Data-Driven Evolutionary Optimization: An Overview and Case Studies

Yaochu Jin[1,2], Handing Wang[3,1], Tinkle Chugh[4], Dan Guo[5], and Kaisa Miettinen[6]

[1] Department of Computer Science, University of Surrey, Guildford GU2 7XH, U.K
[2] Taiyuan University of Science and Technology, Taiyuan 030024, China
[3] School of Artificial Intelligence, Xidian University, Xi'an 710071, China
[4] Department of Computer Science, University of Exeter, United Kingdom, U.K
[5] the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China
[6] University of Jyvaskyla, Faculty of Information Technology, P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland

Abstract. Most evolutionary optimization algorithms assume that the evaluation of the objective and constraint functions is straightforward. In solving many real-world optimization problems, however, such objective functions may not exist, instead computationally expensive numerical simulations or costly physical experiments must be performed for fitness evaluations. In more extreme cases, only historical data are available for performing optimization and no new data can be generated during optimization. Solving evolutionary optimization problems driven by data collected in simulations, physical experiments, production processes, or daily life are termed data-driven evolutionary optimization. In this paper, we provide a taxonomy of different data driven evolutionary optimization problems, discuss main challenges in data-driven evolutionary optimization with respect to the nature and amount of data, and the availability of new data during optimization. Real-world application examples are given to illustrate different model management strategies for different categories of data-driven optimization problems.

Keywords: Data-driven optimization· evolutionary algorithms· surrogate· model management· data science· machine learning

1 Introduction

Many real-world optimization problems are difficult to solve in that they are non-convex or multi-modal, large-scale, highly constrained, multi-objective, and subject to a large amount of uncertainties. Furthermore, the formulation of the optimization problem itself can be challenging, requiring a number of iterations between the experts of the application area and computer scientists to specify the appropriate representation, objectives, constraints and decision variables [52, 14, 84].

Over the past decades, evolutionary algorithms (EAs) have become a popular tool for optimization [17, 23]. Most existing research on EAs is based on an implicit assumption that evaluating the objectives and constraints of candidate solutions is easy and cheap. However, such cheap functions do not exist for many real-world optimization problems. Instead, evaluations of the objectives and / or constraints can be performed only based on data, collected either from physical experiments, numerical simulations, or daily life. Such optimization problems can be called data-driven optimization problems [100]. In addition to the challenges coming from the optimization, data-driven optimization may also be subject to difficulties resulting from the characteristics of data. For example, the data may be distributed, noisy, heterogeneous, or dynamic (streaming data), and the amount of data may be big or small, imposing different challenges to the data-driven optimization algorithm.

In some data-driven optimization problems, evaluations of the objective or constraint functions involve time- or resource-intensive physical experiments or numerical simulations (often referred to as simulation-based optimization). For example, a single function evaluation based on computational fluid dynamic (CFD) simulations could take from minutes to hours [52]. To reduce the computational cost, surrogate models (also known as meta-models [20]) have been widely used in EAs, which are known as surrogate-assisted evolutionary algorithms (SAEAs) [49]. SAEAs perform a limited number of real function evaluations and only a small amount of data is available for training surrogate models to approximate the objective and / or constraint functions [45, 12]. Most machine learning models, including polynomial regression [120], Kriging model [11, 7], artificial neural networks (ANN) [50, 109, 51], and radial basis function networks (RBFN) [113, 80, 90, 88] have been employed in SAEAs. With limited training data, approximation errors of surrogate models are inevitable, which may mislead the evolutionary search. However,

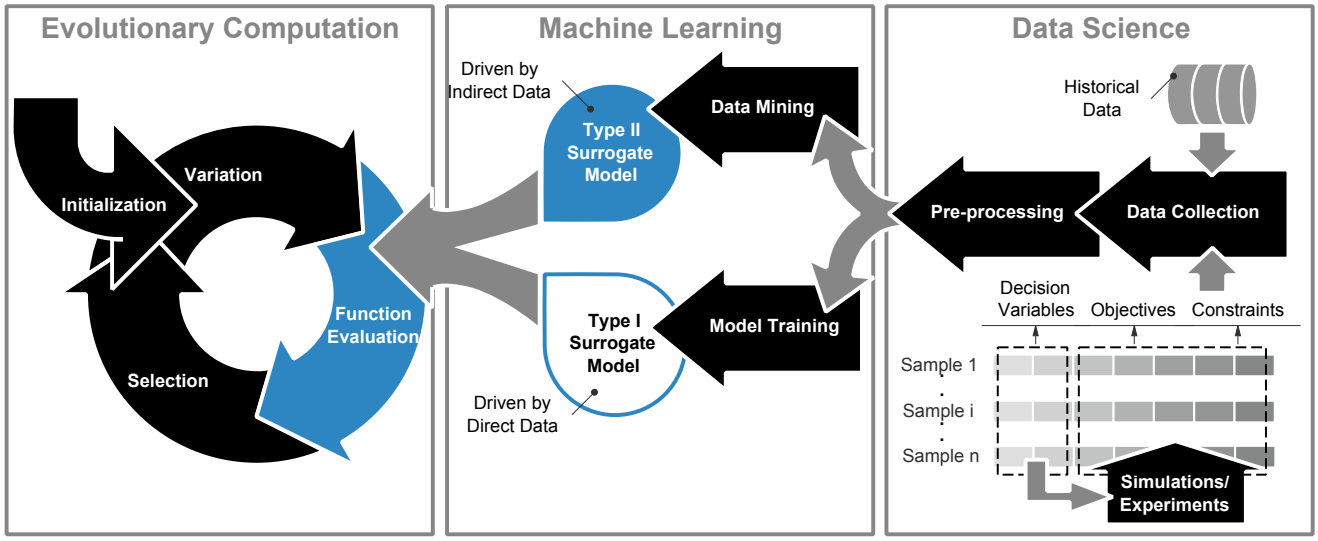

Fig. 1. Main components of data-driven evolutionary optimization.

as shown in [47, 61], an EA may benefit from the approximation errors introduced by surrogates, and therefore, it is essential in SAEAs to make full use of the limited data.

In contrast to the above situation in which collecting data is expensive and only a small amount of data is available, there are also situations in which function evaluations must be done on the basis a large amount of data. The hardness brought by data to data-driven EAs is twofold. Firstly, acquiring and processing data for function evaluations increase the resource and computational cost, especially when there is an abundant amount of data [118]. For example, a single function evaluation of the trauma system design problem [100] needs to process 40,000 emergency incident records. Secondly, the function evaluations based on data are the approximation of the exact function evaluations, because the available data is usually not of ideal quality. Incomplete [65], imbalanced [108, 107], and noisy [46, 106] data bring errors to function evaluations of data-driven EAs, which may mislead the search.

This paper aims to provide an overview of recent advances in the emerging research area of data-driven evolutionary optimization. Section 2 provides more detailed background about data-driven optimization, including a categorization with respect to the nature of the data, whether new data can be collected during optimization, and the surrogate management strategies used in data-driven optimization. Five case studies of real-world data-driven optimization problems are presented in Section 3, representing situations where the amount of data is either small or big, and new data is or is not allowed to be generated during optimization. Open issues for future work in data-driven optimization are discussed in detail in Section 4, and Section 5 concludes the paper.

2 Data-Driven Evolutionary Optimization

Generally speaking, EAs begin with a randomly initialized parent population. In each iteration of EAs, an offspring population is generated via a number of variation operators, crossover and mutation, for instance. All solutions in the offspring population will then be evaluated to calculate their fitness value and assess their feasibility. Then, the new parent population for the next iteration is selected from the offspring population or a combination of the parent and offspring populations.

Fig. 1 presents the three main disciplines involved in data-driven evolutionary optimization, namely, evolutionary computation (including other population-based meta-heuristic search methods), machine learning (including all learning techniques), and data science. While the traditional challenges remain to be handled in each discipline, new research questions may arise when machine learning models are built for efficiently guiding the evolutionary search given various amounts and types of data.

Although they are widely used, surrogates in data-driven evolutionary optimization have a much broaden sense than in surrogate-assisted evolutionary optimization. For example, the "surrogate" in the case study in Section III.B is more a way of reducing the amount of data to be used in fitness evaluations rather than an explicit surrogate model, where an update of the "surrogate" is to adaptively find the right number of data clusters.

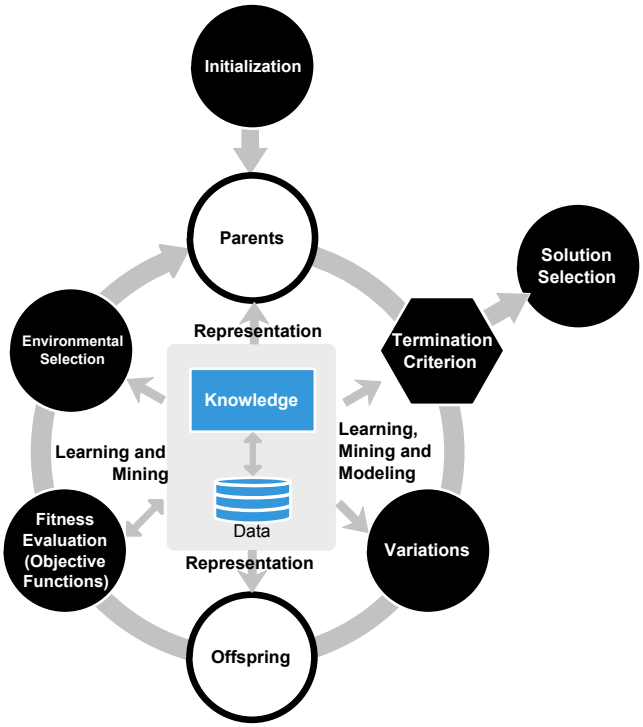

Fig. 2. Data or knowledge that can be incorporated in various components of EAs.

It should also be emphasized that data or domain knowledge can be utilized to speed up the evolutionary search almost in every component of an evolutionary algorithm, as illustrated in Fig. 2. For example, history data can be used to determine the most effective and compact representation of a very large scale complex problem [28]. We also want to note that domain knowledge about the problem structure or information about the search performance acquired in the optimization process can be incorporated or re-used in EAs to enhance the evolutionary search performance. These techniques are usually known as knowledge incorporation in EAs [43].

In the following, we discuss in detail the challenges in data collection and surrogate construction arising from data-driven optimization.

2.1 Data Collection

Different data-driven optimization problems may have completely different data resources and data collection methods. Roughly speaking, data can be classified into two large types: direct and indirect data, consequently resulting in two different types of surrogate modelling and management strategies, as shown in Fig. 1.

- One type of data in data-driven optimization is directly collected from computer simulations or physical experiments, in which case each data item is composed of the decision variables, corresponding objective and / or constraint values, as shown in the bottom right panel of Fig. 1. This type of data can be directly used to train surrogate models to approximate the objective and / or constraint functions, which has been the main focus in SAEAs [45, 12]. We call surrogate models built from direct data Type I surrogate models. Note that during the optimization, EAs may or may not be allowed to actively sample new data.
- The second type of data is called indirect data. For example, some of the objective and constraint functions in the trauma system design problem [100] can only be calculated using emergency incident records. In this case, the data are not presented in the form of decision variables and objective values. However, objective and constraint values can be calculated using the data, which are then further used for training surrogates. We term surrogate models based on indirect data Type II surrogate models. In contrast to direct data, it is usually less likely, if not impossible, for EAs to actively sample new data during optimization.

In addition to the difference in the presentation form of the data, other properties related to data are also essential for data-driven evolutionary optimization, including the cost of collecting data, whether new data is allowed to be

collected during the optimization, and whether data collection can be actively controlled by the EA. Last but not the least, data of multiple fidelity can also be made available for both data types [62, 26, 63, 103].

In the following, we divide data-driven EAs into off-line and on-line methodologies, according to whether new data is allowed to be actively generated by the EA [100].

2.2 Off-line and On-line Data-Driven Optimization Methodologies

Off-line Data-Driven Optimization Methodologies In off-line data-driven EAs, no new data can be actively generated during the optimization process [104], presenting serious challenges to surrogate management. Since no new data can be actively generated, off-line data-driven EAs focus on building surrogate models based on the given data to explore the search space. In this case, the surrogate management strategy heavily relies on the quality and amount of the available data.

- **Data with non-ideal quality**: Real-world data can be incomplete [65], imbalanced [22], or noisy [46, 106]. Consequently, construction of surrogates must take into account these challenges and nevertheless, the resulting surrogates are subject to large approximation errors that may mislead the evolutionary search.
- **Big data**: In off-line data-driven optimization, the amount of the data can be huge, which results in prohibitively large computational cost for data processing and fitness calculation based on the data [118]. The computational cost of building surrogate models also dramatically increases with the increasing amount of the training data.
- **Small data**: Opposite to big data, the amount of available data may be extremely small due to the limited time and resource available for collecting data. Data paucity is often attributed to the fact that numerical simulations of complex systems are computationally very intensive, or physical experiments are very costly. A direct challenge resulting from small data is the poor quality of the surrogates, in particular for off-line data-driven optimization where no new data can be generated during optimization.

