Information Sciences 519 (2020) 317–331

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

An adaptive Bayesian approach to surrogate-assisted evolutionary multi-objective optimization

Xilu Wang[a], Yaochu Jin[a,*], Sebastian Schmitt[b], Markus Olhofer[b]

[a] Department of Computer Science, University of Surrey, Guildford, GU2 7XH, UK
[b] Honda Research Institute Europe GmbH, Carl-Legien-Strasse 30, D-63073 Offenbach/Main, Germany

ARTICLE INFO

Article history:
Received 3 September 2019
Revised 21 January 2020
Accepted 28 January 2020
Available online 30 January 2020

Keywords:
Expensive multi-objective optimization
Surrogate-assisted evolutionary algorithm
Bayesian optimization
Acquisition function
Reference vectors

ABSTRACT

Surrogate models have been widely used for solving computationally expensive multi-objective optimization problems (MOPs). The efficient global optimization (EGO) algorithm, a Bayesian approach to surrogate-assisted optimization, has become very popular in surrogate-assisted evolutionary optimization. In this paper, we propose an adaptive Bayesian approach to surrogate-assisted evolutionary algorithm to solve expensive MOPs. The main idea is to tune the hyperparameter in the acquisition function according to the search dynamics to determine which candidate solutions are to be evaluated using the expensive real objective functions. In addition, the sampling selection criterion switches between an angle based distance and an angle-penalized distance over the course of optimization to achieve a better balance between exploration and exploitation. The performance of the proposed algorithm is examined on a set of benchmark problems and an airfoil design optimization problem using a maximum of 300 real fitness evaluations. Our experimental results show that the proposed algorithm is competitive compared to four popular multi-objective evolutionary algorithms.

1. Introduction

Multi-objective optimization problems (MOPs) are commonly seen in many economic, scientific, and engineering applications. Various types of algorithms have been proposed for solving MOPs. For example, the scalarization technique is one popular method that converts an MOP into a single-objective optimization problem, which can be achieved by the global criterion method, the weighted min-max method, the ϵ-constraint method, normal boundary intersection or reference point methods [1]. In addition, game theory has been applied to many real-world MOPs, such as wireless and communication networks [2,3] and electric systems [4], due to the similarity between MOPs and the game [5]. For example, power optimization for a microgrid is defined as a bi-objective optimization [4]. Two players are assumed to correspond to two objectives, and consequently a modified game theory is adopted to find an optimal solution. Tsiropoulou et al. adopted the non-cooperative game theory to efficiently allocate transmission power and data rate to users in the uplink of a cellular wireless network [6]. Another popular approach is based on evolutionary algorithms (EAs), which have been applied successfully to many real-world complex optimization problems [7]. Since they are population-based methods that are able to obtain multiple

[*] Corresponding author.
E-mail addresses: xw00616@surrey.ac.uk (X. Wang), yaochu.jin@surrey.ac.uk (Y. Jin), sebastian.schmitt@honda-ri.de (S. Schmitt), markus.olhofer@honda-ri.de (M. Olhofer).

https://doi.org/10.1016/j.ins.2020.01.048
0020-0255/© 2020 Published by Elsevier Inc.

Pareto optimal solutions in a single run, EAs are particularly well suited for MOPs. Over the past decades, a large number of multi-objective evolutionary algorithms (MOEAs) have been proposed, such as nondominated sorting genetic algorithm II (NSGA-II) [8], multi-objective evolutionary algorithm based on decomposition (MOEA/D) [9], reference vector guided evolutionary algorithm (RVEA) [10], strength Pareto evolutionary algorithm 2 (SPEA2) [11], and an estimation of distribution algorithm (IM-MOEA) [12], just to name a few.

However, a major criticism against MOEAs is that most of them require a large number of function evaluations to find Pareto optimal solutions. This weakness becomes more obvious when EAs are employed for solving time-consuming multi-objective problems where the evaluation of the objectives involves expensive physical experiments or computationally intensive numerical simulations. This class of problems is common in real-world applications [13], where, e.g., computationally intensive computational fluid dynamics (CFD) simulations must be conducted to evaluate the performance in structural design optimization [14] and in engineering design optimization [15].

To address the challenges in using MOEAs for computationally expensive optimization problems, surrogate-assisted evolutionary algorithms (SAEAs) are widely adopted [16]. SAEAs can be roughly classified into two categories. In the first category, the expensive objective function is approximated by a single or an ensemble of surrogates. More specifically, computationally efficient surrogate models are constructed using historical data and then used to approximate the fitness values of the candidate solutions instead of computing the real fitness. For example, Marjavaara [17] replaced the three-dimensional models of the diffuser geometries with a surrogate model and achieved better performance in the diffuser shape design problem. In the second category, the surrogate works as a classifier to filter the newly generated candidate solutions based on the predicted labels. Consequently, several classification-based SAEAs have been developed, e.g., the classification and Pareto domination based MOEA [18] and the classification-based SAEA (CSEA)[19].

The commonly used surrogate models include the radial basis function (RBF), support vector machines (SVM), radial basis function networks (RBFN), polynomial regression, and Gaussian processes (GPs), also called Kriging model. Among them, GP is one of the most popular surrogate models and has attracted increasing attention. The main reason is that GP can not only predict the objective value of a candidate solution, but also provide a confidence level (a degree of uncertainty) of its predictions [20]. Based on the predicted objective value and its uncertainty, an acquisition function (AF) can be utilized as the model management strategy to determine which candidate solutions should be evaluated by the real objective function. Surrogate-assisted optimization using a GP in combination with an acquisition function (also known as infill criterion) is called Bayesian optimization [21] or efficient global optimization (EGO) [22].

The Bayesian optimization approach, which combines GPs with an acquisition function, have been widely adopted in SAEAs for handling various expensive single-objective optimization problems. For example, in [23] the GP model was used as an inexpensive fitness function to accelerate convergence, and different aggregations of the predicted function value and the predicted standard deviation of the GP were employed as acquisition functions to balance exploration with exploitation. Hoyle et al. adopted a GP model to predict objective function values when optimizing the design of engine air intakes, new data points are then chosen based on the expected improvement function, after finding a promising location, a local exploration is performed to speed up the convergence [24]. In [25], a particle swarm optimization algorithm switches between a local ensemble surrogate model and a global one, interleavingly searching for the best and most uncertain solutions. The algorithm has been shown to perform well on both a set of single-objective test problems and an airfoil design problem. Tian et al. [26] considered the uncertainty and the approximated fitness provided by the GP as two separate objectives instead of combining them into a scalar acquisition function, which was demonstrated to be competitive for solving high-dimensional single-objective optimization problems.

Nevertheless, it is non-trivial to build effective surrogate models and select new data samples based on the working mechanisms of the acquisition function for accelerating convergence and promoting diversity in multi-objective optimization. In [27], an algorithm termed ParEGO was proposed, where the augmented Tchebycheff function is adopted to aggregate multiple objectives into a scalar fitness function so that an existing acquisition function for single-objective optimization can be directly applied for surrogate management. Ponweiser et al. [28] proposed a new Bayesian approach to multi-objective optimization (termed SMS-EGO), in which a hypervolume based acquisition function rather than a weighted aggregation of the objective functions is utilized to convert an MOP into a single-objective optimization problem. However, due to the computational complexity in calculating the hypervolume, SMS-EGO may become very time-consuming as the number of objectives increases. A GP-assisted MOEA/D was reported in [29], where a GP model is built for each subproblem so that an acquisition function can be applied to each subproblem. In [30], Chugh et al. introduced a Kriging-based RVEA (called K-RVEA), which combines the angle penalized distance and the uncertainty for selecting new samples to update the GP. Whenever diversity is needed in K-RVEA, the uncertainty is chosen as the selection criterion by ranking the mean value of uncertainty on different objectives and selecting the one having the maximum averaging uncertainty.

Most surrogate-assisted MOEAs using the Bayesian approach have focused on converting a multi-objective problem into a single-objective one by means of aggregation or decomposition so that acquisition functions can be used for surrogate management. One exception is presented in Guo et al. [31], which suggests to use an ensemble to replace the GP models to reduce the computational complexity especially for high-dimensional problems. To the best of our knowledge, no work has been reported to tune the hyperparameters in the acquisition function to achieve the best trade-off between exploitation and exploration during the search process, which is of pivotal importance in optimization.

The rest of this paper is organized as follows. Section 2 presents a brief introduction to the background knowledge related to Bayesian optimization. In Section 3, a pilot study to investigate the efficiency of each proposed strategy is undergone and the proposed algorithm is described in detail. Numerical simulations are conducted in Section 4, where the results are presented and discussed. Finally, conclusions of this paper are drawn and future work is suggested in Section 5.

2. Background

In this section, Gaussian processes and the widely used acquisition functions are introduced at first, followed by a review of the angle-penalized distance, the selection criterion used in RVEA.

