CrossMark

Bayesian optimization of pump operations in water distribution systems

A. Candelieri[1] · **R. Perego**[1] · **F. Archetti**[1]

© The Author(s) 2018

Abstract Bayesian optimization has become a widely used tool in the optimization and machine learning communities. It is suitable to problems as simulation/optimization and/or with an objective function computationally expensive to evaluate. Bayesian optimization is based on a surrogate probabilistic model of the objective whose mean and variance are sequentially updated using the observations and an "acquisition" function based on the model, which sets the next observation at the most "promising" point. The most used surrogate model is the Gaussian Process which is the basis of well-known Kriging algorithms. In this paper, the authors consider the pump scheduling optimization problem in a Water Distribution Network with both ON/OFF and variable speed pumps. In a global optimization model, accounting for time patterns of demand and energy price allows significant cost savings. Nonlinearities, and binary decisions in the case of ON/OFF pumps, make pump scheduling optimization computationally challenging, even for small Water Distribution Networks. The well-known EPANET simulator is used to compute the energy cost associated to a pump schedule and to verify that hydraulic constraints are not violated and demand is met. Two Bayesian Optimization approaches are proposed in this paper, where the surrogate model is based on a Gaussian Process and a Random Forest, respectively. Both approaches are tested with different acquisition functions on a set of test functions, a benchmark Water Distribution Network from the literature and a large-scale real-life Water Distribution Network in Milan, Italy.

Keywords Global optimization · Bayesian optimization · Pump scheduling optimization · Simulation optimization

✉ A. Candelieri
candelieriantonio@gmail.com

1 Department of Computer Science, Systems and Communications, University of Milano-Bicocca, Viale Sarca 336, 20125 Milan, Italy

1 Introduction

Optimization of Water Distribution Networks (WDNs) operations has been a very important field for the Operation Research (OR) community at least in the last 40 years [1] and many tools from mathematical programming as well as metaheuristics have been proposed. An updated review on optimal water distribution is given in [2] where several classes of existing solutions, including linear programming, nonlinear programming, dynamic programming, metamodeling, heuristics, and metaheuristics are deeply analyzed and referenced. Different issues are considered, such as solutions for sensors placement and leakage detection and localization [3], and more in detail optimization of pumping operations.

In this paper the authors are concerned with pump scheduling optimization (PSO): which pumps are to be operated and with which settings at different periods of the day, so that the energy cost, the largest operational cost for water utilities, is minimized. Nonlinearities, and binary decisions in the case of ON/OFF pumps, make PSO computationally challenging, even for small WDNs [4]. While mathematical programming approaches linearize/convexify the equations regulating the flow distribution in the WDN, the more recent optimization strategies use a hydraulic simulation software which can solve all the equations and provide computed values relatively to objective function (e.g. energy costs) and hydraulic feasibility (e.g. satisfaction of the demand, pressures within a given range, tanks level within min–max range, etc.).

The decision variables can be formulated in 2 ways: the ON/OFF pump state during fixed time intervals or the start/end run times of pumps [5]. Even if the latter results in a decrease in the number of decision variables, the former is the most widely used and will be considered in this paper.

If there are storage tanks in the network, cost reductions can be made by concentrating most of pumping during the time windows when electricity is least expensive and by using this water capacity during the remaining part of the day to meet consumer demand. According to this aim, an accurate demand forecasting system, as the one proposed in [6–8], can be used to set the demand, which basically drives every simulation run, in order to optimize pump schedule in advance. The cost is generally associated with the cost of energy for pumping, and the system performance requires satisfying operational constraints as: assuring pumps can satisfy user demand, keeping pressures within certain bounds to reduce leakage and the risk of pipe burst, keeping reservoir levels within bounds to avoid overflow.

As the sizes of the tested example networks increase from simplified hypothetical examples to large real networks, the number of variables and objectives considered grows leading to increase in the number of function evaluations, computational resources (simulation time increases with the number of hydraulic components of the WDN) and memory requirements. Moreover, the headloss equations used to model water flow in pipes lead to nonlinear optimization problems [9].

Linear approaches have been considered extensively in the past, emphasizing their advantage for obtaining fast solutions, however they require linearity for objective function and constraints, and their performance depends on finding a suitable linearization of the problem at hand. Pasha and Lansey in [10], to linearize the problem, rely on the relationship between energy, pump flow, user demand and tank water levels, while in [11] head-loss convex equation from water supply systems was iteratively linearized and incorporated in linear programming optimization models. As the number of decision variables and constraints increases considerably with the number of time-control intervals, this leads to increased computational and memory requirements. Also nonlinear programming has been used, however since the problem is non-convex, there is no guarantee that the global optimum will be found [12].

Since the 1990's metaheuristics, such as Genetic Algorithms and Simulated Annealing among others, have been increasingly applied given their potential to solve nonlinear, nonconvex and even black-box problems [13] where traditional mathematical programming methods would face difficulties.

In [14] a hybrid approach was applied where genetic algorithm was coupled with two hill-climber search algorithms for improving local search and even though this approach allowed for better performance, in terms of efficiency, than classical genetic algorithms, it still could not be applied in near-real time conditions. Similarly, [12] applied hydraulic network linearization for two stage simulated annealing approach and while the approach finds global near optimal solution it is still limited to off-line optimization problems. Overall, metaheuristic optimization methods offer versatility and a problem-independent global approach that can be applied to complex problems, but over large search spaces, the coupling with the network simulator results in lengthy computation when searching for the optimum solution. Generally, the application of these methods to pump scheduling is rarer since the solutions need to be produced both rapidly and reliably. For these reasons, more efficient deterministic methods have been applied but the simulation runs might still constrain their application to mid-size networks. Thus, all the optimization approaches require to reduce the computational load of the simulator (which is usually the open source EPANET 2.0 [15]): one way to do this is "internal" working on the mathematical model of the hydraulic components, the other is "external" and takes the form of fitting a surrogate function or meta-model which approximates the simulation environment and can be used to drive global optimization strategies.

In this paper we propose Bayesian Optimization (BO) for the solution of PSO, as it is suitable to solve simulation–optimization problems and/or to be applied when the objective function is a black-box as well as it is computationally expensive to evaluate.

The diagonal approach [16] offers global convergence properties and guaranteed accuracy estimation of the global solutions, e.g., in the case of Lipschitz global optimization [17, 18]. A significant application of these methods to hyper-parameter optimization is given, e.g. in [19] for the case of SVM regression or in [20] for the case of signal processing. Random Search [21, 22] and the related metaheuristics like simulated annealing, evolutionary/genetic algorithms, multistart&clustering have recently received fresh attention from the machine learning community and is increasingly considered as a baseline for global optimization as in Hyperband [23]. A further approach is related to the adoption of probabilistic branch and bound for level set estimation [24]. Sequential Model Based Optimization (SMBO), and in particular Bayesian Optimization, came to dominate the landscape of hyper-parameter optimization in the machine learning community [25–27]. BO's key ingredients are a probabilistic model which captures our beliefs—given the evaluations already performed—and an acquisition function which computes the "utility" of each candidate point for the next evaluation of f. The Bayesian framework is particularly useful when evaluations of f are costly, no derivatives are available and/or f is nonconvex/multimodal.

Two alternatives BO approaches are proposed in this paper: a method using Gaussian Processes (GP) to fit the surrogate model and another using Random Forest (RF), which is an ensemble of Decision Trees. Although GP is the most widely adopted approach, RF recently turns out to be more computationally efficient [28], in particular when some decision variables are discrete and/or conditional. A similar framework has been proposed to address Kernel-based Clustering in a leakage localization application [29]. Both GP and RF are presented in Sect. 3.1.

The remainder of the paper is organized as follows. Section 2 presents the pump scheduling optimization (PSO) problem in Water Distribution Networks (WDN), addressing both ON/OFF pumps and Variable Speed Pumps (VSP) and by considering, in the optimization

model, the time patterns of demand and energy price. The well-known EPANET 2.0 simulator is used to check, for every pump schedule, that hydraulic constraints are not violated, demand is met and to compute the associated energy cost. Section 3 is devoted to introduce some background on BO, focusing on surrogate models and acquisition functions considered in the implementation of the proposed BO framework. Section 4 presents the computational results obtained on both test functions and the PSO problem, by comparing different alternatives in the BO framework's set up. Results on WDN refer to two different case studies: a benchmark WDN widely adopted in literature (namely, Anytown) and a real-life large-scale WDN in Milan, Italy. The proposed BO approaches are compared to two commercial products: an "optimization-as-a-service" cloud-based platform (SigOpt[1]) and the combination of ALAMO[2] with BARON[3] for the generation of an accurate surrogate model of the analyzed system and the successive optimization process. Finally, Sect. 5 summarizes relevant findings.

