

Advancing surrogate-based optimization of time-expensive environmental problems through adaptive multi-model search

Spyridon Tsattalios^{*}, Ioannis Tsoukalas, Panagiotis Dimas, Panagiotis Kossieris, Andreas Efstratiadis, Christos Makropoulos

Department of Water Resources and Environmental Engineering, National Technical University of Athens, Heron Polytechniou 5, GR-157 80, Zographou, Greece

ARTICLE INFO

Handling Editor: Daniel P Ames

Keywords:

Surrogate modeling
Machine learning
High-dimensional expensive black-box (HEB) problems
Evolutionary annealing-simplex
Test functions
Hydraulic design

ABSTRACT

Complex environmental optimization problems often require computationally expensive simulation models to assess candidate solutions. However, the complexity of response surfaces necessitates multiple such assessments and thus renders the search procedure too laborious. Surrogate-based optimization is a powerful approach for accelerating convergence towards promising solutions. Here we introduce the Adaptive Multi-Surrogate Enhanced Evolutionary Annealing Simplex (AMSEEAS) algorithm, as an extension of its precursor SEEAS, which is a single-surrogate-based optimization method. AMSEEAS exploits the strengths of multiple surrogate models that are combined via a roulette-type mechanism, for selecting a specific metamodel to be activated in every iteration. AMSEEAS proves its robustness and efficiency via extensive benchmarking against SEEAS and other state-of-the-art surrogate-based global optimization methods in both theoretical mathematical problems and in a computationally demanding real-world hydraulic design application. The latter seeks for cost-effective sizing of levees along a drainage channel to minimize flood inundation, calculated by the time-expensive hydrodynamic model HEC-RAS.

1. Introduction

Simulation models of detailed spatial and temporal resolution have a pivotal role in environmental sciences, also gaining increasing popularity in engineering practice. Such models provide the capability to represent complex physical phenomena, accounting for the spatiotemporal dynamics of all processes of interest, as well to describe their interactions with infrastructures and societal factors. Their utility is further enhanced when these are coupled with optimization methods (Maier et al., 2014). At a conceptual level, combined simulation-optimization schemes (Tsoukalas et al., 2016) can be employed to address both decision-making applications (e.g., optimal design, planning, management and real-time control of environmental systems) and inverse modeling problems as well, aiming to identify optimal model configurations so that the observed responses are faithfully represented. As their name suggests, these use simulation models to evaluate the system's performance, expressed in terms of an objective function of a nonlinear (global) optimization problem, with no analytical solution or derivative information.

The literature is particularly rich in such efforts and advances, which

have been summarized in the review works of Labadie (2004), Fowler et al. (2008), Nicklow et al. (2010), Reed et al. (2013), Ahmad et al. (2014) and Kumar and Yadav (2022), which emphasize on water resources management, as well as by Duan (2003) and Efstratiadis and Koutsoyiannis (2010), which focus on hydrological calibration.

The major obstacle encountered in model-based optimization problems is the required computational workload, which is dictated by the computational cost (e.g., time) of the underlying simulation model. State-of-the-art simulation models have the ability to describe the peculiarities of environmental systems with great accuracy and detail (e.g., in terms of geometry, boundary conditions and spatiotemporal dynamics), yet this comes at a price. Another common category of time-demanding models involves stochastic simulation schemes that are driven by long synthetic data, in order to quantify their probabilistic performance with satisfactory accuracy. In all these cases, the computational time for a single model execution may require from a few minutes up to several hours. On the other hand, depending on the problem dimensionality and the irregularity of the response surface, a typical global optimization algorithm may need to evaluate the objective function (and hence call the simulation model) thousands of times, in

^{*} Corresponding author.

E-mail address: stsattalios@mail.ntua.gr (S. Tsattalios).

order to converge to a satisfactory solution. These two issues combined, result in termination of the optimization procedure after days or even weeks, which makes the optimization task infeasible and, sometimes, even prohibitive.

Among the different strategies proposed by the research community (e.g., [Razavi et al., 2010](#)) to deal with the time issue, including parallel computing, computationally efficient search algorithms, opportunistic avoidance of model evaluations, a particularly interesting one with proven effectiveness, is the use of surrogate modeling techniques. Surrogate models (also known as meta-models) offer an elegant software-based solution, where low-computational-cost approximation models are used to represent the actual (and time-expensive) simulation model's response surface and thus guide the optimization procedure. In particular, the surrogates aim to aid the search procedure, by replacing the actual simulation model to some extent and proposing promising solutions that will possibly lead to a much faster convergence of the optimization. Under this premise, the simulation procedure is called in limited cases, in order to evaluate the objective function wherever suggested by the surrogate model.

From a historical perspective, one of the first works that popularized such approaches is attributed to [Jones et al. \(1998\)](#), who introduced the so-called Efficient Global Optimization (EGO). EGO embeds a Kriging (i. e., Gaussian process) method as a surrogate model to the core of the optimization procedure. Among others, this has the role of locating potentially good solutions that are *worth evaluating* through the actual simulation model.

After the seminal publication of EGO, several optimization schemes incorporated the idea of using surrogate modeling techniques, thus resulting to a vast list of algorithms, where numerous alternative Machine Learning models are employed as surrogate models, such as Polynomials, Radial Basis Functions (RBFs), Random Forests (RFs), Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), etc. (e.g., [Müller et al., 2013](#); [Liu et al., 2014](#); [Golzari et al., 2015](#); [Mallipeddi and Lee, 2015](#); [Awad et al., 2018](#)). The use of surrogate-assisted approaches spans over multiple types of optimization problems, also including discrete, constrained and multiobjective

optimization. Emphasis was also given to time-demanding problems with many control variables, also referred to as *high-dimensional expensive black-box* (HEB) problems. Running advances for HEB problems include the so-called Knowledge Transfer assisted Efficient Global Optimization (KT-EGO) algorithm, which extends the classical version of EGO to handle high (i.e., >20 variables) dimensions ([Wang et al., 2022](#)). Other recently published methods aim at providing hybrid schemes that combine different search and/or surrogate strategies. For instance, [Dong et al. \(2018a\)](#) introduced the Multi-surrogate-based Differential Evolution with Multi-start Exploration (MDEME) algorithm, which uses Differential Evolution, enhanced by three surrogate models (i.e., Kriging, Radial Basis Function, Quadratic Polynomial Response), while [Pan et al. \(2021\)](#) proposed the so-called Surrogate-Assisted Hybrid Optimization (SAHO), that employs two different optimization techniques (i.e., Teaching-Learning-Based Optimization, Differential Evolution), combined with RBF metamodels.

The idea of surrogate-based optimization has also found fertile ground in the domain of water resources and the environment (e.g., [Wang et al., 2014](#); [Tsoukalas and Makropoulos, 2015a](#); [Tsoukalas and Makropoulos, 2015b](#); [Shaw et al., 2017](#); [Xia et al., 2021](#); [Lu et al., 2022](#)), starting from the pioneering work by [Regis and Shoemaker \(2004\)](#). Since then, this approach managed to address real-world optimization problems of significant complexity. For instance, [Yazdi and Salehi Neyshabouri \(2014\)](#) introduced a framework to solve high-dimensional problems for optimizing flood control detention dams. [Wu et al. \(2015\)](#) proposed the so-called Surrogate-based Optimization for Integrated surface water-groundwater Modeling (SOIM) algorithm for water management optimization problems. [Xi et al. \(2017\)](#) proposed a surrogate-assisted approach to efficiently calibrate agricultural-hydrological models on a limited budget. Recently, [Sun et al. \(2022\)](#) developed the so-called Multi-Objective Adaptive Surrogate Modelling-based Optimization for Constrained Hybrid problems (MO-ASMOCH), which is designed to handle problems consisting of both continuous and discrete control variables.

The focus of this research is to provide improved surrogate-based solutions for handling time-demanding global optimization problems (where term “global” is used to denote nonlinear, single-objective, unconstrained optimization problems with continuous variables). This kind of problems is very common in the domain of water resources and the environment, where the objective function is typically defined through a computationally expensive simulation model and the resulting response surface is, in general, multimodal. For convenience, we also consider that the problem is configured in single-objective terms, meaning that the system's performance is expressed via an overall scalar metric, and, more specifically, by means of a “cost” function to minimize. We highlight that this overall metric may aggregate several criteria (potentially, conflicting), and embed as well (few) external constraints, by means of penalties. On the other hand, the vast majority of constraints that are associated with internal modeling procedures (e. g., description of physical processes), are exclusively handled through the underlying simulation model, thus the optimization problem is by definition formalized as unconstrained. Regarding the search space (also known as decision or feasible space), it is formalized as a hypervolume, by assigning lower and upper boundaries to the problem's variables (according to the problem type, these may be referred to as control variables, decision variables, design variables or parameters).

[Table 1](#) contains a list of state-of-the-art (i.e., published during the last decade) surrogate-based algorithms for the case of our interest, i.e., global optimization. It is interesting to notice that very few of them are publicly available, while only three utilize more than one surrogates across the exploration-exploitation procedure. In an attempt to fill this gap, we introduce and provide as open-source software, in Python environment, the so-called Adaptive Multi-Surrogate Enhanced Evolutionary Annealing Simplex (AMSEEAS) algorithm. Its key novelty is the use of multiple surrogates that cooperate to enable significant improvements across the search space. As the search evolves, the most

Table 1

State-of-the-art surrogate-based algorithms for unconstrained single-objective optimization (publicly available algorithms are highlighted with bold).

No	Algorithm	Publication	Multiple Surrogates	Publicly Available
1	KT-EGO	Wang et al. (2022)	Yes	Yes
2	SATLBO	Dong et al. (2021)	No	No
3	SAHO	Pan et al. (2021)	No	No
4	PODS	Xia et al. (2021)	No	Yes
5	-	Müller (2020)	No	No
6	TURBO	Eriksson et al. (2019)	No	Yes
7	IDEASM	Awad et al. (2018)	No	No
8	MDEME	Dong et al. (2018a)	Yes	No
9	HSOSR	Dong et al. (2018b)	Yes	No
10	ASMO-PODE	Gong and Duan (2017)	No	No
11	MSSR	Dong et al. (2016)	No	No
12	SOP	Kritykiernie et al. (2016)	No	Yes
13	SEEAS	Tsoukalas et al. (2016)	No	Yes
14	GSAMP	Dong et al. (2015)	No	No
15	AMGO	Jie et al. (2015)	No	No
16	ESMDE	Mallipeddi and Lee (2015)	No	No
17	SOIM	Wu et al. (2015)	No	No
18	GPME	Liu et al. (2014)	No	No
19	OPUS-RBF	Regis (2014)	No	No
20	ASMO	Wang et al. (2014)	No	No
21	LHO-RBFNN	Yao et al. (2014)	No	No
22	DYCORS	Regis and Shoemaker (2013)	No	Yes
23	SBPSO	Tang et al. (2013)	No	No

effective surrogates are applied more frequently, on the basis of a *self-adaptive probabilistic selection* scheme. AMSEEAS builds upon two existing optimization schemes, i.e., the Surrogate-Enhanced Evolutionary Annealing Simplex (SEEAS) by Tsoukalas et al. (2016) and the Evolutionary Annealing Simplex (EAS) by Efstratiadis and Koutsogiannis (2002).

A brief overview of EAS and SEEAS, and a more detailed description of AMSEEAS, are given in Section 2. To evaluate its effectiveness and efficiency, the proposed algorithm is thoroughly benchmarked against its forerunner (i.e., SEEAS), as well as other state-of-the-art surrogate-based global optimization algorithms. The comparison is realized, initially, via testing all algorithms on a set of mathematical test functions, with complex response surfaces and multiple local optima (Section 3). Furthermore, to evaluate the proposed algorithm's performance on a real-world problem of significant difficulty, AMSEEAS is tested on a hydraulic design problem, where the evaluation of the cost function requires the use of a time-expensive hydrodynamic model (Section 4). The overall analyses illustrate the advantages of AMSEEAS in terms of providing systematically better solutions under a limited computational budget.

2. Optimization methodology

2.1. Evolutionary Annealing Simplex

EAS is a heuristic global optimization algorithm, developed by Efstratiadis and Koutsogiannis (2002). Its main rationale is finding an effective way to combine the strengths of the downhill-simplex local optimization method (Nelder and Mead, 1965) with simulated annealing (Kirkpatrick et al., 1983), also incorporating fundamentals of evolutionary algorithms, namely the concepts of an evolving population and the genetic operators (Ryan, 2003).

In this respect, it combines the flexibility of simulated annealing to escape from local optima, with the ability of the Nelder-Mead method to locate areas of attraction quickly and accurately. This is accomplished through the introduction of a "temperature" variable, T , which determines the randomness assigned to the search procedure. At early stages, temperature is desired to have large values, thus making the system "warm", so that randomness can play a major role to favor the exploration across the entire feasible space. In contrast, as the search evolves, the algorithm is capable of finding areas of attraction and the system gets "colder" (since its temperature decreases), thus the search becomes more deterministic and exploitation can begin.

At each iteration cycle, the generation of new solutions is realized by randomly selecting from the population so far sub-sets of $n+1$ points in the n -dimensional search space (thus each sub-set defines a *simplex*) and employing appropriate geometrical transformations. In order to determine the simplex vertex to be replaced, the associated population members are not being compared exclusively by their objective function value, but a randomness term, related to the current system temperature, is added. In early iterations, randomness is often the most crucial component in the comparison, and, thus, the solution being chosen for replacement may not be the actually worst simplex vertex. Nonetheless, the best vertex (i.e., the solution with the lowest objective function value) is not part of this comparison, in order not to accidentally discard a good solution. Next, the algorithm seeks for improved solutions, based on a stochastic formulation of the standard Nelder-Mead sequence (reflection, expansion, contraction, shrinkage), also introducing additional transformations. If these movements fail to detect improved points, then a mutation mechanism is activated, which ensures diversity among the population members and eventually helps the algorithm escape from possible local optima.

