x), g_2(x)) \\
 g_1(x) &= DI \\
 g_2(x) &= CI
 \end{aligned}
 \tag{10}$$

The Pareto-based bi-indicator selection process is shown in Fig. 4. The convergence and diversity indicators (Fig. 4b and c) are evaluated by the candidate samples (Fig. 4a)

Fig. 2 The framework of BISAEA

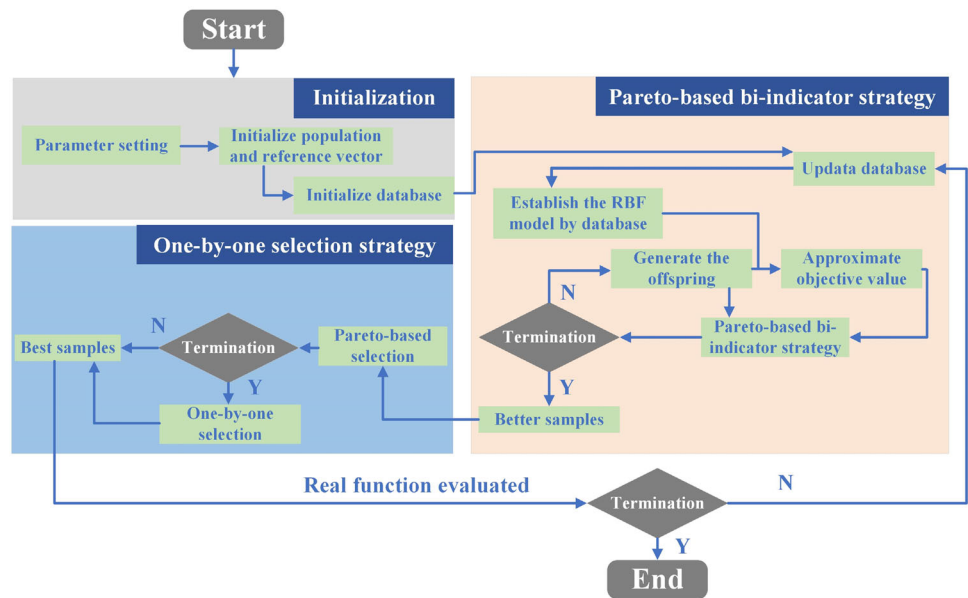


Fig. 3 The non-domination selection of the combination of parent and offspring

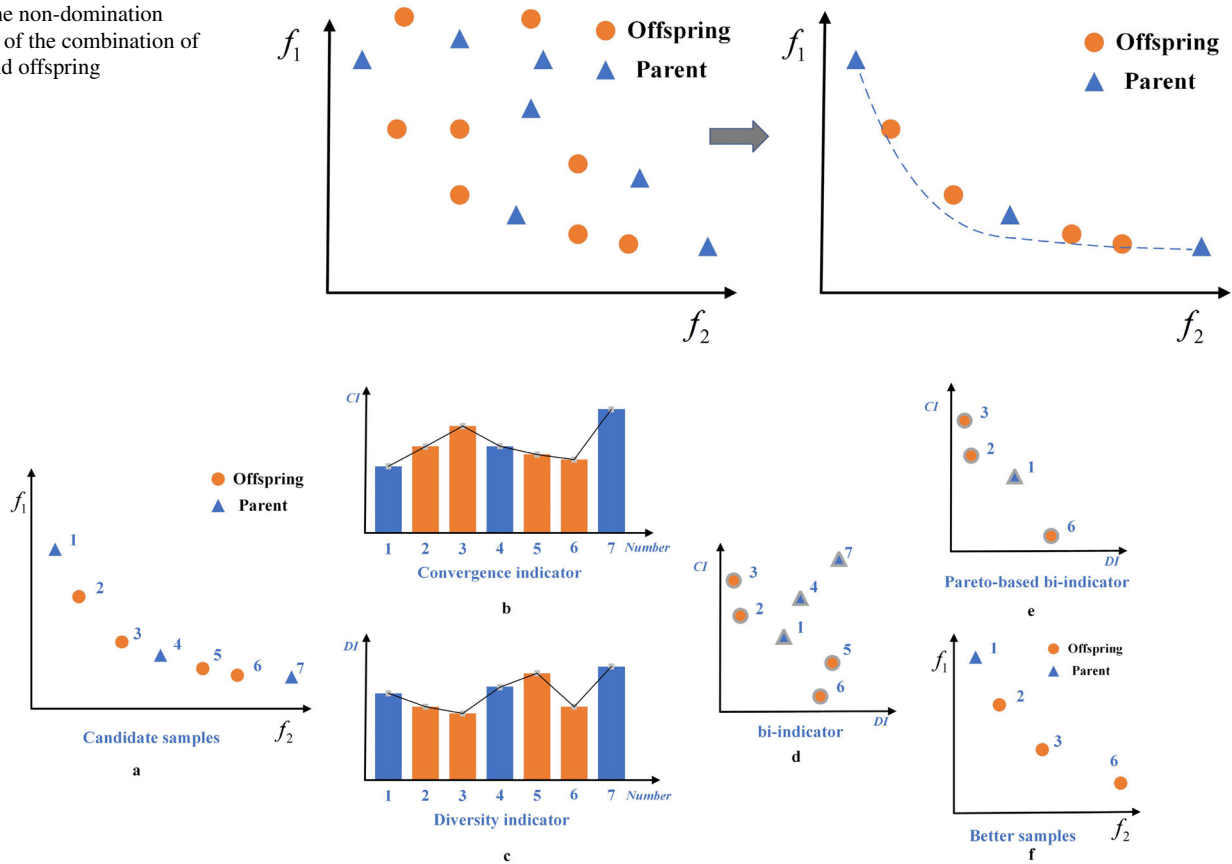


Fig. 4 Selecting samples by bi-indicator

based on I_{ε^+} and minimum angle. It is clear that the convergence indicator and diversity indicator have a big difference for candidate samples. Sample 6 has the advantage of both convergence and diversity indicators, but sample 7 has the opposite performance. To give consideration to both convergence and diversity, the bi-indicator is adopted as the bi-objective as shown in Fig. 4d. The non-dominated sorting method is used to carry out the bi-objective (CI and DI) optimization problem, the new PFs of them are represented in Fig. 4e and the corresponding function values are shown in Fig. 4f. It is obvious that the better samples have better performance compared to the candidate samples.

The one-by-one selection strategy

The one-by-one selection is a selection strategy in MOEAs, which has been proven to be effective in improving the performance of corresponding algorithms [21, 23]. Inspired by the relevant work [21, 23, 41–43], a one-by-one selection strategy is adopted in BISAEA to improve its performance .

Especially, the one-by-one selection strategy has two parts, Pareto-based selection and one-by-one selection. To obtain better convergence, the Pareto-based method is used to select the samples from better samples. The one-by-one selection is proposed to select the best samples based on the angle. The details of the Pareto-based selection strategy are shown in Fig. 5. The better samples and the existing non-dominated samples (PFs samples) are taken as a combination. Then Pareto-based method is employed to obtain the new PFs of them and better samples are selected from new PFs.

The details of the one-by-one strategy are presented in Algorithms 4. To obtain better diversity, the non-dominated solutions of the current population are chosen as the reference points (PFs samples), and the minimum angle between the better samples and the PFs samples is calculated. As shown in Fig. 6, taking the example of selecting the three best samples. The minimum angles between the better samples and PFs samples are represented as colored sectors. Sample 1 corresponding to the maximum angle is first selected as the best sample and merged with the PFs samples (Fig. 6b). Then, samples 5 and 3 are selected with the maximum angle in turn (Fig. 6c and d). The final selection result is shown in Fig. 6e.

Algorithm 4: The one-by-one selection

Input: N_{\max} , the maximum number of new samples for real function evaluation each time; P_{nd} , non-dominated solutions of the population; X_{better} , design variable of better sample points; Y_{better} , approximate objective values of better sample points;

Output: X_{best} , best samples based on sampling strategy;

4: $ref \leftarrow P_{nd}_{obj}$; $X_{best}^{new} \leftarrow X_{best}$; $X_{best} \leftarrow$ empty database;

5: **While** $|X_{best}| \leq N_{\max}$ **do**

6: $\theta \leftarrow$ calculate the angle values between the individual in Y_{best} and its nearest ref ;

7: $X_{r_{best}} \leftarrow$ choose the sample by maximum θ ;

8: $ref \leftarrow \{ref; Y_{r_{best}}\}$; $Y_{r_{best}}$ is the approximate function values of $X_{r_{best}}$ *;/

9: $Y_{best} \leftarrow$ delete the objective values corresponding of $X_{r_{best}}$;

10: $X_{best}^{new} \leftarrow$ delete $X_{r_{best}}$;

11: $X_{best} \leftarrow \{X_{best}; X_{r_{best}}\}$;

12: **End while**

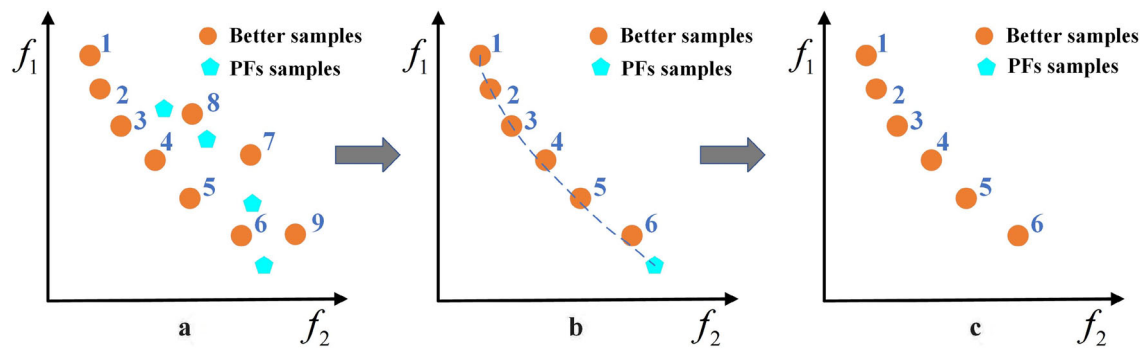


Fig. 5 The non-domination selection of the combination of PFs samples and better samples

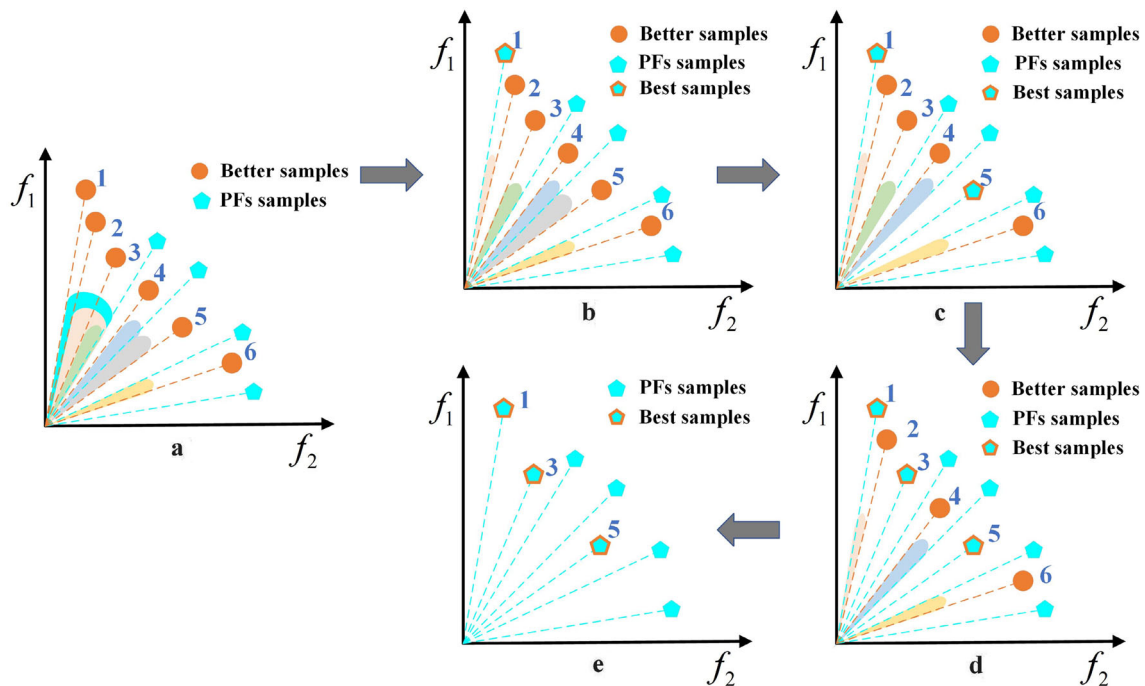


Fig. 6 The process of the one-by-one strategy

Empirical studies

In this section, to validate the effectiveness of BISAEA, we empirically compare it with several state-of-the-art multi-objective optimization algorithms, including several SAEAs (K-RVEA [18], HeEMOEA [25], EDN-ARMOEA [23], KTA2 [21] and several MOEAs (NSGA-III [4], RVEA [6] and IBEA [33]). All the test instances are implemented in PlatEMO [44]. The seven compared algorithms are summarized as follows.

1. K-RVEA [18] is a Kriging-assisted reference vector-guided evolutionary algorithm, which uses Kriging to approximate each objective and uncertainty information is provided to balance convergence and diversity.
2. HeE-MOEA [25] is a heterogeneous ensemble-assisted MOEA, in which a support vector machine and two RBF networks are constructed to enhance the reliability of ensembles for uncertainty estimation.
3. EDN-ARMOEA [23] is an efficient dropout neural network (EDN) assisted indicator-based MOEA [29], in which the EDN replaces the Gaussian process to achieve a good balance between convergence and diversity.
4. KTA2 [21] is a Kriging-assisted two-archive evolutionary algorithm and uses one influential point-insensitive model to approximate each function value. Moreover, an adaptive infill criterion for convergence, diversity and uncertainty is adopted to determine the promising samples for real function evaluation.

5. NSGA-III [4] is an evolutionary multi-objective optimization algorithm using a reference-point-based non-dominated sorting approach.
6. IBEA [33] uses a binary additive ε -indicator ($I_{\varepsilon+}$) as its selection criterion instead of the Pareto dominance criterion.
7. RVEA [6] is a reference vector-guided evolutionary algorithm for multi-objective optimization, and the reference vectors are used to decompose the original multi-objective optimization problem into a number of single-objective subproblems.

The experiments are conducted on 76 test instances from test suite DTLZ [45] and ZDT [46] with 2 and 3 objectives. For each test instance, the MFEs is set to 500, and the numbers of decision variables are set as 10, 15, 30 and 50.

We used the inverted generational distance (IGD) [47] as the performance indicator to assess the performance of the compared SAEAs. In general, the lower the IGD value is, the better the solutions approximate the true PF. All experiments are conducted using MATLAB with an Intel (R) Core (TM) i7, 3.4 GHz CPU. The Wilcoxon rank-sum (WRS) is used to conduct statistical tests at a significance level of 5%. Symbols “+” and “−” indicate that BISAEA is significantly superior and inferior to the compared algorithm, “ \approx ” means that there is no significant difference between BISAEA and the compared algorithm.

Parameter setting

The common parameter settings of all the compared algorithms are listed as follows:

1. The population size is set to 100.
2. The scaling factor (K) is set as 0.05 consistent with the original literature [33].
3. The maximum number of expensive function evaluations is set to 500.
4. The maximum number of iterations (w_{\max}) is set as 20, which is the same as K-RVEA[18].
5. The parameters for reproduction (crossover and mutation) are set to $P_c = 1.0$, $P_m = 1/d$, $\eta_c = 20$, $\eta_m = 20$.
6. The dimension of the design variable is set as 10, 15, 30 and 50.

In addition, for a fair comparison, we adopt the recommended setting in the original literature for specific parameters of compared algorithms.

Behavior study of the BISAEA

Sensitivity analysis of parameters in BISAEA

The maximum number of new samples for real function evaluation each time (N_{\max}) is a key parameter in BISAEA. N_{\max} is set as 1, 3, 5 and 7 to explore the influence of this parameter on the BISAEA, which are name BISAEA_1, BISAEA_3, BISAEA_5 and BISAEA_7. The average IGD results of each algorithm based on 30 independent runs on DTLZ1, 2, 7 and ZDT 1, 2 problems are shown in Table 1, where the WRS test is also listed and the best results are highlighted.

As shown in Table 1, BISAEA_1 has the best performance, followed by BISAEA_3. It is obvious that the performance of algorithms deteriorates with the increase of N_{\max} . However, the smaller values of N_{\max} will lead to a longer calculation time, which can be found in Fig. 7. In Fig. 7, the mean runtime of different problems on BISAEA_1 and BISAEA_3 is displayed based on 30 independent runs. It is clear that the mean runtime of BISAEA_3 is shorter than BISAEA_1, and the performance of the two algorithms is similar as shown in Table 1. Based on computational efficiency and overall performance, N_{\max} is set as 3 in this article.

Effect of the Pareto-based bi-indicator strategy

In this part, we first investigate the effects of the Pareto-based bi-indicator strategy of BISAEA. A variant of BISAEA named BISAEA(one), which does not adopt the Pareto-based bi-indicator strategy and only uses the one-by-one selection strategy. The average IGD results of BISAEA(one) and BISAEA based on 30 independent runs on DTLZ1, 2, 7 and ZDT 1, 2 problems are shown in Table 2, where the WRS test is also listed and the best results are highlighted.

In the benchmark problems of the above test, DTLZ1 has multi-model landscapes that is difficult to converge and DTLZ2 is easy to converge but maintains diversity with difficulty. DTLZ7 has irregular and discontinuous PF, ZDT1 and ZDT2 have convex PF. As shown in Table 2, it is easy to that BISAEA has better performance than BISAEA(one).

