Surrogate-enhanced evolutionary annealing simplex algorithm for effective and efficient optimization of water resources problems on a budget

Ioannis Tsoukalas[a,*], Panagiotis Kossieris[a], Andreas Efstratiadis[a]
and Christos Makropoulos[a]

[a]Department of Water Resources and Environmental Engineering, National Technical University of Athens,
Heroon Polytechneiou 5, GR-157 80, Zographou, Greece

* Corresponding author, Tel. ++302107722886; e-mail: itsoukal@mail.ntua.gr

Revised 2nd version, December, 2015

Highlights

- The novel Surrogate-Enhanced Evolutionary Annealing Simplex algorithm (SEEAS) is proposed.
- Surrogate model is used as global search subroutine and also for identifying promising transitions within simplex-based operators.
- SEEAS outperforms alternative algorithms in six test functions, for 15D and 30D formulations and for two budgets (500 and 1000 function evaluations).
- SEEAS effectively handles the peculiarities of two typical water resources optimization problems, i.e., hydrological calibration and multi-reservoir management.

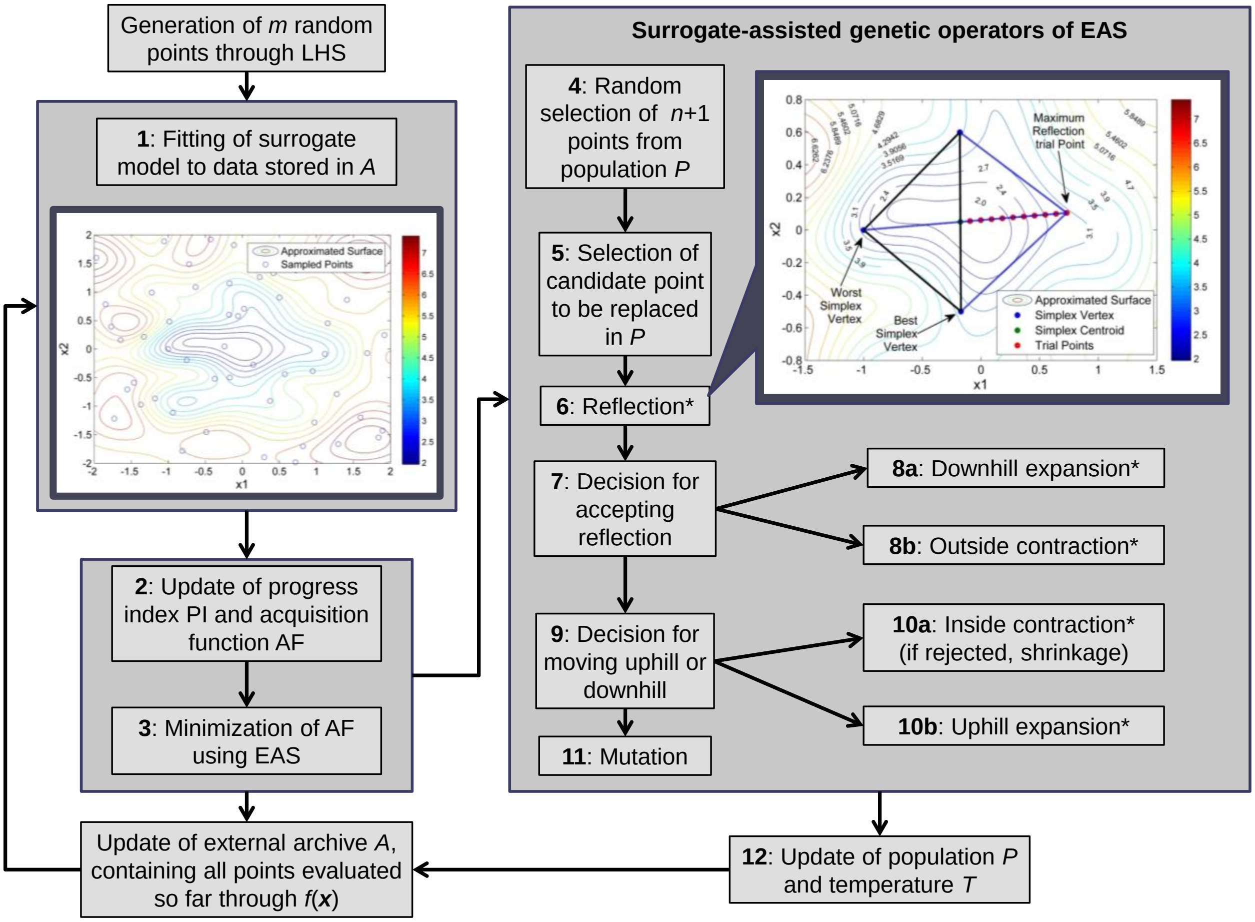
Surrogate-enhanced evolutionary annealing simplex algorithm for effective and efficient optimization of water resources problems on a budget

Ioannis Tsoukalas[a,*], Panagiotis Kossieris[a], Andreas Efstratiadis[a] and Christos Makropoulos[a]

Abstract

In water resources optimization problems, the objective function usually presumes to first run a simulation model and then evaluate its outputs. However, long simulation times may pose significant barriers to the procedure. Often, to obtain a solution within a reasonable time, the user has to substantially restrict the allowable number of function evaluations, thus terminating the search much earlier than required. A promising strategy to address these shortcomings is the use of surrogate modelling techniques. Here we introduce the Surrogate-Enhanced Evolutionary Annealing-Simplex (SEEAS) algorithm that couples the strengths of surrogate modelling with the effectiveness and efficiency of the evolutionary annealing-simplex method. SEEAS combines three different optimization approaches (evolutionary search, simulated annealing, downhill simplex). Its performance is benchmarked against other surrogate-assisted algorithms in several test functions and two water resources applications (model calibration, reservoir management). Results reveal the significant potential of using SEEAS in challenging optimization problems on a budget.

Graphical abstract



Keywords

global optimization; meta-models; radial basis functions; hydrological calibration; multi-reservoir management; synthetic data

Software availability

EAS and SEEAS are available online at: http://www.itia.ntua.gr/en/softinfo/29/

1 Introduction

Coupling of simulation and optimization is a powerful technique that has gained significant attention in water resources science and technology, since it ensures great advantages over the traditional individual implementation of the two approaches (e.g., Koutsoyiannis and Economou, 2003). In this context, a simulation model is used to faithfully represent the dynamics of the system under study in subsequent time steps and next to evaluate its overall performance against one or more user-specified criteria. Provided that these criteria are expressed in terms of objective function, simulation can be driven by an optimization model, which employs systematic search through the parameter (or decision) space to maximize the system performance; at each trial, new values are assigned to the control variables of the simulation model, which runs automatically to update the value of the objective function.

Combined simulation-optimization schemes for water resource systems can be generally classified into two categories: (a) decision-making problems, in which the system properties and associated processes are known a priori, but either some of its design quantities or its management policy are unknown; and (b) calibration problems, in which some internal properties of the system, either physical or conceptual, are unknown and have to be inverted by minimizing the departures of the simulated responses against the observed ones. Despite their different rationale, both types of problems suffer from significant uncertainties and complexities, and they are subject to multiple (and often conflicting) criteria as well as numerous constraints.

For convenience, we consider that all criteria are aggregated in a single objective function representing a global performance measure of the system (an alternative approach would require the formulation of a multiobjective function and the identification of acceptable tradeoffs among conflicting criteria, which is not the case here). We also assume that all "internal" constraints (i.e., constraints associated with the system dynamics) are handled through the simulation model (Koutsoyiannis and Economou, 2003), while any additional "external" constraints, which are usually associated with decision-making problems, are embedded in the objective function, typically as penalty terms. Under this premise, the combined simulation-optimization problem is formalized as the determination of the global optimum (for convenience, minimum) of a nonlinear objective function $f(\boldsymbol{x})$, where $f(\cdot)$ represents the simulation model and $\boldsymbol{x}$ is the vector of control variables. The search space is a hypervolume, since the unique constraints of the problem are the lower and upper bounds of parameters. As $f(\boldsymbol{x})$ is a black-box function, its analytical expression as well as its derivatives are not available, which prohibits the use of gradient-based optimization. Given also that, due to uncertainties and complexities of the system, $f(\boldsymbol{x})$ is non-convex, and thus multimodal (i.e., it contains multiple local optima), derivative-free methods combined with stochastic search approaches are essential to solve this so-called global optimization problem.

The need for advanced global optimization tools (e.g., evolutionary algorithms) has been early recognized by the hydrological community, which has significant experience in their use and also remarkable contribution in their development. In the literature are found numerous reviews of optimization approaches in such problems. For instance, in the context of water resources planning and management, we distinguish the works by Labadie (2004), Fowler et al. (2008), Nicklow et al. (2010), Reed et al. (2013) (emphasis to multiobjective applications) and Ahmad et al. (2014). The literature for hydrological calibration is even more extended. For convenience, we highlight the recent works by Duan (2013) and Efstratiadis and Koutsoyiannis (2010), who provide a comprehensive review of global and multiobjective calibration approaches, respectively. It is also worth mentioning the article by Maier et al. (2014), who summarize the current status of evolutionary algorithms and other metaheuristics, and highlight new directions for future research across water resources applications.

Apparently, in the whole computational procedure, simulation is by far the most time-consuming component. As models become more complex and data-demanding, their requirements in computational time and/or CPU increase substantially (e.g., Tolson and Shoemaker, 2007; Keating et al., 2010; Razavi et al., 2010; Tsoukalas and Makropoulos, 2015a). Typical example is the case of physically-based hydrological models of fine spatial and temporal resolution, in contrast to lumped conceptual rainfall-runoff models. In other applications, referred to as stochastic simulation problems, the computational effort increases two or three orders of magnitude due to the use of synthetic (instead of historical) time series of very large length (e.g., thousands of years), in order to provide estimations for probabilistic quantities (e.g., reliability, risk) with satisfactory accuracy. On the other hand, depending on the number of parameters and the irregularity of the response surface, the optimization algorithm may need to call the simulation model hundreds or thousands of times, in order to converge to a good solution. Therefore, the time effort of simulation imposes a practical barrier to optimization, which is necessary to run with significantly restricted "budget", by means of maximum allowable number of function evaluations. Consider a simulation model that requires approximately 1.5 minutes for a single simulation run and an optimization algorithm that requires 10,000 function evaluations (iterations) to approximate the global minimum. Consequently, the procedure would last more than ten days, which makes it practically infeasible.

According to Razavi et al. (2010), the approaches to alleviate the computational burden imposed by time-consuming simulation models are classified into four main categories: (1) parallel computing (e.g., Schutte et al., 2004; Cheng et al., 2005; Vrugt et al., 2006; Feyen et al., 2007; He et al., 2007; Regis and Shoemaker, 2009; Dias et al., 2013); (2) computationally efficient optimization algorithms (e.g., Tolson and Shoemaker, 2007; Kuzmin et al., 2008; Tan et al., 2008; Tolson et al., 2009); (3) strategies to avoid opportunistically (expensive) model evaluations (e.g., Ostfeld and Salomons, 2005; Razavi et al., 2010; Matott et al., 2012); and (4) surrogate modelling techniques, also referred to as meta-modelling, function approximation, response surface modelling and model emulation (Razavi et al., 2012a), where surrogate approaches are used to approximate the responses of the original simulation model. Parallel computing, allowing the execution of independent simulations by multiple processors, requires significant investments in hardware infrastructure, which makes it impractical for common use. We remark that in order to reduce the entire time of computations three orders of magnitude – a reasonable requirement when dealing with complex simulation models – 1000 parallel processors should be used, which is far from realistic. The other two options, i.e., the improvement of efficiency of existing algorithms, as well as the interruption of the function evaluation procedure, when the model performance seems to be very poor from early steps of simulation, may save some time but not as much as required. On the other hand, surrogate models do not have any specific requirements in computer resources and also ensure very fast computations, since they replace, to some context, the (expensive) simulation model. Their key objective is to generate models that are accurate in a certain region of the search space (i.e., around a potential optimum) and thus intelligently guide the optimization (Couckuyt et al., 2013).

Although response surface approaches go back to 70's (Blanning, 1975), surrogate-based optimization methods have been popularized since the pioneering work by Jones et al. (1998), who developed the Efficient Global Optimization (EGO) algorithm. EGO uses Kriging as surrogate model and an acquisition function (named Expected Improvement), in order to locate potential good samples that should be evaluated through expensive simulation functions (Sacks et al., 1989; Jones et al., 1998). Later, Sasena et al. (2002) implemented and investigated various acquisition functions for EGO. Literature also reports multi-objective versions of EGO (e.g., Knowles, 2005; Ponweiser et al., 2008; Couckuyt et al., 2013).

Other commonly used surrogate models are Radial Basis Functions (RBFs - Powell, 1992; Buhmann, 2003), polynomials (Myers and Montgomery, 1995), artificial neural networks, and support vector machines (Cortes and Vapnik, 1995; Dibike et al., 2001). The use RBFs within the context of evolutionary algorithms was popularized after the publication Regis and Shoemaker (2004). Other typical examples of RBFs are the Multistart Local Metric Stochastic RBF (MLMSRBF) and the ConstrLMSRBF, which handles inequality constraints (Regis and Shoemaker, 2007b; Regis, 2011). Additionally, Regis (2014) and Tang et al. (2012) proposed hybridizations of the particle swarm optimization algorithm (Kennedy and Eberhart, 1995) that use RBFs to assist the search. Shoemaker et al. (2007) developed an evolutionary algorithm that uses an RBF approximation and benchmarked its performance against several test problems, with dimensions ranging from 8-D to 14-D. Finally, Regis and Shoemaker (2013) developed the DYnamic COordinate Search (DYCORS) that uses Response Surface models to handle high-dimensional expensive optimization problems. DYCORS was benchmarked against other RBF-based algorithms in a variety of test problems, ranging from 14-D to 200-D.

Comprehensive reviews of surrogate-based optimization methods can be found in the broader optimization literature (e.g., Jin, 2005; Forrester and Keane, 2009; Jin, 2011). There are also reported several successful applications in time-demanding hydrological problems (e.g., Broad et al., 2005; Mugunthan et al., 2005; Mugunthan and Shoemaker, 2006; Regis and Shoemaker, 2007a; Zou et al., 2007; Kourakos and Mantoglou, 2009; Tsoukalas and Makropoulos, 2015a). Razavi et al. (2012b) summarize the use of surrogate modeling techniques in water resource systems, also classifying the existing meta-modeling frameworks.

It is important to remark that in the context of combined simulation-optimization schemes, surrogate models play the role of black-box approaches that aim establishing a data-driven relationship between the control variables of the simulation model (i.e., explanatory variables) and the objective function of the optimization model (i.e., response variable). Therefore, they clearly do not intend to reproduce the dynamic behavior of the original simulation model (Razavi et al., 2012b). In fact, the task of reproducing the dynamic behavior of the simulation model is performed by a quite different surrogate modelling approach, generally referred to as model reduction or reduced-order modelling. This yields a low-order, dynamic "equivalent" of the simulation model, by preserving, to some extent, the state-space representation of the original model and allowing a physical interpretation of its structure (Castelletti et al., 2012a; Castelletti et al., 2012b).

This paper introduces the Surrogate-Enhanced Evolutionary Simplex-Annealing approach (SEEAS), which is a novel global optimization algorithm, focused on time-expensive functions. Our motivation arises from challenging simulation-optimization problems that are commonly found in water resources, and they impose, in the everyday practice, very limited computational budgets, e.g., of few hundred function evaluations. SEEAS has been designed for both types of such problems, i.e., decision-making and calibration, suffering from different peculiarities and complexities, which are in turn reflected in the different geometry of the associated response surfaces.

