Energy 312 (2024) 133416

Contents lists available at ScienceDirect

Energy

journal homepage: www.elsevier.com/locate/energy

# A hybrid algorithm based on Bayesian optimization and Interior Point OPTimizer for optimal operation of energy conversion systems

Loukas Kyriakidis [a,*], Miguel Alfonso Mendez [b], Martin Bähr [a]

[a] German Aerospace Center, Institute of Low-Carbon Industrial Processes, Simulation and Virtual Design Department, Walther-Pauer-Straße 5, Cottbus, 03046, Germany

[b] von Karman Institute for Fluid Dynamics, Environmental and Applied Fluid Dynamics Department, Waterloosesteenweg 72, Sint-Genesius-Rode, 1640, Belgium

ARTICLE INFO

*Keywords:*
Nonlinear global optimization
Bayesian optimization
IPOPT
Hybrid method
Renewable steam generation

ABSTRACT

Optimization methods are essential to improve the operation of energy conversion systems including energy storage equipment and fluctuating renewable energy. Modern systems consist of many components, operating in a wide range of conditions and governed by nonlinear balance equations. Consequently, identifying their optimal operation (e.g. minimizing operational costs) requires solving challenging optimization problems, with the global optimum often hidden behind many local ones. In this work, we propose a hybrid method that advantageously combines Bayesian optimization (BO) and Interior Point OPTimizer (IPOPT). The BO is a global approach exploiting Gaussian process regression to build a surrogate model of the cost function to be optimized, while IPOPT is a local approach using quasi-Newton updates. The proposed BO-IPOPT combination allows leveraging the parameter space exploration of the BO with the quasi-Newton convergence of IPOPT once solution candidates are in the neighborhood of an optimum. Using a challenging constrained test function, we test BO-IPOPT in accuracy and computational efficiency. Finally, we showcase the proposed method in the optimal operation of a renewable steam generation system. The results show that BO-IPOPT combines high accuracy and computational efficiency, achieving up to 50% better objective function values at the same CPU time than other state-of-the-art methods.

## 1. Introduction

Numerical optimization is widely used to define the optimal operation [1–6] of energy conversion systems with many component combinations. A large number of system components, their interactions and included control parameters such as temperatures, mass flows or bypasses, combined with their nonlinear response, renders the optimization of these systems that are usually high dimensional and nonconvex. Yet, global optimization is required for model predictive control framework [7–10], where optimization methods with high accuracy and low computational cost are necessary.

Optimization methods can be classified into local and global, see e.g. [11,12]. Local methods use the information in the neighborhood of a candidate solution to propose an update (improvement). Depending on whether the update relies on the cost function's gradient computation, these can be further classified as gradient-free (e.g. Nelder–Mead) and gradient-based (e.g. quasi-Newton methods). Local methods converge faster (i.e. with fewer cost function evaluations) than global ones if the starting point is sufficiently close to the optimum or if the cost function is (at least locally) convex. However, these methods are more prone to get stuck in local minima. A simple approach to mitigate this risk is to use multi-start (MS) algorithms [13], consisting in repeating the local optimization from multiple starting points. Leveraging the fast convergence of local methods, this approach can capture a large distribution of local optima, from which the one with the best objective function value can be taken as the best guess for the global optimum.

Global methods can be further classified into deterministic and stochastic. Deterministic global optimization is mainly based on concepts of enumeration, generating cuts, and bounding to feasible regions that do not contain any optimal solution [12]. Nevertheless, commercial deterministic global solvers, like BARON [14], are extremely computational expensive especially in nonlinear nonconvex problems, where the CPU time increases exponentially with the number of variables and constraints. Common simplifications, which aim to reduce the complexity of the problem so that a global method like BARON can provide the global solution in reasonable time, are the linear modeling of system components [15] and the linearization of nonlinear component equations [16]. However, the former does not always guarantee a

* Corresponding author.
*E-mail address:* loukas.kyriakidis@dlr.de (L. Kyriakidis).

https://doi.org/10.1016/j.energy.2024.133416
Received 24 November 2023; Received in revised form 7 August 2024; Accepted 8 October 2024
Available online 15 October 2024

## Nomenclature

### Acronyms

| | |
|---|---|
| BO | Bayesian Optimization |
| CPU | Central Processing Unit |
| EI | Expected Improvement |
| GP | Gaussian Process |
| GPR | Gaussian Process Regression |
| HTF | Heat Transfer Fluid |
| HTHP | High Temperature Heat Pump |
| HTHX | High Temperature Heat Exchanger |
| IPOPT | Interior Point OPTimizer |
| LS | Local Search |
| LTHX | Low Temperature Heat Exchanger |
| TES | Thermal Energy Storage |
| WT | Wind Turbine |

### Latin symbols

| | |
|---|---|
| $c_{p,\mathrm{f}}$ | Heat capacity of thermal oil |
| $c_{p,\mathrm{s}}$ | Heat capacity of thermal storage |
| $\boldsymbol{D}$ | Set of points |
| $d(\cdot,\cdot)$ | Euclidean distance |
| $f$ | Objective function |
| $\boldsymbol{g}$ | Set of inequality constraints |
| $g_{\mathrm{grid}}$ | Electricity price |
| $\boldsymbol{h}$ | Set of equality constraints |
| $\boldsymbol{I}$ | Identity matrix |
| $\boldsymbol{J}$ | Operating costs |
| $\boldsymbol{K}$ | Kernel matrix |
| $K$ | Number of outer iterations |
| $k$ | Kernel function |
| $l$ | Length scale |
| $\dot{m}_{\mathrm{III}}$ | LTHX inlet mass flow |
| $\dot{m}_{\mathrm{II}}$ | HTHX outlet mass flow |
| $\dot{m}_{\mathrm{IV}}$ | LTHX outlet mass flow |
| $\dot{m}_{\mathrm{I}}$ | HTHX inlet mass flow |
| $m_{\mathrm{s}}$ | Mass of thermal storage |
| $n$ | Number of dimensions |
| $N_0$ | Number of initialization points |
| $N_{\mathrm{bc}}$ | Number of best candidates selected at each iteration |
| $N_{\mathrm{c}}$ | Number of candidates considering in EI |
| $P_{\mathrm{grid}}$ | Electric power of grid |
| $P_{\mathrm{HTHP}}$ | Electric power of HTHP |
| $P_{\mathrm{WT}}$ | Electric power of wind turbine |
| $\dot{Q}_{\mathrm{s,ch}}$ | Storage heat flow rate during charge |
| $\dot{Q}_{\mathrm{s,dch}}$ | Storage heat flow rate during discharge |
| $\boldsymbol{R}$ | Rotational speed compressor |
| $t$ | Time |
| $T_1$ | Storage outlet temperature during charge |
| $T_2$ | Inlet temperature steam generator |
| $T_3$ | Outlet temperature steam generator |
| $T_4$ | Storage outlet temperature during discharge |
| $T_{\mathrm{III}}$ | LTHX inlet temperature |
| $T_{\mathrm{II}}$ | HTHX outlet temperature |
| $T_{\mathrm{IV}}$ | LTHX outlet temperature |
| $T_{\mathrm{I}}$ | HTHX inlet temperature |
| $T_{\mathrm{s}}$ | Storage temperature |
| $\bar{u}$ | Objective value of $u$ at $\bar{\boldsymbol{x}}$ |
| $\hat{u}$ | GPR model of $u$ |
| $u$ | Objective function with penalty terms |
| $\bar{\boldsymbol{x}}$ | $\boldsymbol{x}$-value of sample points |
| $\boldsymbol{x}$ | $n$-dimensional decision variable |
| $\boldsymbol{x}^\star$ | $\boldsymbol{x}$-value of the local search |
| $y$ | Objective value of $f$ at $\boldsymbol{x}$ |
| $y^\star$ | $y$-value of the local search |
| $y_{\min}$ | Minimum $y$-value |

### Greek symbols

| | |
|---|---|
| $\alpha$ | Regularization term |
| $\beta_1, \beta_2$ | Fluid bypass for charging and discharging |
| $\gamma$ | Relaxation parameter |
| $\Delta t$ | Discrete time step |
| $\epsilon_{\mathrm{ch}}$ | Effectiveness of charging |
| $\epsilon_{\mathrm{dch}}$ | Effectiveness of discharging |
| $\boldsymbol{\lambda}$, $\boldsymbol{\rho}$ | Penalty weight vectors |
| $\xi$ | Amount of exploration |
| $\boldsymbol{\Sigma}$ | Covariance function |
| $\sigma$ | Standard deviation |
| $\boldsymbol{\Phi}$ | Cumulative distribution function |
| $\phi$ | Probability distribution function |
| $\Omega$ | Set of $\mathbb{R}^n$ |

realistic system behavior, whereas the latter requires an appropriate linearization technique depending on the model complexity, where a balance between accuracy and time efficiency must be found for the definition of the grid fineness.

