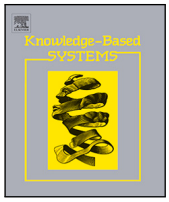
Knowledge-Based Systems 299 (2024) 111959



Contents lists available at ScienceDirect

Knowledge-Based Systems

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





NM2-BO: Non-Myopic Multifidelity Bayesian Optimization

Francesco Di Fiore [a,*], Laura Mainini [b,c,a]

[a] Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Corso Duca degli Abruzzi 24, Turin, 10129, Italy
[b] Department of Aeronautics, Imperial College London, South Kensington Campus, London, SW7 2AZ, United Kingdom
[c] Brahmal Vasudevan Institute for Sustainable Aviation, South Kensington Campus, London, SW7 2AZ, United Kingdom

ARTICLE INFO

Keywords:
Non-myopic multifidelity method
Multifidelity Bayesian optimization
Two-step lookahead multifidelity acquisition function
Multifidelity Gaussian process
Engineering optimization
Simulation-based optimization
Design optimization

ABSTRACT

Bayesian optimization is a popular framework for the optimization of black box functions. Multifidelity methods allow to accelerate Bayesian optimization by exploiting low-fidelity representations of expensive objective functions. Popular multifidelity Bayesian strategies rely on sampling policies that account for the immediate reward obtained evaluating the objective function for a specific input, precluding greater informative gains that might be obtained looking ahead more steps. This paper proposes a Non-Myopic Multifidelity Bayesian Optimization framework (NM2-BO) to grasp the long-term reward from future steps of the optimization. Our computational strategy comes with a two-step lookahead multifidelity acquisition function that maximizes the cumulative reward obtained measuring the improvement of the solution over two steps ahead. Our NM2-BO algorithm is demonstrated for a large set of benchmark problems to stress test across a broad spectrum of mathematical properties, and a physics-based application relevant for the engineering community. In all the cases, we observe that the proposed algorithm outperforms standard MFBO frameworks.

1. Introduction

Solving an optimization problem is a central task in disciplines ranging from engineering to operational research, to computer science and economics. Optimization problems are commonly formalized as the minimization of an objective function by iteratively choosing an input from a feasible set of variables. In many applications the function to optimize does not have a closed-form expression and its evaluation can be computationally expensive.

Bayesian optimization (BO) is a widely used strategy to solve optimization problems involving black-box objective functions that are expensive to compute [1,2]. BO uses a probabilistic surrogate model of the objective function to compute an acquisition function that is repeatedly maximized to select a new input closer to the optimum. Then, the objective function is observed considering this new training sample, and is used to update the surrogate model improving the accuracy of the prediction. BO requires evaluating the objective function several times to assess the solution of the optimization problem. This may result in a prohibitively computational expense for real-world applications requiring the use of lab-scale experiments or time consuming computer-based models, and where a limited number of evaluations of the objective are allowed to compute the optimization.

In practice, many applications allow to use alternative representations at different levels of fidelity to model the objective function, where the lower fidelity levels tend to be less time consuming but produces less accurate results. Multifidelity methods target the efficient combination of multiple sources of information, using low-fidelity analysis to systematically reduce the number of high-fidelity evaluations through a continuous trade-off between cost and accuracy [3–7]. Multifidelity Bayesian optimization (MFBO) combines the function outputs at different levels of fidelity into a single probabilistic model, and uses these information to compute a multifidelity acquisition function that allows to jointly select the new input location and the associated level of fidelity to query [8–12]. Most widely-used MFBO algorithms are based on greedy approaches, where the multifidelity acquisition function measures only the immediate reward of the input selected at each step of the optimization [13–17]. However, the quantification of the long-term reward obtained at the future steps may lead to greater improvements in the search of the optimum, enhancing the computational efficiency of the predictive strategy.

This motivates the interest for non-myopic acquisition functions that are capable to grasp the informative gains obtainable by looking ahead into future steps. In most existing non-myopic methods, the location of the new input is determined through a partially observable Markov decision process (POMDP) [18–20]. However, the solution of the POMDP is computationally intense, limiting this approach to small-scale optimization problems [21]. Several works proposed approximations of the POMDP to formulate two-step lookahead acquisition functions capable

* Corresponding author.
E-mail address: francesco.difiore@polito.it (F. Di Fiore).

https://doi.org/10.1016/j.knosys.2024.111959
Received 6 February 2023; Received in revised form 29 April 2024; Accepted 13 May 2024
Available online 23 May 2024
0950-7051/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

to enhance the performance of BO, while containing the computational expense that would arise looking more steps ahead [18–20,22,23]. However, these non-myopic formulations are conceived for a single fidelity Bayesian framework, which can be prohibitive for objective functions characterized by high computational demand.

In a previous work, we addressed a specific aerodynamic design optimization problem under multiple transitions of the fluid regime through a two-step lookahead domain-aware multifidelity method [24]; the specific formalization of this acquisition function seeks to capture the appropriate aerodynamic model to query introducing awareness about the evolution of the fluid regime with the flight conditions. Accordingly, the proposed methodology is tailored and highly problem dependent without possible extension to a broader class of optimization problems. However, the encouraging results for aerodynamic design catalyze our attention to the opportunity to generalize the non-myopic multifidelity strategy for a broader applicability to optimization problems of different nature in science and engineering.

In this work, we propose a generalized strategy for Non-Myopic Multifidelity Bayesian Optimization (NM2-BO) to maximize the long-term reward as an improvement of the optimal solution. The NM2-BO algorithm is described and demonstrated as a general optimization method for multifidelity optimization problems without limiting the applicability of the formulation, and regardless the variety of mathematical properties that can be traced in physics-based optimization problems. Our framework is based on an original scheme for the maximization of the multifidelity acquisition function based on a two-step lookahead strategy, that allows to predict the informative gains obtainable at future steps of the optimization. We will refer to that as the two-step lookahead multifidelity acquisition function. Specifically, our contributions are:

1. Formalize MFBO as a Dynamic Programming (DP) problem and define the best optimization policy that maximizes the cumulative reward obtained evaluating a specific input with a certain level of fidelity two steps ahead, quantifying its merit considering the long-term benefits.
2. Formulate the two-step lookahead multifidelity acquisition function as the solution of the DP problem encompassing intractable nested expectations and maximizations problems whose closed-form solution is not available. To overcome this issue, the Monte Carlo approach is used to provide a robust estimate of the two-step lookahead multifidelity acquisition function. We also provide details about its integration in a multifidelity Bayesian settings to draw our non-myopic MFBO framework.
3. Illustrate and discuss the proposed NM2-BO algorithm in comparison with standard MFBO methods over a suite of benchmark problems recognized by the international community of multifidelity methods as stress-test problems [25], to assess and validate the performance of multifidelity algorithms over a variety of mathematical properties that can be observed individually or in multiple combinations in real-world applications. In addition, we investigate the capabilities of our NM2-BO for the physics-based shape optimization problem of a RAE-2822 airfoil with the objective of minimizing the drag coefficient subject to constraints on the aerodynamic performances and modified airfoil geometry [26].

The manuscript is organized as follows. Section 2 provides an overview of Bayesian optimization with Gaussian processes and its extension in a multifidelity scenario. Section 3 presents our two-step lookahead multifidelity acquisition function and its implementation in a multifidelity framework. Then, in Section 4 the NM2-BO algorithm is numerically investigated and compared to standard MFBO frameworks over a variety of benchmark problems and on the physics-based airfoil shape optimization application. Finally, Section 5 provides concluding remarks.

2. Problem formulation and background

In this paper, we consider a sequential decision problem in which we seek to minimize a black-box objective function $f : \chi \to \mathbb{R}$ from its observations for iteratively selected sample points $\boldsymbol{x} \in \chi$ in the input space χ. More formally, we can define our optimization problem as follows:

$$\min_{\boldsymbol{x}\in\chi} f(\boldsymbol{x}) \tag{1}$$

In real world applications, the evaluation of the objective function is frequently expensive, therefore the interest in computational strategies to minimize the number of explicit calls of the function.

2.1. Bayesian optimization via Gaussian processes

Bayesian optimization (BO) is a popular approach to solve the global optimization problem in Eq. (1) [1,2]. BO learns a probabilistic surrogate model to predict the values of the objective function across the input space and quantifies the uncertainty of this prediction. These information are used to compute an acquisition function that defines a policy to measure the improvement of the solution in different input locations, which usually encodes a trade-off between the exploration of the input space and the exploitation towards the optimum. BO selects the new input to query by maximizing the acquisition function, and updates the surrogate model with the observation of the objective function at the new point.