2.2. Surrogate-Enhanced Evolutionary Annealing Simplex

SEEAS is a heuristic population-based global optimization method, originally developed by Tsoukalas et al. (2016) and is essentially an extension of EAS, in a sense that a surrogate model (SM) is incorporated to assist the search procedure, in particular a cubic RBF with linear polynomial tail, which is an interpolation method. SEEAS also uses and maintains an external archive, which includes all population members so far, for which the value of the real objective function is computed. Every time a new function value is obtained through the simulation model, the associated point enters the archive, too. All data in the archive are used whenever the interpolation is applied, so that the metamodel can approximate the response surface of the real model more accurately and make useful predictions, that will help the convergence towards the global optimum. As the number of objective function evaluations increases, the archive enlarges, as well, and the updated surrogate becomes progressively more accurate in its predictions.

In fact, the SM in SEEAS has a double role, namely locating, autonomously, new promising points, where the objective function will be evaluated, and providing guidance on the execution of the simplex transformations (as employed in EAS), by indicating promising directions. The RBF comes with an Acquisition Function (AF), which is a well-known technique used in Surrogate-Based Optimization (SBO) in order to balance exploration and exploitation. The AF in SEEAS uses self-adjusting weights, that are updated in every iteration according to the current number of objective function evaluations and the maximum allowed one.

The generation of the initial population is employed via the Latin Hypercube Sampling (LHS) technique (Giunta et al., 2003), which is an established statistical method, that ensures sufficient sampling across the search space. A typical iteration step is carried out as follows: Initially, the SM is fitted to the data of the current external archive. Next, an internal global optimization algorithm (i.e., the original version of EAS) is used to optimize the updated SM, by using as objective function the AF emerging from the fitted RBF. The arising global minimum is a candidate solution to enter the population. If this point is better than the worst solution in the current population, then it replaces it and enters the population, otherwise it is rejected. In any case, it enters the external archive, thus improving the available information about the geometry of the search space. Afterwards, the search procedure is based on the genetic operators of EAS. In this respect, a simplex is randomly created from the existing population and executes the standard simplex movements. A key difference with EAS is that all movements except for shrinkage (i.e., reflection, expansion, contraction) are supported by the metamodel. For instance, after defining the direction of reflection, multiple new candidate points are produced along this, and the SM is applied to dictate which one should be chosen and evaluated through the actual simulation model. The rest elements of EAS are maintained as in the original algorithm, in particular, the mutation operator and the self-adjusting annealing schedule, which controls the system's temperature, and eventually the randomness of the search procedure. At the end of the iteration, there is at least one new point obtained, that enters the population, replacing one of its preexisting members. The search is terminated either by reaching a maximum allowed number of function evaluations or by fulfilling a given convergence criterion.

For a detailed description of SEEAS, regarding the mathematical equations of the RBF, AF and the analysis of the surrogate-enhanced EAS operators, interested readers are encouraged to refer to Tsoukalas et al. (2016).

2.3. Adaptive Multi-Surrogate Enhanced Evolutionary Annealing Simplex

The proposed algorithm is an improved version of SEEAS. The notable difference between the two algorithms is that AMSEEAS is not limited to the incorporation of a single metamodel, namely the RBF surrogate. Its rationale stands on the fact that there is a wide variety of metamodels, listed in the literature, that are able to be embedded within optimization algorithms in order to assist the search procedure, and that there is no specific metamodel clearly superior to the rest ones. In fact, the performance of a specific SM depends on the available data (i.e., points in which the real objective function value is computed) and the peculiarities of the response surface of the underlying optimization problem (dimensionality, complexity, multimodality). In cases where the metamodel does not fit well to the existing data, thus providing poor predictions, there is, consequently, an increased risk to sacrifice a substantial number of function evaluations, practically without any improvement. That being said, it is possible, in some occasions, to apply a surrogate-based approach but actually have the opposite effect from the desirable one, which is the drastic decrease of the computational effort induced by the objective function calls. In this vein, a more effective policy would be detecting whether a surrogate exhibits a good behavior or not, and, if so, completely discarding it from the search procedure.

In the AMSEEAS setting, multiple metamodels coexist and operate as a group, as the optimization evolves. However, the implementation of many SMs does not necessarily guarantee a better convergence behavior. The key challenge is ensuring that all surrogates cooperate effectively, support each other and exploit each other's advantages. In this mindset, AMSEEAS incorporates five SMs in the core of EAS, namely: *i*) a cubic RBF with linear polynomial tail, *ii*) a Random Forest, *iii*) a Support Vector Machine, *iv*) a Gaussian Process with a rational quadratic kernel, and *v*) a Gaussian Process with a Matérn kernel. These five metamodels emerged, among many others, after extensive analysis and experimentation with different optimization problems.

As mentioned in the previous section, the surrogate model in SEEAS has a double role. The first one is providing, on its own, new promising points for real objective function evaluation and the second one is supporting the simplex movements. Regarding the second role, AMSEEAS maintains the same principles and uses the same surrogate as SEEAS (i.e., cubic RBF with linear polynomial tail), exclusively, which seems to co-operate well with the downhill simplex method.

Regarding the first role, the novelty introduced in AMSEEAS involves the creation of a virtual roulette, which is responsible for deciding which out of the five surrogates is activated in every iteration. This is implemented by assigning different probabilities to the five SMs at the beginning of each iteration. In particular, the metamodels that are more likely to make better predictions get a higher probability of being chosen by the roulette for activation. On the contrary, metamodels that seem to have poor fitting on the data are excluded from the roulette spinning, only for this iteration, to avoid wasting an objective function evaluation. Finally, metamodels with marginally satisfactory fitting get a low probability of being selected. Under this premise, either one specific surrogate will be activated (i.e., as chosen by the roulette) or none, when all five metamodels demonstrate too poor fitting on the existing data and, hence, are all excluded from the roulette. The assignment of probabilities is determined by a well-known goodness of fit criterion, namely the Nash-Sutcliffe efficiency (NSE):

$$NSE = 1 - \frac{\sum_{i=1}^n (y_m^i - y_o^i)^2}{\sum_{i=1}^n (y_o^i - \bar{y}_o)^2} \quad (1)$$

where \bar{y}_o is the mean actual value, y_m^i and y_o^i are the i th modeled and actual values, respectively, and n is the size of data. The NSE values range between $-\infty$ and 1. When $NSE = 1$, the model perfectly fits the observations, whereas if $NSE = 0$, the model has the same predictive skill with the mean actual value. Negative NSE values indicate a very bad fitting, with worse predictive skill than the mean, \bar{y}_o .

Based on the aforementioned analysis, the threshold determining whether a surrogate will participate in the roulette spinning or not, in a given iteration, is the value of $NSE = 0$. This is reasonable, since the fact that a metamodel's fitting corresponds to a negative NSE value is a strong indicator that the particular SM will probably waste an objective function evaluation, if chosen by the roulette. On the other hand, metamodels with positive NSE values will be part of the roulette wheel and have a non-zero probability of being selected, which is proportional to their NSE value. In particular, considering m total surrogates ($1 \leq m \leq 5$) with positive NSE values (i.e. $NSE_1, NSE_2, \dots, NSE_m$), we compute their sum ($NSE_{sum} = NSE_1 + NSE_2 + \dots + NSE_m$) and their corresponding probabilities are calculated as $p_1 = NSE_1/NSE_{sum}$, $p_2 = NSE_2/NSE_{sum}$, ..., $p_m = NSE_m/NSE_{sum}$.

In order to estimate the NSE values at each iteration, the total data so far in the archive are split into a training and a test set. The training set contains a randomly determined 80% of points, and the test set contains the remaining 20%. The five surrogates are fitted to the same training set and then make their predictions in the same test set, including the rest available points.

Provided that at least one of the five surrogates has a positive NSE value, the roulette mechanism is activated to select the surrogate to be next used for predicting and providing a new promising point, for evaluating the real objective function. Contrarily, if all metamodels have a negative NSE value, the roulette mechanism remains deactivated, and the iteration continues by "jumping" to the surrogate-assisted genetic operators of SEEAS.

The roulette mechanism contains an additional functionality, i.e., permanently eliminating surrogates, once they reach a specific threshold. More specifically, once a metamodel is chosen by the roulette, it indicates the next point, where the real objective function value will be calculated. If this point is better than the worst point in the current population, it replaces it, otherwise it is ignored. So, if a surrogate makes a bad prediction (i.e., it generates a worse point than the whole current population), then a penalty counter is initialized. When this counter reaches a given maximum value, then the associated metamodel is considered unable for providing any more assistance. Hence, it is permanently removed from the system, allowing for the rest of the metamodels to continue enhancing the optimization procedure. However, if all metamodels reach that maximum penalty value, then the roulette mechanism is discarded, and the search evolves by only employing the surrogate-assisted genetic operators of SEEAS.

A demonstration of the incorporation of multiple metamodels and the corresponding virtual roulette mechanism is illustrated in Fig. 1. Since the cubic RBF with linear polynomial tail has a negative NSE value (i.e., $NSE = -0.050$), it will abstain from the roulette for this particular iteration, whereas the rest of surrogate models will participate in it. Since the Random Forest metamodel exhibits the highest NSE value (i.e., $NSE = 0.916$), it also gains the highest probability (i.e., 46.97%) of being chosen for prediction.

The proposed algorithm includes two final extensions. The first one refers to the sampling method used to generate the initial population. The default method is the LHS technique, however our extensive investigations with test functions (see Section 3) showed that, in some cases, the algorithm performs much better when the Symmetric Latin Hypercube Design (Ye et al., 2000) is employed. The second extension

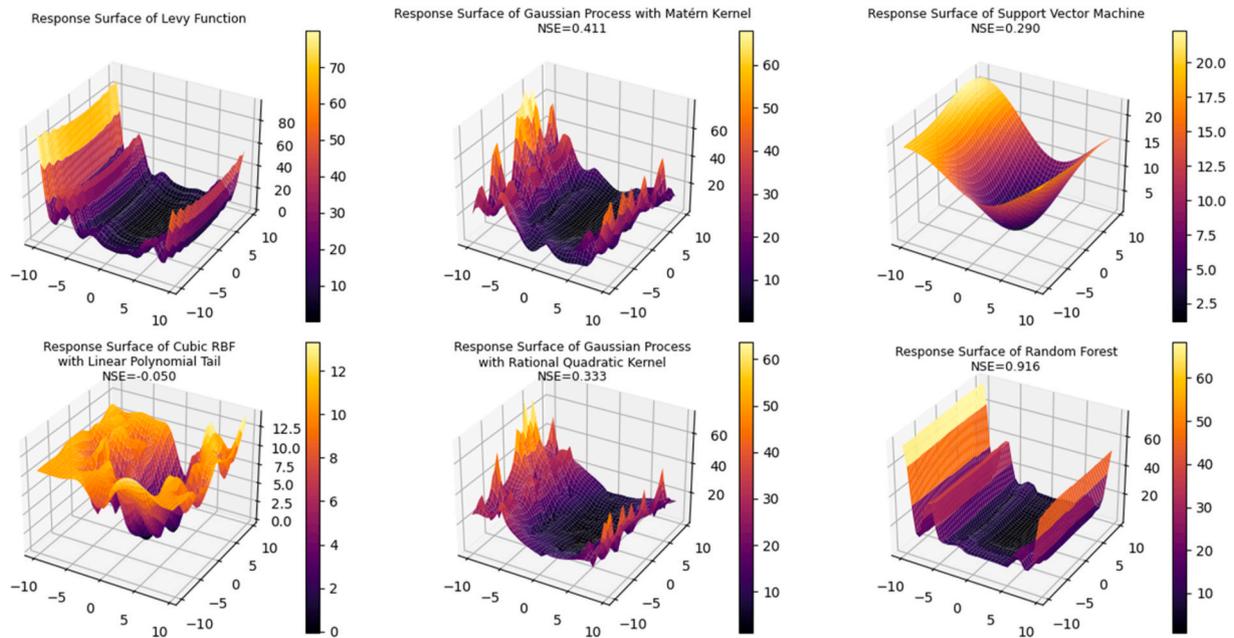


Fig. 1. Real response surface of the 2-D Lévy function in the domain $[-10, 10]$ and approximated ones with the five surrogate models. The sample data contains 200 points, where 160 are used for fitting purposes and 40 for NSE evaluations.

refers to a local search strategy, in an attempt to further improve the best solution emerging at the end of each iteration cycle. More explicitly, after finalizing the surrogate-assisted genetic operators, the current best population solution is acquired and a search across its neighborhood is conducted. In this respect, a large number of points around the optimal one is generated and evaluated through the cubic RBF with linear polynomial tail, in order to indicate the most promising one for employing a real objective function evaluation. If this is better than the current worst point in the population, it replaces it, otherwise it is rejected. Nevertheless, the new point enters the external archive.

Flowcharts of both SEEAS and AMSEEAS are presented in Fig. 2.

3. Benchmarking of optimization algorithms with mathematical test functions

3.1. Problem setup

In order to assess the performance of AMSEEAS with respect to other well-established surrogate-based global optimization methods, we initially evaluate them against a number of standardized optimization tests, involving mathematical functions that exhibit different complexities.

AMSEEAS is benchmarked against five state-of-the-art surrogate-based global optimization algorithms that are listed in Table 1, specifically the ones that are publicly available, namely PODS (Xia et al., 2021), TURBO (Eriksson et al., 2019), SOP (Kritiyakieme et al., 2016), SEEAS (Tsoukalas et al., 2016), DYCORS (Regis and Shoemaker, 2013). In our tests, we also include the classical MLMSRBF method (Regis and Shoemaker, 2007), since it has gained significant popularity over the time, as indicated by the associated number of citations (more than 400). Regarding TURBO, we consider two different configurations of it, depending on the number of trust regions used, which is a critical input argument of the algorithm. Thus, we evaluate two versions, herein referred to as TURBO-1 and TURBO-M, with 1 and $M = 5$ trust regions, respectively. Under this premise, eight algorithms are eventually participating in the benchmarking.