2.1. Gaussian processes

GP is a powerful surrogate model, which can provide a prediction for a new data point, together with a degree of uncertainty. For example, a 1D GP model illustrated in Fig. 1 is trained with three training data, the GP model can then predict the mean fitness and the uncertainty on any new data point. That is, for a given x_1, we can know the model's prediction $\mu(x_1)$, uncertainty $\sigma(x_1)$, and the confidence intervals $\mu(x_1) \pm \sigma(x_1)$, respectively. The uncertainty information denotes the confidence level of the prediction, which is very useful in model management in surrogate-assisted optimization. The idea behind GPs [32] is that there is a multivariate Gaussian distribution on R^n for any finite set of N n-dimensional inputs $\mathbf{X} = \{\mathbf{x}^1, \mathbf{x}^2, \dots \mathbf{x}^N\}^T$. Given a set of inputs $\mathbf{x}$ and their associated function values $\mathbf{y} = \{y^1, y^2, \dots y^N\}^T$ as the training data, a GP model can be expressed by:

$$y(\mathbf{x}) = \mu + \varepsilon(\mathbf{x}) \tag{1}$$

where μ is the mean of the stochastic process, and $\varepsilon(\mathbf{x})$ is drawn from a stochastic process with zero mean but non-zero standard deviation σ

$$\varepsilon(\mathbf{x}) \sim N(0, \sigma^2) \tag{2}$$

The correlation between the error terms of any two data points, $\mathbf{x}^i$ and $\mathbf{x}^j$, heavily depends on the distance between them. In general, the squared exponential function with additional hyperparameters, rather than an Euclidean distance function, is employed to calculate the correlation:

$$d(\mathbf{x}^i, \mathbf{x}^j) = \sum_{k=1}^{m} \theta_k |x_k^i - x_k^j|^{p_k} \tag{3}$$

$$\mathrm{Corr}(\mathbf{x}^i, \mathbf{x}^j) = \exp\left[-d(\mathbf{x}^i, \mathbf{x}^j)\right] \tag{4}$$

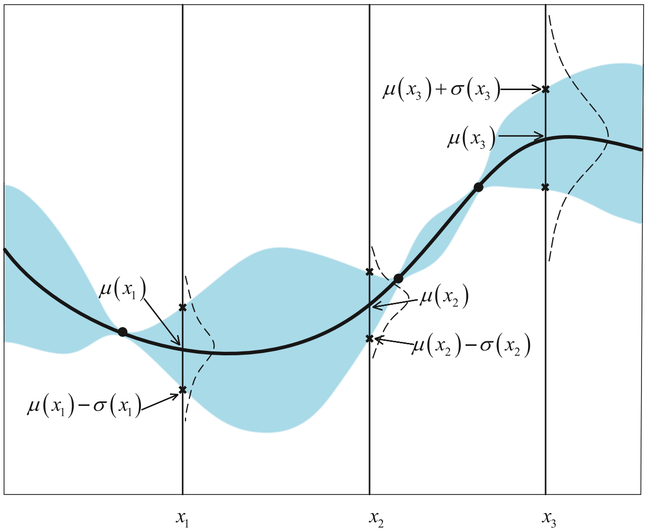


Fig. 1. Illustration of a 1D Gaussian process with three training data points and the predictions of the GP mean values and standard deviation values ($\mu(\cdot)$ and $\sigma(\cdot)$) at three new data points (x_1, x_2 and x_3). The solid line and the shaded area indicate the mean and confidence intervals estimated with the GP model.

where $p_k \in (0, 1)$ controls the smoothness of the function in terms of the kth dimension, and $\theta_k > 0$ denotes the importance of this dimension. When there are N training data, an $N \times N$ correlation matrix $\mathbf{C}$ will be obtained:

$$\mathbf{C} = \begin{pmatrix} \mathrm{Corr}(\mathbf{x}^1, \mathbf{x}^2) & \cdots & \mathrm{Corr}(\mathbf{x}^1, \mathbf{x}^N) \\ \vdots & \ddots & \vdots \\ \mathrm{Corr}(\mathbf{x}^N, \mathbf{x}^1) & \cdots & \mathrm{Corr}(\mathbf{x}^N, \mathbf{x}^N) \end{pmatrix} \tag{5}$$

As a result, $2N + 2$ hyperparameters will determine a GP model, which can be estimated by maximizing the following likelihood function:

$$\psi(\theta_1, ..., \theta_N, p_1, ..., p_N) = -\frac{1}{2}\left(N \ln \sigma^2 + \ln det(\mathbf{C})\right) \tag{6}$$

Therefore, the estimates $\hat{\mu}$ and $\hat{\sigma}^2$ for the true values μ and σ will be obtained

$$\hat{\mu} = \frac{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{y}}{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{1}} \tag{7}$$

$$\hat{\sigma}^2 = \frac{(\mathbf{y} - \mathbf{1}\hat{\mu})^T\mathbf{C}^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu})}{N} \tag{8}$$

where $\mathbf{1}$ denotes a $N \times 1$ column vector of ones. Based on the given parameters, the GP model can predict the mean value together with a variance for a new data point $\mathbf{x}^{new}$,

$$f(\mathbf{x}^{new}) = \hat{\mu} + \mathbf{r}^T\mathbf{C}^{-\mathbf{1}}(\mathbf{y} - \mathbf{1}\hat{\mu}) \tag{9}$$

$$\hat{\sigma}(\mathbf{x}^{new})^2 = \hat{\sigma}^2\left[1 - \mathbf{r}^T\mathbf{C}^{-1}\mathbf{r} + \frac{(1 - \mathbf{r}^T\mathbf{C}^{-1}\mathbf{r})^2}{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{1}}\right] \tag{10}$$

where $\mathbf{r} = (\mathrm{Corr}(\mathbf{x}^{new}, \mathbf{x}^1), ..., \mathrm{Corr}(\mathbf{x}^{new}, \mathbf{x}^N))^T$ presents a correlation vector between $\mathbf{x}^{new}$ and each element $\mathbf{x}^i$ in $\mathbf{X}$.

2.2. Acquisition function

Given a GP model trained by a set of observed data, selecting the appropriate decision vector (a data point in the decision space) is essential for the SAEA to work effectively. Acquisition functions (AFs), inspired from Bayesian decision theory, are designed as metrics for selecting the unobserved data to be evaluated by the real expensive objective functions. AFs are computationally very cheap, and by sequentially finding its optimum we are able to guide the search towards the optimum of the original objective function.

Several AFs have been proposed to select new data points utilizing the information obtained by surrogate models. Expected improvement (EI) [33] calculates the expected value of the improvements beyond the best observed point, and was extended to multi-objective optimization in [34]. Predictive entropy search (PES) proposed in [35] is an alternative for Entropy Search (ES), making use of the information theory.

The AF proposed in this work is inspired from the lower confidence bound (LCB) [36]. LCB is designed to balance the exploration and exploitation by combining the uncertainty with the predicted objective values [37]. Intuitively, for a minimization problem, selecting a point with the maximum amount of uncertainty can encourage more exploratory search; however, selecting most uncertain points only may lead to nearly random search. By contrast, we can perform exploitative search if we pick points where the predicted mean values are minimum, which, however, at a higher risk of getting trapped in a local optimum. Hence, LCB aims to strike a balance between exploration and exploitation:

$$LCB(\mathbf{x}) = \mu(\mathbf{x}) - \kappa\sigma(\mathbf{x}) \tag{11}$$

where κ presents a trade-off constant. LCB implicitly prefers points whose predicted mean value μ is small and the corresponding standard deviation σ is large.

2.3. Angle-penalized Distance Metric

The reference vector guided evolutionary algorithm (RVEA) [10] is a recently proposed MOEA for solving many-objective optimization problems. In RVEA, a set of reference vectors are predefined in the objective space as the preferred search directions that guide the search towards the Pareto front (PF). An angle-penalized distance (APD) is designed based on the reference vectors to determine which candidate solutions should be associated with their closest reference vectors. More precisely, APD combines the convergence and diversity criteria and dynamically adjusts the importance of the two criteria according to the number of objectives and the number of generations. APD is defined as follows:

$$d_{t,i,j} = (1 + P(\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}})) \cdot \|\mathbf{f}_{t,i} - \mathbf{Z}_t^{\min}\| \tag{12}$$

Here, $d_{t,i,j}$ presents the APD value of the ith solution in terms of the jth reference vector in the t-th generation, and $\mathbf{Z}_t^{\min}$ denotes the vector of the minimal objective values at the current generation; $\|\mathbf{f}_{t,i} - \mathbf{Z}_t^{\min}\|$ denotes the translated objective value, adopted as a convergence criterion; $\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}}$ represents the angle between the objective vector $\mathbf{f}_{t,i}$ and the reference vector $\mathbf{v}_{t,j}$; $P(\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}})$ represents a penalty function and serves as a diversity criterion, which is defined as follows:

$$P(\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}}) = M \cdot \left(\frac{t}{t_{\max}}\right)^{\beta} \cdot \frac{\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}}}{\gamma_{\mathbf{v}_{t,j}}} \tag{13}$$

with

$$\gamma_{\mathbf{v}_{t,j}} = min_{i\in 1,\ldots,N, i\neq j}\theta_{\mathbf{v}_{t,i},\mathbf{v}_{t,j}} \tag{14}$$

where M and N represent the number of objectives and the number of reference vectors, respectively; β is a predefined parameter to control the rate of the penalty function; $\gamma_{v_{t,j}}$ denotes the smallest angle between the reference vector $\mathbf{v}_{t,j}$ and its closest neighboring reference vector $\mathbf{v}_{t,i}$.

Typically, the original reference vectors are generated uniformly in the objective space using the simplex-lattice design method. Unfortunately, these vectors are not suited for problems where different objectives have very different ranges. Hence, a reference vector adaptation method was proposed in [10] in the following manner:

$$\mathbf{v}_{t+1,i} = \frac{\mathbf{v}_{0,i} \circ (\mathbf{Z}_t^{\max} - \mathbf{Z}_t^{\min})}{\|\mathbf{v}_{0,i} \circ (\mathbf{Z}_t^{\max} - \mathbf{Z}_t^{\min})\|} \tag{15}$$

where '$\circ$' denotes the Hadamard product that multiplies two vectors of the same size element-wise, $\mathbf{v}_{t+1,i}$ denotes the i-th adapted reference vector for the t-th generation, while $\mathbf{v}_{0,i}$ represents the i-th uniformly distributed reference vector, and $\mathbf{Z}_t^{\max}$ and $\mathbf{Z}_t^{\min}$ denote the vectors consisting the maximum and minimum objective values obtained so far.

3. Proposed algorithm

As mentioned above, Bayesian optimization has been extended to expensive MOPs over the past years. In this section, an adaptive Bayesian approach to surrogate-assisted multi-objective evolutionary optimization algorithm (AB-MOEA) is presented, and Fig. 2 gives the flowchart of the proposed AB-MOEA.