To better investigate the effects of the Pareto-based bi-indicator strategy, the final non-dominated solutions achieved by BISAEA(one) and BISAEA on 10D and 30D are shown in Figs. 8 and 9. Moreover, the true PF of DTLZ1 and DTLZ7 is shown in the last of Figs. 8 and 9. It is obvious that the results of BISAEA in DTLZ1 and DTLZ7 on 10D and 30D have better convergence than BISAEA(one), which also could be found in ZDT1 and ZDT2. The main reason is that the Pareto-based bi-indicator strategy adopts a convergence indicator, which could improve the convergence speed. From the results of DTLZ2 of

Table 1 Statistic results for IGD values obtained by BISAEA_1, 3, 5 and 7

Problem	M	D	BISAEA_1	BISAEA_3	BISAEA_5	BISAEA_7
DTLZ1	2	10	7.6395e+1 (4.05e+0) +	7.9292e+1 (7.83e+0) +	8.3909e+1 (9.75e+0) =	8.9763e+1 (1.32e+1)
	3	10	5.7138e+1 (2.22e+0) +	5.8816e+1 (5.09e+0) +	6.4499e+1 (6.92e+0) =	6.6784e+1 (8.17e+0)
	2	15	1.2353e+2 (1.02e+1) +	1.6472e+2 (3.83e+1) +	1.8823e+2 (4.17e+1) =	1.8716e+2 (4.38e+1)
	3	15	1.0700e+2 (1.87e+1) +	1.4423e+2 (3.80e+1) =	1.4982e+2 (3.19e+1) =	1.4346e+2 (3.28e+1)
	2	30	6.2740e+2 (8.71e+1) +	6.7235e+2 (1.14e+2) =	7.2240e+2 (9.27e+1) =	7.0506e+2 (9.39e+1)
	3	30	5.0547e+2 (5.90e+1) +	5.5668e+2 (5.95e+1) =	5.6399e+2 (7.22e+1) =	5.7789e+2 (6.74e+1)
	2	50	1.1429e+3 (9.22e+1) +	1.2407e+3 (1.09e+2) +	1.3336e+3 (1.18e+2) =	1.3337e+3 (1.19e+2)
	3	50	9.8380e+2 (1.13e+2) +	1.0685e+3 (7.43e+1) +	1.1343e+3 (8.56e+1) =	1.1596e+3 (7.53e+1)
DTLZ2	2	10	1.6119e−3 (4.97e−5) +	2.0605e−3 (8.99e−5) +	2.4825e−3 (1.65e−4) +	2.8149e−3 (1.55e−4)
	3	10	2.8983e−2 (4.03e−4) +	3.1300e−2 (7.08e−4) +	3.3755e−2 (1.25e−3) +	3.7209e−2 (1.39e−3)
	2	15	2.0534e−3 (8.42e−5) +	2.7597e−3 (1.68e−4) +	3.3224e−3 (2.86e−4) +	3.7096e−3 (3.37e−4)
	3	15	3.3585e−2 (9.57e−4) +	3.8020e−2 (1.43e−3) +	4.3887e−2 (3.41e−3) +	4.7173e−2 (2.56e−3)
	2	30	4.2561e−3 (4.05e−4) +	6.4776e−3 (7.78e−4) +	8.6079e−3 (1.51e−3) +	9.6315e−3 (1.53e−3)
	3	30	6.5736e−2 (6.69e−3) +	9.8970e−2 (2.02e−2) +	1.1387e−1 (2.45e−2) =	1.2563e−1 (2.56e−2)
	2	50	1.3574e−2 (2.22e−3) +	3.7317e−2 (1.23e−2) +	4.7406e−2 (1.74e−2) =	4.7077e−2 (1.41e−2)
	3	50	2.3184e−1 (4.56e−2) +	8.0474e−1 (3.92e−1) =	7.5884e−1 (3.06e−1) =	9.0249e−1 (3.79e−1)
DTLZ7	2	10	9.8359e−3 (1.51e−3) =	2.0743e−2 (5.96e−2) =	6.6836e−3 (1.62e−3) +	5.3642e−2 (1.32e−1)
	3	10	5.7445e−2 (3.41e−2) =	6.0245e−2 (4.11e−2) +	8.2064e−2 (1.26e−2) −	6.4189e−2 (4.18e−2)
	2	15	7.9948e−2 (1.65e−1) +	1.3797e−1 (2.03e−1) +	2.2131e−2 (7.28e−3) =	2.2330e−1 (2.19e−1)
	3	15	6.0500e−2 (3.60e−2) +	7.7117e−2 (5.39e−2) +	8.4347e−2 (7.27e−2) +	1.2445e−1 (8.75e−2)
	2	30	2.1170e−1 (2.20e−1) +	3.2023e−1 (1.89e−1) +	2.3700e−1 (2.06e−1) =	4.2848e−1 (2.40e−1)
	3	30	1.3408e−1 (7.29e−2) +	2.6232e−1 (1.21e−1) +	5.7991e−1 (2.09e−1) −	4.2241e−1 (2.35e−1)
	2	50	3.8510e−1 (1.36e−1) +	7.4459e−1 (3.84e−1) =	6.3779e−1 (4.18e−1) =	7.2501e−1 (4.07e−1)
	3	50	8.1394e−1 (3.09e−1) +	1.8058e+0 (7.38e−1) =	2.2826e+0 (3.49e−1) =	2.1154e+0 (7.94e−1)
ZDT1	2	10	1.3184e−3 (5.09e−5) +	1.5550e−3 (2.15e−4) +	1.7780e−3 (1.17e−4) +	2.0839e−3 (1.74e−4)
	2	15	1.4896e−3 (6.84e−5) +	1.8269e−3 (1.09e−4) +	2.2093e−3 (2.23e−4) +	2.5771e−3 (2.17e−4)
	2	30	2.5832e−3 (2.12e−4) +	3.5880e−3 (4.18e−4) +	4.3492e−3 (5.87e−4) +	5.1582e−3 (6.00e−4)
	2	50	5.0972e−3 (5.90e−4) +	8.8269e−3 (2.01e−3) +	1.2089e−2 (3.01e−3) =	1.3300e−2 (2.81e−3)
ZDT2	2	10	1.0644e−3 (2.69e−5) +	1.1999e−3 (4.37e−5) +	1.4134e−3 (6.23e−5) +	1.6400e−3 (8.92e−5)
	2	15	1.2004e−3 (6.69e−5) +	1.3795e−3 (9.62e−5) +	1.5965e−3 (8.78e−5) +	1.9434e−3 (1.51e−4)
	2	30	1.7183e−3 (1.59e−4) +	2.2287e−3 (3.16e−4) +	2.7661e−3 (3.91e−4) +	3.2757e−3 (3.67e−4)
	2	50	2.9302e−3 (3.54e−4) +	4.4114e−3 (7.37e−4) +	5.4924e−3 (6.83e−4) +	6.1095e−3 (9.12e−4)
± / =		30/0/2	25/0/7	14/2/16		

Bold values indicate better results than other compared algorithms

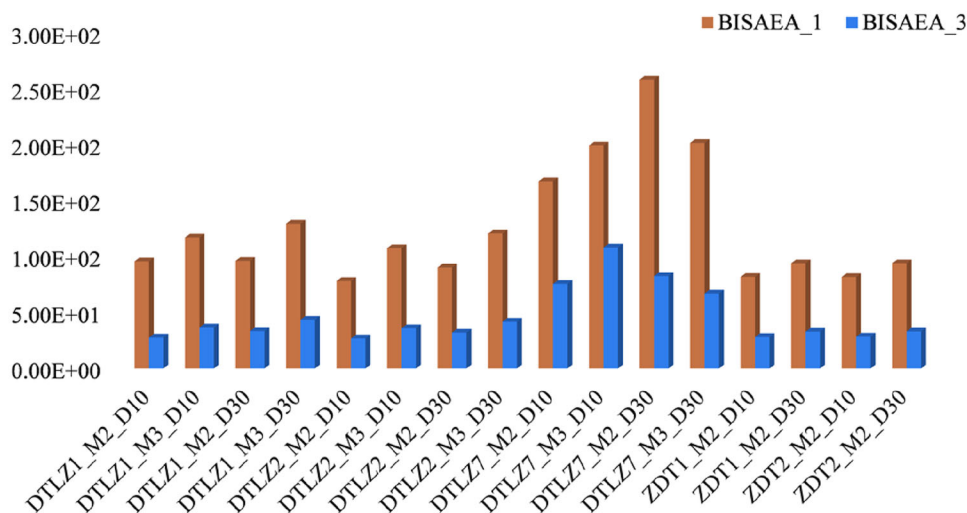
two algorithms in Figs. 8 and 9, both BISAEA(one) and BISAEA could converge in 10D, BISAEA has the better performance based on diversity. For the DTLZ2 on 30D, BISAEA has advantages in convergence and diversity. From the above results, it can be concluded that the Pareto-based bi-indicator strategy of BISAEA not only accelerates convergence but also play an important role in maintaining diversity.

Effect of the one-by-one selection strategy

To further study the role of the one-by-one selection strategy in BISAEA, we compare BISAEA with the BISAEA(pb) and BISAEA(only_one), where BISAEA(pb) only adopts the Pareto-based bi-indicator strategy, and BISAEA(only_one) uses the Pareto-based bi-indicator strategy and only one-time selection strategy to choose the same number new samples.

The experiments are conducted on DTLZ1, 2, 7 and ZDT 1, 2 problems with two and three objectives. The average IGD results of three algorithms based on 30 independent runs are

Fig. 7 The mean runtime with different problems on BISAEA_1 and BISAEA_3



is listed in Table 3, where the WRS test is also listed and the best results are highlighted.

It can be observed that the BISAEA and BISAEA(only_one) have better performance than BISAEA(pb). The reason is that BISAEA(pb) only selects the new samples through the Pareto-based bi-indicator strategy, the selected sample size cannot be effectively controlled, which will cause a waste of real function evaluation times.

To better explain the effects of the one-by-one selection strategy, the true PF and final non-dominated solutions achieved by BISAEA(pb), BISAEA(only_one) and BISAEA on 10D and 30D are shown in Fig. 10. It is obvious that the results of BISAEA(only_one) and BISAEA on 10D and 30D have the better convergence than BISAEA(pb), which shows that reselection after Pareto-based bi-indicator strategy could improve the convergence of the algorithm. From the results of DTLZ2 of BISAEA(only_one) and BISAEA on two and three objectives in Fig. 10, both BISAEA(only_one) and BISAEA could converge in true PFs, and the results of BISAEA has the better diversity. This can indicate that the one-by-one selection strategy is of great significance for the improvement of diversity.

Comparison with other algorithms

Results on DTLZ problems

The results of IGD values achieved by seven algorithms over 30 independent runs on DTLZ problems are summarized in Tables 4 and 5, where the best results are highlighted. Tables 4 and 5 show the statistical results of SAEAs and EAs in DTLZ1-7 with two and three objectives on 10D, 15D, 30D and 50D, respectively.

Both DTLZ1 and DTLZ3 have multi-model landscapes that are difficult to converge. BISAEA and KTA2 have superior performance on DTLZ1 and DTLZ3, followed by IBEA and K-RVEA. The true PF and final non-dominated solutions achieved by the compared algorithms on DTLZ1 associated with the median IGD values are shown in Fig. 11. The IGD values and the final solutions are illustrative of the convergence of BISAEA and KTA2. For DTLZ2, BISAEA achieves a satisfactory result. As shown in Fig. 12, both K-RVEA, KTA2 and BISAEA converge to the true PF, and the final non-dominated solutions achieved by BISAEA are more evenly distributed on the true PF. The reason is that the diversity indicator and one-by-one selection strategy could improve diversity.

DTLZ4 is modified from DTLZ2 and mainly used for measuring the diversity of algorithms. As shown in Tables 4 and 5, BISAEA and KTA2 are the top two algorithms for DTLZ4, BISAEA obtains the best average IGD values on 50D, and KTA2 gets the best results for other dimensions. The main reason is that one influential point-insensitive Kriging model is used in KTA2, which plays an important role in low-dimensional problems. However, the accuracy of Kriging models decreases with the increase of dimensions. The PFs of DTLZ5-7 are irregular, which brings a challenge to obtaining a set of diverse and well-converged solutions. Among them, the PFs of DTLZ5 and DTLZ6 have degenerated curves, DTLZ6 is modified from DTLZ5, and DTLZ7 is discontinuous. It can be seen from Tables 4 and 5, BISAEA achieve better results on these problems.

To better describe the performance of BISAEA in DTLZ series problems, the IGD values iteration process of BISAEA, K-RVEA, HeE-MOEA, EDNARMOEA and KTA2 of DTLZ2 is shown in Fig. 13. It is clear that BISAEA has the satisfactory convergence speed and performance. Especially, BISAEA and KTA2 have similar iteration

Table 2 Statistic results for IGD value obtained by BISAEA and BISAEA(one)

Problem	M	D	BISAEA(one)	BISAEA
DTLZ1	2	10	1.0362e+2 (2.37e+1) –	7.9292e+1 (7.83e+0)
	3	10	7.8871e+1 (1.57e+1) –	5.8816e+1 (5.09e+0)
	2	15	2.1190e+2 (3.61e+1) –	1.6472e+2 (3.83e+1)
	3	15	1.7847e+2 (2.82e+1) –	1.4423e+2 (3.80e+1)
	2	30	5.9339e+2 (5.83e+1) +	6.7235e+2 (1.14e+2)
	3	30	5.5222e+2 (4.75e+1) =	5.5668e+2 (5.95e+1)
	2	50	1.1759e+3 (6.56e+1) +	1.2407e+3 (1.09e+2)
	3	50	1.1293e+3 (8.18e+1) –	1.0685e+3 (7.43e+1)
DTLZ2	2	10	1.3219e–2 (2.07e–3) –	2.0605e–3 (8.99e–5)
	3	10	7.6350e–2 (6.79e–3) –	3.1300e–2 (7.08e–4)
	2	15	3.3959e–2 (5.70e–3) –	2.7597e–3 (1.68e–4)
	3	15	1.5031e–1 (1.65e–2) –	3.8020e–2 (1.43e–3)
	2	30	1.9201e–1 (3.27e–2) –	6.4776e–3 (7.78e–4)
	3	30	7.2481e–1 (1.13e–1) –	9.8970e–2 (2.02e–2)
	2	50	7.8446e–1 (1.11e–1) –	3.7317e–2 (1.23e–2)
	3	50	2.2359e+0 (1.91e–1) –	8.0474e–1 (3.92e–1)
DTLZ7	2	10	3.3689e–2 (9.68e–3) –	2.0743e–2 (5.96e–2)
	3	10	1.6703e–1 (3.35e–2) –	6.0245e–2 (4.11e–2)
	2	15	2.0450e–1 (1.45e–1) –	1.3797e–1 (2.03e–1)
	3	15	3.7979e–1 (5.60e–2) –	7.7117e–2 (5.39e–2)
	2	30	7.1570e–1 (1.31e–1) –	3.2023e–1 (1.89e–1)
	3	30	1.2629e+0 (2.41e–1) –	2.6232e–1 (1.21e–1)
	2	50	1.6216e+0 (3.19e–1) –	7.4459e–1 (3.84e–1)
	3	50	2.5973e+0 (4.25e–1) –	1.8058e+0 (7.38e–1)
ZDT1	2	10	2.3325e–2 (4.93e–3) –	1.5550e–3 (2.15e–4)
	2	15	7.0453e–2 (1.50e–2) –	1.8269e–3 (1.09e–4)
	2	30	2.4689e–1 (2.65e–2) –	3.5880e–3 (4.18e–4)
	2	50	4.7231e–1 (3.96e–2) –	8.8269e–3 (2.01e–3)
ZDT2	2	10	3.0129e–2 (8.29e–3) –	1.1999e–3 (4.37e–5)
	2	15	1.1128e–1 (2.50e–2) –	1.3795e–3 (9.62e–5)
	2	30	7.5782e–1 (1.88e–1) –	2.2287e–3 (3.16e–4)
	2	50	1.3336e+0 (1.85e–1) –	4.4114e–3 (7.37e–4)
± / =			2/29/1	

Bold values indicate better results than other compared algorithms

curves and performance, and the convergence speed of BISAEA wins over KTA2. In addition, the runtime of the above SAEAs is represented in Fig. 14. With the increase of design dimensions, the runtime of K-RVEA, EDNAR-MOEA and KTA2 increase dramatically, and HeE-MOEA and BISAEA has small changes. The main reason is that Kriging and EDN models are adopted in K-RVEA and KTA2, and EDNARMOEA, the training time of the Kriging model increases with the design dimension. Moreover, the GD values of the above SAEAs, EAs and BISAEA are shown in Tables 1 and 2 of the supplementary materials, and the runtime of four SAEAs and BISAEA are

shown in Table 5 of the supplementary materials. It can indicate that KTA2 and BISAEA have similar performance in DTLZ series problems, and BISAEA has advantages in runtime.

We can draw a conclusion from the above analysis, BISAEA and KTA2 can obtain competitive results in DTLZ series problems with two and three objectives. From the perspective of runtime, BISAEA has great advantages over KTA2. Therefore, it is obvious that BISAEA has better performance than the above comparison algorithms.