An alternative to the deterministic global search is the random search of stochastic global optimization. These can be further classified into single candidate (e.g. simulated annealing) or population-based (e.g. genetic algorithms or particle swarms) [17–20] depending on the number of solutions that are iterated upon, or into metaheuristics or surrogate-based depending on the criteria used for the update. Metaheuristic methods (such as genetic algorithms or particle swarms) use a bio-inspired heuristic strategy to advance candidate solutions, while surrogate-based approaches (such as Bayesian optimization [21–24]) build a surrogate model of the function to optimize and use the model to drive the new evaluations. Global stochastic methods can avoid local minima, but tend to require a much larger number of cost-function evaluations than local ones.

The complementary advantages of these methods have motivated various hybrid approaches combining global and local methods [17, 25,26]; the reader is referred to [27] for an overview of hybridization strategies. While most hybrid methods have combined metaheuristics with local methods, the recent focus seeks to combine surrogate-based methods and gradient-based optimization [28–30] to maximize the sample efficiency of the hybrid formulation. Surrogate-based methods build successive approximations of the cost function and generally require less computational resources than metaheuristics approaches, hence enabling for more sample-efficient hybrid methods.

This work explores the combination of Bayesian optimization and a quasi-Newton local approach to determine the optimal operation of energy conversion systems. In particular, we use the classical BO with Gaussian process regression (GPR) and expected improvement [21] together with the Interior Point OPTimizer (IPOPT) [31,32]. Our combination is similar to the one proposed in [30], but differs in how

the local method is integrated with the GPR, as further detailed in Section 2.3.

The aim of this study is to develop a method combining the advantages of global stochastic and local deterministic algorithms. More precisely, our hybrid approach profits from the global exploration of BO and the effective local search of IPOPT. BO explores the entire input space on a global scale, taking into account smooth variations and ignoring details. Then, IPOPT focuses on the regions of interest provided by the BO part. The advantage of this method is, on the one side, that IPOPT increases the convergence speed of BO (requiring fewer cost function evaluations). On the other side, BO determines good starting points for the local solver.

This paper is organized as follows: Section 2 introduces the proposed hybrid method BO-IPOPT. Sections 3 and 4 present the test cases analyzed in this work, namely a constrained test function with known global solution and the constrained optimization problem arising from the operational management of a renewable steam generation system to test the performance of our proposed method. Finally, Section 5 summarizes the main conclusions and provides perspective for future works.

## 2. Optimization methods

In this work, we consider the general constrained optimization problem, defined as:

$$\min_{\boldsymbol{x}\in\Omega}\left\{ f(\boldsymbol{x}) \;\text{ s. t. }\; \boldsymbol{h}(\boldsymbol{x})=\boldsymbol{0}, \boldsymbol{g}(\boldsymbol{x})\le\boldsymbol{0} \right\} \tag{1}$$

with $\boldsymbol{x}\in\Omega\subseteq\mathbb{R}^n$ the $n$-dimensional decision variable contained in set $\Omega$, $f:\mathbb{R}^n\to\mathbb{R}$ the objective function, $\boldsymbol{h}:\mathbb{R}^n\to\mathbb{R}^p$ the set of equality constraints, and $\boldsymbol{g}:\mathbb{R}^n\to\mathbb{R}^q$ the set of inequality constraints. The functions $f,\boldsymbol{h},\boldsymbol{g}$ can be nonlinear and nonconvex, but are assumed to be sufficiently smooth.

We aim to develop an optimization method that provides a sequence of candidate solutions $\{\boldsymbol{x}_K\}$ converging towards the global minimum $\boldsymbol{x}_K\to\boldsymbol{x}^\star$ with the highest probability and the fewer iterations $K$. In what follows, we briefly introduce the BO and IPOPT methods and motivate our interest in their combination.

### 2.1. Bayesian Optimization (BO)

BO is a global black-box optimization approach, i.e. requiring no mathematical definition of the objective function. The main idea is to use the sampling at the candidate solutions $\{\boldsymbol{x}_K\}$ to build a surrogate model of the cost function. Following [21], the surrogate model is usually built with GPR, which is a kernel regression method [33,34] allowing the analytical computation of both the regression and its uncertainties. The uncertainties of the model can be used to balance exploitation, i.e. the tendency to sample where the surrogate predicts best objective values, with exploration, i.e. the tendency to sample where the surrogate has the highest uncertainty.

An approach to deal with constraints, often used in stochastic methods [17], is to augment the objective function as follows:

$$u(\boldsymbol{x}) = f(\boldsymbol{x}) + \boldsymbol{\rho}^\top\boldsymbol{h}(\boldsymbol{x})^2 + \boldsymbol{\lambda}^\top\max(\boldsymbol{0},\boldsymbol{g}(\boldsymbol{x}))^2 \tag{2}$$

where $\boldsymbol{\rho},\boldsymbol{\lambda}\ge\boldsymbol{0}$ are penalty weights vectors associated with the magnitude of constraint violation. Several authors (see e.g. [35,36]) have proposed adaptation methods for these vectors, to adjust the weights for the constraint violation according to the evolution of $f(\boldsymbol{x})$ during the optimization process. Nevertheless, as we here focus on a first proof of concept of the hybridization, we leave these adaptation to future work.

To solve the optimization problem introduced by (1), BO requires the following two ingredients:

1. *GPR*: The GPR builds the surrogate model considering it as a Gaussian process (GP), i.e. a multivariate Gaussian distribution in the domain $\Omega$. This distribution is initialized with a *prior* mean $m(\boldsymbol{x})$ and covariance defined by a kernel function, usually taken as a Gaussian $k(\boldsymbol{x},\boldsymbol{x}') = \exp(-d(\boldsymbol{x},\boldsymbol{x}')^2/2l^2)$, with $l$ the length scale of the kernel and $d(\cdot,\cdot)$ the Euclidean distance between points in $\Omega$. As sample points $\{\bar{\boldsymbol{x}}_1,\dots,\bar{\boldsymbol{x}}_{N_c}\}$ and associated objective values $\bar{u}_n = u(\bar{\boldsymbol{x}}_n)$ are collected, the GPR updates the underlying GP using standard conditioning rules [33,34] such that the (surrogate) predictions in $\boldsymbol{x}\in\Omega$ become $\hat{u}(\boldsymbol{x})\sim\mathrm{GP}(\mu(\boldsymbol{x}),\boldsymbol{\Sigma}(\boldsymbol{x}))$ with mean $\mu(\boldsymbol{x})$ and covariance $\boldsymbol{\Sigma}(\boldsymbol{x})$ functions defined as:
$$\mu(\boldsymbol{x}) = \boldsymbol{K}_*^T\left(\boldsymbol{K}_{**}+\alpha\boldsymbol{I}\right)^{-1}u(\bar{\boldsymbol{x}}) \tag{3}$$
$$\boldsymbol{\Sigma}(\boldsymbol{x}) = \boldsymbol{K}-\boldsymbol{K}_*^T\left(\boldsymbol{K}_{**}+\alpha\boldsymbol{I}\right)^{-1}\boldsymbol{K}_* \tag{4}$$
where $\boldsymbol{K}=k(\boldsymbol{x},\boldsymbol{x})$, $\boldsymbol{K}_*=k(\boldsymbol{x},\bar{\boldsymbol{x}})$, $\boldsymbol{K}_{**}=k(\bar{\boldsymbol{x}},\bar{\boldsymbol{x}})$, $\boldsymbol{I}$ is the identity matrix of appropriate size and $\alpha$ is a regularization parameter that avoids the fully interpolative behavior of the GPR. The main hyperparameters of the regression are the kernel's length scale $l$ and the regularization $\alpha$; the first determines the smoothness of the function, while the second regularizes the sensitivity of the regression towards noise in the case of stochastic objective functions.
2. *Acquisition function*: This function controls the location of the new candidate solutions. The common approach, also used in this work, is to use the expected improvement (EI) function. This is defined as:
$$\mathrm{EI}(\boldsymbol{x}) = \begin{cases} (\hat{u}(\boldsymbol{x}^+)-\mu(\boldsymbol{x})-\xi)\Phi(Z) & \\ +\sigma(\boldsymbol{x})\phi(Z), & \text{if } \sigma(\boldsymbol{x})>0 \\ 0, & \text{if } \sigma(\boldsymbol{x})=0 \end{cases} \tag{5}$$
with
$$Z = \begin{cases} \frac{\hat{u}(\boldsymbol{x}^+)-\mu(\boldsymbol{x})-\xi}{\sigma(\boldsymbol{x})}, & \text{if } \sigma(\boldsymbol{x})>0 \\ 0, & \text{if } \sigma(\boldsymbol{x})=0 \end{cases} \tag{6}$$
where $\Phi$ and $\phi$ denote the cumulative distribution function and the probability distribution function, respectively. The first term controls the exploitation: this term is large if the new samples are close to the expected optima according to the surrogate model. The second term controls exploration: this term is large if the new samples are in the area of large $\sigma(\boldsymbol{x})=\mathrm{diag}(\boldsymbol{\Sigma})$. The parameter $\xi$ sets a threshold over the minimal expected improvement that justifies the exploration and is a hyperparameter of the BO; hence the larger this value, the more exploration is produced. EI effectively balances exploration and exploitation incorporating the probability and the magnitude of improvement. On the one hand, this means that it selects points showing a high predicted mean, which could improve the current best observation. On the other hand, it takes into consideration points with higher uncertainty, where there might be potential for improvement.