Note, however, that a standard criterion to quantify big data and small data still lacks [117], as a sensible definition may depend on the problem and the computational resources available for solving the problem at hand.

Because of the above-mentioned challenges, not many off-line data-driven EAs have been proposed. The strategies for handling data in off-line data-driven EAs can be divided into three categories: data pre-processing, data mining, and synthetic data generation, as shown in Fig. 3.

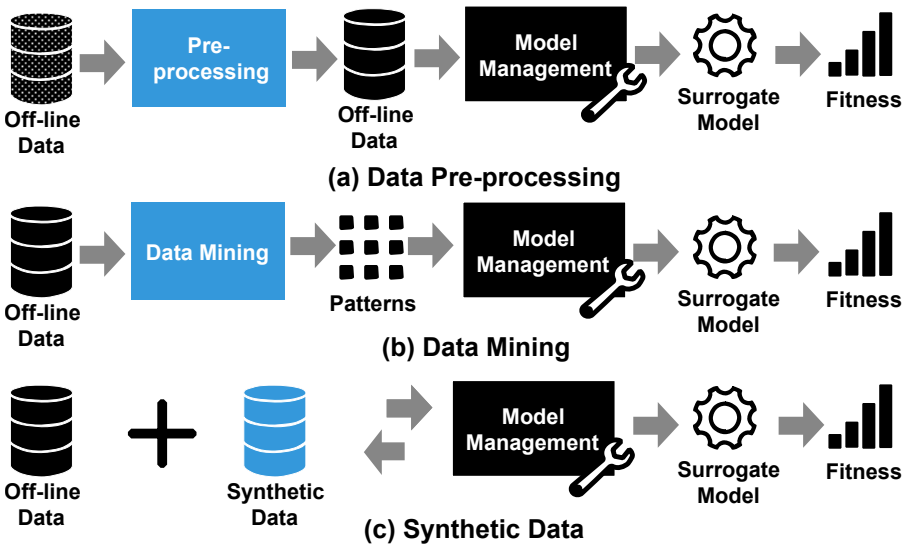

Fig. 3. An illustration of three strategies of handling off-line data in data-driven EAs: data pre-processing, data mining, and generation of synthetic data.

1. **Data pre-processing**: For data with non-ideal quality, pre-processing is necessary. As highlighted in Fig. 3 (a), off-line data must be pre-processed before they are used to train surrogates to enhance the performance of data-driven EAs. Taking the blast furnace problem in [10] as an example, which is a many-objective optimization

problem, the available data collected from production is very noisy. Before building surrogates to approximate the objective functions, a local regression smoothing [16] is used to reduce the noise in the off-line data. Then, Kriging models are built to assist the reference vector guided evolutionary algorithm (RVEA) [9].

2. **Data mining**: When data-driven EAs involve big data, the computational cost may be unaffordable. Since big data often has redundancy [110], existing data mining techniques can be employed to capture the main patterns in the data. As shown in Fig. 3 (b), the data-driven EA is based on the obtained patterns rather than the original data to reduce the computational cost. In the trauma system design problem [100], there are 40,000 records of emergency incidents and a clustering technique is adopted to mine patterns from the data before building surrogate models.
3. **Synthetic data generation**: When the quantity of the data is small and no new data is allowed to be generated, it is extremely challenging to obtain high-quality surrogate models. To address this problem, synthetic data can be generated in addition to the off-line data, as shown in Fig. 3 (c). This idea has shown to be helpful in data-driven optimization of the fused magnesium furnace optimization problem [29], where the size of available data is extremely small and it is impossible to obtain new data during optimization. In the proposed algorithm in [29], a low-order polynomial model is employed to replace the true objective function to generate synthetic data for model management during optimization.

Off-line data-driven EAs are of practical significance in industrial optimization. However, it is hard to validate the obtained optimal solutions before they are really implemented.

On-line Data-Driven Optimization Methodologies Compared with off-line data-driven EAs, on-line data-driven EAs can make additional data available for managing the surrogate models, as shown in Fig. 4. Thus, on-line data-driven EAs are more flexible than off-line data-driven EAs, which offers many more opportunities to improve the performance of the algorithm than off-line data-driven EAs.

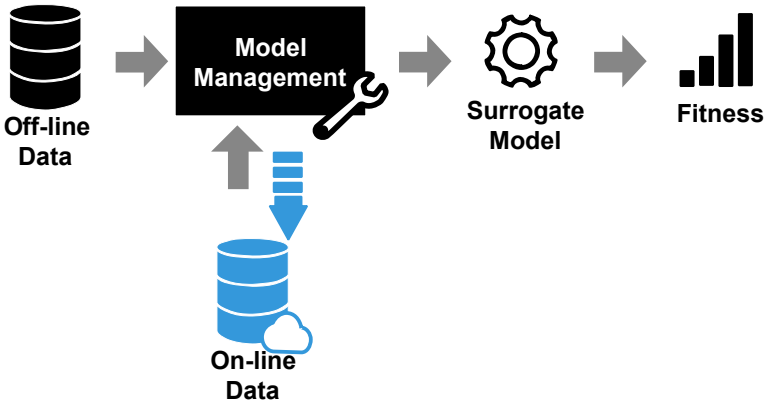

Fig. 4. Surrogate model management in on-line data-driven EAs.

Note that off-line data-driven EAs can be seen as a special case of on-line data-driven EAs in that usually, a certain amount of data needs to be generated to train surrogates before the optimization starts. Thus, methodologies developed for off-line data-driven EAs discussed above can also be applied in on-line data-driven EAs. In the following, we focus on the strategies for managing surrogates during the optimization.

It should be pointed out that generation of new data in on-line data-driven optimization may or may not be actively controlled by the EA. If the generation of new data cannot be controlled by the EA, the main challenge is to promptly capture the information from the new data to guide the optimization process. To the best of our knowledge, no dedicated data-driven EAs have been reported to cope with optimization problems where new data are available but cannot be actively controlled by the EA, which happens when streaming data is involved. In case the EA is able to actively control data generation, desired data can be sampled to effectively update the surrogate models and guide the optimization performance. The frequency and choice of new data samples are important for updating surrogate models. Many model management strategies have been developed, which are mostly generation-based or individual-based [45, 50]. Generation-based strategies [36] adjust the frequency of sampling new data generation by generation, while individual-based strategies choose to sample part of the individuals at each generation.

For on-line data-driven EAs using generation-based model management strategies, the whole population in η generations is re-sampled to generate new data, then the surrogate models are updated based on the new data. The parameter η can be predefined [79, 8] or adaptively tuned according to the quality of the surrogate model [49].

Compared to generation-based strategies, individual-based strategies are more flexible [50, 6]. Typically, two types of sample solutions have been shown to be effective, the samples whose fitness is predicted to be promising, and those whose predicted fitness has a large degree of uncertainty according to the current surrogate.

- **Promising samples** are located around the optimum of the surrogate model, and the accuracy of the surrogate model in the promising area is enhanced once the promising solutions are sampled [49, 50].
- **Uncertain samples** are located in the search space where the surrogate model is likely to have a large approximation error and has not been fully explored by the EA. Thus, sampling these solutions can strengthen exploration of data-driven EAs and most effectively improve the approximation accuracy of the surrogate [20, 45, 6]. So far, different methods for estimating the degree of uncertainty in fitness prediction have been proposed [101]. Probabilistic surrogates such as Kriging models [11, 4] themselves are able to provide a confidence level for their predictions, becoming the most widely used surrogates when the adopted model management needs to use the uncertainty information. In addition, the distance from the sample solution to the existing training data has been used as an uncertainty measure in [6]. Finally, ensemble machine learning models have been proved to be promising in providing the uncertainty information, where the variance of the predictions outputted by the base learners of the ensemble can be used to estimate the degree of uncertainty in fitness prediction [30, 102].

Both promising and uncertain samples are important for on-line data-driven EAs. A number of selection criteria can be adopted to strike a balance between these two types of samples in individual-based strategies, also known as infill sampling criterion or acquisition function in Bayesian optimization [82]. Existing infill criteria include the expected improvement (ExI) [77, 72], probability of improvement (PoI) [21], and lower confidence bound (LCB) [64]. These infill criteria typically aggregate the predicted fitness value and the estimated uncertainty of the predicted fitness into a single-objective criterion. There are also studies that separately select these two types of samples in the individual-based strategies, for instance in [14, 102]. Most recently, a multi-objective infill criterion has been proposed [94], which considers the infill sampling as a bi-objective problem that simultaneously minimizes the predicted fitness and the estimated variance of the predicted fitness. Then, the solutions on the first and last non-dominated fronts are chosen as new infill samples. The proposed multi-objective infill criterion is empirically shown to be promising, in particular for high-dimensional optimization problems.

3 Case Studies

In this section, we present five real-world data-driven optimization problems, including blast furnace optimization, trauma system design, fused magnesium furnace optimization, airfoil shape design, and design of an air intake ventilation system. Four of the five case studies involve multiple objectives. These five applications belong to different data-driven optimization problems in terms of data type, data amount, and availability of new data, as listed in Table 1.

Table 1. Characteristics of five case studies

Sec. No.	Application	Data type	Data quantity	New data availability	No. of objectives
3.1	Blast furnace optimization	Direct	Small	Off-line	8
3.2	Trauma system optimization	Indirect	Big	Off-line	2
3.3	Magnesium furnace performance optimization	Direct	Small	Off-line	3
3.4	Airfoil shape optimization	Direct	Small	On-line	1
3.5	Air intake ventilation system optimization	Direct	Small	On-line	3

3.1 Off-line Small Data-Driven Blast Furnace Optimization

Blast furnaces [5] are very complex systems and running experiments with blast furnaces is costly, time-consuming, and very cumbersome due to complex reaction mechanisms. Thus, decision makers can optimize the operating conditions based only on a limited amount of experimental data.

In blast furnace optimization, the decision variables typically are the amount of several components to be added in the furnace, such as limestone and dolomite, quartzite, manganese, alkali and alumina additives. In total, more than 100 components can be added in the furnace, making optimization and surrogate modelling very challenging. To reduce the number of decision variables, dimension reduction techniques can be adopted by analyzing the influence of decision variables on the objectives to be optimized. The objective functions in blast furnace optimization may include the required properties of the product, and objectives related to the environmental and economic requirements as well.

In [10], an off-line data-driven multi-objective evolutionary algorithm was reported, where 210 data points are available collected by means of real-time experiments in the furnace. The first important challenge after collecting the data is to formulate the optimization problem, i.e., to identify objective functions and decision variables. After several rounds of discussions with the expert involved, eight objectives were identified. Principle component analysis is employed to reduce the number of decision variables and eventually 12 most important decision variables were retained. The objectives and decision variables used in the optimization are presented in Tables 2 and 3, respectively.

Table 2. Objectives of the blast furnace optimization problem

No.	Objective	Task
1	Tuyere cooling heat loss (GJ/hr)	Minimize
2	Total BF gas flow (Nm3/hr)	Maximize
3	Tuyere velocity (m/s)	Maximize
4	Heat loss (GJ/hr)	Minimize
5	Corrected productivity (WV) (t/m3/day)	Maximize
6	Coke rate (Dry) (kg/tHM)	Minimize
7	Plate cooling heat loss (GJ/hr)	Minimize
8	Carbonrate (kg/tHM)	Minimize

Table 3. Decision variables of the blast furnace optimization problem

No.	Decision variable
1	Pellet (%)
2	Sp.Flux consumption (kg/tHM)
3	Limestone (kg/tHM)
4	Dolomite (kg/tHM)
5	LD slag (kg/tHM)
6	Quartzite (kg/tHM)
7	Mn (%)
8	Alkali - additives (kg/tHM)
9	Alumina - additives (kg/tHM)
10	FeO ore (%)
11	SiO2(%)
12	CaO (%)

As can be seen from Table 2, several economical objectives that influence the efficiency of the furnace are also considered. They include minimizing the heat loss, maximizing the gas flow and maximizing the tuyere velocity. After identifying the objective functions and decision variables, the next challenge is to optimize these objectives to obtain optimal process conditions. As mentioned, since no analytical or simulation models are available, surrogates were built for each objective function. Kriging models [25] have been widely used in the literature [21, 24] due to their ability to provide a good approximation from a small amount of data, as well as a degree of uncertainty for the approximated values. Therefore, Kriging model was chosen as the surrogate to assist the optimization algorithm.