2 Pump scheduling optimization

The goal of PSO is to identify the pump schedule, consisting in the status of each pump over time, associated to the lowest energy cost while satisfying hydraulic feasibility (i.e., water demand satisfaction, pressure level within a given operational range, etc.). Status of a pump is its activation—in the case of an ON/OFF pump—or its speed—in the case of a VSP, leading to discrete or continuous decision variables, respectively. Thus, a general formulation of the PSO, involving both the two types of decision variables (i.e. a WDN with both ON/OFF pumps and VSP), is reported as follows:

Objective function (minimizing the energy costs)

$$\min C = \Delta t \sum_{t=1}^{T} c_E^t \left(\sum_{k_b=1}^{N_b} \gamma x_{k_b}^t Q_{k_b}^t \frac{H_{k_b}^t}{\eta_{k_b}} + \sum_{k_v=1}^{N_v} \gamma x_{k_v}^t Q_{k_v}^t \frac{H_{k_v}^t}{\eta_{k_v}} \right) \tag{1}$$

where C — total energy cost over T time steps, Δt — time step width, C_E^t — energy cost at time t, γ — specific weight of water, $Q_{k_j}^t$ — water flow provided through the pump k_j at time t, $H_{k_j}^t$ — head loss on pump k_j at time t, η_{k_j} — efficiency of the pump k_j and the decision variables are:

$x_{k_b}^t \in \{0, 1\}$ with $t = 1, \ldots, T$ and $k_b = 1, \ldots, N_b$ — status of pump k_b at time t (i.e. $ON = 1$, $OFF = 0$)

$x_{k_v}^t \in (0, 1)$ with $t = 1, \ldots, T$ and $k_v = 1, \ldots, N_v$ — speed of pump k_v at time t

where N_b is the number of ON/OFF pumps and N_v is the number of VSPs.

The time horizon usually adopted in PSO is 1 day with hourly time steps, leading to $T = 24$ time steps overall. This choice is basically motivated by the energy tariffs which are typically characterized by a different energy price depending on the hour of the day.

Although the PSO objective function (1) has an apparently closed expression, its exact resolution through mathematical programming requires to approximate the flow equations (linearization or convexification) which regulates the hydraulic behaviour of a pressurized WDN. More recent approaches address the optimization of the objective function by con-

---

[1] https://sigopt.com/.

[2] http://www.minlp.com/alamo.

[3] http://www.minlp.com/baron.

sidering it as "black-box", computing the value associated to a given schedule through a hydraulic simulation software which solves all the flow equations. Naturally, hydraulic simulation is performed at the selected time resolution—usually hourly basis for 24 h—which is adopted for the input data (i.e. demand at consumption points) as well as the outputs obtained at the end of a simulation run (i.e. pressure at nodes, flow/velocity on the pipes).

Depending on the size and complexity of the WDN, time required for a complete simulation can significantly change. This means that in the case of very large and complex WDN evaluating a schedule may also become significantly expensive in terms of time.

Although not expressed in the formulation above, there exists a set of hydraulic constraints which are checked by the hydraulic simulation software. Even in papers reporting the analytical formulation of the constraints, such as in [30], this check is usually verified by running a simulation. A violation message is usually provided by the software in the case a given pump schedule does not satisfy at least one of the hydraulic constraints. Some examples are: a negative pressure in some point of the WDN, impossibility to supply the demand at some consumption point, imbalance of the overall hydraulic system, etc. The hydraulic simulation software used in this study is EPANET 2.0 and a detailed list of errors/warnings provided by EPANET, with respect to hydraulic unfeasibility of a pump schedule, is reported in [15].

Moreover, some further "operational" constraints can be also added to the formulation. For instance, a common requirement is to have pressure values within a given range (not just positive values). Usually, when real-time monitoring is considered, this constraint regards a limited set of nodes of the WDN, which are the most critical ones (those showing the lowest and the highest pressure in the WDN). On the other hand, as simulation can estimate pressure at every node, this constraint can be specified for all of them.

$$P_i^{\min} \leq P_i^t \leq P_i^{Max} \quad i = 1, \ldots, N_{nodes}$$

where P_i^t is the pressure on node i at time t.

Another typical operational constraint is related to the level of water into tank(s): in many studies the level of the tank(s) at the end of the day is required to be not lower than that at the beginning of the same day. This constrain can be mathematically formulated as:

$$L_l^{t=0} \leq L_l^{t=24}$$

where L_l^t is the level of the l-th tanks at time t and $l = 1, \ldots, N_L$, with N_L the number of tanks in the WDN.

Again, P_i^t and L_l^t are computed through hydraulic software simulation.

PSO is computationally hard to solve through an exhaustive search, due to the large number of possible pump schedules even for a very tiny WDN. A simple WDN, consisting of just 1 ON/OFF pump, and considering an hourly resolution and a 24-hours horizon, leads to an optimization problem with 24 discrete decision variables and, consequently, 2^{24} (i.e. more than 16 million) possible pump schedules.

An efficient search method to identify promising pump schedule is therefore crucial to implement a practical decision support for PSO, with the aim to identify an optimal schedule within a limited number of objective function evaluations. This is the motivation of the BO framework for PSO proposed in this paper.

3 Bayesian optimization

The simulation–optimization task is addressed in this study through Sequential Model Based Optimization (SMBO) and, specifically, Bayesian Optimization. The goal is to find a global minimizer of the (black-box) objective function:

$$x^* = \arg\min_x f(x)$$

where $x \in \mathbb{R}^d$ is a d-dimensional vector whose components are the value of the decision variables. In the global optimization x is usually defined in a limited domain (bounding box). In the most general case decision variables can be continuous, discrete as well as conditional.

The BO framework has 2 main components:

- The first is a probabilistic model of the objective function (also known as "surrogate function") whose mean and variance are sequentially updated accordingly to observations of the objective function. Two such models are considered in Sect. 3.1, a Gaussian Process (GP) and a Random Forest (RF).
- The second key component is the acquisition function based on the GP model, whose optimization drives the querying process of the model and sets the next observation at the most "promising" point. Several acquisition functions are considered in Sect. 3.2.

BO has become the most promising approach even in the case there is not any information about derivatives as well as $f(x)$ is not convex or multi-modal. However, when gradients of $f(x)$ can be inferred by the surrogate, they can be incorporated in the BO framework to improve the value through a local search [31].

With respect to the PSO problem, $f(x)$ is the energy cost C reported in (1) while the dimension d of the search space is given by $T \cdot (N_b + N_v)$, where T is the number of time steps and N_b and N_V are the number of ON/OFF and variable speed pumps, respectively. Every component of x refers to a specific time step and a specific pump, so $x_i \in \{0, 1\}$ in case of an ON/OFF pump and $x_i \in [0, 1]$ in case of a VSP.

This notation holds in the following sections which are related to the probabilistic surrogate models and acquisition functions used in this study.

3.1 The probabilistic model

A GP is an extension of the multivariate Gaussian distribution to an infinite dimension stochastic process. Just as a Gaussian distribution is a distribution over a random variable, completely specified by its mean and covariance, a GP is a distribution over functions, completely specified by its mean function m and covariance function k:

$$f(x) \sim GP(m(x); k(x, x_0))$$

It is often useful to intuitively think of a GP as analogous to a function, but instead of returning a single numeric value $f(\bar{x})$ for an arbitrary $\bar{x}$, it returns the mean and variance of a normal distribution over the possible values of $f(\bar{x})$.

For the covariance function k, a very popular choice is the squared exponential function:

$$k(x_i, x_j) = exp\left(-\frac{1}{2}x_i - x_j^2\right)$$

This function approaches 1 as values get close together and 0 as they get further apart. Two points that are close together can be expected to have a very large mutual influence, whereas

distant points have almost none. This is a necessary condition for convergence under the assumptions of [32].