3.1. Algorithm framework

The framework of the proposed AB-MOEA are outlined in Algorithm 1. The algorithm begins with sampling the objective functions for usually around $11n-1$ points, where n is the dimensionality of the decision space, and then uses these samples to train the GP model. The RVEA algorithm is executed using only the surrogate models for 20 generations to obtain an optimized population. This optimized population is utilized to determine u new samples with the proposed adaptive

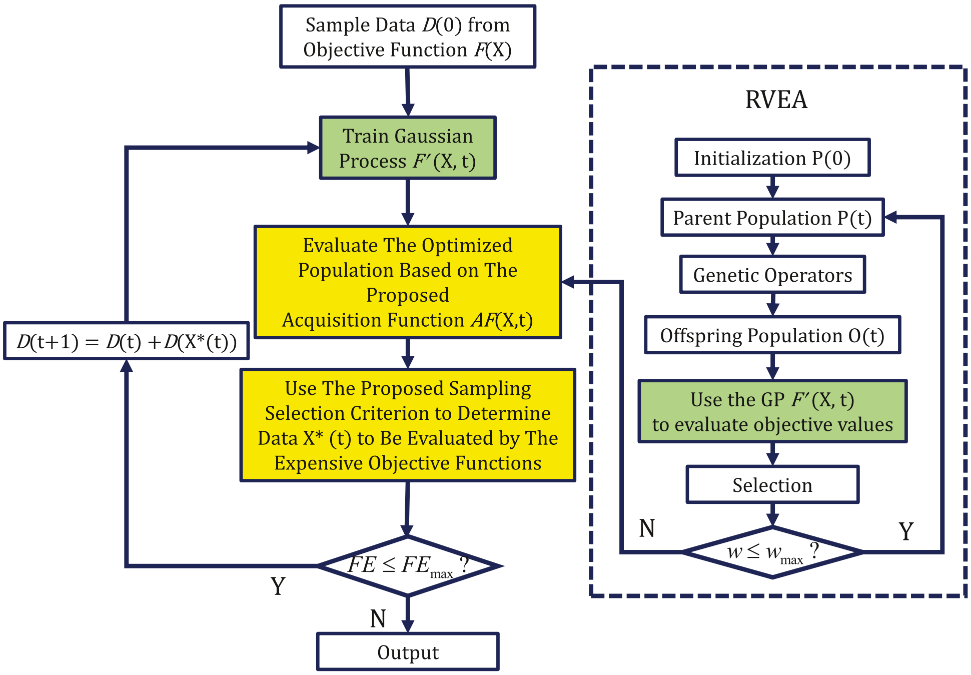

Sample Data D(0) from Objective Function F(X)
Train Gaussian Process F′(X, t)
Evaluate The Optimized Population Based on The Proposed Acquisition Function AF(X,t)
Use The Proposed Sampling Selection Criterion to Determine Data X* (t) to Be Evaluated by The Expensive Objective Functions
D(t+1) = D(t) +D(X*(t))
FE ≤ FE_max ?
Y
N
Output
RVEA
Initialization P(0)
Parent Population P(t)
Genetic Operators
Offspring Population O(t)
Use the GP F′(X, t) to evaluate objective values
Selection
w ≤ w_max ?
N
Y

Fig. 2. Flowchart of AB-MOEA.

Algorithm 1 Framework of AB-MOEA.

```
Input: $FE_{\max}$: the maximum number of real objective function evaluations; $u$: the number of points selected to be re-evaluated in each iteration; $w_{\max}$: the maximum number of generations before updating GPs;
Output: The final solution population
1: Initialization: Sample $11n-1$ points $\mathbf{x}^1$, $\mathbf{x}^2$,..., $\mathbf{x}^{11n-1}$ in the decision space using the Latin Hybercube Sampling method; evaluate the values $\mathbf{y}^1, \ldots, \mathbf{y}^{11n-1}$ of the objective functions on these $11n-1$ points; use $[\mathbf{X}, \mathbf{Y}]$ as training data;
2: train GP models with the training data;
3: while $FE \leqslant FE_{\max}$ do
4:     //Using the surrogate in RVEA//
5:     while $w \leqslant w_{\max}$ do
6:     Generate offspring by the simulated binary crossover (SBX) and the polynomial mutation;
7:     Combine parent and offspring populations and predict their fitness with the GP models;
8:     Select a new population for the next generation;
9:     Update the reference vectors;
10:    $w = w + 1$
11:    endwhile
12:    //Updating the surrogate//
13:    Call the proposed AF to evaluate the individuals in the optimized population;
14:    Call the proposed sampling selection strategy to determine $u$ points to be evaluated by the original objective functions and update $FE = FE + u$; (see Algorithm 2)
15:    Add individuals from step 8 to training data and limit the number of training data and go to step 2;
16: end while
17: return The final solution population $P$
```

acquisition function and the proposed sampling selection criterion. The new samples are evaluated with the real objective functions and the surrogate models are then retrained using the extended training set including the new data samples. The proposed framework is intended to efficiently optimize computationally expensive multi-objective optimization problems by introducing an adaptive acquisition function and a new sampling selection criterion. The adaptive acquisition function and the corresponding sampling selection criterion within this framework aim to achieve a better balance between diversity and convergence, which will be elaborated in the following.

3.2. Adaptive acquisition function

Many ideas have been suggested to adapt the Bayesian approach to MOPs, but no work on adaptation of the AFs has been reported. In this work, an adaptive acquisition function is proposed to dynamically adjust the weights of the uncertainty and the predicted mean fitness value, thereby achieving a better balance between exploration and exploitation at different search stages. The adaptive acquisition function is defined as follows:

$$AF = (1-\alpha) \cdot (\boldsymbol{\mu}_{\mathbf{x}}./\boldsymbol{\mu}_{\max}) + \alpha \cdot (\boldsymbol{\sigma}_{\mathbf{x}}./\boldsymbol{\sigma}_{\max}) \tag{16}$$

where

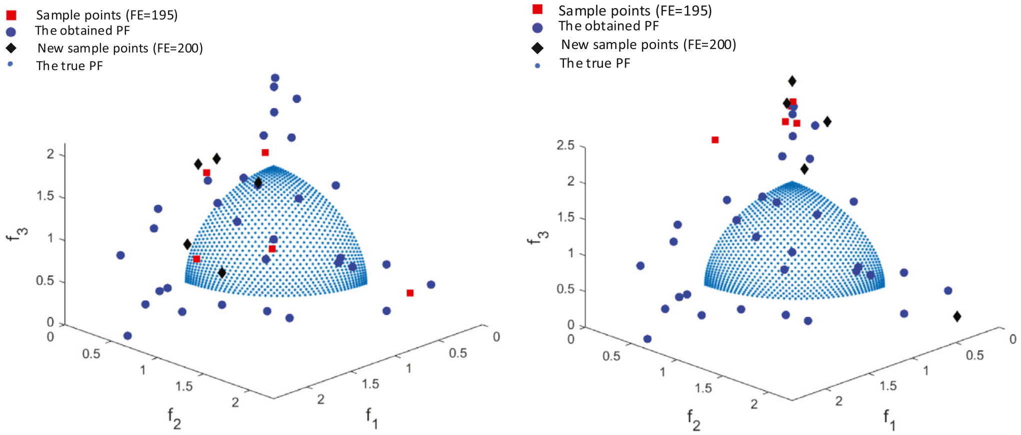
$$\alpha = -0.5 \cdot \cos\left(\frac{FE}{FE_{\max}} \cdot \pi\right) + 0.5 \tag{17}$$

where './' denotes element-by-element division for two vectors of the same size; $\boldsymbol{\mu}_{\mathbf{x}}$ and $\boldsymbol{\sigma}_{\mathbf{x}}$ denote the mean and variance of the predicted function values of each individual $\mathbf{x}$ in the optimized population obtained by RVEA, respectively; $\boldsymbol{\mu}_{max}$ and $\boldsymbol{\sigma}_{max}$ represent the maximum values of the mean and variance values provided by the GP models, respectively. In this way, both the predicted objective value and the uncertainty are normalized to [0,1]. α is an adaptation parameter defined by a cosine function. FE denotes the current number of real objective function evaluations, and FE_{max} represents the predefined maximum number of real objective function evaluations.

The main motivation behind the above adaptive acquisition function is to achieve relatively fast convergence in the early stage by assigning a large weight to $\mathbf{x}$, while more exploitative search is achieved in the later search stage, where we select new pints with APD, so that new data points in the neighborhood of the previously observed data are preferred to be sampled when searching the promising area. Herein we explain briefly why we select new data points with lower uncertainty as the iteration increases. It is assumed that a decision space around the training data sampled so far is the most promising area near the end of the run; therefore, the proposed AF allows to sample new points inside that domain for local search. We also test a variant of the proposed AF,

$$AF_{variant} = AF = (1-\alpha) \cdot (\boldsymbol{\mu}_{\mathbf{x}}./\boldsymbol{\mu}_{\max}) - \alpha \cdot (\boldsymbol{\sigma}_{\mathbf{x}}./\boldsymbol{\sigma}_{\max}). \tag{18}$$

The difference between the two AFs is whether we minimise the uncertainty or not near the end of the run and the experimental results confirm the effectiveness of the suggested AF. As shown in Fig. 3, the proposed AF and its variant sample



(a) Data points sampled by the proposed AF

(b) Data points sampled by the variant

Fig. 3. An example of new data points sampled by the proposed AF and its variant for DTLZ2 (when $FE = 195$ and $FE = 200$), respectively.

five new data points for twice (when $FE = 195$ and $FE = 200$) on DTLZ2 based on the same GP model. It is worth noting that the samples selected by the proposed AF are much closer to the PF, while points sampled by the variant are far away from the PF. These indicate that the proposed criterion in (15) results in more diverse and exploitative search in the later stage, which is of particular importance when the number of fitness evaluated is highly limited.