The data available from the blast furnace is typically noisy and contains outliers. Therefore, pre-processing of the data was needed before building the Kriging models. In [10], a local regression smoothing technique [16] was used to smoothen the fitness landscape. In local regression smoothing, every sample in the data available is assigned with weights and a locally weighted linear regression is used to smoothen the data.

After smoothening the data, a Kriging model was built for each objective function. The next challenge was then to select an appropriate algorithm to optimize eight objectives simultaneously. For this purpose, RVEA [9] was

adopted to optimize the objective functions. RVEA was shown to be competitive on several benchmark problems compared to several EAs. RVEA differs from other many-objective evolutionary algorithms in the selection criterion and a set of adaptive reference vectors for guiding the search. The selection criterion, called angle penalized distance (APD), aims to strike a balance between convergence and diversity. The set of adaptive reference vectors makes sure that a set of evenly distributed solutions can be obtained in the objective space even for problems with different scales of objectives.

In [10], 156 reference vectors were generated and 10000 function evaluations using the Kriging models were performed. A representative set of 100 non-dominated solutions in the objective space is presented in Fig. 5. These

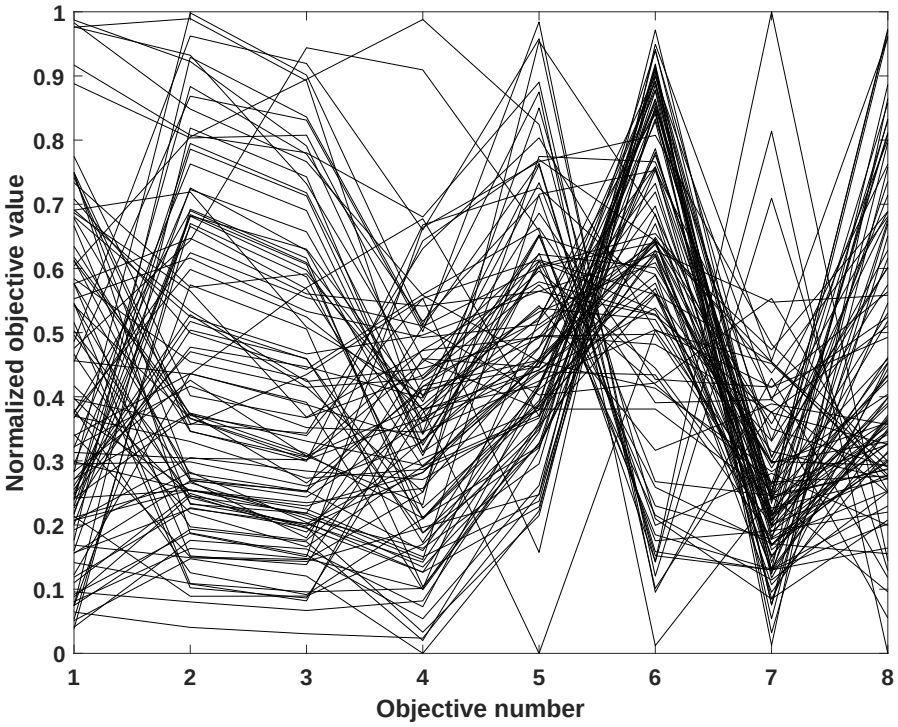

Fig. 5. A representative set of 100 non-dominated solutions in the objective space by using RVEA assisted by Kriging models.

solutions are presented on a normalized scale to maintain the confidentiality of the data. The results clearly show a conflicting nature between the coke rate (the 6th objective in Table II) and productivity (the 5th objective). Moreover, our results show that for many solutions a conflicting nature exists between the productivity (the 5th objective) and gas velocity (the 3rd objective). These solutions were presented to experts and considered to be satisfactory and reasonable, although they remain to be verified in practice.

3.2 Off-line Big Data-Driven Trauma System Design Optimization

The design of trauma systems can be formulated as a combinatorial multi-objective optimization problem to achieve a clinically and economically optimal configuration for trauma centers. In [100], three different clinical capability levels for different injury degrees, i.e., major trauma center (MTC), trauma unit (TU), and local emergency hospital (LEH), were assigned to 18 existing Scottish trauma centers [42]. Designing such a trauma system should be in principle based on the geospatial information, which is hard to measure accurately. However, geospatial information relevant to trauma system design can be implicitly reflected by the incidents occurred during a period of time. Thus, trauma system design based on a large number of incident records can be seen as an off-line data-driven optimization problem.

In evaluating a candidate configuration, all recorded incidents are re-allocated to centers matching their injuries using an allocation algorithm, which is a decision tree to provide all injured persons with matched clinical services and timely transportation to the hospital based on the degree of injuries and the location of the incidents [39]. After

allocating all injured persons to an appropriate hospital by land or air, the allocation algorithm can evaluate the following four metrics.

- Total travel time: the travel time of sending all the patients from the incident locations to the allocated centers is summarized, which is a clinical metric.
- Number of MTC exceptions: some patients with very severe injuries might have to be sent to the nearest TU instead of an MTC, because the nearest MTC in the configuration is too far away. Such cases are denoted as MTC exceptions, which is a metric to assess the clinical performance of the configuration.
- Number of helicopter transfers: some patients must be sent by air due to a large distance from the incident location to the hospital to be sent to. The number of helicopter transfers is an economical metric.
- MTC volume: the number of patients sent to each MTC in the configuration shows its obtained clinical experience.

In [41], the first two metrics (total travel time and number of MTC exceptions) were set as objectives (f_1 and f_2) and the other two (number of helicopter transfers and MTC volume) as constraints. Moreover, the distance between any two TUs in the configuration is constrained, which is not based on the metrics of the simulation. Given the formulation, the trauma system design problem was solved by NSGA-II [18] in [41], where 40,000 incidents (ambulance service patients with their locations and injuries) in one year served as the data for optimization.

Note that evaluating each configuration needs to calculate the objectives and constraints using all data, which makes the function evaluations expensive. For example, it took over 24 hours for NSGA-II to obtain satisfactory results [41]. To reduce such high computational costs, a multi-fidelity surrogate management strategy was proposed to be embedded in NSGA-II in [100].

As the incidents are distributed with a high degree of spatial correlation [40], the data can be approximated by a number of data clusters, which is usually much smaller than the number of data. In this case, it is not necessary to use all data records for function evaluations, and fitness calculations based on the clustered data can be seen as surrogate models approximating the function evaluations [100]. It is conceivable that the approximation error decreases as the number of clusters K increases, but the computational cost increases as well. The multi-fidelity surrogate management strategy [100] tuned the number of clusters as the optimization proceeded according to the allowed root mean square error (RMSE) of the surrogate model on f_1. It is well known that the selection in NSGA-II is based on the non-dominated sorting [105, 95], where the population combining the parent and offspring solutions is sorted into several non-dominated fronts and the better half of the individuals in the combined population is selected as the parent population for the next generation. Thus, the allowed maximum approximation error should not lead to the consequence that solutions in the first front are ranked after the last selected front due to approximation errors. Therefore, the allowed maximum error ER^* was defined as follows:

$$ER^* = \frac{1}{2}\min\{f_1^k - f_1^j\}, 1 \le k \le |F_l|, 1 \le j \le |F_1|, \tag{1}$$

where F_1 is the solution set of the first front and F_l is the solution set of the last selected front. As the evolutionary search of NSGA-II proceeds, the population gets concentrated and moves towards the true Pareto front (PF), and the allowed error ER^* decreases as the number of clusters increases.

Fitness evaluations using the entire data were replaced by surrogate models based on K-clusters of data in NSGA-II. In each generation, the non-dominated solutions were evaluated by the whole data simulation to estimate the error ER of the surrogate model based on K-clustered data. Thus, the relationship between the surrogate error and K was estimated according to the following regression model (K, ER):

$$ER = \frac{1}{\beta_1 + \beta_2 K}. \tag{2}$$

Given the regression parameters β_1 and β_2 and the allowed error ER^*, the adjusted number of clusters K^* can be calculated from Equation (2) as shown in Fig. 6.

By embedding the multi-fidelity surrogate management strategy in NSGA-II [100], we describe the algorithm (called SA-NSGA-II) as follows.

1. **Initialization**
   - Set K to be 18 (the number of hospitals in the system). Cluster the data into K categories.
   - Generate a random initial population and evaluate the population using the surrogate based on K-clustered data.

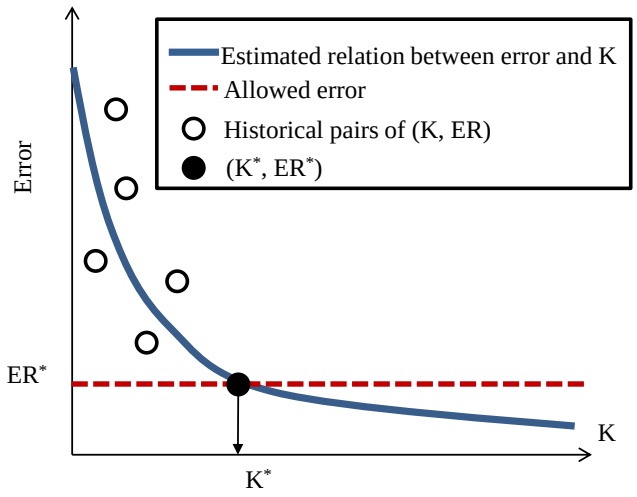

Fig. 6. Illustration of the multi-fidelity surrogate management strategy in one generation of NSGA-II, where the solid line denotes the estimated relationship between the approximation error on f_1 and K from historical (K, ER) pairs denoted by circles, the dotted line is the allowed error ER^* defined in Equation (1), and the dot is the estimated new number of clusters K^*.

2. **Reproduction:** Apply 3-point crossover (probability of 1) and point mutation (probability of 0.2) to the parent population for the offspring population, evaluate the offspring population using the surrogate based on K-clustered data.
3. **Selection:** Combine the parent and offspring populations, select the parent population based on non-dominated sorting and crowding distance.
4. **Fidelity adjustment**
   - Detect the improvement of the non-dominated solution set. Apply the following steps to adjust K if there is no improvement; otherwise, keep K unchanged.
   - Calculate the fitness of the non-dominated solutions using the whole data. Estimate the approximation error ER of the surrogate based on K-clustered data, and record the estimated pair (K, ER).
   - Estimate the relationship between ER and K by the regression model in Equation (2) from those estimated pairs (K, ER).
   - Calculate the allowed error ER^* as Equation (1). If ER^* is smaller than half of ER, set $ER^* = ER/2$.
   - Estimate the new K^* by ER^* based on the obtained regression model if there are enough historical pairs to obtain the regression model, otherwise $K^* = 2K$. If K^* exceeds the limit K_{max}, set $K^* = K_{max}$.
   - Re-cluster the data into K^* categories.
   - Evaluate the parent population using the surrogate based on K^*-clustered data.
5. **Stopping criterion:** If the stopping criterion is satisfied, output the non-dominated solutions, otherwise go to step 2).

SA-NSGA-II is an off-line data-driven EA as no new data can be actively generated during the optimization. Experimental results have shown that SA-NSGA-II can save up to 90 percent of the computation time of NSGA-II [100]. Although the lack of on-line data limits the performance of off-line data-driven EAs, making full use of the off-line data can effectively benefit the optimization process as well. Therefore, handling the off-line data affects the optimization process. We compare the performance of NSGA-II run for 100 generations with its variants using different data handling strategies on the trauma system design problem. The three compared strategies are described below.

- **Random sampling:** Before running NSGA-II, K data points are randomly selected from the whole data for function evaluations.
- **Clustering:** Before running NSGA-II, the whole data is divided into K clusters, which is fixed during the optimization.
- **Adaptive clustering:** SA-NSGA-II is used for the comparison, where the data is adaptively clustered for function evaluations in optimization.

As K_{max} in SA-NSGA-II is set to 2000 [100], we assume that K ranges from 100 to 2000. All the compared algorithms run independently for 20 times. IGD [114], the average distance from a reference PF set to the obtained solution set, is used to assess the performance of compared algorithms. The same settings as in [100] are used, where the reference PF set is obtained from the non-dominated set of 5 runs of NSGA-II based on the whole data. The average IGD values of three compared strategies (random sampling, clustering, and adaptive clustering) over various K values are shown in Fig. 7.

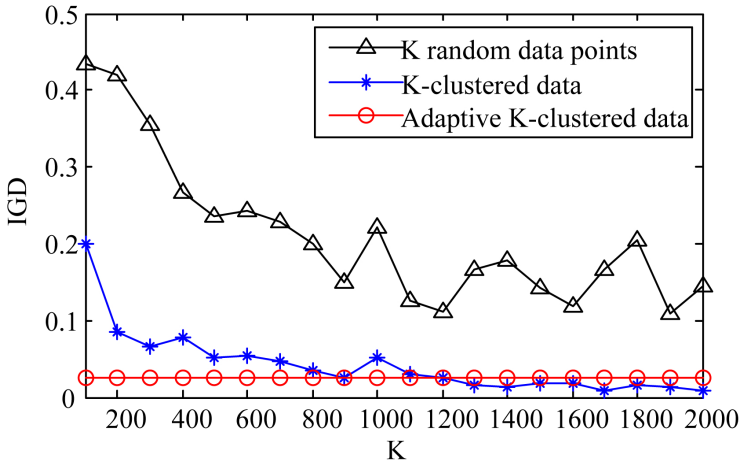

Fig. 7. Average IGD values of three compared strategies (random sampling, clustering, and adaptive clustering) over different K.

