IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 1 June 2024, accepted 17 June 2024, date of publication 19 June 2024, date of current version 26 June 2024.
Digital Object Identifier 10.1109/ACCESS.2024.3416811

RESEARCH ARTICLE

Surrogate Models and Ensemble Strategies for Expensive Evolutionary Optimization: An Industrial Case Study

MARIO GARZA-FABRE, NICOLÁS CORTÉS-GARCÍA, HIRAM GALEANA-ZAPIÉN, (Member, IEEE), AND RICARDO LANDA
Center for Research and Advanced Studies, Cinvestav Campus Tamaulipas, Ciudad Victoria, Tamaulipas 87130, Mexico
Corresponding author: Mario Garza-Fabre (mario.garza@cinvestav.mx)

ABSTRACT Expensive optimization problems are characterized by the significant amount of time and resources needed to determine the quality of potential solutions. This poses severe limitations for the application of metaheuristic optimization methods, such as evolutionary algorithms, as they usually require evaluating many candidate solutions to deliver satisfactory results. Surrogate model-based strategies have become a popular choice to tackle this type of problem. The key idea of these strategies is to build a model which can approximate and (partially) replace the mechanisms for assessing solution quality, such that the use of this less expensive alternative lowers the overall computational cost of the optimization process. This paper analyzes surrogate model-based strategies in the context of a specific, expensive, combinatorial optimization problem: the configuration of the gas distribution system for an electrostatic precipitator. Focusing on this relevant case study from industry, the aim of this paper is twofold: (i) to investigate the most suitable learning techniques for building the surrogate models and (ii) to explore the advantages of ensemble strategies allowing various surrogate models to collaborate during optimization. This contrasts with previous studies where a single, fixed modeling technique is adopted to address this problem. The experimental evaluation involves eight different learning techniques, three alternative ensemble strategies, and two reference approaches from the literature (previously used to tackle this specific problem). Our results reveal the best modeling techniques at the individual level, while highlighting clear benefits of the simultaneous exploitation of multiple surrogate models when facing this particular optimization challenge.

INDEX TERMS Electrostatic precipitator problem, evolutionary algorithms, expensive optimization, surrogate models, surrogate model-based optimization, surrogate-model ensembles.

I. INTRODUCTION

Expensive optimization problems arise in different fields and involve costly, resource-demanding mechanisms to assess the quality of potential solutions [1]. The expensive nature of such mechanisms stems from conditions which are inherent to specific applications; for example, the use of computationally intensive simulations [2], [3], [4], [5], the time-consuming fabrication and physical inspection of candidate prototypes [6], [7], or the need to process large volumes of data [8], [9]. The main challenge in expensive optimization is that the high cost of solution evaluations renders the use of conventional evolutionary algorithms and other metaheuristics impractical, or even prohibitive, as these methods usually evaluate thousands (or tens of thousands) of solution samples during optimization.

Surrogate model-based optimization (SMBO) has attracted increasing attention in recent years [10], [11], [12]. The capabilities of this approach in dealing with expensive problems have earned it recognition as one of the main categories of global optimization techniques in modern taxonomies [13]. As explained in detail in Section II-B,

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

SMBO aims to accelerate the convergence of the search process. This translates into achieving satisfactory results but, at the same time, reducing the number of solution evaluations required to do so (keeping it consistent with the limited budget of resources that may be available in a given application scenario). A key aspect that allows SMBO to accomplish such a goal, is that the evaluations carried out are further exploited to build a *surrogate model* which serves as an inexpensive alternative to the original evaluation mechanisms. This enables additional optimization efforts (based on the approximation model in lieu of the expensive problem mechanisms), increasing performance while maintaining the computational costs reasonable.

The use of *ensembles* has also emerged as a promising approach within the context of SMBO. Given the variety of learning techniques that exist for building the surrogate models [13], [14], but also the fact that a universally best method is unlikely to exist (as per the *no-free-lunch theorem* [15]), the adoption of strategies which can simultaneously leverage the strengths of multiple techniques is, at least conceptually, advantageous. Whereas opting for a single modeling technique implies relying on assumptions about problem characteristics (which are difficult to know in advance, especially in expensive optimization), ensembles may represent a more flexible alternative. Such an approach has reported encouraging results [10], [16], [17], [18], [19], as seen in Section II-E.

A. RESEARCH GOALS AND MOTIVATION

In this paper, we explore the use of SMBO and ensemble strategies in the context of a real-world industrial challenge: the *electrostatic precipitator (ESP) problem* [20], [21]. ESPs are filtering devices commonly used for cleaning the exhaust gases of industrial processes. Optimizing the design of these devices' components maximizes their efficiency, reduces maintenance costs, and is critical to meeting particulate emission regulations [22]. As described in Section II-C, the evaluation of candidate solutions for the ESP problem requires running an intensive *computational fluid dynamics* (CFD) simulation. This characteristic makes it not only computationally expensive, but also a *black-box optimization* [23] scenario whereby problem peculiarities cannot be exploited to devise a tailored algorithmic solution. This has motivated the use of SMBO in previous studies [20], [21], [24], [25], where specific surrogate-modeling techniques (in particular, *kriging*, as seen in Section II-D) have been adopted and proved beneficial in dealing with this optimization task.

Given the expensive, black-box nature of the ESP problem and, therefore, the unknown characteristics of its search landscape, the question of which modeling technique is the most suitable remains open. Hence, rather than choosing a technique *a priori* (as in previous works [20], [21], [24], [25]), we conduct a comparative analysis of eight learning approaches with the aim of identifying the ones standing out in this specific setting. Furthermore, we investigate whether the use of surrogate-model ensembles can yield a better performance than the use of individual models, an approach which has not been considered in the particular context of this application (to the best of our knowledge). Some types of models might perform better than others at different stages of the optimization process, and depending on the training samples available at each stage, with several important factors such as the quantity, variety, quality, and representativeness of these samples. Thus, the combination of multiple modeling techniques through an ensemble strategy may offer an increased robustness. One of the goals of our experimental study is to determine whether this hypothesis holds in practice.

B. CONTRIBUTIONS

In summary, the contributions of this paper are as follows:

- We focus on a particular case study from industry, the ESP problem, which is of high practical relevance and involves clear economic and environmental impacts. This is also an expensive, black-box problem, which makes it interesting from the computational and optimization perspectives.
- Our comparison of surrogate-modeling techniques showcases the most effective learning approaches (at the individual level) for this real-world application. Previous studies have considered specific, fixed models only.
- We explore, for the first time, the use of surrogate-model ensembles in the context of the ESP problem. Our evaluation reveals significant advantages of ensemble strategies over the use of individual models, achieving an almost 4-fold speedup in optimization convergence.
- SMBO applications in combinatorial optimization are scarce compared to the significant volume of research focused on continuous domains [10]. Given the combinatorial nature of the ESP problem, this study contributes to the body of knowledge on this subject.

C. ORGANIZATION

This paper is organized as follows. First, Section II introduces background concepts and the ESP problem, discussing the relevant literature. Then, Section III describes the flexible framework that we consider for the implementation of surrogate models and ensemble strategies. Section IV presents our experimental analysis. Finally, Section V concludes.

II. BACKGROUND CONCEPTS AND RELATED WORK

This section provides background concepts on optimization, expensive optimization problems, and SMBO (Sections II-A and II-B). It also describes the industrial case study considered in this paper (Sections II-C), the previous works related to this application (Sections II-D), and the relevant literature about surrogate-model ensembles (Sections II-E).

A. OPTIMIZATION AND EXPENSIVE PROBLEMS

Optimization refers to the process of identifying the best possible way to solve a given problem. This can be stated more formally as the task of finding a solution $\mathbf{x}^*$, defining a specific configuration for a set of *design parameters* or *decision variables*, such that this configuration yields the best outcome for a certain performance criterion f. Assuming that f is to be minimized (without loss of generality), this means that $f(\mathbf{x}^*) = \min\{f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{X}\}$, where $\mathcal{X}$ denotes the set of all possible solutions, i.e., the *search space*. Criterion $f : \mathcal{X} \to \mathbb{R}$ is a key component, formally known as the *objective function* of the problem. By reflecting solution-quality aspects which are specific to the application domain, f allows us to discriminate among candidates in $\mathcal{X}$, serving the critical purposes of guiding the optimization process.

Whereas some problems have a function f that is readily available to be computed, as it is given by an algebraic, closed-form expression, some others lack such an explicit definition. The latter case corresponds to the so-called *black-box optimization problems* [23] and arises, for example, when evaluating solution quality involves capturing aspects or phenomena which are too complex to be modeled algebraically. In these scenarios, evaluating solution quality often requires specialized simulation software [2], [3], [4], [5], processing large volumes of data [8], [9], or even human intervention [6], [7].

The above situations may cause the evaluation of solution quality to become expensive, in the sense that it is time-consuming and/or resource-intensive (such is the case of the ESP problem studied here, see Section II-C). This gives rise to the category of *expensive optimization problems* [1], for which the high cost of computing f collides with an inherent limitation of conventional optimization methods: these methods usually require sampling and evaluating a large number of candidate solutions during the search process, which results impractical in the context of expensive optimization. These characteristics, therefore, call for alternative methodologies that can handle expensive problems more effectively; one such methodology is SMBO, as discussed in Section II-B.

B. SURROGATE MODEL-BASED OPTIMIZATION

As discussed before, expensive problems prevent the use of conventional optimization techniques due to the high costs that the mechanisms for evaluating solution quality introduce. The goal in expensive optimization is thus to increase convergence speed; more precisely, the goal is to achieve the best possible quality while respecting the limited budget of time and resources available for the task. This leads to a maximum number of *effective solution evaluations* that can be performed (where, by effective, we refer to those using the expensive, problem-specific evaluation mechanisms).