3.3. Adaptive sampling selection criterion

As analyzed in [38], the reference vectors predefined in RVEA divide the search space into several subspaces, with each reference vector specifying a direction along which the Pareto optimal solutions are preferred. A set of solutions associated to each reference vector is expected to be achieved to guarantee the diversity as well as the convergence. To this end, we introduce two different sampling selection strategies that work together with the proposed adaptive AF. Specifically, when $\alpha < 0.5$, the angle-based selection approach is employed to prioritize the population diversity:

$$Angle(\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}}) = M \cdot \frac{\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}}}{\gamma_{\mathbf{v}_{t,j}}} \tag{19}$$

The above sampling selection criterion does not consider the distance from the candidate solution to the reference vector, which usually serves as the convergence criterion in APD. As a result, this sampling selection strategy focuses on promoting the diversity of the population. When $\alpha >= 0.5$, APD is adopted to determine which points are to be sampled. The overall sampling selection criterion utilized in our algorithm is shown in Algorithm 2.

Algorithm 2 Adaptive Sampling Selection Criterion.

```
Input: $\alpha$: the adaptive parameter based on a cosine function; $u$: the number of points selected to be re-evaluated in each iteration; $v$: the reference vectors; $P$: the final population obtained from RVEA
Output: The selected re-evaluated individuals
if $\alpha < 0.5$ then
    // Using the angle-based sampling selection criterion//
    Calculate the $Angle(\theta_{\mathbf{f}_{t,i},\mathbf{v}_{t,j}})$ for each individual in the population and select $u$ new points to be re-evaluated;
elseIf $\alpha >= 0.5$
    //Using APD method//
    Calculate the $d_{t,i,j}$ for each individual in the population and select $u$ new points to be re-evaluated;
end if
```

4. Comparative studies

In this section, numerical experiments are conducted on nine three-objective benchmark problems taken from the DTLZ test suite and seven bi-objective benchmark problems from the UF test suite. To examine the efficiency of the proposed strategies, AB-MOEA is compared with its two variants, AS-MOEA and SM-MOEA. Here, AS-MOEA employs the proposed adaptive AF and the APD sampling selection criterion, while SM-MOEA employs the LCB and the proposed sampling selection criterion. Then we compare AB-MOEA with the original RVEA as well as three representative Bayesian approaches to SAEAs, namely MOEA/D-EGO [29], SMS-EGO [39], and K-RVEA [30]. Although all these compared algorithms except for RVEA employ different kinds of EAs and acquisition functions, respectively, they all adopt the Gaussian processes as the surrogates and use the prediction provided by GPs during the search process. As indicated in [40,41], the computational complexity of training GP is $O(n^3)$, where n is the number of training samples. Therefore, it is worthy of noting that the major computational costs of these algorithms are mainly spent on training the GPs, which depends on the number of training data. Finally, it should be emphasized that the computational complexity of an expensive real fitness evaluation, e.g., a 3D computational fluid dynamics simulation or car crash test often take hours or even days [14,19], which is much higher than that of training a surrogate. That is the reason why the number of real fitness evaluations has been used as the baseline for comparing SAEAs.

In the following section, we begin with briefly introducing the test problems and performance metrics adopted in this work. Afterwards, the details of the experimental settings concerning the four compared algorithms are described. Lastly, the experimental results including the pilot studies and the comparative results, together with the Wilcoxon rank sum test, are presented and discussed.

4.1. Test problems

In our experiments, the proposed algorithm is compared with four selected algorithms on seven benchmark functions (DTLZ1 to DTLZ7) with three objectives suggested in [42], two modified counterparts of DTLZ1 and DTLZ3, and the UF test suite from the CEC2009 MOEA competition with two objectives [43], respectively. Before starting our experiments, we want to have a discussion about the multi-model g function used in DTLZ1 and DTLZ3. The g function, given in the following, is suggested to control the ruggedness of DTLZ1 and DTLZ3 [42],

$$g = 100\left[5 + \sum_{i\in 1,\ldots,n} (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right], i = 1, \ldots, n. \tag{20}$$

where n denotes the number of decision variables. As indicated in [44], 20π within the cosine term triggers excessively ruggedness. Consequently, 2π is adopted to reduce the complexity to a reasonable level. The modified counterparts are denoted as DTLZ1a and DTLZ3a, respectively. As recommended in [42], the number of decision variables for the test instances is set to $n = M + K - 1$, where $K = 5$ is adopted for DTLZ1 and DTLZ1a, $K = 10$ is used for DTLZ2 to DTLZ6 as well as DTLA3a, and $K = 20$ is employed in DTLZ7. M represents the number of objectives. Here, we set $M = 3$.

4.2. Performance metrics

The inverted generational distance (IGD) [45], hypervolume (HV) [46], generational distance (GD) [47] and spread (Δ) [48] metrics are adopted to assess the performance of the algorithms. IGD and HV provide a combined information of the convergence and diversity of the obtained set of solutions, while GD and spread metrics work as the convergence measure and diversity measure, respectively. The PlatEMO toolbox [49] is used to calculate values of these performance metrics in our experiments. Let P^* be a set of uniformly distributed solutions sampled from the objective space along the true PF. Let P be an obtained approximation to the PF.

(1) *GD*: GD measures the average distance between the obtained PF and the true PF of the problem, formulated as follows:

$$GD(P^*, P) = \frac{\sum_{\upsilon\in P} d(\upsilon, P^*)}{|P|} \tag{21}$$

where $|P|$ is the cardinality of the set P and $d(\upsilon, P^*)$ is the minimal Euclidean distance between υ and all points in P^*. The smaller the GD value is, the better the convergence performance.

(2) *Spread* (Δ): Spread is used to evaluate the extent of the Pareto front covered by the obtained set of solutions, defined as

$$\Delta = \frac{\sum_{i=1}^{m} d(E_i, P) + \sum_{\upsilon\in P} |d(\upsilon, P) - \bar{d}|}{\sum_{i=1}^{m} d(E_i, P) + (|P| - m)\bar{d}} \tag{22}$$

where $d(\upsilon, P)$ is the minimum Euclidean distance between υ and all points in P, $(E_i, \ldots, E_m)$ are m extreme solutions in the true PF P^* and $\bar{d} = \frac{\sum_{\upsilon\in P} d(\upsilon, P))}{|P|}$ is the mean distance between the solutions of P. A smaller value of Δ indicates a better diversity of the obtained PF.

(3) *IGD*: The definition of IGD is similar to GD. IGD measures the inverted generational distance from P^* to P, defined as

$$IGD(P^*, P) = \frac{\sum_{\upsilon\in P^*} d(\upsilon, P)}{|P^*|} \tag{23}$$

where $d(\upsilon, P)$ is the minimum Euclidean distance between υ and all points in P. The smaller IGD value, the better the achieved solution set is.

(4) *HV*: HV calculates the volume of the objective space dominated by an approximation set P and dominates P^* sampled from the PF.

$$HV = \text{volume}\left(\cup_{\mathrm{i}=1}^{\mathrm{j}} \vartheta_{\mathrm{i}}\right) \tag{24}$$

where ϑ_i represents the hypervolume contribution of the ith solutions with respect to the reference points. All HV values presented in this work are normalized to [0,1]. Algorithms achieving a larger HV value are better.

4.3. Experimental settings

We run each algorithm on each benchmark problem for 20 independent times, and the Wilcoxon rank sum test is calculated to compare the mean of 20 running results obtained by AB-MOEA and by the compared algorithms at a significance level of 0.05. Symbol "(+)" indicates that the proposed algorithm outperforms the compared algorithm statistically significantly, while "(−)" means that the compared algorithm performs better than AB-MOEA, and "(≈)" means there is no significant difference between them.

AB-MOEA and its variants are implemented in MATLAB R2009a on an Intel Core i7 with 2.21 GHz CPU, and the compared algorithms are implemented in PlatEMO toolbox [49]. In all Bayesian SAEAs, the GP model is constructed using the DACE toolbox [50]. The general parameter settings in the experiments are given as follows: 1) The number of initial training points $N_{train} = 11n - 1$, where n is the number of decision variables. 2) The maximum number of real function evaluations $FE_{\max} = 300$. 3) The maximum number of generations before updating GPs $w_{\max} = 20$. The specific parameter settings for each compared algorithm are the same as recommended in their original papers.

4.4. Experimental results

4.4.1. Pilot studies

To validate the proposed strategies, pilot studies are performed on the above test instances for 20 runs independently, and the mean of the results achieved by the original RVEA is adopted for comparison using the Wilcoxon rank sum test. We calculate the values of the performance metrics for the non-dominated solution set in each run, respectively. The minimum, maximum and mean values of the performance metrics over 20 times are collected and presented in Tables 1 and 2, respectively, where the best result of each benchmark function is highlighted.

First, we assess the efficiency of the proposed sampling selection criterion in the proposed AB-MOEA in addressing expensive MOPs via comparing SM-MOEA with the original RVEA. According to the statistical results presented in Table 1, it can be seen that SM-MOEA significantly outperforms the original RVEA in terms of IGD values on DTLZ1a, DTLZ2, DTLZ3a, DTLZ6 as well as DTLZ7, while there is little difference between SM-MOEA and RVEA on DTLZ1, DTLZ3, DTLZ4 and DTLZ5.

Table 1
Statistical results of the IGD values obtained by SM-MOEA, AS-MOEA, AB-MOEA, and RVEA with the same number of real function evaluations.