From Fig. 7, we can see that for the two variants using random sampling and clustering, the IGD values decrease with an increasing K, because the more data points are used in function evaluations, the more accurate the fitness calculations are. In fact, randomly sampling K data points for the function evaluations fails to extract the data pattern, while a relatively small number of representative data points are still able to describe the main feature of the whole data. Therefore, IGD values resulting from the random sampling strategy are larger (worse) than those from the clustering strategy for various sizes of K. Although a large K leads to better performance, the computational cost becomes higher. From the results in Fig. 7, we can see that the adaptive clustering strategy uses various Ks (up to 2000) during the optimization results in a similar IGD value obtained by using 2000-clustered data, although the former strategy requires much less computational resources than the latter.

From the above experimental results, we can conclude that a properly designed model management strategy can effectively enhance the computational efficiency of the optimization without a serious degradation of the optimization performance.

3.3 Off-line Small Data-Driven Optimization of Fused Magnesium Furnaces

The performance optimization of fused magnesium furnaces aims at increasing the productivity and enhancing the quality of magnesia products while reducing the electricity consumption in terms of optimized set points of electricity consumption for a ton of magnesia (ECT) [58]. Before a production batch, the ECT of every furnace is set by an experienced operator according to the properties of raw materials and the condition of each furnace. Optimizing such a problem should be based on the relationship between ECT set points and each performance index. However, it is very hard, if not impossible, to build analytical functions because of complex physical and chemical processes involved, intermittent material supplies, and sensor failures. As a result, one has to turn to limited and noisy historical production data for optimizing the performance of fused magnesium furnaces, making it an off-line data-driven optimization problem.

Only a small number of noisy data is available because one production batch lasts 10 hours. There are 60 groups of ECT set points and performance indicators for five furnaces, which are all the furnaces connected to one transformer. Therefore, the decision variables are the ECT set points of five furnaces, and the objectives are the average high-quality rate, total output and electricity consumption of five furnaces.

Given a small amount of noisy data, it is hard to construct accurate surrogates. In the GP-assisted NSGA-II [29], termed NSGA-II_GP, two surrogates are built for model management, as shown is Fig. 8. One is a low-order

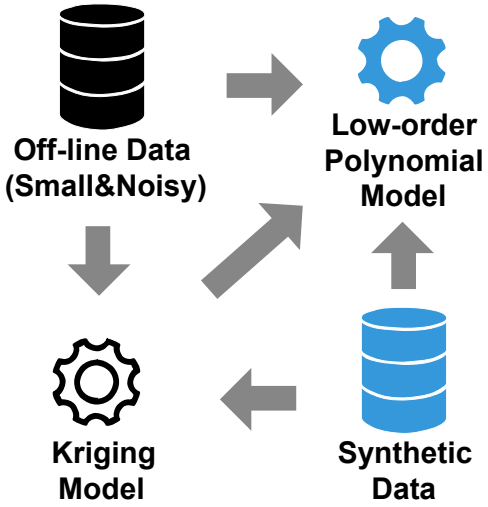

Fig. 8. Model management in off-line data-driven performance optimization of fused magnesium furnaces.

polynomial regression model constructed using the off-line data. This low-order model approximates the unknown real objective function to generate synthetic data for model management, playing the role of the real objective function. The reason for adopting a low-order polynomial model is that it is less vulnerable to over-fitting. The other surrogate is a Kriging model, which is built based on both off-line data and synthetic data. Here, the most promising candidate solutions predicted by the Kriging model are further evaluated using the low-order polynomial model, and the synthetic data generated by the polynomial model are used to update the Kriging model for the next generation. In optimization, expected improvement [72] is adopted to identify the most promising candidate solutions, and k-means clustering is applied in the decision space to choose sampling points, while fuzzy c-means clustering [115] is introduced to limit the number of data for training the Kriging model.

The biggest challenge in the off-line data-driven performance optimization of magnesium furnaces is how to verify the effectiveness of a proposed algorithm due to the lack of real objective functions. To address this issue, the performance of the proposed method was first verified on benchmark problems. During optimization, it is assumed that the real objective function is not available except for a certain amount of data generated before optimization. The resulting optimal solutions are then verified using the real objective functions to assess the effectiveness of the proposed algorithm. Once the algorithm is demonstrated to be effective, it can then be applied to real-world problems. This strategy is illustrated in Fig. 9. To simulate the small amount of noisy data in the

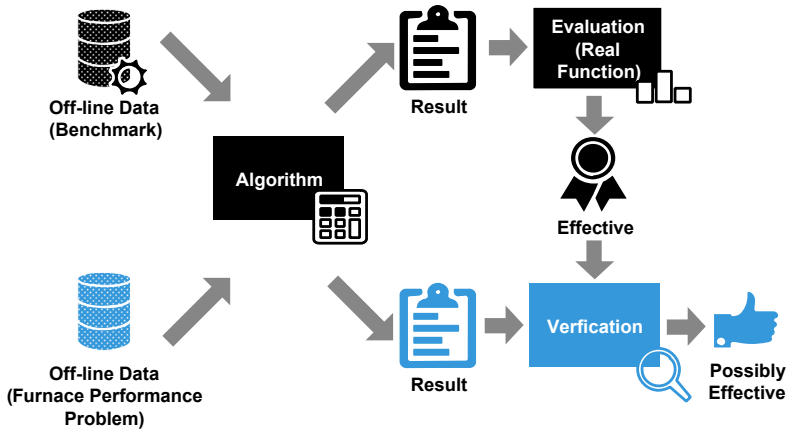

Fig. 9. Illustration of the method to verify the effectiveness of an optimization algorithm in off-line data-driven performance optimization of magnesium furnaces.

furnace performance optimization problem, Latin hypercube sampling (LHS) [87] is first used to generate off-line data using the objective functions of the benchmark problems, to which noise is then added. The noise is generated according to the following equation:

$$noise = (f_{jmax} - f_{jmin}) \times rand, \tag{3}$$

where $rand$ is a random number within [-0.1,0.1], and f_{jmin} and f_{jmax} are the minimum and maximum of real function values of the off-line data in the j-th objective, respectively. In numerical simulations on nine benchmark problems, NSGA-II_GP was compared with the original NSGA-II and a popular surrogate-assisted multi-objective EA, ParEGO [56]. The results on the benchmark problems consistently showed that the performance of NSGA-II_GP is the best.

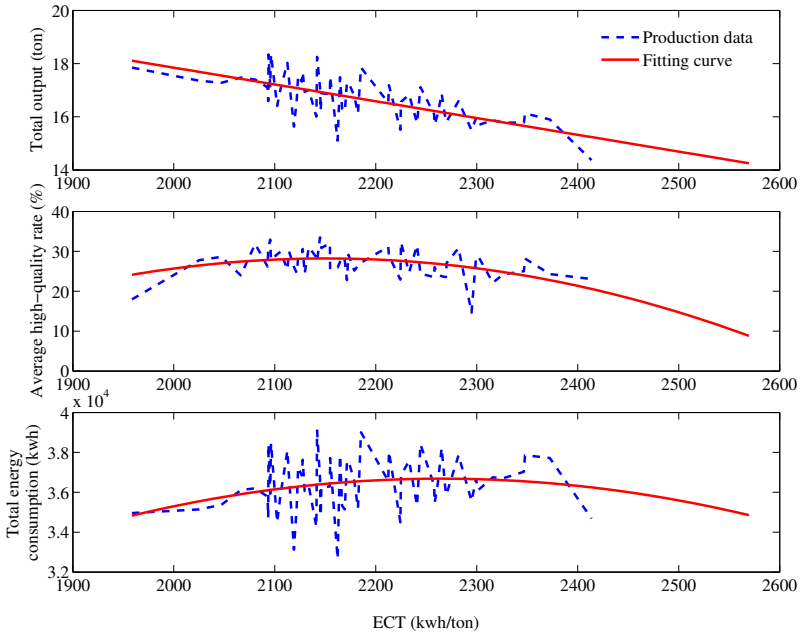

Fig. 10. An example for fitting the production data of one furnace using low-order polynomial models.

In optimizing the furnace performance, first and second order polynomial models are considered to fit the collected production data, and the fitting results of one furnace are plotted in Fig. 10. After the formulation of the furnace performance optimization problem, NSGA-II_GP is applied and the optimization results are plotted in Fig. 11, which shows that NSGA-II_GP has found better ECT set points compared to the off-line data. From the results on benchmark problems and furnaces performance optimization problem, we can conclude that different accuracy surrogates are very helpful to off-line small data-driven optimization.

3.4 On-line Small Data-Driven Optimization of Airfoil Design

Airfoil design is one important component in aerodynamic applications, which changes the airfoil geometry to achieve the minimum drag over lift ratio. However, the evaluation of airfoil geometry is based on time-consuming CFD simulations, therefore only a small number of expensive evaluations is allowed during the design process, resulting in an on-line data-driven optimization problem.

The geometry of an airfoil is represented by a B-spline curve consisting of 14 control points [102]. Therefore, the decision variables are the positions of those 14 control points. The objective is to minimize the average drag over lift ratio in two design conditions, where the drag and lift are measured based on the results from CFD simulations.

In this specific design of RAE2822 airfoil, there are 70 off-line data points describing the relationship between different geometries and their evaluated objective value. In addition, 84 new samples are allowed to be generated during the optimization. The recently proposed on-line data-driven EA, committee-based active learning based surrogate-assisted particle swarm optimization (CAL-SAPSO) was employed to solve the airfoil design problem in [102].

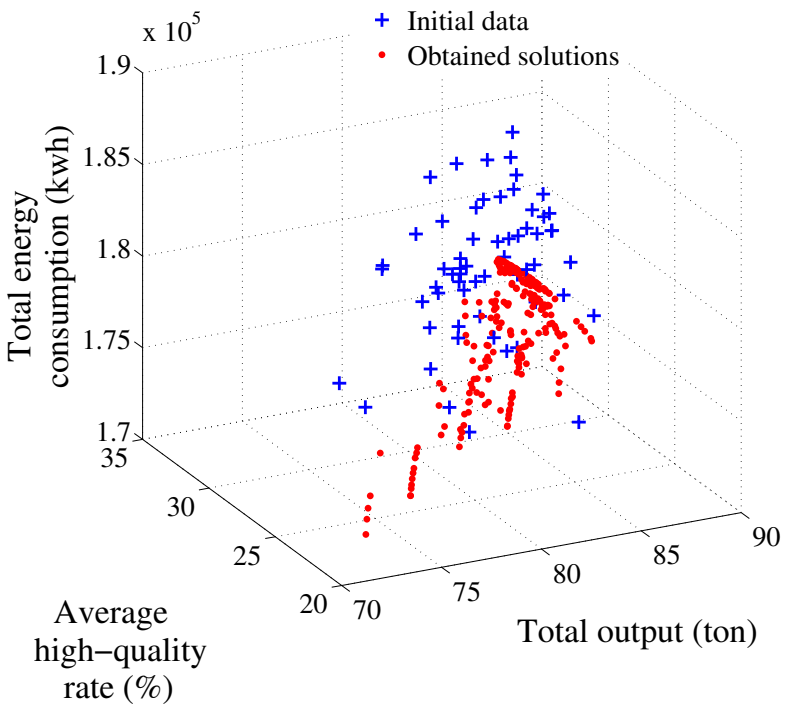

Fig. 11. Optimization results of the furnaces performance problem.

CAL-SAPSO uses two surrogate ensembles composed of a polynomial regression model, an RBFN, and a Kriging model [27] to approximate the expensive objective. One ensemble serves as a global model built from the whole data, while the other is meant to be a local model built from the data belonging to the best 10% objective values found so far. CAL-SAPSO begins with search on the global model, and then switches to the local model when no improvement can be achieved. The found best solutions are always evaluated using the real objective function and both surrogate models are then updated. The two models are used and updated in turn until the allowed maximum number of fitness evaluations is exhausted.

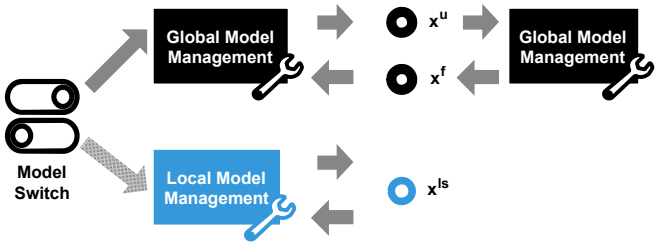

Fig. 12. Model management strategy in CAL-SAPSO.