In order to ensure fair comparisons, we use the same population size for all algorithms, equal to $m = 2 \times (n + 1)$, where n is the number of control variables, as proposed by Regis and Shoemaker (2006). Moreover, in all cases, the generation of the initial population (also referred

to as Design of Experiment, DoE) is employed via the LHS technique, and the default values for all algorithmic inputs and hyperparameters are set, as suggested in the associated publications. As far as the computational workload of the core optimization procedures (e.g., random sampling, geometrical transformations, building of surrogates, check of termination criteria), is minimal, we consider that the simulation is by far the most time-consuming stage and, thus, the total computational time of the search procedure is mainly determined by the total number of function evaluations.

The benchmarking “suite” consists of six well-recognized mathematical functions, the global minimum of which is known a priori and is equal to zero. The six functions are listed in Table 2.

For each optimization problem shown in Table 2, we consider two different dimensions, by setting the number of control variables equal to $n = 15$ and $n = 30$, as well as two different computational budgets, in terms of maximum allowable function evaluations (MFE), which are set equal to $MFE = 500$ and $MFE = 1000$. In 15-D and 30-D problems, the population size is set equal to $m = 32$ and $m = 62$, respectively. These assumptions are realized in order to have a much more detailed overall view of the algorithms’ capabilities and assess their performance, not only in cases where the complexity of the problem increases, but also in cases where the simulation phase is so time-consuming, that only a few hundred objective function evaluations can be executed for obtaining a satisfactory solution, in a reasonable time.

As a result, a total of $6 \times 2 \times 2 = 24$ different optimization problems arise. Each problem is executed 30 times by starting from independent populations, so that sufficient samples can be collected and then assessed. In every algorithm run, the best approximation to the global minimum is retrieved (i.e., closest convergence to zero). Afterwards, we compute the median value across the sample of 30 optimized sets in all 24 optimization problems and across the eight competing algorithmic schemes.

In order to draw valid conclusions about all algorithms’ capabilities and compare their performances, we apply the concept of stochastic dominance (Levy, 1992). In this respect, for each algorithm and each test problem that runs 30 times from different initial populations, we collect the associated optimal values and plot their empirically-derived cumulative distribution function (CDF). To contrast the performance of two algorithms, A and B, in a given problem, we compare their CDFs, symbolized Φ_A and Φ_B , respectively. Algorithm A is considered

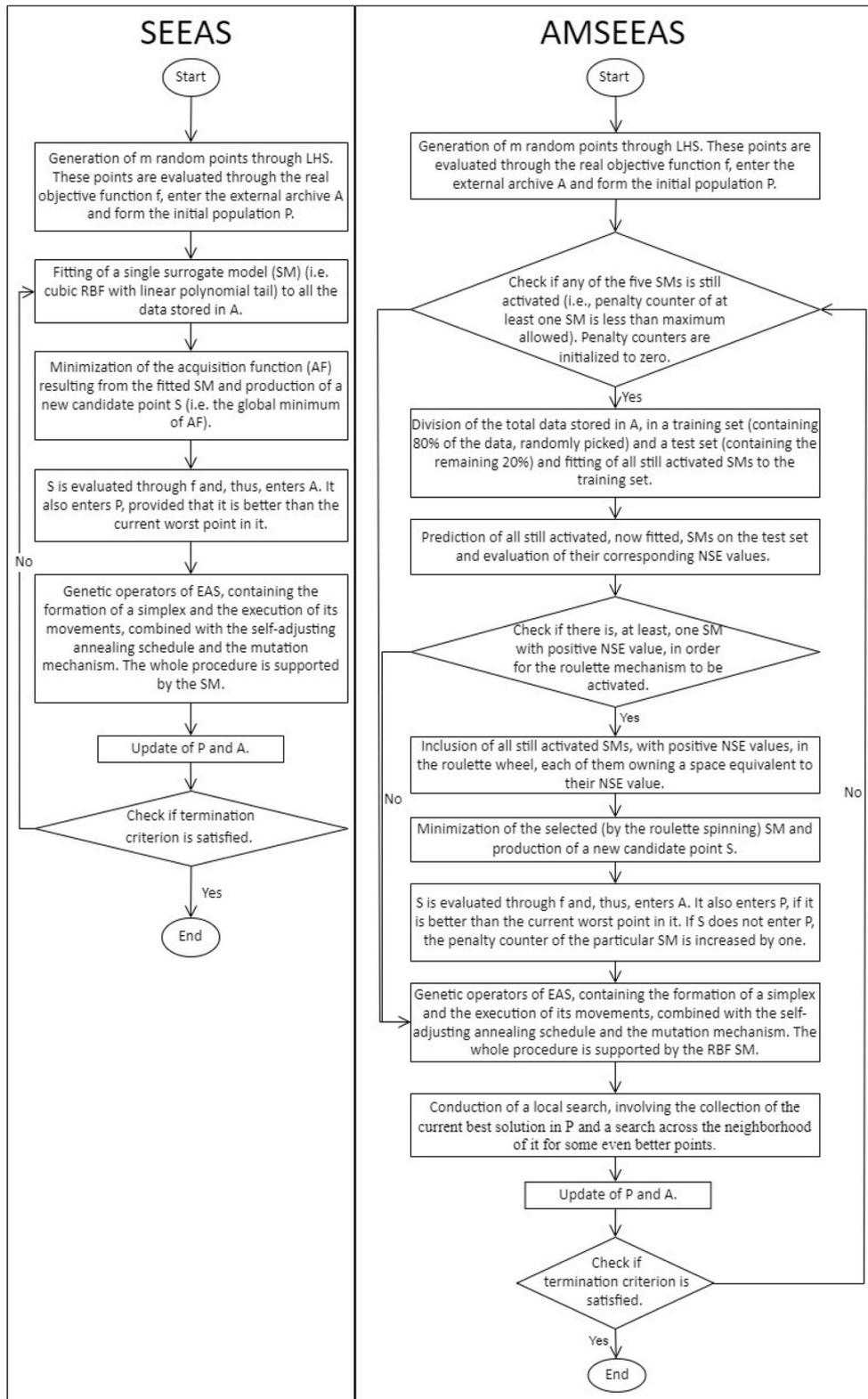


Fig. 2. Flowcharts of SEEAS and AMSEEAS.

stochastically dominant over B, if $\Phi_A(q) > \Phi_B(q)$ for all q and vice versa, where q is a random quantity to minimize. If, however, the two CDFs are intersected, then we compare their median values and thus consider as dominant the algorithm with the better performance at this point, given that the difference at their medians is statistically significant. To ensure statistical significance, we apply the non-parametric Wilcoxon

signed-rank test (Woolson, 2008). The null hypothesis of the test is that the data in Φ_A and Φ_B are samples from continuous distributions with equal medians, while the confidence level is set equal to 95%. If the null hypothesis is not rejected, then the two algorithms are considered equally good.

Table 2
Definition of the six mathematical optimization problems.

Problem	Test Function	Parameters Bounds
OF1	Sphere	[- 5.12, 5.12]
OF2	Ackley	[- 32.768, 32.768]
OF3	Griewank	[- 600, 600]
OF4	Zakharov	[- 5, 10]
OF5	Rastrigin	[- 5.12, 5.12]
OF6	Lévy	[- 10, 10]

3.2. Results

Table 3 and Table 4 depict the performance of the eight competing algorithms in terms of median of the optimized function values found for 30 independent runs over the six test problems in the 15-D and 30-D space, respectively, and under the two budgets (i.e., 500 and 1000

maximum function evaluations).

For each optimization problem, the empirical CDFs of all algorithms are presented in Appendix A (Figure A1 to Figure A6). Whenever required, we also employ the Wilcoxon signed-rank test between the algorithms providing the best (i.e., lowest) median values, in order to help us draw clear conclusions about the algorithm exhibiting stochastic dominance in the particular problem. The summary results of these tests are presented in Table 5, where n is the number of control variables and H indicates the rejection or not of the null hypothesis (i.e., if $H = 0$, the null hypothesis is not rejected and, thus, both algorithms are considered equally good). The rejection or not of the null hypothesis on a particular problem depends on the value of W , which is the outcome of the Wilcoxon test. If $W < W_{crit}$, the null hypothesis is rejected (W_{crit} is equal to 137 and denotes the critical value of W for a two-tailed test with a sample size of 30 and 95% confidence level).

As an example, in Fig. 3, we demonstrate one case where the CDF

Table 3
Median of best solutions in 15-D test problems (optimal results are highlighted with bold).

MFE	Test Function	AMSEEAS	SEEAS	TURBO-1	TURBO-M	PODS	SOP	DYCORDS	MLMSRBF
500	OF1	3.16E-06	0.002	0.055	0.311	3.53E-05	0.001	0.002	0.012
	OF2	0.034	0.838	2.224	3.479	1.341	1.577	0.745	2.353
	OF3	0.724	0.513	1.224	2.098	0.448	0.910	0.921	1.088
	OF4	26.767	53.874	52.140	123.632	45.217	94.844	154.151	147.998
	OF5	15.921	45.061	40.652	54.652	40.421	43.211	37.912	37.696
	OF6	0.104	0.198	2.579	4.524	0.456	2.352	0.681	0.488
1000	OF1	2.39E-06	0.001	2.00E-04	0.001	2.58E-05	9.06E-06	0.001	0.008
	OF2	0.007	0.410	0.221	0.505	0.007	0.133	0.574	1.629
	OF3	0.513	0.360	0.266	0.577	0.114	0.552	0.819	1.027
	OF4	20.117	34.413	14.880	70.441	33.071	74.015	127.557	110.313
	OF5	15.919	31.808	47.916	46.395	32.363	38.314	32.644	34.522
	OF6	0.090	0.114	2.985	0.792	0.001	2.169	0.069	0.191

Table 4
Median of best solutions in 30-D test problems (optimal results are highlighted with bold).

MFE	Test Function	AMSEEAS	SEEAS	TURBO-1	TURBO-M	PODS	SOP	DYCORDS	MLMSRBF
500	OF1	1.93E-04	0.018	3.081	42.692	0.037	0.026	0.073	0.590
	OF2	1.908	1.918	8.895	15.212	4.649	4.032	3.144	4.725
	OF3	0.975	0.807	12.836	95.354	1.135	1.077	1.249	2.974
	OF4	112.587	168.695	292.705	405.185	347.033	333.551	456.956	570.266
	OF5	66.554	121.973	145.826	271.773	85.937	191.689	112.009	156.834
	OF6	0.496	0.630	7.669	52.109	3.137	12.859	2.075	2.302
1000	OF1	6.13E-05	0.005	0.124	0.763	3.35E-04	1.34E-04	0.009	0.270
	OF2	0.787	1.170	2.340	4.011	1.474	2.553	1.108	3.438
	OF3	0.807	0.554	1.421	2.482	0.404	0.823	1.025	2.507
	OF4	82.446	147.120	177.904	319.483	259.642	287.711	409.986	465.057
	OF5	26.864	97.994	67.160	87.387	66.730	117.479	85.728	127.299
	OF6	0.188	0.431	4.195	6.681	0.629	10.215	2.762	1.412

Table 5

Summary results of Wilcoxon signed-rank tests to locate the preferred algorithm(s) for each optimization problem (the preferred algorithm(s) is (are) highlighted with bold).

n	Test Function	MFE = 500				MFE = 1000			
		Preferred	Alternative	W	H	Preferred	Alternative	W	H
15	OF1	AMSEEAS	PODS	0	1	AMSEEAS	SOP	0	1
	OF2	AMSEEAS	DYCORS	188	0	PODS	AMSEEAS	132	1
	OF3	PODS	SEEAS	119	1	PODS	TURBO-1	56	1
	OF4	AMSEEAS	PODS	10	1	TURBO-1	AMSEEAS	91	1
	OF5	AMSEEAS	MLMSRBF	0	1	AMSEEAS	SEEAS	1	1
	OF6	AMSEEAS	SEEAS	171	0	PODS	DYCORS	0	1
30	OF1	AMSEEAS	SEEAS	0	1	AMSEEAS	SOP	0	1
	OF2	AMSEEAS	SEEAS	179	0	AMSEEAS	DYCORS	108	1
	OF3	SEEAS	AMSEEAS	0	1	PODS	SEEAS	19	1
	OF4	AMSEEAS	SEEAS	0	1	AMSEEAS	SEEAS	0	1
	OF5	AMSEEAS	PODS	86	1	AMSEEAS	PODS	0	1
	OF6	AMSEEAS	SEEAS	114	1	AMSEEAS	SEEAS	36	1

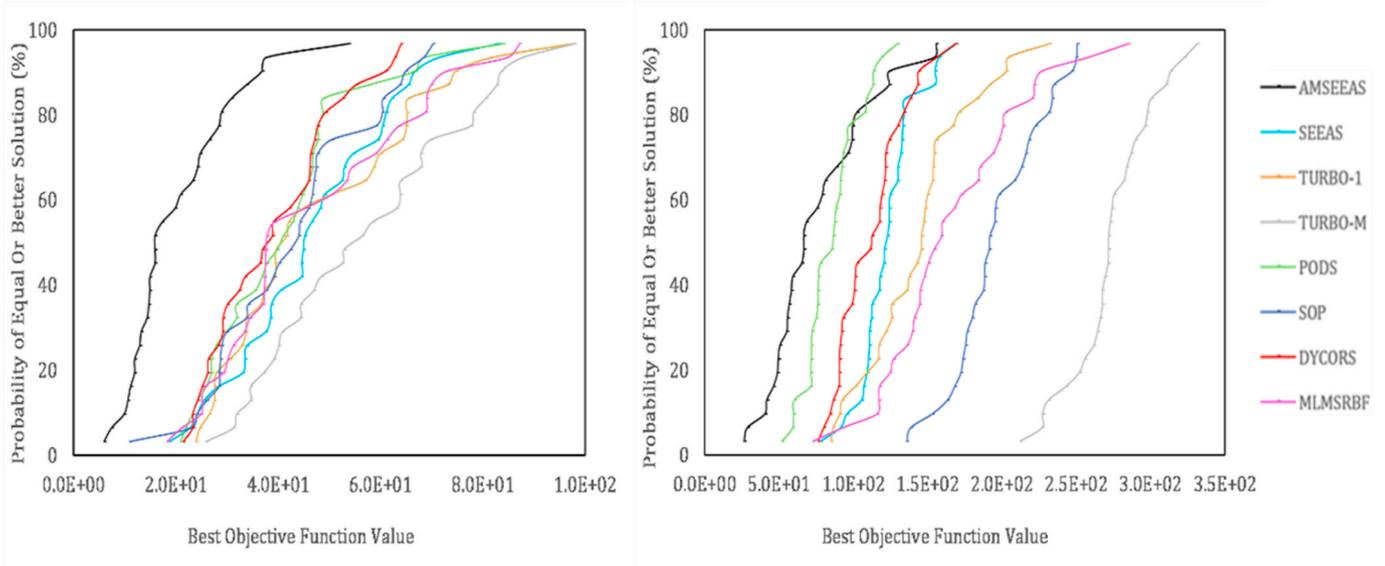


Fig. 3. Empirical CDFs for test function OF5 (*Rastrigin*) with 15 (left) and 30 variables (right), with $MFE = 500$.

plots are sufficient for evaluating which algorithm is dominant in the particular problem (i.e., 15-D Rastrigin case with $MFE = 500$), and one case where the Wilcoxon signed-rank test is essential to detect the dominating method (i.e., 30-D Rastrigin case with $MFE = 500$).