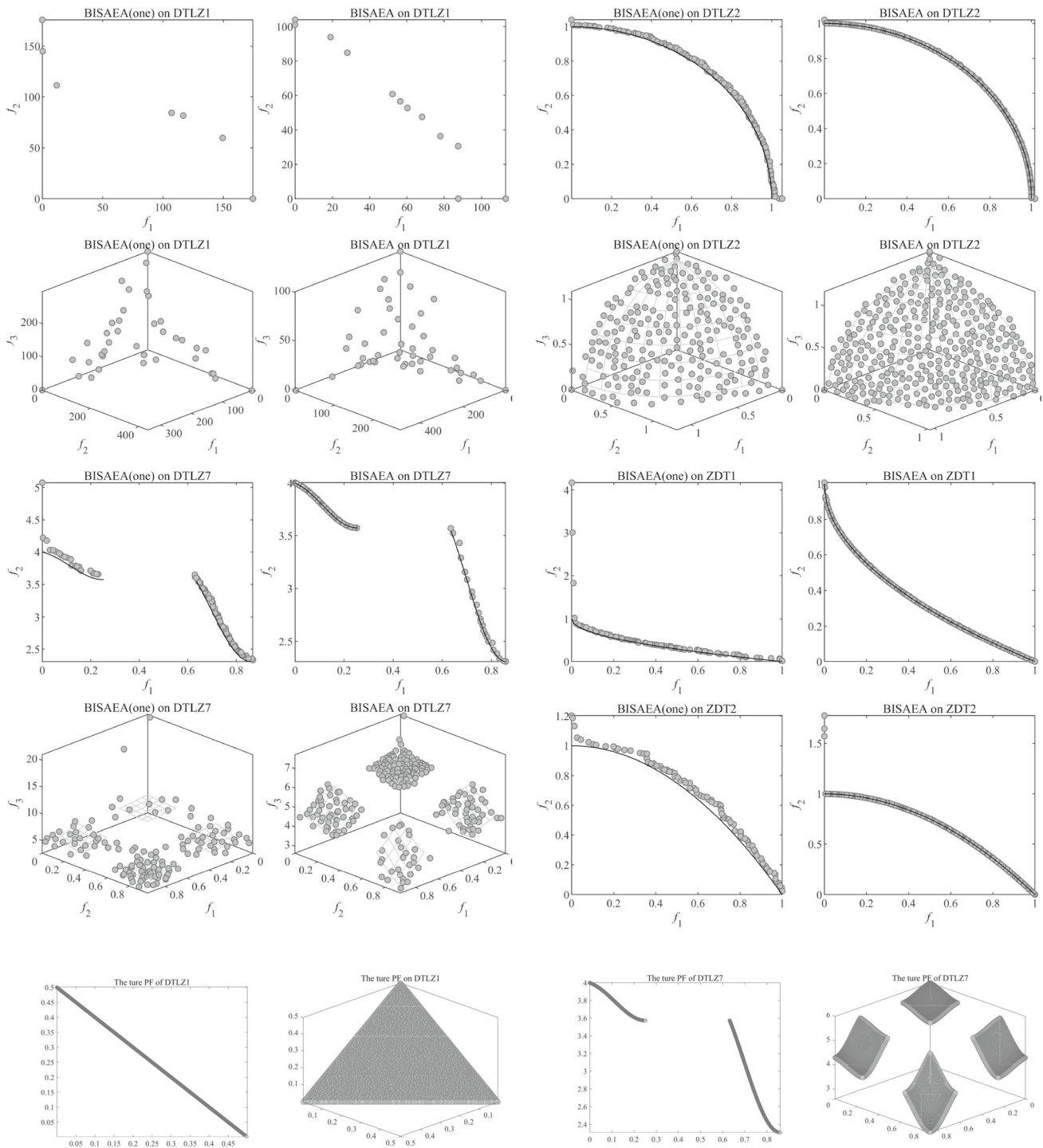


Fig. 8 The PF of DTLZ1,2,7 and ZDT 1,2 associated with the median IGD value on 10D

Results on ZDT problems

To further analyze the performance of BISAEA in two objectives problems, we use the ZDT problems as test problems. ZDT problems include six two-objective test problems, which introduce different difficulties for evolutionary

optimization[46]. We choose five unconstrained problems, referred to as ZDT1–ZDT4 and ZDT6. The statistical results of compared algorithms on ZDT problems are summarized in Tables 6 and 7.

From Tables 6 and 7, it is obvious that BISAEA has the best performance for ZDT1,2, 4 and 6. Both ZDT1 and ZDT4

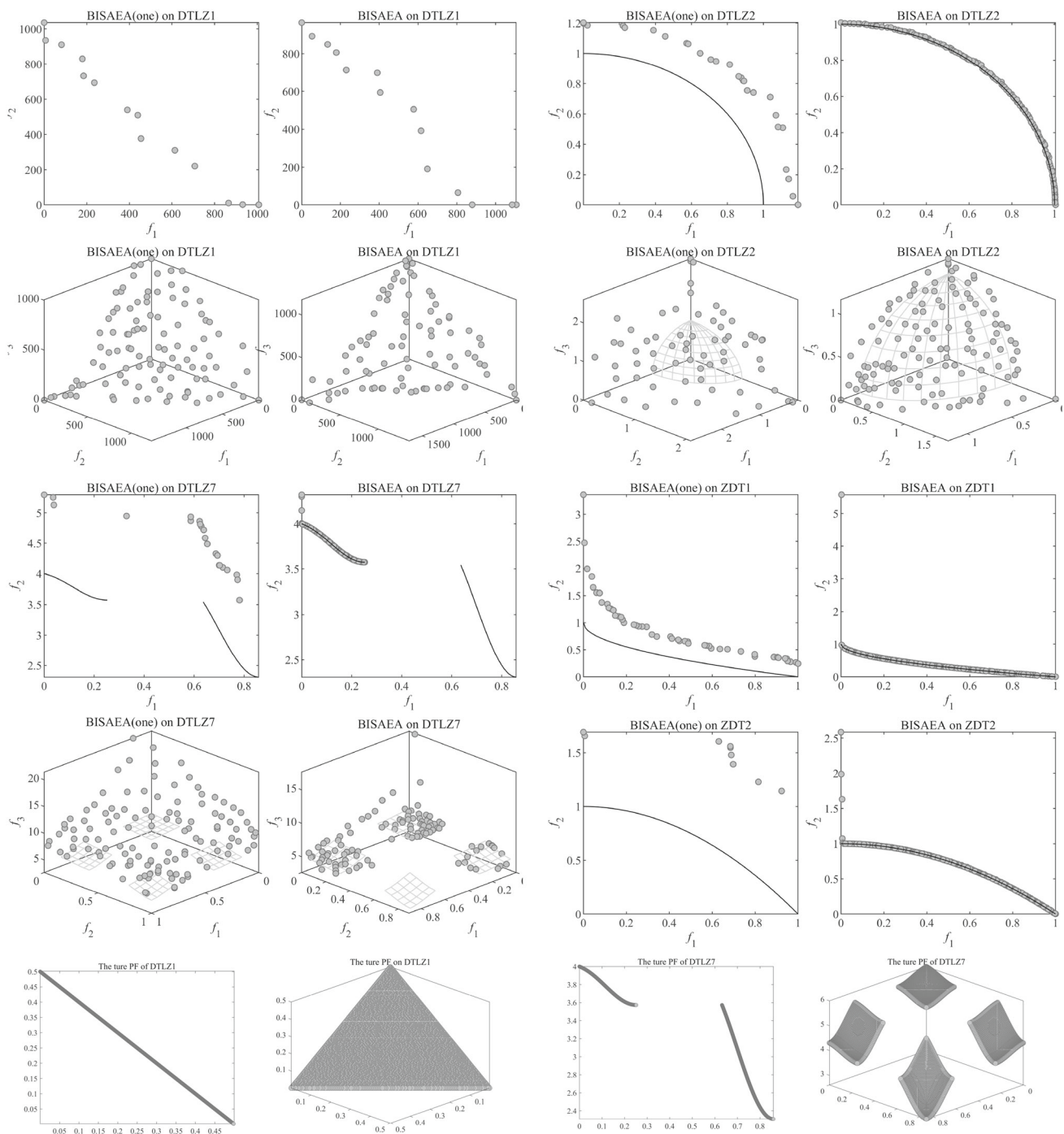


Fig. 9 The PF of DTLZ1,2,7 and ZDT 1,2 associated with the median IGD value on 30D

have convex PF, while ZDT4 is harder to converge. As shown in Fig. 15, only BISA EA and KTA2 can obtain the true PF completely both on ZDT1 and 2 in 30D, and K-RVEA can converge to the PF. Meanwhile, as shown in Table 6, with the dimensions increasing, KTA2 and K-RVEA deteriorate greatly both on convergence and diversity. The reason may be the dimension limitation of the Kriging model.

Then we focus on ZDT2 and ZDT6, which have concave PF, For ZDT2, only BISA EA and KTA2 can obtain the true PF completely on 30D in Fig. 16. The last discussions occur on ZDT3, whose disconnected PF brings a challenge for diversity. From the average IGD values, we find that BISA EA still has a competitive lead.

Table 3 Statistic results for IGD values obtained by BISAEA, BISAEA(pb) and BISAEA(only_one)

Problem	M	D	BISAEA(pb)	BISAEA(only_one)	BISAEA	
DTLZ1	2	10	9.3320e+1 (1.34e+1) –	8.0130e+1 (8.46e+0) =	7.9292e+1 (7.83e+0)	
	3	10	6.7067e+1 (8.95e+0) –	6.3521e+1 (7.57e+0) –	5.8816e+1 (5.09e+0)	
	2	15	1.9863e+2 (5.29e+1) –	1.6221e+2 (2.51e+1) =	1.6472e+2 (3.83e+1)	
	3	15	1.5434e+2 (3.37e+1) =	1.3718e+2 (3.12e+1) =	1.4423e+2 (3.80e+1)	
	2	30	7.2322e+2 (7.00e+1) =	6.6021e+2 (8.95e+1) =	6.7235e+2 (1.14e+2)	
	3	30	6.0511e+2 (5.26e+1) –	5.4264e+2 (6.88e+1) =	5.5668e+2 (5.95e+1)	
	2	50	1.3416e+3 (1.27e+2) –	1.2781e+3 (1.39e+2) =	1.2407e+3 (1.09e+2)	
	3	50	1.1921e+3 (7.65e+1) –	1.1194e+3 (8.59e+1) –	1.0685e+3 (7.43e+1)	
	DTLZ2	2	10	4.3133e−3 (5.24e−4) –	2.6222e−3 (1.65e−4) –	2.0605e−3 (8.99e−5)
		3	10	4.8984e−2 (2.45e−3) –	3.5168e−2 (8.21e−4) –	3.1300e−2 (7.08e−4)
2		15	4.6816e−3 (4.30e−4) –	3.2685e−3 (2.50e−4) –	2.7597e−3 (1.68e−4)	
3		15	5.8785e−2 (2.82e−3) –	4.1540e−2 (1.41e−3) –	3.8020e−2 (1.43e−3)	
2		30	1.1016e−2 (1.42e−3) –	7.3748e−3 (1.05e−3) –	6.4776e−3 (7.78e−4)	
3		30	1.3500e−1 (2.95e−2) –	9.3773e−2 (1.17e−2) =	9.8970e−2 (2.02e−2)	
2		50	5.8195e−2 (1.87e−2) –	3.7896e−2 (1.78e−2) =	3.7317e−2 (1.23e−2)	
3		50	8.6123e−1 (2.50e−1) =	6.7043e−1 (2.60e−1) =	8.0474e−1 (3.92e−1)	
DTLZ7		2	10	6.8003e−2 (1.49e−1) =	9.6600e−2 (1.76e−1) =	2.0743e−2 (5.96e−2)
		3	10	8.4870e−2 (5.86e−2) –	5.8671e−2 (4.29e−2) =	6.0245e−2 (4.11e−2)
	2	15	1.6813e−1 (2.12e−1) –	2.3919e−1 (2.21e−1) =	1.3797e−1 (2.03e−1)	
	3	15	9.9394e−2 (7.26e−2) –	9.1497e−2 (9.64e−2) =	7.7117e−2 (5.39e−2)	
	2	30	3.7272e−1 (2.01e−1) –	3.8673e−1 (2.06e−1) =	3.2023e−1 (1.89e−1)	
	3	30	4.1288e−1 (1.68e−1) –	3.0747e−1 (1.59e−1) =	2.6232e−1 (1.21e−1)	
	2	50	8.0160e−1 (5.24e−1) =	6.9216e−1 (3.53e−1) =	7.4459e−1 (3.84e−1)	
	3	50	1.9138e+0 (6.43e−1) =	1.4943e+0 (6.07e−1) =	1.8058e+0 (7.38e−1)	
	ZDT1	2	10	3.1413e−3 (3.30e−4) –	1.8144e−3 (1.24e−4) –	1.5550e−3 (2.15e−4)
		2	15	3.5626e−3 (4.14e−4) –	2.2269e−3 (3.30e−4) –	1.8269e−3 (1.09e−4)
2		30	6.3949e−3 (9.24e−4) –	3.7603e−3 (5.03e−4) =	3.5880e−3 (4.18e−4)	
2		50	1.4872e−2 (2.68e−3) –	9.6685e−3 (2.01e−3) =	8.8269e−3 (2.01e−3)	
ZDT2	2	10	2.8182e−3 (3.42e−4) –	1.4728e−3 (5.57e−5) –	1.1999e−3 (4.37e−5)	
	2	15	2.9049e−3 (2.62e−4) –	1.6952e−3 (8.53e−5) –	1.3795e−3 (9.62e−5)	
	2	30	4.1529e−3 (3.83e−4) –	2.5173e−3 (2.23e−4) –	2.2287e−3 (3.16e−4)	
	2	50	7.8141e−3 (1.37e−3) –	4.7115e−3 (8.24e−4) =	4.4114e−3 (7.37e−4)	
± / =			0/26/6	0/12/20		

Bold values indicate better results than other compared algorithms

The IGD values iteration process is shown in Fig. 17, it is clear that BISAEA and KTA2 can obtain the desired IGD values quickly (less than 150 NFes). Moreover, the GD values of four SAEAs, three EAs and BISAEA are shown in Tables 3 and 4 of the supplementary materials, and the runtime of four SAEAs and BISAEA are represented in Table 6 of the supplementary materials, which also indicates the superior performance of BISAEA.

Application to engineering problem

Engineering problem description

As a new type of marine equipment, blend-wing-body underwater gliders (BWBUGs) have been used for ocean observation [48, 49]. BWBUGs adopt a smooth connection between their bodies and wings [50–54]. BWBUGs is a complex multidisciplinary system involving shape, skeleton, pressure cabins, and other disciplines. The shape and skeleton design are important parts of the BWBUGs system