The model management strategy in CAL-SAPSO is individual-based, as shown in Fig. 12. Three types of candidate solutions are to be re-evaluated using the real objective function to update the global and local models. A canonical PSO algorithm [83] using a population size of 100 is run for a maximum of 100 iterations. As the model management strategy of CAL-SAPSO is based on query by committee (QBC) [81], the uncertainty is measured by the largest disagreement among the ensemble members. For the global model, the most uncertain solution $\mathbf{x}^u$ is searched for at first using PSO based on the following objective function:

$$\mathbf{x}^u = \arg\max_{\mathbf{x}}(\max(\hat{f}_i(\mathbf{x}) - \hat{f}_j(\mathbf{x}))), \tag{4}$$

where $\hat{f}_i$ and $\hat{f}_j$ $(1 \leq i, j \leq 3)$ is the i-th and j-th models in the surrogate ensemble. After $\mathbf{x}^u$ is evaluated using CFD simulations and added to on-line data, the global surrogate ensemble is updated. Then, PSO is used to search

for the optimum $\mathbf{x}^f$ of the global model as:

$$\mathbf{x}^f = \arg\min_{\mathbf{x}} \hat{f}_{ens}(\mathbf{x}), \tag{5}$$

where $\hat{f}_{ens}(\mathbf{x})$ is the global surrogate ensemble. After $\mathbf{x}^f$ is evaluated using CFD simulations and added to on-line data, the global surrogate ensemble is updated again. If $\mathbf{x}^f$ is not the better than the best solution found so far, CAL-SAPSO switches to the local surrogate ensemble to continue the search. For the local model, only the optimum $\mathbf{x}^{ls}$ of the local model is chosen to be re-evaluated using the real objective function, which is searched using PSO based on

$$\mathbf{x}^{ls} = \arg\min_{\mathbf{x}} \hat{f}^l_{ens}(\mathbf{x}), \tag{6}$$

where $\hat{f}^l_{ens}(\mathbf{x})$ is the local surrogate ensemble. After $\mathbf{x}^{ls}$ is evaluated with the CFD simulations and added to on-line data, the local surrogate ensemble is updated. If $\mathbf{x}^{ls}$ is not better than the best geometry found so far, CAL-SAPSO switches to the global surrogate ensemble to continue the search.

CAL-SAPSO was run on the airfoil design problem for 20 times. The best geometry obtained is shown in Fig. 13, where the objective values are normalized with the objective value of the baseline design. We can see that the solution found by CAL-SAPSO achieved a 35% improvement of the drag over lift ratio over the baseline design using 70 off-line CFD simulations before optimization and 84 ones during the optimization (a total of 154 CFD simulations), which is promising in the application of aerodynamic engineering.

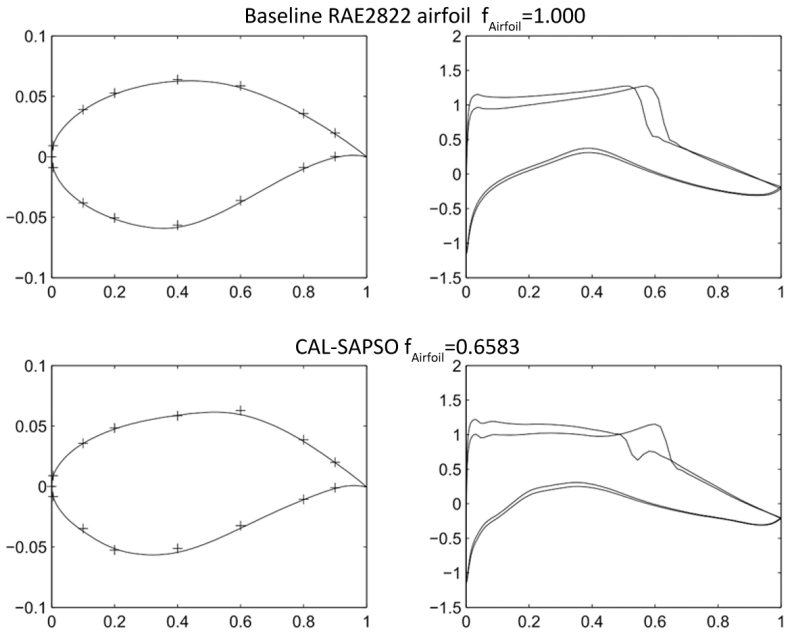

Fig. 13. The baseline design and the best design (geometries and pressure distribution) obtained by CAL-SAPSO.

3.5 On-line Small Data-Driven Optimization of An Air Intake Ventilation System

An air intake ventilation system of an agricultural tractor was considered in [14] for maintaining a uniform temperature inside the cabin and defrost the windscreen. The particular component of interest consist of four outlets and a three-dimensional CATIA model of the component is shown in Fig. 14. To maintain a uniform temperature distribution, the flow rates from all the outlets should be the same. However, these outlets had different diameters and maintaining the same flow rate from all the outlets is not trivial. In addition, the pressure loss should be minimized to increase the energy efficiency of the system. Thus, the optimization problem involves computationally expensive CFD simulations. Before starting the solution process, an initial design used in the ventilation system was provided by the decision maker and a CFD simulation of this initial design is shown in Fig. 15.

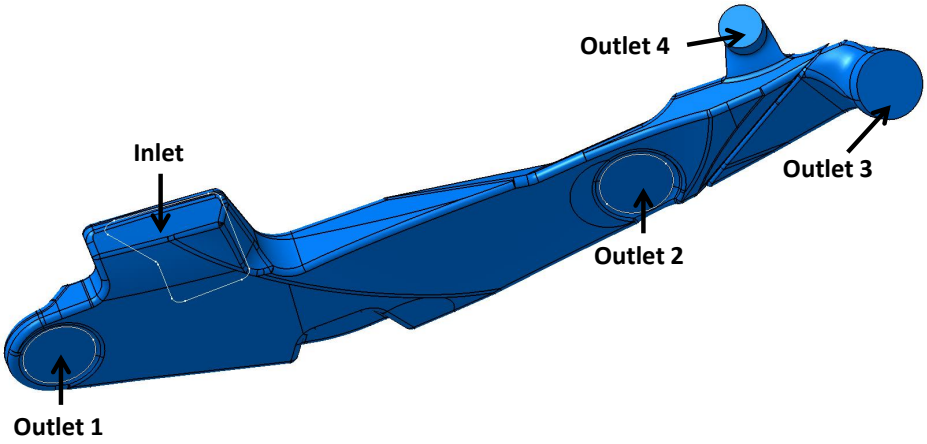

Fig. 14. A three dimensional CATIA model of the component in the air intake ventilation system.

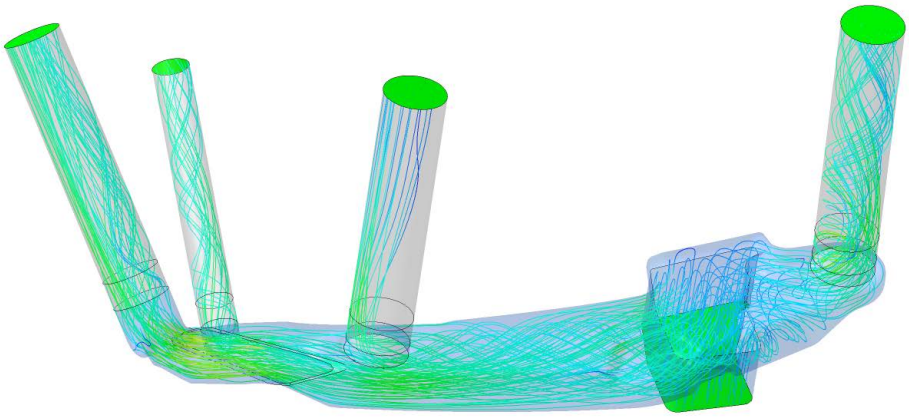

Fig. 15. A CFD simulation of the initial design.

From Fig. 14, we can see that outlet 4 has the smallest diameter compared to the other outlets. Therefore, it is very difficult to make the flow rate from outlet 4 to be equal to those from other outlets. To address this issue, special attention was paid to the flow rate from this outlet. Based on several discussions with an aerodynamic expert, a three-objective optimization problem was finally formulated as follows:

f_1 : Minimize variance between flow rates at outlets 1 to 3
: Minimize $var(Q_{1,3})$
f_2 : Minimize pressure loss of the air intake
: Minimize $P_{inlet} - P_{outlet}$
f_3 : Minimize the difference between the flow rate at outlet 4 and the average of the flow rates at outlets 1 to 3
: Minimize $avg(Q_{1,3}) - Q_4$,

where Q_k represents the flow rate from the k^{th} outlet, $avg(Q_{1,3})$ the average flow rate values from outlets 1-3 and P_{inlet}, and P_{outlet} are the pressure values at the inlet and the outlet, respectively. Note that P_{outlet} is the same among all outlets and equal to the atmospheric pressure.

The third objective makes sure that the flow rate from outlet 4, which has the smallest diameter, can have the same flow rate to the average of flow rates from other outlets. As mentioned, the diameters play a vital role in maintaining a uniform flow rate, therefore the scaling factors of the initial design diameters are used as the decision variables:

$$x_i = \frac{D_i}{D_i^{(initial)}} \quad \text{for} \quad i = 1, \ldots, 4, \tag{7}$$

where D_i is the diameter of the i^{th} outlet and $D_i^{initial}$ is the diameter of the i^{th} outlet in the initial design. The lower (x_i^{lb}) and the upper (x_i^{ub}) bounds of the decision variables are as follows:

$$\begin{aligned} x_i^{lb} &= 0.5 \;\text{ for }\; i = 1, \ldots, 4, \\ x_i^{ub} &= 1.5 \;\text{ for }\; i = 1, \ldots, 4. \end{aligned} \tag{8}$$

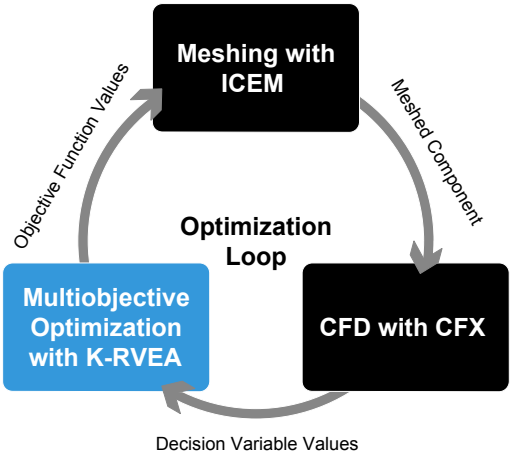

Fig. 16. The optimization loop for multi-objective shape optimization of an air intake ventilation system

Once the multi-objective optimization problem was formulated, the next step was to combine different simulation tools to obtain the objective function values as shown in Fig. 16. ANSYS ICEM [1] was used for meshing the component first and ANSYS CFX [96] for performing CFD simulations afterwards. To ease the solution process, the outlets of the component were prolongated, as shown in Fig. 15.

For optimization, a Kriging-assisted evolutionary algorithm for optimization problems with at least three objectives called K-RVEA [11] was used. The algorithm uses elements from its underlying RVEA [9] for efficiently managing the surrogates. The samples in K-RVEA are selected to strike a balance between convergence and diversity. Another feature of K-RVEA is that a limit on the size of training samples is imposed to reduce the computation time. A flowchart of the algorithm is shown in Fig. 17, where an archive $A1$ is used to store the samples for training and another archive $A2$ for storing all the evaluated samples.

In the algorithm, a number of initial candidate designs are generated using Latin hypercube sampling, which are evaluated with CFD simulations. The evaluated candidate solutions are added to the archives $A1$ and $A2$. Kriging models are built for each objective function by using the samples in $A1$. After running RVEA with the Kriging models for a prefixed number of iterations, samples are selected to update the Kriging models. These samples are selected based on the needs of convergence and diversity which are identified using the reference vectors. Every time the surrogates are updated, the change in the number of empty reference vectors compared to that in the previous update is measured. If the change is less than a predefined parameter, convergence is prioritized. Otherwise, diversity is used as the criterion in selecting candidate solutions to be evaluated using CFD simulations. A fixed number of evenly distributed samples is selected based either on their angle penalized distance, which is the selection criterion in RVEA, or on uncertainty values from the Kriging models.