SEEAS is built upon the Evolutionary Annealing-Simplex (EAS) method (Efstratiadis and Koutsoyiannis, 2002), which is a hybrid scheme combining global and local search strategies and assisted by a RBF surrogate model. SEEAS uses an external archive to maintain all visited solutions in order to formulate, update and exploit the surrogate model during search. There are also some improvements in the key core of EAS, regarding the simplex transitions and the mutation operator. SEEAS is compared and benchmarked against the original version of EAS and three state-of-the-art optimization algorithms that are mentioned before, i.e., DDS (Tolson and Shoemaker, 2007), MLMSRBF (Regis and Shoemaker, 2007b), and DYCORS-LMSRBF (Regis and Shoemaker, 2013). Evaluations are made on the basis of 12 mathematical problems (i.e., six test functions for two alternative dimensions, 15-D and 30-D), a hydrological calibration problem with 11 parameters, configured with both real and synthetic data, and a multi-reservoir management problem with 20 decision variables, using synthetic inflows of 500 years length. The use of synthetic data is one of the novelties of our testing framework. Moreover, most of the known surrogate-based schemes have been only evaluated in calibration problems and not in time-demanding water management applications, with few exceptions (e.g., Razavi et al., 2012b; Tsoukalas and Makropoulos, 2015a).

The results of this extended analysis are very encouraging, since the proposed method is effective and efficient, in terms of locating a satisfactory solution as close as possible to the global optimum, within reasonable computational time, and clearly outperforms the other examined approaches, in almost all tests.

2 Optimization methodology

2.1 Evolutionary Annealing-Simplex

EAS[1] is a heuristic, population-based global optimization technique, originally developed by Efstratiadis and Koutsoyiannis (2002), that couples the strength of simulated annealing in rough search spaces along with the efficiency of the downhill simplex method (Nelder and Mead, 1965) in smoother spaces. Its key idea is the introduction of an external variable T, which plays a role similar to temperature in a real-world annealing process, and determines the degree of randomness of the search procedure. This is expressed through a stochastic term that is relative to temperature and is added to the initial objective function $f(\boldsymbol{x})$, thus getting a modified function $g(\boldsymbol{x}) = f(\boldsymbol{x}) \pm \boldsymbol{u}T$ (where $\boldsymbol{u}$ is a vector of uniformly distributed random numbers). Search is based on an evolving population of feasible points, where critical decisions are driven by the modified function. The genetic operators are either quasi-stochastic geometric transformations, inspired by the downhill simplex method, or fully-probabilistic transitions (mutations). As search proceeds, the system temperature reduces according to an adaptive annealing cooling schedule, and all transitions become more deterministic.

EAS has been successfully employed in several hydrological applications (e.g., Rozos et al., 2004; Nalbantis et al., 2011; Kossieris et al., 2013; Efstratiadis et al., 2014b). It has been also incorporated within advanced modelling tools, i.e., Hydronomeas (Efstratiadis et al., 2004), Hydrogeios (Efstratiadis et al., 2008) and HyetosR (Kossieris et al., 2012) to solve challenging simulation-optimization problems. The original algorithm has been also adapted to handle

[1] EAS and SEEAS are available online at: http://www.itia.ntua.gr/en/softinfo/29/

multiobjective problems (Efstratiadis and Koutsoyiannis, 2008) and stochastic (i.e., noisy) objective functions (Kossieris et al., 2013). Here we introduce an improved verison of EAS, called Surrogate-Enhanced Evolutionary Annealing-Simplex (SEEAS) algorithm, which is presented in detail herein.

2.2 Surrogate-Enhanced Evolutionary Annealing-Simplex

2.2.1 Overview of SEEAS algorithm

The algorithm is a surrogate-enhanced extension of EAS in a way that builds, maintains and exploits surrogate modelling (SM) techniques that generate approximated response surfaces, which allow effectively guiding search towards promising areas of the real response surface. The model used is the RBF, which is a well-known interpolation technique (Figure 1, left). During the iterative procedure, the algorithm maintains an external archive of all visited points, already evaluated through the (expensive) objective function. This archive is used to update the SM, in an attempt to progressively provide more accurate approximations of the current region of interest (i.e. the area around the current best point). In SEEAS, the surrogate model has a double role. The first is providing new points that are added to the current population, and the second is assisting the genetic operators of the downhill simplex scheme to identify suitable directions across the search space (e.g., favorable slopes and new areas of attraction).

In order to balance exploration (i.e., detailed sampling) and exploitation (i.e., blind use of SM), SEEAS uses a weighted metric, termed acquisition function (AF), which accounts for the predictions provided by the SM as well as the spread of all previously evaluated points (by means of a distance quantity). In opposite to common practices that use a standard expression of the AF with constant weights, in our approach the weights are dynamically adjusted, thus improving the efficiency of the algorithm. Details about the acquisition function (AF) are given in Section 2.2.3.

SEEAS follows an iterative search procedure. At the end of each iteration cycle (or generation, according to the terminology of evolutionary theory), we obtain at least one new point that enters the population and replaces one of its existing members. A typical iteration cycle of SEEAS starts by fitting the surrogate model to the current population (initially, this population is randomly generated through Latin Hypercube Sampling, LHS). Next, we run an internal global optimization algorithm (particularly, the original version of EAS) across the surrogate response surface, using as objective the acquisition function (AF), in order to locate a candidate solution to enter the population (provided that this solution outperforms the current worst point). Thereafter, we follow a search procedure that is mostly based on the genetic operators of EAS, enhanced by surrogate-assisted steps in simplex-based transformations.

The general idea is to utilize the information gained by the SM, in order to enhance the current knowledge in the selection of simplex transitions. A characteristic example involving the reflection step is illustrated in Figure 1, right (for simplicity, we demonstrate the predictions of the surrogate model and not the AF). In the original version of EAS, after specifying the direction of reflection (defined by the difference between the worst vertex of the simplex and the centroid of all rest vertices), the algorithm employs a blind trial-and-error procedure, i.e., it generates subsequent random points along this direction and evolves according to their values. In this scheme, the original objective function is called whenever a new trial point is generated. Since the expansion continues as long as the function value improves, this procedure may be quite expensive, in terms of function evaluations. In opposite, in SEEAS we employ a candidate screening procedure using the SM, which allows making multiple trials with negligible computational cost and guiding search using all prior information. Similar screening is employed within all simplex transformations (except shrinkage), thus providing significant aid to the associated decisions.

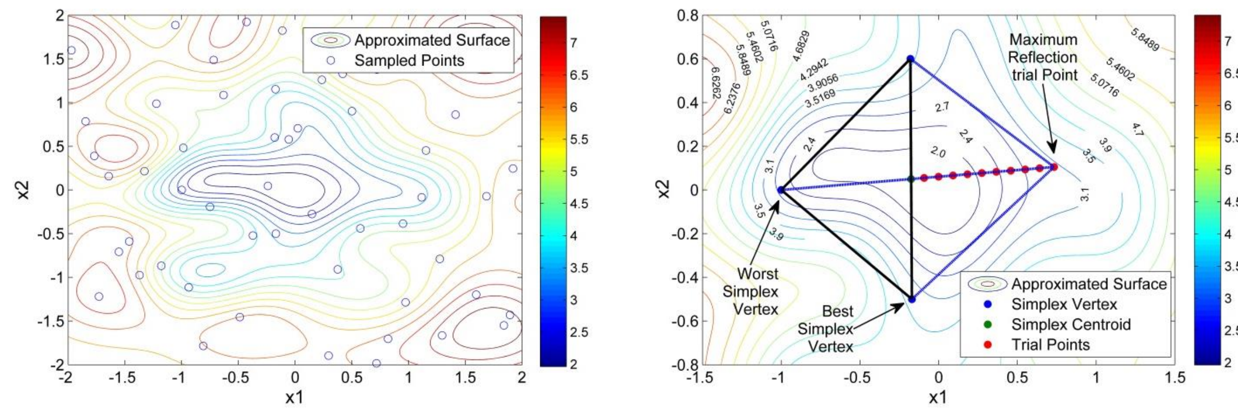


Figure 1: Approximated surface (RBF) in a 2-D example (Ackley function) using all available sample points (left panel). The right panel demonstrates a randomly selected simplex and the modified surrogate-enhanced reflection movement using candidate points on the line formed from the simplex centroid and the maximum reflection point. The simplex is reflected at the candidate point with the minimum function value.

2.2.2 Surrogate model (RBF)

SEEAS implements the Radial Basis Function (RBF) interpolation method (Powell, 1992; Buhmann, 2003), and more specifically the RBF with cubic basis functions and linear polynomial tail. This is a commonly used surrogate model of proven effectiveness, as reported in numerous studies (e.g., Mugunthan et al., 2005; Regis and Shoemaker, 2007a, b; Shoemaker et al., 2007; Regis and Shoemaker, 2013; Müller and Shoemaker, 2014).

The computational procedure of RBF is the following. Given N_s samples $\boldsymbol{x} \in R^n$ with response $\boldsymbol{y}$, we get the pairs $(\boldsymbol{x}_i, y_i)$. The prediction $s(\boldsymbol{x})$ of RBF model at sample point $\boldsymbol{x}$ is given by:

$$s(\boldsymbol{x}) = \sum_{i=1}^{Ns} \lambda_i \, \varphi\left(||\boldsymbol{x} - \boldsymbol{x}_i||\right) + p(\boldsymbol{x}) \tag{1}$$

where $\lambda_i \in R$, φ is a basis function of the form $\varphi(r) = r^3$, $||.||$ is the Euclidean distance (norm) and $p(\boldsymbol{x})$ is a polynomial tail of the form $p(\boldsymbol{x}) = \boldsymbol{b}^T\boldsymbol{x} + a$, where $\boldsymbol{b} = (b_1, \ldots, b_n)^T$ and $a \in R$. The model parameters $\boldsymbol{\lambda}$, $\boldsymbol{b}$, and a are determined by solving the linear system:

$$\begin{bmatrix} \boldsymbol{\Phi} & \boldsymbol{P} \\ \boldsymbol{P}^T & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{c} \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{0} \end{bmatrix} \tag{2}$$

where $\boldsymbol{\Phi}$ is an $N_s \times N_s$ matrix with elements $\varphi_{ij} = \varphi(||\boldsymbol{x}_i - \boldsymbol{x}_j||)$, $\boldsymbol{P}$ is a $N_s \times (n + 1)$ matrix, the i^{th} row of which is $(1, \boldsymbol{x}_i^T)$, $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{Ns})^T$, $\boldsymbol{c} = (b_1, \ldots, b_n, a)^T$, and $\boldsymbol{y} = (y_1, \ldots, y_{Ns})^T$. We mention that the matrix of Eq. (2) is invertible if and only if Rank($\boldsymbol{P}$) = n + 1 (Powell, 1992).

2.2.3 Acquisition function

Acquisition functions (AF) are well-established techniques, aiming to balance exploration-exploitation in surrogate-based optimization algorithms (e.g., Sasena et al., 2002; Forrester and Keane, 2009). SEEAS implements a novel scheme, in which the weights are automatically adjusted during the iterative process, according to the current number of function evaluations and the maximum allowed number of evaluations.

Consider a set of N_s points, $\boldsymbol{x}_s^j$, with known response value, $f(\boldsymbol{x}_s^j)$, and another set of N_c points $\boldsymbol{x}_c^i$, with approximated response values $s(\boldsymbol{x}_c^i)$. The latter are conventionally called candidate points, in the sense that they are used within infilling or internal search procedures, e.g., selection of the most appropriate reflection point in the graphical example of Figure 1. The acquisition function is estimated as follows:

Step A: Standardize the approximated response values of all candidate solutions by setting $s^*(\boldsymbol{x}_c^i) = [s(\boldsymbol{x}_c^i) - s^{min}]/[s^{max} - s^{min}]$, where s^{min} and s^{max} are the corresponding minimum and maximum values.

Step B: Calculate the minimum Euclidean distance of each candidate point $\boldsymbol{x}_c^i$ from all previously evaluated points, $\boldsymbol{x}_s^j$, i.e., $d_i = d^*(\boldsymbol{x}_c^i) = \min_{1 \le j \le NS}||\boldsymbol{x}_c^i - \boldsymbol{x}_s^j||$, and standardize them by setting $d_i^* = (d_i - d^{min})/(d^{max} - d^{min})$, where d^{min} and d^{max} are the corresponding minimum and maximum distances.

Step C: Calculate the weighted value of AF for every candidate point using the formula:

$$\mathrm{AF}_i = w\, s^*(\boldsymbol{x}_c^i) + (1 - w)\, d^*(\boldsymbol{x}_c^i) \tag{3}$$

where w is a dimensionless weighting coefficient, ensuring balance between exploitation and exploration. To finalize the infilling routine, the candidate with the minimum AF value will be selected and assessed through the objective function. As mentioned before, the minimization of the AF across the surrogate search space is carried out through the original EAS algorithm.

2.2.4 Detailed description of SEEAS

Let $f(\boldsymbol{x})$ be a nonlinear objective function in the feasible space $\boldsymbol{x}_L \le \boldsymbol{x} \le \boldsymbol{x}_U$, where $\boldsymbol{x}$ is an n-dimensional vector of continuous control variables (in practice, $f(\boldsymbol{x})$ represents the performance measure of a simulation model). For convenience, we search for the global minimum of $f(\boldsymbol{x})$, allowing a budget of MFE function evaluations. The algorithm uses two archives. The first is the population $P^{[t]}$, which is evolved during the search procedure (where t denotes the iteration cycle or generation), and the second is the so-called external archive $A^{[t]}$, which contains all visited points from the beginning of the optimization ($t = 0$), including the members of the current population. Whenever a new point $\boldsymbol{x}$ is evaluated through the objective function $f(\boldsymbol{x})$, it enters the archive $A^{[t]}$ (the archive may be updated several times within a generation). At the beginning of each new generation t, the surrogate model is re-evaluated by considering the current elements of $A^{[t]}$. The size of the population is $m \geq n + 1$ (i.e., the minimum number of points required to fit a RBF with linear polynomial as well as to formulate a simplex in the n-dimensional space), and remains constant, while the size of the external archive progressively increases, thus ensuring more accurate approximations of the response surface and, consequently, more reliable predictions. The initial population $P^{[0]}$ is generated via the Latin Hypercube Sampling (LHS) technique, which ensures satisfactory spread across the feasible space (Giunta et al., 2003). Apparently, the initial archive $A^{[0]}$ is identical to $P^{[0]}$.

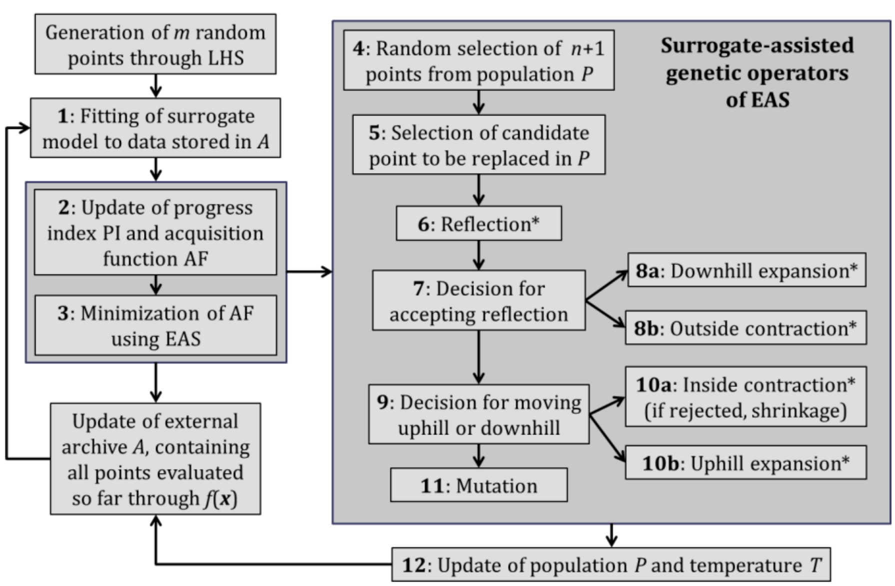


Figure 2: Outline of SEEAS algorithm following the steps explained in section 2.2.4 (* denotes the use of the surrogate model within the associated simplex transformations).