The framework of SMBO aligns with the above goal. As illustrated in Fig. 1, it exploits the reduced number of effective evaluations carried out, so that the knowledge acquired from those evaluations can be used to train a model which enables the consideration of a larger number of candidate solutions (for which quality is estimated by the model rather than explicitly computed using the original evaluation mechanisms). In this way, the effective evaluations

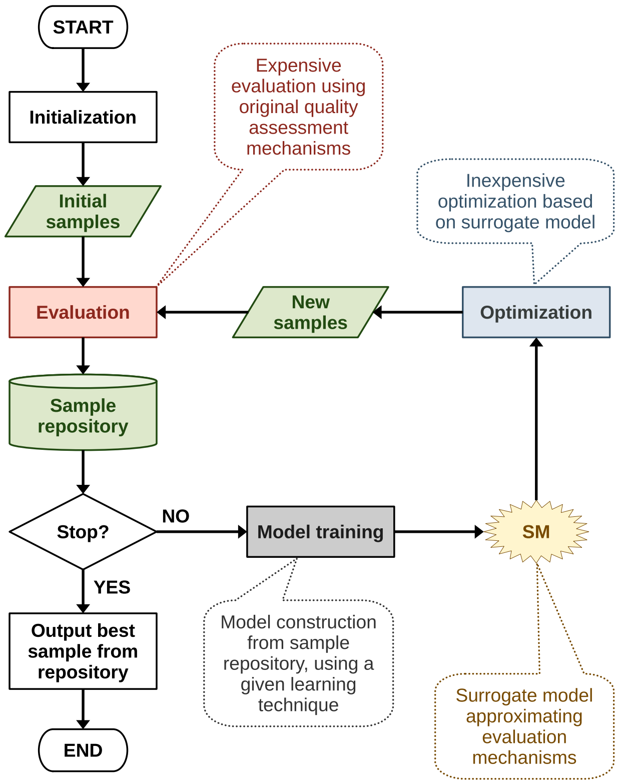
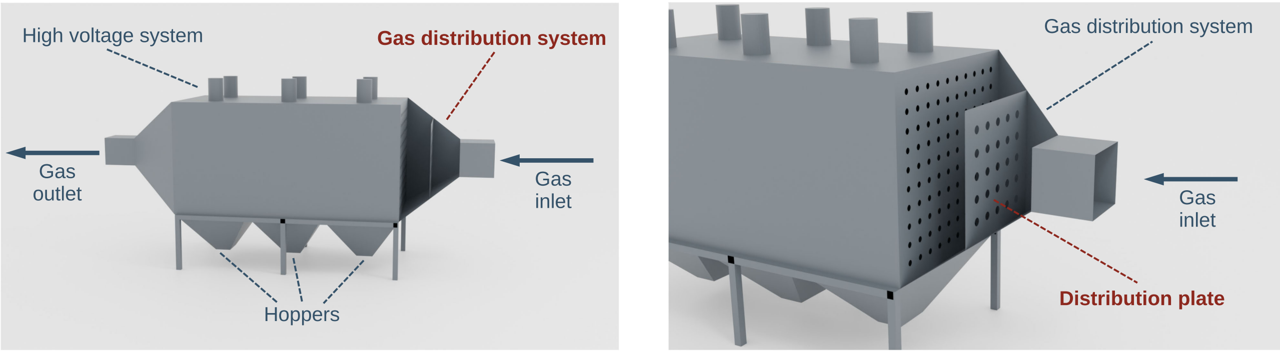


FIGURE 1. Surrogate model-based optimization (SMBO) framework.

represent samples of the search landscape, and the obtained *surrogate model* (also referred to as *metamodel* or *response surface*) serves as an approximation for the landscape's unsampled regions [16].

As can be seen from Fig. 1, the SMBO process starts by generating some initial solution samples, whose quality is determined using the problem's (expensive) evaluation mechanisms. This results in a repository that represents the knowledge acquired so far, as given by the solutions sampled together with their evaluation information. The repository serves as input for building a predictive model, which is then used as a surrogate for the evaluation mechanisms during the application of a given optimizer. Note that this optimizer continues to explore the search space, sampling (potentially many) additional solutions, but these solutions are inexpensively evaluated on the basis of the surrogate model. Only the solution(s) generated as output by this optimizer are evaluated using the original mechanisms, and then added to the repository to contribute to model training in subsequent iterations. This process repeats until a stopping condition is met (usually, until the available budget of evaluations is exhausted).

It is important to highlight that other possible realizations of the SMBO framework exist, defining alternative ways to integrate learning mechanisms and the obtained surrogate models into the optimization process [10], [11], [12]. Also, the description provided above is generic, aiming to illustrate the functioning and rationale of SMBO only. The



(a) Electrostatic precipitator (b) Gas distribution system

FIGURE 2. Illustration of an electrostatic precipitator. The gas distribution system, which is the focus of the optimization problem studied here, involves metal plates that control the flow so that the incoming gases reach the particle-separation zones in optimal conditions (regarding speed and distribution). Electrostatic precipitators, commonly used in cement plants and other industrial settings, are large devices that can weigh between 4500 and 6900 kg.

application of this framework to a given problem requires a precise definition of its components (namely, the particular initialization, learning, and optimization routines adopted, as well as their settings), as specified in Section III for this paper's case study.

C. THE ELECTROSTATIC PRECIPITATOR PROBLEM

Electrostatic precipitation is the technique of electrostatically charging fine particles (using a high-voltage, high-frequency power supply) to separate and remove them from a gas stream [26], [27]. ESPs are filtering devices following the above principle, commonly used to clean the exhaust gases produced by various industrial processes and prevent the release of polluting particles into the environment [22], [28]. These devices are, therefore, essential to allow large-scale industries to comply with stringent emission standards.

An industrial ESP is illustrated in Fig. 2. The figure emphasizes the gas distribution system, whose configuration is the particular subject of the optimization problem addressed in this paper: the ESP problem. The gas distribution system is a critical component that controls the gas flow through the ESP. It consists of a collection of slots where perforated metal plates are inserted to block, reduce, or redirect the gas so that it can spread uniformly across the particle-separation zones of the ESP. The number, location, and porosity of these plates are key parameters on which flow uniformity depends [29]. The proper configuration of this component allows the gas to reach separation zones under optimal conditions (regarding flow distribution and velocity) and is, therefore, essential for the correct operation and efficiency of the entire device.

The ESP problem is an industrial case study which is not only interesting from the optimization and computer science perspectives, but also relevant from a practical standpoint. Addressing this real-world problem involves significant impacts in terms of operational expenses, adherence to regulatory requirements, and environmental considerations [22]. A more formal description of the optimization challenge that this particular application represents is provided below.

The ESP problem considers a device equipped with a gas distribution system, which features S slots and P alternative plate options for each of them. From the optimization viewpoint, each configurable slot constitutes a design parameter, and the discrete (and categorical) nature of the available plate alternatives makes this a combinatorial problem. A potential solution to this problem defines a full configuration, with a specific choice of plate for every slot of the gas distribution system. Besides its exponential search space of P^S possible solutions, the ESP problem is an expensive optimization task. That is, a distinctively challenging characteristic of the ESP problem is the high cost associated with the quality assessment of candidate configurations, as it requires running a (computationally intensive) CFD simulation.

The particular abstraction of the ESP problem studied here was introduced by Rehbach et al. [20], [21], and was adopted as the *Industrial Challenge* within the competitions of the 2020 edition of the *Genetic and Evolutionary Computation Conference* (GECCO 2020).[1] The problem comprises a total of $S = 49$ slots and $P = 8$ plate types. It is important to note that this is a reduced version of an originally more complex problem, with $S = 334$ slots. In the original (real) problem, a single solution evaluation based on the CFD simulation takes about eight hours to run using 16 CPUs, in contrast to the two minutes required for the reduced version [21]. Despite such a significant decrease in complexity, Rehbach et al. highlight that the reduced problem still preserves the main difficult features of the original one, reproducing its rugged search landscape [21]. The consideration of this reduced problem version has allowed us to conduct the extensive experiments and comparative analyses presented later in this paper.

D. PREVIOUS WORK ON THE ESP PROBLEM

The inherently expensive nature of the ESP problem reflects on the fact that SMBO is at the core of all previous efforts to tackle this challenge. To the best of our knowledge, the ESP problem is first considered by Rehbach et al. [20], who adopted *kriging* (also referred to as Gaussian process

[1] https://gecco-2020.sigevo.org

regression) as the surrogate-modeling technique. In addition, the authors propose a parallel computing approach to further deal with problem complexity. The use of kriging in dealing with the ESP problem is also reported in more recent studies by the same research group [21], [30]. These studies propose the incorporation of dimensionality reduction strategies, which helps to further alleviate the computational burden in comparison to standard SMBO. However, this increase in computational efficiency is achieved at the expense of some decreases in optimization performance (i.e., reporting lower solution qualities) for some of the strategies analyzed.

The ESP problem is considered as an example application in a recent study by Karlsson et al. [24]. The work aims to show that continuous surrogate models, such as kriging, can be successfully applied to this kind of problem involving discrete decision variables. Finally, a modified version of the problem, to which some continuous variables are added, is investigated in the context of the *mixed-variable ReLU-based surrogate modeling* (MVR) method by Bliek et al. [25].

As can be seen, previous works have all adopted specific surrogate-modeling techniques to address the ESP problem. Nevertheless, the question of what type of model is the most appropriate for this particular application remains open. Identifying the most suitable model type for a given problem is not straightforward and has been regarded as a challenging task on its own [10], [13]. It cannot be expected that a certain technique will provide the best results in every scenario, which is a consequence of the *no-free-lunch theorem* [15]. Model type selection implies making assumptions about the suitability of particular learning techniques according to problem characteristics [14]. But our inability to explore the search landscape in advance, which is inherent to expensive optimization problems, prevents us from exploiting such specificities to make a fully informed decision. Therefore, rather than choosing a fixed model type in advance (as in previous studies), this paper investigates and compares a range of alternative learning techniques, analyzing which of them are the most effective in dealing with the ESP problem. Furthermore, surrogate-model ensembles have emerged as a promising approach to circumvent the aforementioned difficulties (as discussed in Section II-E); this paper explores such an approach for the first time in the specific context of the ESP problem.

E. SURROGATE-MODEL ENSEMBLES

When the nature of the search landscape is unknown, and a tailored approach is therefore difficult to elucidate, the simultaneous exploitation of multiple surrogate models may represent a suitable alternative, providing an increased flexibility and robustness with respect to the diversity of problem characteristics. Strategies enabling this alternative are known by different names, such as ensemble, multi-surrogate, and committee approaches. Although these terms are often used distinctly to highlight peculiarities of the mechanisms through which multiple surrogates collaborate, we adopt the term *ensemble* to generally refer to strategies allowing two or more models to cooperatively assist the optimization process.

In contrast to single-model approaches, ensembles allow the integration of multiple, potentially heterogeneous surrogates (each constructed using a different learning technique). In the words of Stork et al. [13], "*the goal is to create a sophisticated predictor that surpasses the performance of a single model*". Some models may perform better than others depending on the problem addressed, as well as on the availability of training samples and the regions of the landscape where efforts are concentrated at a given search stage [2]. Thus, ensemble members may complement each other and synergistically provide a more reliable approximation [31].