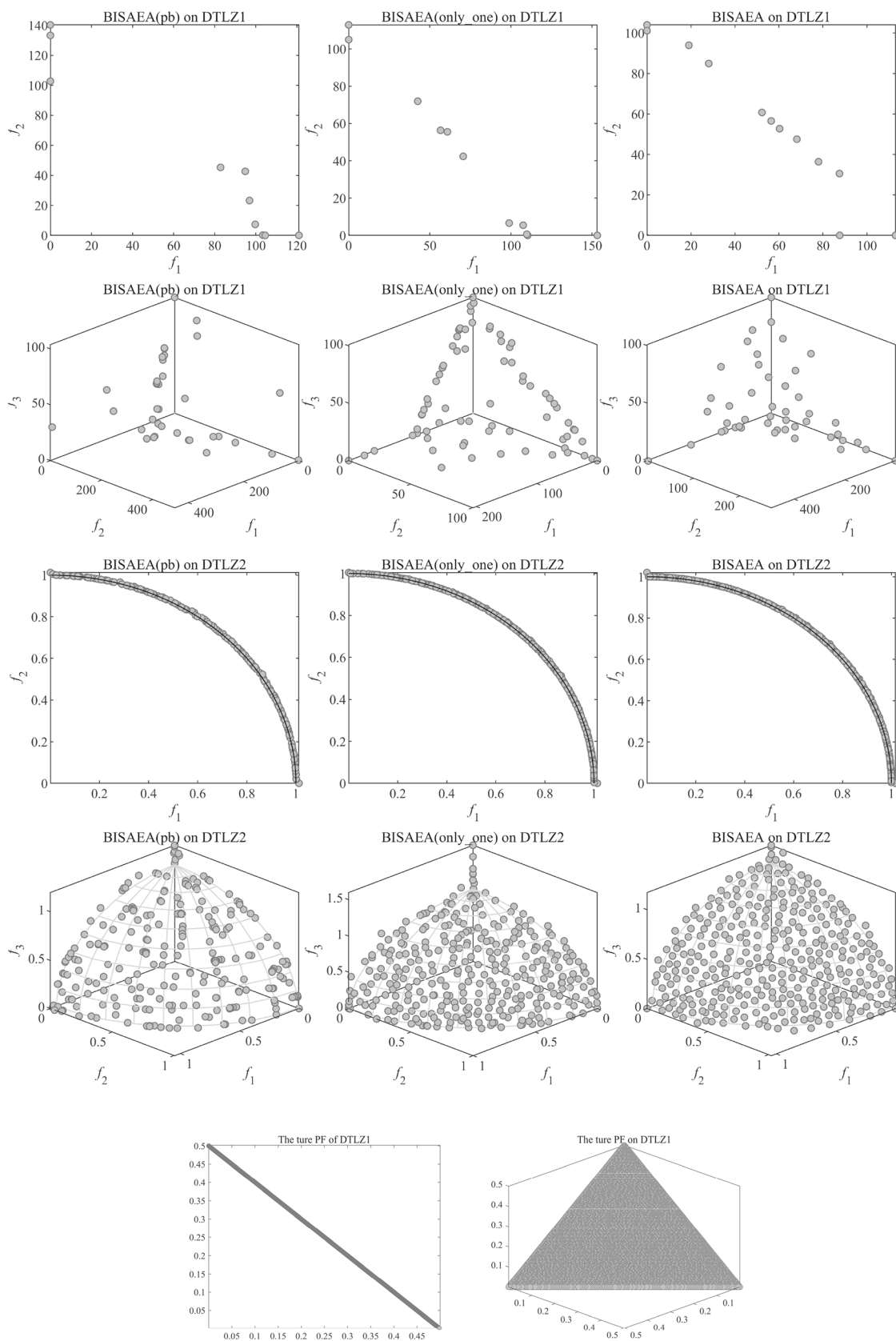


Fig. 10 The PF of DTLZ1,2 associated with the median IGD value on 10D

Table 4 Statistic results of the four SAEAs and BISAEA on DTLZ1-7

Problem	M	D	HeEMOEA	EDNARMOEA	KRVEA	KTA2	BISAEA
DTLZ1	2	10	1.3409e+2 (2.38e+1) –	1.3750e+2 (2.11e+1) –	6.4500e+1 (1.41e+1) +	6.1609e+1 (2.69e+1) +	7.9292e+1 (7.83e+0)
	3	10	1.0111e+2 (2.59e+1) –	8.7901e+1 (1.95e+1) –	7.5653e+1 (1.21e+1) –	3.4060e+1 (1.29e+1) +	5.8816e+1 (5.09e+0)
	2	15	2.9125e+2 (3.70e+1) –	2.7473e+2 (2.39e+1) –	1.4602e+2 (2.51e+1) =	1.5740e+2 (3.25e+1) =	1.6472e+2 (3.83e+1)
	3	15	2.2438e+2 (2.89e+1) –	2.0286e+2 (2.14e+1) –	1.7307e+2 (1.90e+1) –	1.0616e+2 (3.66e+1) +	1.4423e+2 (3.80e+1)
	2	30	7.4978e+2 (5.51e+1) –	7.3458e+2 (5.97e+1) –	6.4423e+2 (5.74e+1) =	4.3683e+2 (8.43e+1) +	6.7235e+2 (1.14e+2)
	3	30	6.1860e+2 (4.10e+1) –	5.8838e+2 (3.74e+1) –	5.7579e+2 (5.11e+1) =	3.9865e+2 (5.54e+1) +	5.5668e+2 (5.95e+1)
	2	50	1.3854e+3 (8.17e+1) –	1.3965e+3 (7.37e+1) –	1.3896e+3 (9.23e+1) –	9.2188e+2 (1.08e+2) +	1.2407e+3 (1.09e+2)
	3	50	1.1711e+3 (5.12e+1) –	1.1848e+3 (5.74e+1) –	1.1873e+3 (7.84e+1) –	8.4456e+2 (7.92e+1) +	1.0685e+3 (7.43e+1)
DTLZ2	2	10	1.2967e–1 (1.83e–2) –	1.2053e–1 (3.53e–2) –	2.0367e–2 (2.29e–3) –	3.3694e–3 (3.05e–4) –	2.0605e–3 (8.99e–5)
	3	10	1.6947e–1 (1.26e–2) –	2.7634e–1 (2.47e–2) –	1.0249e–1 (1.57e–2) –	4.5212e–2 (1.64e–3) –	3.1300e–2 (7.08e–4)
	2	15	1.8129e–1 (2.48e–2) –	2.3988e–1 (3.46e–2) –	3.9598e–2 (7.12e–3) –	5.0391e–3 (4.96e–4) –	2.7597e–3 (1.68e–4)
	3	15	2.2358e–1 (1.20e–2) –	4.9724e–1 (4.64e–2) –	3.3699e–1 (5.27e–2) –	5.7977e–2 (3.07e–3) –	3.8020e–2 (1.43e–3)
	2	30	3.2966e–1 (3.02e–2) –	1.0737e+0 (1.41e–1) –	1.1359e+0 (1.72e–1) –	2.8065e–1 (1.06e–1) –	6.4776e–3 (7.78e–4)
	3	30	3.9421e–1 (3.38e–2) –	1.4456e+0 (1.02e–1) –	1.4185e+0 (9.61e–2) –	2.8956e–1 (6.81e–2) –	9.8970e–2 (2.02e–2)
	2	50	2.6744e+0 (1.19e–1) –	2.7307e+0 (9.85e–2) –	2.6724e+0 (1.21e–1) –	7.0263e–1 (1.53e–1) –	3.7317e–2 (1.23e–2)
	3	50	2.6972e+0 (1.33e–1) –	2.6590e+0 (1.21e–1) –	2.7440e+0 (1.17e–1) –	1.2306e+0 (2.14e–1) –	8.0474e–1 (3.92e–1)
DTLZ3	2	10	3.3926e+2 (5.32e+1) –	3.6902e+2 (5.39e+1) –	1.3702e+2 (3.30e+1) +	1.4734e+2 (4.81e+1) +	1.9313e+2 (1.07e+1)
	3	10	2.5033e+2 (6.88e+1) –	3.0166e+2 (5.17e+1) –	2.0448e+2 (3.16e+1) –	9.5365e+1 (3.41e+1) +	1.7084e+2 (1.44e+1)
	2	15	6.9527e+2 (8.95e+1) –	7.4491e+2 (7.81e+1) –	3.6291e+2 (6.86e+1) =	3.4983e+2 (9.44e+1) =	3.6953e+2 (6.64e+1)
	3	15	5.9585e+2 (9.91e+1) –	6.3873e+2 (1.03e+2) –	4.6531e+2 (7.03e+1) –	3.0228e+2 (8.04e+1) =	3.5118e+2 (7.26e+1)
	2	30	1.9496e+3 (1.07e+2) –	1.9836e+3 (1.49e+2) –	1.6836e+3 (1.83e+2) –	1.1065e+3 (1.74e+2) +	1.5176e+3 (2.67e+2)
	3	30	1.8451e+3 (1.51e+2) –	1.8830e+3 (1.84e+2) –	1.6701e+3 (1.64e+2) =	1.1331e+3 (1.41e+2) +	1.5755e+3 (2.44e+2)
	2	50	3.8596e+3 (1.67e+2) –	3.7981e+3 (1.99e+2) –	3.8093e+3 (1.31e+2) –	2.2774e+3 (2.17e+2) +	3.2828e+3 (3.00e+2)
	3	50	3.7271e+3 (1.66e+2) –	3.7301e+3 (1.98e+2) –	3.7282e+3 (1.76e+2) –	2.4735e+3 (2.28e+2) +	3.4988e+3 (2.18e+2)
DTLZ4	2	10	6.5254e–1 (1.10e–1) –	2.7033e–1 (2.99e–1) =	2.4168e–1 (2.71e–1) =	1.9006e–1 (3.10e–1) +	2.6025e–1 (2.73e–1)

Table 4 (continued)

Problem	M	D	HeEMOEA	EDNARMOEA	KRVEA	KTA2	BISAEA
DTLZ5	3	10	7.7377e−1 (1.09e−1) −	2.1134e−1 (7.39e−2) +	3.0780e−1 (8.12e−2) −	1.6870e−1 (9.22e−2) +	2.6384e−1 (7.66e−2)
	2	15	7.4173e−1 (6.39e−2) −	3.4113e−1 (2.17e−1) =	3.0043e−1 (1.16e−1) =	1.9603e−1 (3.07e−1) +	4.8457e−1 (2.96e−1)
	3	15	9.1736e−1 (6.34e−2) −	3.8305e−1 (9.75e−2) =	5.7510e−1 (1.02e−1) −	3.0615e−1 (1.50e−1) =	3.5543e−1 (1.01e−1)
	2	30	9.1738e−1 (3.78e−2) −	1.0087e+0 (1.97e−1) −	1.4127e+0 (2.01e−1) −	2.5725e−1 (1.03e−1) +	6.3280e−1 (2.90e−1)
	3	30	1.0979e+0 (2.20e−2) −	1.3836e+0 (1.46e−1) −	1.4999e+0 (2.00e−1) −	8.1231e−1 (1.56e−1) =	8.3388e−1 (1.80e−1)
	2	50	3.0126e+0 (1.35e−1)	2.9996e+0 (1.23e−1)	2.9724e+0 (1.79e−1)	1.1960e+0 (2.69e−1) −	1.0947e+0 (4.37e−1)
	3	50	3.0709e+0 (1.39e−1) −	3.0493e+0 (1.74e−1) −	3.0706e+0 (1.48e−1) −	2.5623+0(2.36e−1) −	1.8947e+0 (3.71e−1)
	2	10	1.2715e−1 (2.44e−2) −	1.2490e−1 (3.67e−2) −	2.0320e−2 (2.47e−3) −	3.3111e−3 (2.25e−4) −	2.0776e−3 (6.19e−5)
	3	10	1.2023e−1 (2.09e−2) −	1.3876e−1 (3.13e−2) −	5.9025e−2 (1.16e−2) −	4.1243e−3 (3.11e−4) +	6.6947e−3 (7.54e−4)
	2	15	1.8143e−1 (2.99e−2) −	2.5536e−1 (5.19e−2) −	3.7728e−2 (6.60e−3) −	1.6142e−2 (6.16e−2) −	2.7275e−3 (1.84e−4)
DTLZ6	3	15	1.6186e−1 (2.51e−2) −	3.4360e−1 (4.53e−2) −	2.4227e−1 (5.15e−2) −	1.1234e−2 (2.21e−3) +	1.2950e−2 (1.98e−3)
	2	30	3.2941e−1 (2.59e−2) −	1.0677e+0 (1.27e−1) −	1.0505e+0 (2.26e−1) −	2.6262e−1 (1.19e−1) −	6.6529e−3 (9.90e−4)
	3	30	3.0025e−1 (2.71e−2) −	1.3488e+0 (7.25e−2) −	1.2876e+0 (1.18e−1) −	2.9689e−1 (7.06e−2) −	5.1419e−2 (1.77e−2)
	2	50	2.7186e+0 (1.40e−1) −	2.6881e+0 (1.49e−1) −	2.6498e+0 (1.31e−1) −	6.6110e−1 (1.11e−1) −	3.5339e−2 (1.21e−2)
	3	50	2.6686e+0 (1.44e−1) −	2.6602e+0 (1.16e−1) −	2.6620e+0 (1.51e−1) −	1.1191e+0 (1.87e−1) −	4.8702e−1 (2.36e−1)
	2	10	7.4777e+0 (1.89e−1) −	4.6700e+0 (5.11e−1) −	2.7382e+0 (5.30e−1) −	1.6696e+0 (5.16e−1) −	1.2833e+0 (6.16e−1)
	3	10	6.6758e+0 (1.36e−1) −	5.0008e+0 (4.32e−1) −	2.4475e+0 (4.53e−1) −	9.3281e−1 (3.38e−1) =	1.1076e+0 (4.22e−1)
	2	15	1.1804e+1 (2.45e−1) −	8.6550e+0 (4.65e−1) −	5.7639e+0 (6.60e−1) −	3.4433e+0 (8.35e−1) =	2.9998e+0 (8.80e−1)
	3	15	1.0979e+1 (1.54e−1) −	9.5937e+0 (3.62e−1) −	5.4353e+0 (5.73e−1) −	2.8091e+0 (6.36e−1) +	3.2537e+0 (7.91e−1)
	2	30	2.4999e+1 (1.78e−1) −	2.2192e+1 (7.25e−1) −	2.0118e+1 (1.14e+0) −	1.1491e+1 (1.17e+0) =	1.2193e+1 (2.05e+0)
DTLZ7	3	30	2.4093e+1 (1.81e−1) −	2.3004e+1 (3.63e−1) −	1.8396e+1 (1.24e+0) =	1.3571e+1 (1.09e+0) +	1.8120e+1 (1.54e+0)
	2	50	4.2689e+1 (2.33e−1) −	4.2664e+1 (1.77e−1) −	4.2717e+1 (2.54e−1) −	2.7917e+1 (1.53e+0) +	3.7312e+1 (2.62e+0)
	3	50	4.1847e+1 (2.64e−1) −	4.1895e+1 (2.96e−1) −	4.1816e+1 (2.23e−1) −	3.0710e+1 (1.23e+0) +	3.8086e+1 (9.05e−1)
	2	10	2.4596e+0 (3.52e−1) −	3.8341e−1 (2.18e−1) −	1.4410e−2 (1.31e−3) +	1.6487e−1 (2.15e−1) =	2.0743e−2 (5.96e−2)
	3	10	3.6692e+0 (7.47e−1) −	9.6622e−1 (4.65e−1) −	9.7619e−2 (1.17e−2) −	3.9541e−1 (2.93e−1) −	6.0245e−2 (4.11e−2)

Table 4 (continued)

Problem	M	D	HeEMOEA	EDNARMOEA	KRVEA	KTA2	BISAEA
	2	15	3.3430e+0 (2.61e−1) −	6.9979e−1 (2.73e−1) −	2.0047e−2 (2.31e−3) +	1.3559e−1 (2.05e−1) +	1.3797e−1 (2.03e−1)
	3	15	5.4267e+0 (5.40e−1) −	1.4048e+0 (5.03e−1) −	1.1011e−1 (1.96e−2) −	4.8675e−1 (2.52e−1) −	7.7117e−2 (5.39e−2)
	2	30	4.7311e+0 (2.53e−1) −	1.5376e+0 (4.88e−1) −	6.9000e−2 (1.08e−2) +	9.3187e−2 (1.78e−1) +	3.2023e−1 (1.89e−1)
	3	30	7.6319e+0 (4.90e−1) −	2.8999e+0 (6.97e−1) −	2.2106e−1 (3.22e−2) =	4.3813e−1 (2.67e−1) −	2.6232e−1 (1.21e−1)
	2	50	6.2358e+0 (3.06e−1) −	6.1476e+0 (2.69e−1) −	6.2055e+0 (2.08e−1) −	2.1595e−1 (7.70e−1) +	7.4459e−1 (3.84e−1)
	3	50	9.3212e+0 (5.28e−1) −	9.4918e+0 (4.14e−1) −	9.2960e+0 (4.09e−1) −	4.9007e−1 (2.74e−1) +	1.8058e+0 (7.38e−1)
± / =			0/56/0	1/52/3	4/44/8	27/20/9	

Bold values indicate better results than other compared algorithms

Table 5 Statistic results of the three EAs and BISAEA on DTLZ1-7

Problem	M	D	IBEA	NSGAIII	RVEA	BISAEA
DTLZ1	2	10	8.9705e+1 (1.92e+1) −	1.1169e+2 (2.03e+1) −	1.1403e+2 (2.08e+1) −	7.9292e+1 (7.83e+0)
	3	10	6.1169e+1 (1.49e+1) =	7.5513e+1 (1.79e+1) −	7.3201e+1 (1.74e+1) −	5.8816e+1 (5.09e+0)
	2	15	2.0436e+2 (3.59e+1) −	2.3368e+2 (3.25e+1) −	2.4955e+2 (3.97e+1) −	1.6472e+2 (3.83e+1)
	3	15	1.6065e+2 (2.33e+1) =	1.8778e+2 (2.59e+1) −	1.8539e+2 (2.83e+1) −	1.4423e+2 (3.80e+1)
	2	30	6.4079e+2 (4.11e+1) +	6.8150e+2 (5.00e+1) =	7.0809e+2 (5.84e+1) =	6.7235e+2 (1.14e+2)
	3	30	5.0965e+2 (4.84e+1) +	5.6609e+2 (4.31e+1) =	5.7020e+2 (4.88e+1) =	5.5668e+2 (5.95e+1)
	2	50	1.2267e+3 (6.33e+1) =	1.3201e+3 (7.51e+1) −	1.3405e+3 (1.19e+2) −	1.2407e+3 (1.09e+2)
DTLZ2	3	50	1.0508e+3 (6.16e+1) =	1.0957e+3 (5.82e+1) =	1.1288e+3 (6.85e+1) −	1.0685e+3 (7.43e+1)
	2	10	1.6697e−1 (2.13e−2) −	2.1632e−1 (3.07e−2) −	2.4720e−1 (3.77e−2) −	2.0605e−3 (8.99e−5)
	3	10	2.0839e−1 (2.00e−2) −	2.4287e−1 (2.26e−2) −	2.7418e−1 (2.53e−2) −	3.1300e−2 (7.08e−4)
	2	15	3.5507e−1 (4.06e−2) −	4.1835e−1 (4.68e−2) −	4.6346e−1 (5.10e−2) −	2.7597e−3 (1.68e−4)
	3	15	3.9019e−1 (4.18e−2) −	4.3309e−1 (4.09e−2) −	4.7060e−1 (5.67e−2) −	3.8020e−2 (1.43e−3)
	2	30	1.0386e+0 (1.05e−1) −	1.2409e+0 (1.14e−1) −	1.2846e+0 (1.02e−1) −	6.4776e−3 (7.78e−4)
	3	30	1.1158e+0 (7.43e−2) −	1.2255e+0 (9.33e−2) −	1.2486e+0 (8.47e−2) −	9.8970e−2 (2.02e−2)
DTLZ3	2	50	2.2410e+0 (1.08e−1) −	2.4751e+0 (1.54e−1) −	2.5457e+0 (1.51e−1) −	3.7317e−2 (1.23e−2)
	3	50	2.2662e+0 (1.51e−1) −	2.4563e+0 (1.38e−1) −	2.5064e+0 (1.56e−1) −	8.0474e−1 (3.92e−1)
	2	10	2.3746e+2 (5.48e+1) −	2.8508e+2 (5.72e+1) −	2.9069e+2 (5.41e+1) −	1.9313e+2 (1.07e+1)
	3	10	1.9465e+2 (3.81e+1) −	2.2716e+2 (4.31e+1) −	2.0111e+2 (4.23e+1) −	1.7084e+2 (1.44e+1)
	2	15	5.3677e+2 (9.09e+1) −	6.0598e+2 (9.92e+1) −	6.4872e+2 (8.90e+1) −	3.6953e+2 (6.64e+1)
	3	15	4.6432e+2 (8.14e+1) −	5.2688e+2 (7.72e+1) −	5.1939e+2 (7.74e+1) −	3.5118e+2 (7.26e+1)
	2	30	1.6690e+3 (1.02e+2) −	1.8324e+3 (1.24e+2) −	1.8855e+3 (1.18e+2) −	1.5176e+3 (2.67e+2)
DTLZ4	3	30	1.5921e+3 (1.20e+2) =	1.6545e+3 (1.47e+2) =	1.6994e+3 (1.49e+2) =	1.5755e+3 (2.44e+2)
	2	50	3.3337e+3 (2.07e+2) =	3.5211e+3 (1.81e+2) −	3.6072e+3 (2.05e+2) −	3.2828e+3 (3.00e+2)
	3	50	3.2769e+3 (1.77e+2) +	3.4302e+3 (1.67e+2) =	3.4019e+3 (2.07e+2) =	3.4988e+3 (2.18e+2)
	2	10	4.4127e−1 (2.52e−1) −	4.4991e−1 (1.44e−1) −	2.8553e−1 (7.51e−2) −	2.6025e−1 (2.73e−1)
	3	10	4.6196e−1 (1.20e−1) −	5.4432e−1 (1.11e−1) −	3.1638e−1 (7.13e−2) −	2.6384e−1 (7.66e−2)
	2	15	6.5213e−1 (2.31e−1) −	6.8995e−1 (1.39e−1) =	5.0468e−1 (1.26e−1) =	4.8457e−1 (2.96e−1)