Test problem	SM-MOEA				AS-MOEA				AB-MOEA				RVEA		
	min	mean		max	min	mean		max	min	mean		max	min	mean	max
DTLZ1	16.67	33.14	(≈)	56.21	14.26	28.39	(+)	**43.13**	16.58	**27.29**	(+)	52.88	**10.08**	33.68	51.56
DTLZ1a	2.04	5.36	(+)	9.58	0.96	1.89	(+)	3.48	**0.35**	**1.47**	(+)	**2.70**	5.21	23.79	48.80
DTLZ2	0.09	0.12	(+)	0.18	0.09	0.12	(+)	0.21	**0.08**	**0.09**	(+)	**0.17**	0.23	0.42	0.49
DTLZ3	185.7	347.6	(≈)	456.4	199.4	319.1	(+)	446.9	199.1	**310.8**	(+)	**420.7**	**154.2**	374.4	479.7
DTLZ3a	26.63	99.90	(+)	246.8	**9.08**	30.82	(+)	110.7	17.05	**30.65**	(+)	**58.97**	173.2	330.4	482.0
DTLZ4	0.36	0.53	(≈)	0.85	**0.15**	0.50	(≈)	0.78	0.23	**0.43**	(+)	**0.68**	0.30	0.56	0.71
DTLZ5	0.34	0.35	(≈)	0.37	0.34	0.34	(≈)	0.35	**0.06**	**0.09**	(+)	**0.19**	0.13	0.35	0.49
DTLZ6	3.91	4.70	(+)	5.54	3.89	4.66	(+)	**5.33**	**3.83**	**4.60**	(+)	5.56	6.22	7.89	8.50
DTLZ7	2.74	3.46	(+)	4.59	3.13	4.40	(+)	6.07	**0.58**	**0.81**	(+)	**1.40**	3.45	6.41	7.82

Table 2
Statistical results of the HV values obtained by SM-MOEA, AS-MOEA, AB-MOEA, and RVEA with the same number of real function evaluations.

Test problem	SM-MOEA				AS-MOEA				AB-MOEA				RVEA		
	min	mean		max	min	mean		max	min	mean		max	min	mean	max
DTLZ1	0.968	0.985	(+)	0.995	**0.979**	0.989	(+)	0.995	0.971	**0.993**	(+)	**0.996**	0.901	0.939	0.995
DTLZ1a	0.569	0.802	(+)	0.975	0.971	0.990	(+)	0.997	**0.973**	**0.992**	(+)	**0.998**	0.000	0.166	0.736
DTLZ2	0.305	0.443	(+)	0.489	0.270	0.432	(+)	0.485	**0.350**	**0.444**	(+)	**0.488**	0.021	0.111	1.000
DTLZ3	0.863	0.910	(+)	0.956	**0.876**	**0.922**	(+)	0.957	0.862	0.921	(+)	0.957	0.737	0.832	**0.985**
DTLZ3a	0.000	0.238	(+)	0.728	0.000	**0.857**	(+)	**0.982**	**0.403**	0.855	(+)	0.979	0.000	0.000	0.000
DTLZ4	0.000	0.015	(≈)	0.062	0.000	0.060	(≈)	**0.443**	**0.009**	**0.063**	(≈)	0.201	0.000	0.022	0.144
DTLZ5	0.212	0.255	(+)	0.282	**0.242**	**0.266**	(+)	**0.284**	0.108	0.118	(+)	0.147	0.000	0.006	0.068
DTLZ6	0.737	0.797	(+)	0.831	0.731	0.790	(+)	0.847	**0.751**	**0.801**	(+)	**0.834**	0.366	0.415	0.587
DTLZ7	0.000	0.000	(≈)	0.000	0.000	0.000	(≈)	0.000	**0.069**	**0.070**	(+)	**0.104**	0.000	0.000	0.000

Similarly, SM-MOEA outperforms RVEA on all tested instances except for DTLZ4 and DTLZ7 in terms of the HV metric. These results indicate SAEAs with the help of the suggested sampling selection criterion are able to improve the balance between the diversity and convergence.

Next, we compare AS-MOEA with RVEA to investigate the effectiveness of the adaptive acquisition function. On DTLZ1-3, DTLZ1a, DTLZ3a as well as DTLZ6, both the IGD and HV metrics suggest that AS-MOEA has achieved significant improvements over RVEA as the adaptive AF utilizing a nonlinear aggregation function to take both the uncertainty and the predicted mean fitness value into account. Since the maintenance of a good distribution of solutions is highly desirable on DTLZ4, the approximate PF obtained by AS-MOEA is no worse than that obtained by RVEA with the limited function evaluations. These results together confirm the effectiveness of the suggested adaptive AF.

From the above comparative results, we can conclude that the combination of the adaptive AF and the adaptive sampling selection criterion are helpful to achieve diverse and converged solutions using a limited number of real fitness evaluations. In addition, AB-MOEA significantly outperforms RVEA on all the benchmark functions in terms of the IGD metric. A similar conclusion can be drawn on the results in terms of the HV values, as presented in Table 2. All results presented in Tables 1 and 2 provide strong evidence confirming that both the adaptive AF and the new sampling selection criterion are able to improve the quality of solutions, and a combination of these two can further contribute to performance improvement.

4.4.2. Comparison with Surrogate-assisted MOEAs

In this subsection, the performance of the proposed AB-MOEA is compared with the state-of-the-art Bayesian SAEAs, including K-RVEA, MOEA/D-EGO, and SMS-EGO, in terms of IGD, HV, GD and spread metrics. Tables 3 and 4 present comparative results of IGD and HV, respectively. To evaluate the diversity and convergence separately, GD and spread metrics

Table 3
Statistical results of the IGD values obtained by K-RVEA, MOEA/D-EGO, SMS-EGO, and AB-MOEA with the same number of real function evaluations.

Test problem	K-RVEA				MOEA/D-EGO				SMS-EGO				AB-MOEA		
	min	mean		max	min	mean		max	min	mean		max	min	mean	max
DTLZ1	**13.56**	28.46	(≈)	45.85	18.52	38.82	(+)	57.98	25.31	35.14	(+)	**40.04**	16.58	**27.29**	52.88
DTLZ1a	0.58	1.68	(≈)	2.72	1.21	7.60	(+)	22.16	**0.07**	**1.15**	(≈)	7.58	0.35	1.47	**2.70**
DTLZ2	0.13	0.18	(+)	0.24	0.36	0.42	(+)	0.48	0.23	0.29	(+)	0.34	**0.09**	**0.09**	**0.17**
DTLZ3	214.3	355.0	(+)	430.9	225.8	253.8	(+)	401.6	204.7	**219.7**	(-)	**239.0**	**199.0**	310.8	420.7
DTLZ3a	31.41	77.06	(+)	212.4	121.6	250.7	(+)	398.6	**2.41**	36.08	(≈)	132.5	**17.05**	**30.65**	58.97
DTLZ4	0.30	0.47	(≈)	0.69	0.50	0.70	(+)	0.82	0.65	0.91	(+)	1.04	**0.23**	**0.43**	**0.68**
DTLZ5	0.10	0.15	(+)	0.25	0.28	0.34	(+)	0.42	0.07	0.12	(+)	0.20	**0.06**	**0.09**	**0.19**
DTLZ6	3.96	4.60	(≈)	5.58	**0.74**	**3.04**	(-)	**4.56**	3.69	4.66	(≈)	5.61	3.83	4.60	5.56
DTLZ7	1.00	4.28	(+)	9.66	2.61	6.57	(+)	8.87	0.59	1.33	(+)	1.78	**0.58**	**0.81**	**1.40**
UF1	0.09	0.12	(+)	0.17	0.14	0.23	(+)	0.37	0.07	0.14	(+)	0.53	**0.09**	**0.09**	**0.10**
UF2	0.06	0.08	(+)	0.14	0.13	0.18	(+)	0.24	**0.05**	0.07	(≈)	**0.10**	0.06	**0.06**	0.07
UF3	0.56	0.84	(+)	1.07	0.61	0.94	(+)	1.35	0.86	1.01	(+)	1.16	**0.34**	**0.48**	**0.65**
UF4	0.09	0.10	(-)	0.12	0.09	0.11	(-)	**0.12**	**0.07**	**0.10**	(-)	0.12	0.14	0.16	0.18
UF5	0.72	1.32	(+)	2.62	2.04	2.85	(+)	4.08	0.61	1.53	(+)	2.06	**0.47**	**0.70**	**1.08**
UF6	0.82	1.24	(+)	2.02	1.14	1.84	(+)	2.76	1.01	1.23	(+)	1.59	**0.82**	**1.06**	**1.24**
UF7	0.88	1.24	(+)	1.47	0.14	0.29	(+)	0.57	0.07	0.11	(+)	0.38	**0.07**	**0.09**	**0.11**

Table 4
Statistical results of the HV values obtained by K-RVEA, MOEA/D-EGO, SMS-EGO, and AB-MOEA with the same number of real function evaluations.