Similarly to EAS, the surrogate-enhanced algorithm also uses an auxiliary parameter, $T^{[t]}$, called temperature. The concept originates from simulated annealing, where the key role of temperature is ensuring balance between randomness and determinism. In SEEAS, temperature is dynamically adjusted (i.e., reduced) using empirical rules, considering the extreme values, $f_{\min}^{[t]}$ and $f_{\max}^{[t]}$, of the current population $P^{[t]}$, and a dimensionless progress index, defined as

$$\mathrm{PI} = \log(\mathrm{FE})/\log(\mathrm{MFE}) \tag{4}$$

where FE is the current number of function evaluations and MFE is the maximum allowable number of FE, which is a user-specified termination criterion.

A typical iteration cycle of SEEAS, an outline of which is illustrated in Fig. 2, comprises the following steps (generation index t is omitted for simplicity):

Step 1: The interpolation surface $s(\boldsymbol{x})$ is updated using the current information stored in the external archive A (i.e., all points evaluated so far through the original objective function).

Step 2: The weighting coefficient of the AF is updated using the empirical formula:

$$w = \max[0.75, \min(\mathrm{PI}, 0.95)] \tag{5}$$

The above formula ensures that at the early stages of optimization, more weight is given to exploration (up to 0.25), but gradually its contribution diminishes thus not exceeding 0.05.

Step 3: A new point $\boldsymbol{x}_{\mathrm{p}}$ is generated by minimizing AF, using the original version of EAS for internal optimization. The new point is evaluated through $f(\boldsymbol{x})$ and replaces the worst point of the current population, if the latter is worse (higher) than $f(\boldsymbol{x}_{\mathrm{p}})$.

Step 4: A set of $n + 1$ points is randomly selected from the current population, in order to formulate the vertices of a simplex in the n-dimensional search space, symbolized $S = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n+1}]$. The elements of S are sorted such as $f(\boldsymbol{x}_1)$ corresponds to the best (lowest) and $f(\boldsymbol{x}_{n+1})$ to the worst value of the objective function.

Step 5: From the subset $[\boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n+1}]$ we select a candidate point $\boldsymbol{x}_{\mathrm{w}}$ to be replaced in the population, based on the modified, quasi-stochastic objective function:

$$g(\boldsymbol{x}) = f(\boldsymbol{x}) + u\,T \tag{6}$$

where u is a uniform random number in the interval [0, 1]. By adding the stochastic component $u\,T$ to the objective function $f(\boldsymbol{x})$, the algorithm behaves as in between random and downhill search. At the early stages of optimization, when temperature is still high, any point except for the best one can be replaced. On the other hand, in the limiting case $T \rightarrow 0$, the actually worst point, i.e., $\boldsymbol{x}_{n+1}$, is replaced, as considered in the original downhill simplex method.

Step 6: A set of N_r trial points $\boldsymbol{x}_{cr}^{k}$ are generated by reflecting the simplex according the formula:

$$\boldsymbol{x}_{cr}^{k} = \boldsymbol{g} + (0.5 + \delta_k)\,(\boldsymbol{g} - \boldsymbol{x}_\mathrm{w}) \tag{7}$$

where $\boldsymbol{g}$ is the centroid of the subset $[\boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n+1}]$ and δ_k is a scale coefficient equally spread in the interval [0, 1], thus $\delta_k = (k - 1)/(N_\mathrm{r} - 1)$, for $k = 1, \ldots, N_\mathrm{r}$. Among all candidates, we select the one that minimizes AF, which we will next call the reflection point, $\boldsymbol{x}_\mathrm{r}$. The reflection point is evaluated on the basis of the objective function and enters the external archive.

Step 7: If $f(\boldsymbol{x}_\mathrm{r}) < f(\boldsymbol{x}_\mathrm{w})$, we replace $\boldsymbol{x}_\mathrm{w}$ by $\boldsymbol{x}_\mathrm{r}$ in the population and move to steps 8a or 8b, according to the outcome of its comparison with the current best vertex, i.e., $f(\boldsymbol{x}_\mathrm{r}) < f(\boldsymbol{x}_1)$. Otherwise, we move to step 9, to decide whether $\boldsymbol{x}_\mathrm{r}$ should be accepted or withdrawn, thus seeking another candidate.

Step 8a: If $f(\boldsymbol{x}_\mathrm{r}) < f(\boldsymbol{x}_1)$, the vector $\boldsymbol{x}_\mathrm{r} - \boldsymbol{x}_1$ defines a direction of minimization. We remark that the detection of downhill slopes in high-dimensional spaces of complex geometry is not an often case. This makes essential to take advantage in order to accelerate the search procedure, by employing a sequence of N_e trial expansion steps through the recursive formula:

$$\boldsymbol{x}_{ce}^{k} = \boldsymbol{g} + \delta_k\,(\boldsymbol{x}_\mathrm{r} - \boldsymbol{g}) \tag{8}$$

where δ_k is a scale coefficient given by $\delta_k = \delta_{k-1} + (k - 1)/(N_\mathrm{e} - 1)$, for $k = 1, \ldots, N_\mathrm{e}$. The expansion continues as long as the AF value is improved (or until reaching the bounds of the feasible space). The optimal (in terms of AF) trial point, $\boldsymbol{x}_\mathrm{e}$, is kept in the external archive and replaces $\boldsymbol{x}_\mathrm{r}$ in the current population, provided that $f(\boldsymbol{x}_\mathrm{e}) < f(\boldsymbol{x}_\mathrm{r})$. In that case, the algorithm moves to step 12 to finalize the cycle.

Step 8b: If $f(\boldsymbol{x}_r) > f(\boldsymbol{x}_1)$, we attempt detecting a promising solution in the neighborhood of $\boldsymbol{x}_1$, by employing N_c trial contractions of the simplex in the interval between the centroid and the reflection point, according to the formula:

$$\boldsymbol{x}_{cc}^{k} = \boldsymbol{g} + (0.25 + 0.5\delta_k)\,(\boldsymbol{x}_r - \boldsymbol{g}) \tag{9}$$

where $\delta_k = (k - 1)/(N_c - 1)$, for $k = 1, \ldots, N_c$. The optimal (in terms of AF) trial point, $\boldsymbol{x}_c$, is kept in the external archive and replaces $\boldsymbol{x}_r$ in the current population, provided that $f(\boldsymbol{x}_c) < f(\boldsymbol{x}_r)$. In that case, the algorithm moves to step 12 to finalize the generation cycle.

Step 9: If $f(\boldsymbol{x}_r) > f(\boldsymbol{x}_w)$, we use the modified objective function (7) to decide whether employing inside contraction of the simplex, thus seeking for a potential local optimum, or expanding towards a non-optimal (i.e., uphill) direction, in an attempt to escape from the current area of attraction. In this respect, if $g(\boldsymbol{x}_r) > g(\boldsymbol{x}_w)$ we move to step 10a, otherwise we move to step 10b.

Step 10a: We reject $\boldsymbol{x}_r$ and implement N_c trial inside contractions of the simplex in the interval between the centroid and the worst point, according to the formula:

$$\boldsymbol{x}_{cc}^{k} = \boldsymbol{g} - (0.25 + 0.5\delta_k)\,(\boldsymbol{g} - \boldsymbol{x}_r) \tag{10}$$

where $\delta_k = (k - 1)/(N_c - 1)$, for $k = 1, \ldots, N_c$. The optimal (in terms of AF) trial point, $\boldsymbol{x}_c$, is kept in the external archive and replaces $\boldsymbol{x}_w$ in the current population, provided that $f(\boldsymbol{x}_c) < f(\boldsymbol{x}_w)$. Otherwise, the simplex shrinks towards the best vertex $\boldsymbol{x}_1$, such as:

$$\boldsymbol{x}_{s,i} = 0.5(\boldsymbol{x}_1 + \boldsymbol{x}_i) \text{ for } i = 2, \ldots, n + 1 \tag{11}$$

We remark that the above transformation is the sole evolving mechanism of the algorithm allowing the simultaneous generation of multiple points; particularly, n new points are generated that replace all previous vertices in the current population. This can be considered as milestone of the search procedure, in the sense that a local minimum, lying in the neighborhood of $\boldsymbol{x}_1$, has been surrounded. This is the time to reduce the temperature of the optimization system by a reduction factor ψ. In contrast to EAS, where ψ is a constant parameter of the annealing cooling schedule, usually taking values into the interval 0.90–0.99, in its surrogate-enhanced version ψ is automatically adjusted to also account for the progress index PI, using the following expression:

$$\psi = \max(1 - \text{PI}, 0.50) \quad (12)$$

The threshold of 0.50 prohibits a fast reduction of temperature and therefore maintains enough randomness within decisions, which in turn prohibits early convergence to local optima. After reducing T, the iteration cycle is finalized (step 12).

Step 10b: The reflection point $\boldsymbol{x}_{\mathrm{r}}$ is accepted although being worse than $\boldsymbol{x}_{\mathrm{w}}$. Next, N_{u} uphill (i.e., maximization) movements are performed using the same formula with multiple expansion (eq. 9), in an attempt to pass the hill and discover adjacent regions of attraction. This geometrical transformation was introduced by Pan and Wu (1998), to facilitate the simplex escaping from local minima. Similarly to previous steps, we use the AF to determine the optimum uphill point, $\boldsymbol{x}_{\mathrm{u}}$. If $f(\boldsymbol{x}_{\mathrm{u}}) < f(\boldsymbol{x}_{\mathrm{r}})$, this point is kept in the external archive and replaces $\boldsymbol{x}_{\mathrm{r}}$ in the current population, while the algorithm moves to step 12 to finalize the generation cycle. Otherwise, none of the simplex transformations results to a better solution than the worst vertex $\boldsymbol{x}_{\mathrm{w}}$, thus the last option is to attempt a pure stochastic generator, referred to as mutation (step 11).

***Step 11*:** We seek a random point out of the typical range of the current population, defined on the basis of the mean, $\boldsymbol{\mu}_P$, and standard deviation, $\boldsymbol{\sigma}_P$, of all members of P. In this respect, we generate a normally-distributed point $\boldsymbol{x}_{\mathrm{m}}$ out of the interval $[\boldsymbol{\mu}_P - \boldsymbol{\sigma}_P, \boldsymbol{\mu}_P + \boldsymbol{\sigma}_P]$, which is accepted if $f(\boldsymbol{x}_{\mathrm{m}}) < f(\boldsymbol{x}_{\mathrm{r}})$. Otherwise, we account for a user-specified mutation probability p_{m} in order to accept or not the randomly generated point, $\boldsymbol{x}_{\mathrm{m}}$, and replacing $\boldsymbol{x}_{\mathrm{r}}$ in the current population. Anyway, since $\boldsymbol{x}_{\mathrm{m}}$ is evaluated through the objective function, it enters the external archive.

Step 12: Considering the new member (or members, in the particular case of simplex shrinkage) of the population, we re-evaluate the current minimum, $\boldsymbol{x}_{\min}$, and maximum, $\boldsymbol{x}_{\max}$, and their function values, $f_{\min}$ and $f_{\max}$. We also re-evaluate the current number of function evaluations, FE, and check whether this hasn't exceeded the termination criterion, MFE. Finally, we re-evaluate the temperature so that $T \leq \xi(f_{\max} - f_{\min})$, where $\xi \geq 1$ is a user-specified parameter of the annealing schedule, usually set between 2 to 5. This restriction prevents T taking extremely high values, which would deteriorate the efficiency of SEEAS, as far as search would become too random.

To run the algorithm, it is essential providing values for all input arguments, which are the number of desirables steps within different simplex transitions (N_r, N_e, N_c, N_u), the mutation probability p_m, and the adjusting factor ξ of the annealing cooling schedule. Recommended values, also used in all next benchmarking tests, are $N_r = N_e = N_c = N_u = 20$, $p_m = 0.10$ and $\xi = 2$. These values were determined on the basis of extended investigations within the development of SEEAS, and they have been also validated through the sensitivity analysis of section 4.4.

3 Benchmarking methodology

3.1 Benchmarking protocol

To assess the performance of SEEAS we compared it with the original version of EAS as well as three state-of-the-art optimization algorithms, which are synoptically presented in section 3.3. Two of the benchmark algorithms, i.e., DYCORS and MLMSRBF, are surrogate-assisted, while EAS and DDS do not employ surrogate models through search.

A variety of test problems were examined, theoretical as well as real-world. Briefly, the hereafter called benchmarking "suite" includes six mathematical test functions, formulated with 15 and 30 control variables, a hydrological calibration problem with real and synthetic data, and a time-expensive multi-reservoir management problem (6×2 + 1×2 + 1 = 15 problems, in total).

To ensure fair comparison and safely infer about the performance of the algorithms we attempted to ensure as much as similar configurations, as summarized in Table 1. In all problems we employed multiple independent runs, using the same population size and the same random generation technique, i.e., LHS. The population size was set equal to $m = 2(n + 1)$, as recommended by Regis and Shoemaker (2007a) and Regis and Shoemaker (2013), where n is the problem dimension (i.e., the number of control variables). We remark that other researchers relate the initial population size (also referred to as design of experiment, DoE) to the available computational budget, quantified in terms of MFE, in order to design a more detailed metamodel; for instance, Razavi et al. (2012b) suggest that $m = \max[2(n + 1), 0.1 \text{ MFE}]$. However, in our tests we avoided associating m with MFE, in order to investigate the impacts of the problem dimension to the performance of the examined algorithms. Furthermore, we preferred saving resources for the evolutionary procedure, instead of spending a non-negligible part of our budget to the initial DoE.

Each problem but the last was solved considering two alternative computational budgets, MFE (500 and 1000). We run all tests with two different budgets (instead of the maximum of them) since all examined algorithms (except EAS) involve parameters depending on MFE (in particular, SEEAS uses the progress index PI, defined in eq. (4), within the annealing cooling schedule). Finally, for the three surrogate-based methods (SEEAS, DYCORS, MLMSRBF) we employed the same metamodel (RBF with cubic basis functions and linear polynomial tail), thus ensuring similar computational effort for building, updating and exploiting the RBF (Razavi et al., 2012a). We remark that in real-world problems the effort of the optimization routines (including metamodel fitting) is much less than the effort of simulation, and therefore the runtime of the overall search procedure is practically relative to MFE.

All computations were implemented in MATLAB mathematical environment using a 3.0 GHz Intel Core i5 processor with 4 GB of RAM, running on Windows 8 OS. For the SEEAS method we employed the typical input arguments given in section 2.2.4, while for the other algorithms, i.e., EAS, DDS, MLMSRBF and DYCORS, we used the default values suggested in the associated articles (Efstratiadis and Koutsoyiannis, 2002; Regis and Shoemaker, 2007b; Tolson and Shoemaker, 2007; Regis and Shoemaker, 2013).

Table 1: Configuration of benchmarking suite.