The function values are drawn according to a multivariate normal distribution $N(0, K)$ where the kernel matrix K is given by:

$$K = \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_t) \\ \vdots & \ddots & \vdots \\ k(x_t, x_1) & \dots & k(x_t, x_t) \end{bmatrix}$$

Of course, the diagonal values of this matrix are 1 (each point is perfectly correlated with itself), which is only possible in a noise-free environment.

The GP-based surrogate model is fitted according to data obtained in the previous iterations of the SMBO process. Let consider to be at iteration t, that means t evaluations of the objective function have been already performed and stored in the dataset $D_{1:t} = (x_i, f(x_i))$, BO will suggest what is the next x_{t+1} to evaluate. Then, by the properties of Gaussian processes, $\{f_1, \dots f_t\}$ and f_{t+1} are jointly Gaussian, where, for simplicity, f_i denotes $f(x_i)$:

$$\begin{bmatrix} \{f_1, \dots, f_t\} \\ f_{t+1} \end{bmatrix} \sim N\left(0, \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k}^T & k(x_{t+1}, x_{t+1}) \end{bmatrix}\right)$$

where

$$\mathbf{k} = [k(x_{t+1}, x_1)\, k(x_{t+1}, x_2)\dots k(x_{t+1}, x_t)]$$

It is then easy to derive an expression for the predictive distribution:

$$P(f_{t+1}|D_{1:t}, x_{t+1}) = N\left(\mu_t(x_{t+1}), \sigma_t^2(x_{t+1})\right)$$

where

$$\mu_t(x_{t+1}) = \mathbf{k}^T \mathbf{K}^{-1}\{f_1, \dots, f_t\}$$

$$\sigma_t^2(x_{t+1}) = k(x_{t+1}, x_{t+1}) - \mathbf{k}^T\mathbf{K}^{-1}\mathbf{k}$$

A major problem with GP, and more generally with BO, is that it scales poorly with dimensionality of the problem (i.e. number of decision variables); furthermore, the computational complexity for fitting the GP, at each iteration of the SMBO process, is $O(n^3)$ where n is the number of evaluations performed so far, so time for proposition, as well as overall wall-clock time, increases over the iterations. To reduce the impact of dimensionality some proposals have been recently suggested [33, 34].

When the number of decision variables is high—greater than 20—as well as some of them are discrete and/or conditional, other surrogate models are more promising, such as Random Forest (RF) [35, 36]. RF is an ensemble learning method, based on decision trees, for both classification and regression problems. According to the originally proposed implementation, RF aims at generating a multitude of decision trees, at training time, and providing as output the mode of the classes (classification) the mean/median prediction (regression) of the individual trees. Decision trees are ideal candidates for ensemble methods since they usually have low bias and high variance, making them very likely to benefit from the averaging process. RF mostly differs from other typical ensemble methods (e.g. voting systems) in the way they introduce random perturbations into the training phase. The basic idea is to combines bagging, to sample examples from the original dataset, and random selection of features, in order to train every tree of the forest. Injecting randomness simultaneously with both strategies yields one the most effective off-the-shelf methods in machine learning,

working surprisingly well for almost any kind of problems, allowing for generating a collection of decision trees with controlled variance. Let denote with S the size of the forest (i.e. the number of decision trees in the forest) and $T_i(x)$ the output provided by the i-th decision tree for the input x, the variance of the RF-based estimator, for the regression case, can be easily computed as follows:

$$Var\left(\frac{1}{S}\sum_{i=1}^{S}T_i(x)\right) = Cov\left(\frac{1}{S}\sum_{i=1}^{S}T_i(x), \frac{1}{S}\sum_{j=1}^{S}T_j(x)\right) = \frac{1}{S^2}\sum_{i=1}^{S}\sum_{j=1}^{S}Cov\left(T_i(x), T_j(x)\right)$$

$$= \frac{1}{S^2}\sum_{i=1}^{S}\left(\sum_{j\neq i}^{S}Cov\left(T_i(x), T_j(x)\right) + Var\left(T_i(x)\right)\right) \leq \frac{1}{S^2}\sum_{i=1}^{S}\left((S-1)\rho\sigma^2 + \sigma^2\right)$$

$$= \frac{S(S-1)\rho\sigma^2 + S\sigma^2}{S^2} = \frac{S(S-1)\rho\sigma^2}{S} + \frac{\sigma^2}{S} = \rho\sigma^2 - \frac{\rho\sigma^2}{S} + \frac{\sigma^2}{S}$$

$$= \rho\sigma^2 + \sigma^2\frac{1-\rho}{S}$$

where σ^2 is the variance computed over all the $T_i(x)$ and $\rho\sigma^2 = \max_{i,j} Cov\left(T_i(x), T_j(x)\right)$.

Thus, variance of the RF estimator decreases with σ^2 and ρ decreasing (i.e. if the number m of selected features decreases) and with the size S the forest increasing.

Variance is crucial to replace GPs with RF in the SMBO framework; the possibility to compute the variance of the RF allows for keep the overall sequential optimization process as is, since fitting the surrogate model in the SMBO process is a regression task. Therefore, RF can be easily adopted by replacing machine learning concepts with those related to SMBO (e.g., features with decision variables and dataset with set of evaluations performed so far). Following a basic description of the RF learning algorithm.

Random Forest learning algorithm

Given:

- S the size of the forest (i.e. number of decision trees in the forest)

- N the number of examples in the dataset (i.e. evaluations of the objective functions)

- M the number of features (i.e. decision variables) representing every example

- m the number of features (i.e. decision variables) to be randomly selected from the M available for each leaf node of the tree to be split

- n_{min} the minimum node size in the tree

1: for $i = 1$ to S

2:        sample, with replacement, a subset of N instances from the dataset

3:        for every terminal node (leave) of the i-th decision tree having size lower than n_{min}

4:                select m features (i.e. decision variables) at random from the M available

5:                pick the best feature (i.e. decision variable) and split among the possible m

6:                split the node in two children nodes (new leaves of the decision tree)

7:        end for

8: end for

With respect to PSO, it is important to highlight that using GP or RF as surrogate model may significantly affect effectiveness and efficiency of the overall BO framework proposed. More specifically, a surrogate model based on RF should be, at least ideally, well suited for WDNs with only ON/OFF pumps or mixed (both ON/OFF and VSP) pumps. Furthermore, a RF-based surrogate offers higher scalability with respect to the number of evaluations of the objective functions, leading to a wall-clock time lower than that required by a GP-based surrogate, given the same number of function evaluations. According to these considerations, using RF as probabilistic model of the BO framework should be the most profitable choice.

3.2 Acquisition functions

The acquisition function is the mechanism to implement the trade-off between exploration and exploitation in the BO. The basic idea is to improve over the best solution observed so far ("best seen"), even if an improvement is not guaranteed from one iteration to the next one, due to the exploration. In particular, any acquisition function aims to guide the search of the optimum towards points with potential high values of objective function either because the prediction of $f(x)$, based on the surrogate model, is high or the uncertainty is high (or both). While *exploiting* means to consider the area providing more chance to improve the current solution (with respect to the current surrogate model), *exploring* means to move towards less explored regions of the search space.

Probability of Improvement (PI) was the first acquisition function proposed in the literature [37], where the parameter $\xi \geq 0$ is used to balance between exploration and exploitation ($\xi = 0$ means pure exploitation).

$$PI(x) = P\left(f(x) \geq f\left(x^+\right) + \xi\right) = \phi\left(\frac{\mu(x) - f\left(x^+\right) - \xi}{\sigma(x)}\right)$$

However, one of the drawback of PI is that it is biased towards exploitation.