However, there remains the question of how to accomplish such an integration of multiple models. Various approaches have been explored [32]. A simple and intuitive strategy is to always select the most-promising surrogate from the ensemble members (the one with the lowest prediction error, as observed during training). This has reported better results than the use of fixed, individual surrogate models [16]. Linear combinations of multiple surrogates have also been considered, where models are (locally and adaptively) weighted according to their prediction variances [2] or errors [16]. The weighting of ensemble members has also been addressed as an optimization problem itself [33]. The aggregation of surrogate models is further investigated in other independent studies [31], [34], [35], including more comprehensive analyses on the issues of how to choose and properly weight the models to be combined [36], [37]. Instead of explicitly defining weights, an interesting alternative which has been explored is stacking [10], [38]: a number of individual surrogates are first trained, and a (potentially) more robust final model is then built by exploiting the predictions of such individual surrogates as explanatory (input) variables.

Multiple surrogate models have also been used to simultaneously capture global and local aspects of the fitness landscape. In [39], for instance, a global surrogate is first constructed from all available training samples, and an evolutionary search is performed based on it. Then, the best solution reached during such a global search, and a number of samples adjacent to it, are used to build a local model optimized within a local search procedure. Local surrogate models are also considered in [17], where each individual generated through the evolutionary operators is subjected to local search improvement, relying on a surrogate model trained locally within its neighborhood (using only the nearest samples). The novelty of this proposal is that the modeling technique is chosen independently for each solution on the basis of a measure of evolvability. Such a measure reflects the fitness improvement that can be expected by using each available modeling technique, which is inferred from historical data collected throughout the search.

The importance of balancing global and local models has been highlighted [18]. It was shown empirically that global surrogates are useful in smoothing the landscape, which is particularly relevant to deal with multimodality. Instead, local models offer more accurate approximations, facilitating the exploitation of local optima basins. Some algorithms switch from global to local surrogate models and vice versa; whenever no improvement is observed using one of these strategies, they switch to the other [40]. To improve model performance at the local level, it has also been proposed that their training focuses on the highest-fitness samples only [41], which contrasts with the idea of constructing local models by filtering samples based on the notion of distance. A similar approach was reported more recently [19], where the resulting local model and a global surrogate trained on all samples constitute the two separate tasks to be solved by a multi-tasking evolutionary optimization method. It is argued that optimizing the global surrogate contributes to exploration, while optimizing the local model intensifies the search in promising sub-regions.

Ensembles can also be used to determine the degree of uncertainty in solution-quality estimations. A high discrepancy in the outputs produced by the ensemble members is indicative of high uncertainty [2], [40], which suggests an undersampled region of the landscape (the inclusion of new samples from that region to the training set is likely to increase prediction accuracy). Some works have used ensembles to improve computational efficiency in algorithms that exploit uncertainty information during optimization [42], [43]. The use of a single (global) surrogate model (namely, kriging), which informs about uncertainty but scales poorly regarding the size of the training set, is replaced with an ensemble of either homogeneous local computations of the same model [42] or heterogeneous more scalable alternatives [43], which can deliver uncertainty information less expensively.

Furthermore, multiple surrogates have been employed separately to conduct parallel local searches, each based on a different model. In [44], for example, every candidate individual is replaced by the best outcome from a set of parallel local searches starting from it. A related approach is adopted in [31], where each individual is replaced by the best out of two local-search outcomes. The first local search optimizes a weighted aggregation of multiple models, intended to mitigate the "*curse of uncertainty*" caused by the inaccuracies of the individual surrogates. In contrast, the second local search seeks to exploit the "*bless of uncertainty*", using a single surrogate model as a means to smooth the fitness landscape.

III. FLEXIBLE FRAMEWORK FOR SURROGATE MODEL-BASED OPTIMIZATION

This section describes the specific framework adopted in this study for the analysis and comparison of surrogate models and ensemble strategies. First, general aspects of this framework are discussed in Section III-A, complementing explanations regarding SMBO provided earlier in Section II-B. Then, Sections III-B, III-C, and III-D elaborate on the particular instantiation of the SMBO framework we consider, including the internal optimizer and settings used, as well as the specific modeling techniques and ensemble strategies evaluated.

Note that the literature spans a variety of approaches that enable the incorporation of surrogate models into the search process, either individually or in the form of ensembles [10]. Rather than investigating the best possible method, our aim is to adopt a fixed but flexible framework which allows us to assess the impact of using different model types and ensemble strategies on optimization performance. Also, whereas some works have explored the notion of locality of the models (such as those discussed in Section II-E), our study focuses on global surrogates trained over all available landscape samples. Moreover, we adopt an *online* (also called *sequential*) approach to SMBO, where models are (repeatedly) trained as samples are systematically gathered throughout the optimization process; this contrasts with the *offline* approach where models are built in advance and completely replace the problem's evaluation mechanisms during the search [45].

A. OVERALL FUNCTIONING OF THE SMBO FRAMEWORK

The general functioning of the flexible SMBO framework considered in this work is outlined in Algorithm 1. As can be seen, the process starts by initializing a repository, R_0, with $|R_0| = N_R$ candidate solutions (Algorithm 1, Line 2). The solution samples in R_0 are evaluated using the real (expensive) objective function (Algorithm 1, Line 3), so that this repository can later be used as the training set during the construction of the initial surrogate model(s), to obtain an incipient approximation to the problem's landscape.

Algorithm 1 Flexible SMBO Framework

1: $t \leftarrow 0$
2: $R_t \leftarrow$ Initialization(N_R) ▷ Create repository with N_R initial solutions
3: Evaluation(R_t) ▷ Evaluate using real objective function
4: **while** ⟨ stopping criterion ⟩ **do** ▷ Iterate until stop condition is met
5: $\{m_t^1, \ldots, m_t^n\} \leftarrow$ ModelTraining(R_t) ▷ Train n models over R_t
6: $x \leftarrow$ Optimization($\{m_t^1, \ldots, m_t^n\}$) ▷ Optimization based on models
7: Evaluation(x) ▷ Evaluate using real objective function
8: $R_{t+1} \leftarrow R_t \cup \{x\}$ ▷ Include new solution to repository
9: $t \leftarrow t + 1$
10: **end while**
11: Return the best solution from R_t as output

Once the initial repository R_0 has been created and all samples in it have been evaluated, the iterative process of SMBO is performed until a given stop condition is met (Algorithm 1, Lines 4-10). At each iteration t, a set of $n \geq 1$ surrogate models are trained over all samples in R_t (Algorithm 1, Line 5). The $n \geq 1$ resulting models are used (in one way or another) by a chosen optimizer as a replacement for the objective function within an inexpensive, internal optimization process (Algorithm 1, Line 6). In other words, the evaluation of solutions in this internal optimization process relies on the surrogate models instead of the expensive mechanisms of the problem. Only x, the new candidate solution delivered by the internal optimizer as output (i.e., the one with the best quality, as estimated by the models), is evaluated using the original (expensive) objective function (Algorithm 1, Line 7). After this, solution x is included in R_t to serve as a training sample in subsequent iterations (Algorithm 1, Line 8). The addition of a new sample at each iteration is expected to contribute to defining increasingly accurate surrogates and, thus, to the discovery of higher-quality solutions over time. In the end, the best sample from R_t is delivered as the final solution.

It is worth emphasizing the following aspects regarding the training process and the usage of the obtained surrogate models by the internal optimizer:

- The training data consists of both the solution samples in R_t and their quality-assessment information, as derived through the *effective evaluation* of solutions.[2] Data is gradually collected as new candidate solutions are discovered and evaluated at each iteration, therefore leading to an online approach to SMBO, as discussed earlier.
- Models are intended to capture the relationship between the characteristics and the quality of solutions, so that they can be exploited to estimate (predict) the quality of other solution samples during the internal optimization process, replacing the original evaluation mechanisms.
- Models do not need to be accurate at estimating solution quality. What is important is that models allow us to properly discriminate among candidate solution samples, so that they can guide the internal optimization process towards promising regions of the search space. Even poor estimations can be useful, as long as they induce a reasonable ordering among solutions. Hence, model accuracy is a sufficient but not necessary condition to enable such an effective discrimination.
- Model training and the subsequent estimation of solution quality by the obtained models is significantly cheaper than the effective evaluation through the original problem mechanisms, as we are dealing with expensive optimization scenarios (refer to Section II-A).
- Each of the $n \geq 1$ predictive models can potentially be obtained through a different machine learning technique.

The following subsections provide details on the particular components and parameter settings adopted for the above framework, including the initialization routine and the internal optimizer chosen, as well as the surrogate-modeling techniques and ensemble strategies evaluated.

B. OPTIMIZER, ALGORITHMIC CHOICES, AND SETTINGS

We adopt specific algorithmic design choices and settings, keeping them constant throughout the experiments of this study. This allows us to more accurately evaluate the impact of varying the components that are particularly relevant for the purposes of this research: the surrogate-modeling techniques and ensemble strategies. First, the training repository is initialized with $N_R = 10$ candidate solutions, which in all cases are generated uniformly at random. In both the outer framework described in Section III-A and the internal optimizer introduced below, candidate solutions are encoded using the genetic representation illustrated in Fig. 3. This is an integer encoding with genotype length $\ell = S$, specifying which of the $P = 8$ plate choices is used to configure each of the $S = 49$ slots in the gas distribution system of the ESP (see Section II-C). As the stop condition for the whole SMBO process, we consider a fixed budget of $E_{max} = 200$ evaluations of the real (expensive) objective function. This means that, after creating and evaluating the initial solution repository (which consumes $N_R = 10$ evaluations), the method outlined in Algorithm 1 is allowed to perform a total of 190 iterations (each producing and evaluating an additional sample).

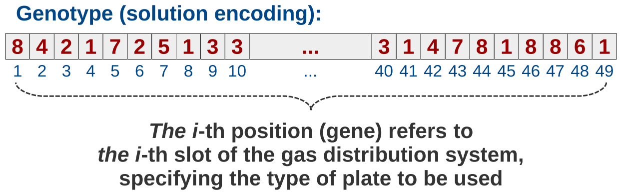


FIGURE 3. Genetic representation used to encode candidate solutions. The genotype involves $\ell = S$ genes, accounting for the $S = 49$ slots of the gas distribution system in the ESP device considered in this study. Each gene can assume an allele value from set $\{1, 2, \ldots, 8\}$, denoting the $P = 8$ plate types available for the configuration of each slot.