Problem	Algorithms	Number of control variables, n	Max. function evaluations (MFE)	Independent Runs with random initial populations	Population size	Surrogate model (metamodel)
Test functions	All	15	500, 1000	30	32	RBF with cubic basis functions and linear polynomial tail
Test functions	All	30	500, 1000	30	62	
Model calibration with real data	All	11	500, 1000	30	24	
Toy calibration with synthetic data	All	11	500, 1000	30	24	
Multireservoir management problem	SEEAS, DYCORS, MLMSRBF	20	500	10	42	

3.2 Performance evaluation approach

Following the ideas of Razavi et al. (2012a) and Matott et al. (2012), after implementing all runs for each specific optimization problem solved with a specific algorithm, we plotted the cumulative distribution function (CDF) of the optimal values of $f(\boldsymbol{x})$ obtained within the specific budget. In order to quantify the probability of attaining an equal or better solution, we used the concept of stochastic dominance (SD), introduced by Levy (1992), to compare the CDFs of the algorithms. Let Φ_A and Φ_B be the CDFs of algorithms A and B, respectively. Assuming the minimization of a random quantity q, we assume that A dominates B if $\Phi_A(q) > \Phi_B(q)$ for all q, and vice versa. On the contrary, if the two CDFs are intersected at some point q_u, then SD is not applicable. In this case, we evaluated the median point, i.e., the one with 50% probability of exceedance, and considered as better the algorithm with the best performance at this point. In fact, to ensure that the difference of the two algorithms at the point of interest is statistically significant, we employed the non-parametric Mann–Whitney U-test (MWU; Mann and Whitney, 1947). The null hypothesis of the MWU test is that data in Φ_A and Φ_B are samples from continuous distributions with equal medians. The confidence level of the MWU test was set to 95%.

3.3 Brief description of benchmark optimization algorithms

3.3.1 Dynamically Dimensioned Search (DDS)

Dynamically Dimension Search[2] (DDS), is a stochastic, single-solution based algorithm, developed by Tolson and Shoemaker (2007) to locate near-optimal solutions with few function evaluations. DSS is designed to search globally at the early stages and more locally when approaching a user-specified number of maximum function evaluations (MFE). It evolves by perturbing the current best solution in randomly selected dimensions, using an evolutionary operator based on the normal distribution. The probability of selecting a dimension to perturb is proportional to the current number of function evaluations and MFE. The transition from global to local search is employed by dynamically reducing the number of perturbed dimensions. In the literature are reported several successful applications of DDS (e.g., Tolson et al., 2009; Razavi et al., 2010; Matott et al., 2012; Razavi et al., 2012a; Regis and Shoemaker, 2013).

3.3.2 Multistart Local Metric Stochastic RBF algorithm (MLMSRBF)

Regis and Shoemaker (2007b) developed the Multistart Local Metric Stochastic RBF[3] (MLMSRBF), which is surrogate-assisted optimization algorithm that can be considered as extension of DDS. The first step is the implementation of the initial DoE to fit the surrogate model (particularly, RBF), which evolves by perturbing the current best point (similar to DDS), using normal distribution with zero mean and a specified covariance matrix. Additionally, in order to locate promising candidates, the algorithm uses a metric that balances the RBF prediction and the minimum distance from previously evaluated points (this is similar to the acquisition function introduced in 2.2.3, but with constant weights). The global character of the algorithm is further enhanced by implementing multiple DoEs. This multistart strategy is enabled only if the algorithm appears to have been trapped to a local minimum. Regis and Shoemaker (2007b) demonstrated the

[2] https://github.com/akamel001/Dynamic-Dimension-Search
[3] https://courses.cit.cornell.edu/jmueller/ or http://people.sju.edu/~rregis/pages/software.html

efficiency of MLMSRBF in several benchmark problems, including 17 multimodal test functions and a 12-dimensional groundwater bioremediation problem. In the literature are also reported other successful applications of the method (e.g., Mugunthan et al., 2005; Mugunthan and Shoemaker, 2006; Regis and Shoemaker, 2013).

3.3.3 DYnamic COordinate Search-Multistart Local Metric Stochastic RBF (DYCORS-LMSRBF)

The DYCORS framework was recently proposed by Regis and Shoemaker (2013) for surrogate-based optimization of high-dimensional expensive functions. The authors presented two versions, DYCORS-LMSRBF and DYCORS-DDSRBF[4]. The former is extension of LMSRBF and the latter is a surrogate-assisted DDS (here we use DYCORS-LMSRBF that performed slightly better than DYCORS-DDSRBF). DYCORS employs a strategy similar to DDS by dynamically and probabilistically reducing the number of perturbed dimensions until reaching the MFE. In order to generate trial candidate points (on the selected/perturbed dimensions) the algorithm uses a normal distribution with zero mean and standard deviation σ_n, but this does not remain constant, since σ_n is dynamically adjusted to control the range of perturbation. Moreover, DYCORS-LMSRBF is cycling through a set of weights in order to balance exploration and exploitation of the surrogate model. The authors assessed the performance of the two algorithms against several optimization schemes in a variety of test problems, among which a 14-D hydrological calibration problem.

4 Test functions

4.1 Setup of optimization problems

The first suite of benchmark problems involves the optimization of six well-known mathematical problems (test functions), combining two alternative formulations in terms of number of variables (n = 15 and 30), and two algorithmic configurations in terms of MFE (500 and

[4] https://courses.cit.cornell.edu/jmueller/

1000). This setting allowed for assessing the performance of the algorithms against increasing levels of dimensionality and increasing computational budget. Considering two alternative dimensions and two computational budgets, we configured four different problems for each test function, i.e., 24 optimization problems in total. According to the benchmarking protocol explained in section 3.1, for all problems, we employed 30 independent runs, thus randomly changing the initial population of each search experiment. The population size of all algorithms we set equal to 32 and 62, for the 15-D and 30-D formulations, respectively.

Table 2 summarizes the main characteristics of the examined test functions, which represent search spaces of different complexity. Two of them (Sphere and Zakharov) are unimodal, while the rest are multimodal (Ackley, Griewank, Rastrigin, Levy). In all cases the global minimum is known and equal to zero. The analytical expression of the test functions and the bounds of their variables are given in the Appendix.

Table 2: Summary characteristics of test functions (their references are given in Appendix).

Problem	Test function	Response surface properties
OF1	Sphere	Unimodal and convex
OF2	Ackley	Multimodal with many local minima
OF3	Griewank	Multimodal with many regularly distributed local minima
OF4	Zakharov	Unimodal with a plate-shaped valley
OF5	Rastrigin	Multimodal with many local minima
OF6	Levy	Multimodal with many local minima and parabolic valleys

4.2 Statistical evaluation of optimal solutions

An initial assessment of the performance of the five examined algorithms was made on the grounds of mean and standard deviation of the best function values obtained from each optimization set (i.e., 30 independent runs of the algorithm). The closest to zero is the mean and the lowest the standard deviation indicates that the algorithm reaches the theoretical optimum with high accuracy and reliability.

The statistical superiority of SEEAS is exhibited in all problem configurations, as shown in Table 3 and Table 4, for problem dimensions n = 15 and 30, respectively. Specifically, for the 15-D formulation (Table 3), SEEAS achieves the best performance (i.e., the lowest mean) in three out of six (OF1, OF3, OF6) and four out of six problems (OF1, OF2, OF3, OF6), for MFE = 500 and 1000, respectively. By doubling the dimensionality of the test functions to $n = 30$, thus significantly increasing the complexity of the associated optimization problems, SEEAS outperforms the other algorithms in four out of six (OF1, OF2, OF3, OF6) and three out of six problems (OF1, OF3, OF6), for MFE = 500 and 1000, respectively (Table 4). Considering all alternative configurations, SEEAS is optimal for 14 out of 24 problems, DYCORS and EAS are optimal for 4 out of 24, and DDS is optimal for 3 out of 24. MLMSRBF does not outperform in none of the 24 test problems.

As expected, the increase of computational budget from 500 to 1000 improves the performance of all algorithms. In general, the most significant improvement is achieved by EAS and DDS, which is reasonable since these algorithms are not surrogate-assisted, thus they are by definition designed to proceed slower than the other schemes. The convergence behavior of the algorithms is further investigated in next section.

It is also worth mentioning that all algorithms exhibit poor performance against functions OF4 (Zakharov) and OF5 (Rastrigin), since they fail locating satisfactory solutions for the given budgets. In particular, the plate-shaped valley of Zakharov function makes extremely difficult fitting metamodels, which degenerates to hyperplane with practically zero slopes. It is not surprising that EAS ensures the best solutions, although these are still far from the theoretical optimum. EAS has been designed to also handle flat response surfaces, which are often met in water management optimization problems, as further explained in section 6.3. On the other hand, DDS is the algorithm that generally ensures the best solution of the Rastrigin problem. Again, this is not surprising, since the search space of this function is extremely rough, with multiple local minima, thus the most stochastic of all schemes is expected to be the most efficient.

Table 3: Mean and standard deviation of best solutions in 15-D test problems (optimal results are highlighted).

MFE	Test function	EAS		DDS		SEEAS		DYCORS		MLMSRBF	
		Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
500	OF1	1.938	0.978	0.852	0.479	**0.002**	0.001	**0.002**	0.001	0.019	0.014
	OF2	7.159	1.723	6.025	1.314	0.812	0.233	**0.809**	0.372	2.231	0.658
	OF3	7.682	2.997	2.626	1.269	**0.538**	0.118	0.885	0.084	1.085	0.052
	OF4	**39.434**	14.894	137.447	52.366	59.144	28.023	158.669	47.788	150.411	49.875
	OF5	86.245	14.148	**24.887**	7.081	46.268	15.359	38.958	12.340	45.920	18.803
	OF6	1.905	0.877	0.681	0.314	**0.203**	0.105	1.208	1.406	1.344	2.129
1000	OF1	0.378	0.177	0.150	0.079	**0.001**	0.001	0.001	0.000	0.011	0.007
	OF2	3.523	0.936	3.847	0.528	**0.437**	0.208	0.607	0.092	1.862	0.556
	OF3	2.444	1.061	1.505	0.299	**0.368**	0.140	0.809	0.082	1.040	0.037
	OF4	**26.828**	17.895	97.541	38.226	41.290	26.639	121.266	36.925	121.359	37.730
	OF5	59.735	17.012	**11.233**	3.136	29.733	12.838	33.585	13.490	35.784	11.031
	OF6	0.767	0.292	0.234	0.104	**0.124**	0.060	0.536	0.860	0.524	0.863

Table 4: Mean and standard deviation of best solutions in 30-D test problems (optimal results are highlighted).

MFE	Test function	EAS		DDS		SEEAS		DYCORS		MLMSRBF	
		Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev	Mean	StDev
500	OF1	4.305	1.163	9.516	2.737	**0.019**	0.006	0.083	0.034	0.739	0.708
	OF2	9.923	1.160	12.872	1.329	**1.878**	0.301	4.297	3.721	6.193	4.362
	OF3	17.866	3.455	38.398	12.050	**0.782**	0.118	1.265	0.079	3.459	1.927
	OF4	**117.821**	28.757	562.145	113.230	173.240	44.185	472.815	90.897	575.424	174.073
	OF5	228.693	18.442	132.149	24.567	122.658	19.427	**112.046**	23.076	165.437	46.846
	OF6	6.338	2.652	15.823	5.481	**0.659**	0.184	3.407	2.540	7.326	10.944
1000	OF1	2.529	0.933	2.112	0.791	**0.006**	0.004	0.011	0.004	0.358	0.177
	OF2	6.516	0.845	7.670	0.924	1.206	0.297	**1.085**	0.168	3.643	1.103
	OF3	8.836	2.617	8.273	2.679	**0.549**	0.093	1.020	0.026	2.420	0.713
	OF4	**94.598**	20.317	412.238	118.573	151.472	54.097	403.812	93.081	491.425	146.097
	OF5	198.335	16.587	**71.598**	15.028	98.371	19.505	85.267	22.956	134.864	39.193
	OF6	2.683	0.736	3.921	2.215	**0.443**	0.126	4.213	5.440	2.865	4.583

4.3 Evaluation of convergence behavior

In order to further investigate the convergence behavior of the algorithms, we plotted the average (out of 30 trials) value of the best point found so far against the number of function evaluations (Figures 2-7). Each figure refers to a specific test function and comprises four charts, for the alternative configurations (two dimensions × two MFE).

In most cases, SEEAS exhibits the faster convergence, evidently because the expansion mechanisms supported by the metamodel (which provides enhanced overview of the surface geometry), allow implementing steep downhill transitions. In general, the great advantage of the simplex-based transitions is the indirect use of the concept of gradient, which favors quick location of regions of attraction of local optima. This is of particular importance in computational expensive problems, where the algorithm should quickly detect promising descent directions. In fact, SEEAS is clearly superior to the other two surrogate-assisted algorithms (DYCORS and MLMSRBF) in all problems, except for Rastrigin. The most impressive case is the Levy problem, where SEEAS locates a very good solution after the first one hundred of function evaluations (Figure 9a), while the mean best value found by other algorithms so far is even two orders of magnitude higher. Similar are the results for the Griewank function (Figure 9b), which could be interpreted as a rough, multimodal version of sphere. A plausible explanation for this is the combined effect of the knowledge gained by the metamodel, which easily recognizes the spherical structure of Griewank, and the simplex-based operators, using approximations of the gradient of the function.

An interesting conclusion is that, regarding SEEAS, the increase of the computation budget has mild effects in the improvement of the mean best solution. This is another evidence of the suitability of SEEAS for extremely time-demanding optimization problems, in which the desirable number of function evaluations should be minimal.

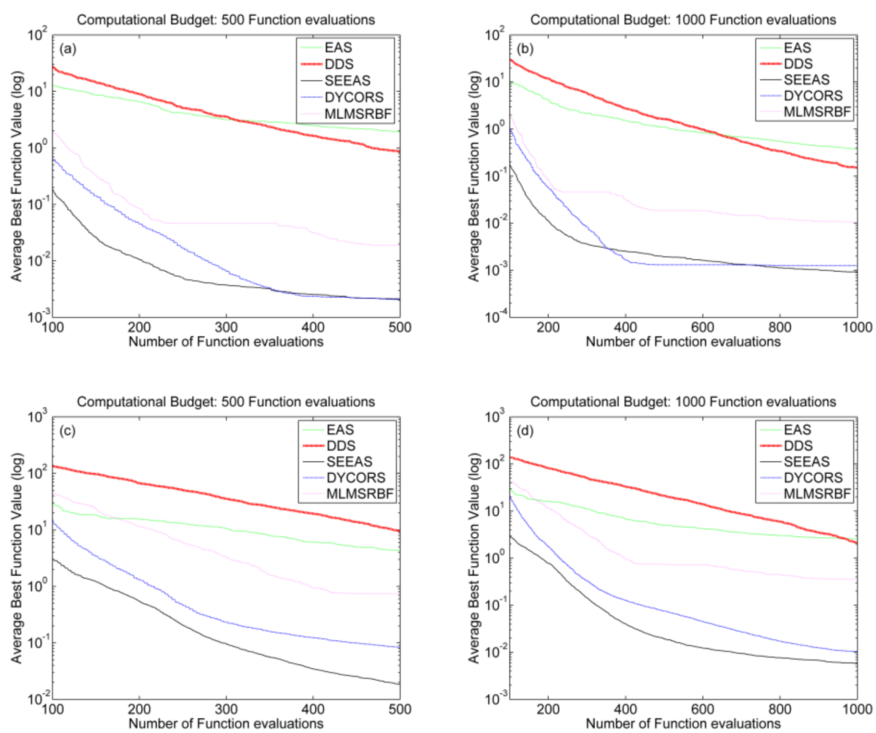


Figure 3: Convergence curves for test function OF1 (Sphere) with 15 (a, b) and 30 variables (c, d), with MFE=500 (a, c) and MFE=1000 (b, d).