Table 5 (continued)

Problem	M	D	IBEA	NSGAIII	RVEA	BISAEA
DTLZ5	3	15	6.8419e−1 (1.30e−1) –	7.6928e−1 (1.41e−1) –	5.3273e−1 (9.05e−2) –	3.5543e−1 (1.01e−1)
	2	30	1.3396e+0 (1.56e−1) –	1.5413e+0 (1.40e−1) –	1.3186e+0 (1.34e−1) –	6.3280e−1 (2.90e−1)
	3	30	1.3947e+0 (1.34e−1) –	1.5587e+0 (1.58e−1) –	1.2860e+0 (1.38e−1) –	8.3388e−1 (1.80e−1)
	2	50	2.4330e+0 (1.99e−1) –	2.7520e+0 (1.72e−1) –	2.4952e+0 (2.32e−1) –	1.0947e+0 (4.37e−1)
	3	50	2.4970e+0 (1.71e−1) –	2.7413e+0 (1.95e−1) –	2.5091e+0 (1.94e−1) –	1.8947e+0 (3.71e−1)
	2	10	1.6975e−1 (2.47e−2) –	2.1532e−1 (2.67e−2) –	2.5035e−1 (3.47e−2) –	2.0776e−3 (6.19e−5)
	3	10	1.4922e−1 (2.60e−2) –	1.7030e−1 (2.52e−2) –	2.1576e−1 (3.21e−2) –	6.6947e−3 (7.54e−4)
	2	15	3.4219e−1 (2.61e−2) –	4.2387e−1 (4.96e−2) –	4.6903e−1 (6.37e−2) –	2.7275e−3 (1.84e−4)
	3	15	3.1667e−1 (4.34e−2) –	3.8163e−1 (5.27e−2) –	4.1837e−1 (5.33e−2) –	1.2950e−2 (1.98e−3)
	2	30	1.0414e+0 (1.03e−1) –	1.2638e+0 (7.65e−2) –	1.2921e+0 (9.26e−2) –	6.6529e−3 (9.90e−4)
DTLZ6	3	30	1.0649e+0 (1.05e−1) –	1.1770e+0 (1.07e−1) –	1.2001e+0 (1.26e−1) –	5.1419e−2 (1.77e−2)
	2	50	2.2324e+0 (1.25e−1) –	2.4856e+0 (1.25e−1) –	2.5112e+0 (1.45e−1) –	3.5339e−2 (1.21e−2)
	3	50	2.2331e+0 (1.57e−1) –	2.3818e+0 (2.03e−1) –	2.4091e+0 (1.88e−1) –	4.8702e−1 (2.36e−1)
	2	10	5.9682e+0 (3.37e−1) –	6.5187e+0 (3.80e−1) –	6.6901e+0 (3.00e−1) –	1.2833e+0 (6.16e−1)
	3	10	5.2989e+0 (4.28e−1) –	5.6166e+0 (3.34e−1) –	5.8568e+0 (3.11e−1) –	1.1076e+0 (4.22e−1)
	2	15	1.0218e+1 (3.34e−1) –	1.0687e+1 (3.92e−1) –	1.1094e+1 (2.61e−1) –	2.9998e+0 (8.80e−1)
	3	15	9.5146e+0 (4.52e−1) –	1.0039e+1 (3.63e−1) –	1.0232e+1 (4.00e−1) –	3.2537e+0 (7.91e−1)
	2	30	2.3092e+1 (5.40e−1) –	2.3733e+1 (3.89e−1) –	2.4277e+1 (4.13e−1) –	1.2193e+1 (2.05e+0)
	3	30	2.2480e+1 (4.91e−1) –	2.3092e+1 (4.62e−1) –	2.3318e+1 (4.74e−1) –	1.8120e+1 (1.54e+0)
	2	50	4.0453e+1 (6.06e−1) –	4.1096e+1 (5.72e−1) –	4.1778e+1 (4.60e−1) –	3.7312e+1 (2.62e+0)
DTLZ7	3	50	4.0341e+1 (5.13e−1) –	4.0755e+1 (4.30e−1) –	4.0834e+1 (5.06e−1) –	3.8086e+1 (9.05e−1)
	2	10	2.4935e+0 (4.85e−1) –	2.9541e+0 (5.28e−1) –	2.5411e+0 (6.42e−1) –	2.0743e−2 (5.96e−2)
	3	10	3.5856e+0 (7.75e−1) –	3.9470e+0 (9.24e−1) –	2.6243e+0 (9.71e−1) –	6.0245e−2 (4.11e−2)
	2	15	3.2111e+0 (7.17e−1) –	4.0825e+0 (5.41e−1) –	3.5442e+0 (5.01e−1) –	1.3797e−1 (2.03e−1)
	3	15	5.3629e+0 (7.94e−1) –	5.5426e+0 (1.04e+0) –	4.3252e+0 (7.94e−1) –	7.7117e−2 (5.39e−2)
	2	30	4.6749e+0 (5.09e−1) –	5.1326e+0 (4.72e−1) –	5.0683e+0 (4.37e−1) –	3.2023e−1 (1.89e−1)
	3	30	7.1924e+0 (6.81e−1) –	7.5326e+0 (6.63e−1) –	6.4971e+0 (7.41e−1) –	2.6232e−1 (1.21e−1)
	2	50	5.5111e+0 (2.59e−1) –	5.7764e+0 (3.71e−1) –	5.6319e+0 (4.14e−1) –	7.4459e−1 (3.84e−1)
	3	50	8.2928e+0 (6.01e−1) –	8.7589e+0 (5.62e−1) –	8.1387e+0 (6.35e−1) –	1.8058e+0 (7.38e−1)
	± / =		3/47/6	0/50/6	0/51/5	

Bold values indicate better results than other compared algorithms

design. For the multidisciplinary design for shape and skeleton, a larger lift-drag ratio (L/D), smaller skeleton volume ratio (V_s/V) and better skeleton strength (σ_s) are expected. The L/D is affected by the shape, angle of attack (AOA) and velocity. The AOA and velocity are set as 2° and 1.028 m/s in this article. The larger L/D means the glider has a larger glide angle, which is of great significance to improve the glide range. The smaller the skeleton volume ratio, means the smaller the ratio of skeleton volume to glider volume, the more energy and other equipment it can carry when the glider’s gravity and buoyancy are balanced. Moreover, the strength of the skeleton shall be as large as possible to ensure that the glider has higher safety.

The multidisciplinary design of the shape and skeleton for BWBUGs is shown in Fig. 18. The geometry models of the shape and skeleton are obtained by using the parametric methods. The CFD technology is adopted to evaluate the performance of shape by pretreatment and CFD, the FEA technology is used to evaluate the strength of the skeleton. In addition, the shape design is the basis of the skeleton, which means that shape affects the strength of the skeleton in the same thickness of the skeleton. The shape of BWBUGs can be divided into plane shape and wing shape. Both the fuselage profile and the transition mode of the fuselage and wing reflect the wing-body fusion arrangement. The plane shape of the fuselage is created in this part using the Bezier curve.

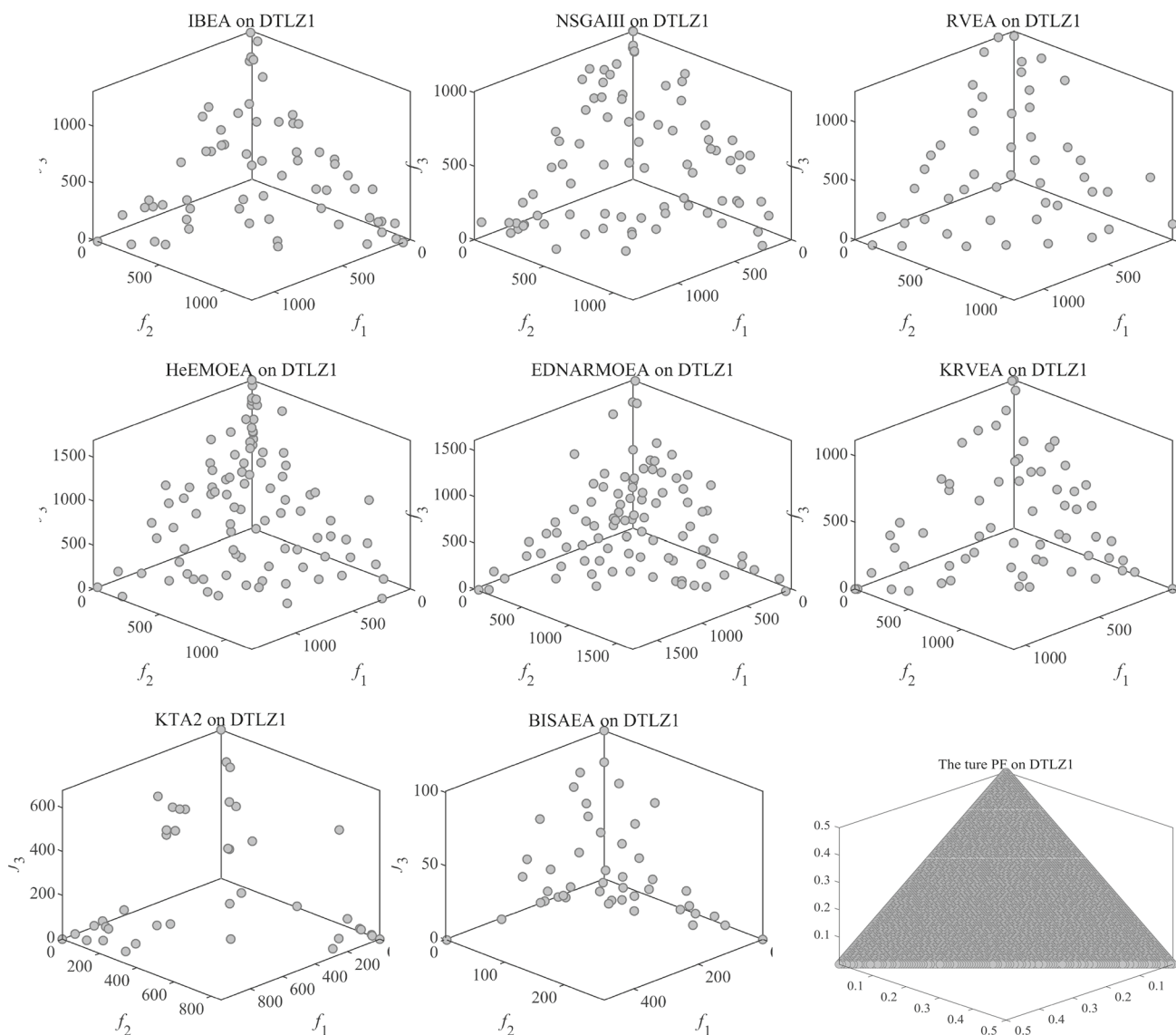


Fig. 11 The PF of DTLZ1 associated with the median IGD value on 30D

For the design of the shape, the layouts of 4 sections are adopted as shown in Fig. 18, where L and D_1 are set as 1500 mm and 1000 mm respectively, and the other nine parameters are design variables. Among them, the locations of section 1 and section 4 are fixed. And the locations of the other sections are related to the variable Z_2, Z_3 and L . Selecting NACA0022, NACA0019, NACA0016 and NACA0012 as the reference airfoil for sections 1, 2, 3, and 4, respectively. The Class-Shape function Transformation (CST) method is used for wing parameterization [55], and the airfoil is controlled by five design variables. There are 29 design variables for the shape design, and the range is shown in Eq. (11)–(16).

$$Section1 \begin{cases} A_{Low} = [-0.0860, -0.0741, -0.0835, -0.0574, -0.0861] \\ A_{Up} = [0.0860, 0.0741, 0.0835, 0.0574, 0.0861] \end{cases} \quad (11)$$

$$Section2 \begin{cases} A_{Low} = [-0.0717, -0.0618, -0.0696, -0.0478, -0.0718] \\ A_{Up} = [0.0717, 0.0618, 0.0696, 0.0478, 0.0718] \end{cases} \quad (12)$$

$$Section3 \begin{cases} A_{Low} = [-0.0573, -0.0494, -0.0557, -0.0383, -0.0574] \\ A_{Up} = [0.0573, 0.0494, 0.0557, 0.0383, 0.0574] \end{cases} \quad (13)$$

$$Section4 \begin{cases} A_{Low} = [-0.0287, -0.0247, -0.0278, -0.0191, -0.0287] \\ A_{Up} = [0.0287, 0.0247, 0.0278, 0.0191, 0.0287] \end{cases} \quad (14)$$