The aforementioned analyses clearly indicate the superiority of AMSEEAS, in most of examined problems. In 15-D configurations, the proposed algorithm dominates in five (*OF1*, *OF2*, *OF4*, *OF5*, *OF6*) and two (*OF1*, *OF5*) problems, for $MFE = 500$ and $MFE = 1000$, respectively. As the number of control variables and thus the complexity of the optimization problem increases, AMSEEAS performs even more efficiently. In particular, it achieves best performance in five (*OF1*, *OF2*, *OF4*, *OF5*, *OF6*) problems, for both computational budgets.

Overall, AMSEEAS is stochastically dominant in 17 out of 24 problems, PODS in 5, whereas SEEAS in 3. The rest algorithms exhibit less satisfactory performance. That is, TURBO-1 and DYCORS are optimal in only one case each, while TURBO-M, SOP and MLMSRBF remain sub-optimal across all problems.

It is important to highlight that all algorithms perform poorly against OF4 (Zakharov) and OF5 (Rastrigin), as none of them exhibits satisfactory convergence to the global minima, especially in the 30-D space. This is not surprising, as these functions produce complicated response surfaces, thus making it extremely difficult for the metamodels to fit the data and offer useful predictions.

3.3. Convergence behavior analysis

In order to extend our analysis regarding the proposed algorithm's effectiveness, we examined one additional issue, referred to the convergence speed towards the global optimum, in terms of objective function evaluations. Briefly, in every run (out of 30, in total) of each out of 24 optimization problems, we retrieved the best approximations to the associated global optimum, as the objective function evaluations increase towards their maximum allowed limit (i.e., $MFE = 500$ or 1000). Subsequently, we computed the median values of the best solutions so far, in order to plot and evaluate how fast the optimization evolves in each case. The resulting convergence curves are presented in Appendix B (Figure B1 to Figure B6). In Fig. 4, we present, for illustration purposes, the resulting convergence curves of all competing algorithms in a particular optimization problem.

As the figures indicate, in some optimization problems (i.e., Sphere, Zakharov) AMSEAS exhibits faster convergence towards the global optimum, even from the first few hundreds of function evaluations, thus, clearly outperforming the rest optimization methods. Besides that, even when AMSEAS is initially evolving relatively slowly (i.e., Ackley, Rastrigin, Lévy), afterwards its behavior is remarkably improved. For instance, in the 30-D Lévy cases, while SEEAS outperforms AMSEAS at the early stages of the optimization procedures, after approximately four hundred evaluations the latter converges much faster. Another noteworthy case is the Zakharov function, for which the performance of SEEAS and other surrogate-based approaches is rather poor (see discussion by Tsoukalas et al., 2016). However, the multi-model approach coupled with the roulette and penalty mechanisms, that are introduced in the AMSEAS version, made the algorithm clearly more effective.

4. Real-world benchmarking in a computationally demanding hydraulic design problem

Since the application of surrogate-assisted methods in practice involves HEB problems, in order to obtain a more comprehensive picture of the proposed algorithm's capabilities, we also test how AMSEAS performs in such a real-world problem, by comparing it to its precursor, i.e., SEEAS.

4.1. Study area and optimization problem definition

The optimization problem involves a hydraulic design study in the context of a broader flood risk assessment analysis, which is described in detail in the recent work by Efstratiadis et al. (2022). The area of

application is the lower course of Trachones stream, which crosses highly urbanized suburbs of Athens, Greece. Its total drainage area is approximately 24 km^2 and it extends in the south of Athens, between the foothills of Mount Hymettus and the coast.

The design optimization task refers to the sizing of levees along the open parts of the lower drainage network, which is conceptually configured by means of 27 lateral weirs that are represented in the HEC-RAS environment. We remark that the three out of 27 Lateral Structures are internal and act as Side Channel Spillways, by transferring flow from one channel section to another. These levees do not produce overflow and, thus, are not part of the design variables set.

The control variables of the design optimization problem are the elevations, h_i , of 24 out of 27 individual levees, which are allowed to receive non-negative values up to 1.0 m . Two conflicting criteria are considered, namely the total overflow occurring from the major channel segments over their associated levees and the total construction cost. In this respect, as the elevation increases, the expected overflow decreases, however the construction cost increases, accordingly. As explained latter, for a given configuration of the drainage system, the overflow over the levees is estimated through a hydrodynamic model, driven with a specific design flood event. On the other hand, the construction cost of each levee is estimated by:

$$C_i = c L_i h_i d \quad (2)$$

where L_i is the length of each individual levee, d is the crest width (same for all levees), which is set equal to 2 m , and c is an indicative unit construction cost, which is, for convenience, set equal to $30\text{€}/\text{m}^3$.

The objective function to minimize is expressed in dimensionless terms as follows,

$$f(h_1, h_2, \dots, h_n) = \frac{TOV}{TOV_{max}} + \frac{TC}{TC_{max}} \quad (3)$$

where $TOV = \sum_{i=1}^n Vi$ and $TC = c(\sum_{i=1}^n L_i h_i d)$. Specifically, TOV denotes the total overflow, TC stands for the overall construction cost of levees, Vi is the overflow of each individual weir, i , and $n = 24$ is the number of control variables. Similarly, TOV_{max} is the maximum potential total overflow, which refers to the 'do nothing' solution (i.e., zero increase of elevation in all 24 lateral structures, and hence zero cost) and is equal to $3272.6 \times 10^3 \text{ m}^3$, while TC_{max} denotes the maximum overall construction cost, by assuming that all 24 individual levees are elevated by $h_{max} = 1.0 \text{ m}$ and is equal to $515.9 \times 10^3 \text{ €}$. The above formulation of the objective function makes the optimization task independent of the uncertainty induced by the subjective assignment of input arguments d and c .

The hydrodynamic simulations are performed with the HEC-RAS 6.1 software, under one-dimensional analysis with 5 m spatial resolution and a computational time step of 30 s . The coupling of HEC-RAS with the optimization algorithms is implemented in a Python environment, by utilizing the HEC-RAS Controller (Goodell, 2014), which is part of the HEC-RAS application programming interface (API). The Controller incorporates a wealth of procedures, that allow the manipulation of HEC-RAS externally by setting input data, retrieving input or output data, and performing common functions, such as opening and closing HEC-RAS, changing plans, model running and plotting output data. Similar works employing the aforementioned API (e.g., for automation purposes) are those of Siqueira et al. (2016), Leon and Goodell (2016) and Dysarz (2018).

Regarding the hydrodynamic modeling procedure, since the longitudinal slopes of the stream branches are quite significant and the flow velocities are high, the so-called Local Partial Inertia (LPI) technique (Fread et al., 1996) is used for improving the stability of the numerical solution, by setting a Froude number threshold equal to 0.01. For the Manning's coefficient parametrization, the computational domain is classified into three specific friction zones, i.e., cross-sections constructed by concrete, gabions, and natural terrain, for which we apply

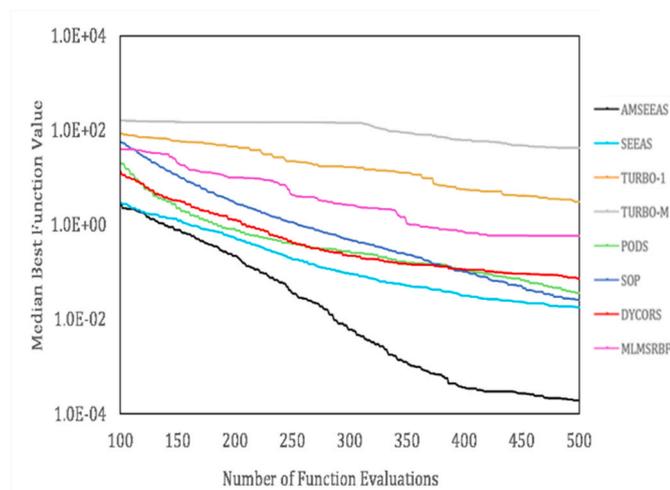


Fig. 4. Convergence curves for test function OF1 (Sphere) with 30 variables and $MFE = 500$.

0.016, 0.025, and 0.030 $s/m^{1/3}$, respectively.

The input flood event corresponding to the one-dimensional hydrodynamic model boundary conditions is selected from the set of stochastic weather scenarios produced by Efstratiadis et al. (2022) and refers to a return period of 500 years and a 90% confidence level.

This challenging problem reveals the actual value of SBO techniques, since to evaluate the objective function, a time-demanding HEC-RAS simulation needs to be executed first. Specifically, by using a 3 GHz Intel Core i9 processor with a 32 GB of RAM, a single run lasts 2 min. Consequently, to obtain an optimized design within a reasonable time period, only a few hundreds of function evaluations are “allowed”.

4.2. Performance comparison of the algorithms

As the computational workload is quite heavy, we consider a maximum allowed number of objective function evaluations equal to $MFE = 500$, while to extract sufficient statistical outcomes, we repeat the optimization procedure with SEEAS and AMSEEAS for a total of ten times. We remark that the computational time is almost fully dictated by the simulation stage, which in turn mainly depends on the execution of HEC-RAS. Provided that both algorithms are allowed to “call” the simulation 500 times, each optimization run requires approximately 17 hours.

In contrast to theoretical test functions, the global minimum of this highly complex engineering design problem cannot be known a priori. On the other hand, by setting all design variables equal to zero, it is easy to detect that the ‘do nothing’ solution results in a total overflow equal to $3272.6 \times 10^3 m^3$ and a zero cost, thus an objective function value equal to 0.50.

The key results obtained with SEEAS and AMSEEAS are shown in Table 6. These indicate that AMSEEAS ensures much better performance in comparison to SEEAS, while the outcomes of the latter are actually worse than the ‘do nothing’ scenario.

The superiority of AMSEEAS over SEEAS is shown even more clearly if we plot the CDFs of the associated optimal objective function values. As shown in Fig. 5, AMSEEAS is considered stochastically dominant over SEEAS.

In Fig. 6 we also plot the convergence curves of the two algorithms, to evaluate how fast, in terms of number of objective function evaluations, does the convergence procedure evolve. Once again, it is clear, that AMSEEAS outperforms SEEAS during the entire search procedure.

Finally, in Table 7 we compare the optimized control variables (in terms of elevation increase) of the overall best solution found by each algorithm, out of the ten optimization trials.

The layout of the best design solution of each algorithm is demonstrated in Fig. 7. A colorized scale ranging from blue to red is used to

Table 6

Optimal solutions found by SEEAS and AMSEEAS after ten independent runs of the design optimization problem.

Run	SEEAS			AMSEEAS		
	Total overflow (1000 m ³)	Total cost (10 ³ €)	Objective function value	Total overflow (1000 m ³)	Total cost (10 ³ €)	Objective function value
1	2947.2	151.0	0.5965	2956.8	28.5	0.4794
2	2895.1	131.5	0.5698	2957.5	23.9	0.4750
3	3140.0	105.3	0.5818	2903.9	33.6	0.4762
4	3128.9	116.9	0.5914	2795.4	55.8	0.4811
5	3043.0	135.7	0.5965	2961.0	23.9	0.4756
6	3081.2	139.0	0.6055	2902.6	32.2	0.4746
7	3066.8	131.2	0.5957	2957.0	23.7	0.4747
8	3193.6	128.6	0.6126	2902.8	32.6	0.4751
9	3027.8	132.3	0.5908	2914.2	40.2	0.4842
10	2957.1	159.2	0.6061	2904.2	32.0	0.4748
Mean	3048.1	133.1	0.5947	2915.5	32.6	0.4771

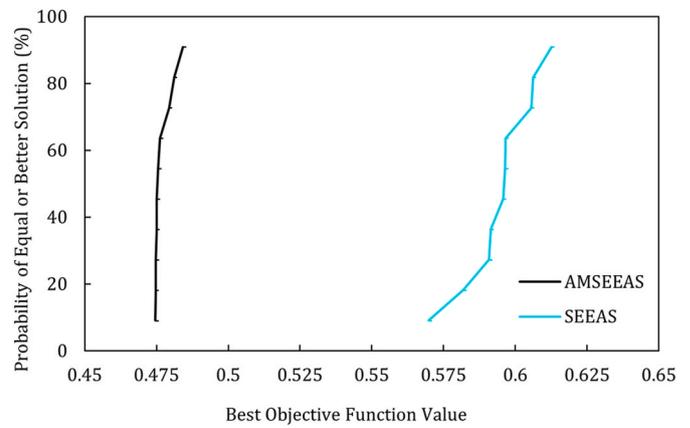


Fig. 5. Empirical CDFs of the optimal function values for the two algorithms.