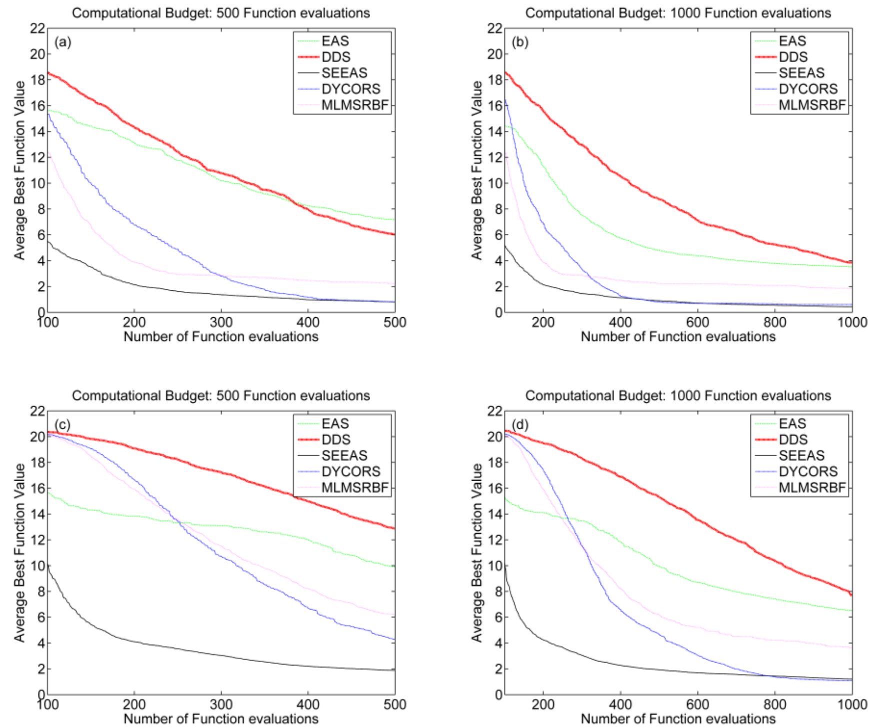


Figure 4: Convergence curves for test function OF2 (Ackley) with 15 (a, b) and 30 variables (c, d), with MFE=500 (a, c) and MFE=1000 (b, d).

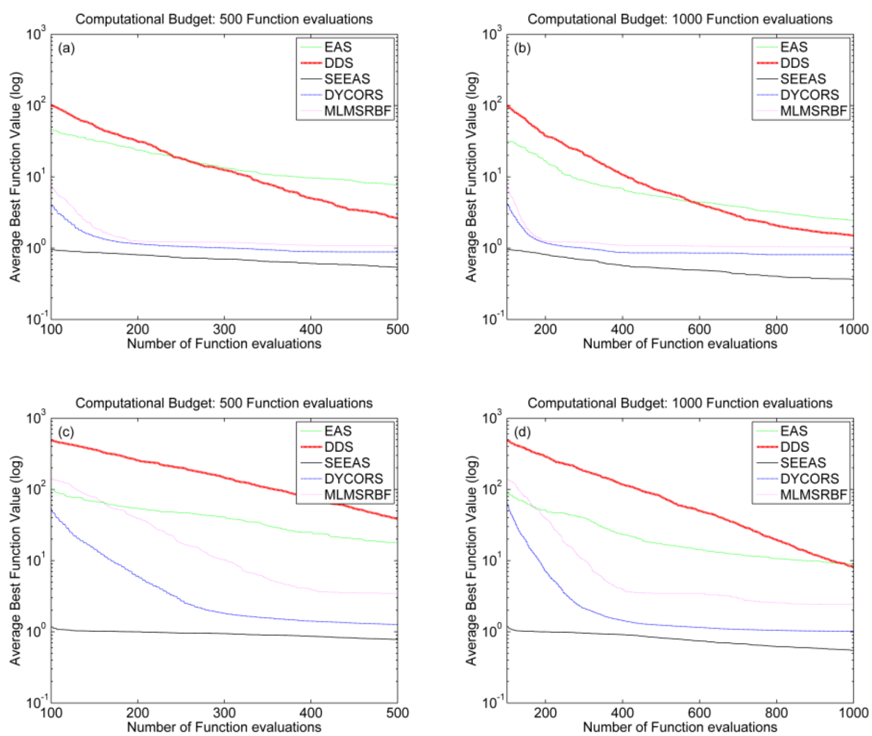


Figure 5: Convergence curves for test function OF3 (Griewank) with 15 (a, b) and 30 variables (c, d), with MFE=500 (a, c) and MFE=1000 (b, d).

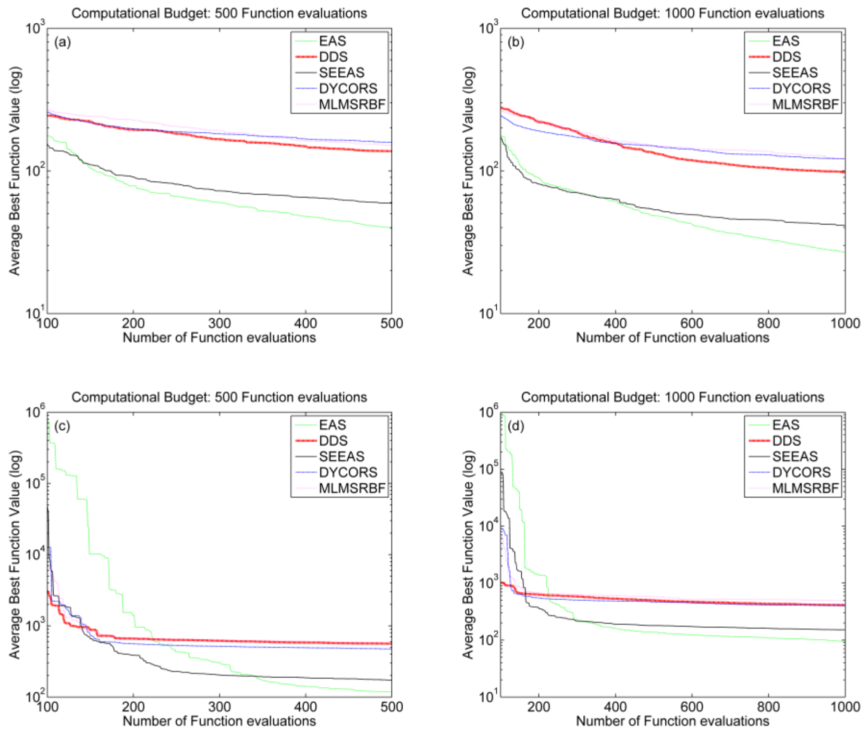


Figure 6: Convergence curves for test function OF4 (Zakharov) with 15 (a, b) and 30 variables (c, d), with MFE=500 (a, c) and MFE=1000 (b, d).

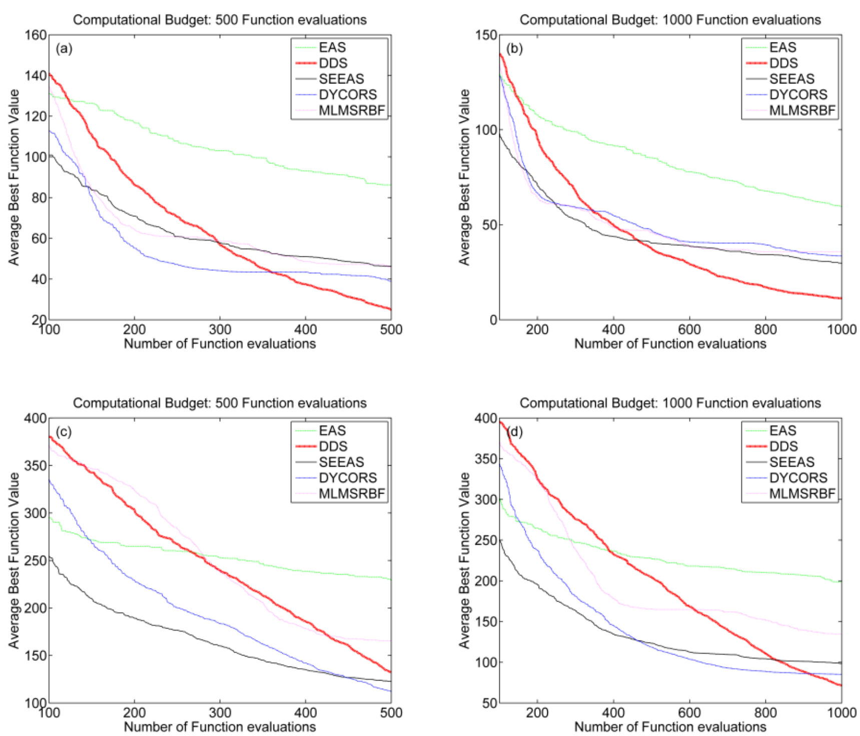


Figure 7: Convergence curves for test function OF5 (Rastrigin) with 15 (a, b) and 30 variables (c, d), with MFE=500 (a, c) and MFE=1000 (b, d).

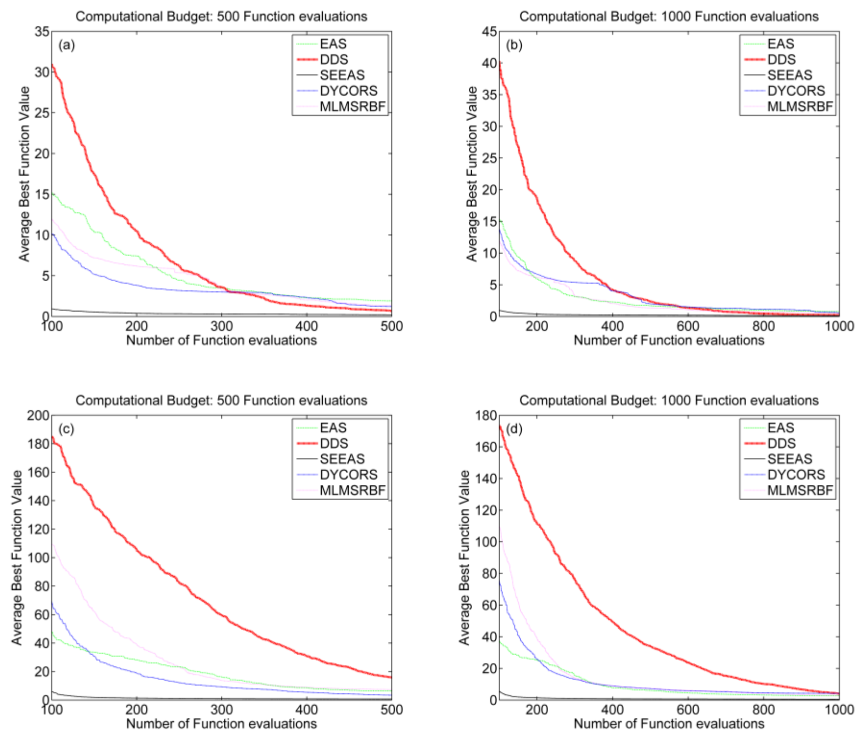


Figure 8: Convergence curves for test function OF6 (Levy) with 15 (a, b) and 30 variables (c, d), with MFE=500 (a, c) and MFE=1000 (b, d).

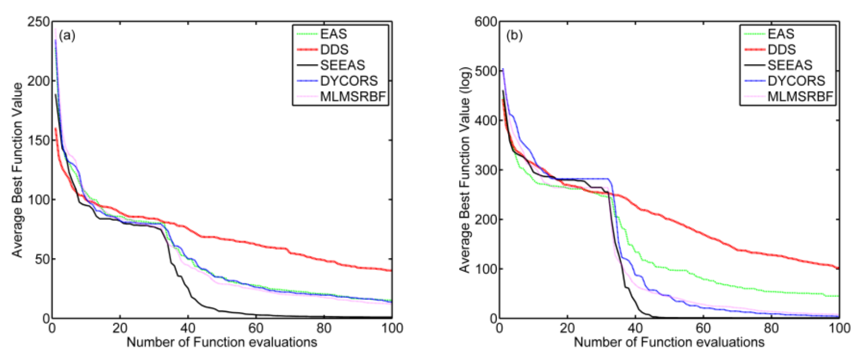


Figure 9: Initial part of convergence curves up to 100 function evaluations, for test functions Levy (a) and Griewank (b), for the case of MFE=500 and 15 variables.

4.4 Sensitivity analysis against input parameters of SEEAS

As mentioned in section 2.2.4, SEEAS requires determining several input arguments, in terms of step parameters N_r, N_e, N_c and N_u, mutation probability p_m, and adjusting factor ξ of the annealing cooling schedule. In order to investigate the sensitivity of SEEAS against the default values adopted so far (i.e., $N_r = N_e = N_c = N_u = 20$, $p_m = 0.10$ and $\xi = 2$), we employed 30 independent runs of the tests functions (for n = 15 variables and MFE = 500), assigning different values to its input parameters. The configurations and summary statistics, in terms of means and standard deviation of the optimal solution of each set of optimizations, are given in Table 5 to Table 7

The analysis justifies our recommendations for the input parameters of SEEAS. As shown in Table 5, the performance of the algorithm is significantly improved by increasing the common value of the step parameters from 5 to 20, while it is slightly improved by further increasing this value to 50. Actually, the simplex transitions are considerably assisted by using the outcomes of the surrogate model within local search; however, it does not make sense calling the SM too many times, which introduces unnecessary computations with marginally only benefit. Regarding the mutation probability (Table 6), the algorithm provides almost identical results for p_m values as low as 0.05 or 0.10, yet its performance is clearly deteriorated by increasing this probability up to 0.30. This is also a non-surprising conclusion, since it is well-known that in evolutionary algorithms the mutation operator should be occasionally called in order to avoid making search too random. Finally, the setup with $\xi = 2$ provides systematically better results compared to $\xi = 1$, while it exhibits either better or similar performance when the annealing cooling parameter increases up to $\xi = 4$ (Table 7). Nevertheless, a common outcome form the above investigations is the relatively low sensitivity of SEEAS against the examined configurations, for most of test problems.

Table 5: Mean and standard deviation of best solutions in 15-D test problems for MFE = 500, for different values of the four step parameters of SEEAS (for p_m = 0.10 and ξ = 2).

Test function	$N_r = N_e = N_c = N_u = 5$		$N_r = N_e = N_c = N_u = 20$		$N_r = N_e = N_c = N_u = 50$	
	Mean	StDev	Mean	StDev	Mean	StDev
OF1	0.002	0.001	0.002	0.001	0.001	0.001
OF2	1.724	0.641	0.812	0.233	0.806	0.258
OF3	0.714	0.15	0.538	0.118	0.504	0.133
OF4	87.043	32.027	59.144	28.023	58.030	28.911
OF5	51.299	21.579	46.268	15.359	45.101	13.634
OF6	0.266	0.228	0.203	0.105	0.192	0.114

Table 6: Mean and standard deviation of best solutions in 15-D test problems for MFE = 500, for different values of mutation probability p_m (for $N_r = N_e = N_c = N_u = 20$ and $\xi = 2$).

Test function	$p_m = 0.05$		$p_m = 0.10$		$p_m = 0.30$	
	Mean	StDev	Mean	StDev	Mean	StDev
OF1	0.002	0.001	0.002	0.001	0.002	0.001
OF2	0.895	0.345	0.812	0.233	0.994	0.480
OF3	0.534	0.125	0.538	0.118	0.654	0.149
OF4	61.071	21.639	59.144	28.023	61.876	29.762
OF5	46.176	15.088	46.268	15.359	49.930	15.636
OF6	0.226	0.088	0.203	0.105	0.277	0.095

Table 7: Mean and standard deviation of best solutions in 15-D test problems for MFE = 500, for different values of mutation probability p_m and cooling parameter ξ (for $N_r = N_e = N_c = N_u = 20$ and $p_m = 0.10$).

Test function	$\xi = 1$		$\xi = 2$		$\xi = 4$	
	Mean	StDev	Mean	StDev	Mean	StDev
OF1	0.003	0.002	0.002	0.001	0.002	0.002
OF2	0.978	0.349	0.812	0.233	0.896	0.381
OF3	0.759	0.141	0.538	0.118	0.894	0.172
OF4	69.603	29.140	59.144	28.023	62.896	35.297
OF5	66.730	15.871	46.268	15.359	45.403	17.132
OF6	0.423	0.113	0.203	0.105	0.206	0.151

4.5 Suitability assessment based on stochastic dominance

For each problem and each algorithm we illustrate the empirical CDFs, using the sample of 30 best solutions found after the termination of the corresponding search procedures. Based on them, we calculated the medians of the CDFs (Table 8), and next employed the MWU test between the algorithms providing the best (lower) medians, to assess whether the obtained differences are significant. The results of all tests are summarized in Table 9.