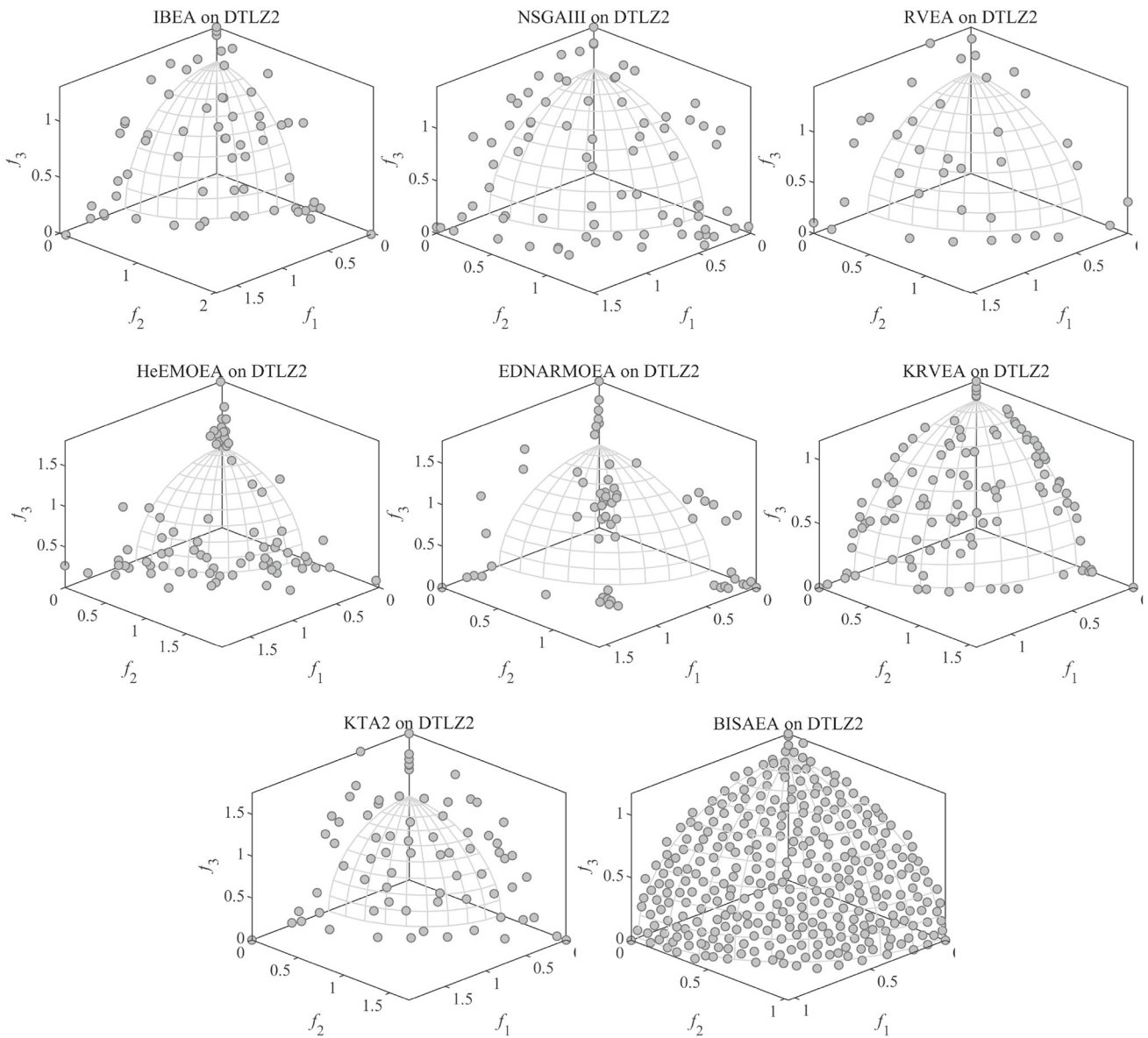


Fig. 12 The PF of DTLZ2 associated with the median IGD value on 30D

Fig. 13 The IGD values iteration process of five algorithms in DTLZ2

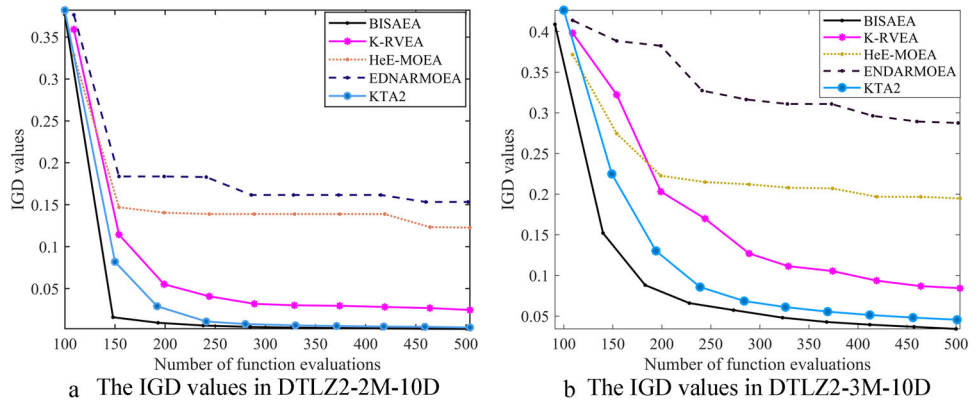


Table 6 Statistic results of the four SAEAs and BISAEA on ZDT1-4 and ZDT6

Problem	<i>M</i>	<i>D</i>	HeEMOEA	EDNARMOEA	KRVEA	KTA2	BISAEA
ZDT1	2	10	1.0281e+0 (1.01e−1) −	1.8917e−1 (5.90e−2) −	1.1312e−2 (1.06e−3) −	1.1820e−2 (4.66e−2) −	1.5550e−3 (2.15e−4)
	2	15	1.3351e+0 (1.22e−1) −	2.4880e−1 (4.62e−2) −	1.5800e−2 (1.65e−3) −	3.2916e−2 (1.08e−1) −	1.8269e−3 (1.09e−4)
	2	30	1.8132e+0 (1.02e−1) −	5.2488e−1 (5.61e−2) −	5.0245e−2 (6.41e−3) −	5.3424e−2 (9.74e−2) −	3.5880e−3 (4.18e−4)
	2	50	2.2679e+0 (1.26e−1) −	2.2570e+0 (9.37e−2) −	2.2121e+0 (1.29e−1) −	8.3931e−2 (9.06e−2) −	8.8269e−3 (2.01e−3)
ZDT2	2	10	1.6423e+0 (1.34e−1) −	5.6629e−1 (1.18e−1) −	1.8198e−2 (1.22e−2) −	2.6712e−3 (1.97e−4) −	1.1999e−3 (4.37e−5)
	2	15	2.1389e+0 (1.35e−1) −	6.7696e−1 (1.00e−1) −	2.1282e−2 (3.75e−3) −	2.9352e−3 (2.04e−4) −	1.3795e−3 (9.62e−5)
	2	30	2.9447e+0 (1.05e−1) −	1.1408e+0 (5.72e−2) −	6.3094e−2 (2.40e−2) −	3.5756e−2 (1.70e−1) −	2.2287e−3 (3.16e−4)
	2	50	3.6615e+0 (1.43e−1) −	3.6217e+0 (1.36e−1) −	3.5843e+0 (1.79e−1) −	3.0886e−1 (3.87e−1) −	4.4114e−3 (7.37e−4)
ZDT3	2	10	7.9179e−1 (8.50e−2) −	1.8172e−1 (7.71e−2) =	1.9172e−2 (3.23e−3) +	1.2023e−1 (1.40e−1) +	1.7556e−1 (1.28e−1)
	2	15	1.0220e+0 (9.51e−2) −	2.5109e−1 (9.07e−2) +	2.4015e−2 (4.52e−3) +	3.3851e−2 (3.32e−2) +	3.6723e−1 (1.45e−1)
	2	30	1.3921e+0 (1.32e−1) −	4.3252e−1 (4.23e−2) +	8.5094e−2 (2.91e−2) +	5.7714e−2 (5.57e−2) +	8.6531e−1 (2.56e−1)
	2	50	1.8685e+0 (9.83e−2)	1.7883e+0 (1.07e−1)	1.8129e+0 (1.24e−1)	8.8091e−2 (4.88e−2) +	1.3010e+0(2.55e−1)
ZDT4	2	10	4.0543e+1 (7.78e+0) −	5.1264e+1 (8.02e+0) −	2.2411e+1 (7.26e+0) =	2.5554e+1(8.41e+0) =	2.3471e+1 (1.27e+1)
	2	15	9.8411e+1 (9.92e+0) −	9.9193e+1 (1.13e+1) −	5.7013e+1 (1.79e+1) =	5.8623e+1(3.56e+1) =	5.3084e+1 (3.47e+1)
	2	30	3.1577e+2 (1.77e+1) −	2.5631e+2 (1.65e+1) −	2.7171e+2 (4.45e+1) −	2.2315e+2(5.69e+1) =	2.1170e+2 (6.90e+1)
	2	50	6.6238e+2 (2.82e+1) −	6.6665e+2 (2.35e+1) −	6.7149e+2 (2.18e+1) −	4.7555e+2(7.64e+1) =	4.6919e+2 (4.71e 1)
ZDT6	2	10	6.0048e+0 (1.70e−1) −	3.3810e+0 (3.49e−1) −	6.6671e−1 (1.41e−1) −	2.5080e−1 (1.31e−1) =	2.5396e−1 (6.76e−2)
	2	15	6.4747e+0 (9.21e−2) −	4.5413e+0 (3.04e−1) −	8.8849e−1 (1.50e−1) −	4.4565e−1 (1.69e−1) =	4.5887e−1 (1.30e−1)
	2	30	7.0091e+0 (5.86e−2)	5.8895e+0 (1.84e−1)	4.6339e+0 (5.69e−1)	1.4004e+0 (2.33e−1)−	1.0318e+0 (2.38e−1)
	2	50	7.3938e+0 (8.08e−2) −	7.3965e+0 (4.74e−2) −	7.3917e+0 (7.99e−2) −	2.5689e+0 (3.33e−1)−	1.9276e+0 (3.81e−1)
±/=		0/20/0	2/17/1	3/16/1	4/10/6		

Bold values indicate better results than other compared algorithms

$$\begin{aligned}
 &Plane \begin{cases} P_{low} = [350, 150, 350, 1000, 150, 300, 500, 300, 150]mm \\ P_{up} = [450, 300, 400, 1200, 250, 450, 600, 450, 250]mm \end{cases} \\
 &(D_2, D_3, D_4, D_5, Z_1, Z_2, Z_3, Z_4, Z_5)
 \end{aligned} \tag{15}$$

The skeleton is generated based on the shape. Subsequently, the skeleton is defined by 7 geometric parameters in Fig. 18. To be specific, *t* denotes the thickness of the skeleton, and others related to the geometrical parameters of the shape are given by using Eqs. (17)–(19).

$$\begin{aligned}
 A_{shape} &= [section1, section2, section3, section4, Plane] \\
 rh_1 &= Z_1 \\
 rh_2 &= Z_3 - Z_1
 \end{aligned} \tag{16}$$

Table 7 Statistic results of the three EAs and BISAEA on ZDT1-4 and ZDT6

Problem	<i>M</i>	<i>D</i>	IBEA	NSGAIII	RVEA	BISAEA
ZDT1	2	10	8.7593e−1 (1.41e−1) –	1.0637e+0 (2.16e−1) –	9.7894e−1 (1.48e−1) –	1.5550e−3 (2.15e−4)
	2	15	1.1380e+0 (1.46e−1) –	1.3951e+0 (1.99e−1) –	1.3390e+0 (1.96e−1) –	1.8269e−3 (1.09e−4)
	2	30	1.6866e+0 (1.63e−1) –	1.8020e+0 (1.93e−1) –	1.8489e+0 (1.60e−1) –	3.5880e−3 (4.18e−4)
	2	50	1.9660e+0 (1.21e−1) –	2.1150e+0 (1.12e−1) –	2.0787e+0 (1.21e−1) –	8.8269e−3 (2.01e−3)
ZDT2	2	10	1.7053e+0 (2.96e−1) –	1.8406e+0 (2.38e−1) –	1.6528e+0 (2.67e−1) –	1.1999e−3 (4.37e−5)
	2	15	2.2121e+0 (2.74e−1) –	2.3846e+0 (1.73e−1) –	2.2743e+0 (2.70e−1) –	1.3795e−3 (9.62e−5)
	2	30	2.9188e+0 (2.02e−1) –	3.0900e+0 (1.99e−1) –	2.9336e+0 (1.52e−1) –	2.2287e−3 (3.16e−4)
	2	50	3.3341e+0 (1.89e−1) –	3.5145e+0 (1.60e−1) –	3.4235e+0 (1.63e−1) –	4.4114e−3 (7.37e−4)
ZDT3	2	10	7.2864e−1 (1.47e−1) –	9.4271e−1 (1.82e−1) –	8.0754e−1 (1.39e−1) –	1.7556e−1 (1.28e−1)
	2	15	9.5246e−1 (1.18e−1) –	1.0936e+0 (1.47e−1) –	1.1496e+0 (1.57e−1) –	3.6723e−1 (1.45e−1)
	2	30	1.2698e+0 (1.66e−1) –	1.4742e+0 (1.49e−1) –	1.4374e+0 (1.84e−1) –	8.6531e−1 (2.56e−1)
	2	50	1.5288e+0 (1.43e−1) –	1.6983e+0 (1.74e−1) –	1.6540e+0 (1.86e−1) –	1.3010e+0 (2.55e−1)
ZDT4	2	10	5.6893e+1 (9.70e+0) –	6.3144e+1 (9.08e+0) –	4.3003e+1 (8.81e+0) –	2.3471e+1 (1.27e+1)
	2	15	1.1943e+2 (1.63e+1) –	1.2502e+2 (1.42e+1) –	9.8758e+1 (1.13e+1) –	5.3084e+1 (3.47e+1)
	2	30	3.2021e+2 (1.72e+1) –	3.3539e+2 (1.93e+1) –	2.8763e+2 (2.06e+1) –	2.1170e+2 (6.90e+1)
	2	50	6.1078e+2 (2.89e+1) –	6.3009e+2 (2.57e+1) –	5.6607e+2 (3.24e+1) –	4.6919e+2 (4.71e+1)
ZDT6	2	10	5.8567e+0 (3.55e−1) –	6.2337e+0 (2.07e−1) –	6.1199e+0 (2.27e−1) –	2.5396e−1 (6.76e−2)
	2	15	6.3480e+0 (1.59e−1) –	6.5646e+0 (2.09e−1) –	6.5756e+0 (2.27e−1) –	4.5887e−1 (1.30e−1)
	2	30	6.9154e+0 (1.11e−1) –	7.1103e+0 (9.63e−2) –	7.0477e+0 (1.07e−1) –	1.0318e+0 (2.38e−1)
	2	50	7.1834e+0 (8.61e−2) –	7.3058e+0 (6.37e−2) –	7.3032e+0 (8.56e−2) –	1.9276e+0 (3.81e−1)
±/=		0/20/0	0/20/0	0/20/0		

Bold values indicate better results than other compared algorithms

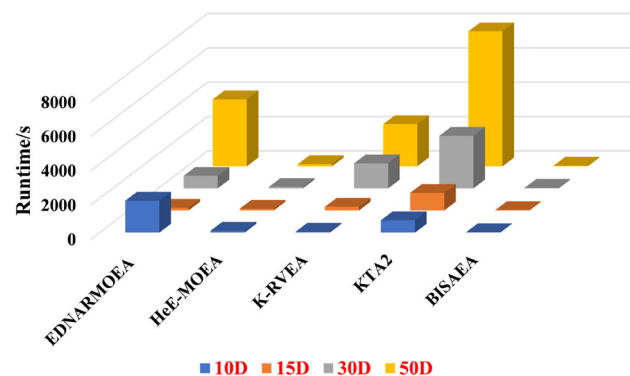


Fig. 14 The runtime of five algorithms of DTLZ2-2M with different dimension

$$rh_3 = \frac{Z_4 - Z_3}{4} \tag{17}$$

$$rv_1 = \frac{D_4}{2}$$

$$rv_2 = \frac{D_4}{2} + \frac{D_2}{4} \tag{18}$$

$$rv_3 = \frac{D_2}{4}$$

$$t \in [5, 10]mm \tag{19}$$

BWBUGs pump water in/out of the reservoir when moving, and the pressure on its outer surface is very small. This article mainly considers the stress in the process of dipping into the water, the distributed force is set as 1000 N in all skeleton.

To obtain better performance of the multidisciplinary design for shape and skeleton, L/D , V_s/V and σ_s are set as the objective. Hence, a 30-dimensional multi-objective optimization problem is summarized below:

$$\min F(\mathbf{x}) = \{-L/D, V_s/V, \sigma_s\}^T$$

$$s.t. \ x_i \in [A_{shape}, A_{skeleton}], i = 1, 2, \dots, 30 \tag{20}$$

$$A_{shape} \in [A_{shapeLow}, A_{shapeUp}], A_{skeleton} \in t$$

Optimization results

BISAEA, K-RVEA, and HeE-MOEA are all operated with 500 NFEs, with a population size of 100, for the multidisciplinary design of the shape and skeleton of BWBUGs. The same 200 initial sample points from BISAEA, K-RVEA, and HeE-MOEA are shared for fair comparison and are displayed in Fig. 19. Following the run, the obtained solutions are shown in Fig. 20.

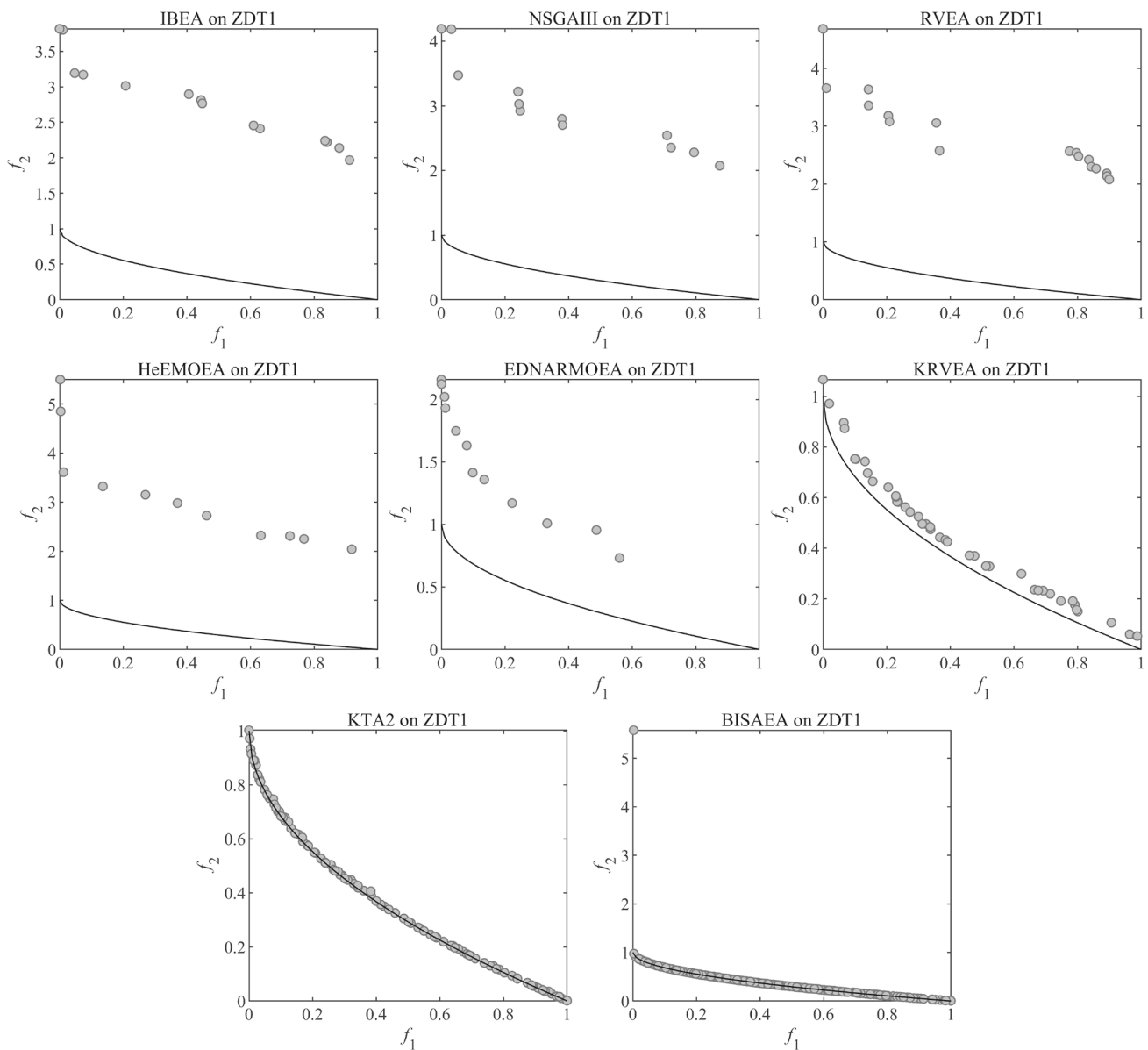


Fig. 15 The PF of ZDT1 in the run associated with the median IGD value on 30D

To further compare the performance mathematically, hypervolume is adopted because the actual PF is unknown. The hypervolume values of K-RVEA, HeE-MOEA and BISAEA are 0.6098, 0.5876 and 0.5986, respectively. It is obvious that K-RVEA and BISAEA are at the same level. In addition, the influence of calculation time on optimization efficiency is considered. The computation time of three algorithms under the different number of samples in the optimization process is shown in Fig. 21. The computation times only include the modeling and evolution of the optimization process, leaving out the evaluation of true functions. It is obvious that the time of K-RVEA utilized in different numbers of samples differs greatly, and the time grows as

the number of samples rises. HeE-MOEA also has a rule that is somewhat similar to that of K-RVEA. In comparison to K-RVEA and HeE-MOEA, BISAEA shows negligible time variation over a range of sample sizes. The superiority of BISAEA in the multidisciplinary design optimization of BWBUGs can be demonstrated by combining hypervolume values and computation time.