We used a maximum of 200 expensive function evaluations (CFD simulations) in K-RVEA. Forty non-dominated solutions were generated, which are shown in Fig. 18 in the objective space. The values of the objective functions are normalized to maintain the data confidentiality. These solutions were presented to an aerodynamic expert and a final solution was selected based on his preferences. The final solution and the solution corresponding to the initial design are also shown in Fig. 18. The final selected design has an equal pressure loss (the second objective) but significant improvements in the first and the third objectives (related to minimization of differences between the flow rates) compared to the initial design. A good balance in flow rates means more can be passed into the cabin without any extra consumption of energy.

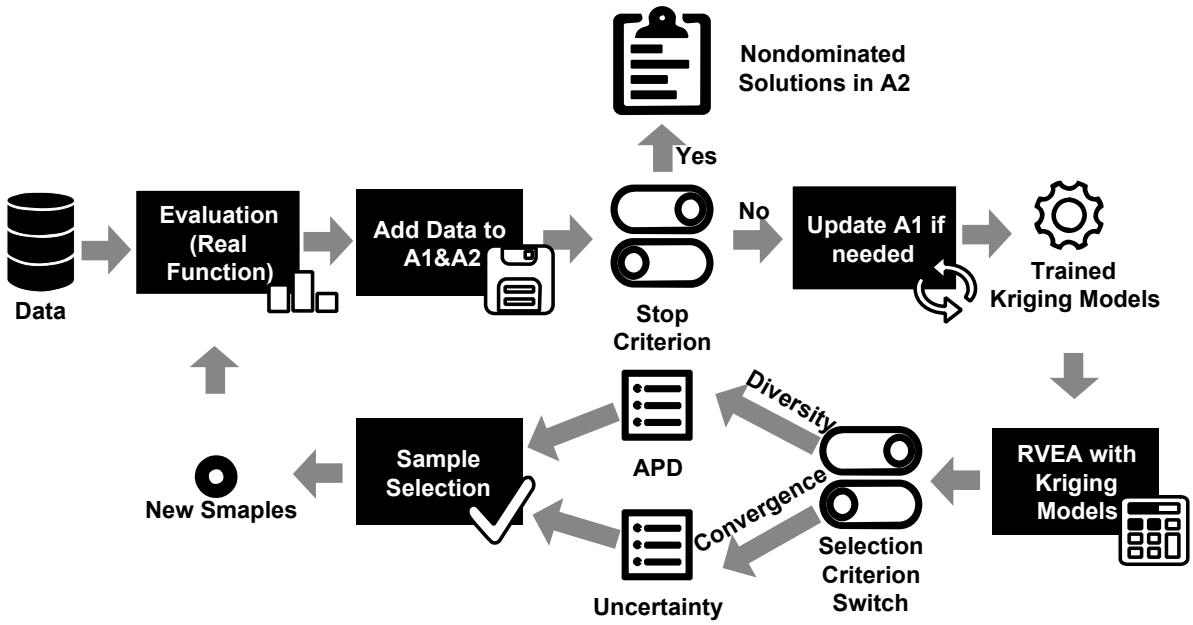

Fig. 17. The flowchart representing the main steps in K-RVEA

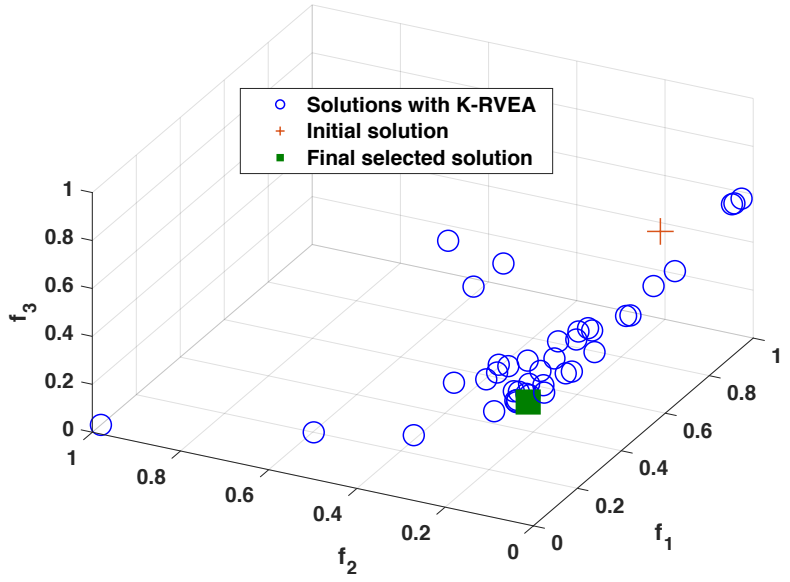

Fig. 18. Non-dominated solutions in the objective (normalized) for the air intake ventilation system

4 Challenges and Promises

4.1 On-line Data-Driven Optimization

In on-line data-driven optimization, the main goals are to enhance the accuracy of the surrogate models and balance the convergence and diversity. Thus, model management is critical in on-line data-driven EAs. In an ideal scenario, any EA can be used in on-line data-driven optimization. However, in reality, the EA and the surrogates should be integrated seamlessly to ensure the success of the surrogate-assisted optimization algorithm. In the following, we highlight a few major challenges in on-line data-driven optimization.

Selection of surrogate models: When developing an on-line data-driven EA, the first challenge is to select an appropriate surrogate model. Several surrogate models, e.g., Kriging, ANN, RBFN, and support vector regression

can be used and there is very little theoretical guidance in the literature for choosing the surrogate model. In many cases, a surrogate model is selected based on the experience of the user (e.g., an engineer). For instance, RBFN was used in [59] to solve an optimization problem of coastal aquifer management because of its popularity for groundwater applications. Generally speaking, however, stochastic models such as Kriging models may be preferred if an infill criterion is to be used for model management. As discussed in [30], the main limitation of Kriging models is their possibly large computational complexity when a large number of training samples is involved. In this case, ensembles are good alternatives to Kriging models due to their scalable computational complexity.

Using surrogate models: Once surrogate models are selected, the next question is how to use them in the EA. For instance, approximating objective functions [12], classifying samples according to their fitness [74], predicting ranks [66], or hypervolume [78] or approximating a scalarizing function by converting a multi-objective optimization problem to a single-objective problem [56, 93] and approximating the PF [35] are possible ways of using a surrogate model.

Selection of training data: How to select the training data is another challenge. In on-line data-driven optimization, surrogates need to be continuously updated to enhance their accuracy and to improve the exploration of the EA as well. Samples for training should be selected in such a way that both convergence and diversity are taken into account. Most on-line data-driven EAs start with generating a number of candidate solutions using a design of experiment technique, e.g., LHS [87]. Afterwards, a model management strategy, including popular infill criteria such as expected improvement [53] and many other generation- or individual-based model management strategies [45] can be used for selecting candidate solutions to be re-evaluated using the real objective functions and then re-train or update the surrogates. All sampling techniques and model management strategies have advantages and limitations and could be tailored to a particular class of problems as well as the EA used.

Size of training data: Another important challenge, which is usually overlooked in many on-line optimization algorithms is the size of the training data. For example, using a large number of samples may dramatically increase the computational complexity, in particular when the Kriging model is employed as the surrogate. Therefore, one should pay attention to using an appropriate size of data for training in on-line data-driven optimization.

Selection of EA: As mentioned, a model management strategy needs to be employed to select candidate solutions for evaluation using the real objective functions and for training the surrogates. In many on-line optimization algorithms in the literature, not much attention has been paid towards the selection of EAs. This can be attributed to the assumption that a good approximation or near-optimal solution can be obtained by any EA. However, in reality, different EAs have different advantages and limitations and they should be used based on the properties of the problem to be solved. For instance, using a dominance-based EA for problems with more than three objectives may not be a viable choice.

Handling objectives with different latencies: In many real-world multi-objective optimization problems, objectives may have different computation times of different objectives. For instance, in [70, 33], a decision variable is also used as an objective function and the computation time for evaluating such objective functions is negligible compared to other simulation-based objective functions. Some existing and recent studies can be applied to expensive MOPs with different latencies among objective functions. For instance, in [60], a transfer learning was used to build surrogate models among correlated objectives. In an extended work in [67], the authors used transfer learning for sharing information between different parts of the Pareto front. However, they considered the objectives with the same computation time.

A recent work on this topic has been proposed [15] for bi-objective optimization problems, where an algorithm called HK-RVEA was applied to solve problems with objectives of different computation time. Adapting on-line data-driven EAs for handling different latencies among objective functions is one of the important challenges in on-line data-driven optimization.

Termination criterion and performance metric: Last but not the least, a proper stopping or termination criterion and a measure for the performance of the algorithm are also very important when using an on-line optimization algorithm. Where to stop is very important especially for problems with expensive evaluations. For instance, running an algorithm if there is no improvement in the quality of solutions may lead to waste of resources. In the literature, typically conventional performance metrics, e.g., IGD or hypervolume are used to measure the performance of on-line data-driven EAs. These metrics are influenced by several parameters such as the size of the reference set in calculating IGD and may not provide a precise measurement. The effect of parameters on performance metrics has been analyzed in details in [37, 38]. For both performance measure and termination criterion, one should also consider the performance of the surrogate model including the accuracy and uncertainty.

In addition to challenges mentioned above, several other challenges exist related to the characteristics of the problem to be solved. These are dimensions in the objective and decision spaces, handling constraints, and mixed-integer or combinatorial optimization problems. Some on-line optimization algorithms, e.g., K-RVEA [11, 13], CSEA

[74], and SL-PSO [88] have been proposed to tackle these challenges. However, many real-world on-line data-driven problems are constrained [54, 85, 75, 76, 34, 48] and / or of mixed-integer decision variables [68, 71, 121, 112, 99, 73, 2]. Currently, many issues of data-driven EAs for constrained and mixed-integer problems remain open and deserve more attention.

Despite of several challenges, on-line data-driven EAs have the potential of solving optimization problems with different characteristics. The wide applicability of on-line data-driven EAs has demonstrated that on-line data-driven surrogate-assisted evolutionary optimization is of paramount practical importance. Some key promising directions in developing on-line data-driven EAs include 1) using ensemble of surrogate models [51, 102, 85, 30], 2) enhancing the convergence by using a combination of local and global surrogate models [61, 98, 119, 88, 111], 3) decreasing the computational complexity of the problem to be considered (or problem approximation) by using multi-fidelity models [44, 62, 26, 103], 4) using fitness inheritance [86], fitness imitation [55] and fitness estimation [91, 90], and 5) using advanced machine learning techniques such as semi-supervised learning [92, 89], active learning [90, 102] and transfer learning [19, 69, 32].

4.2 Off-line Data-Driven Optimization

Unlike on-line data-driven optimization, no new data can be made available for updating surrogate models during off-line data-driven optimization or for validating the found optimal solutions before they are eventually implemented. Therefore, the main challenges in off-line data-driven optimization may come from the following three aspects.

Lack of data during optimization: One serious challenge is the unavailability of new data during the optimization. Without creating new data for model management during the optimization process, the search ability of off-line data-driven EAs can be limited since surrogate models are built barely based on the data generated off-line. How to effectively use the given data heavily affects the performance of an off-line data-driven EA. As far as we know, several advanced machine learning techniques can be employed to alleviate the limitation. For example, semi-supervised learning [92] can enrich the off-line labelled data by using unlabeled data for training. Data mining techniques [100] can be used to extract patterns from the off-line data to guide the optimization process. In addition, ensemble learning [104] can repeatedly use the training data to enhance the search performance in offline data-driven optimization. Furthermore, transfer optimization techniques [32] including sequential transfer optimization, multi-task optimization, and multi-form optimization are able to reuse knowledge from other similar problems. While sequential transfer optimization learns from historical problems, multi-task optimization [31, 19, 69] simultaneously solves multiple similar problems. Finally, multi-form optimization employs multiple formulations (including multiple fidelity levels of the evaluations [100, 103]) of the original problem to share useful information.

Model reliability: In off-line data-driven optimization, no new data is available to assess the quality of surrogate models, making it very challenging to ensure the reliability. Consequently, the optimization process can be very likely misled. To enhance the reliability of the surrogate models based on off-line data only, multiple heterogeneous or homogeneous surrogate models [30, 3] can be adopted using ensemble learning [116]. Furthermore, cross validation [57, 97] is helpful in accuracy estimation and model selection.

Performance verification: The most challenging issue in real-world off-line data-driven optimization is the verification of the solutions found by the optimization algorithm before they are implemented due to the lack of true optimum. In [29], the proposed algorithm is indirectly verified using benchmark problems. Such a verification method is based on an assumption that benchmark problems are similar to the real-world optimization problem to a certain degree. Unfortunately, often little a priori knowledge about real-world optimization problems is available, making it hard to choose the right benchmark problems to reliably test the performance of the algorithm on the real-world problems.