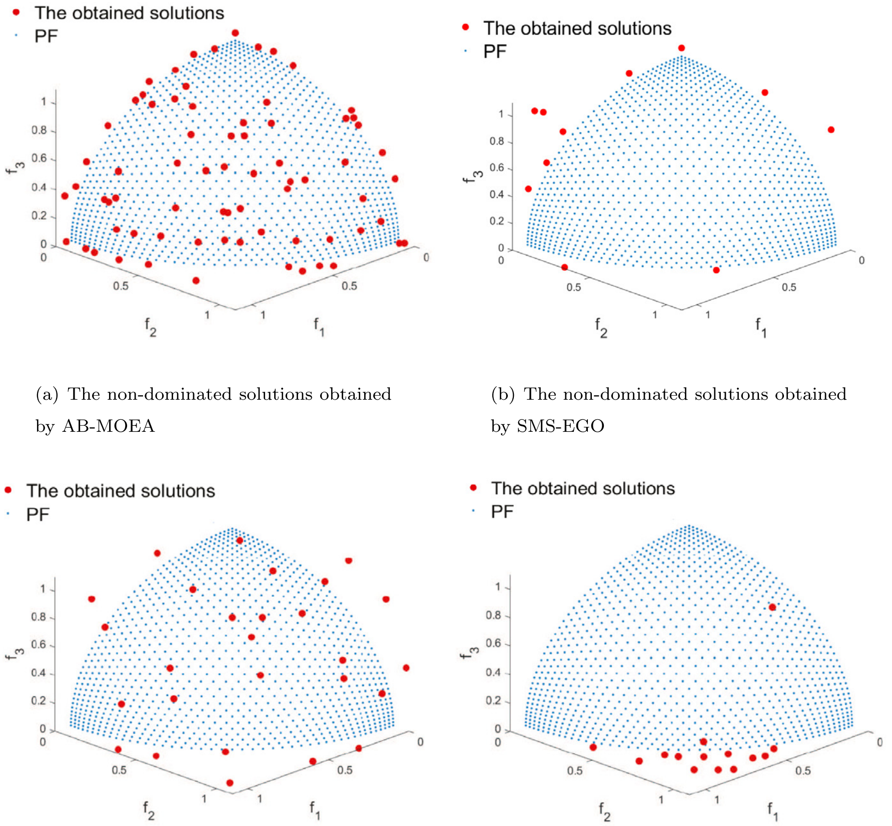
Test problem	K-RVEA				MOEA/D-EGO				SMS-EGO				AB-MOEA		
	min	mean		max	min	mean		max	min	mean		max	min	mean	max
DTLZ1	0.930	0.962	(+)	0.989	0.982	0.983	(+)	0.983	**0.991**	0.992	(+)	0.994	0.971	**0.993**	**0.995**
DTLZ1a	0.936	0.975	(+)	0.993	0.246	0.513	(+)	0.781	**0.999**	**1.000**	(-)	**1.000**	0.973	0.992	0.998
DTLZ2	0.262	0.340	(+)	0.438	0.104	0.124	(+)	0.144	0.215	0.304	(+)	0.393	**0.350**	**0.444**	**0.488**
DTLZ3	0.858	0.900	(+)	0.933	0.870	0.925	(+)	0.979	**0.981**	**0.984**	(-)	**0.986**	0.862	0.921	0.957
DTLZ3a	0.000	0.373	(+)	0.743	0.000	0.000	(+)	0.000	**0.822**	0.844	(≈)	0.865	0.403	**0.855**	**0.979**
DTLZ4	0.000	0.050	(≈)	0.174	0.007	0.015	(+)	0.022	0.000	0.000	(+)	0.000	**0.009**	**0.063**	**0.201**
DTLZ5	0.009	0.068	(+)	0.108	0.000	0.001	(+)	0.002	0.108	0.112	(≈)	0.116	**0.108**	**0.118**	**0.147**
DTLZ6	0.678	0.792	(≈)	0.849	**0.875**	**0.878**	(-)	**0.882**	0.732	0.803	(≈)	0.874	0.751	0.801	0.834
DTLZ7	0.000	0.014	(+)	0.089	0.000	0.000	(+)	0.000	0.066	0.066	(+)	0.087	**0.069**	**0.070**	**0.104**
UF1	0.49	0.56	(+)	0.59	0.26	0.40	(+)	0.51	0.02	0.53	(+)	0.61	**0.55**	**0.57**	**0.58**
UF2	0.60	0.62	(+)	0.64	0.38	0.47	(+)	0.53	0.60	0.63	(≈)	0.66	**0.63**	**0.64**	**0.65**
UF3	0.00	0.03	(+)	0.13	0.00	0.01	(+)	0.05	0.00	0.00	(+)	0.01	**0.07**	**0.18**	**0.29**
UF4	0.27	0.30	(-)	0.31	0.27	0.29	(-)	0.31	**0.27**	**0.30**	(-)	**0.33**	0.21	0.22	0.25
UF5	0.00	0.00	(+)	0.02	0.00	0.00	(+)	0.00	0.00	0.00	(+)	0.01	0.00	**0.03**	**0.14**
UF6	0.18	0.39	(+)	0.48	0.03	0.21	(+)	0.34	0.30	0.42	(+)	0.49	**0.64**	**0.79**	**0.89**
UF7	0.00	0.00	(+)	0.00	0.08	0.24	(+)	0.38	0.28	0.43	(+)	0.48	**0.42**	**0.46**	**0.48**

Table 5
Statistical results of the GD values obtained by K-RVEA, MOEA/D-EGO, SMS-EGO, and AB-MOEA with the same number of real function evaluations.

Test problem	K-RVEA				MOEA/D-EGO				SMS-EGO				AB-MOEA		
	min	mean		max	min	mean		max	min	mean		max	min	mean	max
DTLZ1a	3.52	9.32	(+)	16.05	14.91	31.38	(+)	55.33	**0.01**	13.27	(≈)	49.39	2.64	**3.65**	**4.86**
DTLZ2	0.01	0.03	(+)	0.04	0.07	0.11	(+)	0.14	**0.01**	0.06	(+)	0.12	0.02	**0.04**	**0.10**
DTLZ3a	38.09	64.72	(+)	102.15	127.19	166.70	(+)	215.85	24.91	**69.75**	(≈)	**111.23**	**12.37**	78.13	114.98
DTLZ7	0.18	1.55	(+)	2.63	1.55	1.77	(+)	2.44	**0.01**	0.66	(+)	4.91	0.15	**0.31**	**0.83**
UF2	0.004	0.010	(≈)	0.02	0.04	0.07	(+)	0.11	0.01	0.01	(+)	0.05	**0.006**	**0.009**	**0.014**
UF4	0.016	**0.020**	(−)	**0.023**	0.019	0.024	(−)	0.027	**0.015**	0.025	(−)	0.035	0.027	0.032	0.036
UF6	0.429	0.75	(+)	1.59	0.78	1.24	(+)	2.09	0.43	0.61	(+)	1.57	**0.36**	**0.55**	**1.10**
UF7	**0.008**	0.03	(+)	0.11	0.03	0.16	(+)	0.40	0.02	0.07	(+)	0.26	0.009	**0.014**	**0.029**

Table 6
Statistical results of the spread values obtained by K-RVEA, MOEA/D-EGO, SMS-EGO, and AB-MOEA with the same number of real function evaluations.

Test problem	K-RVEA				MOEA/D-EGO				SMS-EGO				AB-MOEA		
	min	mean		max	min	mean		max	min	mean		max	min	mean	max
DTLZ1a	0.663	1.213	(≈)	1.787	0.528	1.058	(≈)	1.465	**0.439**	1.096	(≈)	1.853	0.883	**0.995**	**1.110**
DTLZ2	**0.470**	**0.594**	(+)	**0.699**	0.587	0.759	(≈)	0.961	0.700	0.909	(+)	1.080	0.545	0.777	1.129
DTLZ3a	0.580	0.848	(+)	1.214	0.560	0.755	(+)	**0.960**	0.711	1.064	(≈)	1.655	**0.225**	**0.611**	1.563
DTLZ7	0.789	1.006	(≈)	1.178	0.773	**0.883**	(+)	**0.986**	**0.335**	1.213	(≈)	5.429	0.821	1.032	1.346
UF2	0.595	0.780	(+)	0.931	0.547	0.859	(+)	1.235	**0.319**	0.833	(+)	1.128	0.533	**0.700**	**0.970**
UF4	0.575	0.708	(≈)	0.859	0.453	0.649	(≈)	0.852	**0.453**	**0.626**	(≈)	0.848	0.538	0.680	**0.810**
UF6	0.780	1.059	(+)	1.881	0.648	1.007	(+)	1.429	0.707	0.884	(≈)	**1.106**	**0.707**	**0.873**	1.174
UF7	0.673	0.996	(+)	1.340	0.666	0.968	(+)	1.516	**0.441**	0.908	(≈)	1.681	0.596	**0.795**	**1.046**



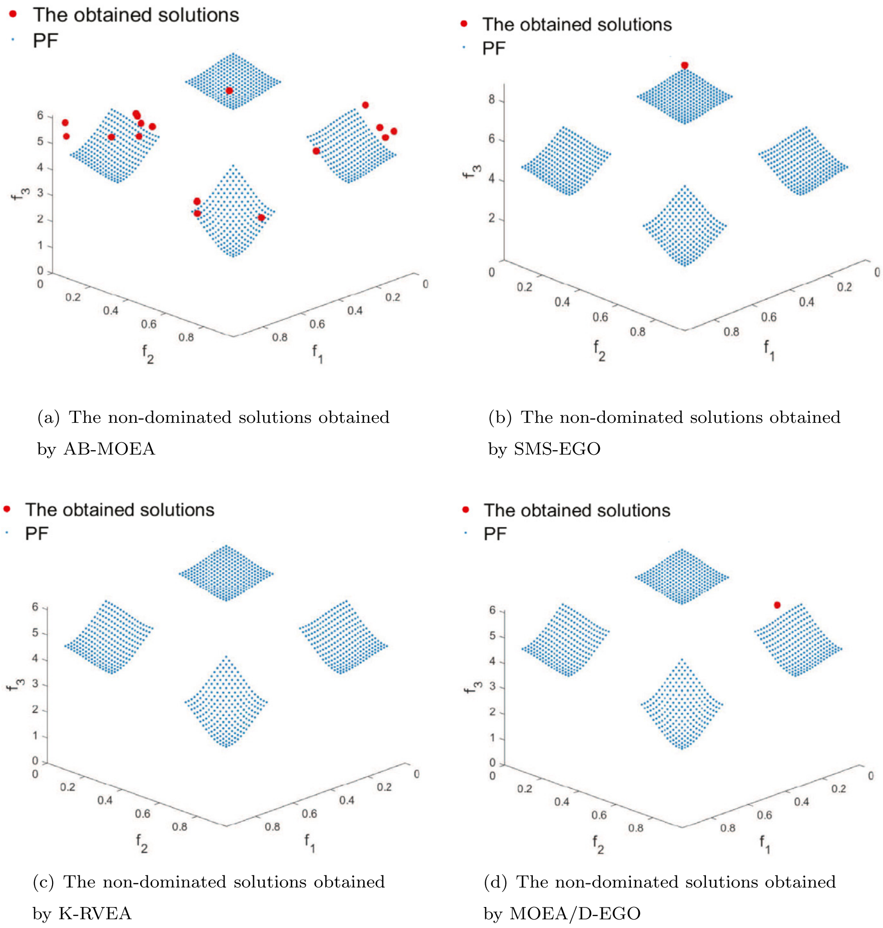
(a) The non-dominated solutions obtained by AB-MOEA

(b) The non-dominated solutions obtained by SMS-EGO

(c) The non-dominated solutions obtained by K-RVEA

(d) The non-dominated solutions obtained by MOEA/D-EGO

Fig. 4. Visualization of the non-dominated solutions obtained by the four compared algorithms on DTLZ2.