Besides, the three non-domination solutions of common PF from the three algorithms are selected in Fig. 20, and the pressure nephogram of shape, stress nephogram of the skeleton, and multidisciplinary system model are shown in Figs. 22 and 23.

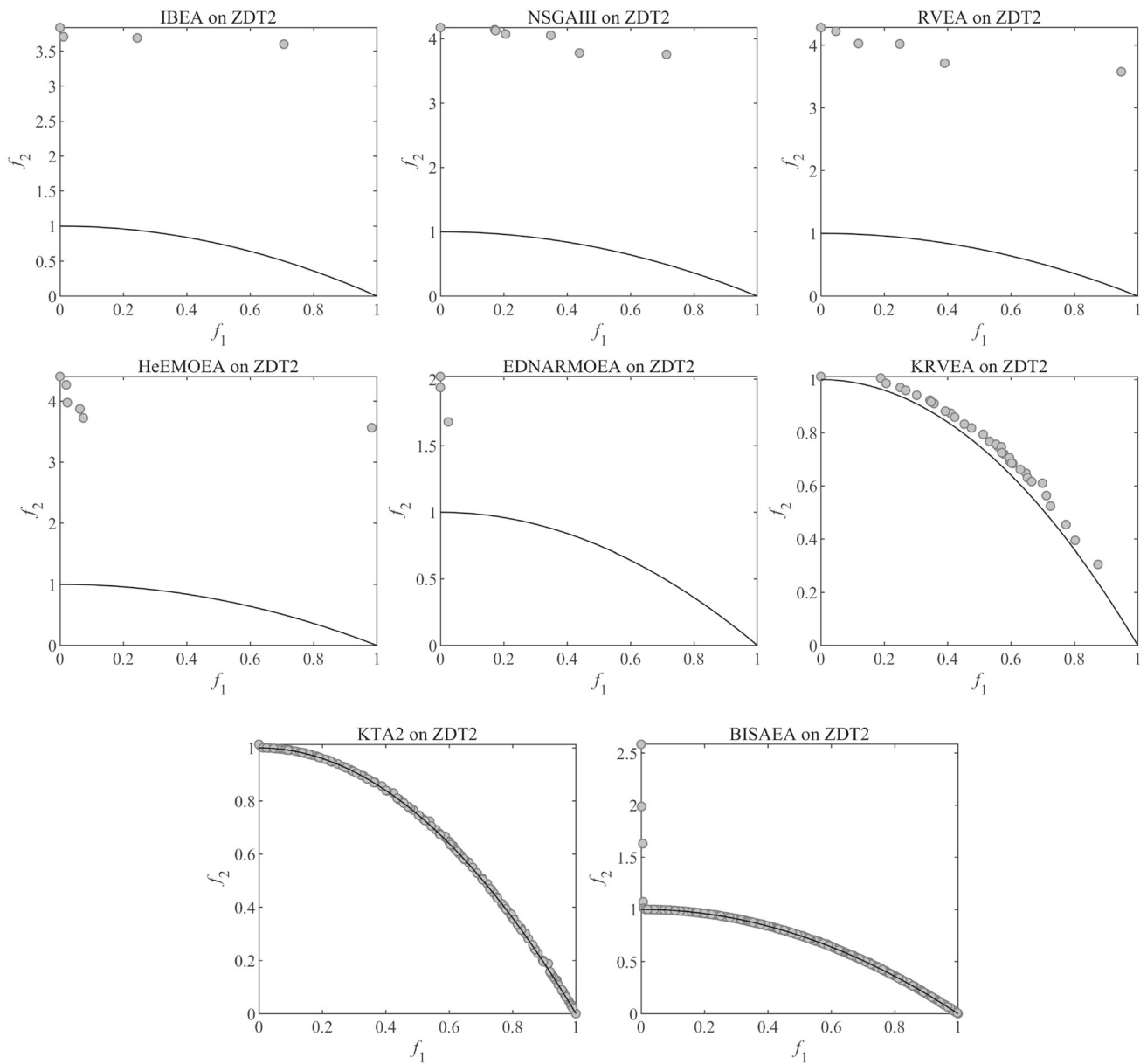
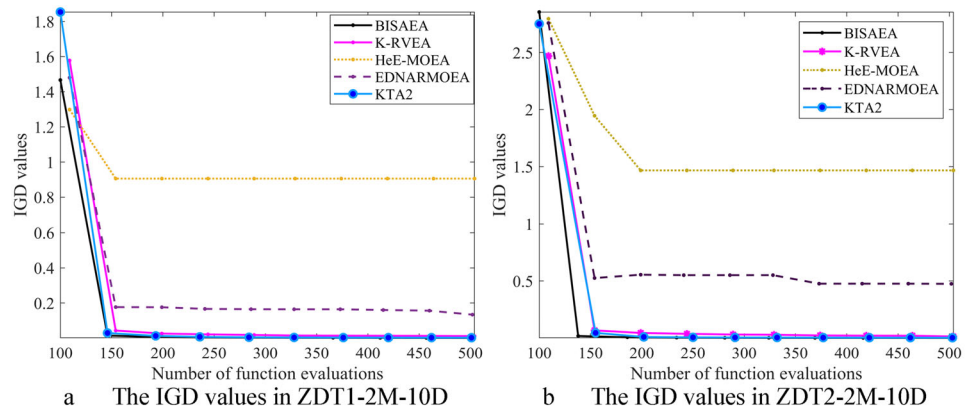


Fig. 16 The PF of ZDT2 in the run associated with the median IGD value on 30D

Fig. 17 The IGD values iteration process of five algorithms in ZDT1 and 2



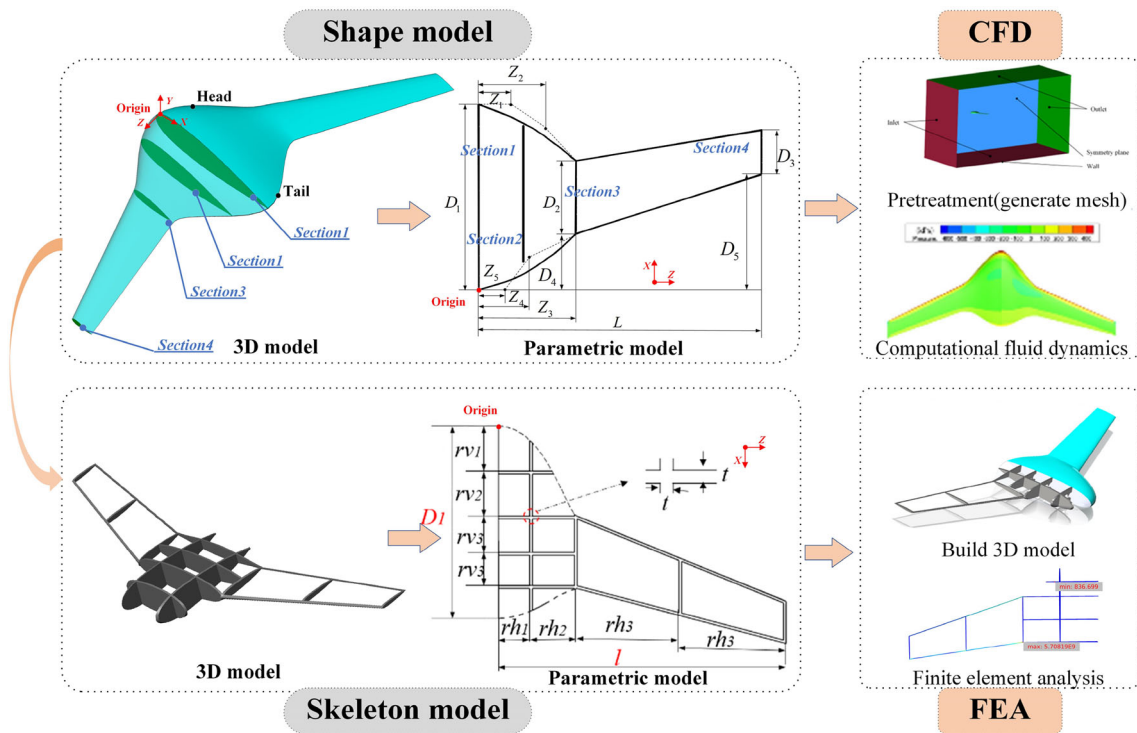


Fig. 18 The multidisciplinary design of BWBUGs

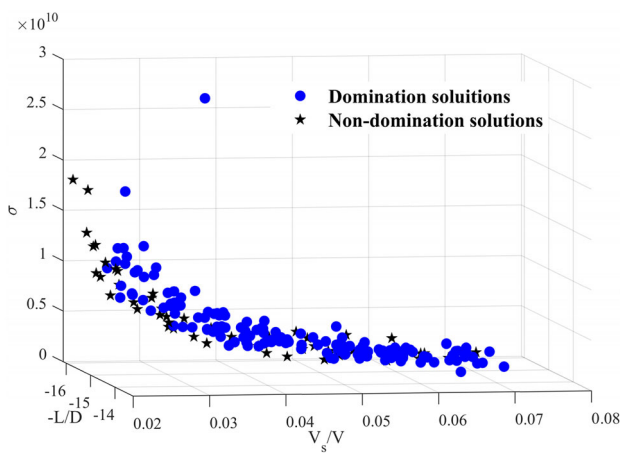


Fig. 19 The initial sample points

Conclusions

In this paper, a bi-indicator-based surrogate-assisted multi-objective evolutionary algorithm named BISAEA is proposed based on two strategies. Pareto-based bi-indicator strategy employs the convergence and diversity indicator as the bi-objective to enhance the convergence and diversity, and the bi-indicator is calculated by the approximate objective values of the RBF model. To further improve the performance

of the selected samples, a one-by-one selection strategy is adopted to filter samples. Besides, BISAEA is compared with the four state-of-art SAEAs and three EAs in DTLZ and ZDT benchmark problems on the dimensions of 10, 15, 30 and 50, BISAEA shows high efficiency and a good balance between convergence and diversity. Finally, BISAEA is applied to the multidisciplinary optimization of blend-wing-body underwater gliders with 30 decision variables and three objectives, and the results show its effectiveness on the engineering problem. The overall experiment results and application show that BISAEA has significant competitiveness for some state-of-art algorithms.

For future research, BISAEA may get a further study on the many-objectives optimization and this algorithm will be applied to more engineering problems. Moreover, we will attempt to use the surrogate models to approximate the indicator values directly and adopt some latest technologies in artificial intelligence and machine learning to improve the accuracy of the approximation of surrogate models.