5 Conclusions

The importance of data-driven surrogate-assisted evolutionary optimization cannot be overestimated for EAs to be applied to solve a large class of real-world problems in which no analytical objective functions are available. Unfortunately, this line of research has so far attracted less attention in the evolutionary computation community than it should have due to the following reasons. First, there is a gap between the demands from the industry and the research interests in the academia. Second, there is a lack of dedicated benchmark problems for data-driven optimization that can be made available to researchers and practitioners with few exceptions [103]. Finally, new data-driven surrogate-assisted optimization algorithms are often required to be validated on real-world expensive problems, making it hard for most researchers to perform research in this area due to the lack of access to real-world problems and lack of computational resources.

This paper aims to promote research interests in the evolutionary computation community and attract more attention to data-driven evolutionary optimization, simply because data-driven optimization is indispensable for applying EAs to complex real-world problems. Meanwhile, data-driven surrogate-assisted evolutionary optimization provides a unique platform for creating synergies between machine learning, evolutionary computation and data science, potentially leading to the emergence of a new interdisciplinary area, where many research directions should be considered in the future. Firstly, benchmark problems that are extracted from real-world data-driven optimization applications are highly in demand. Secondly, most existing SAEAs deal with on-line direct data-driven optimization problems. Thus, effective new algorithms should be developed for other types of data-driven optimization problems, where the techniques of both machine learning and data science can be helpful. Finally, data-driven EAs for solving real-world optimization problems should be highly encouraged.

6 Acknowledgements

We would like to thank Prof. Nirupam Chakaraborti for his consent to use the blast furnace case study in this article. We also want to thank Valtra Inc. for providing the air intake ventilation optimization problem. Thanks also go to Dr Jan O Jansen for providing us with the data of the trauma system problem and to Dr John Doherty for his support for using CFD simulations in airfoil optimization. Finally, we would like to thank Prof. Tianyou Chai for the magnesium furnace data.

References

1. ANSYS, Inc.: ANSYS ICEM CFD Tutorial Manual (2013)
2. Bartz-Beielstein, T., Zaefferer, M.: Model-based methods for continuous and discrete global optimization. Applied Soft Computing **55**, 154–167 (2017)
3. Bhattacharjee, K.S., Singh, H.K., Ray, T., Branke, J.: Multiple surrogate assisted multiobjective optimization using improved pre-selection. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 4328–4335. IEEE (2016)
4. Binois, M., Ginsbourger, D., Roustant, O.: Quantifying uncertainty on Pareto fronts with Gaussian process conditional simulations. European Journal of Operational Research **243**(2), 386–394 (2015)
5. Biswas, A.K.: Principles of Blast Furnace Ironmaking: Theory and Practice. Cootha Publication House, Australia (1981)
6. Branke, J., Schmidt, C.: Faster convergence by means of fitness estimation. Soft Computing **9**(1), 13–20 (2005)
7. Buche, D., Schraudolph, N.N., Koumoutsakos, P.: Accelerating evolutionary algorithms with Gaussian process fitness function models. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **35**(2), 183–194 (2005)
8. Bull, L.: On model-based evolutionary computation. Soft Computing **3**(2), 76–82 (1999)
9. Cheng, R., Jin, Y., Olhofer, M., Sendhoff, B.: A reference vector guided evolutionary algorithm for many objective optimization. IEEE Transactions on Evolutionary Computation **20**(5), 773–791 (2016)
10. Chugh, T., Chakraborti, N., Sindhya, K., Jin., Y.: A data-driven surrogate-assisted evolutionary algorithm applied to a many-objective blast furnace optimization problem. Materials and Manufacturing Processes **32**, 1172–1178 (2017)
11. Chugh, T., Jin, Y., Miettinen, K., Hakanen, J., Sindhya, K.: A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization. IEEE Transactions on Evolutionary Computation **22**, 129–142 (2018)
12. Chugh, T., Sindhya, K., Hakanen, J., Miettinen, K.: A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms. Soft Computing (2018), to appear
13. Chugh, T., Sindhya, K., Miettinen, K., Hakanen, J., Jin, Y.: On constraint handling in surrogate-assisted evolutionary many-objective optimization. In: et al., J.H. (ed.) Proceedings of the Parallel Problem Solving from Nature-PPSN. pp. 214–224. Springer (2016)
14. Chugh, T., Sindhya, K., Miettinen, K., Jin, Y., Kratky, T., Makkonen, P.: Surrogate-assisted evolutionary multiobjective shape optimization of an air intake ventilation system. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 1541–1548. IEEE (2017)
15. Chugh, T., Allmendinger, R., Ojalehto, V., Miettinen, K.: Surrogate-assisted evolutionary biobjective optimization for objectives with non-uniform latencies. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 609–616. ACM (2018)
16. Cleveland, W., Loader, C.: Smoothing by Local Regression: Principles and Methods, pp. 10–49. Physica-Verlag HD (1996)
17. Dasgupta, D., Michalewicz, Z.: Evolutionary algorithms in engineering applications. Springer Science & Business Media (2013)
18. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (2002)

19. Ding, J., Yang, C., Jin, Y., Chai, T.: Generalized multi-tasking for evolutionary optimization of expensive problems. IEEE Transactions on Evolutionary Computation (2017), to appear
20. Emmerich, M., Giotis, A., Özdemir, M., Bäck, T., Giannakoglou, K.: Metamodel–assisted evolution strategies. In: Proceedings of the Parallel Problem Solving from Nature-PPSN, pp. 361–370. Springer (2002)
21. Emmerich, M.T., Giannakoglou, K.C., Naujoks, B.: Single-and multiobjective evolutionary optimization assisted by Gaussian random field metamodels. IEEE Transactions on Evolutionary Computation **10**(4), 421–439 (2006)
22. Fernández, A., del Río, S., Chawla, N.V., Herrera, F.: An insight into imbalanced big data classification: outcomes and challenges. Complex & Intelligent Systems **3**(2), 105–120 (2017)
23. Fleming, P.J., Purshouse, R.C.: Evolutionary algorithms in control systems engineering: a survey. Control Engineering Practice **10**(11), 1223–1241 (2002)
24. Forrester, A., Keane, A.: Recent advances in surrogate-based optimization. Progress in Aerospace Sciences **45**, 50–79 (2009)
25. Forrester, A., Sobester, A., Keane, A.: Engineering Design via Surrogate Modelling: A practical guide. Wiley (2008)
26. Forrester, A.I., Sóbester, A., Keane, A.J.: Multi-fidelity optimization via surrogate modelling. In: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences. vol. 463, pp. 3251–3269. The Royal Society (2007)
27. Goel, T., Haftka, R.T., Shyy, W., Queipo, N.V.: Ensemble of surrogates. Structural and Multidisciplinary Optimization **33**(3), 199–216 (2007)
28. Gräning, L., Menzel, S., Ramsay, T., Sendhoff, B.: Application of sensitivity analysis for an improved representation in evolutionary design optimization. In: International Conference on Genetic and Evolutionary Computation. pp. 417–420 (2012)
29. Guo, D., Chai, T., Ding, J., Jin, Y.: Small data driven evolutionary multi-objective optimization of fused magnesium furnaces. In: IEEE Symposium Series on Computational Intelligence. pp. 1–8. IEEE, Athens, Greece (December 2016)
30. Guo, D., Jin, Y., Ding, J., Chai, T.: Heterogeneous ensemble based infill criterion for evolutionary multi-objective optimization of expensive problems. IEEE Transactions on Cybernetics (2018), to appear
31. Gupta, A., Ong, Y.S., Feng, L.: Multifactorial evolution: toward evolutionary multitasking. IEEE Transactions on Evolutionary Computation **20**(3), 343–357 (2016)
32. Gupta, A., Ong, Y.S., Feng, L.: Insights on transfer optimization: Because experience is the best teacher. IEEE Transactions on Emerging Topics in Computational Intelligence **2**(1), 51–64 (2018)
33. Hakanen, J., Sahlstedt, K., Miettinen, K.: Wastewater treatment plant design and operation under multiple conflicting objective functions. Environmental Modelling & Software **46**, 240–249 (2013)
34. Handoko, S.D., Kwoh, C.K., Ong, Y.S.: Feasibility structure modeling: an effective chaperone for constrained memetic algorithms. IEEE Transactions on Evolutionary Computation **14**(5), 740–758 (2010)
35. Hartikainen, M., Miettinen, K., Wiecek, M.M.: PAINT: Pareto front interpolation for nonlinear multiobjective optimization. Computational Optimization and Applications **52**(3), 845–867 (July 2012)
36. Hüsken, M., Jin, Y., Sendhoff, B.: Structure optimization of neural networks for evolutionary design optimization. Soft Computing **9**(1), 21–28 (2005)
37. Ishibuchi, H., Setoguchi, Y., Masuda, H., Nojima, Y.: Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. IEEE Transactions on Evolutionary Computation **21**(2), 169–190 (April 2017). https://doi.org/10.1109/TEVC.2016.2587749
38. Ishibuchi, H., Imada, R., Setoguchi, Y., Nojima, Y.: Reference point specification in Hypervolume calculation for fair comparison and efficient search. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 585–592. ACM, New York, NY, USA (2017)
39. Jansen, J.O., Campbell, M.K.: The GEOS study: Designing a geospatially optimised trauma system for scotland. The Surgeon **12**(2), 61–63 (2014)
40. Jansen, J.O., Morrison, J.J., Wang, H., He, S., Lawrenson, R., Campbell, M.K., Green, D.R.: Feasibility and utility of population-level geospatial injury profiling: prospective, national cohort study. Journal of Trauma and Acute Care Surgery **78**(5), 962–969 (2015)
41. Jansen, J.O., Morrison, J.J., Wang, H., He, S., Lawrenson, R., Hutchison, J.D., Campbell, M.K.: Access to specialist care: optimizing the geographic configuration of trauma systems. Journal of Trauma and Acute Care Surgery **79**(5), 756–765 (2015)
42. Jansen, J.O., Morrison, J.J., Wang, H., Lawrenson, R., Egan, G., He, S., Campbell, M.K.: Optimizing trauma system design: the GEOS (geospatial evaluation of systems of trauma care) approach. Journal of Trauma and Acute Care Surgery **76**(4), 1035–1040 (2014)
43. Jin, Y. (ed.): Knowledge Incorporation in Evolutionary Computation. Springer (2005)
44. Jin, Y.: A comprehensive survey of fitness approximation in evolutionary computation. Soft Computing **9**(1), 3–12 (2005)
45. Jin, Y.: Surrogate-assisted evolutionary computation: Recent advances and future challenges. Swarm and Evolutionary Computation **1**(2), 61–70 (2011)
46. Jin, Y., Branke, J.: Evolutionary optimization in uncertain environments-a survey. IEEE Transactions on Evolutionary Computation **9**(3), 303–317 (2005)
47. Jin, Y., Hüsken, M., Sendhoff, B.: Quality measures for approximate models in evolutionary computation. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 170–173 (2003)