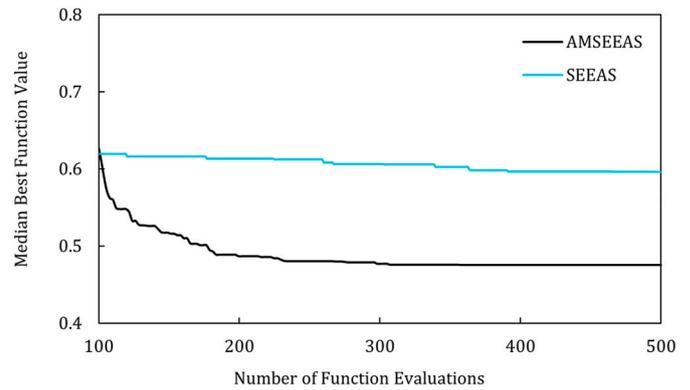


Fig. 6. Convergence curves of the two algorithms.

Table 7

Comparison between the overall optimal design variables found by SEEAS and AMSEEAS.

Lateral structure	Elevation increase (m)		Elevation difference (m)
	Optimal solution		
	SEEAS	AMSEEAS	
1	0.00	0.00	0.00
2	0.09	0.00	0.09
3	0.27	0.00	0.27
4	0.00	0.13	-0.13
5	0.00	0.00	0.00
6	0.53	0.28	0.25
7	0.81	0.00	0.81
8	0.00	0.00	0.00
9	0.00	0.00	0.00
10	0.80	0.00	0.80
11	0.00	0.00	0.00
12	0.41	0.00	0.41
13	0.00	0.00	0.00
14	0.00	0.00	0.00
15	0.00	0.00	0.00
16	0.00	0.00	0.00
17	0.00	0.00	0.00
18	1.00	0.96	0.04
19	0.00	0.00	0.00
20	0.18	0.00	0.18
21	0.00	0.00	0.00
22	0.00	0.00	0.00
23	0.00	0.00	0.00
24	0.00	0.00	0.00

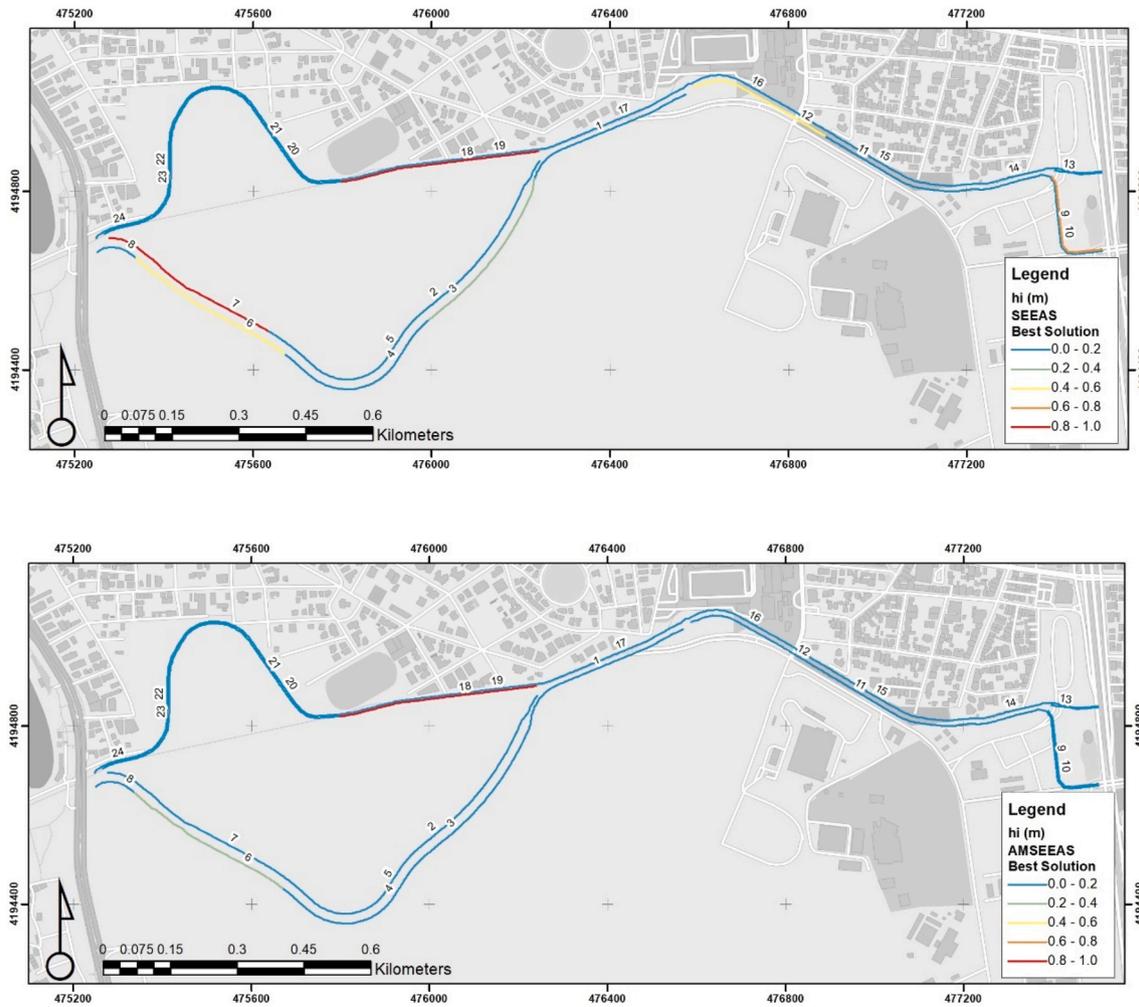


Fig. 7. Map of study area showing the lateral structures' elevations proposed by the best solution of SEEAS (up) and AMSEAS (down).

highlight the amount of additional elevation among the levees of the area in each case, with the blue values referring to the lowest possible elevations ($0.0m - 0.2m$) and the red values to the highest ones ($0.8m - 1.0m$). It is evident, that the best solution found by SEEAS requires higher elevation of the individual levees in comparison to the AMSEAS one, which justifies the huge difference in the construction costs. In particular, the solution proposed by AMSEAS is cheaper by 75.51%, while resulting in only 0.26% more total overflow than the SEEAS solution.

5. Conclusions

Global optimization problems are usually handled through objective functions, the values of which are available after the execution of a black box simulation model. However, several of the models used in environmental sciences require high computational effort, thus introducing significant barriers to the optimization procedure. The conflicting aspects of model accuracy and computational hardware requirements, led to the search for new ideas and tools to achieve satisfactory solutions within reasonable time. In this respect, SBO techniques are a well-established approach for such challenging problems.

This study introduces the Adaptive Multi-Surrogate Enhanced Evolutionary Annealing Simplex (AMSEAS) method, key novelty of which is the effective co-operation of multiple surrogate models, to ensure flexibility against objective functions and associated response surfaces of different characteristics. A virtual roulette is introduced to decide which SM should be activated in every iteration. The probability

of each metamodel being selected by the roulette for prediction, depends on how accurately it fits existing data. Each metamodel comes with a penalty counter, which increases whenever the metamodel makes a bad prediction. These counters can permanently discard surrogates when reaching a specific threshold.

To assess the efficiency and effectiveness of the proposed algorithm, we initially benchmarked it against six state-of-the-art surrogate-based global optimization methods, in six characteristic theoretical mathematical problems with alternative settings (i.e., two alternative dimensions and two computational budgets, thus resulting to 24 optimization problems, in total). The results emerging from this analysis are encouraging, as AMSEAS proves its robustness, by outperforming the other algorithms in most of problem settings. In particular, AMSEAS is considered stochastically dominant over its competitors in 17 out of 24 problems.

Next, we contrasted AMSEAS against SEEAS, i.e., the original method on which the proposed algorithm is based, in a highly challenging real-world problem, from the field of hydraulic design of flood protection works. A key barrier to such problems is the computational burden induced by the use of detailed hydrodynamic models (in this case, HEC-RAS), to assess the performance of a specific design in terms of flood hazard (and eventually, flood risk), while also keeping the cost of the proposed hydraulic infrastructures, to a minimum. Additional challenges are induced by the problem dimension, since in the examined case we were looking for optimizing 24 design variables, that represent the levee heights across the open channel network. Within our analysis, we detected the optimal solutions found by AMSEAS and SEEAS, from a

set of ten independent runs and under a limited computational budget of only 500 function evaluations. The design proposed by AMSEEAS ensures a substantially decreased construction cost with respect to SEEAS, and at the same time a low flood inundation risk, marginally only exceeding the estimated inundation hazard that results from the optimal design by SEEAS.

Our extended analyses with the mathematical problems and the hydraulic design application, as well, indicate that AMSEEAS is a robust and efficient optimization algorithm, able to handle computationally challenging HEB problems in practice, without compromising neither on simulation model sophistication nor on proper probabilistic treatment of complex environmental problems. This is due to the fact that the proposed method does not simply incorporate multiple surrogate models to support the optimization procedure. The introduced mechanisms behave in a completely stochastic manner and result in the adjustment of the exploited metamodels on the given objective function to minimize. Depending on the characteristics and irregularities of the response surface, some metamodels might be of actual assistance, while others might have the opposite behaviour and “undermine” the convergence process by pointing towards directions other than the region of interest. However, the inclusion of the virtual roulette and penalty mechanisms ensures that all inappropriate surrogates for a given problem will abstain from the search procedure, while the more appropriate ones will be given the chance to assist it, up until they stop producing useful information. This strategy assures that the number of wasted objective function evaluations is limited to a minimum extent and that the genetic

operators of SEEAS get the most proper assistance on any given optimization problem. Eventually, this strategy seems to be a very promising one, towards ensuring adaptation of the search procedure to response surfaces of varying characteristics.

Software and data availability

A Python implementation of EAS, SEEAS and AMSEEAS and the rest of our work are available online at <https://github.com/spyrostsat/AMSEEAS>.

Acknowledgements

The research leading to these results has received funding from the European Union’s Horizon 2020 research and innovation programme, from the EU Horizon 2020 Green Deal call, under grant agreement No 101037084, for the research project IMPETUS “Dynamic Information Management Approach for the implementation of Climate Resilient adaptation packages in European regions“. The research and its conclusions reflect only the views of the authors, and the European Union is not liable for any use that may be made of the information contained herein. The authors would like to thank the Editor, Dr. Saman Razavi, and the three reviewers, Dr. Jeremy White and two anonymous ones, for their constructive comments, suggestions and critique, which helped to provide a significantly improved manuscript.

Appendix A. Empirical CDFs for mathematical test functions

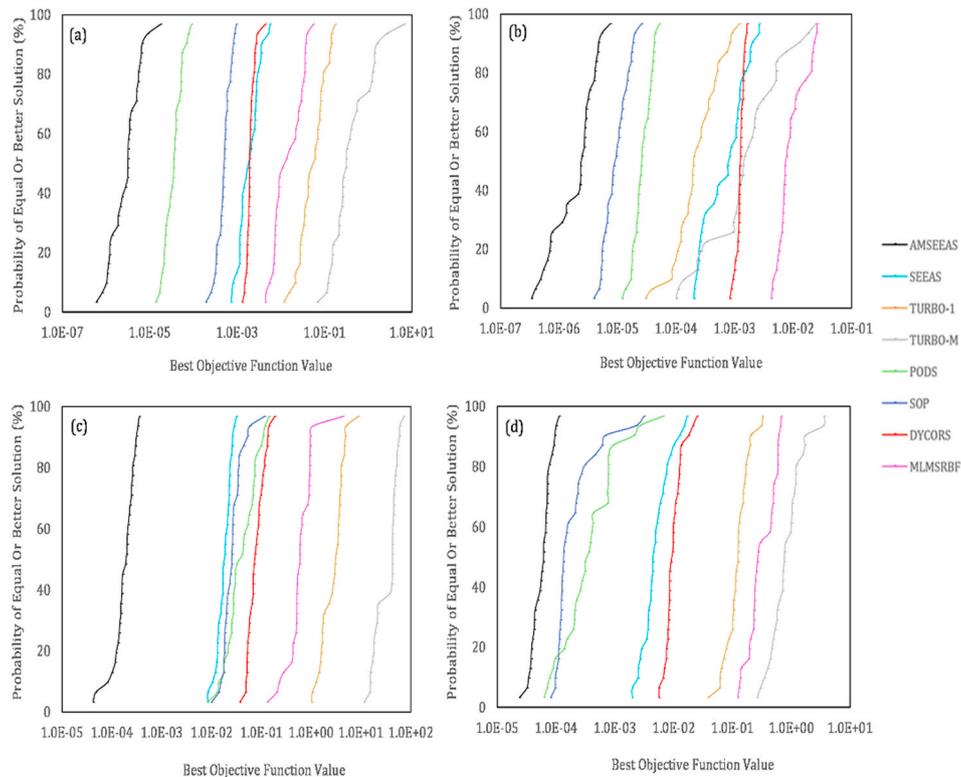


Fig. A1. Empirical CDFs for test function OF1 (Sphere) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