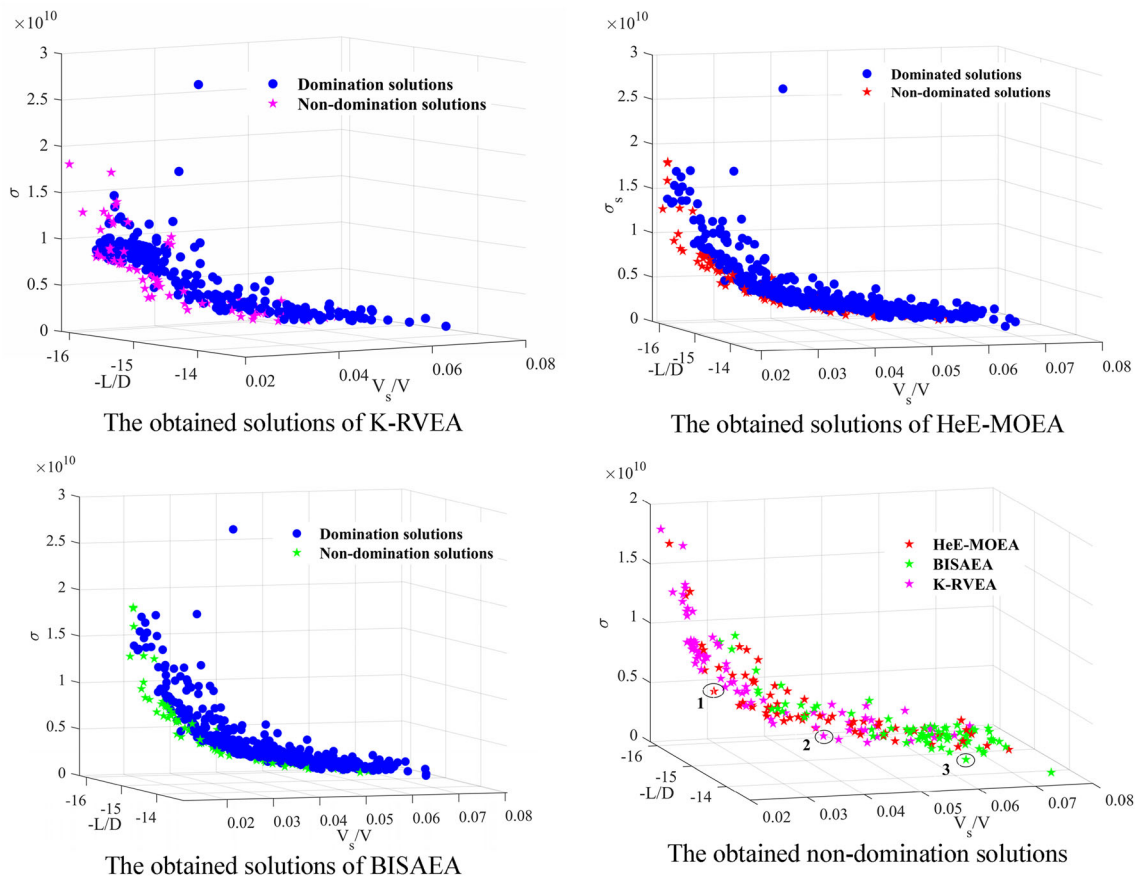


Fig. 20 The obtained solutions of K-RVEA, HeE-MOEA and BISAEA

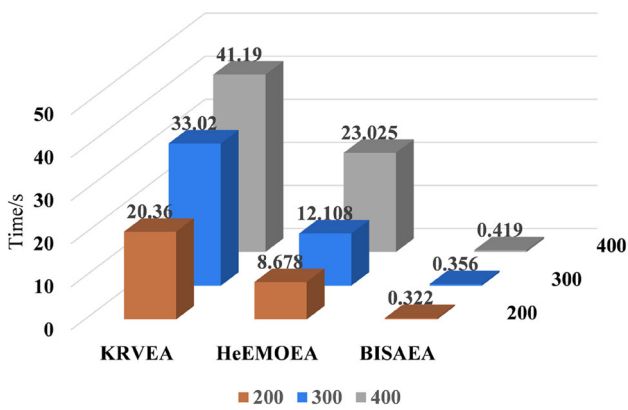


Fig. 21 The computation time of three algorithms under 200, 300 and 400 samples

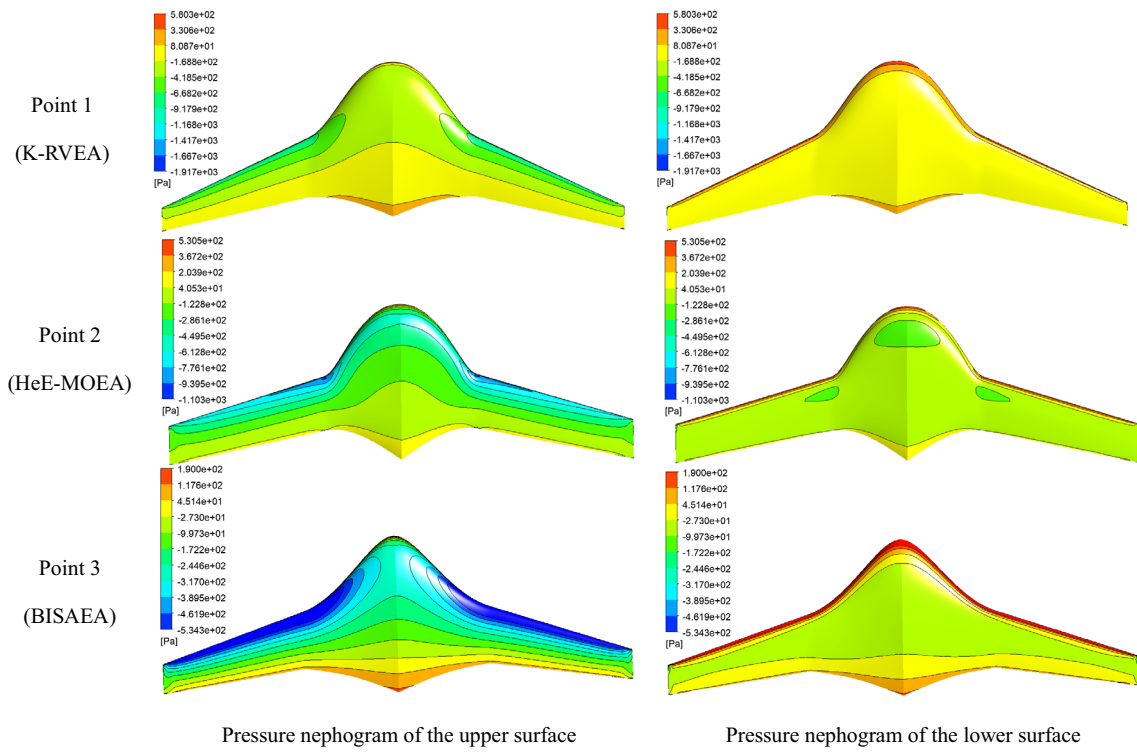


Fig. 22 Three typical pressure nephogram of shape

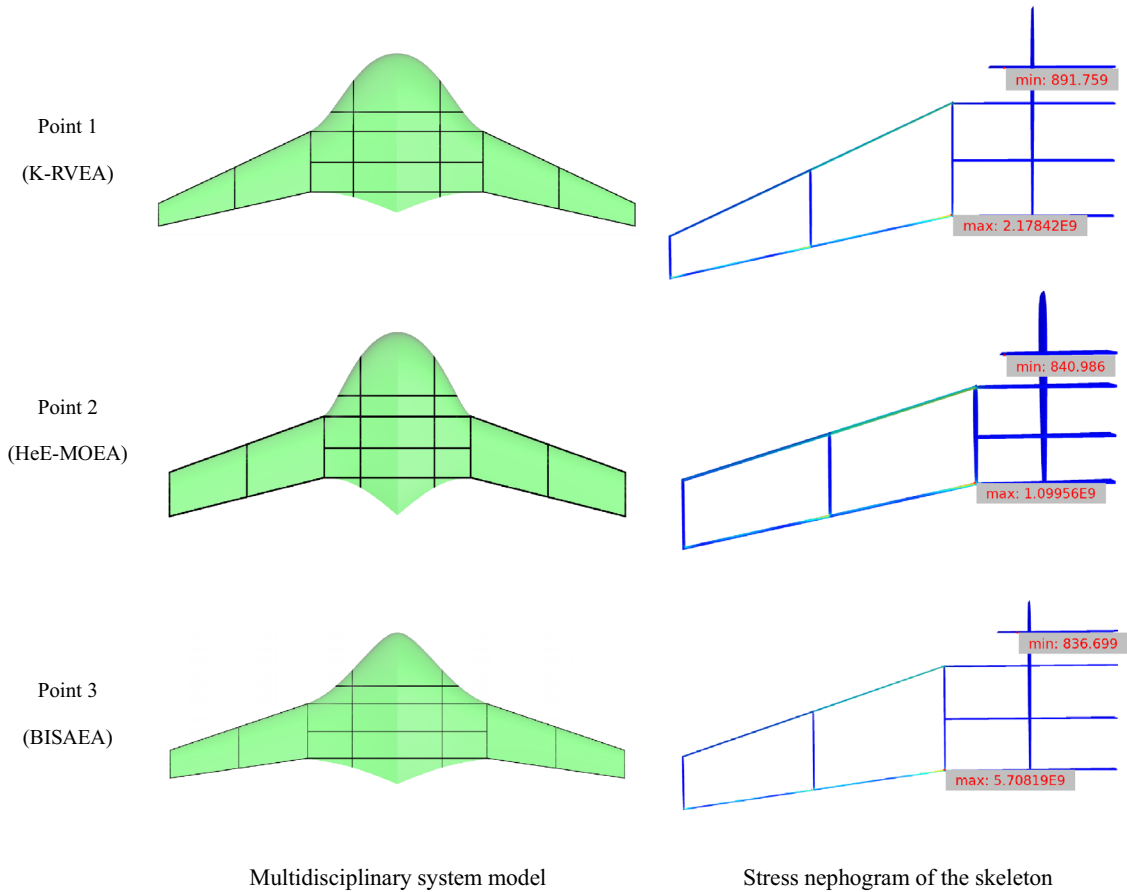


Fig. 23 Three typical schemes and their nephogram

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s40747-023-00969-w>.

Acknowledgements This project is supported by the National Natural Science Foundation of China (Grant No. 52175251, 51875466). Besides, the research work is also supported by the Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University.

Data availability Enquiries about data availability should be directed to the authors.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethics approval This article is ethical, and this research has been agreed upon.

Consent for publication The picture materials quoted in this article have no copyright requirements, and the source has been indicated.

Informed consent Informed consent was obtained from all individual participants included in the study.

Human and animal rights We use no animals in this research.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Jin Y, Sendhoff B (2009) A systems approach to evolutionary multiobjective structural optimization and beyond. *IEEE Comput Intell Mag* 4(3):62–76. <https://doi.org/10.1109/MCI.2009.933094>
- Miettinen K (2012) *Nonlinear multiobjective optimization*, vol 12. Springer Science & Business Media, Berlin
- Bader J, Zitzler E (2011) HypE: an algorithm for fast hypervolume-based many-objective optimization. *Evol Comput* 19(1):45–76. https://doi.org/10.1162/EVCO_a_00009
- Deb K, Jain H (2013) An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE Trans Evol Comput* 18(4):577–601. <https://doi.org/10.1109/TEVC.2013.2281535>
- Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput* 11(6):712–731. <https://doi.org/10.1109/TEVC.2007.892759>
- Cheng R, Jin Y, Olhofer M, Sendhoff B (2016) A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Trans Evol Comput* 20(5):773–791. <https://doi.org/10.1109/TEVC.2016.2519378>
- Jin Y (2005) A comprehensive survey of fitness approximation in evolutionary computation. *Soft Comput* 9(1):3–12
- Chen W, Wang P, Dong H (2022) Surrogate-based bilevel shape optimization for blended-wing-body underwater gliders. *Eng Optim*. <https://doi.org/10.1080/0305215X.2022.2057480>
- Li J, Wang P, Dong H, Shen J (2022) A two-stage surrogate-assisted evolutionary algorithm (TS-SAEA) for expensive multi-/many-objective optimization. *Swarm Evol Comput*. <https://doi.org/10.1016/j.swevo.2022.101107>
- Li J, Wang P, Dong H, et al. (2022). A classification surrogate-assisted multi-objective evolutionary algorithm for expensive optimization. *Knowledge-Based Systems*, 242: 108416. <https://doi.org/10.1016/j.knsys.2022.108416>
- Gunst RF, Myers RH, Montgomery DC (1996) Response surface methodology: process and product optimization using designed experiments. *Technometrics* 38(3):285. <https://doi.org/10.1080/00401706.1996.10484509>
- Martin JD, Simpson TW (2004) Use of kriging models to approximate deterministic computer models. *AIAA J* 43(4):853–863. <https://doi.org/10.2514/1.8650>
- Geman S, Bienenstock E, Doursat R (1992) Neural networks and the bias/variance dilemma. *Neural Comput* 4(1):1–58. <https://doi.org/10.1162/neco.1992.4.1.1>
- Dyn N, Levin D, Ripka S (1986) Numerical procedures for surface fitting of scattered data by radial functions. *SIAM J Sci Stat Comput* 7(2):639–659. <https://doi.org/10.1137/0907043>
- Wang X, Wang GG, Song B, Wang P, Wang Y (2019) A novel evolutionary sampling assisted optimization method for high dimensional expensive problems. *IEEE Trans Evol Comput* 23(9):815–827. <https://doi.org/10.1109/TEVC.2019.2890818>
- Dong H, Dong Z (2020) Surrogate-assisted Grey wolf optimization for high-dimensional, computationally expensive black-box problems. *Swarm Evol Comput* 57:100713. <https://doi.org/10.1016/j.swevo.2020.100713>
- Dong H, Wang P, Yu X, Song B (2020) Surrogate-assisted teaching-learning-based optimization for high-dimensional and computationally expensive problems. *Appl Soft Comput* 99(2):106934. <https://doi.org/10.1016/j.asoc.2020.106934>
- Chugh T, Jin Y, Miettinen K, Hakanen J, Sindhya K (2018) A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization. *IEEE Trans Evol Comput* 22(1):129–142. <https://doi.org/10.1109/TEVC.2016.2622301>
- Zhan D, Cheng Y, Liu J (2017) Expected improvement matrix-based infill criteria for expensive multiobjective optimization. *IEEE Trans Evol Comput* 21(6):956–975. <https://doi.org/10.1109/TEVC.2017.2697503>
- Liu Q, Cheng R, Jin Y, Heiderich M, Rodemann T (2022) Reference vector-assisted adaptive model management for surrogate-assisted many-objective optimization. *IEEE Trans Syst Man Cybernet Syst*. <https://doi.org/10.1109/TSMC.2022.3163129>
- Song Z, Wang H, He C, Jin Y (2021) A Kriging-assisted two-archive evolutionary algorithm for expensive many-objective optimization. *IEEE Trans Evol Comput* 25(6):1013–1027. <https://doi.org/10.1109/TEVC.2021.3073648>
- Zhang Q, Liu W, Tsang E, Virgians B (2010) Expensive multi-objective optimization by MOEA/D with gaussian process model. *IEEE Trans Evol Comput* 14(3):456–474. <https://doi.org/10.1109/TEVC.2009.2033671>
- Guo D, Wang X, Gao K, Jin Y, Ding J, Chai T (2021) Evolutionary optimization of high-dimensional multiobjective and many-objective expensive problems assisted by a dropout neural network. *IEEE Trans Syst Man Cybernet Syst PP*(99):1–14. <https://doi.org/10.1109/TSMC.2020.3044418>

24. Lin J, He C, Cheng R (2022) Adaptive dropout for high-dimensional expensive multiobjective optimization. *Complex Intell Syst* 8(1):271–285. <https://doi.org/10.1007/s40747-021-00362-5>
25. Guo D, Jin Y, Ding J, Chai T (2018) Heterogeneous ensemble-based infill criterion for evolutionary multiobjective optimization of expensive problems. *IEEE Trans Cybern* 49(3):1012–1025. <https://doi.org/10.1109/TCYB.2018.2794503>
26. Pan L, He C, Tian Y, Wang H, Zhang X, Jin Y (2018) A classification-based surrogate-assisted evolutionary algorithm for expensive many-objective optimization. *IEEE Trans Evol Comput* 23(1):74–88. <https://doi.org/10.1109/TEVC.2018.2802784>
27. Zhang J, Zhou A, Zhang G (2015) A classification and Pareto domination based multiobjective evolutionary algorithm. In: 2015 IEEE congress on evolutionary computation (CEC). IEEE, pp 2883–2890. <https://doi.org/10.1109/CEC.2015.7257247>
28. Yevseyeva I, Guerreiro AP, Emmerich M, Fonseca CM (2014) A portfolio optimization approach to selection in multiobjective evolutionary algorithms. In: International conference on parallel problem solving from nature. Springer, Cham, pp 672–681. https://doi.org/10.1007/978-3-319-10762-2_66
29. Tian Y, Cheng R, Zhang X et al (2017) An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility. *IEEE Trans Evol Comput* 22(4):609–622. <https://doi.org/10.1109/TEVC.2017.2749619>
30. Gómez RH, Coello CAC (2013) MOMBI: a new metaheuristic for many-objective optimization based on the R2 indicator. In: 2013 IEEE congress on evolutionary computation. IEEE, pp 2488–2495. <https://doi.org/10.1109/CEC.2013.6557868>
31. Hernández Gómez R, Coello Coello C A. (2015). Improved metaheuristic based on the R2 indicator for many-objective optimization//Proceedings of the 2015 annual conference on genetic and evolutionary computation, 679–686. <https://doi.org/10.1145/2739480.2754776>
32. Zitzler E, Künzli S (2004) Indicator-based selection in multiobjective search. In: International conference on parallel problem solving from nature, Springer, Berlin, Heidelberg, pp 832–842. https://doi.org/10.1007/978-3-540-30217-9_84
33. Zitzler E, Thiele L, Laumanns M, Fonseca CM, Da Fonseca VG (2003) Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans Evol Comput* 7(2):117–132. <https://doi.org/10.1109/TEVC.2003.810758>
34. Wang H, Jiao L, Yao X (2014) Two_Arch2: an improved two-archive algorithm for many-objective optimization. *IEEE Trans Evol Comput* 19(4):524–541. <https://doi.org/10.1109/TEVC.2014.2350987>
35. Li B, Tang K, Li J, Yao X (2016) Stochastic ranking algorithm for many-objective optimization based on multiple indicators. *IEEE Trans Evol Comput* 20(6):924–938. <https://doi.org/10.1109/TEVC.2016.2549267>
36. Li M, Yang S, Liu X (2015) Pareto or non-Pareto: bi-criterion evolution in multiobjective optimization. *IEEE Trans Evol Comput* 20(5):645–665. <https://doi.org/10.1109/TEVC.2015.2504730>
37. Jin R, Simpson TW (2001) Comparative studies of metamodeling techniques under multiple modeling criteria. *Struct Multidiscip Optim* 23(1):1–13. <https://doi.org/10.2514/6.2000-4801>
38. Cai X, Gao L, Li X (2019) Efficient generalized surrogate-assisted evolutionary algorithm for high-dimensional expensive problems. *IEEE Trans Evol Comput* 24(2):365–379. <https://doi.org/10.1109/TEVC.2019.2919762>
39. McKay MD, Beckman RJ, Conover WJ (2000) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 42(1):55–61. <https://doi.org/10.1080/00401706.2000.10485979>
40. Helton JC, Davis FJ (2003) Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliab Eng Syst Saf* 81(1):23–69. [https://doi.org/10.1016/S0951-8320\(03\)00058-9](https://doi.org/10.1016/S0951-8320(03)00058-9)
41. He L, Ishibuchi H, Trivedi A, Wang H, Nan Y, Srinivasan D (2021) A survey of normalization methods in multiobjective evolutionary algorithms. *IEEE Trans Evol Comput* 25(6):1028–1048. <https://doi.org/10.1109/TEVC.2021.3076514>
42. Singh HK, Bhattacharjee KS, Ray T (2018) Distance-based subset selection for benchmarking in evolutionary multi/many-objective optimization. *IEEE Trans Evol Comput* 23(5):904–912. <https://doi.org/10.1109/TEVC.2018.2883094>
43. Wang H, Jin Y (2018) A random forest-assisted evolutionary algorithm for data-driven constrained multiobjective combinatorial optimization of trauma systems. *IEEE Trans Cybern* 50(2):536–549. <https://doi.org/10.1109/TCYB.2018.2869674>
44. Tian Y, Cheng R, Zhang X, Jin Y (2017) PlatEMO: a MATLAB platform for evolutionary multi-objective optimization. *IEEE Comput Intell Mag*. <https://doi.org/10.1109/MCI.2017.2742868>
45. Deb K (2005) Scalable test problems for evolutionary multi-objective optimization. *Evolutionary multiobjective optimization: theoretical advances and applications*. https://doi.org/10.1007/1-84628-137-7_6
46. Deb K (1999) Multi-objective genetic algorithms: problem difficulties and construction of test problems. *Evol Comput* 7(3):205–230. <https://doi.org/10.1162/evco.1999.7.3.205>
47. Czyżżak P, Jaskiewicz A (1998) Pareto simulated annealing—a metaheuristic technique for multiple-objective combinatorial optimization. *J Multi-criteria Decis Anal* 7(1):34–47. [https://doi.org/10.1002/\(SICI\)1099-1360\(199801\)7:1%3C34::AID-MCDA161%3E3.0.CO;2-6](https://doi.org/10.1002/(SICI)1099-1360(199801)7:1%3C34::AID-MCDA161%3E3.0.CO;2-6)
48. Stuntz A, Kelly JS, Smith RN (2016) Enabling persistent autonomy for underwater gliders with ocean model predictions and terrain-based navigation. *Front Robot AI* 3:23. <https://doi.org/10.3389/frobt.2016.00023>
49. Bachmayer R, Leonard NE, Graver J, Fiorelli E, Bhatta P, Paley D (2004) Underwater gliders: Recent developments and future applications. In: Underwater Technology. UT '04. 2004 international symposium on 2004. <https://doi.org/10.1109/UT.2004.1405540>
50. D'Spain GL, Zimmerman R, Jenkins SA, Luby JC, Brodsky P (2007) Underwater acoustic measurements with a flying wing glider. *J Acoust Soc Am* 121(5):3107–3107. <https://doi.org/10.1121/1.4782033>
51. Li J, Wang P, Dong H, Wu X, Chen X, Chen C (2020) Shape optimization of blended-wing-body underwater gliders based on free-form deformation. *Ships Offshore Struct* 15(3):227–235. <https://doi.org/10.1080/17445302.2019.1611989>
52. Wang W, Dong H, Wang P, Li J, Shen J (2022) A model-based multi-disciplinary conceptual design for blended-wing-body underwater gliders. *Ships Offshore Struct*. <https://doi.org/10.1080/17445302.2022.2126126>
53. Sun C, Song B, Peng W (2015) Parametric geometric model and shape optimization of an underwater glider with blended-wing-body. *Int J Naval Archit Ocean Eng* 7(6):995–1006. <https://doi.org/10.1515/ijnaoe-2015-0069>
54. Wang ZY, Jian-Cheng YU, Zhang AQ, Wang YX, Zhao WT (2017) Parametric geometric model and hydrodynamic shape optimization of a flying-wing structure underwater glider. *China Ocean Eng* 31(006):709–715. <https://doi.org/10.1007/s13344-017-0081-7>
55. Hicks RM, Henne PA (1978) Wing design by numerical optimization. *J Aircr* 15(7):407–412. <https://doi.org/10.2514/3.58379>