48. Jin, Y., Oh, S., Jeon, M.: Incremental approximation of nonlinear constraint functions for evolutionary constrained optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 1–8. IEEE (2010)
49. Jin, Y., Olhofer, M., Sendhoff, B.: On evolutionary optimization with approximate fitness functions. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 786–793. Morgan Kaufmann Publishers Inc. (2000)
50. Jin, Y., Olhofer, M., Sendhoff, B.: A framework for evolutionary optimization with approximate fitness functions. IEEE Transactions on Evolutionary Computation **6**(5), 481–494 (2002)
51. Jin, Y., Sendhoff, B.: Reducing fitness evaluations using clustering techniques and neural network ensembles. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 688–699. Springer (2004)
52. Jin, Y., Sendhoff, B.: A systems approach to evolutionary multiobjective structural optimization and beyond. IEEE Computational Intelligence Magazine **4**(3), 62–76 (2009)
53. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. Journal of Global Optimization **13**(4), 455–492 (1998)
54. Kazemi, M., Wang, G.G., Rahnamayan, S., Gupta, K.: Metamodel-based optimization for problems with expensive objective and constraint functions. Journal of Mechanical Design **133**(1), 014505 (2011)
55. Kim, H.S., Cho, S.B.: An efficient genetic algorithm with less fitness evaluation by clustering. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 887–894. IEEE (2001)
56. Knowles, J.: ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. IEEE Transactions on Evolutionary Computation **10**(1), 50–66 (2006)
57. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: International Joint Conference on Artificial Intelligence. vol. 14, pp. 1137–1145. Montreal, Canada (1995)
58. Kong, W., Chai, T., Ding, J., Yang, S.: Multifurnace optimization in electric smelting plants by load scheduling and control. IEEE Transactions on Automation Science and Engineering **11**(3), 850–862 (2014)
59. Kourakos, G., Mantoglou, A.: Development of a multi-objective optimization algorithm using surrogate models for coastal aquifer management. Journal of Hydrology **479**, 13–23 (2013)
60. Le, M.N., Ong, Y.S., Menzel, S., Seah, C.W., Sendhoff, B.: Multi co-objective evolutionary optimization: Cross surrogate augmentation for computationally expensive problems. In: Evolutionary Computation (CEC), 2012 IEEE Congress on. pp. 1–8. IEEE (2012)
61. Lim, D., Jin, Y., Ong, Y.S., Sendhoff, B.: Generalizing surrogate-assisted evolutionary computation. IEEE Transactions on Evolutionary Computation **14**(3), 329–355 (2010)
62. Lim, D., Ong, Y.S., Jin, Y., Sendhoff, B.: Evolutionary optimization with dynamic fidelity computational models. In: International Conference on Intelligent Computing. pp. 235–242. Springer (2008)
63. Liu, B., Koziel, S., Zhang, Q.: A multi-fidelity surrogate-model-assisted evolutionary algorithm for computationally expensive optimization problems. Journal of Computational Science **12**, 28–37 (2016)
64. Liu, B., Zhang, Q., Gielen, G.G.: A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems. IEEE Transactions on Evolutionary Computation **18**(2), 180–192 (2014)
65. Liu, Y., Shang, F., Jiao, L., Cheng, J., Cheng, H.: Trace norm regularized CANDECOMP/PARAFAC decomposition with missing data. IEEE Transactions on Cybernetics **45**(11), 2437–2448 (2015)
66. Loshchilov, I., Schoenauer, M., Sebag, M.: Comparison-based optimizers need comparison-based surrogates. In: Proceedings of the Parallel Problem Solving from Nature-PPSN, pp. 364–373. Springer (2010)
67. Luo, J., Gupta, A., Ong, Y.S., Wang, Z.: Evolutionary optimization of expensive multiobjective problems with co-sub-pareto front gaussian process surrogates. IEEE Transactions on Cybernetics (2018)
68. van der Merwe, R., Leen, T.K., Lu, Z., Frolov, S., Baptista, A.M.: Fast neural network surrogates for very high dimensional physics-based models in computational oceanography. Neural Networks **20**(4), 462–478 (2007)
69. Min, A.T.W., Ong, Y.S., Gupta, A., Goh, C.K.: Multi-problem surrogates: Transfer evolutionary multiobjective optimization of computationally expensive problems. IEEE Transactions on Evolutionary Computation (2017), to appear
70. Mogilicharla, A., Chugh, T., Majumder, S., Mitra, K.: Multi-objective optimization of bulk vinyl acetate polymerization with branching. Materials and Manufacturing Processes **29**, 210–217 (2014)
71. Moraglio, A., Kattan, A.: Geometric generalisation of surrogate model based optimisation to combinatorial spaces. In: Proceedings of the European Conference on Evolutionary Computation in Combinatorial Optimization. pp. 142–154. Springer (2011)
72. Namura, N., Shimoyama, K., Obayashi, S.: Expected improvement of penalty-based boundary intersection for expensive multiobjective optimization. IEEE Transactions on Evolutionary Computation **21**(6), 898–913 (2017)
73. Nguyen, S., Zhang, M., Tan, K.C.: Surrogate-assisted genetic programming with simplified models for automated design of dispatching rules. IEEE Transactions on Cybernetics **47**(9), 2951–2965 (2017)
74. Pan, L., He, C., Tian, Y., Wang, H., Zhang, X., Jin, Y.: A classification based surrogate-assisted evolutionary algorithm for expensive many-objective optimization. IEEE Transactions on Evolutionary Computation (2018), to appear
75. Parr, J.M., Forrester, A.I., Keane, A.J., Holden, C.M.: Enhancing infill sampling criteria for surrogate-based constrained optimization. Journal of Computational Methods in Sciences and Engineering **12**(1, 2), 25–45 (2012)
76. Poloczek, J., Kramer, O.: Local SVM constraint surrogate models for self-adaptive evolution strategies. In: Proceedings of the Annual Conference on Artificial Intelligence. pp. 164–175. Springer (2013)
77. Ponweiser, W., Wagner, T., Vincze, M.: Clustered multiple generalized expected improvement: A novel infill sampling criterion for surrogate models. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 3515–3522. IEEE (2008)

78. Rahat, A.A.M., Everson, R.M., Fieldsend, J.E.: Alternative infill strategies for expensive multi-objective optimisation. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 873–880. GECCO '17, ACM, New York, NY, USA (2017)
79. Ratle, A.: Accelerating the convergence of evolutionary algorithms by fitness landscape approximation. In: Proceedings of the Parallel Problem Solving from Nature-PPSN. pp. 87–96. Springer (1998)
80. Regis, R.G.: Evolutionary programming for high-dimensional constrained expensive black-box optimization using radial basis functions. IEEE Transactions on Evolutionary Computation **18**(3), 326–347 (2014)
81. Seung, H.S., Opper, M., Sompolinsky, H.: Query by committee. In: Proceedings of the fifth annual workshop on Computational Learning Theory. pp. 287–294. ACM (1992)
82. Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., de Freitas, N.d.F.: Taking the human out of the loop: A review of bayesian optimization. Proceedings of the IEEE **104**(1), 148–175 (2016)
83. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 69–73 (1998)
84. Sindhya, K., Ojalehto, V., Savolainen, J., Niemistö, H., Hakanen, J., Miettinen, K.: Coupling dynamic simulation and interactive multiobjective optimization for complex problems: An apros-nimbus case study. Expert Systems with Applications **41**(5), 2546–2558 (2014)
85. Singh, H.K., Ray, T., Smith, W.: Surrogate assisted simulated annealing (SASA) for constrained multi-objective optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). pp. 1–8. IEEE (2010)
86. Smith, R., Dike, B., Stegmann, S.: Fitness inheritance in genetic algorithms. In: Proceedings of the ACM Symposium on Applied Computing. pp. 345–350. ACM (1995)
87. Stein, M.: Large sample properties of simulations using latin hypercube sampling. Technometrics **29**(2), 143–151 (1987)
88. Sun, C., Jin, Y., Cheng, R., Ding, J., Zeng, J.: Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems. IEEE Transactions on Evolutionary Computation **21**(4), 644–660 (2017)
89. Sun, C., Jin, Y., Tan, Y.: Semi-supervised learning assisted particle swarm optimization of computationally expensive problems. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 45–52. ACM (2018)
90. Sun, C., Jin, Y., Zeng, J., Yu, Y.: A two-layer surrogate-assisted particle swarm optimization algorithm. Soft Computing **19**(6), 1461–1475 (2015)
91. Sun, C., Zeng, J., Pan, J., Xue, S., Jin, Y.: A new fitness estimation strategy for particle swarm optimization. Information Sciences **221**, 355–370 (2013)
92. Sun, X., Gong, D., Jin, Y., Chen, S.: A new surrogate-assisted interactive genetic algorithm with weighted semisupervised learning. IEEE Transactions on Cybernetics **43**(2), 685–698 (2013)
93. Tabatabaei, M., Hartikainen, M., Sindhya, K., Hakanen, J., Miettinen, K.: An interactive surrogate-based method for computationally expensive multiobjective optimisation. Journal of the Operational Research Society pp. 1–17 (2018)
94. Tian, J., Tan, Y., Zeng, J., Sun, C., Jin, Y.: Multi-objective infill criterion driven Gaussian process assisted particle swarm optimization of high-dimensional expensive problems. IEEE Transactions on Evolutionary Computation (2018), submitted
95. Tian, Y., Wang, H., Zhang, X., Jin, Y.: Effectiveness and efficiency of non-dominated sorting for evolutionary multi-and many-objective optimization. Complex & Intelligent Systems **3**(4), 247–263 (2017)
96. Trev, N.: CFX: Computational Fluid Dynamics, Ansys, HVAC. International Book Market Service Limited (2012)
97. Viana, F.A., Picheny, V., Haftka, R.T.: Conservative prediction via safety margin: design through cross-validation and benefits of multiple surrogates. In: Proceedings of the ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. pp. 741–750. American Society of Mechanical Engineers (2009)
98. Villanueva, D., Haftka, R.T., Le Riche, R., Picard, G.: Locating multiple candidate designs with dynamic local surrogates. In: 10th World Congress on Structural and Multidisciplinary Optimization (WCSMO-10) (2013)
99. Wang, D.J., Liu, F., Wang, Y.Z., Jin, Y.: A knowledge-based evolutionary proactive scheduling approach in the presence of machine breakdown and deterioration effect. Knowledge-Based Systems **90**, 70–80 (2015)
100. Wang, H., Jin, Y., Janson, J.O.: Data-driven surrogate-assisted multi-objective evolutionary optimization of a trauma system. IEEE Transactions on Evolutionary Computation **20**(6), 939–952 (2016)
101. Wang, H.: Uncertainty in surrogate models. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 1279–1279. ACM (2016)
102. Wang, H., Jin, Y., Doherty, J.: Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems. IEEE Transactions on Cybernetics **47**(9), 2664–2677 (2017)
103. Wang, H., Jin, Y., Doherty, J.: A generic test suite for evolutionary multi-fidelity optimization. IEEE Transactions on Evolutionary Computation (2018), to appear
104. Wang, H., Jin, Y., Sun, C., Doherty, J.: Offline data-driven evolutionary optimization using selective surrogate ensembles. IEEE Transactions on Evolutionary Computation (2018), to appear
105. Wang, H., Yao, X.: Corner sort for Pareto-based many-objective optimization. IEEE Transactions on Cybernetics **44**(1), 92–102 (2014)
106. Wang, H., Zhang, Q., Jiao, L., Yao, X.: Regularity model for noisy multiobjective optimization. IEEE Transactions on Cybernetics **46**(9), 1997–2009 (2016)
107. Wang, S., Minku, L.L., Yao, X.: Resampling-based ensemble methods for online class imbalance learning. IEEE Transactions on Knowledge and Data Engineering **27**(5), 1356–1368 (2015)

108. Wang, S., Yao, X.: Multiclass imbalance problems: Analysis and potential solutions. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics **42**(4), 1119–1130 (2012)
109. Willmes, L., Back, T., Jin, Y., Sendhoff, B.: Comparing neural networks and kriging for fitness approximation in evolutionary optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). vol. 1, pp. 663–670. IEEE (2003)
110. Wu, X., Zhu, X., Wu, G.Q., Ding, W.: Data mining with big data. IEEE Transactions on Knowledge and Data Engineering **26**(1), 97–107 (2014)
111. Yu, H., Tan, Y., Zeng, J., Sun, C., Jin, Y.: Surrogate-assisted hierarchical particle swarm optimization. Information Sciences **454-455**, 59–72 (2018)
112. Yuan, B., Li, B., Weise, T., Yao, X.: A new memetic algorithm with fitness approximation for the defect-tolerant logic mapping in crossbar-based nanoarchitectures. IEEE Transactions on Evolutionary Computation **18**(6), 846–859 (2014)
113. Zapotecas Martínez, S., Coello Coello, C.A.: MOEA/D assisted by RBF networks for expensive multi-objective optimization problems. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 1405–1412. ACM (2013)
114. Zhang, Q., Zhou, A., Zhao, S., Suganthan, P., Liu, W., Tiwari, S.: Multiobjective optimization test instances for the CEC 2009 special session and competition. Tech. rep., University of Essex, Colchester, UK and Nanyang Technological University, Singapore, Special Session on Performance Assessment of Multi-Objective Optimization Algorithms, Technical Report (2008)
115. Zhang, Q., Liu, W., Tsang, E., Virginas, B.: Expensive multiobjective optimization by MOEA/D with Gaussian process model. IEEE Transactions on Evolutionary Computation **14**(3), 456–474 (2010)
116. Zhou, Z.H.: Ensemble Methods: Foundations and Algorithms. CRC Press (2012)
117. Zhou, Z.H., Chawla, N.V., Jin, Y., WILLIma, G.J.: Big data opportunities and challenges: Discussions from data analytics perspectives. IEEE Computational Intelligence Magazine **9**(4), 62–74 (2014)
118. Zhou, Z.H.Z., Chawla, N.V., Jin, Y., Williams, G.J.: Big data opportunities and challenges: Discussions from data analytics perspectives. IEEE Computational Intelligence Magazine **9**(4), 62–74 (2014)
119. Zhou, Z., Ong, Y.S., Nair, P.B., Keane, A.J., Lum, K.Y.: Combining global and local surrogate models to accelerate evolutionary optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **37**(1), 66–76 (2007)
120. Zhou, Z., Ong, Y.S., Nguyen, M.H., Lim, D.: A study on polynomial regression and Gaussian process global surrogate model in hierarchical surrogate-assisted evolutionary algorithm. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC). vol. 3, pp. 2832–2839. IEEE (2005)
121. Zhuang, L., Tang, K., Jin, Y.: Metamodel assisted mixed-integer evolution strategies based on kendall rank correlation coefficient. In: International Conference on Intelligent Data Engineering and Automated Learning. pp. 366–375. Springer (2013)