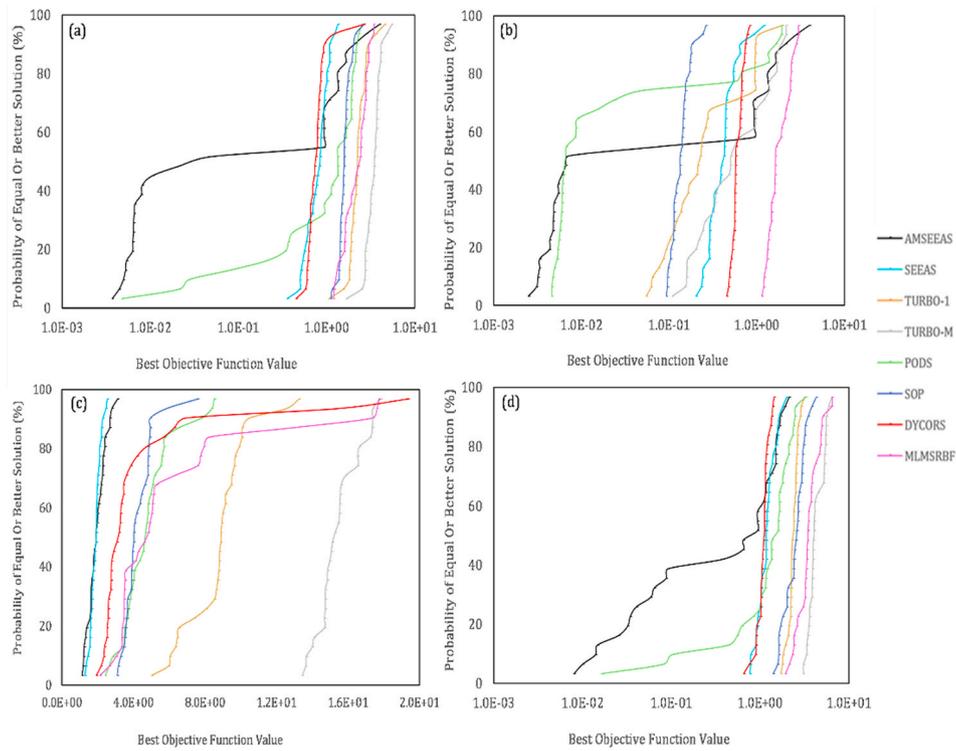


Fig. A2. Empirical CDFs for test function OF2 (Ackley) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

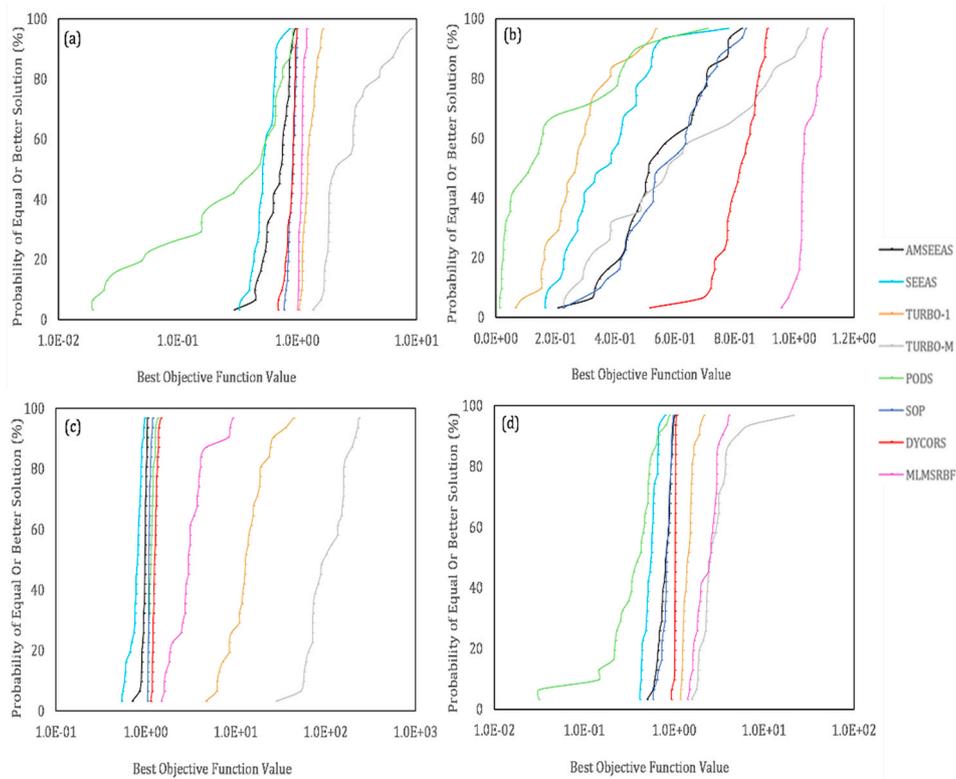


Fig. A3. Empirical CDFs for test function OF3 (Griewank) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

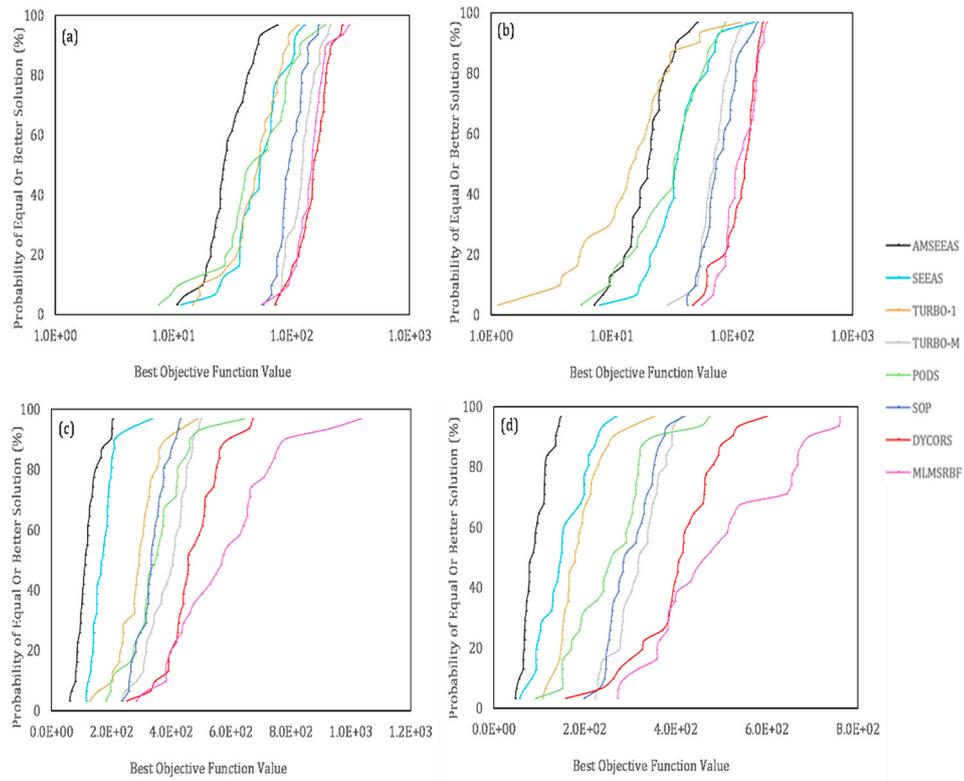


Fig. A4. Empirical CDFs for test function OF4 (Zakharov) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

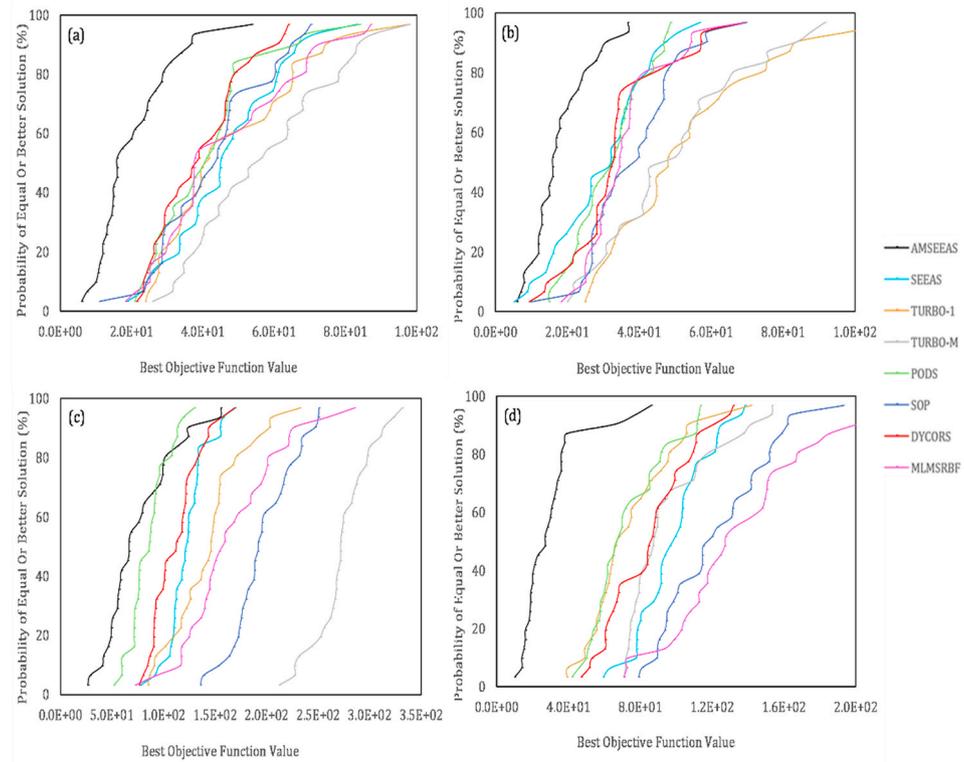


Fig. A5. Empirical CDFs for test function OF5 (Rastrigin) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

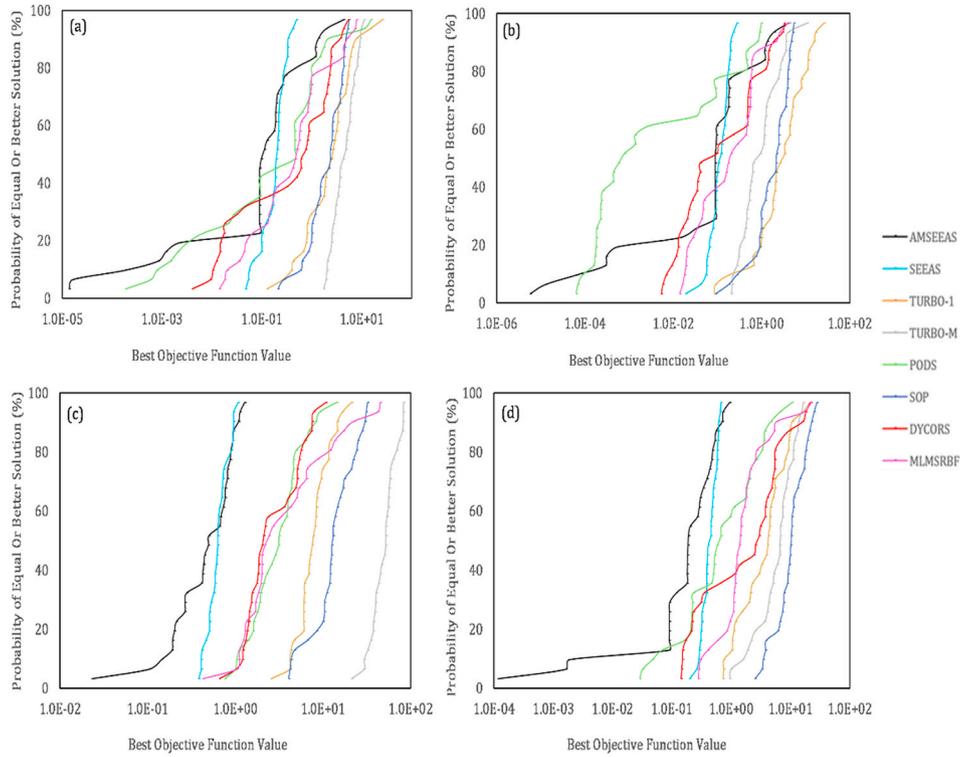


Fig. A6. Empirical CDFs for test function OF6 (Lévy) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

Appendix B. Convergence curves for mathematical test functions

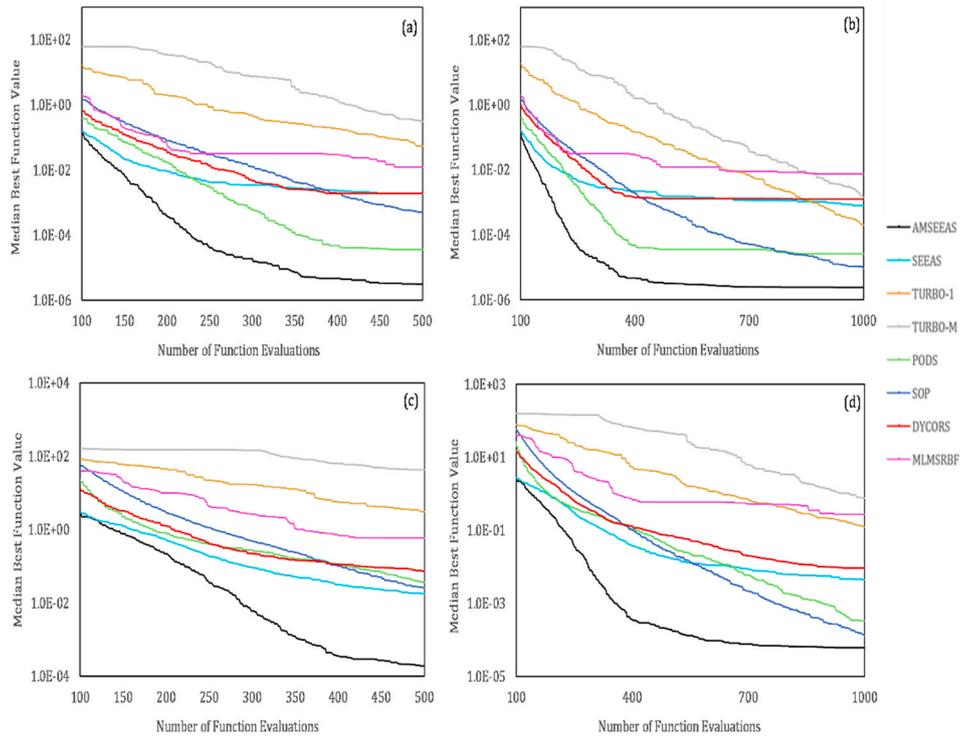


Fig. B1. Convergence curves for test function OF1 (Sphere) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