In the classical BO, the algorithm begins with a set of randomly chosen candidate solutions and iterates alternating one update of the GPR and the maximization of EI to define new candidates.

### 2.2. Interior Point OPTimizer (IPOPT)

The open-source software package IPOPT [32] is a popular tool for solving large-scale nonlinear optimization problems. The solver is mainly based on a primal–dual interior-point method combined with a filter line-search method [31]. IPOPT can also be applied to nonconvex problems, but should be at least once, ideally twice, continuously differentiable.

Algorithmically, IPOPT transforms any inequality constraints of the original problem (Eq. (1)) into natural logarithmic barrier terms in the objective function, including a barrier parameter, thus considering a parametric problem. Based on this, a sequence of equality-constrained problems (barrier problems) is solved for decreasing values of the mentioned barrier parameter. This approach is repeated until a point satisfies the first-order Karush-Kuhn–Tucker optimality conditions. Note that also maximizers and saddle points satisfy the latter optimality conditions, but IPOPT internally uses a Hessian regularization that avoids obtaining maximizers and saddle points. For each barrier subproblem, a Newton-type algorithm with line search is applied, including the solution of indefinite sparse symmetric linear systems. In particular, the overall performance (runtime, accuracy and robustness) strongly depends on the properties of the chosen sparse linear solver.

Overall, IPOPT is a highly efficient solver to find a local solution of a large-scale nonlinear nonconvex constrained optimization problem. However, the computed local optimum depends strongly on the selected starting point. Consequently, the initialization determines the rate of convergence to a solution and to which optimum the algorithm converges. It should be noted that the local optimization method can also perform poorly and even fail if the initial guess is unfavorable. To overcome these challenges, MS algorithms are often used, as already mentioned in Section 1. The idea behind this is to run an optimization algorithm multiple times from different starting points and the best objective function value is considered as the best estimate for the global optimum.

### 2.3. Hybrid method BO-IPOPT

The proposed BO-IPOPT combination is illustrated in Algorithm 1. The algorithm starts with a set of initialization points to generate a first surrogate model via GPR (cf. lines 1–4). In doing so, the evaluation of the initialization points is possible, since the objective function is assumed to be known.

The main part of the algorithm consists of alternating steps of BO and IPOPT until the total number of outer iterations $K$ is satisfied (cf. lines 6–13). More specifically, new candidates for the GPR are first selected by evaluating the acquisition function EI. These new candidates most likely do not fulfill the constraints of the optimization problem considered, especially in nonlinear nonconvex problems of higher dimension with a large number of constraints. However, instead of updating the surrogate model directly with the newly selected candidates (as in the classical BO), IPOPT then considers the new candidates as starting points and determines the optimum for each of them, so the surrogate model is then updated based on these local optima.

It should be noted that the number of best candidates selected at each outer iteration should be chosen considering that the GPR scales with the cube of the samples involved because of the inversion of the covariance matrix $\boldsymbol{K}_{**}$ via Cholesky factorization. Nevertheless, in all the investigated test cases, the computational cost of the objective function evaluation (local search) significantly outweighs the cost of the GPR for the selected hyperparameters (small number of samples involved), as explained in Sections 3 and 4. This happens because the slope in the CPU time for the local search is larger than the overhead of the GPR, which shows that the overhead of the GPR is not particularly relevant for these two cases and the hyperparameters considered. Moreover, it should be underlined that the proposed approach uses only feasible and numerical stable solutions (in line 9): if the solution of IPOPT is not feasible or contains numerical issues like NaN or inf values, the algorithm selects the next best candidate, and IPOPT proceeds with this.

Finally, our hybrid method returns (line 14) the minimum objective function value included in the set of points $D$ without considering the initial set of points $D_0$, so that $y_{\min}$ always represents at least a local solution of Eq. (1) or the global minimum in the best case.

**Algorithm 1** Hybrid Method BO-IPOPT

**Input:** length scale $l$; regularization term $\alpha$ of GPR; amount of exploration $\xi$; number of initialization points $N_0$; number of candidates $N_c$ considering in EI; number of best candidates $N_{bc}$ selected at each iteration; penalty weight vectors $\boldsymbol{\rho}, \boldsymbol{\lambda}$; number of outer iterations $K$;

1: generate a number of points $\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_{N_0}\}$ in $\Omega$;
2: evaluate $y_n = u(\boldsymbol{x}_n)$ for $n = 1{:}N_0$;
3: let $D_0 = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N_0}$;
4: construct a GPR model $\hat{u}_0$ from $D_0$;
5: $n = N_0, j = 0$;
6: **while** $j < K$ **do**
7: generate a new group of points $\{\bar{\boldsymbol{x}}_1, \dots, \bar{\boldsymbol{x}}_{N_c}\}$ in $\Omega$;
8: $\{\boldsymbol{x}_{n+1}, \dots, \boldsymbol{x}_{n+N_{bc}}\} = \arg\max \mathrm{EI}(\bar{\boldsymbol{x}}_i; \hat{u}_n)$ for $i = 1{:}N_c$;
9: solve $[y_k^\star, \boldsymbol{x}_k^\star] = \mathrm{IPOPT}(\boldsymbol{x}_k)$ for $k = n+1{:}n+N_{bc}$;
10: $D_{j+1} = D_j \cup \{(\boldsymbol{x}_{n+1}^\star, y_{n+1}^\star), \dots, (\boldsymbol{x}_{n+N_{bc}}^\star, y_{n+N_{bc}}^\star)\}$
11: update GPR model $\hat{u}_{j+1}$ from $D_{j+1}$;
12: $n = n + N_{bc}, j = j + 1$;
13: **end while**
14: **return** $y_{\min} = \min\{y_j^\star\}_{j=1}^{KN_{bc}}$;

The advantage of the proposed BO-IPOPT method is that IPOPT improves the convergence of the BO by moving some of the candidate solutions towards optima. When these are local optima, the EI evaluation allows for maintaining global exploration and improving the regression where needed. If any of these is a global optimum, the EI keeps favoring its sampling in the following iterations. Overall, this method is not only efficient for BO to speed up its convergence but also for IPOPT, since BO effectively determines good initial points for the local solver at each outer iteration $K$ based on the acquisition function EI. It is worth noticing that any local solver could replace IPOPT, which was chosen here because of its efficient, robust, open-source implementation.

As mentioned in Section 1, the recently proposed approach BOwLS [30] (cf. corresponding Algorithm 3 in the mentioned article) is similar to ours. In [30], the BO framework is also used to determine the local solver's starting points and thus create a suitable MS formulation. However, it should be noted that there are some technical differences to our version. First, the initial GPR model is built from the results of the local searches (not random ones); this brings the risk of restricting the sampling region of the GPR. Second, the initial and updated GPR models are based on the value of the objective function from the local optima but on the inputs of the initial points *before* using the local solver ($\boldsymbol{x}$ instead of $\boldsymbol{x}^\star$). Consequently, the GPR at each iteration does not approximate the underlying function but another one that shares the same local minima identified thus far. Finally, BOwLS uses the conjugate gradient method from the SciPy package as a local search, thus not accounting for constraints and the feasibility set of the problem.