Another popular acquisition function is the Expected Improvement (EI) [38] which measures the expectation of the improvement on $f(x)$ with respect to the predictive distribution of the surrogate model. Even in this case parameter ξ is used to handle the trade-off between exploitation and exploration. When exploring, points associated to high uncertainty of the surrogate model are chosen, while when exploiting, points associated to high value of the mean of the surrogate model are selected.

$$EI(x) = \begin{cases} \left(\mu(x) - f\left(x^+\right) - \xi\right)\phi(Z) + \sigma(x)\phi(Z) \; if \, \sigma(x) > 0 \\ 0 \; if \, \sigma(x) = 0 \end{cases}$$

where ϕ and Φ are the probability and the cumulative distribution functions, respectively, and

$$Z = \begin{cases} \frac{\mu(x) - f(x^+) - \xi}{\sigma(x)} & if \quad \sigma(x) > 0 \\ 0 & if \quad \sigma(x) = 0 \end{cases}$$

Finally, another acquisition function selects the next point to evaluate according to the Confidence Bound concept [39, 40]: Lower Confidence Bound—LCB—and Upper Confidence Bound—UCB—in the minimization and maximization case, respectively. For the case of minimization, LCB is given by:

$$LCB(x) = \mu(x) - k\sigma(x)$$

where $k \geq 0$ is the parameter to manage the exploration/exploitation trade-off ($k = 0$ means pure exploitation).

To solve the black-box optimization the optimization of the acquisition function is required but, usually, this is cheaper and easier than to optimize the objective function. Methods used are basically deterministic, such as the derivative free DIRECT [41] or multi-start BFGS (Broyden–Fletcher–Goldfarb–Shanno) [42]. Another relevant approach is the so called "focus search", proposed in the mlrMBO software to handle numeric, categorical, mixed and hierarchical/conditional decision variable spaces, and therefore well suited for a RF based surrogate model.

3.3 Software environment

In order to implement the BO framework proposed for solving the pump scheduling optimization problem, the R package named "mlrMBO" has been adopted: it is a flexible and comprehensive toolbox for model-based optimization (MBO). This toolbox has been designed for both single- and multi-objective optimization with mixed continuous, categorical and conditional decision variables, and it has been implemented in a modular fashion, such that single components can be easily replaced or adapted by the user for specific use cases, e.g., any regression learner from the R package "mlr" for machine learning can be used (more than 60 machine learning regression algorithms), and infill criteria and infill optimizers are easily exchangeable.

The mlrMBO toolbox implements the SMBO framework giving the possibility to customize each step of the overall sequential optimization process. An initial set of evaluations has to be performed in order to create the first instance of the surrogate model, which will be then updated over the iterations of the SMBO process. The user can manually specify the initial set of points to be evaluated or generate them either completely at random, coarse grid designs or by using space-filling Latin Hypercube Sampling (LHS) [21].

With respect to the surrogate model, two default choices are available: GP for continuous unconditional decision variables and RF for the other cases. In any case, any other regression algorithm (named "learner" in mlrMBO) can be used to fit the surrogate model. Three different acquisition functions—also named infill criteria in mlrMBO—are available: EI, Augmented EI and LCB/UCB. The toolbox is anyway extendible, so user can implement its own acquisition function.

Finally, different termination criteria can be defined depending on the specific optimization problem: maximum number of function evaluations is reached, the optimum is found (in the case that optimum value of the objective function is known) or "budget" is exhausted (e.g., overall wall-clock time or costs for cloud computing).

4 Experimental results

This section presents the results obtained both on a set of test functions and two PSO case studies: a benchmark well known in the PSO literature and a real-life WDN in Milan, Italy. All the experiments, with exception of SigOpt (which is a cloud-based platform), were carried out on a machine with: intel i7 (3.4gHz) processor with 4 cores and 16Gb of RAM.

4.1 Experiments on test functions

As preliminary evaluation, 12 different test functions has been considered to compare different SMBO alternatives. Four are test functions well known in global optimization literature: Branin (2D), Hartman-3D, Hartman-6D and Schwefel-20D; the other 8 test func-

Table 1 Parameters configuration for the generation of the GKLS test functions

GKLS test function	Number of local minima	Radius of local minima	Distance from the paraboloid vertex to the global minimum	Global minimum value
GKLS3D, GKLS3D-ND, GKLS6D, GKLS6D-ND	5	0.10	0.20	− 1.50
GKLS2D, GKLS2D-ND, GKLS20D, GJLS20D-ND	20	0.10	0.20	− 1.50
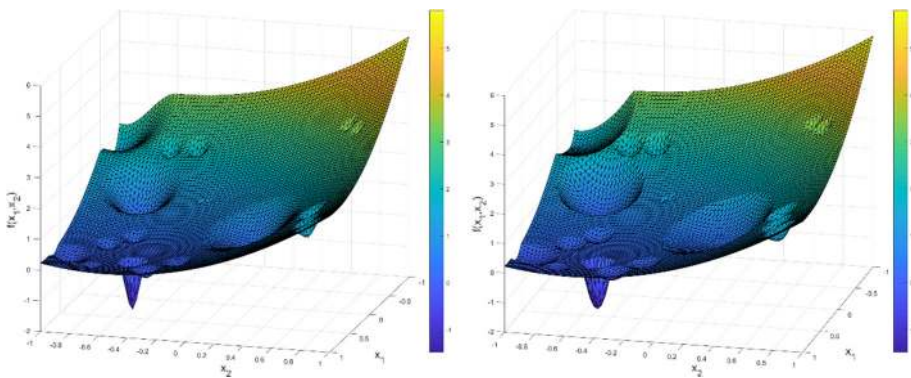
Fig. 1 Two-dimensional test functions generated through the GKLS software: diiferentiable (on the left) and not-differentiable (on the right)

tions have been obtained through the GKLS software [43]: GKLS2D, GKLS3D, GKLS6D and GKLS20D are differentiable functions, GKLS2D-ND, GKLS3D-ND, GKLS6D-ND and GKLS20D-ND are not differentiable functions. The generation mechanism is based on the perturbation of a paraboloid, performed by setting up 4 different parameters. These parameters, and the values set up for the different test cases, are reported in the following Table 1. For all the GKLS functions the decision variables are bounded in the unitary hypercube.

For illustrative purposes, the differentiable and not differentiable functions generated through GKLS, in the 2D case, are reported in the following Fig. 1.

Experiments on test functions have been performed with the aim to have a preliminary understanding about the benefits provided by BO and its different alternatives, in particular relatively to the adoption of GP or RF for the surrogate model and the selection of EI or CB as acquisition function. As they are test functions, the optimum is known—both x^* and $f(x^*)$—thus the difference between the optimum $f(x^*)$ and the "best seen" $f(x^+)$, after t evaluations of the objective function, is computed. More specifically, a budget of 1500 evaluations was fixed for each experiment (i.e. $t = 1500$).

The following tables report, separately, the difference between $f(x^*)$ and $f(x^+)$ and the function evaluation (also indicated with "iteration)" at which best seen occurred for the first time. Latin Hypercube Sampling (LHS) is used to initialize the design, with only 5 initial

Table 2 Difference between known optimum and optimal solution (i.e. best seen)

	GP-LCB	GP-EI	RF-LCB	RF-EI	SigOpt	ALAMO+BARON
Branin-2D	**0.0000**	**0.0000**	**0.0000**	0.0006	0.0001	0.0231
Hartmann-3D	**0.0002**	**0.0002**	0.7731	0.0004	**0.0002**	0.0528
Hartmann-6D	0.2800	0.2803	0.3426	0.2851	**0.0026**	3.3224
Schwefel-20D	5418.6950	4766.8740	**3788.6270**	5046.0320	5330.7000	8372.7920
GKLS2D	–	**0.0000**	1.3162	**0.0000**	0.0001	1.0740
GKLS2D-ND	–	–	**0.0000**	**0.0000**	0.0003	0.6180
GKLS3D	–	0.4906	**0.0006**	0.4909	0.4906	1.5140
GKLS3D-ND	–	**0.4905**	**0.4905**	0.4907	0.4906	1.4460
GKLS6D	1.5000	1.5001	**0.0263**	1.5183	1.5001	1.9990
GKLS6D-ND	**0.0215**	1.1774	0.0270	1.5117	0.6118	1.9990
GKLS20D	1.8163	**1.7083**	2.7499	2.9320	1.8491	6.5759
GKLS20D-ND	**1.7852**	1.8415	2.4220	3.4086	1.8042	6.5759

Table 3 Function evaluation corresponding tot he first occurrence of the best seen

	GP-LCB	GP-EI	RF-LCB	RF-EI	SigOpt	ALAMO+BARON
Branin-2D	44	153	624	1093	704	868
Hartmann-3D	28	42	860	181	62	500
Hartmann-6D	1218	917	1211	657	447	500
Schwefel-20D	1316	860	1024	1060	653	500
GKLS2D	–	142	1445	1145	174	516
GKLS2D-ND	–	–	1065	1431	305	564
GKLS3D	–	417	1038	1201	185	500
GKLS3D-ND	–	503	1101	1287	234	500
GKLS6D	777	1411	968	1295	974	500
GKLS6D-ND	1480	57	1352	455	724	500
GKLS20D	1498	1349	1434	1015	1404	500
GKLS20D-ND	815	1013	823	1407	1475	500

evaluations, 7 in the cases of 6D test functions and 21 in the case of 20D test functions, since LHS requires a number of evaluations greater than the dimensions.