(a) The non-dominated solutions obtained by AB-MOEA

(b) The non-dominated solutions obtained by SMS-EGO

(c) The non-dominated solutions obtained by K-RVEA

(d) The non-dominated solutions obtained by MOEA/D-EGO

Fig. 5. Visualization of the non-dominated solutions obtained by the four compared algorithms on DTLZ7.

are calculated on part of the test problems and the results are presented in Tables 5 and 6, respectively. The best one of each instance is highlighted. To further illustrate the advantage of the proposed algorithm, the non-dominated solution set obtained by each of the four compared algorithms (in the run that achieved the medium performance out of 20 independent runs) on DTLZ2 and DTLZ7 are visualized in Figs. 4 and 5. From the results, we can clearly see the better performance of the proposed algorithm.

The statistical results in terms of IGD values obtained by the four algorithms are summarized in Table 3. For the three-objective benchmarks, it is apparent that AB-MOEA has achieved the best approximate PF on all test problems except for DTLZ3 (SMS-EGO obtained the best IGD values) and DTLZ6 (MOEA/D-EGO obtained the best IDG values). The reason behind this may be that DTLZ3 has a multimodal fitness landscape, and DTLZ6 has a plenty of disconnected Pareto optimal regions in the decision space. Actually, in our experiments, all algorithms failed to converge to the PF on DTLZ3 and DTLZ6 due to the limited budget of function evaluations. For the bi-objective test suite, we can clearly see that the IGD results obtained by AB-MOEA are much better than the compared algorithms on all UF test problems except on UF4. According to the Wilcoxon rank sum test, the proposed algorithm significantly outperforms the compared algorithms on most of the test problems.

Similar conclusions can be drawn from the results given in Table 4. From these results, we can see that AB-MOEA performs much better than K-RVEA on all three-objective benchmark problems considered in this work but DTLZ4 and DTLZ6, on which AB-MOEA and K-RVEA perform comparably. MOEA/D-EGO is able to achieve well converged and evenly distributed final solutions on DTLZ6; however, its performance on the rest of problems is much worse than AB-MOEA. Regarding the bi-objective benchmarks, the best results of HV on UF1-UF3 and UF5-UF7 are obtained by AB-MOEA. It is worthy of noting that SMS-EGO outperforms AB-MOEA on DTLZ1a, DTLZ3 and UF4 in terms of the HV metric. Note, however, that SMS-EGO is very time-consuming [29]. In conclusion, the proposed algorithm shows the best overall performance.

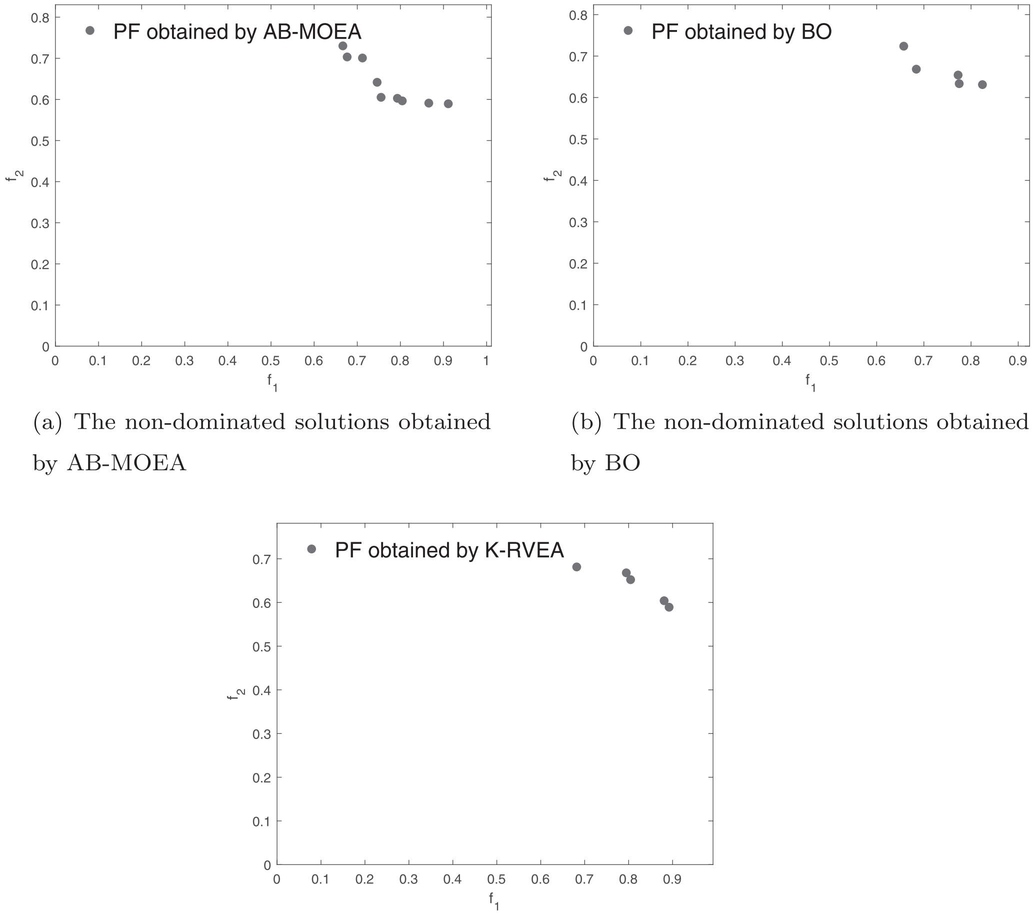
To further verify the performance of AB-MOEA, two performance indicators, GD and spread metrics are adopted to investigate the convergence and diversity of the four algorithms. Since the algorithms failed to converge to the PF on some of the benchmarks due to the limited evaluation budget, we select some of them to examine the convergence and diversity performance separately. As listed in 5, AB-MOEA obtains the best GD values compared to other three algorithms on a vast majority of the test problems, confirming the better performance indicated by IGD and HV values. On the other hand, the values of the spread metrics given in 6 indicate that AB-MOEA shows certain advantages over MOEA/D-EGO and K-RVEA concerning the diversity of the obtained solutions. Note that SMS-EGO also shows competitive diversity performance; however, it is significantly outperformed by AB-MOEA with respect to the convergence performance.

5. A Case study on airfoil design

To verify the performance of the proposed algorithm on real-world optimization problems, we compare AB-MOEA, K-RVEA and Bayesian optimization (BO) on a transonic airfoil design optimization problem. It is noted that time-consuming computational fluid dynamics (CFD) simulations are required to evaluate the performance of an airfoil design, which is therefore a computationally expensive optimization problem and a limited budget of evaluations (here, 300 evaluations) is affordable. In the following, a brief introduction to the problem is given and then the simulation results are presented.

We consider an airfoil design problem based on the RAE2822 airfoil test case from the GARTEUR AG52 project [25]. Here, 14 points of a nonrational B-spline (NURBS) are used to control the curvature of the upper and lower surfaces of the airfoil, then CFD simulations are carried out to calculate the drag and lift coefficients (C_d and C_l) for any given geometry of the airfoil. We aim to optimize the geometry of the airfoil to minimize the drag coefficient and maximize the lift coefficient.

We compare AB-MOEA with K-RVEA and BO [37] in terms of IGD and HV. Each algorithm is run for ten times on the airfoil design problem and the mean values of HV and IGD are presented in Table 7. Note that in calculating IGD, the reference set is the non-dominated solution set of all solutions obtained by three compared algorithms. The non-dominated solution set obtained by each algorithm is also visualized in Fig. 6. From Table 7 and Fig. 6, we can see that the proposed algorithm has also achieved the best results on the real-world design problem.



(a) The non-dominated solutions obtained by AB-MOEA

(b) The non-dominated solutions obtained by BO

(c) The non-dominated solutions obtained by K-RVEA

Fig. 6. The obtained non-dominated solutions on the airfoil design problem.

Table 7
The mean values of HV and IGD obtained by AB-MOEA, BO, K-RVEA on the airfoil design problem.

Algorithm	HV	IGD
AB-MOEA	**0.0423**	**0.0328**
BO	0.0407	0.0389
K-RVEA	0.0348	0.0402

6. Conclusion

Surrogate-assisted evolutionary algorithms have shown great promises to solve expensive multi-objective optimization problems. However, the limited budget of function evaluations requires that the algorithm is able to quickly converge while ensuring a diverse distribution of the obtained solutions. Therefore, it is highly demanded to design SAEAs that can achieve an optimal balance between exploration and exploitation. For this purpose, we proposed an adaptive Bayesian approach to surrogate-assisted evolutionary algorithm to improve the efficiency in solving expensive MOPs. Taken both the diversity and convergence into account, an adaptive acquisition function is designed by adjusting the weights of the uncertainty and the mean objective value in the acquisition function on top of an adaptive sampling selection criterion that can better balance diversity and convergence. The effectiveness of the introduced strategies and the performance of the proposed algorithm are investigated on a set of widely used benchmark problems and an airfoil design optimization problem. Our results demonstrate that the proposed algorithm significantly outperforms the compared algorithms on most test problems and is also able to achieve better performance on the airfoil design problem.

Although the proposed algorithm is competitive for solving most test problems used in our experiments, we find that the proposed algorithm suffers from slow convergence when solving DTLZ1, DTLZ3 and UF6. This might be attributed to the poor prediction quality of GP models due to the strong ruggedness of the fitness landscape. Consequently, future research could investigate the use of multiple surrogate models and more sophisticated methods for estimating the uncertainty. In addition, new acquisition functions are desirable when dealing with MOPs whose objective functions and / or constraints have very different computational complexities.