Commonly, BO relies on a Gaussian Process (GP) as the surrogate model to approximate the objective function [27]. The GP distribution $GP\left(\mu(\boldsymbol{x}), \kappa(\boldsymbol{x}, \boldsymbol{x}')\right)$ is a joint Gaussian distribution over any finite set of N random input points $\boldsymbol{x}_N = \{\boldsymbol{x}_n\}_{n=1}^N$ completely specified by its mean $\mu(\boldsymbol{x}) = \mathbb{E}[f(\boldsymbol{x})]$ and covariance (also referred as kernel) function $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}\left[(f(\boldsymbol{x}) - \mu(\boldsymbol{x}))(f(\boldsymbol{x}') - \mu(\boldsymbol{x}'))\right]$. We assume that $f \sim GP\left(0, \kappa(\boldsymbol{x}, \boldsymbol{x}')\right)$ is the prior belief about the objective and that we have collected observations $\boldsymbol{y}_N = \{y_n\}_{n=1}^N$, where $y_n \sim \mathcal{N}(f(\boldsymbol{x}_n), \sigma_\epsilon(\boldsymbol{x}_n))$ and σ_ϵ is the standard deviation of the normally distributed noise. The formalization of the collected observations including a Gaussian noise variance permits to define the relationship between the function values $f(\boldsymbol{x})$ and the observed responses y as the likelihood function $p(y_n|f(\boldsymbol{x}_n)) = \mathcal{N}(y_n|f(\boldsymbol{x}_n), \sigma_\epsilon^2 \boldsymbol{I})$, where $\boldsymbol{I}$ is the identity matrix. In deterministic analysis, this noise element – also called nugget term – permits a more reliable evaluation of the maximum likelihood function and estimate of the hyperparameters of the Gaussian process surrogate model [27,28] given a contained condition number of the covariance matrix, while in non-deterministic analysis it is necessary to model measurement errors. According to the Bayesian inference principle, the Gaussian process prediction μ and associated uncertainty σ^2 is computed through regression: the prior belief about the objective function $p(f(\boldsymbol{x}_N))$ is combined with the likelihood function $p(\boldsymbol{y}_N|f(\boldsymbol{x}_N))$ to compute the posterior distribution $p(f(\boldsymbol{x}_N)|\boldsymbol{y}_N) \propto p(\boldsymbol{y}_N|f(\boldsymbol{x}_N))p(f(\boldsymbol{x}_N))$, representing the updated surrogate model of the objective function. Assuming the prior of the objective as a GP: $f \sim GP(0, \kappa(\boldsymbol{x}, \boldsymbol{x}'))$ with zero mean function $\mu(\boldsymbol{x}) = 0$, the posterior distribution is a Gaussian process completely specified by its mean μ and variance σ^2

$$\mu(\boldsymbol{x}) = \kappa_n(\boldsymbol{x})^T \left(\boldsymbol{K} + \sigma_\epsilon \boldsymbol{I}\right)^{-1} \boldsymbol{y}_N \tag{2}$$

$$\sigma^2(\boldsymbol{x}) = \kappa(\boldsymbol{x}, \boldsymbol{x}) - \kappa_n(\boldsymbol{x})^T \left(\boldsymbol{K} + \sigma_\epsilon \boldsymbol{I}\right)^{-1} \kappa_n(\boldsymbol{x}) \tag{3}$$

where κ_n is defined as $\kappa_n(\boldsymbol{x}) \doteq \left(\kappa(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, \kappa(\boldsymbol{x}, \boldsymbol{x}_N)\right)$.

The acquisition function $U : \chi \to \mathbb{R}^+$ under the GP prior depends on the previous observations of the objective and determines which point in χ should be evaluated in the next optimization step via maximization $\boldsymbol{x}_{new} \in \max_{\boldsymbol{x}\in\chi} U(\boldsymbol{x})$. There is a variety of acquisition functions commonly adopted in literature, such as the expected improvement (EI) [29], upper confidence bound (UCB) [30], entropy search

(ES) [31], max-value entropy search [32] and predictive entropy search (PES) [33]. In this work we focus on GP based Bayesian optimization with the EI acquisition function, given its simple implementation and reasonable performances in practice:

$$U_{EI}(\boldsymbol{x}) = \begin{cases} \left(\mu(\boldsymbol{x}) - f(\boldsymbol{x}^+)\right)\Phi(\gamma(\boldsymbol{x})) + \sigma(\boldsymbol{x})\phi(\gamma(\boldsymbol{x})) & \text{if} \quad \sigma(\boldsymbol{x}) > 0 \\ 0 & \text{if} \quad \sigma(\boldsymbol{x}) = 0 \end{cases} \tag{4}$$

where $\boldsymbol{x}^+$ is the current location of the best value of the objective function, Φ is the cumulative distribution function and ϕ is the probability density function of the standard normal distribution, and $\gamma(\boldsymbol{x}) = \left(\mu(\boldsymbol{x}) - f(\boldsymbol{x}^+)\right)/\sigma(\boldsymbol{x})$ is the standardized improvement.

2.2. Multifidelity Bayesian optimization

Many applications allow multifidelity representations of the objective function $\{f^{(1)}, \dots, f^{(L)}\}$, where the higher the level of fidelity l the more accurate and costly the evaluation of $f^{(l)}(\boldsymbol{x})$. We indicate with L the highest-fidelity level, that is considered as the ground truth source of information about the objective function. Multifidelity Bayesian optimization learns a single probabilistic model combining low-fidelity and high-fidelity observations, and chooses the location of the next input and the level of fidelity to query through a multifidelity acquisition function. Among a variety of emulators that can be used in the multifidelity scenario [34–36], we adopt the multifidelity Gaussian process given the successful application over a broad variety of methodologies and challenging optimization problems [16,37,38]. This surrogate is defined extending the Gaussian process to a multifidelity scenario through an autoregressive scheme [3]:

$$f^{(l)} = \varrho^{(l-1)} f^{(l-1)}(\boldsymbol{x}) + \nu^{(l)}(\boldsymbol{x}) \quad l = 2, \dots, L \tag{5}$$

where $\varrho^{(l-1)}$ is a constant scaling factor that tunes successive representations $f^{(l)}$ and $f^{(l-1)}$, and $\nu^{(l)}$ models the discrepancy between $f^{(l)}$ and the scaled model $\varrho^{(l-1)} f^{(l-1)}$. This discrepancy $\nu^{(l)}$ is assumed to be a Gaussian process characterized by mean function $\upsilon(\boldsymbol{x})^T \beta^{(l)}$ and covariance $\kappa^{(l)}\left(\boldsymbol{x}, \boldsymbol{x}'\right)$, where υ is the vector of regression functions and $\beta^{(l)}$ are the coefficients weighting these regression functions. In this work, we use the Gaussian correlation model as the covariance function:

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \varsigma^{2(l)} \exp\{-\sum_{m=1}^{M} \varpi_m^{(l)} (\boldsymbol{x}_m - \boldsymbol{x}'_m)^2\} \tag{6}$$

where $\varpi = (\varpi_1^{(l)}, \varpi_2^{(l)}, \dots, \varpi_M^{(l)})$ are the roughness parameters, and $\varsigma^{2(l)}$ is the process variance of the l-th level of fidelity.

Hence, we can obtain the posterior mean $\mu^{(l)}(\boldsymbol{x})$ and variance $\sigma^{2(l)}(\boldsymbol{x})$ of the objective through the covariance matrix $\boldsymbol{K}(i,j) = \left[\kappa\left(\left(\boldsymbol{x}_i, l_i\right), \left(\boldsymbol{x}_j, l_j\right)\right) + \delta_{ij}\sigma_\epsilon(\boldsymbol{x}_i)\right]$ and $\kappa_N^{(l)}(\boldsymbol{x}) = \left[\kappa\left(\left(\boldsymbol{x}_i, l_i\right), (\boldsymbol{x}, l)\right)\right]$:

$$\mu^{(l)}(\boldsymbol{x}) = \kappa_N^{(l)}(\boldsymbol{x})^T \boldsymbol{K}^{-1} \boldsymbol{y} \tag{7}$$

$$\sigma^{2(l)}(\boldsymbol{x}) = \kappa\left((\boldsymbol{x}, l), (\boldsymbol{x}, l)\right) - \kappa_N^{(l)}(\boldsymbol{x})^T \boldsymbol{K}^{-1} \kappa_N^{(l)}(\boldsymbol{x}) \tag{8}$$

where δ_{ij} is the Kronecker delta function. The hyperparameters $(\varrho, \beta, \varpi, \varsigma)$ of the multifidelity Gaussian process surrogate model are estimated by maximizing the likelihood function [39].

The multifidelity acquisition function is computed starting from the information of the multifidelity GP. Multifidelity acquisition functions include the multifidelity expected improvement (MFEI) [8], multifidelity predictive entropy Search [10], multifidelity probability of improvement [11], and multifidelity max-value entropy search [12].

We base our non-myopic multifidelity framework on the MFEI given its robust and effective performance over a broad variety of analytical and real-world optimization problems [40,41]:

$$U_{MFEI}(\boldsymbol{x}, l) = U_{EI}(\boldsymbol{x}) \alpha_1(\boldsymbol{x}, l) \alpha_2(\boldsymbol{x}, l) \alpha_3(\boldsymbol{x}, l) \tag{9}$$

where $U_{EI}(\boldsymbol{x})$ is the expected improvement in Eq. (4) evaluated at the highest level of fidelity. The utility functions α_1, α_2 and α_3 are defined as follows:

$$\alpha_1(\boldsymbol{x}, l) = corr\left[f^{(l)}(\boldsymbol{x}), f^{(L)}(\boldsymbol{x})\right] = \frac{\kappa((\boldsymbol{x}, l), (\boldsymbol{x}, L))}{\sqrt{\sigma^{2(l)}(\boldsymbol{x})\sigma^{2(L)}(\boldsymbol{x})}} \tag{10}$$

$$\alpha_2(\boldsymbol{x}, l) = 1 - \frac{\sigma_\epsilon}{\sqrt{\sigma^{2(l)}(\boldsymbol{x}) + \sigma_\epsilon^2}} \tag{11}$$

$$\alpha_3(l) = \frac{\lambda^{(L)}}{\lambda^{(l)}}. \tag{12}$$

$\alpha_1(\boldsymbol{x}, l)$ accounts for the decrease of the accuracy of the objective function evaluation as the adopted level of fidelity decreases; it is defined as the correlation between the evaluation of the objective function at the l-th fidelity and the high-fidelity evaluation. $\alpha_2(\boldsymbol{x}, l)$ considers the reduction of the uncertainty on the multifidelity GP prediction after a new evaluation of the objective function. $\alpha_3(l)$ accounts for the computational cost $\lambda^{(l)}$ associated with the level of fidelity of the objective representations: it is defined as the ratio between the cost per evaluation of the high-fidelity model and the cost of the l-th fidelity model. Typically, the computational cost $\lambda^{(l)}$ of the l-th level of fidelity is evaluated as the average CPU time required to solve the l-th model of the objective function. Accordingly, if $\alpha_1(\boldsymbol{x}, l)$ is excluded from the formulation, the algorithm would query only the low-fidelity model to reduce the computational cost of optimization; if $\alpha_2(\boldsymbol{x}, l)$ is not considered, the sampling procedure might concentrate resources in regions of the input space already explored; if $\alpha_3(l)$ is discarded, the algorithm would evaluate only the high-fidelity model to improve the accuracy of the results.