With respect to the commercial "optimization-as-a-service" solution there is no possibility to set anything related to the optimization process performed by SigOpt. The experiments with ALAMO in combination with BARON use all the possible base functions for the generation of the surrogate model (i.e., constant, linear, exponential, logarithmic, sine, cosine) (Tables 2, 3).

Although it is not possible to identify a "winner" among the proposed approaches, some relevant insights can be retrieved:

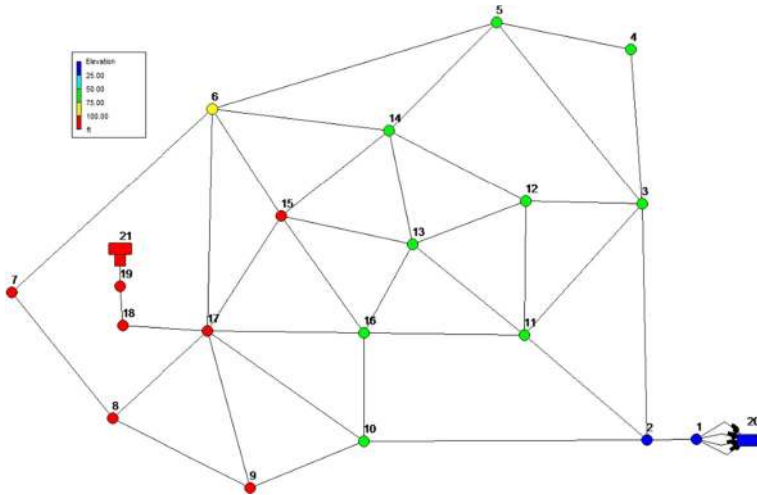- All the approaches are effective on low-dimensions, in particular on the test function Branin2D;

Fig. 2 The Anytown WDN

- BO with RF as surrogate model and LCB as acquisition function is the only approach providing the best solution on 50% of all the test functions considered;
- BO with GP as surrogate model, in some cases, is not able to perform all the function evaluations, resulting in a premature interruption of the SMBO process. This issue arises more frequently when the acquisition function is LCB;
- ALAMO with BARON does not use all the budget (1500 function evaluations) because error between surrogate model and preliminary function evaluations is very low, so the surrogate is considered accurate. However, when optimization is performed on the generated surrogate model, the value of the optimal solution is quite different from the known optimum of the related test function.

4.2 Experiments on the Anytown WDN

After the preliminary evaluation on the test functions, we have applied the proposed BO approaches on the PSO problem. As first case study, a benchmark WDN named Anytown, originally proposed in [11], has been considered. This WDN consists of 37 pipes, 19 nodes, 1 tank (having elevation and diameter equal to 65.53 and 12.20 m, respectively) and 1 reservoir, with 4 pumps installed at the source to supply water. The Anytown WDN, modelled through EPANET, is reported in the following Fig. 2.

A recent work has addressed the PSO on the Anytown network through Harmony Search [44], a meta-heuristics search method. It is important to underline that we do not compare directly with Harmony Search: it has been recently proposed to solve the PSO problem on Anytown as the plethora of meta-heuristics—extensively Genetic Algorithms—proposed in the last decade. Our aim is to propose a more mathematically sound approach to solve the PSO problem by exploiting black-box optimization and making it an effective and efficient tool for practical/operational problems, such as PSO in WDNs. However, the more recent, detailed and completely replicable PSO setting on Anytown is—at our knowledge—proper reported in [44].

Table 4 Results on the case study related to 4 ON/OFF pumps

Acquisition Function	Learner	Energy cost ($)	Best seen iteration	Overall clock time
LCB	GP	1281.01	360	22,785.44
	RF	**1054.98**	183	2698.45
EI	GP	1219.11	292	25,096.23
	RF	1117.90	149	3505.28

Table 5 Results on the case study related to 4 VSPs

Acquisition function	Learner	Energy cost ($)	Best seen iteration	Overall clock time
LCB	GP	653.55	356	21,256.48
	RF	687.43	43	3412.79
EI	GP	**609.17**	290	23,594.36
	RF	719.36	286	4342.30
	ALAMO+BARON	1087.54	NA	4005.54

We have used a penalty on the objective function in the case warnings occur during the corresponding simulation run (i.e. some hydraulic constraint is not satisfied). However, contrary to [44], who used as penalty a default cost equals to 9999, we have decided to adopt, as penalty value, the cost obtained by using the "naïve" schedule consisting in having all the 4 pumps switched ON for all the hours of the day. We have adopted the same setting reported in [44]: minimum and maximum tank heads are set to 67.67 and 76.20 m, respectively. Hourly demand factors ranged from 0.4 to 1.2 and base demand is doubled from the Anytown test case to allow longer pump operation and tank storage, defining a total inflow between 161.51 and 484.53 lps. Finally, an electricity tariff with a price of 0.0244 $/kW h between 0:00–7:00 and 0.1194 $/kW h between 8:00 and 24:00 was considered for simulations.

The objective function $f(x)$ is given in (1) as the energy cost associated to the pump schedule, and x_t represents the state of the pumps at a specific time step (i.e. hour). If an ON/OFF pump is considered, the domain of the decision variable is {0, 1}, while it is a continuous variable in [0, 1] in the case of VSP. As previously mentioned, $f(x)$ is not available in closed form but just as a black-box whose evaluation requires the run an EPANET simulation. Furthermore, we assume that each evaluation produces an unbiased (not noisy) point-wise estimation of $f(x)$ because we consider that both WDN structure and demand do not change from one simulation to another.

Usually PSO on this WDN has been targeted considering ON/OFF pumps in the system, leading to $4 \times 24 = 96$ variables. Thus, the total number of possible schedules is $2^{4 \times 24} = 2^{96}$, with an hourly step and a 24 h horizon; consequently, the total search space is 7.92×10^{28}.

The number of overall evaluations has been fixed at 800, where 400 are used for the initial design (i.e. generation of the first surrogate model) and the other 400 are related to the iterations of the SMBO process. For the initial 400 evaluations, the LHS approach has been used.

The following Tables 4 and 5 report, separately, the results obtained when ON/OFF pumps and VSPs are considered. As natural, the adoption of VSPs allows for lower energy costs (due

Table 6 Results on the case study related to 3 ON/OFF pumps

| Acquisition Function	Learner	Energy cost ($)	Best seen iteration	Overall clock time
LCB	GP	900.54	142	16,091.20
	RF	870.84	3	1887.06
EI	GP	**842.25**	390	15,879.03
	RF	859.23	306	2828.00

to the nature of the pumps), even lower than the optimal value found in [44]. Furthermore, when VSPs are considered, using a GP-based surrogate model results to be a better choice, than using RF, in minimizing the energy costs. On the contrary, the RF is better than GP in the case of ON/OFF pumps. This information basically confirms that the selection of a strategy to infer the surrogate model should depend on the type of decision variables. The VSPs case has been also optimized through ALAMO in combination with BARON: the energy cost associated to the optimal solution identified on its surrogate model is higher than the BO approaches.

The iteration associated to the occurrence of the best seen and the overall wall clock time are also reported. It is easy to note that, in the case a GP-based surrogate is used, the overall clock time is one order of magnitude higher. This basically proves the most relevant drawback of GPs: they scale poorly with the number of decision variables, since computational complexity is $O(n^3)$ with n the number of function evaluations.