Declaration of Competing Interest

None.

CRediT authorship contribution statement

Xilu Wang: Formal analysis, Writing - original draft. **Yaochu Jin:** Conceptualization, Formal analysis, Writing - original draft. **Sebastian Schmitt:** Conceptualization, Formal analysis, Writing - original draft. **Markus Olhofer:** Conceptualization, Formal analysis, Writing - original draft.

Acknowledgments

The authors are grateful to Dr Handing Wang for her assistance in testing the algorithms on the airfoil design problem. The work was supported in part by a Royal Society International Exchanges Program under No. IEC\NSFC\170279.

References

[1] R.T. Marler, J.S. Arora, Survey of multi-objective optimization methods for engineering, Struct. Multidisc. Optim. 26 (6) (2004) 369–395.
[2] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, T. Basar, Coalitional game theory for communication networks, IEEE Signal Process. Mag. 26 (5) (2009) 77–97.
[3] E.E. Tsiropoulou, P. Vamvakas, G.K. Katsinis, S. Papavassiliou, Combined power and rate allocation in self-optimized multi-service two-tier femtocell networks, Comput. Commun. 72 (2015) 38–48.
[4] F.A. Mohamed, H.N. Koivo, Multiobjective optimization using modified game theory for online management of microgrid, Eur. Trans. Electr. Power 21 (1) (2011) 839–854.
[5] R. Meng, Y. Ye, N.-g. Xie, Multi-objective optimization design methods based on game theory, in: 2010 8th World Congress on Intelligent Control and Automation, IEEE, 2010, pp. 2220–2227.
[6] E.E. Tsiropoulou, P. Vamvakas, S. Papavassiliou, Joint utility-based uplink power and rate allocation in wireless networks: a non-cooperative game theoretic framework, Phys. Commun. 9 (2013) 299–307.
[7] Y. Jin, Surrogate-assisted evolutionary computation: Recent advances and future challenges, Swarm Evol. Comput. 1 (2) (2011) 61–70.
[8] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[9] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731.
[10] R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A reference vector guided evolutionary algorithm for many-objective optimization, IEEE Trans. Evol. Comput. 20 (5) (2016) 773–791.
[11] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the strength pareto evolutionary algorithm, TIK-Rep. 103 (2001).
[12] R. Cheng, Y. Jin, K. Narukawa, B. Sendhoff, A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling, IEEE Trans. Evol. Comput. 19 (6) (2015) 838–856.

[13] D. Rodriguez-Roman, A surrogate-assisted genetic algorithm for the selection and design of highway safety and travel time improvement projects, Saf. Sci. 103 (2018) 305–315.

[14] Y. Jin, B. Sendhoff, A systems approach to evolutionary multiobjective structural optimization and beyond, IEEE Comput. Intell. Mag. 4 (3) (2009) 62–76.

[15] A. Habib, H.K. Singh, T. Ray, A multiple surrogate assisted multi/many-objective multi-fidelity evolutionary algorithm, Inf. Sci. 502 (2019) 537–557.

[16] Y. Jin, H. Wang, T. Chugh, D. Guo, K. Miettinen, Data-driven evolutionary optimization: an overview and case studies, IEEE Trans. Evol. Comput. 23 (3) (2018) 442–458.

[17] B.D. Marjavaara, T.S. Lundström, T. Goel, Y. Mack, W. Shyy, Hydraulic turbine diffuser shape optimization by multiple surrogate model approximations of pareto fronts, J. Fluids Eng. 129 (9) (2007) 1228–1240.

[18] J. Zhang, A. Zhou, G. Zhang, A classification and pareto domination based multiobjective evolutionary algorithm, in: 2015 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2015, pp. 2883–2890.

[19] L. Pan, C. He, Y. Tian, H. Wang, X. Zhang, Y. Jin, A classification-based surrogate-assisted evolutionary algorithm for expensive many-objective optimization, IEEE Trans. Evol. Comput. 23 (1) (2018) 74–88.

[20] B. Shahriari, K. Swersky, Z. Wang, R.P. Adams, N. De Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proc. IEEE 104 (1) (2015) 148–175.

[21] J. Snoek, H. Larochelle, R.P. Adams, Practical Bayesian optimization of machine learning algorithms, in: Advances in Neural Information Processing Systems, 2012, pp. 2951–2959.

[22] F.A. Viana, R.T. Haftka, L.T. Watson, Efficient global optimization algorithm assisted by multiple surrogate techniques, J. Glob. Optim. 56 (2) (2013) 669–689.

[23] D. Buche, N.N. Schraudolph, P. Koumoutsakos, Accelerating evolutionary algorithms with Gaussian process fitness function models, IEEE Trans. Syst. Man Cybern.Part C (Appl. Rev.) 35 (2) (2005) 183–194.

[24] N. Hoyle, N.W. Bressloff, A.J. Keane, Design optimization of a two-dimensional subsonic engine air intake, AIAA J. 44 (11) (2006) 2672–2681.

[25] H. Wang, Y. Jin, J. Doherty, Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems, IEEE Trans. Cybern. 47 (9) (2017) 2664–2677.

[26] J. Tian, Y. Tan, J. Zeng, C. Sun, Y. Jin, Multi-objective infill criterion driven Gaussian process assisted particle swarm optimization of high-dimensional expensive problems, IEEE Trans. Evol. Comput. 23 (3) (2019) 459–472.

[27] J. Knowles, ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems, IEEE Trans. Evol. Comput. 10 (1) (2006) 50–66.

[28] B. Naujoks, N. Beume, M. Emmerich, Metamodel-assisted SMS-EMOA applied to airfoil optimization tasks, in: Proceedings EUROGEN, 5, 2005.

[29] Q. Zhang, W. Liu, E. Tsang, B. Virginas, Expensive multiobjective optimization by MOEA/D with Gaussian process model, IEEE Trans. Evol. Comput. 14 (3) (2009) 456–474.

[30] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, K. Sindhya, A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization, IEEE Trans. Evol. Comput. 22 (1) (2018) 129–142.

[31] D. Guo, Y. Jin, J. Ding, T. Chai, Heterogeneous ensemble-based infill criterion for evolutionary multiobjective optimization of expensive problems, IEEE Trans. Cybern. 49 (3) (2018) 1012–1025.

[32] C.E. Rasmussen, Gaussian processes in machine learning, in: Summer School on Machine Learning, Springer, 2003, pp. 63–71.

[33] D.R. Jones, M. Schonlau, W.J. Welch, Efficient global optimization of expensive black-box functions, J. Glob. Optim. 13 (4) (1998) 455–492.

[34] D. Zhan, Y. Cheng, J. Liu, Expected improvement matrix-based infill criteria for expensive multiobjective optimization, IEEE Trans. Evol. Comput. 21 (6) (2017) 956–975.

[35] J.M. Hernández-Lobato, M.W. Hoffman, Z. Ghahramani, Predictive entropy search for efficient global optimization of black-box functions, in: Advances in Neural Information Processing Systems, 2014, pp. 918–926.

[36] N. Srinivas, A. Krause, S.M. Kakade, M. Seeger, Gaussian process optimization in the bandit setting: no regret and experimental design (2010) 1015–1022.

[37] J. Liu, Z. Han, W. Song, Comparison of infill sampling criteria in kriging-based aerodynamic optimization, in: 28th Congress of the International Council of the Aeronautical Sciences, 2012, pp. 23–28.

[38] R. Cheng, T. Rodemann, M. Fischer, M. Olhofer, Y. Jin, Evolutionary many-objective optimization of hybrid electric vehicle control: From general optimization to preference articulation, IEEE Trans. Emerg. Top.Comput. Intell. 1 (2) (2017) 97–111.

[39] M. Emmerich, N. Beume, B. Naujoks, An EMO algorithm using the hypervolume measure as selection criterion, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2005, pp. 62–76.

[40] M.T. Emmerich, K.C. Giannakoglou, B. Naujoks, Single-and multiobjective evolutionary optimization assisted by gaussian random field metamodels, IEEE Trans. Evol. Comput. 10 (4) (2006) 421–439.

[41] J. Hensman, N. Fusi, N.D. Lawrence, Gaussian processes for big data, in: Conference on Uncertainty in Artificial Intelligence, 2013, pp. 282–290.

[42] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable multi-objective optimization test problems, in: Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02, 1, IEEE, 2002, pp. 825–830.

[43] Q. Zhang, A. Zhou, S. Zhao, P.N. Suganthan, W. Liu, S. Tiwari, Multiobjective optimization test instances for the CEC 2009 special session and competition, Technical Report, 2008.

[44] C. Yang, J. Ding, Y. Jin, T. Chai, Off-line data-driven multi-objective optimization: knowledge transfer between surrogates and generation of final solutions, IEEE Trans. Evol. Comput. (2019).

[45] Q. Zhang, A. Zhou, Y. Jin, RM-MEDA: a regularity model-based multiobjective estimation of distribution algorithm, IEEE Trans. Evol. Comput. 12 (1) (2008) 41–63.

[46] L. While, P. Hingston, L. Barone, S. Huband, A faster algorithm for calculating hypervolume, IEEE Trans. Evol. Comput. 10 (1) (2006) 29–38.

[47] D.A. Van Veldhuizen, G.B. Lamont, Multiobjective evolutionary algorithm research: a history and analysis, Technical Report, Citeseer, 1998.

[48] Y.-N. Wang, L.-H. Wu, X.-F. Yuan, Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure, Soft Comput. 14 (3) (2010) 193.

[49] Y. Tian, R. Cheng, X. Zhang, Y. Jin, PlatEMO: a MATLAB platform for evolutionary multi-objective optimization, IEEE Comput. Intell. Mag. 12 (4) (2017) 73–87.

[50] S.N. Lophaven, H.B. Nielsen, J. Sondergaard, DACE - A Matlab Kriging Toolbox, 2002.