**Algorithm 2** MS-IPOPT

**Input:** Number of outer iterations $K$;

1: **for** $i = 1 : K$ **do**
2: generate a random point $\boldsymbol{x}_i$ in $\Omega$;
3: solve $[y_i^\star, \boldsymbol{x}_i^\star] = \mathrm{IPOPT}(\boldsymbol{x}_i)$;
4: **end for**
5: **return** $y_{\min} = \min\{y_i^\star\}_{i=1}^{K}$;

## 3. Constrained Ackley function

In this section, we showcase the proposed BO-IPOPT approach on the well-known Ackley function, often used to test optimization algo-

rithms [37]. The performance of BO-IPOPT is compared with the classical random-based MS-IPOPT (cf. Algorithm 2) and the BOwLS [30] in terms of accuracy and CPU time. All algorithms were implemented in Python 3.8. More specifically, BO was implemented using the sklearn library [38] for the GPR, while IPOPT is used via the Python Optimization Modeling Objects (Pyomo) software package [39]. To provide the first and second derivative, Pyomo uses the automatic differentiation features in the Ampl Solver Library.

All computations were carried out on a machine with Intel(R) Core(TM) i7-8665U CPU. To allow for a fair comparison between hybrid approaches, we implement IPOPT as the local solver for the BOwLS since the conjugate gradient method from the SciPy package used in [30] is not designed to handle constrained problems. Additionally, for consistency, we consider the same parameters for both hybrid methods, i.e. $N_0 = 10$, $N_c = 1000$, $N_{bc} = 2$, $\xi = 0.01$, $\alpha = 0.1$, $l = 100$, $\boldsymbol{\rho} = \boldsymbol{\lambda} = \mathbf{100}$, and the same acquisition function EI. Concerning the random-based MS-IPOPT, we implement it with $N_{bc} = 2$ (cf. line 2 in Algorithm 2), meaning that two random points are generated for each outer iteration $K$ and used as starting points for the IPOPT solver.

The Ackley function is a challenging problem that is nonconvex and highly multi-modal with multiple local minima and one global minimum, making it difficult for optimization algorithms to find the global optimum. The presence of numerous peaks and valleys tests the algorithm's ability to escape local minima, while its large dimensionality tests the algorithm's ability to handle the curse of dimensionality.

Since this function was originally designed as an unconstrained optimization test case, we add two inequality constraints as proposed in [37]. These constraints represent two different types of conditions: a linear inequality and a nonlinear inequality constraint. Their purpose is to help evaluate how well the optimization algorithms handle different kinds of constraints and they make the problem more realistic, as real-world problems typically involve a mix of linear and nonlinear constraints. The resulting constrained optimization problem is defined as follows:

$$\begin{aligned}\min f(\boldsymbol{x}) = \; & -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right) \\ & - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e\end{aligned} \tag{7}$$

$$\text{s. t.} \quad \sum_{i=1}^{n} x_i \le 0, \quad \|\boldsymbol{x}\|_2 - 5 \le 0, \quad \boldsymbol{x} \in [-5, 10]^n \tag{8}$$

where $n$ denotes the problem's dimension, set to $n = 100$ for this experimental study. The optimization problem defined in Eqs. (7)–(8) has a global minimum at $\boldsymbol{x} = \mathbf{0}$ with objective function value 0, i.e. $f(\boldsymbol{x}) = 0$. Since all implemented algorithms are based on a certain randomness that could lead to a varying optimizer's performance from run to run, our numerical experiments were repeated 100 times with a number of outer iterations $K = 300$. By conducting multiple runs, we can compute statistical measures such as mean providing a more reliable assessment of the algorithm's performance and ensuring that the results are not overly dependent on a particular starting point.

The optimization results for the three methods are shown in Fig. 1. The figure compares the averaged minimum objective function value obtained and the averaged CPU time versus the number of outer iterations as well as the corresponding solver performance, i.e. averaged minimum objective function value versus averaged CPU time. Comparing the optimization results obtained using BO-IPOPT, MS-IPOPT, and BOwLS in terms of both the minimum objective function value and the CPU time is interesting for the following reasons: on the one hand, the minimum objective function value indicates how well each optimization algorithm performs in terms of finding the best solution to the problem. A lower objective function value typically represents a better solution. On the other hand, the CPU time measures the computational resources required by each method to reach the solution. This is crucial in practical applications where time and computational resources are limited. The results on the right of Fig. 1 help to compare the minimum objective function value reached by each optimizer at a certain CPU time. This figure shows that the proposed BO-IPOPT outperforms the other two optimizers. As expected, MS-IPOPT and BO-IPOPT start with a higher minimum objective function value than BOwLS because both methods use a random initialization, while BOwLS considers local optima as initialization for the initial GP model.

BO-IPOPT converges faster towards the global minimum. More precisely, the proposed method shows a noteworthy 24% and 47% improvement in minimum objective function value after ≈40 s as well as 17% and 36% improvement after ≈80 s compared to MS-IPOPT and BOwLS, respectively. A more substantial enhancement is achieved by our hybrid approach after ≈30 s, reaching 28% and 50% compared to MS-IPOPT and BOwLS, respectively. The faster convergence of BO-IPOPT can be explained as follows: on the one hand, MS-IPOPT naturally converges more slowly to the global optimum because it strongly depends on its randomly chosen starting points, while BO in BO-IPOPT provides good starting points to IPOPT at each outer iteration and helps the hybrid method to converge faster to the global minimum than MS-IPOPT. On the other hand, the GPR in BO-IPOPT has better performance in providing a better surrogate model since the sampling is more spread (random points for the initial GP model) than in BOwLS, which narrows the sampling near the local optima (local minima for the initial GP model).

The selected large value for $l$ enables the BO's GP model to explore the entire input space on a global scale, taking into account smooth variations and ignoring details of the function to be optimized. Then, IPOPT focuses on the regions of interest provided by the BO part as starting points, ensuring that all constraints are satisfied at each outer iteration $K$. This approach increases the probability that the IPOPT in the hybrid method will avoid becoming stuck in local minima by leveraging the globally informed guidance from the BO part. In addition, the small number for the initialization points $N_0$ avoids unreasonable increase in CPU time due to the correlation matrix inversion in the GPR of the BO part, especially in high dimensions. Regarding CPU time, the simplest MS-IPOPT outperforms both hybrid methods, as shown in Fig. 1; this gives an order of magnitude of the costs for training the GPR surrogate model in the two hybrid methods.

It should be underlined that an exhaustive comparison of the two hybrid methods in several benchmark problems should also include a broader range of hyperparameters; this will be carried out in the extended version of this work. For the investigated test case, the performance gap appears to depend on the outer iterations $K$: as shown in Fig. 1, at $K = 100$, BO-IPOPT is well ahead in the averaged minimum objective function value, but BOwLS reaches similar objectives at $K \approx 170$.

The number of initialization points $N_0$ is a second crucial parameter. It is expected that larger values will favor BO-IPOPT, since they enable better exploration of the solution space, while the local optimization in BOwLS restricts the exploration of the BO. Moreover, increasing $N_0$ increases the number of (initial) local searches in the BOwLS and thus directly leads to higher computational costs in contrast to BO-IPOPT. However, a large $N_0$ increases the computational cost of the GPR, which grows cubically with the number of training data (hence the size) of matrix $\boldsymbol{K}_{**}$ in Eq. (3). Nevertheless, in all the investigated problems, the cost for the GPR is significantly lower than the cost of evaluating the objective function and this parameter has thus a minor impact on the computational costs.

## 4. Renewable steam generation

The second test case is the operation optimization problem of an industrial energy conversion system for renewable steam generation. This power-to-heat system (cf. Fig. 2) was recently proposed in [40] and is currently used as a practical benchmark to evaluate the algorithms. In the following, we first briefly describe the test case and the resulting optimization problem (Section 4.1) and then present and discuss the results of the optimization (Section 4.2).

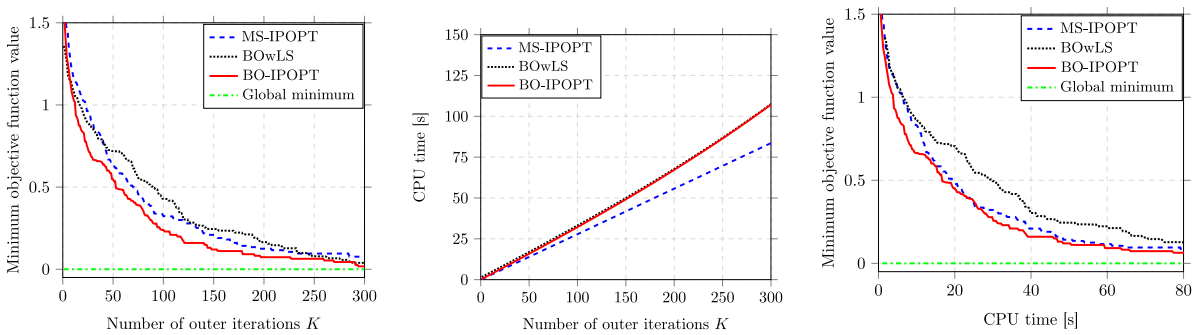


**Fig. 1.** Optimization results for the constrained Ackley function (Eqs. (7)–(8)): comparison of the averaged minimum objective function value (**left**), the averaged CPU time (**middle**) and the respective solver performance (**right**) over 100 trials using MS-IPOPT, BOwLS and BO-IPOPT in relation to the number of outer iterations $K$. The global minimum is also visualized.