The outcomes of the MWU test are in full accordance with previous conclusions, and clearly prove the statistical suitability of SEEAS. Considering the full set of problems, SEEAS is evaluated as “preferred” or “equally good” in 18 out of 24 cases. Next best method is DYCORS, which is preferred or equally good in 6 out of 24 cases. If we isolate the less beneficial subset, i.e., the formulation with 30 decision variables under the lower computational budget (lower left panel of Table 9), the superiority of SEEAS is even more evident.

Table 8: Median of best function values obtained from all algorithms.

n	Problem	MFE = 500					MFE = 1000				
		EAS	DDS	SEEAS	DYCORS	MLMSRBF	EAS	DDS	SEEAS	DYCORS	MLMSRBF
15	OF1	1.457	0.684	**0.002**	0.002	0.012	0.380	0.131	**0.001**	0.001	0.008
	OF2	7.367	5.942	0.838	**0.745**	2.353	3.519	3.877	0.410	0.574	1.629
	OF3	7.446	2.312	**0.513**	0.921	1.088	2.211	1.400	**0.360**	0.819	1.027
	OF4	**34.205**	133.574	53.874	154.151	147.998	**25.224**	98.089	34.413	127.557	110.313
	OF5	85.223	**24.714**	45.061	37.912	37.696	58.926	**10.813**	31.808	32.644	34.522
	OF6	1.592	0.616	**0.198**	0.681	0.488	0.765	0.216	0.114	**0.069**	0.191
30	OF1	4.391	9.828	**0.018**	0.073	0.590	2.516	1.860	**0.005**	0.009	0.270
	OF2	9.844	13.110	**1.918**	3.144	4.725	6.579	7.831	1.170	**1.108**	3.438
	OF3	17.758	36.453	**0.807**	1.249	2.974	8.741	7.920	**0.554**	1.025	2.507
	OF4	**114.878**	540.070	168.695	456.956	570.266	**95.274**	386.140	147.120	409.986	465.057
	OF5	232.766	130.090	121.973	**112.009**	156.834	200.952	71.160	97.994	**85.728**	127.299
	OF6	5.264	15.496	**0.630**	2.075	2.302	2.458	2.918	**0.431**	2.762	1.412

Table 9: Summary results of MWU test to infer about the preferred algorithm. *H*-value indicates the rejection or not of the null hypothesis, i.e., if *H* = 0, the null hypothesis is not rejected.

n	Problem	MFE = 500				MFE = 1000			
		Preferred	Alternative	p-Value	*H*	Preferred	Alternative	p-Value	*H*
15	OF1	**SEEAS**	**DYCORS**	5.298E-01	0	**SEEAS**	MLMSRBF	3.020E-11	1
	OF2	**DYCORS**	**SEEAS**	3.478E-01	0	**SEEAS**	DYCORS	1.492E-06	1
	OF3	**SEEAS**	DYCORS	9.919E-11	1	**SEEAS**	DYCORS	1.206E-10	1
	OF4	**EAS**	SEEAS	3.034E-03	1	**EAS**	SEEAS	9.883E-03	1
	OF5	**DDS**	DYCORS	9.514E-06	1	**SEEAS**	DYCORS	2.028E-07	1
	OF6	**SEEAS**	MLMSRBF	3.644E-02	1	**DYCORS**	**SEEAS**	6.952E-01	0
30	OF1	**SEEAS**	DYCORS	3.020E-11	1	**SEEAS**	DYCORS	2.154E-06	1
	OF2	**SEEAS**	DYCORS	3.474E-10	1	**DYCORS**	**SEEAS**	9.926E-02	0
	OF3	**SEEAS**	DYCORS	3.020E-11	1	**SEEAS**	DYCORS	3.020E-11	1
	OF4	**EAS**	SEEAS	6.526E-07	1	**EAS**	SEEAS	3.157E-05	1
	OF5	**DYCORS**	**SEEAS**	5.746E-02	0	**DYCORS**	DDS	1.988E-02	1
	OF6	**SEEAS**	DYCORS	1.094E-10	1	**SEEAS**	MLMSRBF	2.572E-07	1

5 Hydrological calibration

5.1 Study area, simulation model and calibration setup

Hydrological calibration is probably the most typical global optimization problem in water resources. Numerous studies have been published dealing with calibration and its shortcomings, arising from the multiple sources of uncertainty that govern all aspects of the parameter estimation procedure (Efstratiadis and Koutsoyiannis, 2010). Here we investigated the calibration of a lumped simulation model, applied to Boeoticos Kephisos river basin, in Eastern Greece (1850 km^2). The basin extends over a heavily-modified karst system with multiple peculiarities, as result of complex interactions between surface and groundwater processes as well as human interventions, by means of surface and groundwater abstractions. This hydrosystem has been subject of comprehensive research, through alternative approaches(Rozos et al., 2004; Efstratiadis et al., 2008; Nalbantis et al., 2011). Monthly precipitation, potential evapotranspiration, runoff and groundwater abstraction data are available for a 77-year period (Oct. 1907 to Sep. 1984), to be used as inputs in simulations.

For the representation of the basin processes we applied a lumped version of Hydrogeios model (Efstratiadis et al., 2008). The basin is vertically subdivided into three storage elements that represent interception, soil moisture and groundwater. The model estimates the main responses of the basin, i.e., actual evapotranspiration, surface and groundwater runoff and groundwater losses, using nine parameters and two initial conditions, i.e., the water levels of soil and groundwater tanks at the beginning of simulation. A brief description of the model parameters and their feasible bounds assigned in optimizations is given in Table 10.

Based on the above data and tools, we formulated two optimization problems, using as objective function the well-known Nash-Sutcliffe efficiency metric (NSE). The first one follows the typical calibration paradigm (i.e., inverse modelling), in which the model parameters are unknown and the model is fitted to the observed runoff of the basin. In the second formulation, also referred to as "toy" calibration, we considered the (arbitrary) parameter set shown in Table 10, which ensures a relatively high NSE value. Next, we run the model forward to obtain synthetic runoff time series, for the given parameters and the same hydrological inputs, and finally we used these synthetic runoff data to infer the model parameters. The key difference of the second approach is that since the theoretical values of model parameters are *a priori* known, the theoretical optimum is by definition one. On the contrary, in real-data calibrations both the value and the location of the global optimum are unknown. A plausible (but not certain) approximation of the maximum NSE is 0.775, which was estimated by running EAS for multiple initial populations, allowing a reasonably large number of function evaluations (MFE = 5000). The key advantage of toy calibration is that the search procedure is not affected by structural and observation errors (i.e., both the model and the data are considered perfect), which allows fairly evaluating the performance of the optimization methods. In addition, since the value of the global optimum is by definition higher than is the case of real data, the optimization problem itself becomes harder to solve.

Similarly to test functions, we assessed the performance of SEEAS against the other four algorithms assuming 30 independent runs and the typical computational budgets of 500 and 1000 function evaluations (each evaluation involves the implementation of a full simulation, for given parameters). We underline that the current suite of problems is only regarded as a computational exercise, aiming to test the algorithms against challenging problems of real-world type. In an operational context, hydrological calibration is far from a blind optimization game, since it should also account for issues such as the model predictive capacity and the physical interpretation of the optimized parameters (Efstratiadis and Koutsoyiannis, 2010).

Table 10: Model parameters, feasible bounds and values assigned for toy calibrations.

Parameter	Description and units	Lower value	Upper value	Toy value
r	Interception capacity (mm)	0.010	100.0	13.0
c	Recession coefficient for direct runoff (-)	0.010	1.000	0.098
k	Soil capacity (mm)	5.0	600.0	506.7
l	Recession coefficient for interflow (-)	0.010	1.000	0.922
κ	Interflow threshold, as ratio of soil capacity (-)	0.010	1.000	0.945
m	Recession coefficient for percolation (-)	0.010	1.000	0.064
φ	Recession coefficient for baseflow (-)	0.010	1.000	0.031
y_B	Threshold for baseflow generation (mm)	5.0	300.0	35.9
ξ	Recession coefficient for underground losses (-)	0.010	1.000	0.068
s_0	Initial soil moisture storage (mm)	0.0	600.0	5.1
y_0	Initial groundwater storage (mm)	5.0	300.0	111.2

5.2 Model calibration with unknown parameters

Table 11 summarizes the statistical characteristics of the set of 30 optimal solutions found under the two budgets. SEEAS clearly outperforms all other algorithms in terms of mean and median values of NSE. In particular, the mean optimal efficiency is 0.632 and 0.727, for MFE = 500 and 1000, respectively, while the medians are even higher (0.714 and 0.747, respectively). In addition, the variability of NSE values is the lowest among all algorithms. For comparison, EAS reaches a median efficiency of only 0.448, for MFE = 500, but it is considerably increased up to 0.719, for MFE = 1000. In this last case, the mean NSE is only 0.572, due to the existence of some quite low values in the sample of 30 optimal solutions, which converge to a local optimum far from the global one. Finally, the statistical performance of the other three schemes (DDS, DYCORS, and MLMSRBF) is much less satisfactory, particularly under the restricted budget of 500 simulations.

Table 11: Statistical characteristics of NSE values obtained from all algorithms.

Budget / Statistics	MFE = 500					MFE = 1000				
	EAS	DDS	SEEAS	DYCORS	MLMSRBF	EAS	DDS	SEEAS	DYCORS	MLMSRBF
Min	0.331	0.329	0.279	0.204	0.141	0.331	0.330	0.331	0.246	0.268
Average	0.513	0.426	**0.632**	0.389	0.447	0.572	0.467	**0.727**	0.525	0.537
StDev	0.176	0.174	0.172	0.171	0.205	0.200	0.193	0.078	0.185	0.190
Median	0.448	0.331	**0.714**	0.306	0.369	0.719	0.331	**0.747**	0.505	0.651
Max	0.753	0.766	0.763	0.753	0.752	0.764	0.762	0.769	0.755	0.752

The above conclusions are further justified when comparing the convergence curves (Figure 10) and the CDFs (Figure 11) of the five algorithms. It is shown that after 300 (for MFE = 500) or 400 (for MFE = 1000) simulations, SEEAS evolves much faster, thus locating much higher NSE values than other algorithms. The performance of EAS is also very satisfactory, given that it clearly outperforms the other three state-of-the-art algorithms, two of which are also surrogate-assisted. Similarly, in terms of CDFs, in the low-budget scenario, SEEAS ensures NSE values greater than 0.65 in 23 out of 30 calibrations (Figure 11a). At the same problem, EAS performs better than other algorithms, particularly DDS, which is usually trapped to a remote local optimum. By increasing the computational budget to MFE = 1000, SEEAS systematically dominates all other schemes, ensuring NSE values greater than 0.70 in 28 out of 30 independent calibrations (Figure 11b).

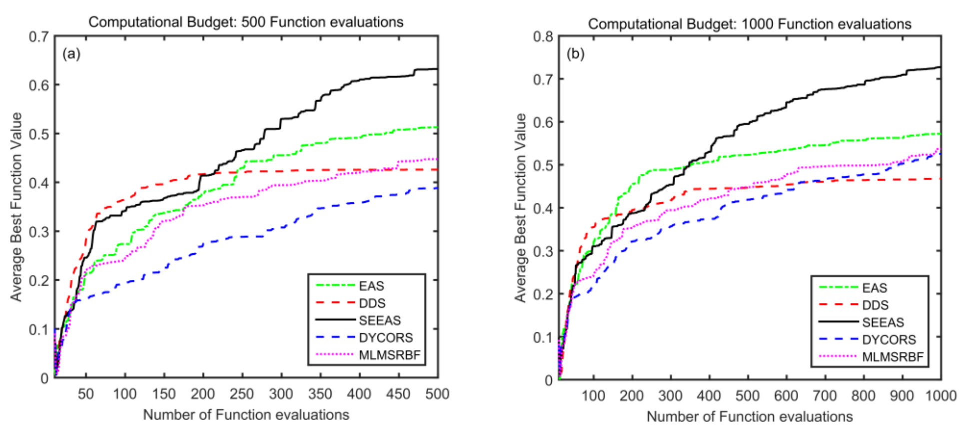


Figure 10: Convergence curves for MFE = 500 (a) and MFE = 1000 (b).

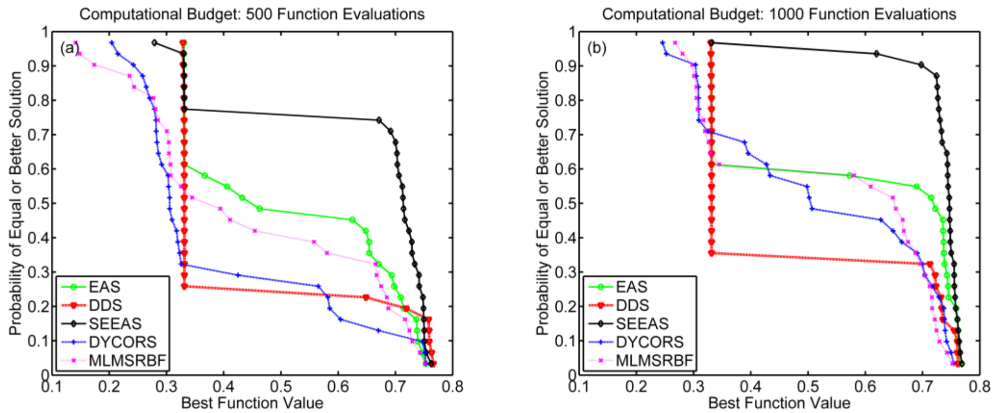


Figure 11: Empirical CDFs of best NSE values for MFE= 500 (a) and MFE = 1000 (b).

5.3 Toy calibration with synthetic runoff

As explained above, toy calibrations are more challenging, in the sense that the theoretical values of the model parameters are known, thus corresponding to unit efficiency. The outcomes of all associated tests, in terms of statistical characteristics of the best NSE value found so far, convergence curves and CDFs are shown in

Table 12, Figure 12 and Figure 13, respectively.

Table 12: Statistical characteristics of NSE values obtained from all algorithms.

Budget / Statistics	MFE = 500					MFE = 1000				
	EAS	DDS	SEEAS	DYCORS	SRBF	EAS	DDS	SEEAS	DYCORS	SRBF
Min	0.527	0.525	0.400	0.376	0.343	0.527	0.526	0.662	0.438	0.462
Average	0.688	0.694	**0.787**	0.641	0.640	0.793	0.742	**0.963**	0.679	0.735
StDev	0.172	0.206	0.207	0.186	0.213	0.215	0.211	0.058	0.202	0.204
Median	0.620	0.528	**0.910**	0.529	0.517	0.947	0.737	**0.981**	0.551	0.832
Max	0.981	0.987	0.978	0.961	0.970	0.989	0.986	0.987	0.975	0.980

The configuration of the calibration problem with synthetic runoff data further highlights the superiority of SEEAS against other algorithms. Specifically, the median NSE value found after only 500 simulations is 0.910, while the next best value is only 0.620, which is obtained through EAS. The increased computational budget ensures almost perfect calibrations (mean NSE = 0.963, median = 0.981), with minimal variability (standard deviation 0.058). For this budget, the median of EAS is also remarkably high (NSE = 0.947). Furthermore, SEEAS outperforms all other algorithms from the early search steps. Actually, for MFE = 500, until the first ~200 simulations DDS is competent, but then its improvement rate is significantly restricted. For the increased budget of 1000 simulations, SEEAS is clearly the best option, while EAS remains very competent. At the 2/3 of the budget, SEEAS achieves efficiency values up to 0.90, while EAS reaches values around 0.75. Even more exciting are the CDF charts, particularly for MFE = 1000; in this case, SEEAS achieves NSE values greater than 0.95 in 29 out of 30 calibration trials, and the original EAS approach also provides NSE values greater than 0.95 in 27 out of 30 trials. This obviously indicates the remarkable reliability and robustness of the two algorithms, in contrast to other methods that generally fail to reach the known optimum in reasonable time, thus requiring multiple independent runs to ensure statistically good calibrations.