Taking into account the ON/OFF pumps case, there is a substantial difference between the optimal schedules we have identified through BO and the one found via Harmony Search in [44]: our optimal schedules use all the 4 pumps while the optimal one obtained in [44] uses just 3 pumps. A possible motivation is that the overall number of function evaluations are not sufficient (800): although the number of function evaluations performed in [44] is not declared, a comparison with 2000 generations of a Genetic Algorithm based approach is reported, leading to suppose that a similar number of evaluations was used also for Harmony Search. We have therefore decided to perform a further experiment by forcing the same pump to be shut down; thus, the decision variables are in this case related to 3 pumps, leading to $3 \times 24 = 72$ decision variables and $2^{3 \times 24} = 2^{72}$ possible schedules. The search space is still huge, and BO can be still profitably applied. Again, we consider both the ON/OFF pumps and VSP cases; moreover, the reduction from 96 to 72 variables allows us to make a comparison, in the VSP case, between the BO framework implemented through mlrMBO and the commercial "optimization-as-a-service" platform named SigOpt. The following Table 6 reports the results obtained in the case of 3 ON/OFF pumps.

The pump schedules obtained have a lower energy cost with respect to those obtained from the previous BO process (performed on 4 ON/OFF pumps) leading to results more similar to [44]. In particular, the best result is obtained when a GP surrogate model is used and EI is chosen as acquisition function. However, in terms of energy cost, GP and RF differ only by 2–3%, while in terms of number of iterations needed to identify the best seen, and more important overall wall-clock time, RF is significantly better. The following Fig. 3 shows the main information related to the optimal pump schedule: the pumps status and the level of tank—even with respect to the level reported in [44]—for every hour of the day.

Finally, the following Table 7 summarizes results obtained in the case of VSPs. For this experiment we have also adopted SigOpt as further comparison. Again, considering 3 VSPs
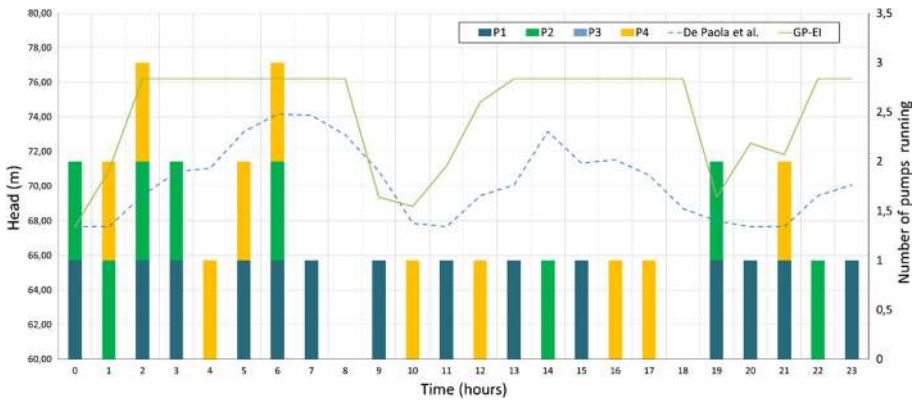
Fig. 3 Pumps schedule obtained through a BO process with a GP surrogate model and EI as acquisition function. Status of each pump for each hour is provided (pump "P3" is always OFF); level of the tank for every hour in also depicted, in comparison with that obtained by [44]

Table 7 Results on the case study related to 3 VSPs

| Acquisition Function	Learner	Energy cost	Best seen iteration	Overall clock time
LCB	GP	659.27	301	17,588.74
	RF	**520.42**	240	3118.83
EI	GP	549.77	47	16,028.29
	RF	667.48	385	3828.93
	SigOpt	540.27	705	4223.78
	ALAMO+BARON	875.44	NA	3997.41

instead of 3 ON/OFF pumps gives the possibility to further reduce energy cost. In this specific case, SigOpt is near to the optimal value provided by RF-based BO with LCB as acquisition function. The main advantage provided by SigOpt is the high scalability which allows for increasing the number of evaluations and therefore improving the optimal value.

Finally, the following Fig. 4 shows information related to the optimal pump schedule identified: the speed of the pumps (from 0 to 1) and the tank level—along with that reported in [44]—for every hour of the day.

4.3 Experiments on a real-life large-scale WDN

After the experiments on the benchmark WDN, we have addressed a real-life large-scale WDN in Milan, Italy. The water distribution service in the urban area of Milan is managed by Metropolitana Milanese (MM) through a cyber-physical system consisting of: a transmission network (550 wells and pipelines bringing water to 29 storage and treatment facilities located inside the city) and a distribution network (26 different pump stations spread over the city—95 pumps overall—which take water from the storage and treatment facilities and supply it to the final customers, about 1.350.000 habitants). Financial burden for MM is approximately 16.000.000 € for energy costs, with 45% of them due to pumping operations in the distribution network [45]. During the project ICeWater—co-founded by the European Union—a Pressure
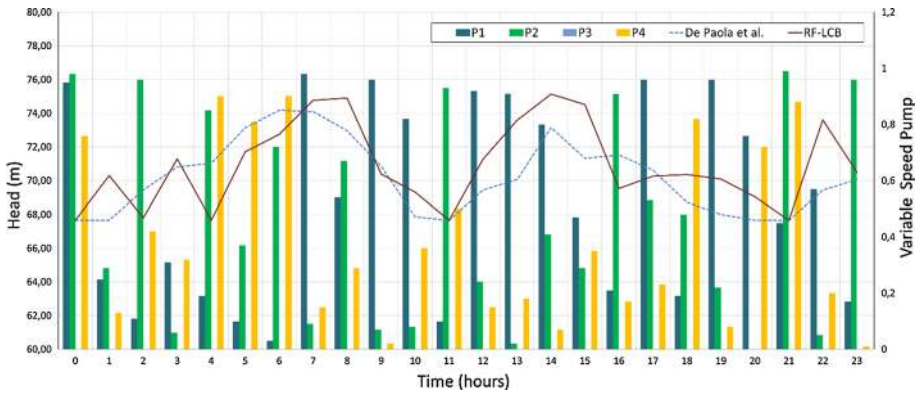
Fig. 4 Pumps schedule obtained through a BO process with a RF surrogate model and LCB as acquisition function. Status of each pump for each hour is provided (pump "P3" is always OFF); level of the tank for every hour in also depicted, in comparison with that obtained by [44]

Fig. 5 EPANET model of the Pressure Management Zone (PMZ) "Abbiategrasso", in Milan, Italy

Management Zone (PMZ), namely "Abbiategrasso", was identified in the South of the city and isolated from the rest of the network.

The EPANET model of the Abbiategrasso PMZ, reported in the following Figs. 5, 6 was presented in [45]: it consists of 7855 nodes, 6073 pipes, 1961 valves, 1 reservoir and 4 pumps. Two of them, the oldest ones, are ON/OFF, while the other two, the newest ones installed in 2014, are VSPs.
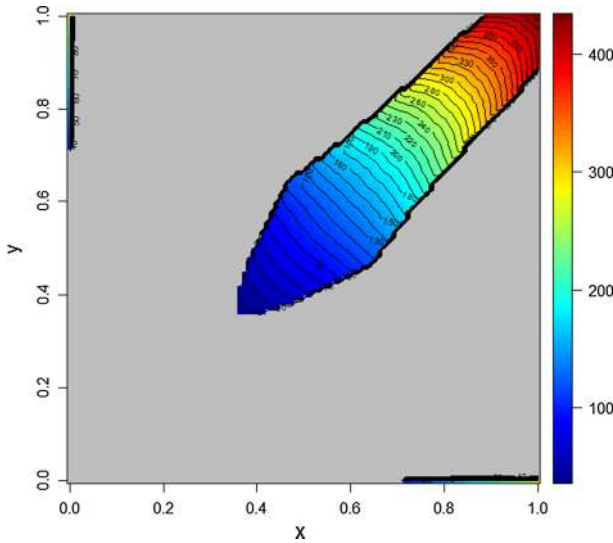
Fig. 6 Objective function and feasible region for a subset of decision variables of the real world large scale case study (WDN in Milan): *x* and *y* are the speed of first and second pump, respectively, for the first hour of the day

As previously reported in [45], preliminary results showed that two pumps are sufficient to match supply and demand within the PMZ. This consideration allows for limiting the number of decision variables (only two pumps are, at most, active simultaneously) and quantify which is the economic return of MM in using VSPs instead of ON/OFF pumps.