Regarding the internal optimizer, we use the *genetic algorithm* (GA) sketched in Algorithm 2. After generating an initial population of candidate individuals (Algorithm 2, Line 3), the GA follows a standard evolutionary process consisting of parent selection, genetic variation, and survivor selection (Algorithm 2, Lines 5-11), which repeats until a given criterion is satisfied. The aspect worth emphasizing is that, as discussed before, this algorithm is provided with a set of $n \geq 1$ surrogate models, which will serve the purposes of (inexpensively) estimating the quality of individuals during this internal optimization process (Algorithm 2, Lines 4 and 8). Therefore, the GA navigates the problem's landscape, as approximated by the given surrogate models. Note that the case with $n = 1$ corresponds to a conventional SMBO scheme, where a single surrogate model is directly used to impose an ordering among candidate individuals and guide the evolutionary process. The modeling techniques evaluated are discussed in Section III-C. However, when $n > 1$, an ensemble strategy is required such that, in one way or another, the multiple models can contribute to steering the search. Section III-D discusses the ensemble strategies considered.

The GA uses a random initialization routine and a fixed population size of $N_P = 100$ individuals. Standard,

[2] As discussed in Section II-B, we consider that an effective evaluation occurs when solution quality is explicitly computed using the original (expensive) objective function (or evaluation mechanisms) of the problem.

Algorithm 2 Model-Based GA Used as Internal Optimizer

1: $M \leftarrow \{m_t^1, \ldots, m_t^n\}$ ▷ Given set of n surrogate models
2: $g \leftarrow 0$
3: $P_g \leftarrow$ Initialization(N_P) ▷ Create population of N_P initial individuals
4: ModelEvaluation(P_g, M) ▷ Evaluate/rank individuals over M
5: **while** ⟨ stopping criterion ⟩ **do**
6: $P'_g \leftarrow$ MatingSelection(P_g) ▷ Select parents for reproduction
7: $O_g \leftarrow$ VariationOperators(P'_g) ▷ Generate offspring individuals
8: ModelEvaluation(O_g, M) ▷ Evaluate/rank individuals over M
9: $P_{g+1} \leftarrow$ SurvivorSelection($P_g \cup O_g$) ▷ Select next generation's population
10: $g \leftarrow g + 1$
11: **end while**
12: Return the best individual from P_g as output

TABLE 1. Modeling techniques and ensemble strategies investigated.

Acronym	Learning technique / Strategy	Individual / Ensemble
AD	Adaptive Boosting (AdaBoost)	Individual model
DT	Decision Tree	Individual model
KN	k-Nearest Neighbors	Individual model
KR	Kriging (Gaussian Process Regression)	Individual model
NN	Neural Network (Multilayer Perceptron)	Individual model
RB	Radial Basis Function Network	Individual model
RF	Random Forest	Individual model
SV	Support Vector Regression	Individual model
ER	Lowest-Error Model Selection	Ensemble strategy
AV	Average of Predicted Values	Ensemble strategy
RK	Average of Induced Ranks	Ensemble strategy

well-known operators are adopted for the selection and genetic variation processes. Specifically, parent selection is accomplished by means of (deterministic) binary tournament. For variation, the uniform crossover and uniform mutation operators are used (which are applied with probabilities $p_c = 0.9$ and $p_m = 1/\ell$, respectively, where ℓ is the genotype length). Finally, survivor selection chooses the best individuals from the union of the parent and offspring populations, ensuring the removal of duplicates to promote diversity. The evolutionary process is repeated for a total of $G_{max} = 100$ generations.

C. SURROGATE-MODELING TECHNIQUES

We consider eight distinct supervised learning techniques for constructing the surrogate models. These modeling techniques are listed in Table 1. Note that a two-letter acronym has been assigned to refer to these approaches hereafter. Also, note that we are concerned with regression techniques, as the models are expected to predict continuous values reflecting solution quality (albeit some of these techniques are applicable to classification problems as well). Finally, it is worth noting that two of these approaches, AD and RF, are ensemble methods on their own (their predictions are computed from those produced by a collection of base learners). We are interested in investigating the effectiveness of AD and RF as individual surrogate-modeling techniques, but also the advantages of combining them with other learning methods through the ensemble strategies described in Section III-D.

It is important to remark that the eight learning techniques analyzed involve hyperparameters whose configuration impacts the obtained models and their predictions. With the aim to favor a fair comparison and to give all techniques the opportunity to perform at their best, the model-training process includes cross-validated tuning of hyperparameters in all the cases. Notice that training (and the embedded tuning) occurs at every iteration of the SMBO process (as seen in Algorithm 1 and in Section III-A), and that the best settings identified for each model type may vary from one iteration to another. This makes it challenging to investigate the influence of each specific parameter on performance (an assessment which is beyond the scope of this study).

D. ENSEMBLE STRATEGIES

Below we introduce three mechanisms for enabling the integration of multiple surrogate models into the optimization process. They are implemented at the internal optimization step of the SMBO framework (Algorithm 1, Line 6), more specifically at the evaluation of candidate individuals (Algorithm 2, Lines 4 and 8). These mechanisms, and the acronyms adopted to refer to them, are summarized in Table 1.

1) LOWEST-ERROR MODEL SELECTION

As discussed in Section III-A, the accuracy of model predictions is not necessary as long as they are sufficiently informative to enable a proper discrimination among candidate solutions. Yet, the accuracy can be exploited as a measure to identify which of the n surrogates provided to the internal optimizer as input approximates the problem's landscape better.

In the *lowest-error model selection* (ER) strategy, every application of the internal optimizer relies on the single, most-accurate surrogate model (out of the n models provided to this algorithm as input). This is determined on the basis of the (cross-validated) prediction errors observed during training (the *root mean squared error* is used here), prior to the invocation of the internal optimizer (Algorithm 1, Line 5). Note that the model selected (i.e., the one identified as the most accurate) may be different from one iteration of the SMBO process to the next. This allows multiple models to participate and contribute to the overall search process. Also, this behavior is consistent with the fact that models' accuracy may change over time; some models may be more effective than others at early stages, when the training set cardinality is small, but this may change as new training samples are collected throughout the search. This approach has been explored with success in previous studies (e.g., [16]).

2) AVERAGE OF PREDICTED VALUES

The *average of predicted values* (AV) ensemble strategy computes a final estimation of solution quality as the arithmetic mean of the values predicted by the given set of n surrogate models. This is equivalent to using a weighted aggregation (as explored, for example, in [33] and [37], and other works discussed in Section II-E), but using equal weights for all the models being combined. In this strategy, therefore, all the n ensemble members collaborate simultaneously to guide the search during every application of the internal optimizer; this contrasts with the ER strategy described above, where a single model (the most accurate one) is selected each time.

3) AVERAGE OF INDUCED RANKS

The last ensemble strategy, *average of induced ranks* (RK), is similar to the above-defined AV approach in that all the n models are simultaneously exploited to discriminate among candidate solutions during the internal optimization process. The RK strategy, however, averages the ranks induced by the predicted quality values of the solutions, rather than averaging their predicted qualities directly (as strategy AV does). That is, all candidate solutions are ranked independently based on the predictions of each individual ensemble member, such that the best solution is assigned rank 1, the second-best solution is assigned rank 2, and so on. Let $r_i(\mathbf{x})$ be the rank of solution $\mathbf{x}$ according to the predictions of the i-th surrogate model, $1 \leq i \leq n$; the average rank of $\mathbf{x}$ is given by:

$$R(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} r_i(\mathbf{x}) \tag{1}$$

The average rank of all solutions is computed and used to impose a final ordering between them, guiding the mating and survivor selection processes of the internal GA optimizer.

IV. EXPERIMENTS AND RESULTS

This section reports the results of our evaluation of surrogate-modeling strategies, centering on a specific industrial case study: the ESP problem (see Section II-C). First of all, Section IV-A discusses the experimental conditions, including the performance assessment criteria, reference approaches, and settings considered. Our initial experiments, presented in Section IV-B, involve the analysis and comparison of surrogate-modeling techniques at the individual level. Section IV-C covers the second part of our experiments, which concerns ensemble strategies allowing multiple promising models (as identified in Section IV-B) to be employed simultaneously. Then, Section IV-D contrasts the results of the best-performing individual models and ensemble strategies (as observed in Sections IV-B and IV-C), including also reference methods previously investigated in the specific context of the ESP problem. Finally, Section IV-E concludes this evaluation by analyzing how the use of surrogate models and particularly ensemble strategies impacts on computational efficiency.

A. EVALUATION SETUP

Our investigation of surrogate-modeling approaches in the context of the ESP problem starts by evaluating the suitability of a range of learning techniques. This involves eight distinct techniques, namely, AD, DT, KN, KR, NN, RB, RF, and SV, as defined in Section III-C. Then, our analysis centers on exploring the advantages of surrogate-model ensembles, considering three strategies that enable the integration of multiple surrogate models into the optimization process; namely, strategies ER, AV, and RK, introduced in Section III-D.

It is important to remark that all the experiments of this study consider the flexible SMBO framework whose functioning, design components, and parameter settings have already been specified in Section III. Below, we describe our performance assessment considerations as well as the approaches adopted as references during this evaluation.

1) PERFORMANCE ASSESSMENT

This paper is concerned with a specific, expensive optimization challenge: the ESP problem. In particular, our experiments investigate whether the use of different surrogate-modeling techniques and ensemble strategies allows us to discover better solutions to this problem. Accordingly, throughout our experiments, the main performance criterion used is solution quality, as given by the values scored for the real (expensive) objective function of this problem (to be minimized).

In Sections IV-B, IV-C, and IV-D, results are mostly presented in the form of convergence curves, reporting the best solution quality achieved so far along the SMBO process (at each iteration, every time a new solution is evaluated, Line 7 of Algorithm 1). This is complemented with plots summarizing the performance observed after a limited budget of 50 solution evaluations, and after exhausting the $E_{max} = 200$ evaluations adopted as stopping condition. We report the *area under the convergence curve* (AUCC) as an additional indicator to capture convergence behavior, with lower AUCC values indicating a deeper and/or faster convergence.

Section IV-E, on the other hand, focuses on the impact of the studied surrogate-modeling techniques on computational efficiency. We primarily assess convergence speed, which reflects on the number of objective function evaluations required to reach a certain solution quality. The reduction that individual and ensemble surrogate-model strategies achieve in the number of evaluations is measured in terms of *speedup* and how it translates into significant execution time savings.

Due to the stochastic nature of the methods analyzed, the results reported in all the cases correspond to statistics computed from a set of 20 independent repetitions of each experiment. The (non-parametric) *Mann-Whitney U test* is used for analyzing statistical significance, considering a significance level of $\alpha = 0.05$ and *Holm-Bonferroni* correction.