3. Non-myopic multifidelity Bayesian optimization

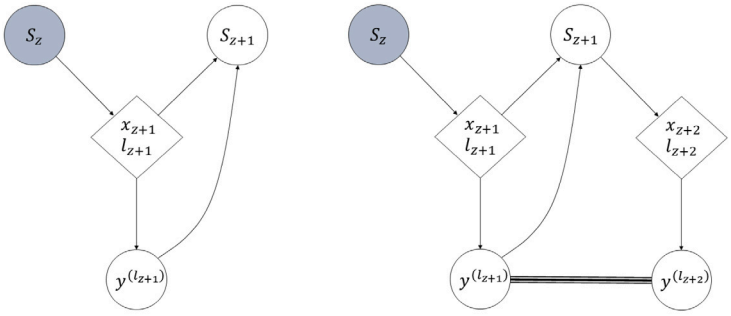
In this section, we formalize the MFBO as a dynamic programming problem and provide a general overview of the dynamic programming approach (Section 3.1). Accordingly, we define the two-step lookahead multifidelity acquisition function (Section 3.2) solving the specific MFBO problem. The solution of our acquisition function requires an approximation of its intractable formulation for which we adopt the Monte Carlo approach (Section 3.3). The lookahead strategy is then combined with a Bayesian approach for multifidelity optimization into our non-myopic MFBO scheme (Section 3.4).

3.1. Dynamic programming problem

This section overviews some of the key aspects of Markov decision processes and dynamic programming to address the optimal decision making process for dynamic systems subject to uncertainty.

Markov Decision Processes (MDPs) are discrete-time stochastic control processes that permit to model the sequential decision making process under uncertainty [42]. MDPs consist of three main ingredients: (i) Markov chains model the transitions to future states of the system given the initial state, (ii) a decision-making model implements an action at each sequential state transition, and (iii) an objective formalized as an utility function evaluates the benefits of any decision with a trade-off between obtaining the utility from the current state and altering the opportunities to obtain utility in the future. The goal of MDPs is to determine an optimal strategy as a set of decisions to satisfy the objectives over time.

Solving MDPs requires the definition of a procedure to perform statistical inference of the system behavior to predict future states, and to build a dynamic model to depict the transitions of the system from one state to the other after exploring every possible decision. Dynamic programming (DP) is an optimization strategy that permits to solve MDPs partitioning the problem in simpler sub-problems defined in a recursive way across several transitions [43]. DP addresses the decision making process through an optimal policy formulated as a sequence of rules that, regardless of the initial state and the decision, makes the



(a) Myopic MFBO (b) Non-Myopic MFBO

Fig. 1. Bayesian networks describing the myopic (Fig. 1(a)) and the non-myopic (Fig. 1(b)) multifidelity Bayesian scheme. The shaded nodes (S_z) are known, and the diamond nodes are the results of a decision. In the non-myopic scheme, the two y nodes are correlated under the multifidelity Gaussian process model.

remaining decisions optimal with regard to the state resulting from the first decision. Thus, the effects of the current decision contribute to both the current utility and to future utility through its effects on the future states of the system.

Let us consider a dynamic system fully characterized at each stage z by a state $s_z \in \mathcal{S}_z$, where $\mathcal{S}_z$ denotes a set of states that represent all the possible configurations of the system at each time step. The objective of the DP approach is the maximization of an utility function over a time horizon T which measures the reward obtained applying an action $c_z \in C_z$ to a certain state. The dynamic system evolves according to a transition model represented as a Markov process in which the future state s_{z+1} depends on the current state s_z and the action adopted c_z:

$$s_{z+1} = \mathcal{F}_z(s_z, c_z, d_z) \tag{13}$$

where $\mathcal{F}_z : \mathcal{S}_z \times C_z \times D_z$ is the dynamics of the system at the stage z, and $d_z \in D_z(s_z, c_z)$ is a random disturbance modeled as a random variable with probability distribution $P(\cdot|s_z, c_z)$.

DP seeks to construct an optimal policy that maximizes the chance of achieving the objective over a time horizon. A policy $\boldsymbol{\pi} = \{\pi_1, \dots, \pi_T\}$ is defined as a sequence of rules $\pi_z : \mathcal{S}_z \to C_z$ that maps each state s_z to the action c_z applied at that state. At each stage z, we can define the value function $J_z^{\boldsymbol{\pi}}(s_z)$ as the expected reward obtained using the generic policy $\boldsymbol{\pi}$ at the state s_z:

$$J_z^{\boldsymbol{\pi}}(s_z) = \mathbb{E}\left[\beta^{T+1} r_{T+1}(s_{T+1}) + \sum_{z=1}^{T} \beta^{z-1} r_z(s_z, \pi_z(s_z), d_z)\right] \tag{14}$$

where $r_z : \mathcal{S}_z \times C_z \times D_z$ is the stage reward function that quantifies the immediate reward of applying an action c_z to a state z_z under the disturbances d_z, and β is the discount factor representing the relative reward gained in the next period with reference to the current time. Eq. (14) includes two elements: the discounted terminal reward r_{T+1} which is a function of the state s_{T+1} acquired by the system after the last decision, and the sum of the discounted utilities from stage z to the horizon T.

Considering a set of admissible policies Π, the optimal policy $\boldsymbol{\pi}^*$ is determined assuming that the last action c_T taken at the horizon time T is optimal and conditions the actions at the previous stages. Thus, the optimal value function $J_z^{\boldsymbol{\pi}^*}$ is defined maximizing the long-term expected reward:

$$J_z^{\boldsymbol{\pi}^*}(s_z) = \max_{\boldsymbol{\pi} \in \Pi} J_z^{\boldsymbol{\pi}}(s_z) \tag{15}$$

Following the DP backward iteration algorithm, it is possible to define a recursive relationship that identifies for each state a value function $J_z(s_z)$ for the stage z, given that stage $J_{z+1}(s_{z+1})$ has already been solved:

$$J_z^{\boldsymbol{\pi}^*}(s_z) = \max_{c_z \in C_z} \mathbb{E}\left[r_z(s_z, c_z, d_z) + \beta J_{z+1}(\mathcal{F}_z(s_z, c_z, d_z))\right] \tag{16}$$

Then, the procedure is initialized by setting $J_{T+1}(s_{T+1}) = r_{T+1}(s_{T+1})$ and the solution is determined working backward from $z = T$ to $z = 1$.

3.2. Formulating the two-step lookahead multifidelity acquisition function

Multifidelity Bayesian optimization can be formulated as a dynamic programming (DP) problem. DP is conceived to address decision problems modeling the decision making process under uncertainty as a dynamic system, where its dynamics characterizes the sequence of decisions to achieve a specific goal [21,43]. Considering MFBO as a system governed by a discrete stage dynamics, the DP problem consists of (i) a dynamic system as the probabilistic surrogate model that synthesizes the multifidelity representations of the objective function, (ii) a system dynamics that describes how the surrogate model is updated when a new input is observed adopting a certain level of fidelity, and (iii) a goal that can be quantified with a measure of the two-step ahead improvement of the solution quality achieved by evaluating a specific input.

DP addresses the decision making process through a policy that determines a decision given the information about the dynamic system. In the following, we aim to construct an optimal policy for MFBO that allows to select the input point together with the level of fidelity to query predicting benefits two steps ahead. Fig. 1 summarizes the main differences between the decision making process realized by the standard myopic MFBO (Fig. 1(a)) and the proposed non-myopic MFBO paradigm (Fig. 1(b)). The myopic procedure makes a sub-optimal decision with respect to the current knowledge of the system without considering possible future scenarios. Conversely, the non-myopic multifidelity scheme does not ignore the future decision that will be made by the algorithm in the future step, and capitalizes from this information to maximize the utility of the decision.

Let us consider the generic stage z of MFBO, and define the input $\boldsymbol{x}$ and the observation of the objective function $y^{(l)}$ at the l-th level of fidelity, where the levels of fidelity can be in principle extended to any finite number. The system is fully characterized by a state s_z corresponding to the training dataset $S_z = \{\boldsymbol{x}_n, y^{(l_n)}, l_n\}_{n=1}^N$ of N observations, and an action $c_z = \{\boldsymbol{x}_{z+1}, l_{z+1}\}$ that activates the dynamics of the system. Thus, a policy π_z is defined as a function that maps the state s_z to an action $c_z = \pi_z(s_z)$.