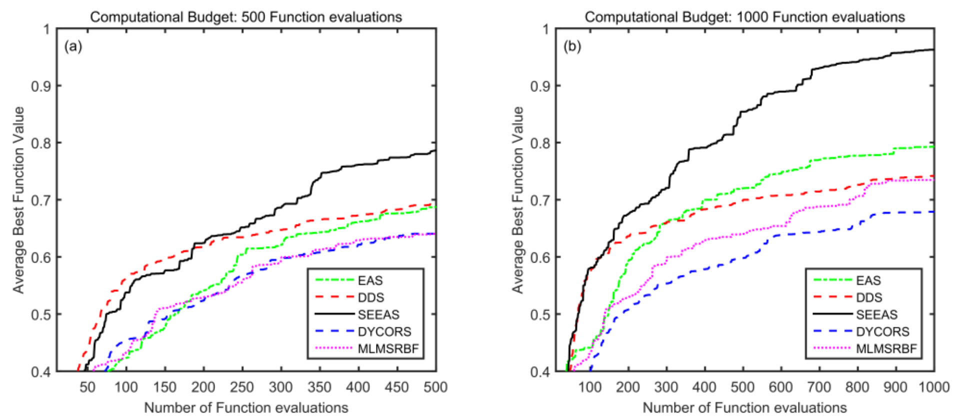


Figure 12: Convergence curves for MFE = 500 (a) and MFE = 1000 (b).

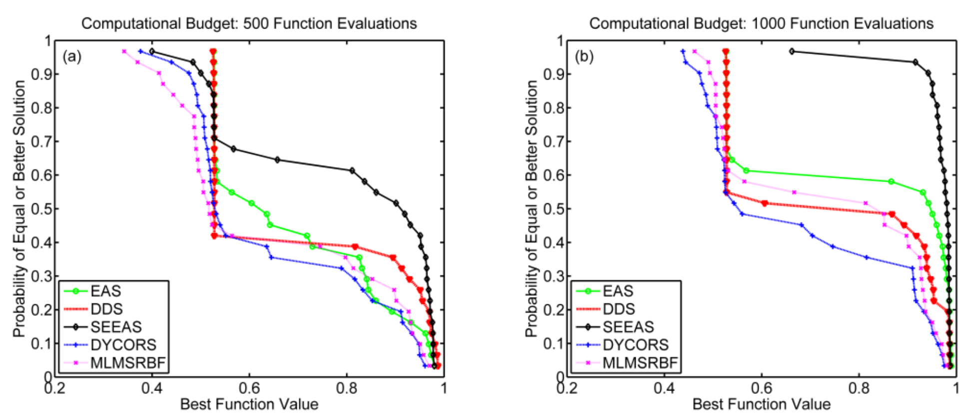


Figure 13: Empirical CDFs of best NSE values for MFE= 500 (a) and MFE = 1000 (b).

6 Optimization of multi-reservoir system performance

6.1 Problem statement

The second real-world test application involves the optimization of the operation of a multi-reservoir system in North-Eastern Greece. The objective was the development of uncertainty-aware operational rules that maximize the mean annual economic benefit of the system from energy production. The operation model of the hydrosystem is driven by synthetic hydrological data of 500 years length, thus drastically encumbering the computational time of simulation. In contrast to previous benchmarking tests, this problem is fully representative of real-world optimizations on a budget, given that a single function evaluation (i.e., a 500-year simulation) required ~90 s. Due to time limitations, we compared SEEAS only against the two other surrogate-assisted algorithms, i.e., DYCORS and MLMSRBF. For each algorithm, we employed 10 independent optimizations, allowing 500 function evaluations (thus each optimization run required about 12.5 hours).

6.2 The parameterization-simulation-optimization scheme

The reservoir system extends along the downstream branch of Nestos, a transboundary river shared by Bulgaria and Greece. It comprises three serially-connected hydroelectric reservoirs (Thesavros 381 MW; Platanovrysi 116 MW; Temenos 19 MW) and a small irrigation reservoir at the outlet. The first two power plants are reversible, thus employing pumped-storage to maximize energy efficiency. The river flows are mostly regulated in the most upstream reservoir (Thesavros), while the rest of projects have limited storage capacity.

The monthly operation of the system is represented by the well-known modelling tool WEAP21 (Yates et al., 2005). Hydrological inputs are inflows to Thesavros, as well as rainfall and evapotranspiration over all reservoir areas. The configuration of the simulation problem is explained in the recent articles by Tsoukalas and Makropoulos (2015a) and Tsoukalas and Makropoulos (2015b), where are also provided further details about the study area and associated data.

Since the size of historical hydrological data (1968-1982; 1991-1995) is not sufficient to extract safe conclusions about the long-term performance of the system, we used instead synthetic time series of 500 years length that were generated through Castalia software[5] (Efstratiadis et al., 2014a). Castalia employs a multivariate stochastic simulation scheme to generate synthetic time series that reproduce the statistical properties of the parent historical data, at multiple temporal scales. In the specific study, in which the time step of simulation is monthly, the model preserves the observed means, standard deviations, skewness coefficients, first order autocorrelations and cross-correlations at the monthly and annual scales; it also reproduces the long-term persistence (Hurst-Kolmogorov dynamics) at the annual and over-annual scales, thus accounting for the changing behavior of hydroclimatic processes (Koutsoyiannis, 2011).

The model decision variables were expressed in terms of energy targets, which are assigned to the associated system components, i.e., the three power plants (the two reversible). The targets refer to power production (forward operation of turbines) and consumption (backward operation, i.e., pumping). All targets were seasonally varying, considering four seasons per year, but they did not change over time (steady-state simulation). In this context, we parameterized the operation of the reservoir system through $(3 + 2) \times 4 = 20$ energy targets. The upper bound of target values was set equal to the installed capacity of the corresponding machine (turbine or pump), while all lower bounds were set zero. At each simulation step, for given (i.e., provided by the stochastic model) inflows and known initial conditions (reservoir storages), the model transforms energy targets to equivalent minimum flow constraints, thus forcing the model to pass the required amount of water to produce (or consume) the desired amount of energy.

In the formulation of the optimization problem, we assessed the long-term performance of the system in terms of mean energy benefit from the three energy components. Based on a slight

[5] http://www.itia.ntua.gr/en/softinfo/2/

modification of the expression introduced by Efstratiadis et al. (2012), we evaluated the monthly benefit b_i gained from each component i by:

$$b_i = c_B\, e_i^* + c_S \max(e_i - e_i^*, 0) + c_D \min(e_i - e_i^*, 0) - c_P\, p_i \tag{13}$$

where e_i^* is the target energy that corresponds to the specific season of the year, e_i and p_i are the actual energy production and consumption (in the case of pumped-storage), which are estimated through the simulation model, c_B and c_S are unit profits for firm and secondary energy production, respectively, c_D is a unit penalty cost for energy deficits, and c_P is the unit pumping cost. The unit profit or cost values were set 0.43, 0.23, 0.80 and 0.23 €/kWh, respectively.

6.3 Results

In Figure 14a are plotted the average convergence curves of the three algorithms, while in Figure 14b are illustrated the corresponding CDFs, estimated on the basis of optimal results obtained from 10 independent trials. Once again, SEEAS outperforms both DYCORS and MLMSRBF, considering the budget of 500 trials, although their differences are relatively small. Algorithms have almost similar behavior until ~300 FE, but then SEEAS evolves faster. In terms CDFs, SEEAS stochastically dominates MLMSRBF which, in turn, dominates DYCORS.

The two figures reveal the key peculiarity of reservoir optimization problems, which is the formulation of flat response surfaces, indicating low sensitivity of the system performance against the associated parameters. This is due to the existence of numerous constraints, physical and operational, which significantly restrict the flexibility of decisions. Generally, the decision variables of water management models represent desirable quantities (by means of target storages, target abstractions, target flows, etc.) that may be infeasible across a wide range of the decision space. In such cases, the actual (i.e., simulated) decisions and the system performance are mainly determined by the system constraints, which in turn results to the formulation of extended valleys across the response surface.

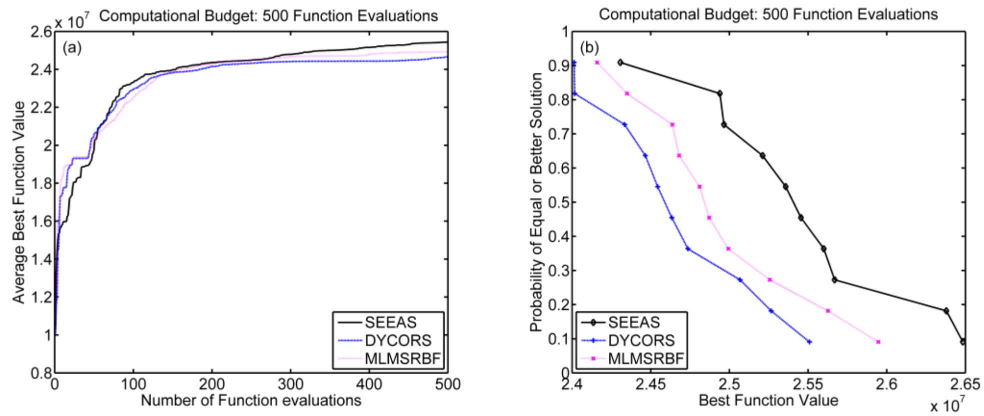


Figure 14: Convergence curves (a) and empirical CDFs (b) for MFE = 500.

7 Conclusions

The increasing model requirements, in order to allow process descriptions at fine spatial and temporal resolutions, have substantially increased the hardware requirements, in terms of computational resources and time. In this respect, surrogate-based optimization methods have gained significant attention, since they promise handling high-demanding optimization problems with limited budget. This study introduces a surrogate-enhanced extension of the evolutionary annealing-simplex (EAS) method. This new scheme, called SEEAS, uses the RBF metamodel to assist the generating mechanisms of EAS and identify promising solutions with low computational cost.

The effectiveness and efficiency of SEEAS have been demonstrated on the basis of a benchmarking suite comprising a variety of optimization problems, both theoretical and real-world. All problems were examined with alternative formulations and different budgets. The performance characteristics of SEEAS (statistical characteristics of best solution found so far, convergence behavior, stochastic dominance), were compared against three other state-of-the-art algorithms, as well as the original EAS algorithm. SEEAS outperformed all other methods in 18 out of 24 of theoretical problems (six functions with alternative configurations). Moreover, SEEAS was clearly superior in all real-world applications. Specifically, in hydrologic calibrations SEEAS showed consistency and robustness in locating near optimal solutions. In the first sub-case, using real runoff data, SEEAS located parameter sets with efficiency values larger than 0.70 in 28 out of 30 independent runs. Similarly, in the toy application with synthetic runoff, where the location of the optimum is *a priori* known, SEEAS ensured efficiency values larger than 0.90 in 29 out of 30 runs. Finally, in the optimization of the multi-reservoir system, SEEAS also exhibited the best behavior.

It is interesting mentioning that the two real-world applications are representative of the most typical global optimization problems in water resources. Both problems are very demanding, due to the complexity of their search space geometry. In particular, the goodness-of-fit measures used in calibrations generate highly irregular response surfaces with many local optima at all scales, in contrast to performance measures employed in water management problems, which usually compose extended smooth areas. A common characteristic of the two problems is the interactions between the model variables (or subsets of them), which is a major reason of multimodality (i.e., existence of multiple local optima with almost similar performance). The variety of generating mechanisms and quasi-stochastic transitions of SEEAS provide flexibility to handle search spaces with such peculiarities and so diverse geometry, while other algorithms seem to be less generic.

A well-known shortcoming of hybrid optimization algorithms (also including SEEAS) is the need for defining a number of input arguments which may confuse and even discourage non-experienced users. However, in the case of SEEAS, we recommend the use of generic values for the associated inputs, which have been specified after extended investigations. In fact, our analyses indicated that the algorithm is little sensitive against its input parameters, provided that reasonable values are assigned to them. This is also a strong evidence of the robustness of SEEAS.

Current research focuses on further improving the performance of SEEAS, by testing new simplex transformations and investigating other metamodels. Moreover, the authors are working towards extending SEEAS to handle noisy functions and developing a multi-objective version of the algorithm.

Acknowledgments

The authors would like to thank the Editor, Dr. Andrea Emilio Rizzoli, and the three reviewers, Dr. Saman Razavi and two anonymous ones, for their constructive comments, suggestions and critique, which helped providing a much improved manuscript.

References

Ackley, D.H., 1987. A connectionist machine for genetic hillclimbing. Kluwer Academic Publishers.

Ahmad, A., El-Shafie, A., Razali, S., Mohamad, Z., 2014. Reservoir Optimization in Water Resources: a Review. Water Resources Management 28(11) 3391-3405.

Blanning, R.W., 1975. The construction and implementation of metamodels. Transactions of The Society for Modeling and Simulation International 24(6) 177-184.

Broad, D., Dandy, G., Maier, H., 2005. Water Distribution System Optimization Using Metamodels. Journal of Water Resoures Planning and Management 131(3) 172-180.

Buhmann, M.D., 2003. Radial Basis Functions. Cambridge University Press.

Castelletti, A., Galelli, S., Ratto, M., Soncini-Sessa, R., Young, P.C., 2012a. A general framework for Dynamic Emulation Modelling in environmental problems. Environmental Modelling & Software 34(0) 5-18.

Castelletti, A., Galelli, S., Restelli, M., Soncini-Sessa, R., 2012b. Data-driven dynamic emulation modelling for the optimal management of environmental systems. Environmental Modelling & Software 34(0) 30-43.

Cheng, C.-T., Wu, X.-Y., Chau, K.W., 2005. Multiple criteria rainfall–runoff model calibration using a parallel genetic algorithm in a cluster of computers / Calage multi-critères en modélisation pluie–débit par un algorithme génétique parallèle mis en œuvre par une grappe d'ordinateurs. Hydrological Sciences Journal 50(6) null-1087.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine Learning 20(3) 273-297.

Couckuyt, I., Deschrijver, D., Dhaene, T., 2013. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. Journal of Global Optimization 1-20.

Dias, B.H., Tomim, M.A., Marcato, A.L.M., Ramos, T.P., Brandi, R.B.S., Junior, I.C.d.S., Filho, J.A.P., 2013. Parallel computing applied to the stochastic dynamic programming for long term operation planning of hydrothermal power systems. European Journal of Operational Research 229(1) 212-222.

Dibike, Y., Velickov, S., Solomatine, D., Abbott, M., 2001. Model Induction with Support Vector Machines: Introduction and Applications. Journal of Computing in Civil Engineering 15(3) 208-216.

Duan, Q., 2013. Global Optimization for Watershed Model Calibration, Calibration of Watershed Models. American Geophysical Union, pp. 89-104.

Efstratiadis, A., D. Bouziotas, Koutsoyiannis, D., 2012. The parameterization-simulation-optimisation framework for the management of hydroelectric reservoir systems, Hydrology and Society, EGU Leonardo Topical Conference Series on the hydrological cycle 2012: Torino, European Geosciences Union.

Efstratiadis, A., Dialynas, Y., Kozanis, S., Koutsoyiannis, D., 2014a. A multivariate stochastic model for the generation of synthetic time series at multiple time scales reproducing long-term persistence. Environmental Modelling & Software 62(0) 139-152.

Efstratiadis, A., Koutsoyiannis, D., 2002. An evolutionary annealing-simplex algorithm for global optimisation of water resource systems, 5th International Conference on Hydroinformatics, Cardiff, UK, 1423–1428, IWA.