2) REFERENCE APPROACHES

The main (and mandatory) baseline against which the performance of SMBO approaches is evaluated in this study corresponds to the GA by itself; that is, to the same GA used as internal optimizer in our SMBO framework (see Algorithm 2 and its description in Section III-B), but without using surrogate models and always evaluating candidate solutions through the problem's original objective function. Considering this GA as our main benchmark allows us to assess the impact of introducing surrogate models (either individually or in the form of ensembles) on both solution quality and convergence speed. To enable a fair comparison, the GA is granted the same total budget of $E_{max} = 200$ effective evaluations as for the SMBO approaches. Therefore, we adopt parameter settings which are consistent with such a budget, namely, a population size of $N_P = 10$ and a total number of generations of $G_{max} = 20$ (all other parameters and operators of the GA remain the same as described in Section III-B).

Our comparison of the best-performing individual surrogate models and ensemble strategies, presented in Section IV-D, also includes two reference approaches that have been previously explored in the particular context of the ESP problem. The first approach is the *efficient global optimization* (EGO) method by Jones et al. [46]. EGO showed the best optimization performance in the experiments conducted recently by Rehbach et al. [21]. As discussed in Section II-D, Rehbach et al. proposed the incorporation of dimensionality reduction techniques to improve computational efficiency. However, this caused some decreases in solution quality, and the standard EGO (without enforcing dimensionality reduction) delivered the best results for the ESP problem [21]. EGO is thus an appropriate reference to consider in this study. The implementation of EGO available through the CEGO R package [47] is used here, such as reported in [21].

As the second reference approach, we consider the *mixed-variable ReLU-based surrogate modeling* (MVR) method proposed recently by Bliek et al. [25]. MVR has been particularly developed to tackle problems involving both continuous and discrete design parameters. Although the ESP problem features discrete, categorical variables only, the authors included five continuous variables (in addition to the original 49 variables, see Section II-C) for the purposes of testing. In this study, however, the original version of the ESP problem is always considered (with 49 discrete variables). We use the implementation of MVR which has been made available by its authors (https://github.com/lbliek/MVRSM).

B. ANALYSIS OF INDIVIDUAL SURROGATE MODELS

The computational costs associated with the ESP problem, as well as its black-box optimization nature, challenge our ability to make *a priori* decisions regarding the choice of suitable surrogate-modeling techniques. Therefore, we conduct a comparative analysis of eight popular approaches (see Section III-C): {AD, DT, KN, KR, NN, RB, RF, SV}. The baseline GA is also included in this analysis as a reference.

Some of the modeling techniques are known to offer certain robustness with respect to the nature of design variables, but others may find difficulties given the discrete, categorical variables of the ESP problem. Whilst an in-depth exploration of mechanisms to deal with this type of variable is beyond the scope of this paper, we aimed to identify conditions under which all models can have a reasonable operation, thus favoring a fair comparison. Hence, a preliminary evaluation was carried out to determine whether each model can handle the original encoding of variables, or if the use of an alternative binary (*one-hot*) encoding can boost their performance. The analysis presented here (and in subsequent subsections) considers the encoding that allows each individual modeling technique to obtain better results (the main findings of such a preliminary evaluation are summarized in the Appendix).

The results of our comparative analysis are presented in Figs. 4 and 5. First, we can observe very different convergence behaviors from Fig. 4. An interesting result is that the baseline GA displays a better and faster convergence

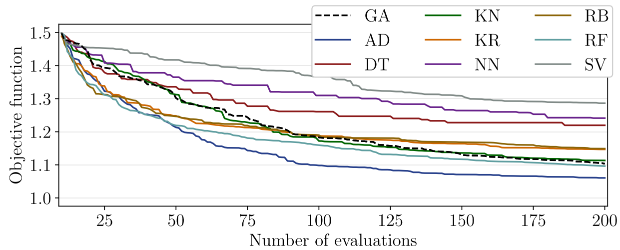


FIGURE 4. Convergence behavior exhibited by the eight surrogate-modeling techniques. Results of the baseline GA are included as a reference.

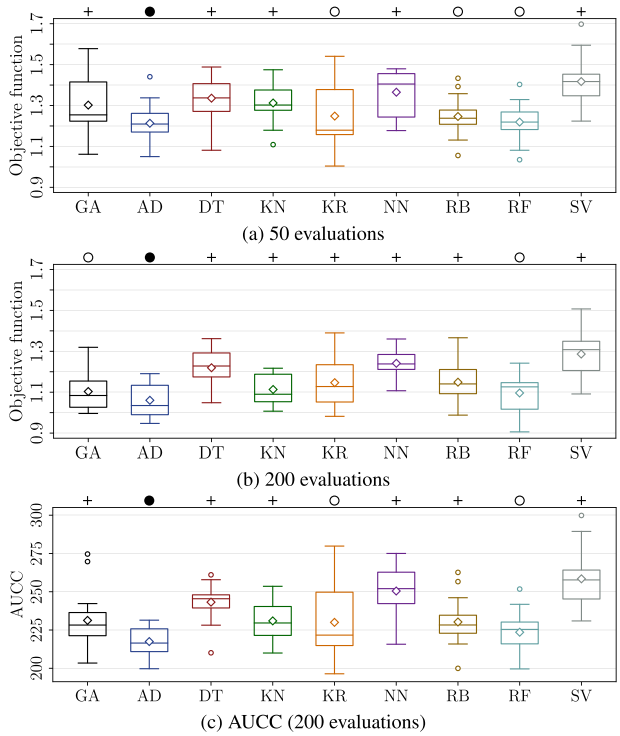


FIGURE 5. Performance of the eight surrogate-modeling techniques and the baseline GA. The plots report the objective values achieved after (a) 50 and (b) 200 solution evaluations, as well as the results for the (c) AUCC indicator. Marker • highlights the method with the best (lowest) average objective value; markers + and ∘ indicate, respectively, whether or not a statistically significant difference is observed with respect to this best-performing approach.

ability than SMBO approaches relying on some of the surrogate models, in particular SV, NN, and DT, which report the worst performances during this evaluation (as confirmed by the results of Fig. 5). Indeed, the average objective values reached by the GA at the end of the search (after 200 solution evaluations) are better (i.e., lower) than those obtained using most of the surrogate models. The only exceptions are AD and RF, achieving a deeper convergence in the end, but with no statistically significant differences over the GA, as seen from Fig. 5b. The slope of the convergence curve of the GA suggests that, if provided with a larger budget of evaluations, it would be able to improve solution quality even further. This confirms the effectiveness of the GA, which is our chosen optimizer integrated within the adopted SMBO framework.

Despite the competitive performance shown by the GA on its own, the incorporation of some surrogate models, namely, AD, KR, RB, and RF, has significantly accelerated convergence during the initial stages of the optimization process; refer, for example, to the results obtained after 50 evaluations in Fig. 5a. This increased convergence speed is particularly relevant when facing a very restrictive budget of solution evaluations. Out of these four approaches, AD and RF exhibit the most consistent trends of improvement throughout the search, being the best performers in this comparative analysis. It is noteworthy that, by themselves, AD and RF are ensemble learning techniques (as discussed in Section III-C). The fact that these approaches scored the best results in this test supports the relevance of ensemble methods in an SMBO setting, and encourages the investigation of more sophisticated ensembles further combining AD and RF with other different learning techniques (as explored in Section IV-C).

C. EVALUATION OF SURROGATE-MODEL ENSEMBLES

The three ensemble strategies described in Section III-D, ER, AV, and RK, enable the collaborative integration of a set of n surrogate models within the SMBO process. A clearly important aspect refers to the specific surrogate models which are chosen as the ensemble members. This section delves into such a key aspect, analyzing the impact of ensemble conformation on the performance of these strategies.

We define two different sets of surrogate models based on observations from their evaluation in Section IV-B. An ensemble size of $n = 4$ is considered in both cases:

$$\mathcal{A} = \{\text{AD, KN, KR, RF}\} \quad \mathcal{B} = \{\text{KR, RB, RF, SV}\}$$

On the one hand, set $\mathcal{A}$ consists of: AD and RF, the overall best performers; KR, which reports an accelerated convergence at early optimization stages; and KN, which scores the third best (average) performance at the end of the search. Thus, by involving only models providing competitive results at the individual level, set $\mathcal{A}$ may be expected to yield a competitive performance at the ensemble level as well. On the other hand, we are interested in investigating robustness, by evaluating whether the inclusion of low-performance models compromises the effectiveness of the ensemble strategies. Therefore, together with some well-performing surrogates (namely, KR, RB, and RF), set $\mathcal{B}$ includes the SV model, which shows the poorest performance in the comparative analysis of Section IV-B.

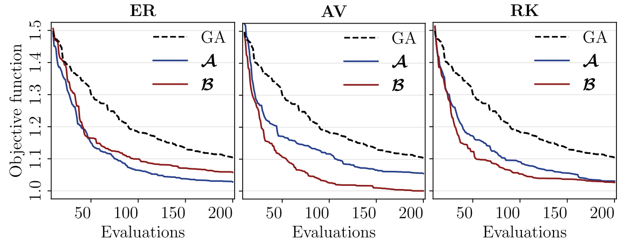


FIGURE 6. Convergence behavior of ensemble strategies ER, AV, and RK when using surrogate-model sets $\mathcal{A}$ and $\mathcal{B}$ (the baseline GA is shown as a reference).

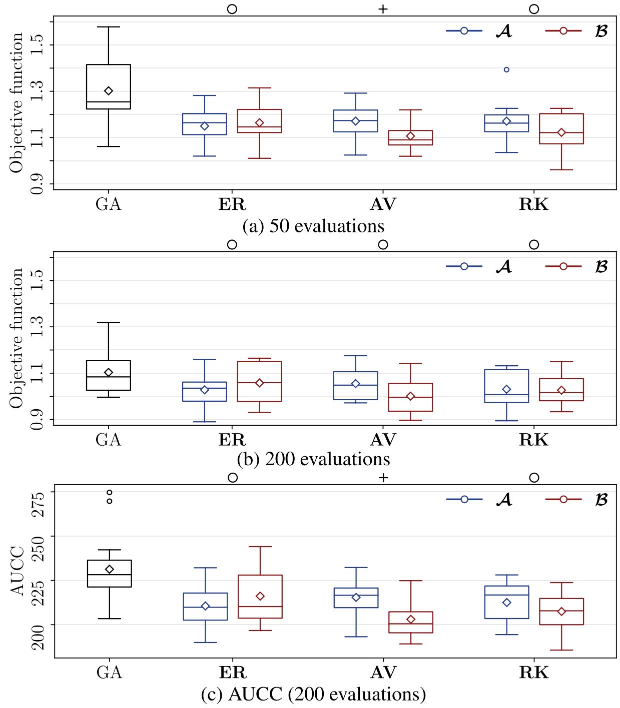


FIGURE 7. Contrasting the performance of ensemble strategies ER, AV, and RK when using sets $\mathcal{A}$ and $\mathcal{B}$ of surrogate models. Plots refer to the objective values achieved after (a) 50 and (b) 200 evaluations, and to the (c) AUCC indicator. At the top, the results of each strategy are independently marked either + or ◦, depending on whether or not a statistically significant difference is observed between sets $\mathcal{A}$ and $\mathcal{B}$. The baseline GA is included as a reference.