Given the state and the action of the system, we introduce the disturbances to represent a simulated value of the objective function at $\{\boldsymbol{x}_{z+1}, l_{z+1}\}$, defined as a random variable $d_z^{(l)} \sim \mathcal{N}(\mu_z^{(l)}(\boldsymbol{x}_{z+1}), \sigma_z^{2(l)}(\boldsymbol{x}_{z+1}))$ characterized by the mean (Eq. (7)) and variance (Eq. (8)) of the multifidelity Gaussian process. At the new stage $z+1$, the system evolves to a new state s_{z+1} following its dynamics, corresponding to the augmented dataset $S_{z+1} = S_z \cup \{\boldsymbol{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}\}$:

$$S_{z+1} = \mathcal{F}(\boldsymbol{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z) \tag{17}$$

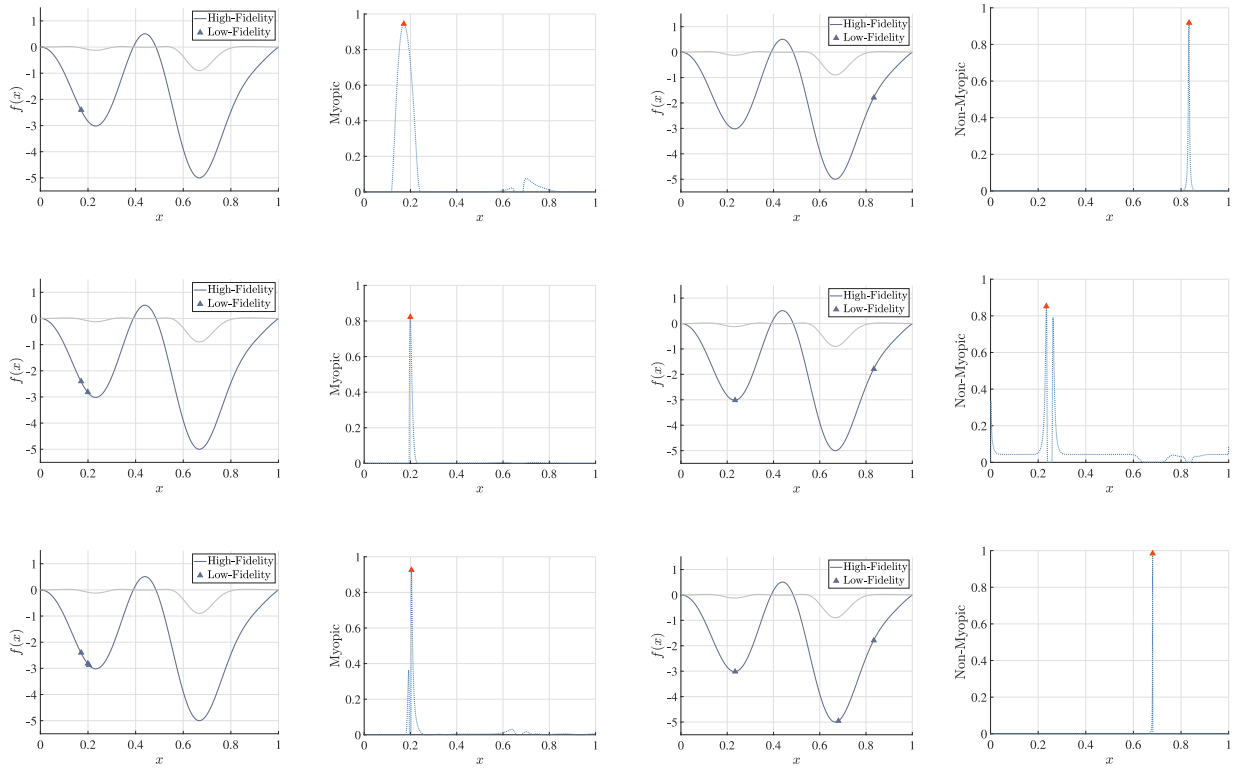


Fig. 2. Illustration of the searches conducted through a non-myopic MFBO and myopic MFBO for the toy example of the minimization of a 1-dimensional multifidelity synthetic function. Each row represents an iteration of the optimization procedure, and reports the high-fidelity queries together with the corresponding acquisition function for myopic MFBO (first two columns) and non-myopic MFBO (last two columns).

The disturbances d_{z+1} are then characterized using the multifidelity Gaussian process conditioned on S_{z+1}. We define a stage reward function to measure the benefits of applying the action c_z to a state s_z subject to the disturbances d_z. For the MFBO dynamic system, we formulate the stage reward function as the reduction of the objective function achieved at the stage $z+1$ with respect to z:

$$r_z(\boldsymbol{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z) = (f_z^{*(L)} - f_{z+1}^{(L)})^+ \tag{18}$$

where $f_z^{*(L)}$ is the minimum value of the objective function at z evaluated at the highest level of fidelity. Following the dynamic programming backward iteration scheme in Eq. (16), we formalize the recursive relationship between each stage of the MFBO process considering the last decision at the future stage $z+1$ – horizon – already solved. Accordingly, the two-step lookahead multifidelity acquisition function at a generic stage z can be formalized as a sequence of actions under a generic policy $\boldsymbol{\pi}$ whose benefits are quantified as the expected reward obtained adopting that policy at the stage z:

$$\begin{aligned} &U_z^{\boldsymbol{\pi}}(\boldsymbol{x}_{z+1}, l_{z+1}, S_z) = \\ &\quad \mathbb{E}[r_z(\boldsymbol{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z) + J_{z+1}(\mathcal{F}(\boldsymbol{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z))] \end{aligned} \tag{19}$$

where the expectation is taken with respect to the disturbances, $J_{z+1}(\mathcal{F}(\cdot))$ is the long-term expected reward, and $\mathbb{E}\left[r_z(\cdot)\right] = U_{MFEI}(\boldsymbol{x}_{z+1}, l_{z+1})$ is the multifidelity expected improvement (Eq. (9)).

Following the DP principle, the objective is to maximize the long term reward defining an optimal policy π_z^*. For the case of our two-step lookahead multifidelity acquisition function, the optimal policy is the one that identifies the optimal input variable and the associated level of fidelity to query at the second future step to minimize the cumulative expected loss. Thus, we define the long term reward J_{z+1} as the maximum of the multifidelity expected improvement conditioned on the training set S_{z+1}:

$$J_{z+1} = \max(U_{MFEI}(\boldsymbol{x}_{z+2}, l_{z+2})) \tag{20}$$

Combining Eq. (19) with Eq. (20), we formalize the two-step lookahead multifidelity acquisition function:

$$\begin{aligned} &U_z^{\boldsymbol{\pi}^*}(\boldsymbol{x}_{z+2}, l_{z+2}, S_{z+1}) = \\ &\quad U_{MFEI}(\boldsymbol{x}_{z+1}, l_{z+1}) + \mathbb{E}\left[\max(U_{MFEI}(\boldsymbol{x}_{z+2}, l_{z+2}))\right] \end{aligned} \tag{21}$$

We emphasize that $U_z^{\boldsymbol{\pi}^*}$ depends on the action taken at the two-step ahead iteration $\boldsymbol{x}_{z+2}, l_{z+2}$ and on the state of the MFBO system S_{z+1} after the first step through the fact that the result of the decision as the evaluation of the objective function $f^{(l_{z+2})}(\boldsymbol{x}_{z+2})$ is influenced by the mean function $\mu^{(l_{z+1})}$ and standard deviation function $\sigma^{2(l_{z+1})}$ of the multifidelity Gaussian process under S_{z+1} from which $U_{MFEI}(\boldsymbol{x}_{z+2}, l_{z+2})$ is computed.

Fig. 2 shows the results for a synthetic optimization problem comparing three high-fidelity iterations of our non-myopic multifidelity search with the standard myopic counterpart based on the multifidelity expected improvement. The myopic acquisition function concentrates the high-fidelity samples near a local optimum of the objective function, without exploring the domain with accurate evaluations. This provides a relatively small improvement of the global solution of the optimization problem: the algorithm might use several queries of the objective function to depart from the suboptimal valley and move towards the global solution. Conversely, the non-myopic methodology uses high-fidelity queries to significantly explore the domain during the first iterations. Those information are beneficial for future evaluations: the exploration provides awareness about the distribution of the objective function and its local optimum that is used to maximize the future gains and place the last sample near the actual global optimum — third iteration.

3.3. Estimating the two-step lookahead multifidelity acquisition function

The evaluation of the two-step lookahead multifidelity acquisition function (Eq. (21)) is intractable in practice for the MFBO setting given

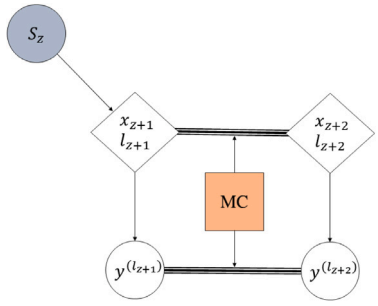


Fig. 3. Monte Carlo approximation of the relationship between the future stage $z+2$ and the stage $z+1$. The shaded nodes (S_z) are known, and the diamond nodes are the results of a decision. Comparing with Fig. 1(b), the nested maximization problems required for each pair $\boldsymbol{x}_{z+1}, l_{z+1}$ and $\boldsymbol{x}_{z+2}, l_{z+2}$ are now addressed approximating our acquisition function $U_z^{\pi^*}$ through the Monte Carlo strategy.

the demand for significant computational resources to compute the expectation operators and nested maximizations. In particular, the long term reward in Eq. (21) requires the knowledge of the MFBO dynamic system at the step $z+1$ in terms of mean $\mu_{z+1}^{(l)}$ and variance $\sigma_{z+1}^{(l)}$ of the multifidelity Gaussian process: the exact computation of those elements is computationally intensive given the extensive amount of all the possible actions $\{\boldsymbol{x}_{z+1}, l_{z+1}\}$ in the action space. To find a tractable approximation of Eq. (21), we adopt a Monte Carlo sampling to approximate $\mu_{z+1}^{(l)}$ and $\sigma_{z+1}^{(l)}$ via simulation. Fig. 3 illustrates the proposed Monte Carlo technique. Specifically, we use the reparameterization strategy proposed by Wilson et al. [44] to formulate the value of the objective function at the first step ahead with l-th level of fidelity:

$$f^{(l)}\left(\boldsymbol{x}_{z+1}\right) = \mu_z^{(l)} + \boldsymbol{C}_z^{(l)}\left(\boldsymbol{x}_{z+1}\right) Z \tag{22}$$

where $\boldsymbol{C}_z^{(l)}$ is the Cholesky decomposition of the covariance matrix $\boldsymbol{K}_z$, and Z is an independent standard normal random variable. Then, we use Eq. (22) to compute the mean and variance of the multifidelity GP at the step $z+1$ for the generic input $\boldsymbol{x}$:

$$\mu_{z+1}^{(l)}(\boldsymbol{x}) = \mu_z^{(l)}(\boldsymbol{x}) + \boldsymbol{H}_z^{(l)}(\boldsymbol{x})\, Z \tag{23}$$

$$\sigma_{z+1}^{(l)}(\boldsymbol{x}) = \sigma_z^{(l)}(\boldsymbol{x}) - \boldsymbol{H}_z^{(l)}(\boldsymbol{x})\, \boldsymbol{H}_z^{(l)}(\boldsymbol{x})^T \tag{24}$$

where $\boldsymbol{H}_z^{(l)}(\boldsymbol{x}) = \kappa_z^{(l)}(\boldsymbol{x})\, \boldsymbol{C}_z^{(l)-1}(\boldsymbol{x})$.