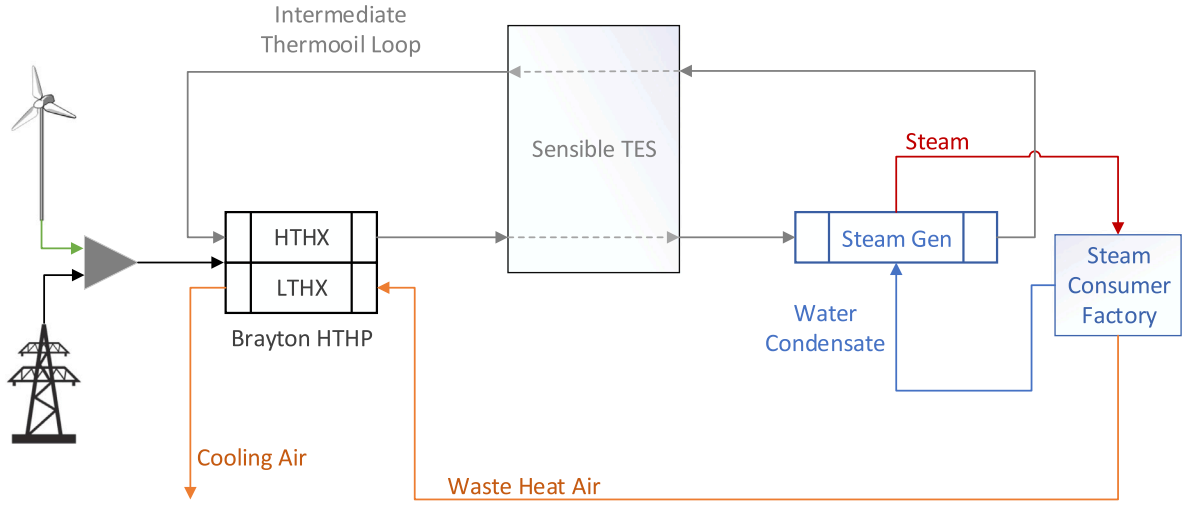


**Fig. 2.** Illustration of the investigated industrial energy conversion system for electrified steam generation recently proposed in [40]. The system consists of a HTHP, a TES and a SG, where the HTHP is powered by electricity from a wind turbine or the power grid. The HTHP uses waste heat air stream as a heat source and enables charging and discharging of the TES via an intermediate thermal oil stream. Furthermore, constant heat demand for the steam consumer factory must be satisfied.

### *4.1. Problem description*

The considered electrified system shown in Fig. 2 aims to provide constant process heat in the form of super-heated steam for an industrial process. The multi-component system mainly consists of 4 units: (i) a wind turbine (WT) to produce renewable electricity driving the system; (ii) a closed reverse Brayton cycle high-temperature heat pump (HTHP), powered by electricity from the WT or the power grid, to generate process heat; (iii) a sensible thermal energy storage (TES) to store excess thermal energy generated during periods of high wind power or low electricity prices; (iv) the steam generator (SG) for providing process steam via an intermediate thermal oil stream and controllable fluid bypasses.

The HTHP and SG models were created with the process simulation software Ebsilon [41], with the former being able to simulate part load behavior. The TES model is developed using a lumped capacitance approach, while the WT power output is modeled from the specific power curve at hub height, i.e. the power curve determines the wind power generated as a function of the wind speed extrapolated to the corresponding height.

In the system configuration (cf. Fig. 3), the HTHP provides high-temperature process heat to an intermediate circuit routed through the TES or directly to the SG via a controllable fluid bypass. Thermal oil is chosen as heat transfer fluid (HTF) in the intermediate loop due to its compactness and fluid phase within the temperature range. During the charging process, the temperature in the TES is heated up by the HTF before it enters the SG; discharging operation is vice versa. In discharge mode, the HTHP's power consumption can be significantly reduced since less heat has to be supplied to the intermediate loop to ensure constant steam generation. In idle operation, the TES is completely bypassed by the HTF. The cold outlet stream is not used for cooling applications in the current setup. For more details, we refer the reader to [40].

Optimization aims to determine the cost-optimal operation, i.e. minimizing operational costs considering the fluctuating wind energy and electricity prices. To build an algebraic model problem, the underlying HTHP and SG models are converted into an algebraic form using polynomial surrogate models, as described in [40]. This results in an algebraic nonlinear nonconvex constrained optimization problem that can be formulated in a discrete setting as:

$$\min \; J(P_{\text{grid}}) = \sum_{k=1}^{n} P_{\text{grid}}^{k} g_{\text{grid}}^{k} \Delta t \tag{9}$$

s. t.

$$P_{\text{grid}}^{k} + P_{\text{WT}}^{k} = 3F_{\text{HTHP}}\left(T_{\text{I}}^{k}, \dot{m}_{\text{I}}^{k}, T_{\text{III}}^{k}, R^{k}\right) \tag{10}$$

$$T_{\text{II}}^{k} = F_{\text{HTHX}}\left(T_{\text{I}}^{k}, \dot{m}_{\text{I}}^{k}, T_{\text{III}}^{k}, R^{k}\right) \tag{11}$$

$$T_{\text{IV}}^{k} = F_{\text{LTHX}}\left(T_{\text{I}}^{k}, \dot{m}_{\text{I}}^{k}, T_{\text{III}}^{k}, R^{k}\right) \tag{12}$$

$$T_{2}^{k} = T_{\text{II}}^{k}\beta_{1}^{k} + T_{1}^{k}\left(1-\beta_{1}^{k}\right) \tag{13}$$

$$T_{\text{I}}^{k} = T_{3}^{k}\beta_{2}^{k} + T_{4}^{k}\left(1-\beta_{2}^{k}\right) \tag{14}$$

$$T_{2}^{k} = 201.92 + \frac{1819.32}{3\dot{m}_{\text{II}}^{k}} \tag{15}$$

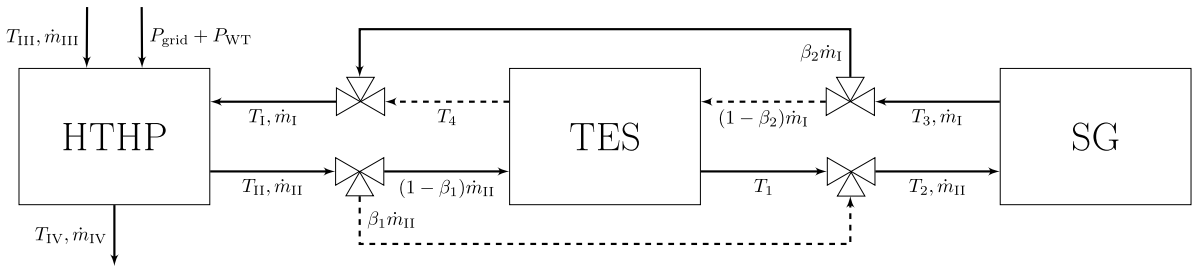


**Fig. 3.** Detailed flow chart [40] of studied system (cf. Fig. 2) including HTHP, TES and SG. HTHP and SG are represented by polynomial surrogate models, while TES is modeled by an effectiveness model. An adjustable fluid bypass $\beta_1 \in [0,1]$ is used to control the heat input to the TES and SG, conversely the bypass $\beta_2 \in [0,1]$ regulates the heat input to the HTHP depending on the thermal state $T_s$ of the TES and SG outlet stream. The *solid* and *dashed* lines indicate the charging and discharging mode in a simplified way. In addition, simultaneous charging and discharging is not allowed. For example, *charge* mode is $\beta_1 \in [0,1), \beta_2 = 1$, *discharge* mode implies $\beta_2 \in [0,1), \beta_1 = 1$, and *idle* mode represents $\beta_1 = \beta_2 = 1$.