The optimization setting is the same defined for the benchmark WDN. A relevant difference is related to the electricity price: the Time-Of-Use (TOU) tariff in place for the real-life case study is:

- (low-price) 0.0626 €/kW h between 00:00–7:00 and between 23:00–24:00
- (mid-price) 0.0786 €/kW h between 08:00 and 19:00
- (high-price) 0.0806 €/kW h between 07:00–08:00 and between 19:00–23:00

All the available data for the study, including the reported energy tariff, are related to the year 2014.

The other relevant difference, with respect to Anytown, is the number of decision variables (2 instead of 4), and, consequently, the size of the search space. For the ON/OFF pumps case (i.e., binary decision variables) the overall number of possible pump schedules is $2^{2 \times 24} = 2^{48}$, that is about 2.81×10^{14}, in any case too large to perform an exhaustive search. The number of overall objective function evaluations (i.e. EPANET simulation runs) has been fixed at 800, with 400 for the initial design (i.e. generation of the first surrogate model) and the remaining 400 for the iterations of the SMBO process. For the initial 400 evaluations, the LHS approach has been used. The following tables summarize the obtained results for the ON/OFF pumps and the VSPs setting, respectively (Tables 8, 9).

With respect to the ON/OFF pumps case, only BO with RF as surrogate model was able to identify a feasible solution. Moreover, the value of the best seen is independent on the acquisition function.

With respect to the VSPs case, some important considerations are reported:

Table 8 Optimal energy costs identified for the real-life large-scale WDN, the Abbiategrasso PMZ in Milan, with two ON/OFF pumps

Acquisition function	Learner	Energy cost	Best seen iteration	Overall clock time
LCB	GP	427.37*	596	3986.43
	RF	**271.47**	524	4018.76
EI	GP	427.37*	168	3990.26
	RF	**271.47**	473	4049.37

Values marked with the symbol * are penalties: this means that a feasible solution has not been identified at the end of the optimization process

Table 9 Optimal energy costs identified for the real-life large-scale WDN, the Abbiategrasso PMZ in Milan, with two VSPs

Acquisition Function	Learner	Energy cost	Best seen iteration	Overall clock time
LCB	GP	**100.44**	1	82,563.94
	RF	192.46	703	6945.39
EI	GP	195.60	76	50,228.34
	RF	160.08	557	7390.14
	SigOpt	200.30	590	7998.23
	ALAMO + BARON	427.37*	500	3045.56

Values marked with the symbol * are penalties: this means that a feasible solution has not been identified at the end of the optimization process

- Energy costs are naturally lower than those obtained on the ON/OFF pumps case, thanks to the nature of the pumps—and, therefore, nature of the decision variables;
- ALAMO in combination with BARON is not able to identify a feasible solution;
- Although BO with GP as surrogate model and LCB as acquisition function provided the lowest energy cost, using GP is too computational expensive (overall wall clock time is about 10 times that required by RF). This makes BO with GP unsuitable from an operational point of view, since too much time is required to generate a suggestion and turn decision into action timely.

Finally, the real-life large-scale case study allowed to highlight a further and important difficulty in solving the PSO problem. Solutions in the search space are not all feasible; however, the feasible region is not known a priori. As for the evaluation of the objective function, also the feasibility check is performed through EPANET: more specifically, EPANET returns a numeric value (i.e., energy cost) when the considered point in the search space (i.e., pump schedule) is feasible, otherwise it returns a specific error code.

As previously mentioned, we have assigned a penalty to unfeasible points, with the aim to move towards feasible points/regions. This approach is quite naïve because the penalty does not depend on the entity of violation, but it is the approach usually adopted in the PSO literature. This leads to almost "flat" objective functions, especially when the unknown feasible region results extremely narrow with respect to the overall search space. The two WDN use cases considered, but more specifically the real-life WDN, seem to be characterized by this kind of situation: in the following figures it is possible to see the extremely narrow

feasible region for the real-life WDN, with x and y speed of the first and the second VSP, respectively. To have a 2D representation, the first hour of the day is considered, only.

Since BO gives, along the sequential optimization process, some chance to explore unseen regions of the search space, there is some opportunity to get to the feasible region. On the contrary, ALAMO in combination with BARON just tries to generate a surrogate model as accurate as possible. Thus, it is deceived by the false flatness of the objective function induced by the penalty and the narrow extent of the feasible region. A possibility to exploit the potential of ALAMO in combination with BARON, in a future work, could consists in using it instead of GP or RF within the SMBO process. Using ALAMO in combination with BARON, within the BO framework, with the aim to create/update the surrogate model requires to understand how to model uncertainty of the surrogate, similarly to the GP and RF approaches.

5 Conclusions

Energy saving in WDN is one of the most challenging issues of smart urban water management and PSO is critical to reduce energy costs while guaranteeing a satisfactory water supply service. Simulation–optimization is usually adopted to solve the PSO, where the optimization is mostly performed via meta-heuristics, such as Genetic Algorithms or Harmony Search, and simulation is widely performed via EPANET.

According to the emerging interest on Sequential Model Based Optimization (SMBO), in particular Bayesian Optimization (BO), to solve black-box and expensive optimization processes, we have proposed the SMBO as optimization engine of the simulation–optimization PSO problem and compared different alternative approaches. A set of test functions has been initially used to make preliminary considerations about the different alternative schemes proposed, then two PSO case studies have been addressed: a widely studied benchmark WDN, namely Anytown, and a real-life large-scale WDN in Milan, Italy. Considering different alternative was motivated by the type of decision variables, which are discrete in the ON/OFF pumps case and continuous in the VSPs case.

Although an "always winning" approach cannot be identified, relevant conclusions can be summarized. BO with RF as surrogate model and LCB acquisition function was the only approach providing the best solution in 50% of the test functions considered.

With respect to PSO on the two case studies, SMBO proved to be effective, even using a limited number of function evaluations (400 for initial design and 400 for the BO process) with respect to the overall search space. However, GP-based and RF-based surrogate models alternate in providing the best result. Although this could appear inconclusive, the most relevant conclusion is related to the time needed to obtain a solution: GP-based BO requires from 5 to 10 times the wall clock time needed to RF-based BO, independently on the acquisition function considered. Thus, time-to-decision could be the key for the final choice on the SMBO configuration to use. From an operational point of view, wall clock time is acceptable for the benchmark WDN only: a suitable pump schedule can be identified in few hours (from 1 to 5, according to the set-up of the SMBO framework) which is a reasonable time, in particular when some reliable urban water demand forecasting solution is available. Wall clock time for RF-based SMBO resulted 5 times lower than GP-based one's (1 vs. 5 h), meaning that, in the case the time required by the GP-based SMBO is anyway acceptable, the number of evaluations for the RF-based SMBO could be increased to raise the chance to find a better pump schedule while keeping wall-clock time acceptable.

A different conclusion arises with respect to the results on the real-life WDN: while RF-based BO requires about 2 h for performing 800 iterations of the SMBO process, in the VSPs case, GP-based BO may require about 22 h, making it unsuitable for optimizing pump schedule in a real-life setting. Therefore, RF-based BO is the most effective and efficient choice.

Finally, the real-life WDN allowed to highlight that PSO is characterized by a feasible region but it is unknown a priori. Like the value of the objective function, feasibility of a solution (i.e., a pump schedule) can only be verified by querying the black-box (i.e., EPANET). A naïve strategy based on assigning a constant penalty value to unfeasible points could be inefficient in BO and completely ineffective in the generation and consecutive optimization of an accurate surrogate model (i.e., ALAMO in combination with BARON). Future works will address how to include an estimation of the feasibility region, as well as a more sophisticated and effective quantification of penalties to unfeasible points.