Eq. (23) and Eq. (24) are used to estimate the multifidelity expected improvement at the step $z+1$:

$$U_{MFEI}(\boldsymbol{x}_{z+2}, l_{z+2}) \sim \hat{U}_{MFEI}(\boldsymbol{x}_{z+2}, l_{z+2}, Z) \tag{25}$$

To approximate the expectation term of Eq. (21), we sample the random variable Z and compute $\mathbb{E}[\hat{U}_z^{\pi^*}(\boldsymbol{x}_{z+2}, l_{z+2}, S_{z+1}, Z)]$ averaging over many realizations of the two-step lookahead multifidelity acquisition $\hat{U}_z^{\pi^*}$ evaluated using Eq. (25).

Although the need of numerous evaluations of $\hat{U}_z^{\pi^*}$, we emphasize that our Monte Carlo approach requires the evaluation of Eq. (23) and Eq. (24) which are inexpensive to compute. Thus, the computational cost associated with the approximation of the two-step lookahead multifidelity acquisition function is negligible if compared with the cost of real-world optimization problems in science and engineering applications, where usually the evaluation of the objective function requires the query of expensive physics-based models.

3.4. NM2-BO algorithm

The Algorithm 1 and Fig. 4 detail the steps of our Non-Myopic Multifidelity Bayesian Optimization (NM2-BO) scheme. The optimization starts with the definition of the d-dimensional input space $\chi \in \mathbb{R}^d$ assembled through a Latin Hypercube design of experiments [45],

Algorithm 1 NM2-BO: Non-Myopic Multifidelity Bayesian Optimization

Input: Feasible set $\chi \in \mathbb{R}^d$, multifidelity objective function $f^{(l)}(\boldsymbol{x})$ and the multifidelity Gaussian process prior $GP(0, \kappa^{(l)}(\boldsymbol{x}, \boldsymbol{x}'))$
Output: $f^* = \min f(\boldsymbol{x})$

```
S_0 ← {x_n, y^(l_n), l_n}_{n=1}^{N_0} collect initial observations
μ_0^(l), σ_0^2(l) ← learn the initial multifidelity GP
i ← 1
repeat
    Load x_i and associated l_i
    Evaluate y^(l_i)(x_i)
    S_i ← S_{i-1} ∪ {x_i, y^(l_i), l_i}
    μ_i^(l), σ_i^2(l) ← update the multifidelity GP
    z ← i
    Compute U_MFEI(x_{z+1}, l_{z+1})
    for j ← 1, N_MC do
        Z_j ← N(0, 1)
        μ_{z+1}^(l), σ_{z+1}^2(l) ← estimate the multifidelity GP
        Compute Û_MFEI(x_{z+2}, l_{z+2}, Z_j)
    end for
    return Û_z^{π*} = {Û_{z_j}^{π*}}_{j=1}^{N_MC}
    U_i^{π*} = E[Û_z^{π*}]
    [x_{i+1}, l_{i+1}] ← max(U_i^{π*})
    i + 1 ← i
until B_i ≤ B_max
return x* that minimize f(x) over S_i
```

together with the objective function at different levels of fidelity $f^{(l)}$, and the multifidelity Gaussian process prior $GP(0, \kappa^{(l)}(\boldsymbol{x}, \boldsymbol{x}'))$. We draw an initial subset of feasible input points $\{\boldsymbol{x}_n\}_{n=1}^{N_0}$ and associated levels of fidelity $\{l_n\}_{n=1}^{N_0}$, where N_0 is the number of points sampled at the beginning of the optimization. Contextually, we collect observations of the objective function $\{y^{(l_n)}\}_{n=1}^{N_0}$ and determine the initial state $S_0 = \{\boldsymbol{x}_n, y^{(l_n)}, l_n\}_{n=1}^{N_0}$. The multifidelity GP prior together with the initial state S_0 induce the multifidelity Gaussian process posterior with mean $\mu^{(l)}$ (Eq. (7)) and variance $\sigma^{2(l)}$ (Eq. (8)), representing the first surrogate model of the objective function.

Let us now consider the generic iteration i of the optimization search. The NM2-BO algorithm selects the input $\boldsymbol{x}_i$ along with the associated level of fidelity l_i at the previous step $i-1$, and computes the value of the objective function $y^{(l_i)}(\boldsymbol{x}_i)$. This information is used to augment the state $S_i = S_{i-1} \cup \{\boldsymbol{x}_i, y^{(l_i)}, l_i\}$ and update the multifidelity Gaussian process conditioned on S_i.

At the same iteration i, we compute the two-step lookahead multifidelity acquisition function to determine the next input $\boldsymbol{x}_{i+1}$ to evaluate and the associated level of fidelity l_{i+1}. We indicate with $z = i$ the current step of the optimization and with $z+1$ and $z+2$ the first and second step ahead, respectively. The first term of the acquisition function (Eq. (21)) is computed on the current state $S_z = S_i$, while the second term is estimated using the Monte-Carlo approach discussed in Section 3.3. Considering the j-th Monte Carlo simulation, we sample independently a random variable Z_j normally distributed, and estimate the mean $\mu_{s+1}^{(l)}$ and variance $\sigma_{s+1}^{2(l)}$ of the multifidelity GP computing Eq. (23) and Eq. (24). Then, we evaluate the second term of the acquisition function through Eq. (25) and compute $\hat{U}_{z_j}^{\pi^*}$. The Monte Carlo algorithm iterates until a maximum number of simulations N_{MC} is reached and returns the realizations of the approximated acquisition function $\{\hat{U}_{z_j}^{\pi^*}\}_{j=1}^{N_{MC}}$. This allows to estimate the two-step multifidelity acquisition function U_i as the expectation taken over the Monte Carlo simulations.

Then, similarly to the multifidelity Bayesian optimization (Section 2.2), we maximize $U_i^{\pi^*}$ and determine the next input to evaluate

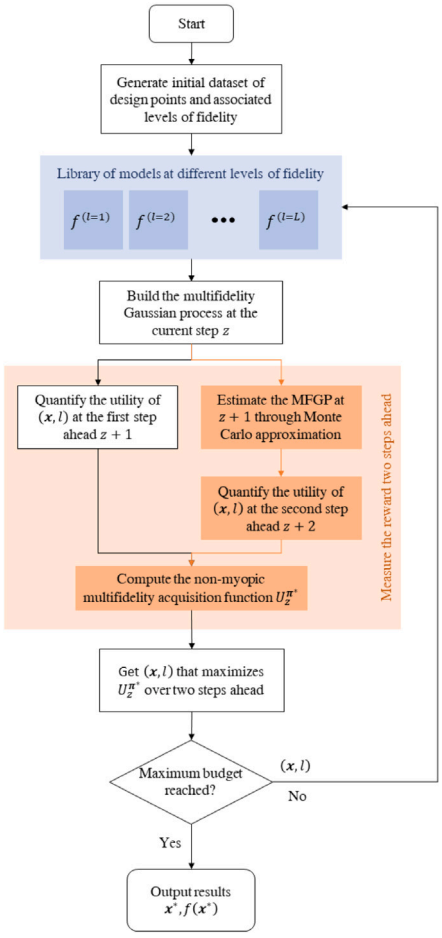


Fig. 4. Schematic representation of the NM2-BO framework. In orange are reported the distinguishing processes of NM2-BO with respect to a standard multifidelity Bayesian optimization framework.

$\boldsymbol{x}_{i+1}$ and the associated level of fidelity l_{i+1} to query. The NM2-BO algorithm is iterated until a maximum computational budget $B_i = B_{max}$ is reached, where B_i is the cumulative computational cost expended until iteration i.

4. Experiments

In this section, we illustrate and discuss the proposed NM2-BO algorithm in comparison with popular MFBO approaches over a set of well accepted benchmark problems specifically conceived to stress-test multifidelity methods (Section 4.1) and an aerodynamic shape optimization problem to minimize the drag coefficient of a RAE-2822 airfoil (Section 4.2). As competing algorithms, we implemented three standard MFBO frameworks based on multifidelity expected improvement (MFEI) [8], multifidelity probability of improvement (MFPI) [11], and multifidelity max-value entropy search (MFMES) [12]. For all the results presented in this section, we impose the same initial settings for both the NM2-BO and the standard MFBO algorithms. We define the number of points $n_0^{(l)}$ for each l-th fidelity level that characterize the initial dataset S_0, together with the maximum computational budget B_{max} quantified as the sum of the computational costs associated with observations of the objective function at the l-th level of fidelity. The initial observations of the objective function at different levels of fidelity are determined through a Latin Hypercube Sampling [46] and are used to compute the first surrogate model as illustrated in Section 3.4. We use the Gaussian kernel (Eq. (6)) for all the GP covariances, and optimize the hyperparameters of the kernel and mean functions of the GP via Maximum Likelihood Estimation [39].