$$T_3^k = 196.3 - \frac{188.4}{3\dot{m}_{\mathrm{I}}^k} \tag{16}$$

$$\dot{Q}_{\mathrm{s,ch}}^k = 3\dot{m}_{\mathrm{II}}^k c_{p,\mathrm{f}} \left(T_{\mathrm{II}}^k - T_1^k\right)\left(1-\beta_1^k\right) \tag{17}$$

$$\dot{Q}_{\mathrm{s,dch}}^k = 3\dot{m}_{\mathrm{I}}^k c_{p,\mathrm{f}} \left(T_4^k - T_3^k\right)\left(1-\beta_2^k\right) \tag{18}$$

$$\dot{Q}_{\mathrm{s,ch}}^k \dot{Q}_{\mathrm{s,dch}}^k \leq \gamma \tag{19}$$

$$T_1^k = T_{\mathrm{II}}^k - \epsilon_{\mathrm{ch}}\left(T_{\mathrm{II}}^k - T_{\mathrm{s}}^{k-1}\right) \tag{20}$$

$$T_4^k = T_3^k - \epsilon_{\mathrm{dch}}\left(T_3^k - T_{\mathrm{s}}^{k-1}\right) \tag{21}$$

$$T_{\mathrm{s}}^k = T_{\mathrm{s}}^{k-1} + \frac{\dot{Q}_{\mathrm{s,ch}}^k - \dot{Q}_{\mathrm{s,dch}}^k}{m_{\mathrm{s}} c_{\mathrm{p,s}}} \Delta t \tag{22}$$

$$T_{\mathrm{s}}^0 = \widetilde{T}_0, \quad T_{\mathrm{s}}^n = T_{\mathrm{s}}^0 \tag{23}$$

$$T_{\mathrm{I}}^k \in [177, 250], \ \dot{m}_{\mathrm{I}}^k \in [5, 16] \tag{24}$$

$$T_{\mathrm{III}}^k \in [60, 100], \ R^k \in [0.8, 1.53] \tag{25}$$

with a uniformly spaced time grid $t_k = \Delta t_k$ for $k = 1, \ldots, n$, so that the functions are considered only at the discrete time points, i.e. $T_I^k := T_I(t_k)$. The linear objective function (Eq. (9)) relies on the system's operating cost that is directly related to the consumed grid power of the HTHP. The power balance and outlet temperatures of the HTHP are described by Eqs. (10)–(12), where $F_{\mathrm{HTHP}}$, $F_{\mathrm{HTHX}}$, and $F_{\mathrm{LTHX}}$ represent the corresponding surrogate models as a function of inlet temperatures, mass flow and rotational shaft speed. The bypass modeling is represented by Eqs. (13) and (14), while Eqs. (15) and (16) reflect the SG surrogate models. The charging and discharging heat flows (Eqs. (17), (18) and (19)) depend on the HTF mass flow, the temperature level and the fluid flow bypasses. Moreover, charging and discharging at the same time is not allowed.

The constraints in Eqs. (20)–(22) relate to the TES effectiveness model and the storage temperature change. For a complete set-up, an initial storage temperature $\widetilde{T}_0$ is required, which is assumed to be the same at the end of the operating period (Eq. (23)). The simplified models in Eqs. (10)–(12) are valid within the box constraints in Eqs. (24) and (25), while other variables are naturally limited by the system itself. The factor 3 in Eqs. (10) and (15)–(18) arises because the surrogate models were derived for a single HTHP, but three HTHPs are operated in parallel to keep the component dimensions within a moderate scale.

### 4.2. Optimization results

This section presents the optimization results of the proposed system for renewable steam generation. We consider a one-day system operation and set the time step to $\Delta t = 1$ h, giving the total number of discrete steps $n = 24$. The scenario for the WT power production and the grid electricity price are displayed in Fig. 4 as a function of time during 24 h. The time horizon considered, resulting in 408 decision variables controlling the system, makes this nonlinear nonconvex optimization problem with a large number of equality and inequality constraints (360 in total) unfeasible for a commercial global optimization solver like BARON.

The optimization results are presented in Fig. 5. As in Section 3, we compare the averaged minimum objective function value, the averaged CPU time and the corresponding solver performance of the proposed BO-IPOPT to MS-IPOPT and BOwLS. The same hyperparameters as in the previous test case are considered. Our numerical experiments are repeated 20 times for averaging out the stochastic nature of the optimizers. We reduce this number with respect to the previous test case because this optimization problem requires larger computational time due to the larger dimension and the large set of equality constraints. Since the global minimum of this optimization problem is unknown, we use MS-IPOPT with 10,000 different initialization points to explore the parameter space and for each point MS-IPOPT determines the optimum. Using 10,000 starting points ensures a comprehensive exploration of the parameter space. This large number increases the probability of finding the global minimum or a solution very close to it. The best result, herein considered as an estimate of the global optimum, lies at 1046.53€. The same optimum is also obtained for BO-IPOPT and BOwLS with 10,000 different initialization points and is shown in Fig. 5 on the left and the right.

As in the previous test case, BO-IPOPT outperforms the other approaches in terms of convergence, while both hybrid methods in the real-world case converge faster towards the best reference solution than the MS-IPOPT. The same observations made from the previous test case on the role of the starting points in the initial GPR for BO-IPOPT and BOwLS apply to this case. Moreover, the same factors as in the constrained Ackley function contribute to the faster convergence of BO-IPOPT compared to MS-IPOPT. However, it is worth noticing that none of the optimizers approach the best known solution (estimated by the 10,000 IPOPT iterations) within the $K = 300$ outer iterations on average. This, together with the minor improvements (slower convergence towards optimum compared to constrained Ackley function) achieved by all optimizers, highlights the complexity of the optimization problem, which is nonlinear, nonconvex and high-dimensional with a large number of constraints. BO-IPOPT shows an improvement in the minimum objective function value of ≈1% compared to the other two methods. To address the complexity of the problem, identifying the best hyperparameters for BO-IPOPT and selecting a higher number of best candidates ($N_{\mathrm{bc}} > 2$) at each outer iteration, whose IPOPT evaluations run in parallel to explore different regions of the parameter space simultaneously, can be incorporated into future optimization strategies.

In terms of computational cost, Fig. 5 shows a nearly linear trend for both the hybrid methods and the MS-IPOPT, with the slope being much larger than in the previous test case because of the larger dimensionality of the problem. This trend in the computational time remains the same for two different numbers of best candidates $N_{\mathrm{bc}}$, as seen in Fig. 6. The linear trend confirms that the computational cost of the GPR is negligible compared to the cost function evaluation (local search of

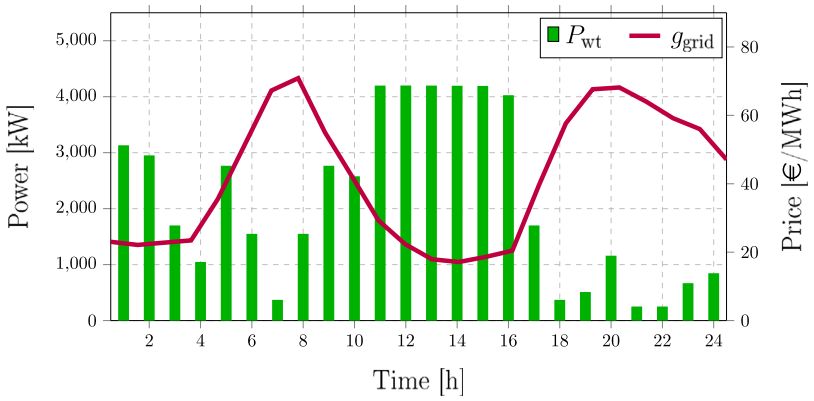


**Fig. 4.** Visualization of 24 h reference data $P_{wt}$ (WT power output) and $g_{pr,grid}$ (electricity price) to minimize operating cost in Eqs. (9)–(25).

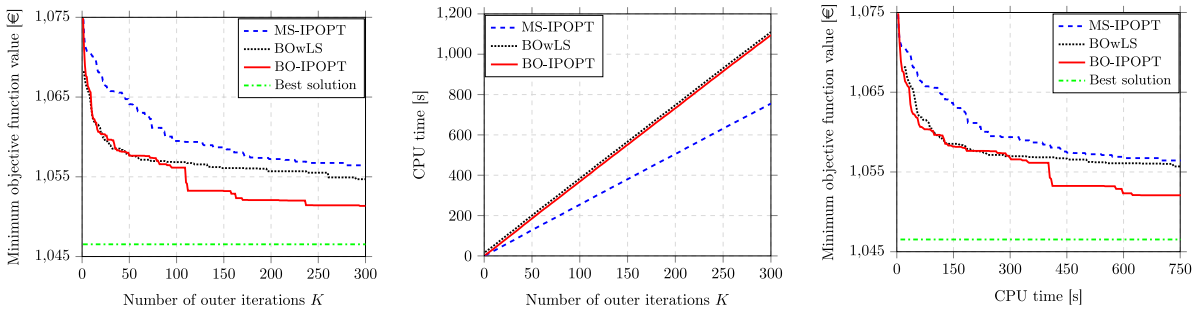


**Fig. 5.** Optimization results for the use case renewable steam generation (Eqs. (9)–(25)): comparison of the averaged minimum objective function value (**left**), the averaged CPU time (**middle**) and the respective solver performance (**right**) over 20 trials using MS-IPOPT, BOwLS and BO-IPOPT in relation to the number of outer iterations $K$. The best known solution (computed by 10,000 IPOPT iterations) is also visualized as an estimate of the global optimum.