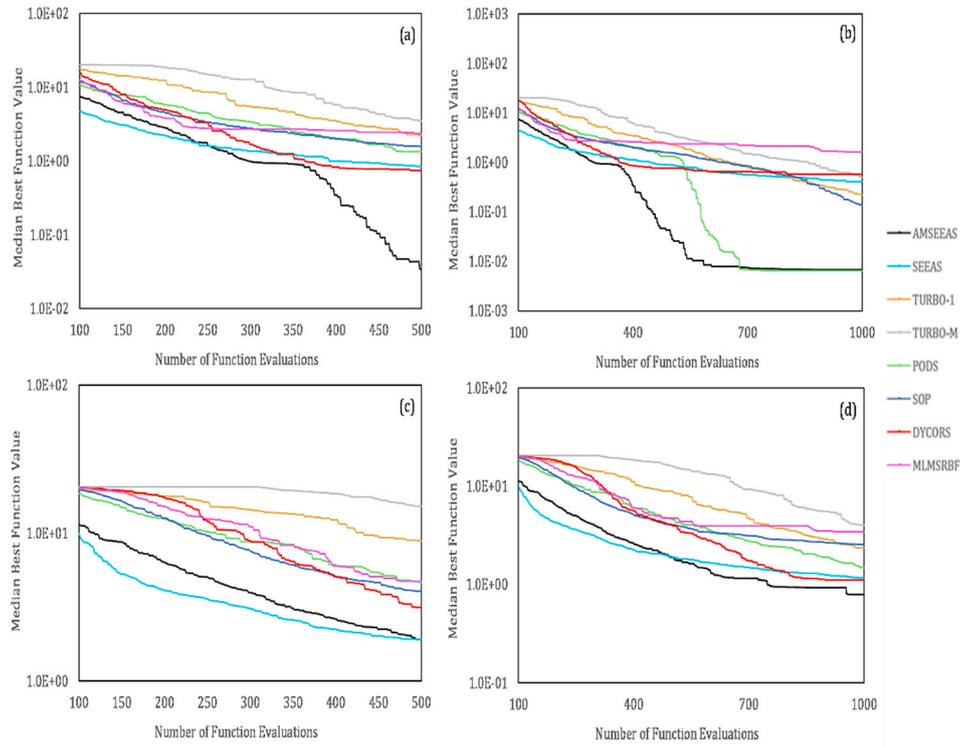


Fig. B2. Convergence curves for test function OF2 (Ackley) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

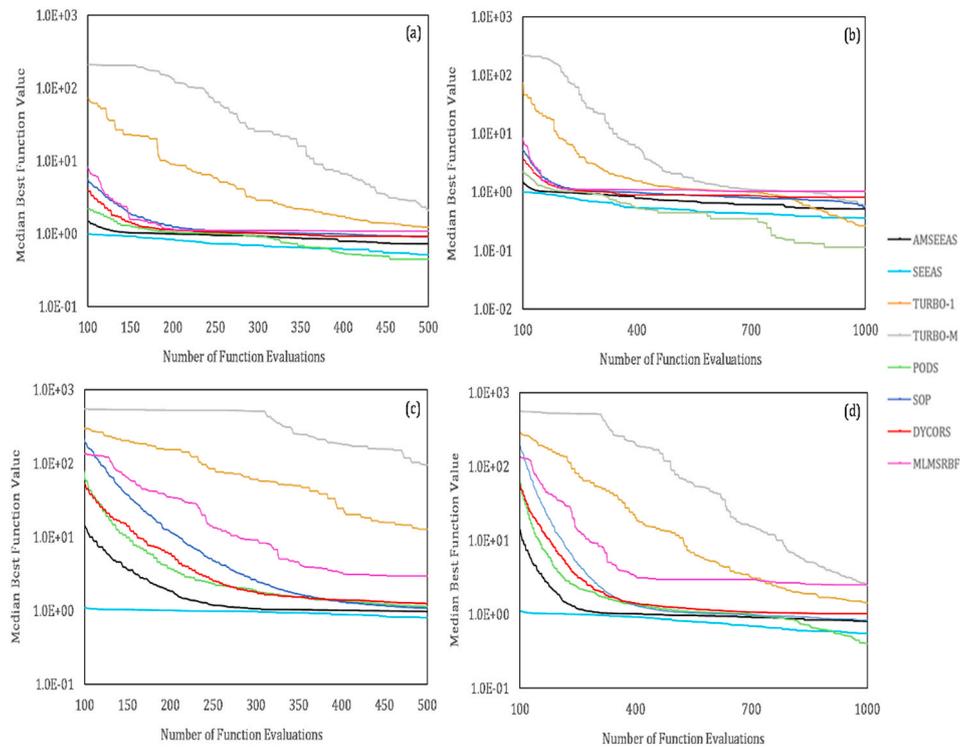


Fig. B3. Convergence curves for test function OF3 (Griewank) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

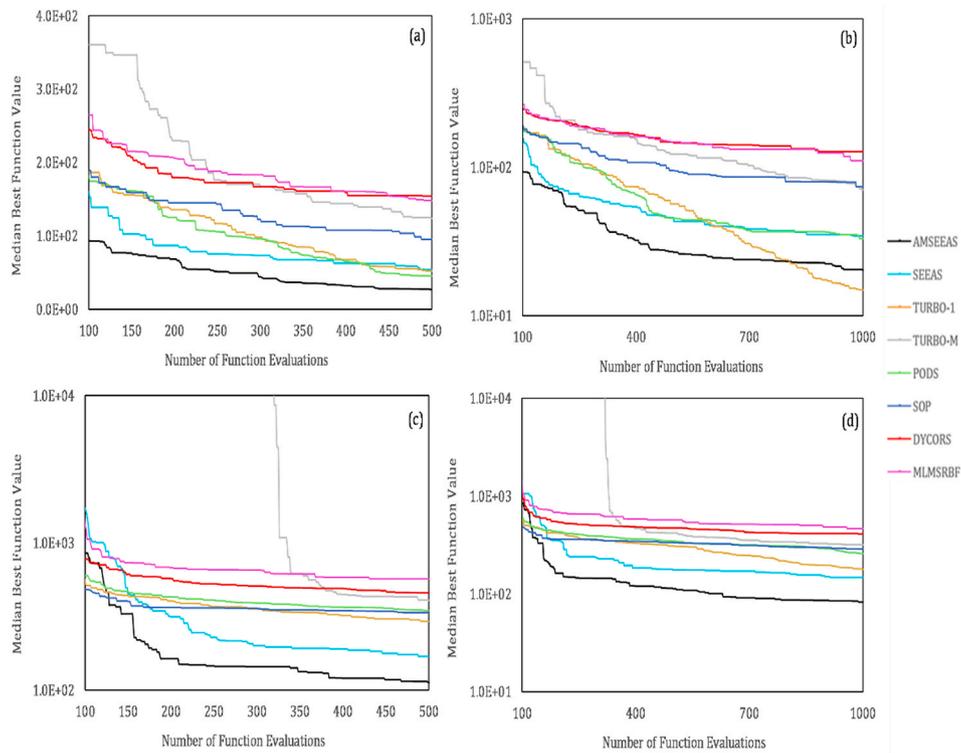


Fig. B4. Convergence curves for test function OF4 (Zakharov) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

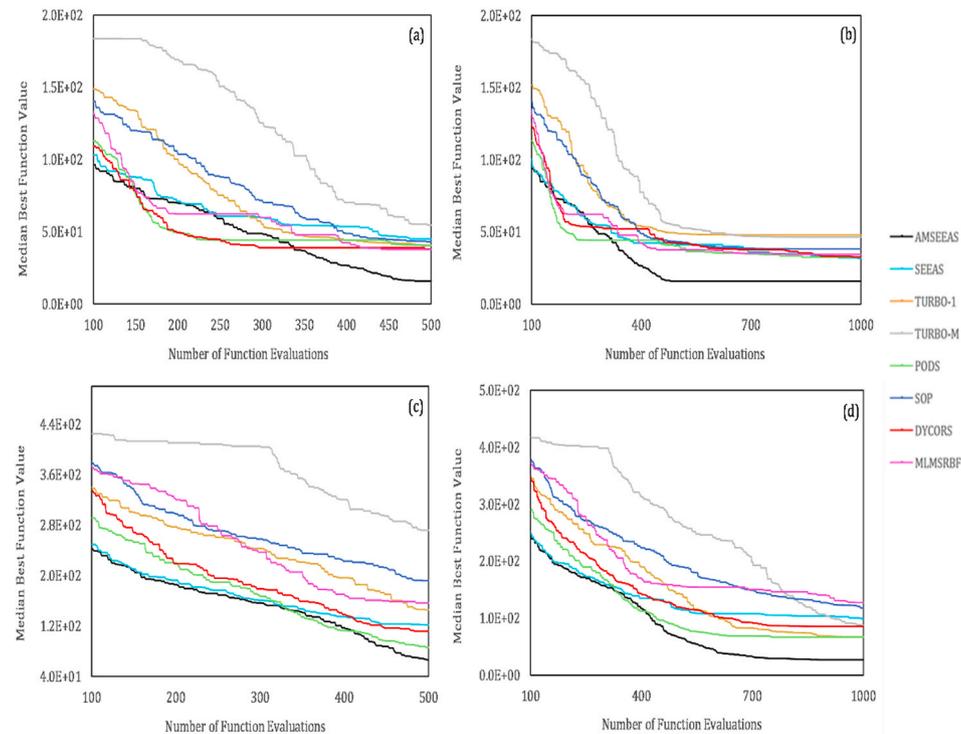


Fig. B5. Convergence curves for test function OF5 (Rastrigin) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

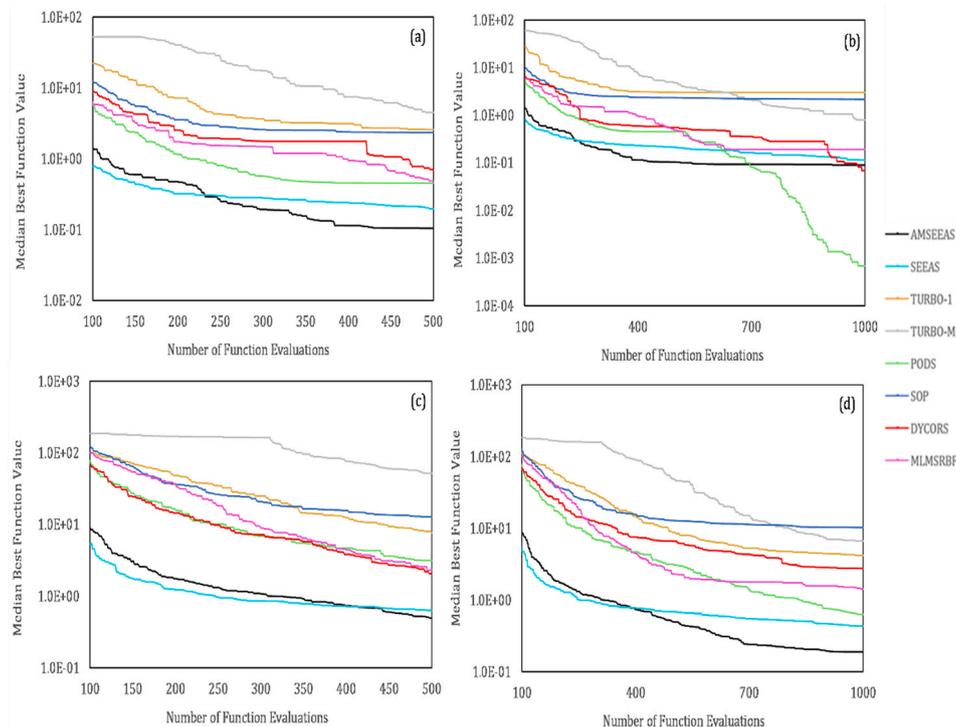


Fig. B6. Convergence curves for test function OF6 (Lévy) with 15 (a, b) and 30 variables (c, d), with $MFE = 500$ (a, c) and $MFE = 1000$ (b, d).