Figs. 6 and 7 summarize the performance of the six resulting combinations of ensemble strategies and surrogate-model sets. All six approaches report a remarkably superior performance in comparison to the baseline GA, both in terms of convergence speed and solution quality at the end of the search process. Strategy ER performs slightly better when using model set $\mathcal{A}$, but the differences with respect to the use of set $\mathcal{B}$ are not statistically significant. Contrary to what was anticipated, strategy AV (based on the aggregation of model predictions) obtains better results using set $\mathcal{B}$, in spite of the inclusion of the (poor-performing) SV model (with significant performance differences at 50 evaluations and regarding indicator AUCC, Figs. 7a and 7c, respectively). Likewise, model set $\mathcal{B}$ leads to a better performance of strategy RK (aggregating model-induced ranks), although no significant differences are observed with respect to the use of set $\mathcal{A}$.

The clear superiority of all six configurations analyzed with respect to the baseline GA highlights the suitability of the three ensemble strategies and the relevance of this type of approach in the particular context of our case study. Moreover, the ensemble strategies have shown some robustness to the combination of surrogate models of varying quality and reliability; this property could be enhanced by considering a weighting scheme for the ensemble members (such as in the methods discussed in Section II-E). The results of this section suggest that ensemble strategy AV, using model set $\mathcal{B}$, represents the best choice to address the ESP problem. The three ensemble strategies are further evaluated in Section IV-D through comparisons against the best-performing individual surrogate models and some references from the literature.

D. COMPARISON OF INDIVIDUAL SURROGATE MODELS, ENSEMBLE STRATEGIES, AND REFERENCE APPROACHES

After independently evaluating individual surrogate models and ensemble strategies in previous sections, here we contrast the results obtained by: (i) the three best-performing modeling techniques at the individual level, AD, RF, and KR; (ii) our three ensemble strategies, $\text{ER}_{\mathcal{A}}$, $\text{AV}_{\mathcal{B}}$, $\text{RK}_{\mathcal{B}}$, with subscripts specifying the surrogate-model set adopted in each case, according to the results of Section IV-C; and (iii) two reference methods from the literature, EGO and MVR. As before, the baseline GA is also included in this comparison as our main benchmark. The results are shown in Figs. 8 and 9.

Whilst the individual surrogate models increase convergence speed with respect to the baseline GA (as seen in Section IV-B), the three ensemble strategies and reference method EGO achieve a notably deeper convergence during the first 50 evaluations. As shown in Fig. 9a, strategy $\text{AV}_{\mathcal{B}}$ scores the best performance at these early stages, with statistically significant differences against all other approaches, except $\text{RK}_{\mathcal{B}}$. A similar trend is observed in Fig. 9b regarding indicator AUCC, whereas Fig. 9c highlights the three ensemble strategies as the best performers at the end of the search (after 200 evaluations), with no significant differences between them. Although all three strategies are found to be highly competitive, $\text{AV}_{\mathcal{B}}$ and $\text{RK}_{\mathcal{B}}$ exhibit a more accelerated initial convergence than $\text{ER}_{\mathcal{A}}$. This suggests that the aggregation of surrogate models, either by combining their predictions ($\text{AV}_{\mathcal{B}}$) or the ranks they induce ($\text{RK}_{\mathcal{B}}$), is more effective than model selection ($\text{ER}_{\mathcal{A}}$), which is consistent with previously reported findings [16].

An unexpected result is that of reference method MVR, which is found to be the less successful approach in this comparison. In contrast, EGO shows one of the most promising performances, being surpassed only by the ensemble strategies and competing closely with AD. An interesting aspect of EGO is that it uses kriging as the surrogate-modeling technique, the same technique that we evaluate within our SMBO framework and refer to as KR during these experiments.

Note, however, that EGO reports much better results than KR in this evaluation. The use of distinct optimization frameworks explains, to a certain degree, the performance variations between EGO and KR. More importantly, though, EGO employs the so-called *expected improvement* (EI) infill criterion (or acquisition function). At each iteration of our adopted SMBO framework, we search for a solution

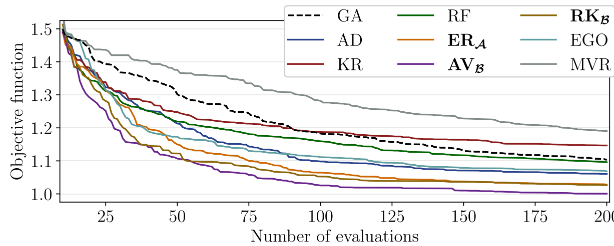


FIGURE 8. Convergence behavior of the best-performing individual surrogates (AD, KR, RF), ensemble strategies ($\text{ER}_{\mathcal{A}}$, $\text{AV}_{\mathcal{B}}$, $\text{RK}_{\mathcal{B}}$), and methods from the literature (EGO, MVR). The baseline GA is included as a reference.

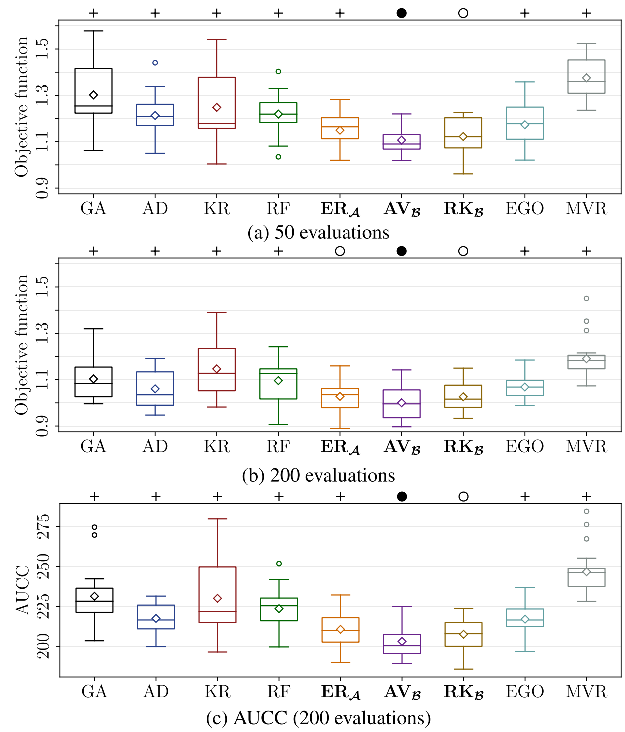


FIGURE 9. Comparison of selected surrogate models (AD, KR, RF), ensemble strategies ($\text{ER}_{\mathcal{A}}$, $\text{AV}_{\mathcal{B}}$, $\text{RK}_{\mathcal{B}}$), reference approaches (EGO, MVR), and the baseline GA. Plots report the results at (a) 50 and (b) 200 evaluations, and those for the (c) AUCC indicator. Marker • indicates the method with the best average objective value; + and ∘ indicate, respectively, whether or not a significant difference is seen with respect to this best-performing method.

optimizing the (predicted) objective function value, so that this new sample is evaluated and included in the training repository (Algorithm 1, Lines 6-8). Rather than focusing on predicted objective values only, criterion EI exploits additional information regarding model uncertainty to achieve a more suitable exploration/exploitation balance [48]. Thus, the consideration of this type of infill criteria seems to be a promising avenue towards improving the performance of the (individual and ensemble-based) surrogate-modeling approaches analyzed.

E. IMPACT ON COMPUTATIONAL EFFICIENCY

So far, our experiments have shown that the use of surrogate models, and particularly the simultaneous use of multiple surrogates through an ensemble strategy, has the potential to significantly improve the quality of the solutions discovered for the ESP problem. Moreover, our analysis has revealed that these approaches are able to provide a substantial increase in convergence speed, matching or even surpassing the solution quality reported by the baseline GA but without the need to exhaust the allocated budget of solution evaluations. This final part of our assessment pays further attention to the latter aspect, which impacts directly on computational efficiency and is therefore critical in expensive optimization.

First, we analyze how many solution evaluations are required by the studied approaches to reach a certain solution quality. More specifically, we take the GA's average final performance, that is, the average solution quality reported by the GA after exhausting the full budget of $E_{max} = 200$ evaluations, and determine the number of evaluations that model-based approaches need to carry out so that their average performance matches such a baseline. Then, we explore how these numbers of required evaluations translate into execution time estimations. These estimations assume that evaluating solution quality takes eight hours by means of the ESP problem's objective function (refer to Section II-C for details), as reported in [21].[3] Finally, the observed reductions in the number of evaluations and running times are captured and analyzed in terms of speedup; if we let α and σ be the number of evaluations consumed by the baseline GA and a given alternative approach, respectively, we compute the speedup offered by the alternative approach as α/σ (speedup can equivalently be computed from the estimated times).