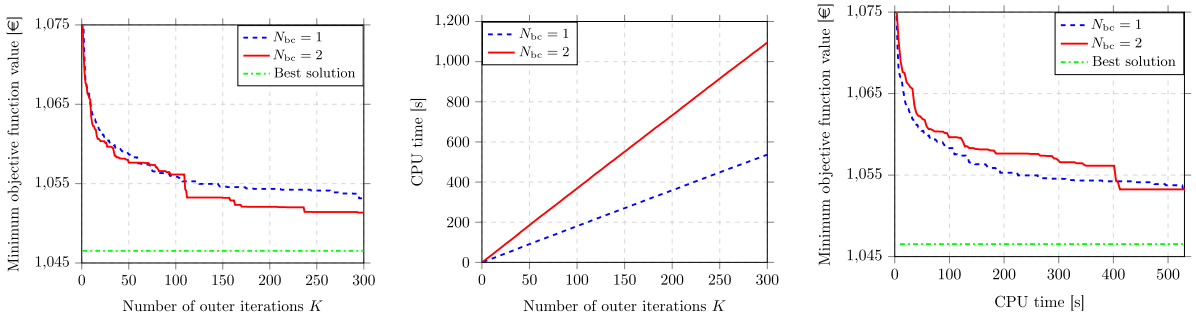


**Fig. 6.** Optimization results for the use case renewable steam generation (Eqs. (9)–(25)): comparison of the averaged minimum objective function value (**left**), the averaged CPU time (**middle**) and the respective solver performance (**right**) over 20 trials using BO-IPOPT with $N_{bc} = 1$ and $N_{bc} = 2$ in relation to the number of outer iterations $K$. The best known solution (computed by 10,000 IPOPT iterations) is also visualized as an estimate of the global optimum.

BO-IPOPT) for the number of initialization points ($N_0 = 10$) considered. Therefore, problems of scalability are not observed for the size of the sampling at hand. Moreover, the success of the hybrid approach and thus of the BO part implies the success of the GPR even for the overall limited number of samples involved. Given the large dimensions of both problems, these results are particularly promising, as it is well known that BO faces the curse of dimensionality (the exponential increase of samples required to identify a model as the dimensions increase). A more comprehensive investigation using a wider range of benchmark problems is necessary to determine whether the proposed hybrid formalism can effectively overcome this limitation, and future work will explore this aspect in greater detail. It is noteworthy that several advanced strategies have been developed to mitigate the curse in GPR, including variable selection, additive decomposition, sparse approximations, and low-dimensional embeddings (see [42,43]). These techniques can be readily incorporated into our formalism, potentially leading to further improvements in high-dimensional settings.

The choice of hyperparameters clearly influences the performance of both hybrid methods. Among these, the number of candidates $N_{bc}$ updated via the local solver appears a critical parameter governing the optimization convergence rate as well as the computational effort. Fig. 6 shows the impact of this hyperparameter in the optimization performance. A higher number of $N_{bc}$ leads to a significant increase in the computational cost of BO-IPOPT with a simultaneously minor improvement in its convergence rate. For a better comparison, we also display the solver performance on the right of Fig. 6, showing a slightly better performance of our hybrid approach with $N_{bc} = 1$ compared to $N_{bc} = 2$. Future work will analyze the impact of various hyperparameters on the optimization performance of the proposed hybrid method in more depth. Moreover, it is of interest to investigate in future studies the influence of the problem setup (e.g. initial and boundary conditions, time step $\Delta t$) on the performance of BO-IPOPT.

## 5. Conclusion and future work

This work presents a novel method called BO-IPOPT to determine the optimal operation of energy conversion systems. BO-IPOPT effectively combines Bayesian optimization and Interior Point OPTimizer, allowing profit from the global exploration of the surrogate modeling in the BO with the quasi-Newton local convergence of IPOPT. In BO-IPOPT, BO is made aware of the constraints via penalty terms in the objective function, while IPOPT is naturally aware of the constraints during its updates.

We demonstrate the proposed method in a challenging constrained test case and the optimization of a complex industrial energy conversion system for renewable steam generation. The optimization performance is compared to the classical MS-IPOPT with random initialization and the recently introduced hybrid BOwLS.

In both cases, the proposed hybrid method clearly outperforms BOwLS and MS-IPOPT in convergence rate. On the one hand, a substantial improvement in objective function value of up to 50% at the same CPU time is achieved compared to the other two optimizers in the constrained Ackley function. On the other hand, the complexity of the real-world use case leads the optimizer to a better objective function value of ≈1% in comparison to the other two competing methods. The performance gain with respect to BOwLS appears to be linked to the different initialization criteria and the use of surrogate models that approximate the original function.

Regarding the computational cost, the overhead of the hybrid methods is mostly linked to the construction of the GPR at each iteration. Nevertheless, this is found to be negligible in both cases, showing a linear trend in the CPU time as the number of outer iterations increases. The major increase in the CPU time for both hybrid methods is primarily due to the additional number of best candidates leading directly to more IPOPT evaluations. However, these can be carried out independently and could be easily parallelized. This will be the subject of future work.

Future work will proceed along three axes. First, it is of interest to analyze the performance of BO-IPOPT in a larger number of benchmark problems with varying dimension and complexity, considering in the comparison with state-of-the-art methods also other BO variants from the literature except for MS-IPOPT and BOwLS. Second, we seek to improve the handling of the constraints in the BO portion of the algorithm. Finally, we aim to address hyperparameter optimization for BO-IPOPT, and consider the sensitivity of its performance to identify the optimal balance between accuracy and computational cost.

## CRediT authorship contribution statement

**Loukas Kyriakidis:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Miguel Alfonso Mendez:** Writing – review & editing, Supervision, Resources, Methodology. **Martin Bähr:** Writing – review & editing, Visualization, Supervision, Software, Resources, Conceptualization.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The authors declare no competing interests. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

## Data availability

Data will be made available on request.

## References

[1] Mei F, Zhang J, Lu J, Lu J, Jiang Y, Gu J, Yu K, Gan L. Stochastic optimal operation model for a distributed integrated energy system based on multiple-scenario simulations. Energy 2021;219:1–13. http://dx.doi.org/10.1016/j.energy.2020.119629.

[2] Chen X, Wang C, Wu Q, Dong X, Yang M, He S, Liang J. Optimal operation of integrated energy system considering dynamic heat-gas characteristics and uncertain wind power. Energy 2020;198:1–17. http://dx.doi.org/10.1016/j.energy.2020.117270.

[3] Xu Y, Yan C, Liu H, Wang J, Yang Z, Jiang Y. Smart energy systems: A critical review on design and operation optimization. Sustainable Cities Soc 2020;62:1–19. http://dx.doi.org/10.1016/j.scs.2020.102369.

[4] Lu Y, Wang S, Sun Y, Yan C. Optimal scheduling of buildings with energy generation and thermal energy storage under dynamic electricity pricing using mixed-integer nonlinear programming. Appl Energy 2015;147:49–58. http://dx.doi.org/10.1016/j.apenergy.2015.02.060.

[5] Sass S, Mitsos A. Optimal operation of dynamic (energy) systems: When are quasi-steady models adequate? Comput Chem Eng 2019;124:133–9. http://dx.doi.org/10.1016/j.compchemeng.2019.02.011.

[6] Maleki A, Rosen MA, Pourfayaz F. Optimal operation of a grid-connected hybrid renewable energy system for residential applications. Sustainability 2017;9:1–20. http://dx.doi.org/10.3390/su9081314.

[7] Wei S, Gao X, Zhang Y, Li Y, Shen J, Li Z. An improved stochastic model predictive control operation strategy of integrated energy system based on a single-layer multi-timescale framework. Energy 2021;235:1–13. http://dx.doi.org/10.1016/j.energy.2021.121320.

[8] Parisio A, Rikos E, Tzamalis G, Glielmo L. Use of model predictive control for experimental microgrid optimization. Appl Energy 2014;115:37–46. http://dx.doi.org/10.1016/j.apenergy.2013.10.027.

[9] Palma-Behnke R, Benavides C, Lanas F, Severino B, Reyes L, Llanos J, Śaez D. A microgrid energy management system based on the rolling horizon strategy. IEEE Trans Smart Grid 2013;4:996–1006. http://dx.doi.org/10.1109/TSG.2012.2231440.