References

1. Fantozzi, M., Popescu, I., Farnham, T., Archetti, F., Mogre, P., Tsouchnika, E.: ICT for efficient water resources management: the ICeWater energy management and control approach. Procedia Eng. **70**, 633–640 (2014)
2. Mala-Jetmarova, H., Sultanova, N., Savic, D.: Lost in optimisation of water distribution systems? A literature review of system operation. Environ. Model. Softw. **93**, 209–254 (2017)
3. Candelieri, A., Soldi, D., Archetti, F.: Cost-effective sensors placement and leak localization—The Neptun pilot of the ICeWater project. J. Water Supply Res. Technol. AQUA **64**(5), 567–582 (2015)
4. Naoum-Sawaya, J., Ghaddar, B., Arandia, E., Eck, B.: Simulation-optimization approaches for water pump scheduling and pipe replacement problems. Eur. J. Oper. Res. **246**(1), 293–306 (2015)
5. Bagirov, A.M., Barton, A.F., Mala-Jetmarova, H., Al Nuaimat, A., Ahmed, S.T., Sultanova, N., Yearwood, J.: An algorithm for minimization of pumping costs in water distribution systems using a novel approach to pump scheduling. Math. Comput. Model. **57**(3–4), 873–886 (2013)
6. Candelieri, A.: Clustering and support vector regression for water demand forecasting and anomaly detection. Water **9**(3), 224 (2017)
7. Candelieri, A., Soldi, D., Archetti, F.: Short-term forecasting of hourly water consumption by using automatic metering readers data. Procedia Eng. **119**(1), 844–853 (2015)
8. Candelieri, A., Archetti, F.: Identifying typical urban water demand patterns for a reliable short-term forecasting—The icewater project approach. Procedia Eng. **89**, 1004–1012 (2014)
9. D'Ambrosio, C., Lodi, A., Wiese, S., Bragalli, C.: Mathematical programming techniques in water network optimization. Eur. J. Oper. Res. **243**(3), 774–788 (2015)
10. Pasha, M.F.K., Lansey, K.: Optimal pump scheduling by linear programming. In: World Environmental and Water Resources Congress, pp. 395–404 (2009)
11. Price, E., Ostfeld, A.: Iterative linearization scheme for convex nonlinear equations: application to optimal operation of water distribution systems. J. Water Resour. Plan. Manag. **139**(3), 299–312 (2013)
12. McCormick, G., Powell, R.S.: Derivation of near-optimal pump schedules for water distribution by simulated annealing. J. Oper. Res. Soc. **55**, 728–736 (2004)
13. Wu, W., Dandy, G., Maier, H.: Optimal control of total chlorine and free ammonia levels in a water transmission pipeline using artificial neural networks and genetic algorithms. J. Water Resour. Plan. Manag. **141**(7), 04014085 (2014)
14. Van Zyl, J.E., Savic, D.A., Walters, G.A.: Operational optimization of water distribution systems using a hybrid genetic algorithm. J. Water Resour. Plan. Manag. **130**(2), 160–170 (2004)

15. Rossman, L.A.: EPANET2 User's Manual. U.S. Environmental Protection Agency, Washington, DC (2000)
16. Sergeyev, Y.D., Kvasov, D.E.: Deterministic Global Optimization: An Introduction to the Diagonal Approach. Springer, Berlin (2017)
17. Sergeyev, Y.D., Kvasov, D.E.: Global search based on efficient diagonal partitions and a set of Lipschitz constants. SIAM J. Optim. **16**(3), 910–937 (2006)
18. Sergeyev, Y.D., Kvasov, D.E.: A deterministic global optimization using smooth diagonal auxiliary functions. Commun. Nonlinear Sci. Numer. Simul. **21**(1–3), 99–111 (2015)
19. Barkalov, K., Polovinkin, A., Meyerov, I., Sidorov, S., Zolotykh, N.: SVM Regression parameters optimization using parallel global search algorithm. In: International Conference on Parallel Computing Technologies, pp. 154–166 (2013)
20. Gillard, J.W., Kvasov, D.E.: Lipschitz optimization methods for fitting a sum of damped sinusoids to a series of observations. Stat. Interface **10**, 59–70 (2017)
21. Zabinsky, Z.B.: Stochastic adaptive search for global optimization, 1st edn. Springer, US (2003)
22. Steponavičě, I., Shirazi-Manesh, M., Hyndman, R.J., Smith-Miles, K., Villanova, L.: On sampling methods for costly multi-objective black-box optimization. Springer Optim. Its Appl. **107**, 273–296 (2016)
23. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: a novel bandit-based approach to hyperparameter optimization, pp. 1–48 (2016). arXiv:1603.06560
24. Huang, H., Zabinsky, Z.B.: Adaptive probabilistic branch and bound with confidence intervals for level set approximation. In: Winter Simulation Conference (2013)
25. Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., De Freitas, N.: Taking the human out of the loop: a review of Bayesian optimization. Proc. IEEE **104**(1), 148–175 (2016)
26. Snoek, J., Larochelle, H., Adams, R.P.: Practical bayesian optimization of machine learning algorithms, pp. 1–12 (2012)
27. Archetti, F., Betrò, B.: A probabilistic algorithm for global optimization. Calcolo **16**(3), 335–343 (1979)
28. Hutter, F., Hoos, H.H., Leyton-Brown, K.: Sequential model-based optimization for general algorithm configuration. In: International Conference on Learning and Intelligent Optimization (LION 5), pp. 507–523 (2011)
29. Candelieri, A., Giordani, I., Archetti, F.: Automatic configuration of Kernel-based clustering: an optimization approach. In: International Conference on Learning and Intelligent Optimization (LION). Springer, Cham, pp. 34–49 (2017)
30. Castro Gama, M., Pan, Q., Salman, M.A., Jonoski, A.: Multivariate optimization to decrease total energy consuption in the water supply system of Abbiategrasso (Milan, Italy). Environ. Eng. Manag. J. **14**(9), 2019–2029 (2015)
31. Wu, J., Poloczek, M., Wilson, A.G., Frazier, P.I.: Bayesian optimization with gradients, 1–16 (2017)
32. Mockus, J.: Application of Bayesian approach to numerical methods of global and stochastic optimization. J. Glob. Optim. **4**, 347–365 (1994)
33. Wang, Z., Zoghi, M., Hutter, F., Matheson, D., De Freitas, N.: Bayesian optimization in high dimensions via random embeddings. In: Proceedings of International Jt. Conference on Artificial Intelligence, pp. 1778–1784 (2013)
34. Kandasamy, K., Schneider, J., Pòczos, B.: High dimensional bayesian optimisation and bandits via additive models. Int. Conf. Mach. Learn. **37**, 295–304 (2015)
35. Ho, T.K.: Random decision forests. In: Conference in Document Analysis and Recognition, pp. 278–282 (1995)
36. Breiman and Liaw: U.S. trademark registration number 3185828 (2006)
37. Kushner, H.J.: A new method of locating the maximum point of an arbitrary multi-peak curve in the presence of noise. J. Basic Eng. **86**, 97–106 (1964)
38. Mockus, J., Tiesis, V., Zilinskas, A.: The application of Bayesian methods for seeking the extremum. In: Dixon, L., Szego, G. (eds.) Towards Global Optimisation 2, pp. 117–130. Elsevier, Amsterdam (1978)
39. Auer, P.: Using confidence bounds for exploitation-exploration trade-offs. J. Mach. Learn. Res. **3**(3), 397–422 (2002)
40. Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.: Gaussian process optimization in the bandit setting: no regret and experimental design, pp. 1–8 (2009)
41. Jones, D.R., Perttunen, C.D., Stuckman, B.E.: Lipschitzian optimization without the Lipschitz constant. J. Optim. Theory Appl. **79**(1), 157–181 (1993)
42. Fletcher, R.: Practical Methods of Optimization. John Wiley & Sons, Hoboken (2013)
43. Gaviano, M., Kvasov, D.E., Lera, D., Sergeyev, Y.D.: Software for generation of classes of test functions with known local and global minima for global optimization. ACM Trans. Math. Softw. **29**(4), 469–480 (2003)

44. De Paola, F., Fontana, N., Giugni, M., Marini, G., Pugliese, F.: An application of the harmony-search multi-objective (HSMO) optimization algorithm for the solution of pump scheduling problem. Procedia Eng. **162**, 494–502 (2016)
45. Castro-Gama, M., Pan, Q., Lanfranchi, E.A., Jomoski, A., Solomatine, D.P.: Pump scheduling for a large water distribution network. Milan, Italy. Procedia Eng. **186**, 436–443 (2017)