4.1. Analytical benchmarks

We validate our methodology through the set of benchmark functions proposed by [25] including the Forrester function standard (Section 4.1.1) and discontinuous (Section 4.1.2), the Rastrigin function shifted and rotated (Section 4.1.3), the ALOS functions (Section 4.1.4), the Rosenbrock function (Section 4.1.5), the Paciorek function (Section 4.1.6), a mass spring system optimization problem (Section 4.1.7), and the Borehole function (Section 4.1.8). This set of functions is specifically selected by an international community to validate multifidelity methods through challenging mathematical characteristics of the objective function that are frequent in physics-based optimization problems, keeping contained the computational expense associated with its evaluation [25]. Table 1 summarizes the experimental setup for each test function.

4.1.1. Forrester function

The Forrester function [39] is a one-dimensional multimodal function well accepted as a benchmark test for multifidelity algorithms:

$$f^{(4)}(\boldsymbol{x}) = (6\boldsymbol{x} - 2)^2 \sin(12\boldsymbol{x} - 4) \tag{26}$$

The optimization problem (Eq. (1)) is defined over the domain $\chi = [0, 1]$ with the minimum of the objective located at $x^* = 0.7572$ and given by $f^* = -6.0207$. A total of three lower-fidelity levels are defined as follows:

$$f^{(3)}(\boldsymbol{x}) = (5.5\boldsymbol{x} - 2.5)^2 \sin(12\boldsymbol{x} - 4) \tag{27}$$

$$f^{(2)}(\boldsymbol{x}) = 0.75 f_1(\boldsymbol{x}) + 5(\boldsymbol{x} - 0.5) - 2 \tag{28}$$

$$f^{(1)}(\boldsymbol{x}) = 0.5 f_1(\boldsymbol{x}) + 10(\boldsymbol{x} - 0.5) - 5. \tag{29}$$

4.1.2. Jump forrester function

The jump Forrester function [47] problem introduces a discontinuous behavior in the formulation of the Forrester function in order to assess the performance of the algorithms in presence of objective functions that are not continuous in all the input space. The high-fidelity jump Forrester is defined as follows:

$$f^{(2)}(\boldsymbol{x}) = \begin{cases} (6\boldsymbol{x} - 2)^2 \sin(12\boldsymbol{x} - 4) & 0 \le \boldsymbol{x} \le 0.5 \\ (6\boldsymbol{x} - 2)^2 \sin(12\boldsymbol{x} - 4) + 10 & 0.5 < \boldsymbol{x} \le 1 \end{cases} \tag{30}$$

The input ranges is defined as $\chi = [0, 1]$ and the optimum equal to $f^* = -0.9863$ at $x^* = 0.1426$. The low-fidelity function is formulated as:

$$f^{(1)}(\boldsymbol{x}) = \begin{cases} 0.5 f^{(2)}(\boldsymbol{x}) + 10(\boldsymbol{x} - 0.5) - 5 & 0 \le \boldsymbol{x} \le 0.5 \\ 0.5 f^{(2)}(\boldsymbol{x}) + 10(\boldsymbol{x} - 0.5) - 2 & 0.5 < \boldsymbol{x} \le 1 \end{cases} \tag{31}$$

4.1.3. Rastrigin function shifted and rotated

The shifted and rotated Rastrigin function is a benchmark function used to simulate real world applications where the objective function could be highly multimodal:

$$f^{(l)}(\boldsymbol{x}) = v_2^2 + v_1^2 - \cos(10\pi v_2) - \cos(10\pi v_1) + 2 \tag{32}$$

where

$$\boldsymbol{v} = R(\vartheta)(\boldsymbol{x} - \boldsymbol{x}^*) \tag{33}$$

with

$$R(\vartheta) = \begin{bmatrix} \cos(\vartheta) & -\sin(\vartheta) \\ \sin(\vartheta) & \cos(\vartheta) \end{bmatrix} \tag{34}$$

where R is the rotation matrix, $\vartheta = 0.2$ is the rotation angle. The optimization problem (Eq. (1)) is defined in $\chi = [-0.1, 0.2]^2$ and the minimum is located at $\boldsymbol{x} = [0.1, 0.1]^2$ given by $f^* = 0$ To obtain the levels of fidelity, a resolution error [48] is defined as follows:

$$e(v, \varphi) = \sum_{i=1}^{2} \theta(\varphi) \cos^2(10\pi\Theta(\varphi) v_i + 0.5\pi\Theta(\varphi) + \pi) \tag{35}$$

Table 1
Summary of the experiments setup.

Function	$n_0^{(4)}$	$n_0^{(3)}$	$n_0^{(2)}$	$n_0^{(1)}$	B_{max}	$\lambda^{(4)}$	$\lambda^{(3)}$	$\lambda^{(2)}$	$\lambda^{(1)}$
Forrester	1	2	3	5	100	1	0.5	0.1	0.05
Jump Forrester	–	–	2	5	100	–	–	1	0.2
Rastrigin	–	10	20	30	200	–	1	0.065	0.0039
ALOS 1D	–	–	2	5	100	–	–	1	0.2
ALOS 2D	–	–	5	10	30	–	–	1	0.2
ALOS 3D	–	–	7	14	300	–	–	1	0.2
Rosenbrock 2D	–	–	5	10	200	–	–	1	0.5
Rosenbrock 5D	–	–	15	30	500	–	–	1	0.5
Rosenbrock 10D	–	–	50	250	1000	–	–	1	0.5
Paciorek	–	–	5	10	200	–	–	1	0.2
Mass Spring	–	–	4	10	400	–	–	1	1/60
Borehole	–	–	100	500	800	–	–	1	0.5

where $\Theta(\varphi) = 1 - 0.0001\varphi$. The different levels of fidelity are determied assigning $\varphi = 10000$ for the high-fidelity model ($l = 3$), $\varphi = 5000$ for the intermediate level of fidelity ($l = 2$), and $\varphi = 2500$ for the low fidelity model ($l = 1$).

4.1.4. ALOS functions

The Agglomeration of Locally Optimized Surrogate (ALOS) models are a set of functions conceived to evaluate the capability of multifidelity optimization in presence of objective functions characterized by oscillatory phenomena at different frequencies along the domain. The high-fidelity ALOS functions are defined for one and two dimensions [49], and extended to three dimensions as follows [25]:

$$f^{(2)}(x_1) = \sin\left[30(x_1 - 0.9)^4\right]\cos\left[2(x_1 - 0.9)\right] + (x_1 - 0.9)/2 \tag{36}$$

$$f^{(2)}(x_1, x_2) = f^{(2)}(x_1) + 2x_2^2 \sin(x_1 x_2) \tag{37}$$

$$f^{(2)}(x_1, x_2, x_3) = f^{(2)}(x_1, x_2) + 3x_3^3 \sin(x_1 x_2 x_3) \tag{38}$$

The domain of the input is defined as the interval $[0, 1]^d$ where $d = 1, 2, 3$ is the dimensionality of the problem. The optimum is $f^* = -0.62489$ located at $x^* = 0.2746$ for $d = 1$, $f^* = -0.56271$ corresponding to the input $x^* = [0, \dots, 0]^d$ for $d = 2, 3$. The low-fidelity ALOS functions are derived through bridge functions:

$$f^{(1)}(x_1) = (f^{(1)}(x_1) - 1 + x_1)/(1 + 0.25x_1) \tag{39}$$

$$f^{(1)}(x_1, x_2) = (f^{(1)}(x_1, x_2) - 2 + x_1 + x_2)/(5 + 0.25x_1 + 0.5x_2) \tag{40}$$

$$f^{(1)}(x_1, x_2, x_3) = (f^{(1)}(x_1, x_2, x_3) - 2 + x_1 + x_2 + x_3)/(5 + 0.25x_1 + 0.5x_2 - 0.75x_3) \tag{41}$$

4.1.5. Rosenbrock function

The Rosenbrock function [50] is a popular non-convex d-dimensional optimization benchmark problem defined in the interval $\chi = [-2, 2]^d$ as follows:

$$f^{(2)}(\boldsymbol{x}) = \sum_{i=1}^{d-1} 100(\boldsymbol{x}_{i+1} - \boldsymbol{x}_i^2)^2 + (1 - \boldsymbol{x}_i)^2 \tag{42}$$

The minimum of Eq. (42) is located at $x^* = [1, \dots, 1]^d$ corresponding to $f^* = 0$. We consider as the low fidelity level the linear mapping defined as follows:

$$f^{(1)}(\boldsymbol{x}) = \sum_{i=1}^{d-1} 50(\boldsymbol{x}_{i+1} - \boldsymbol{x}_i^2)^2 + (-2 - \boldsymbol{x}_i)^2 - \sum_{i=1}^{d} 0.5\boldsymbol{x}_i \tag{43}$$

In the experiments, we consider $d = 2, 5, 10$ to investigate the performances of the algorithms as the dimension of the design space increases.