The results obtained from the three aforementioned perspectives (namely, number of evaluations, execution time, and speedup) are separately presented in Fig. 10. Similar to Section IV-D, results are shown for the best-performing individual modeling techniques (AD and RF), ensemble strategies ($ER_{\mathcal{A}}$, $AV_{\mathcal{B}}$, and $RK_{\mathcal{B}}$), and reference approaches (the baseline GA and EGO). Notice that we have

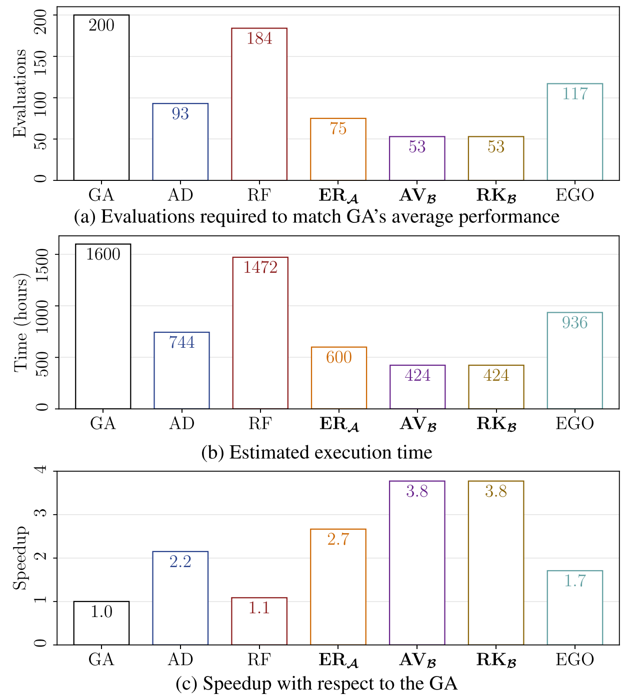


FIGURE 10. Results of selected surrogate models (AD, RF), ensemble strategies ($ER_{\mathcal{A}}$, $AV_{\mathcal{B}}$, $RK_{\mathcal{B}}$), and references (EGO) in terms of: (a) reduction of the number of evaluations; (b) estimated time savings; and (c) speedup. All three aspects are calculated by taking the GA's average final performance (after 200 solution evaluations) as the baseline.

excluded model KR and reference method MVR (originally included in Section IV-D), as the average performance of these approaches does not reach the baseline defined by the GA within the budget of $E_{max} = 200$ evaluations (see Figs. 8 and 9).

As can be seen from Fig. 10a, all six SMBO approaches exhibit a reduction in the number of evaluations with respect to the GA. Any reduction is meaningful, given the high cost of evaluations that characterizes expensive optimization scenarios. However, once again, our results confirm the superiority of ensemble approaches over the use of individual surrogate models. The best performers are ensembles $AV_{\mathcal{B}}$ and $RK_{\mathcal{B}}$, reducing the number of evaluations by almost 75%. Ensemble $ER_{\mathcal{A}}$ also shows a competitive performance, consuming only 75 evaluations to reach a solution quality that is at least as good as that produced by the GA after 200 evaluations.

The significance of the above reductions becomes more evident when they are analyzed from the perspective of execution time, as reported in Fig. 10b. Consider, for example, that the solution quality that the baseline GA would attain after almost 67 days (1,600 hours) could be delivered within 18 days (424 hours) by using either ensemble $AV_{\mathcal{B}}$ or ensemble $RK_{\mathcal{B}}$, leading to an almost 4-fold speedup in convergence according to Fig. 10c. This translates into clear, tangible benefits that, in addition to time savings, could

[3] As explained in Section II-C, we consider a reduced and less computationally expensive approximation of the ESP problem, which has made the extensive analyses of this paper possible. Considering the evaluation times reported in the literature for the original version of the ESP problem allows us to present more reliable (and realistic) execution time estimations.

involve other potential positive impacts (such as the economic impact).

V. CONCLUSION

We explored surrogate model-based approaches in the context of a real-world problem involving the design of an electrostatic precipitator. This is a relevant industrial application, posing a computationally expensive, black-box, combinatorial optimization challenge. Given the inherent difficulty of *a priori* identifying the most appropriate modeling technique under such conditions, we conducted a comparative analysis involving eight popular choices. To our knowledge, this is the first comparison of this sort with a focus on this specific application. Furthermore, we explored, for the first time in the context of this problem, surrogate-model ensembles. We evaluated three strategies allowing the collaborative integration of multiple surrogate models into the optimization process.

Our analysis of (individual) modeling techniques revealed that Adaptive Boosting (AdaBoost) and Random Forest were the most effective ones. Nevertheless, performance was further improved by using surrogate-model ensembles. Especially, a strategy based on the aggregation of multiple surrogates outperformed all individual models and some references from the literature. Not only did ensembles score better results at the end of the optimization process, but they also reported an almost 4-fold speedup in convergence. This is of particular importance when facing very restrictive budgets of solution evaluations due to the high computational costs they imply.

Our findings confirm that the conceptual advantages of ensembles can translate into meaningful gains in practice. Despite the promising results obtained, only basic, intuitive ensemble strategies were considered. Several potential pathways to further enhance these results can be identified, for example: devising more sophisticated model-integration mechanisms; incorporating weighting schemes to prioritize ensemble members according to their reliability (or other criteria); exploring the notion of locality of surrogate models; considering alternative infill (model management) criteria; and evaluating the impact of varying the size and specific elements of the ensembles. Some of these topics have reported encouraging results in other domains; we will devote part of our future work to exploring these pathways in the context of our case study. Finally, an inherent limitation of this study relates to the potential lack of generality of our observations, as it centers on a particular application. The insights gained motivate us to replicate this analysis in other scenarios, which will contribute to overcoming the above limitation.

APPENDIX A
IMPACT OF VARIABLE ENCODING

Here, we summarize our findings regarding the impact of variable encoding on the effectiveness of the eight (individual) surrogate-modeling techniques. In particular, we compare two alternative encodings: (i) the original discrete, categorical encoding of the ESP problem; and (ii) the binary, one-hot encoding, which maps each of the $S = 49$ decision variables to an eight-bit binary string (with each bit representing one of the $P = 8$ different values that a variable can assume).

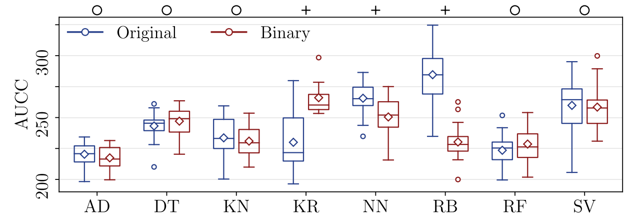


FIGURE 11. Impact of variable encoding on the performance of the eight studied surrogate-modeling techniques. The original and binary (one-hot) encodings are compared. Markers + and ∘ at the top indicate, respectively, whether or not a statistically significant difference is observed for each individual comparison.

As shown in Fig. 11, changing variable encoding does not lead to statistically significant differences in the performance of five model types (AD, DT, KN, RF, and SV). However, highly significant differences can be observed for the remaining three techniques (KR, NN, and RB). On the one hand, KR is negatively affected by the binary encoding, which might be explained by the increase in dimensionality. On the other hand, the binary encoding clearly improves the performance of NN and RB, highlighting the inability of these techniques to handle the nature of the ESP problem's variables directly.

The experiments of Section IV consider the encoding offering the best results for each technique. Three models use the original encoding, {DT, KR, RF}, and the remaining five models use the binary encoding, {AD, KN, NN, RB, SV}.

REFERENCES

[1] A. I. Forrester, A. Sóbester, and A. J. Keane, *Engineering Design via Surrogate Modelling*. Hoboken, NJ, USA: Wiley, 2008, doi: 10.1002/9780470770801.
[2] L. E. Zerpa, N. V. Queipo, S. Pintos, and J.-L. Salager, "An optimization methodology of alkaline-surfactant-polymer flooding processes using field scale numerical simulation and multiple surrogates," *J. Petroleum Sci. Eng.*, vol. 47, no. 3, pp. 197–208, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0920410505000379
[3] A. L. Marsden, M. Wang, J. E. Dennis, and P. Moin, "Trailing-edge noise reduction using derivative-free optimization and large-eddy simulation," *J. Fluid Mech.*, vol. 572, pp. 13–36, Feb. 2007.
[4] J. Branke, Md. Asafuddoula, K. S. Bhattacharjee, and T. Ray, "Efficient use of partially converged simulations in evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 1, pp. 52–64, Feb. 2017.
[5] A. Kenny, T. Ray, and H. K. Singh, "An iterative two-stage multifidelity optimization algorithm for computationally expensive problems," *IEEE Trans. Evol. Comput.*, vol. 27, no. 3, pp. 520–534, Jun. 2022.
[6] R. J. Preen and L. Bull, "Toward the coevolution of novel vertical-axis wind turbines," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 284–294, Apr. 2015.
[7] G. Kochanski, D. Golovin, J. Karro, B. Solnik, S. Moitra, and D. Sculley, "Bayesian optimization for a better dessert," in *Proc. NIPS Workshop Bayesian Optim.*, Long Beach, CA, USA, 2017, pp. 1–10.
[8] Y. Bi, B. Xue, and M. Zhang, "Instance selection-based surrogate-assisted genetic programming for feature learning in image classification," *IEEE Trans. Cybern.*, vol. 53, no. 2, pp. 1118–1132, Feb. 2023.