[10] Ma D, Zhang L, Sun B. An interval scheduling method for the CCHP system containing renewable energy sources based on model predictive control. Energy 2021;236:1–16. http://dx.doi.org/10.1016/j.energy.2021.121418.

[11] Nocedal J, Wright SJ. Numerical optimization. Springer series in operations research and financial engineering, vol. 2, New York: Springer New York; 2006, http://dx.doi.org/10.1007/978-0-387-40065-5.

[12] Hendrix EM, Tóth BG. Introduction to nonlinear and global optimization. Springer optimization and its applications, vol. 1, New York: Springer New York; 2010, http://dx.doi.org/10.1007/978-0-387-88670-1.

[13] Martí R, Lozano JA, Mendiburu A, Hernando L. Multi-start methods. In: Martí R, Pardalos PM, Resende MGC, editors. Handbook of heuristics. Cham: Springer International Publishing; 2018, p. 155–75. http://dx.doi.org/10.1007/978-3-319-07124-4_1.

[14] Pardalos P, Tawarmalani M, Sahinidis NV. Convexification and global optimization in continuous and mixed-integer nonlinear programming. Vol. 65, Boston, MA: Springer US; 2002, http://dx.doi.org/10.1007/978-1-4757-3532-1.

[15] Parisio A, Rikos E, Glielmo L. A model predictive control approach to microgrid operation optimization. IEEE Trans Control Syst Technol 2014;22:1813–27. http://dx.doi.org/10.1109/TCST.2013.2295737.

[16] Bischi A, Taccari L, Martelli E, Amaldi E, Manzolini G, Silva P, Campanari S, Macchi E. A detailed MILP optimization model for combined cooling, heat and power system operation planning. Energy 2014;74:12–26. http://dx.doi.org/10.1016/j.energy.2014.02.042.

[17] Long Q, Wu C. A hybrid method combining genetic algorithm and hooke-jeeves method for constrained global optimization. J Ind Manag Optim 2014;10(4):1279–96. http://dx.doi.org/10.3934/jimo.2014.10.1279.

[18] Wang Y, Zhang Y, Zhang C, Zhou J, Hu D, Yi F, Fan Z, Zeng T. Genetic algorithm-based fuzzy optimization of energy management strategy for fuel cell vehicles considering driving cycles recognition. Energy 2023;263:1–13. http://dx.doi.org/10.1016/j.energy.2022.126112.

[19] Zhang W, Maleki A, Rosen MA, Liu J. Optimization with a simulated annealing algorithm of a hybrid system for renewable energy including battery and hydrogen storage. Energy 2018;163:191–207. http://dx.doi.org/10.1016/j.energy.2018.08.112.

[20] Chaduvula H, Das D. Analysis of microgrid configuration with optimal power injection from grid using point estimate method embedded fuzzy-particle swarm optimization. Energy 2023;282:1–15. http://dx.doi.org/10.1016/j.energy.2023.128909.

[21] Jeong S, Obayashi S. Efficient global optimization (EGO) for multi-objective problem and data mining. In: 2005 IEEE congress on evolutionary computation. Vol. 3, 2005, p. 2138–45. http://dx.doi.org/10.1109/CEC.2005.1554959.

[22] Lan G, Tomczak JM, Roijers DM, Eiben A. Time efficiency in optimization with a Bayesian-evolutionary algorithm. Swarm Evol Comput 2022;69:1–14. http://dx.doi.org/10.1016/j.swevo.2021.100970.

[23] Berk J, Nguyen V, Gupta S, Rana S, Venkatesh S. Exploration enhanced expected improvement for Bayesian optimization. In: Berlingerio M, Bonchi F, Gärtner T, Hurley N, Ifrim G, editors. Machine learning and knowledge discovery in databases. Cham: Springer International Publishing; 2019, p. 621–37. http://dx.doi.org/10.1007/978-3-030-10928-8_37.

[24] Brochu E, Cora VM, de Freitas N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. 2010, http://dx.doi.org/10.48550/arXiv.1012.2599, abs/1012.2599.

[25] Hedar A-R, Fukushima M. Hybrid simulated annealing and direct search method for nonlinear unconstrained global optimization. Optim Methods Softw 2002;17:891–912. http://dx.doi.org/10.1080/1055678021000030084.

[26] Kelner V, Capitanescu F, Léonard O, Wehenkel L. A hybrid optimization technique coupling an evolutionary and a local search algorithm. J Comput Appl Math 2008;215:448–56. http://dx.doi.org/10.1016/j.cam.2006.03.048.

[27] Cherki I, Chaker A, Djidar Z, Khalfallah N, Benzergua F. A sequential hybridization of genetic algorithm and particle swarm optimization for the optimal reactive power flow. Sustainability 2019;11(14):1–12. http://dx.doi.org/10.3390/su11143862.

[28] Luo J, Chen Z, Zheng Y. A gradient-based method assisted by surrogate model for robust optimization of turbomachinery blades. Chin J Aeronaut 2022;35:1–7. http://dx.doi.org/10.1016/j.cja.2021.07.019.

[29] Zhang D, Zhang B, Wang Z, Zhu X. An efficient surrogate-based optimization method for BWBUG based on multifidelity model and geometric constraint gradients. Math Probl Eng 2021;2021:1–13. http://dx.doi.org/10.1155/2021/6939863.

[30] Gao Y, Yu T, Li J. Bayesian optimization with local search. In: Nicosia G, Ojha V, La Malfa E, Jansen G, Sciacca V, Pardalos P, Giuffrida G, Umeton R, editors. Machine learning, optimization, and data science. Cham: Springer International Publishing; 2020, p. 350–61. http://dx.doi.org/10.1007/978-3-030-64580-9_30.

[31] Wächter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math Program 2006;106:25–57. http://dx.doi.org/10.1007/s10107-004-0559-y.

[32] COIN-OR Foundation I. Interior Point Optimizer (IPOPT). https://github.com/coin-or/Ipopt.

[33] Rasmussen CE, Williams CKI. Gaussian processes for machine learning. The MIT Press; 2005, http://dx.doi.org/10.7551/mitpress/3206.001.0001.

[34] Jiang J, Yu Y, Min H, Cao Q, Sun W, Zhang Z, Luo C. Trip-level energy consumption prediction model for electric bus combining Markov-based speed profile generation and Gaussian processing regression. Energy 2023;263:125866. http://dx.doi.org/10.1016/j.energy.2022.125866.

[35] Picheny V, Gramacy RB, Wild S, Digabel SL. Bayesian optimization under mixed constraints with a slack-variable augmented Lagrangian. In: Proceedings of the 30th international conference on NIPS. 2016, p. 1443–51, https://papers.nips.cc/paper_files/paper/2016/hash/31839b036f63806cba3f47b93af8ccb5-Abstract.html.

[36] Gramacy RB. Surrogates: Gaussian process modeling, design and optimization for the applied sciences. Boca Raton, Florida: Chapman Hall/CRC; 2020, http://bobby.gramacy.com/surrogates/.

[37] Eriksson D, Poloczek M. Scalable constrained Bayesian optimization. In: Banerjee A, Fukumizu K, editors. Proceedings of the 24th international conference on artificial intelligence and statistics. Proceedings of machine learning research, vol. 130, San Diego, California, USA: PMLR; 2021, p. 730–8, https://proceedings.mlr.press/v130/eriksson21a.html.

[38] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Édouard Duchesnay. Scikit-learn: Machine learning in python. J Mach Learn Res 2011;12(85):2825–30, http://jmlr.org/papers/v12/pedregosa11a.html.

[39] Hart W, Watson J, Woodruff D. Pyomo: Modeling and Solving Mathematical Programs in Python, 3 (3) (2011) 219–260. http://dx.doi.org/10.1007/s12532-011-0026-8.

[40] Walden JV, Bähr M, Glade A, Gollasch J, Tran AP, Lorenz T. Nonlinear operational optimization of an industrial power-to-heat system with a high temperature heat pump, a thermal energy storage and wind energy. Appl Energy 2023;344:121247. http://dx.doi.org/10.1016/j.apenergy.2023.121247.

[41] EBSILON®Professional. https://help.ebsilon.com/EN/EBSILON_Professional_Documentation.html.

[42] Binois M, Wycoff N. A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization. ACM Trans Evol Learn Optim 2022;2(2):1–26. http://dx.doi.org/10.1145/3545611.

[43] Tripathy R, Bilionis I, Gonzalez M. Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation. J Comput Phys 2016;321:191–223. http://dx.doi.org/10.1016/j.jcp.2016.05.039.