4.1.6. Paciorek function with noise

This multifidelity problem considers the Paciorek function [51] with a noise term as the high-fidelity model, and uses a noisy low-fidelity representation that permits to simulate real-world applications where the measurements of the objective function are affected by noise [25]. The noisy high-fidelity Paciorek function is defined as:

$$f^{(2)}(\boldsymbol{x}) = \sin\left(\frac{1}{\boldsymbol{x}_1 \boldsymbol{x}_2}\right) + rand.normal(0, 0.0125) \tag{44}$$

where a uniformly distributed random noise parameter is included to simulate the noise. Across the input domain $\chi = [0.3, 1]^2$ the optimum is equal to $f^* = -1$ and located in a belt of equal-minimum input points. The noisy low-fidelity function is formulated as follows:

$$f^{(1)}(\boldsymbol{x}) = f^{(2)}(\boldsymbol{x}) - 2.25\cos\left(\frac{1}{\boldsymbol{x}_1 \boldsymbol{x}_2}\right) + rand.normal(0, 0.075) \tag{45}$$

4.1.7. Mass spring system

This benchmark problem consists of a coupled mass spring system, where two masses are connected by two springs. The system is characterized by a friction-less dynamics along an horizontal plane, where the masses are treated as point masses concentrated in their center of gravity and the elastic force of the springs is modeled through the Hooke's law. Assuming that the system is connected to an ideal wall through the first spring, its equations of motion are given by:

$$m_1 \ddot{p}_1(t) = -k_1 p_1(t) + k_2[p_2(t) - p_1(t)] \tag{46a}$$

$$m_2 \ddot{p}_2(t) = -k_2[p_2(t) - p_1(t)] - k_1 p_2(t) \tag{46b}$$

where m_1, m_2 are the masses, k_1, k_2 are the Hooke's constants of the springs and $p_1(t), p_2(t)$ are the positions of the masses as a function of time t. The fourth-order Runge–Kutta method is adopted to solve the system of first-order ODEs (46), setting the time-step size $dt = 0.01$ for the high-fidelity and $dt = 0.6$ for the low-fidelity analysis. The optimization task consists of determining the input variable $\boldsymbol{x} = [m_1, m_2, k_1, k_2]$ that minimizes $p_1(t = 6)$ over the design space $\chi = [1, 4]^4$, with initial conditions of motion $p_1 = 1, p_2 = 0, \dot{p}_1 = \dot{p}_2 = 0$.

4.1.8. Borehole function

The Borehole function models the water flow through a borehole. We adopt the following multifidelity setting to compute the water flow rate [52]:

$$f^{(2)}(\boldsymbol{x}) = \frac{2\pi x_3(x_4 - x_6)}{\log(x_2/x_1)\left(1 + \frac{2x_7 x_3}{\log(x_2/x_1)x_1^2 x_8} + \frac{x_3}{x_5}\right)} \tag{47}$$

$$f^{(1)}(\boldsymbol{x}) = \frac{5x_3(x_4 - x_6)}{\log(x_2/x_1)\left(1.5 + \frac{2x_7 x_3}{\log(x_2/x_1)x_1^2 x_8} + \frac{x_3}{x_5}\right)} \tag{48}$$

where $x_1 \in [0.05, 0.15]$ is the radius of the borehole, $x_2 \in [100, 50000]$ is the radius of influence, $x_3 \in [63070, 115600]$ is the transmissivity of upper aquifer, $x_4 \in [990, 1110]$ is the potentiometric head of lower aquifer, $x_5 \in [63.1, 116]$ is the transmissivity of lower aquifer, $x_6 \in [700, 820]$ is

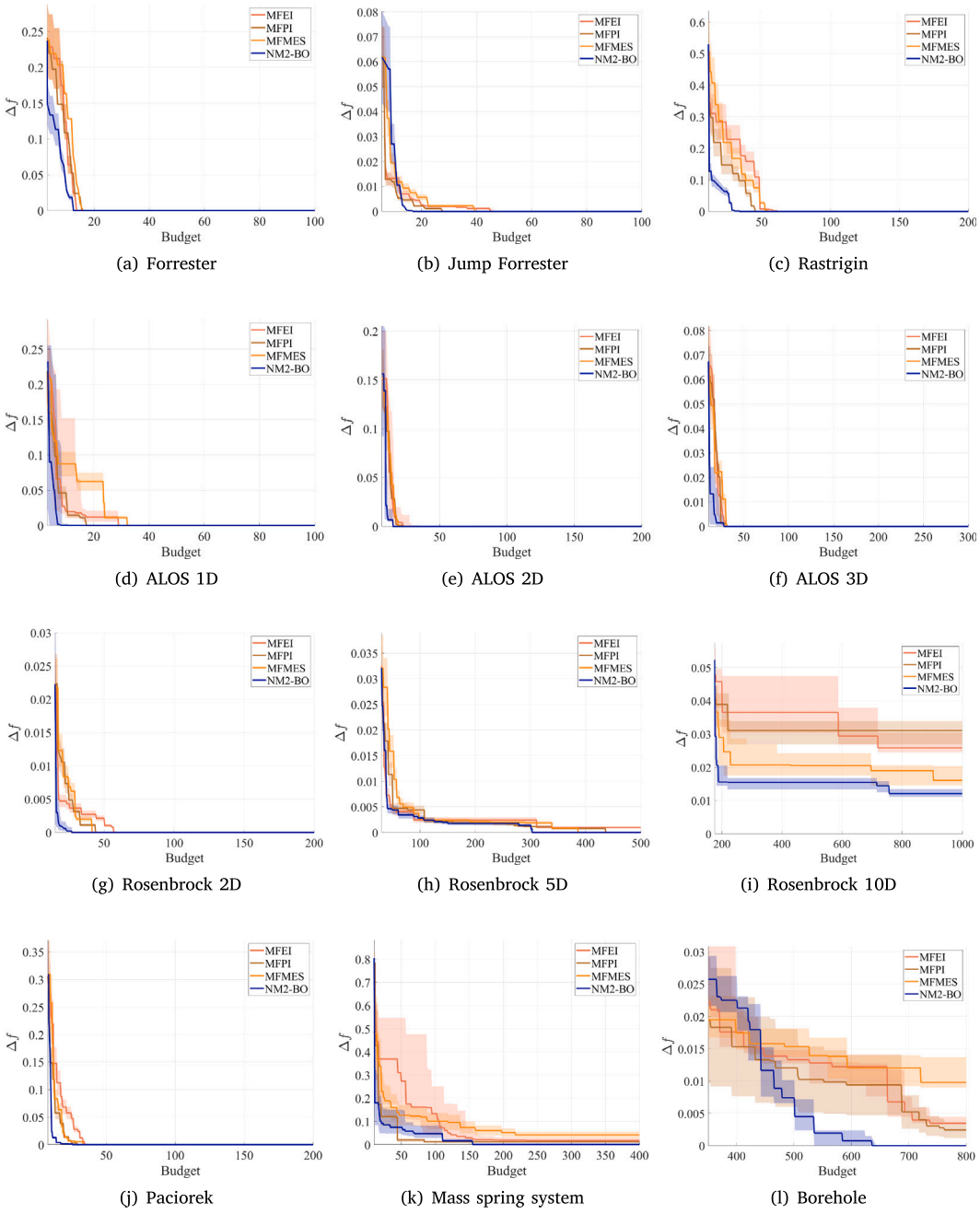


Fig. 5. Error of the minimum of the objective function obtained with the NM2-BO algorithm and compared to the baseline MFBO algorithms for the analytical benchmark problems.

the potentiometric head of lower aquifer, $x_7 \in [1120, 1680]$ is the length of the borehole, and $x_8 \in [9855, 12045]$ is the hydraulic conductivity of the borehole. The minimum of Eq. (47) is $f^* = 9.1884\ m^3/yr$ corresponding to the design $x^* = [0.0504, 3584, 87130, 1007, 112.3, 813.2, 1.658, 9.858]$.

4.1.9. Results and discussion

In the following, the results are observed and discussed in terms of difference error normalized in the input space:

$$\Delta f = \frac{f(\boldsymbol{x}_i^*) - f^*}{f_{max} - f^*} \tag{49}$$

where $f(\boldsymbol{x}_i^*)$ is the best value of the objective function determined by the algorithm at the i-th iteration, f^* is the analytical solution of Eq. (1) and $f_{max} = \max_{\boldsymbol{x} \in \chi} f(\boldsymbol{x})$. The error Δf is plotted as a function of the computational Budget B to compare the optimization results achieved by the algorithms. To quantify and compensate the influence of the random design of experiments, we carried out the tests by running 25 trials considering a random initialization for each benchmark problem, and reported the result in terms of median values of the error together with the values falling in the interval between the 25th and 75th percentiles.

Fig. 5 summarizes the results we obtained from the experiments discussed in Section 4.1. In all the benchmark problems, the Non-Myopic Multifidelity Bayesian Optimization (NM2-BO) outperforms the

baseline multifidelity Bayesian frameworks (MFEI-MFPI-MFMES) in terms of the reduction of the error values, with a much smaller or comparable computational cost.

Indeed, considering the optimization of the Forrester function (Fig. 5(a)) with four levels of fidelity, our algorithm permits to determine the optimal solution with a significant reduction of the expended budget if compared with the baseline MFBO algorithms. Also for the case of the Jump Forrester problem (Fig. 5(b)), the NM2-BO algorithm is capable to identify the optimum point with a fraction of the maximum computational budget while the competing methods reduce the error consuming more computational resources: MFPI requires almost double the resource expenditure and both MFMES and MFEI demand for more than three times the computational cost of NM2-BO to converge.

A significant performance of our non-myopic framework can be observed in the optimization of the Rastrigin function (Fig. 5(c)) considering three levels of fidelity. The step-like behavior of the error occurs when the optimizer falls in a local minimum of the multimodal function, leading to a constant error until a new minimum is observed. This behavior is emphasized for the myopic baseline MFBO algorithms while is almost not appreciable for the NM2-BO: those outcomes suggest that the lookahead property allows to handle objective functions with a pronounced multimodal nature. A possible explanation for this finding might be attributed to an efficient form of exploration, where the observation of high-uncertainty regions of the input domain improves future evaluations and leads to an effective exploitation towards the optimum. This is in agreement with what observed by Ginsbourger and LeRiche [18] for their non-myopic single-fidelity optimization algorithm.

The results for the ALOS benchmark problem illustrate that the NM2-BO achieves a significant acceleration of the optimization procedure if compared with the baseline MFBO frameworks for the one- (Fig. 5(d)), two- (Fig. 5(e)) and three- (Fig. 5(f)) dimensional problems. The performance of NM2-BO might be explained with the computation of an accurate surrogate model thanks to the efficient exploration also highlighted for the Rastrigin test case, that effectively guides the search for the optimum in presence of objective functions with oscillatory nature at different frequencies.