[9] S. Liu, H. Wang, W. Peng, and W. Yao, "A surrogate-assisted evolutionary feature selection algorithm with parallel random grouping for high-dimensional classification," *IEEE Trans. Evol. Comput.*, vol. 26, no. 5, pp. 1087–1101, Oct. 2022.
[10] T. Bartz-Beielstein and M. Zaefferer, "Model-based methods for continuous and discrete global optimization," *Appl. Soft Comput.*, vol. 55, pp. 154–167, Jun. 2017.
[11] P. Jiang, Q. Zhou, and X. Shao, *Surrogate-Model-Based Design and Optimization*. Singapore: Springer, 2020, pp. 135–236, doi: 10.1007/978-981-15-0731-1_7.
[12] R. Alizadeh, J. K. Allen, and F. Mistree, "Managing computational complexity using surrogate models: A critical review," *Res. Eng. Design*, vol. 31, no. 3, pp. 275–298, Jul. 2020, doi: 10.1007/s00163-020-00336-7.
[13] J. Stork, A. E. Eiben, and T. Bartz-Beielstein, "A new taxonomy of global optimization algorithms," *Natural Comput.*, vol. 21, no. 2, pp. 219–242, Jun. 2022, doi: 10.1007/s11047-020-09820-4.
[14] A. I. J. Forrester and A. J. Keane, "Recent advances in surrogate-based optimization," *Prog. Aerosp. Sci.*, vol. 45, nos. 1–3, pp. 50–79, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0376042108000766
[15] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.
[16] T. Goel, R. T. Haftka, W. Shyy, and N. V. Queipo, "Ensemble of surrogates," *Structural Multidisciplinary Optim.*, vol. 33, no. 3, pp. 199–216, Jan. 2007, doi: 10.1007/s00158-006-0051-9.
[17] M. N. Le, Y. S. Ong, S. Menzel, Y. Jin, and B. Sendhoff, "Evolution by adapting surrogates," *Evol. Comput.*, vol. 21, no. 2, pp. 313–340, May 2013, doi: 10.1162/EVCO_a_00079.
[18] C. Sun, Y. Jin, J. Zeng, and Y. Yu, "A two-layer surrogate-assisted particle swarm optimization algorithm," *Soft Comput.*, vol. 19, no. 6, pp. 1461–1475, Jun. 2015, doi: 10.1007/s00500-014-1283-z.
[19] P. Liao, C. Sun, G. Zhang, and Y. Jin, "Multi-surrogate multi-tasking optimization of expensive problems," *Knowledge-Based Syst.*, vol. 205, Oct. 2020, Art. no. 106262. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705120304536
[20] F. Rehbach, M. Zaefferer, J. Stork, and T. Bartz-Beielstein, "Comparison of parallel surrogate-assisted optimization approaches," in *Proc. Genetic Evol. Comput. Conf.* New York, NY, USA: Association for Computing Machinery, 2018, pp. 1348–1355, doi: 10.1145/3205455.3205587.
[21] F. Rehbach, L. Gentile, and T. Bartz-Beielstein, "Variable reduction for surrogate-based optimization," in *Proc. Genetic Evol. Comput. Conf.* New York, NY, USA: Association for Computing Machinery, Jun. 2020, pp. 1177–1185, doi: 10.1145/3377930.3390195.
[22] S. Ahmad, K. University, and J. Smail, "The optimum operation conditions for electrostatic precipitator (ESP) in particulate emitter industries," in *Proc. Int. Conf. Pure Appl. Sci.*, Koysinjaq, Irak, Aug. 2018, pp. 23–24.
[23] S. Alarie, C. Audet, A. E. Gheribi, M. Kokkolaras, and S. Le Digabel, "Two decades of blackbox optimization applications," *EURO J. Comput. Optim.*, vol. 9, 2021, Art. no. 100011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2192440621001386
[24] R. Karlsson, L. Bliek, S. Verwer, and M. de Weerdt, "Continuous surrogate-based optimization algorithms are well-suited for expensive discrete problems," in *Artificial Intelligence and Machine Learning*, M. Baratchi, L. Cao, W. A. Kosters, J. Lijffijt, J. N. van Rijn, and F. W. Takes, Eds. Cham, Switzerland: Springer, 2021, pp. 48–63.
[25] L. Bliek, A. Guijt, S. Verwer, and M. de Weerdt, "Black-box mixed-variable optimisation using a surrogate model that satisfies integer constraints," in *Proc. Genetic Evol. Comput. Conf. Companion*. New York, NY, USA: Association for Computing Machinery, Jul. 2021, pp. 1851–1859, doi: 10.1145/3449726.3463136.
[26] A. Mizuno, "Electrostatic precipitation," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 7, no. 5, pp. 615–624, Jun. 2000.
[27] F. Carastro, J. Clare, A. Goodman, P. Wheeler, J. Leach, and T. Hosking, "A 100kv switch mode series resonant power supply for industrial electrostatic precipitators," in *Proc. IEEE Int. Power Modulator High Voltage Conf.*, 2010, pp. 485–488.
[28] K. R. Parker, *Why an Electrostatic Precipitator?*. Dordrecht, The Netherlands: Springer, 1997, pp. 1–10, doi: 10.1007/978-94-009-1553-4_1.
[29] X.-L. Ye, Y.-B. Su, B.-Y. Guo, and A.-B. Yu, "Multi-scale simulation of the gas flow through electrostatic precipitators," *Appl. Math. Model.*, vol. 40, nos. 21–22, pp. 9514–9526, Nov. 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0307904X16303341
[30] F. Rehbach, L. Gentile, and T. Bartz-Beielstein, "Feature selection for surrogate model-based optimization," in *Proc. Genetic Evol. Comput. Conf. Companion*. New York, NY, USA: Association for Computing Machinery, Jul. 2019, pp. 399–400, doi: 10.1145/3319619.3322020.
[31] D. Lim, Y. Jin, Y.-S. Ong, and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 329–355, Jun. 2010.
[32] J. Stork, M. Friese, M. Zaefferer, T. Bartz-Beielstein, A. Fischbach, B. Breiderhoff, B. Naujoks, and T. Tušar, *Open Issues in Surrogate-Assisted Optimization*. Cham, Switzerland: Springer, 2020, pp. 225–244, doi: 10.1007/978-3-030-18764-4_10.
[33] M. Friese, T. Bartz-Beielstein, T. Bäck, B. Naujoks, and M. Emmerich, "Weighted ensembles in model-based global optimization," in *Proc. AIP Conf.*, no. 1, 2019, Art. no. 020003. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/1.5089970
[34] D. Lim, Y.-S. Ong, Y. Jin, and B. Sendhoff, "A study on metamodeling techniques, ensembles, and multi-surrogates in evolutionary computation," in *Proc. 9th Annu. Conf. Genetic Evol. Comput.* New York, NY, USA: Association for Computing Machinery, Jul. 2007, pp. 1288–1295, doi: 10.1145/1276958.1277203.
[35] J.-Y. Li, Z.-H. Zhan, C. Wang, H. Jin, and J. Zhang, "Boosting data-driven evolutionary algorithm with localized data generation," *IEEE Trans. Evol. Comput.*, vol. 24, no. 5, pp. 923–937, Oct. 2020.
[36] E. Sanchez, S. Pintos, and N. V. Queipo, "Toward an optimal ensemble of kernel-based approximations with engineering applications," *Structural Multidisciplinary Optim.*, vol. 36, no. 3, pp. 247–261, Sep. 2008, doi: 10.1007/s00158-007-0159-6.
[37] E. Acar and M. Rais-Rohani, "Ensemble of metamodels with optimized weight factors," *Structural Multidisciplinary Optim.*, vol. 37, no. 3, pp. 279–294, Jan. 2009, doi: 10.1007/s00158-008-0230-y.
[38] G. Xu, H. Wei, J. Wang, X. Chen, and B. Zhu, "A local weighted linear regression (LWLR) ensemble of surrogate models based on stacking strategy: Application to hydrodynamic response prediction for submerged floating tunnel (SFT)," *Appl. Ocean Res.*, vol. 125, Aug. 2022, Art. no. 103228. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141118722001663
[39] Y. Tenne and S. W. Armfield, "A versatile surrogate-assisted memetic algorithm for optimization of computationally expensive functions and its engineering applications," in *Success in Evolutionary Computation*. Berlin, Germany: Springer, 2008, pp. 43–72, doi: 10.1007/978-3-540-76286-7_3.
[40] H. Wang, Y. Jin, and J. Doherty, "Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2664–2677, Sep. 2017.
[41] H. Yu, Y. Tan, J. Zeng, C. Sun, and Y. Jin, "Surrogate-assisted hierarchical particle swarm optimization," *Inf. Sci.*, vols. 454–455, pp. 59–72, Jul. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025518303293
[42] J. Lu, B. Li, and Y. Jin, "An evolution strategy assisted by an ensemble of local Gaussian process models," in *Proc. 15th Annu. Conf. Genetic Evol. Comput.* New York, NY, USA: Association for Computing Machinery, 2013, pp. 447–454, doi: 10.1145/2463372.2463425.
[43] D. Guo, Y. Jin, J. Ding, and T. Chai, "Heterogeneous ensemble-based infill criterion for evolutionary multiobjective optimization of expensive problems," *IEEE Trans. Cybern.*, vol. 49, no. 3, pp. 1012–1025, Mar. 2019.
[44] Z. Zhou, Y. S. Ong, M. H. Lim, and B. S. Lee, "Memetic algorithm using multi-surrogates for computationally expensive optimization problems," *Soft Comput.*, vol. 11, no. 10, pp. 957–971, Aug. 2007, doi: 10.1007/s00500-006-0145-8.
[45] Z. Zhang, B. Liu, Y. Yu, and Q. S. Cheng, "A microwave filter yield optimization method based on off-line surrogate model-assisted evolutionary algorithm," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 6, pp. 2925–2934, Jun. 2022.
[46] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, no. 4, pp. 455–492, 1998, doi: 10.1023/A:1008306431147.

[47] M. Zaefferer, J. Stork, M. Friese, A. Fischbach, B. Naujoks, and T. Bartz-Beielstein, ''Efficient global optimization for combinatorial problems,'' in *Proc. Annu. Conf. Genetic Evol. Comput.* New York, NY, USA: Association for Computing Machinery, 2014, pp. 871–878, doi: 10.1145/2576768.2598282.
[48] D. Zhan and H. Xing, ''Expected improvement for expensive optimization: A review,'' *J. Global Optim.*, vol. 78, no. 3, pp. 507–544, Nov. 2020, doi: 10.1007/s10898-020-00923-x.

MARIO GARZA-FABRE received the M.Sc. and Ph.D. degrees in computer science from the Center for Research and Advanced Studies (Cinvestav), Mexico, in 2009 and 2014, respectively. From 2015 to 2018, he was a Research Associate at the University of Manchester and Liverpool John Moores University, U.K. He joined the Cinvestav Campus Tamaulipas in 2018, where he currently holds the position as an Associate Professor of optimization and computational intelligence. His main research interests include analysis and design of heuristic optimization techniques, and their application to problems from diverse areas such as bioinformatics, data mining and machine learning, transportation, and communications.

NICOLÁS CORTÉS-GARCÍA received the B.S. degree in mechanical engineering from Metropolitan Autonomous University, Campus Azcapotzalco, in 2018, and the M.Sc. degree in engineering and computational technologies from the Center for Research and Advanced Studies (Cinvestav), Campus Tamaulipas, Mexico, in 2021. He is currently working as a Software Engineer in the private sector. His main research interest includes evolutionary computing and its application to challenging optimization problems.

HIRAM GALEANA-ZAPIÉN (Member, IEEE) received the B.S. degree in electronics and telecommunications engineering from the Autonomous University of Nuevo Leon, in 2001, and the Ph.D. degree in telecommunications engineering from the Polytechnic University of Catalonia, Barcelona, in 2011. From 2000 to 2004, he worked as a Network Provisioning Engineer for a Mexican Internet and data provider. Since 2011, he has been a Researcher with the Center for Research and Advanced Studies (Cinvestav), Tamaulipas, Mexico. His research interests include resource management, on-device/federated learning, and m-health, with applications to cellular communications, mobile computing, and the Internet of Things.

RICARDO LANDA received the M.Sc. and Ph.D. degrees in electrical engineering from the Center for Research and Advanced Studies (Cinvestav), Mexico, in 2002 and 2007, respectively. He was a Research Fellow at the School of Computer Science, University of Birmingham, U.K., from 2008 to 2009. He has been a Professor with Cinvestav, since 2009. He has worked on algorithms for constraint handling and multiobjective optimization based on differential evolution, evolutionary programming, and cultural algorithms. His research interests include mainly large scale optimization and both single and multiobjective.