Efstratiadis, A., Koutsoyiannis, D., 2008. Fitting Hydrological Models on Multiple Responses Using the Multiobjective Evolutionary Annealing-Simplex Approach, In: Abrahart, R., See, L., Solomatine, D. (Eds.), Practical Hydroinformatics. Springer Berlin Heidelberg, pp. 259-273.

Efstratiadis, A., Koutsoyiannis, D., 2010. One decade of multi-objective calibration approaches in hydrological modelling: a review. Hydrological Sciences Journal-Journal Des Sciences Hydrologiques 55(1) 58-78.

Efstratiadis, A., Koutsoyiannis, D., Xenos, D., 2004. Minimizing water cost in water resource management of Athens. Urban Water Journal 1(1) 3-15.

Efstratiadis, A., Nalbantis, I., Koukouvinos, A., Rozos, E., Koutsoyiannis, D., 2008. HYDROGEIOS: a semi-distributed GIS-based hydrological model for modified river basins. Hydrol. Earth Syst. Sci. 12(4) 989-1006.

Efstratiadis, A., Nalbantis, I., Koutsoyiannis, D., 2014b. Hydrological modelling of temporally-varying catchments: facets of change and the value of information. Hydrological Sciences Journal null-null.

Feyen, L., Vrugt, J.a., Nualláin, B.Ó., van der Knijff, J., De Roo, A., 2007. Parameter optimisation and uncertainty assessment for large-scale streamflow simulation with the LISFLOOD model. Journal of Hydrology 332(3-4) 276-289.

Forrester, A., Keane, A., 2009. Recent advances in surrogate-based optimization. Progress in Aerospace Sciences 45(1–3) 50-79.

Fowler, K.R., Reese, J.P., Kees, C.E., Dennis Jr, 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. Advances in Water Resources 31(5) 743-757.

Giunta, A.A., Wojtkiewicz Jr, S.F., Eldred, M.S., 2003. Overview of Modern Design of Experiments Methods for Computational Simulations, Proceedings of the 41st AIAA Aerospace Sciences Meeting and Exhibit: Reno, NV.

Griewank, A.O., 1981. Generalized descent for global optimization. Journal of Optimization Theory and Applications 34(1) 11-39.

He, K., Li, Z., Shoubin, D., Liqun, T., Jianfeng, W., Chunmiao, Z., 2007. PGO: A parallel computing platform for global optimization based on genetic algorithm. Comput. Geosci. 33(3) 357-366.

Jin, Y., 2005. A comprehensive survey of fitness approximation in evolutionary computation. Soft Computing 9(1) 3-12.

Jin, Y., 2011. Surrogate-assisted evolutionary computation: Recent advances and future challenges. Swarm and Evolutionary Computation 1(2) 61-70.

Jones, D., Schonlau, M., Welch, W., 1998. Efficient global optimization of expensive black-box functions. Journal of Global Optimization 13 455-492.

Jong, K.A.D., 1975. An analysis of the behavior of a class of genetic adaptive systems. University of Michigan, p. 266.

Keating, E.H., Doherty, J., Vrugt, J.A., Kang, Q., 2010. Optimization and uncertainty assessment of strongly nonlinear groundwater models with high parameter dimensionality. Water Resources Research 46(10) W10517.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization, Neural Networks, 1995. Proceedings., IEEE International Conference on, pp. 1942-1948 vol.1944.

Knowles, J., 2005. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multi-objective optimization problems. IEEE Transactions on Evolutionary Computation 10(1) 50-66.

Kossieris, P., Efstratiadis, A., Koutsoyiannis, D., 2013. The use of stochastic objective functions in water resource optimization problems, 5th EGU Leonardo Conference – Hydrofractals 2013 – STAHY '13, Kos Island, Greece, European Geosciences Union, International Association of Hydrological Sciences, International Union of Geodesy and Geophysics.

Kossieris, P., Koutsoyiannis D., Onof C., Tyralis H., Efstratiadis A., 2012. HyetosR: An R package for temporal stochastic simulation of rainfall at fine time scales, European Geosciences Union General Assembly: Geophysical Research Abstracts, Vol. 14, Vienna, 11718, European Geosciences Union.

Kourakos, G., Mantoglou, A., 2009. Pumping optimization of coastal aquifers based on evolutionary algorithms and surrogate modular neural network models. Advances in Water Resources 32(4) 507-521.

Koutsoyiannis, D., 2011. Hurst-Kolmogorov Dynamics and Uncertainty. JAWRA Journal of the American Water Resources Association 47(3) 481-495.

Koutsoyiannis, D., Economou, A., 2003. Evaluation of the parameterization-simulation-optimization approach for the control of reservoir systems. Water Resources Research 39(6) 1170.

Kuzmin, V., Seo, D.-J., Koren, V., 2008. Fast and efficient optimization of hydrologic model parameters using a priori estimates and stepwise line search. Journal of Hydrology 353(1-2) 109-128.

Labadie, J.W., 2004. Optimal operation of multireservoir systems: State-of-the-art review. Journal of Water Resources Planning and Management-Asce 130(2) 93-111.

Levy, A., Montalvo, A., 1985. The Tunneling Algorithm for the Global Minimization of Functions. SIAM Journal on Scientific and Statistical Computing 6(1) 15-29.

Levy, H., 1992. Stochastic dominance and expected utility: survey and analysis. Manage. Sci. 38(4) 555-593.

Müller, J., Shoemaker, C., 2014. Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems. Journal of Global Optimization 60(2) 123-144.

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., Castelletti, A., Giuliani, M., Reed, P.M., 2014. Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions. Environmental Modelling & Software 62(0) 271-299.

Mann, H.B., Whitney, D.R., 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. 50-60.

Matott, L.S., Tolson, B.A., Asadzadeh, M., 2012. A benchmarking framework for simulation-based optimization of environmental models. Environmental Modelling & Software 35(0) 19-30.

Mugunthan, P., Shoemaker, C.A., 2006. Assessing the impacts of parameter uncertainty for computationally expensive groundwater models. Water Resources Research 42(10) W10428.

Mugunthan, P., Shoemaker, C.A., Regis, R.G., 2005. Comparison of function approximation, heuristic, and derivative-based methods for automatic calibration of computationally expensive groundwater bioremediation models. Water Resources Research 41(11) 1-17.

Myers, R.H., Montgomery, D.C., 1995. Response Surface Methodology: Process and Product in Optimization Using Designed Experiments. John Wiley \& Sons, Inc.

Nalbantis, I., Efstratiadis, A., Rozos, E., Kopsiafti, M., Koutsoyiannis, D., 2011. Holistic versus monomeric strategies for hydrological modelling of human-modified hydrosystems. Hydrol. Earth Syst. Sci. 15(3) 743-758.

Nelder, J.A., Mead, R., 1965. A Simplex Method for Function Minimization. The Computer Journal 7(4) 308-313.

Nicklow, J., Reed, P., Savic, D., Dessalegne, T., Harrell, L., Chan-Hilton, A., Karamouz, M., Minsker, B., Ostfeld, A., Singh, A., Zechman, E., Evolutionary, A.T.C., 2010. State of the Art for Genetic Algorithms and Beyond in Water Resources Planning and Management. Journal of Water Resources Planning and Management-Asce 136(4) 412-432.

Ostfeld, A., Salomons, S., 2005. A hybrid genetic - instance based learning algorithm for CE-QUAL-W2 calibration. Journal of Hydrology 310(1-4) 122-142.

Pan, L., Wu, L., 1998. A hybrid global optimization method for inverse estimation of hydraulic parameters: Annealing-Simplex Method. Water Resources Research 34(9) 2261-2269.

Ponweiser, W., Wagner, T., Biermann, D., Vincze, M., 2008. Multiobjective Optimization on a Limited Budget of Evaluations Using Model-Assisted \mathcal{S}-Metric Selection, In: Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N. (Eds.), Parallel Problem Solving from Nature – PPSN X. Springer Berlin Heidelberg, pp. 784-794.

Powell, M.J.D., 1992. The theory of radial basis function approximation in 1990, Light, Ed Advances in Numerical Analysis Advances in Numerical Analysis, vol. 2: wavelets,subdivision algorithms and radial basis functions. Oxford University Press, Oxford, pp. 105-210.

Rastrigin, L.A., 1974. Systems of extremal control, Nauka, Moscow. (In Russian).

Razavi, S., Tolson, B.A., Burn, D.H., 2012a. Numerical assessment of metamodelling strategies in computationally intensive optimization. Environmental Modelling & Software 34(0) 67-86.

Razavi, S., Tolson, B.A., Burn, D.H., 2012b. Review of surrogate modeling in water resources. Water Resources Research 48(7) W07401.

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 Resources Research 46(11) W11523.

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. Advances in Water Resources 51 438-456.

Regis, R., Shoemaker, C., 2009. Parallel Stochastic Global Optimization Using Radial Basis Functions. INFORMS J. on Computing 21(3) 411-426.

Regis, R., Shoemaker, C., 2013. Combining radial basis function surrogates and dynamic coordinate search in high-dimensional expensive black-box optimization. Engineering Optimization 45(5) 529-555.

Regis, R.G., 2011. Stochastic radial basis function algorithms for large-scale optimization involving expensive black-box objective and constraint functions. Computers & Operations Research 38(5) 837-853.

Regis, R.G., 2014. Particle swarm with radial basis function surrogates for expensive black-box optimization. Journal of Computational Science 5(1) 12-23.

Regis, R.G., Shoemaker, C.A., 2004. Local function approximation in evolutionary algorithms for the optimization of costly functions. IEEE Transactions on Evolutionary Computation 8(5) 490-505.

Regis, R.G., Shoemaker, C.A., 2007a. Improved strategies for radial basis function methods for global optimization. Journal of Global Optimization 37(1) 113-135.

Regis, R.G., Shoemaker, C.A., 2007b. A stochastic radial basis function method for the global optimization of expensive functions. Informs Journal on Computing 19(4) 497-509.

Rozos, E., Efstratiadis, A., Nalbantis, I., Koutsoyiannis, D., 2004. Calibration of a semi-distributed model for conjunctive simulation of surface and groundwater flows / Calage d’un modèle semi-distribué pour la simulation conjointe d’écoulements superficiels et souterrains. Hydrological Sciences Journal 49(5) null-842.

Sacks, J., Welch, W., Mitchell, T., Wynn, H., 1989. Design and analysis of computer experiments (with discussion). Journal of Statistical Science 4 409-435.

Sasena, M., Papalambros, P., Goovaerts, P., 2002. Exploration of Metamodeling Sampling Criteria for Constrained Global Optimization. Engineering Optimization 34(3) 263-278.

Schutte, J.F., Reinbolt, J.A., Fregly, B.J., Haftka, R.T., George, A.D., 2004. Parallel global optimization with the particle swarm algorithm. International Journal for Numerical Methods in Engineering 61(13) 2296-2315.

Shoemaker, C.A., Regis, R.G., Fleming, R.C., 2007. Watershed calibration using multistart local optimization and evolutionary optimization with radial basis function approximation. Hydrological Sciences Journal-Journal Des Sciences Hydrologiques 52(3) 450-465.

Tan, C.-C., Tung, C.-P., Chen, C.-H., Yeh, W.W.G., 2008. An integrated optimization algorithm for parameter structure identification in groundwater modeling. Advances in Water Resources 31(3) 545-560.

Tang, Y., Chen, J., Wei, J., 2012. A surrogate-based particle swarm optimization algorithm for solving optimization problems with expensive black box functions. Engineering Optimization 45(5) 557-576.

Tolson, B.A., Asadzadeh, M., Maier, H.R., Zecchin, A., 2009. Hybrid discrete dynamically dimensioned search (HD-DDS) algorithm for water distribution system design optimization. Water Resources Research 45(12) W12416.

Tolson, B.A., Shoemaker, C.A., 2007. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. Water Resources Research 43(1).

Tsoukalas, I., Makropoulos, C., 2015a. Multiobjective optimisation on a budget: Exploring surrogate modelling for robust multi-reservoir rules generation under hydrological uncertainty. Environmental Modelling & Software 69 396-413.

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 Resources Management 29(13) 4719-4734.

Vrugt, J.A., O Nuallain, B., Robinson, B.A., Bouten, W., Dekker, S.C., Sloot, P.M.A., 2006. Application of parallel computing to stochastic parameter estimation in environmental models. Computers & Geosciences 32(8) 1139-1155.

Yates, D., Sieber, J., Purkey, D., Huber-Lee, A., 2005. WEAP21: A Demand, priority, and preference driver water planning model. Part 1: Model Characteristics. Water International 30 487-500.

Zakharov G., K., 1981. Optimization of step-wise control systems. Avtomat. i Telemekh 8 5-9.

Zou, R., Lung, W.-S., Wu, J., 2007. An adaptive neural network embedded genetic algorithm approach for inverse water quality modeling. Water Resources Research 43(8).

Appendix: Mathematical Test Functions

A. *OF1*, *OF2*: *De Jong's Function 1* (*Sphere Function*) (Jong, 1975).

$$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2$$

$$x_i \in [-5.12,\ 5.12],\ i = 1, \ldots, d$$

$$OF1:\ d = 15,\ OF2:\ d = 30$$

$$\text{minimum: } f(\mathbf{x}^*) = 0,\ \text{at } \mathbf{x}^* = (0, \ldots, 0)$$

B. *OF3*, *OF4*: *Ackley Function* (Ackley, 1987).

$$f(\mathbf{x}) = -20\exp(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}) - \exp(\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)) + 20 + \exp(1)$$

$$x_i \in [-32.768,\ 32.768],\ i = 1, \ldots, d$$

$$OF3:\ d = 15,\ OF4:\ d = 30$$

$$\text{minimum: } f(\mathbf{x}^*) = 0,\ \text{at } \mathbf{x}^* = (0, \ldots, 0)$$

C. OF5, OF6: Griewank function (Griewank, 1981).

$$f(\mathbf{x}) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d}\cos(\frac{x_i}{\sqrt{i}}) + 1$$

$$x_i \in [-600,\ 600],\ i = 1, \ldots, d$$

$$OF5:\ d = 15,\ OF6:\ d = 30$$

$$\text{minimum: } f(\mathbf{x}^*) = 0,\ \text{at } \mathbf{x}^* = (0, \ldots, 0)$$

D. OF7, OF8: Zakharov function (Zakharov G., 1981).

$$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2 + (\sum_{i=1}^{d} 0.5 i x_i)^2 + (\sum_{i=1}^{d} 0.5 i x_i)^4$$

$$x_i \in [-5,\ 10],\ i = 1, \ldots, d$$

$$OF7:\ d = 15,\ OF8:\ d = 30$$

$$\text{minimum: } f(\mathbf{x}^*) = 0,\ \text{at } \mathbf{x}^* = (0, \ldots, 0)$$

E. OF9, OF10: Rastrigin function (Rastrigin, 1974).

$$f(\mathbf{x}) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$$

$$x_i \in [-5.12,\ 5.12],\ i = 1, \ldots, d$$

$OF9:\ d=15,\ OF10:\ d=30$

minimum: $f(\mathbf{x}^*)=0,$ at $\mathbf{x}^*=(0,\ldots,0)$

F. *OF11, OF12: Levy function* (Levy and Montalvo, 1985).

$$f(\mathbf{x})=\sin^2(\pi\omega_1)+\sum_{i=1}^{d-1}(\omega_i-1)^2[1+10\sin^2(\pi\omega_i+1)]+(\omega_d-1)^2[1+\sin^2(2\pi\omega_d)]\ ,$$

where $\omega_i=1+\dfrac{x_i-1}{4}$ for all $i=1,\ldots,d$

$x_i\in[-10,\ 10],\ i=1,\ldots,d$

$OF11:\ d=15,\ OF12:\ d=30$

minimum: $f(\mathbf{x}^*)=0,$ at $\mathbf{x}^*=(1,\ldots,1)$