References

- Ahmad, A., El-Shafie, A., Razali, S.F.M., Mohamad, Z.S., 2014. Reservoir optimization in water resources: a review. *Water Resour. Manag.* 28 (11), 3391–3405. <https://doi.org/10.1007/s11269-014-0700-5>.
- Awad, N.H., Ali, M.Z., Mallipeddi, R., Suganthan, P.N., 2018. An improved differential evolution algorithm using efficient adapted surrogate model for numerical optimization. *Inf. Sci.* 451 (452), 326–347. <https://doi.org/10.1016/j.ins.2018.04.024>.
- Dong, H., Song, B., Wang, P., Huang, S., 2015. A kind of balance between exploitation and exploration on kriging for global optimization of expensive functions. *J. Mech. Sci. Technol.* 29 (5), 2121–2133. <https://doi.org/10.1007/s12206-015-0434-1>.
- Dong, H., Song, B., Dong, Z., Wang, P., 2016. Multi-start Space Reduction (MSSR) surrogate-based global optimization method. *Struct. Multidiscip. Optim.* 54 (4), 907–926. <https://doi.org/10.1007/s00158-016-1450-1>.
- Dong, H., Li, C., Song, B., Wang, P., 2018a. Multi-surrogate-based Differential Evolution with multi-start exploration (MDEME) for computationally expensive optimization. *Adv. Eng. Software* 123, 62–76. <https://doi.org/10.1016/j.advengsoft.2018.06.001>.
- Dong, H., Song, B., Wang, P., Dong, Z., 2018b. Hybrid surrogate-based optimization using space reduction (HSOSR) for expensive black-box functions. *Appl. Soft Comput.* 64, 641–655. <https://doi.org/10.1016/j.asoc.2017.12.046>.
- Dong, H., Wang, P., Yu, X., Song, B., 2021. Surrogate-assisted teaching-learning-based optimization for high-dimensional and computationally expensive problems. *Appl. Soft Comput.* 99, 106934. <https://doi.org/10.1016/j.asoc.2020.106934>.
- Duan, Qingyun, 2003. *Global Optimization for Watershed Model Calibration. Calibration Of Watershed Models*, vol. 6. American Geophysical Union, pp. 89–104.
- Dysarz, T., 2018. Application of Python scripting techniques for control and automation of HEC-RAS simulations. *Water* 10 (10), 1382. <https://doi.org/10.3390/w10101382>.
- Efstratiadis, A., Koutsyiannis, D., 2002. An evolutionary annealing-simplex algorithm for global optimisation of water resource systems. In: *Proceedings of the Fifth International Conference on Hydroinformatics*. International Water Association, Cardiff, UK, pp. 1423–1428. <https://doi.org/10.13140/RG.2.1.1038.6162>.
- Efstratiadis, A., Koutsyiannis, D., 2010. One decade of multi-objective calibration approaches in hydrological modelling: a review. *Hydrol. Sci. J.* 55 (1), 58–78. <https://doi.org/10.1080/02626660903526292>.
- Efstratiadis, A., Dimas, P., Pouliasis, G., Tsoukalas, I., Kossieris, P., Bellos, V., Sakki, G.-K., Makropoulos, C., Michas, S., 2022. Revisiting flood hazard assessment practices under a hybrid stochastic simulation framework. *Water* 14 (3), 457. <https://doi.org/10.3390/w14030457>.
- Eriksson, D., Pearce, M., Gardner, J.R., Turner, R., Poloczek, M., 2019. *Scalable Global Optimization via Local Bayesian Optimization*.
- Fowler, K.R., Reese, J.P., Kees, C.E., Dennis, J.E., Kelley, C.T., Miller, C.T., Audet, C., Booker, A.J., Couture, G., Darwin, R.W., Farthing, M.W., Finkel, D.E., Gablonsky, J.M., Gray, G., Kolda, T.G., 2008. Comparison of derivative-free optimization methods for groundwater supply and hydraulic capture community problems. *Adv. Water Resour.* 31 (5), 743–757. <https://doi.org/10.1016/j.advwatres.2008.01.010>.
- Fread, D.L., Ming, J., Janice, M.L., 1996. *An LPI Numerical Solution for Unsteady Mixed Flow Simulation*. American Society of Civil Engineers.
- Giunta, A., Wojtkiewicz, S., Eldred, M., 2003. Overview of modern design of experiments methods for computational simulations (invited). *41st aerospace sciences Meeting and exhibit*. <https://doi.org/10.2514/6.2003-649>.
- Golzari, A., Haghighat Sefat, M., Jamshidi, S., 2015. Development of an adaptive surrogate model for production optimization. *J. Petrol. Sci. Eng.* 133, 677–688. <https://doi.org/10.1016/j.petrol.2015.07.012>.
- Gong, W., Duan, Q., 2017. An adaptive surrogate modeling-based sampling strategy for parameter optimization and distribution estimation (ASMO-PODE). *Environ. Model. Software* 95, 61–75. <https://doi.org/10.1016/j.envsoft.2017.05.005>.
- Goodell, C., 2014. *BREAKING THE HEC-RAS CODE, first ed.*
- Jie, H., Wu, Y., Ding, J., 2015. An adaptive metamodel-based global optimization algorithm for black-box type problems. *Eng. Optim.* 47 (11), 1459–1480. <https://doi.org/10.1080/0305215X.2014.979814>.
- Jones, D.R., Schonlau, M., Welch, W.J., 1998. Efficient global optimization of expensive black-box functions. *J. Global Optim.* 13.
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. *Optimization by Simulated Annealing*, vol. 220.
- Krityakierne, T., Akhtar, T., Shoemaker, C.A., 2016. SOP: parallel surrogate global optimization with Pareto center selection for computationally expensive single objective problems. *J. Global Optim.* 66 (3), 417–437. <https://doi.org/10.1007/s10898-016-0407-7>.
- Kumar, V., Yadav, S.M., 2022. A state-of-the-Art review of heuristic and metaheuristic optimization techniques for the management of water resources. *Water Supply*. <https://doi.org/10.2166/ws.2022.010>.
- Labadie, J.W., 2004. Optimal operation of multireservoir systems: state-of-the-art review. *J. Water Resour. Plann. Manag.* 130 (2), 93–111. [https://doi.org/10.1061/\(ASCE\)0733-9496\(2004\)130:2\(93\)](https://doi.org/10.1061/(ASCE)0733-9496(2004)130:2(93)).
- Leon, A.S., Goodell, C., 2016. Controlling HEC-RAS using MATLAB. *Environ. Model. Software* 84, 339–348. <https://doi.org/10.1016/j.envsoft.2016.06.026>.
- Levy, H., 1992. Stochastic dominance and expected utility: survey and analysis. *Manag. Sci.* 38 (4), 555–593. <https://doi.org/10.1287/mnsc.38.4.555>.
- Liu, B., Zhang, Q., Gielen, G.G.E., 2014. A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems. *IEEE Trans. Evol. Comput.* 18 (2), 180–192. <https://doi.org/10.1109/TEVC.2013.2248012>.
- Lu, W., Xia, W., Shoemaker, C.A., 2022. Surrogate global optimization for identifying cost-effective green infrastructure for urban flood control with a computationally expensive inundation model. *Water Resour. Res.* 58 (4) <https://doi.org/10.1029/2021WR030928>.

- Maier, H.R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L.S., Cunha, M.C., Dandy, G.C., Gibbs, M.S., Keedwell, E., Marchi, A., Ostfeld, A., Savic, D., Solomatine, D.P., Vrugt, J.A., Zecchin, A.C., Minsker, B.S., Barbour, E.J., Kuczera, G., Pasha, F., et al., 2014. Evolutionary algorithms and other metaheuristics in water resources: current status, research challenges and future directions. *Environ. Model. Software* 62, 271–299. <https://doi.org/10.1016/j.envsoft.2014.09.013>.
- Mallipeddi, R., Lee, M., 2015. An evolving surrogate model-based differential evolution algorithm. *Appl. Soft Comput.* 34, 770–787. <https://doi.org/10.1016/j.asoc.2015.06.010>.
- Müller, J., 2020. An algorithmic framework for the optimization of computationally expensive bi-fidelity black-box problems. *INFOR Inf. Syst. Oper. Res.* 58 (2), 264–289. <https://doi.org/10.1080/03155986.2019.1607810>.
- Müller, J., Shoemaker, C.A., Piché, R., 2013. SO-MI: a surrogate model algorithm for computationally expensive nonlinear mixed-integer black-box global optimization problems. *Comput. Oper. Res.* 40 (5), 1383–1400. <https://doi.org/10.1016/j.cor.2012.08.022>.
- Nelder, J.A., Mead, R., 1965. A simplex method for function minimization. *Comput. J.* 8 (1) <https://doi.org/10.1093/comjnl/8.1.27>.
- Nicklow, J., Reed, P., Savic, D., Dessalegne, T., Harrell, L., Chan-Hilton, A., Karamouz, M., Minsker, B., Ostfeld, A., Singh, A., Zechman, E., 2010. State of the art for genetic algorithms and beyond in water resources planning and management. *J. Water Resour. Plann. Manag.* 136 (4), 412–432. [https://doi.org/10.1061/\(ASCE\)WR.1943-5452.0000053](https://doi.org/10.1061/(ASCE)WR.1943-5452.0000053).
- Pan, J.S., Liu, N., Chu, S.C., Lai, T., 2021. An efficient surrogate-assisted hybrid optimization algorithm for expensive optimization problems. *Inf. Sci.* 561, 304–325. <https://doi.org/10.1016/j.ins.2020.11.056>.
- Razavi, S., Tolson, B.A., Matott, L.S., Thomson, N.R., MacLean, A., Seglenieks, F.R., 2010. Reducing the computational cost of automatic calibration through model preemption. *Water Resour. Res.* 46 (11) <https://doi.org/10.1029/2009WR008957>.
- Reed, P.M., Hadka, D., Herman, J.D., Kasprzyk, J.R., Kollat, J.B., 2013. Evolutionary multiobjective optimization in water resources: the past, present, and future. *Adv. Water Resour.* 51, 438–456. <https://doi.org/10.1016/j.advwatres.2012.01.005>.
- Regis, R.G., 2014. Particle swarm with radial basis function surrogates for expensive black-box optimization. *f Comp. Sci.* 5 (1), 12–23. <https://doi.org/10.1016/j.jocs.2013.07.004>.
- Regis, R.G., Shoemaker, C.A., 2004. Local function approximation in evolutionary algorithms for the optimization of costly functions. *IEEE Trans. Evol. Comput.* 8 (5), 490–505. <https://doi.org/10.1109/TEVC.2004.835247>.
- Regis, R.G., Shoemaker, C.A., 2006. Improved strategies for radial basis function methods for global optimization. *J. Global Optim.* 37 (1) <https://doi.org/10.1007/s10898-006-9040-1>.
- Regis, R.G., Shoemaker, C.A., 2007. A stochastic radial basis function method for the global optimization of expensive functions. *Inf. J. Comput.* 19 (4) <https://doi.org/10.1287/ijoc.1060.0182>.
- Regis, R.G., Shoemaker, C.A., 2013. Combining radial basis function surrogates and dynamic coordinate search in high-dimensional expensive black-box optimization. *Eng. Optim.* 45 (5) <https://doi.org/10.1080/0305215X.2012.687731>.
- Ryan, C., 2003. Evolutionary algorithms and metaheuristics. In: *Encyclopedia of Physical Science and Technology*. Elsevier, pp. 673–685. <https://doi.org/10.1016/B0-12-227410-5/00847-4>.
- Shaw, A.R., Smith Sawyer, H., LeBoeuf, E.J., McDonald, M.P., Hadjerioua, B., 2017. Hydropower optimization using artificial neural network surrogate models of a high-fidelity hydrodynamics and water quality model. *Water Resour. Res.* 53 (11), 9444–9461. <https://doi.org/10.1002/2017WR021039>.
- Siqueira, V.A., Sorribas, M.V., Bravo, J.M., Collischonn, W., Lisboa, A.M.V., Trinidad, G. G.V., 2016. Real-time updating of HEC-RAS model for streamflow forecasting using an optimization algorithm. *RBRH* 21 (4), 855–870. <https://doi.org/10.1590/2318-0331.011616086>.
- Sun, R., Duan, Q., Mao, X., 2022. A multi-objective adaptive surrogate modelling-based optimization algorithm for constrained hybrid problems. *Environ. Model. Software* 148, 105272. <https://doi.org/10.1016/j.envsoft.2021.105272>.
- Tang, Y., Chen, J., Wei, J., 2013. A surrogate-based particle swarm optimization algorithm for solving optimization problems with expensive black box functions. *Eng. Optim.* 45 (5), 557–576. <https://doi.org/10.1080/0305215X.2012.690759>.
- Tsoukalas, I., Makropoulos, C., 2015a. Multiobjective optimisation on a budget: exploring surrogate modelling for robust multi-reservoir rules generation under hydrological uncertainty. *Environ. Model. Software* 69, 396–413. <https://doi.org/10.1016/j.envsoft.2014.09.023>.
- Tsoukalas, I., Makropoulos, C., 2015b. A surrogate based optimization approach for the development of uncertainty-aware reservoir operational rules: the case of nestos hydrosystem. *Water Resour. Manag.* 29 (13), 4719–4734. <https://doi.org/10.1007/s11269-015-1086-8>.
- Tsoukalas, I., Kossieris, P., Efstratiadis, A., Makropoulos, C., 2016. Surrogate-enhanced evolutionary annealing simplex algorithm for effective and efficient optimization of water resources problems on a budget. *Environ. Model. Software* 77. <https://doi.org/10.1016/j.envsoft.2015.12.008>.
- Wang, C., Duan, Q., Gong, W., Ye, A., Di, Z., Miao, C., 2014. An evaluation of adaptive surrogate modeling based optimization with two benchmark problems. *Environ. Model. Software* 60, 167–179. <https://doi.org/10.1016/j.envsoft.2014.05.026>.
- Wang, Q., Song, L., Chen, Y., Ma, G., Guo, Z., Li, J., 2022. KT-EGO: a knowledge transfer assisted efficient global optimization algorithm for solving high-dimensional expensive black-box problems. *Eng. Optim.* 1–19. <https://doi.org/10.1080/0305215X.2022.2139374>.
- Woolson, R.F., 2008. Wilcoxon signed-rank test. In: *Wiley Encyclopedia of Clinical Trials*. John Wiley & Sons, Inc. <https://doi.org/10.1002/9780471462422.eoc979>.
- Wu, B., Zheng, Y., Wu, X., Tian, Y., Han, F., Liu, J., Zheng, C., 2015. Optimizing water resources management in large river basins with integrated surface water-groundwater modeling: a surrogate-based approach. *Water Resour. Res.* 51 (4), 2153–2173. <https://doi.org/10.1002/2014WR016653>.
- Xi, M., Lu, D., Gui, D., Qi, Z., Zhang, G., 2017. Calibration of an agricultural-hydrological model (RZWQM2) using surrogate global optimization. *J. Hydrol.* 544, 456–466. <https://doi.org/10.1016/j.jhydrol.2016.11.051>.
- Xia, W., Shoemaker, C., Akhtar, T., Nguyen, M.-T., 2021. Efficient parallel surrogate optimization algorithm and framework with application to parameter calibration of computationally expensive three-dimensional hydrodynamic lake PDE models. *Environ. Model. Software* 135, 104910. <https://doi.org/10.1016/j.envsoft.2020.104910>.
- Yao, W., Chen, X.Q., Huang, Y.Y., van Tooren, M., 2014. A surrogate-based optimization method with RBF neural network enhanced by linear interpolation and hybrid infill strategy. *Optim. Methods Software* 29 (2), 406–429. <https://doi.org/10.1080/10556788.2013.777722>.
- Yazdi, J., Salehi Neyshabouri, S.A.A., 2014. Adaptive surrogate modeling for optimization of flood control detention dams. *Environ. Model. Software* 61, 106–120. <https://doi.org/10.1016/j.envsoft.2014.07.007>.
- Ye, K.Q., Li, W., Sudjianto, A., 2000. Algorithmic construction of optimal symmetric Latin hypercube designs. *J. Stat. Plann. Inference* 90 (1), 145–159. [https://doi.org/10.1016/S0378-3758\(00\)00105-1](https://doi.org/10.1016/S0378-3758(00)00105-1).