The experiments on the Rosenbrock function are conducted for input spaces of various dimensionality ($d = 2, 5, 10$). For $d = 2$ (Fig. 5(g)), the NM2-BO provides larger accelerations of the optimization process, and identifies the optimum solution with a fraction of the computational cost demanded by the baseline MFBO approaches. As the dimension increases to $d = 5$ (Fig. 5(h)), MFEI converges to a sub-optimal solution while MFMES and MFPI achieve the exact identification of the optimum with higher computational expenditure if compared with our NM2-BO algorithm. At $d = 10$ (Fig. 5(i)), the algorithms are not capable to reach the optimum with the allocated budget. This undesirable performance is due to the difficulties of the GPs to accurately model high dimensional functions. After the initial stages, the error decreases only with large computational expense for both the non-myopic and the standard MFBO schemes. We observed that the algorithms here mostly query the low-fidelity model to contain the computational expense during a secondary exploration phase.

The outcomes for the Paciorek test case (Fig. 5(j)) shows that our non-myopic algorithm outperforms the standard MFBO frameworks reducing the error to zero with a fraction of the computational cost. These results demonstrate that the lookahead strategy permits to improve the performance of optimization even in the presence of noise in measurements. In addition, the NM2-BO framework reaches the minimum function value with a smaller cost than the competing standard MFBO for both the optimization problem of the mass spring system (Fig. 5(k)) and of the Borehole function (Fig. 5(l)), consistently with the results obtained for the other benchmark functions.

The results achieved for all the benchmark problems reveal that the non-myopic multifidelity strategy leads to overall better performance and accelerations with respect to the standard multifidelity settings. In addition, the outcomes demonstrate the capability of NM2-BO to efficiently explore the input space with contained computational expense and take advantage from the acquired information to locally exploit towards the optimum.

4.2. Airfoil shape optimization problem

The airfoil shape optimization task aims at computing the optimized geometrical shape that minimizes the drag coefficient C_d of a RAE 2822 airfoil, subject to maintaining a constant lift coefficient C_l. Additional constraints include the pitching moment C_m, the flight conditions, and the geometry of the airfoil. The shape of the RAE 2822 is modified through the open-source code WG2AER developed by Quagliarella and Diez [26], where the initial geometry of the airfoil is linearly combined with the shape modification functions through the weights $\boldsymbol{w}$. Specifically, we use the six shape modification polynomial functions and assign the weights $\boldsymbol{w} = [w_1, \dots, w_6]$ at each function to modify the shape of the RAE 2822 airfoil. The design optimization problem is formulated as follows:

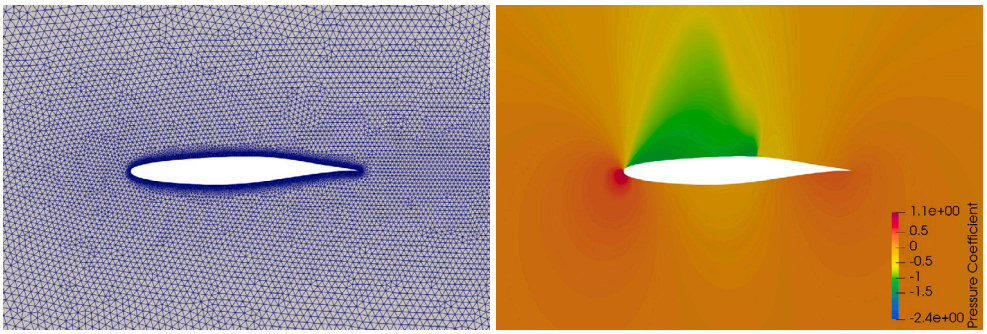
$$
\begin{aligned}
\min_{\boldsymbol{x}\in\chi} \quad & C_d(\boldsymbol{x}) \\
& \boldsymbol{x} = [w_1, \dots, w_6] && \text{(50a)} \\
s.t. \quad & C_l = 0.824 && \text{(50b)} \\
& -0.1 \le C_m \le -0.01 && \text{(50c)} \\
& M = 0.734 && \text{(50d)} \\
& h = 10\,000 \text{ m} && \text{(50e)} \\
& t/c = 0.1211 && \text{(50f)} \\
& r \ge 0.007c && \text{(50g)} \\
& \tau \ge 5^\circ && \text{(50h)} \\
& t_{85}/c \ge 0.02 && \text{(50i)} \\
& \chi = [-1, 1]^6 && \text{(50j)}
\end{aligned}
$$

where M is the flight Mach number and h is the flight altitude, t represents the thickness of the airfoil, c is the chord, τ is the trailing edge angle, and t_{85} is the thickness of the airfoil at the 85% of the chord. The search for the optimal airfoil shape aims at identifying the combination of design variables, namely the weights of the shape modification functions $\boldsymbol{w}$, that minimizes the drag coefficient $C_d(\boldsymbol{x})$ over the design space χ defined by the move limits for the weights $I_w = [-1, 1]^6$. The constraint on the lift coefficient is satisfied considering the angle of attack α as a free parameter in the CFD simulations and the constraint on the maximum thickness is automatically satisfied scaling the airfoil after the geometry modifications. The remaining constraints are considered formalizing the corresponding unconstrained optimization problem through the quadratic penalty functions as follows:

$$
\begin{aligned}
\min_{\boldsymbol{x}\in\chi} \; & C_d + c_1 g^+(C_m, -0.1) + c_1 g^-(C_m, -0.01) + \\
& c_2 g^+(t/c, 0.007) + c_3 g^+(\tau, 5^\circ) + c_4 g^+(t_{85}/c, 0.02)
\end{aligned} \tag{51}
$$

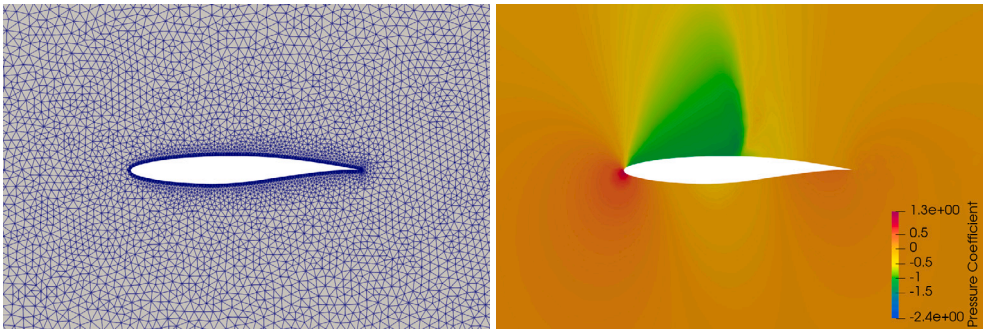
where g^+ and g^- are the quadratic penalty functions [26], and $c_1 = 1000$, $c_2 = 5000$, $c_3 = 10$, and $c_4 = 30$.

The flow-field modeling consists of two levels of fidelity that are assembled using an open-source computational tool for the aerodynamic analysis [26]. Both the aerodynamic models take in input the airfoil geometry modified through the assignment of the weights $\boldsymbol{w}$ together with the flight altitude h, the Reynolds number Re and the free-stream Mach number M, and return the drag coefficient C_d, lift coefficient C_l, and the pitching moment coefficient C_m. We assume the flow-field in a fully turbulent condition and model the aerodynamic domain through



(a) (b)

Fig. 6. (a) High-fidelity discretization of the computational domain with 90000 elements, and (b) high-fidelity pressure coefficient contours for the unmodified RAE-2822 airfoil.



(a) (b)

Fig. 7. (a) Low-fidelity discretization of the computational domain with 30000 elements, and (b) low-fidelity pressure coefficient contours for the unmodified RAE-2822 airfoil.

the Reynolds Averaged Navier–Stokes (RANS) equations. The software numerically solves the RANS equations using a finite volume method, with an automatic procedure to generate a specific mesh for each modified shape of the RAE 2822.

We determine the level of fidelity of the aerodynamic simulation by modifying the element scale factor ES that defines the density of the computational mesh, where the lower the element scale the more refined is the discretization of the flow-field. For the optimization problem in Eq. (50), two levels of fidelity are considered for the aerodynamic modeling: we set $ES = 2.5$ for the high-fidelity model corresponding to a grid of about 90000 cells, and $ES = 12$ for the low-fidelity model with about 30000 cells. On average, the high-fidelity model requires 35 minutes of computational time using a laptop PC equipped with Intel Core i7-6700HQ (2.6 GHz) and 32 GB of RAM, while the low-fidelity model reduces the evaluation cost of the 58% requiring 15 minutes. Fig. 6 illustrates the high-fidelity computational grid (Fig. 6(a)) and the corresponding distribution of the pressure coefficient (Fig. 6(b)) computed with the high-fidelity aerodynamic model, while Fig. 7 provides details about the low-fidelity mesh (Fig. 7(a)) and correlated distributions of the pressure coefficient (Fig. 7(b)) in output from the low-fidelity aerodynamic model; all those outcomes are obtained considering the unmodified RAE-2822 airfoil. The flow solver leverages for all the levels of fidelity the finite-volume code SU2 v6.2.0 to compute numerical solutions of the RANS equation using the Spalart–Allmaras turbolence model, a II order monotone upstream-centered scheme and an adaptive CFL number [53]. We require computational residuals minor than 10^{-6} as the convergence criteria, with a fixed maximum number of iterations of 20000. The computational grid is generated through the GMSH software v4 where an automated procedure allows to determine an hybrid grid of triangles and quadrangles elements, balancing the accuracy and efficiency of the CFD computation with the robustness of the grid generation when the airfoil geometry is modified [54].

4.2.1. Results and discussion

To discuss the performance obtained for the aerodynamic constrained shape optimization of the RAE 2822 airfoil, we evaluate the results in terms of:

$$C_d^* = \min(C_d(\boldsymbol{x})) \tag{52}$$

The minimum drag coefficient C_d^* provides a measure of the improvement of the design solution that allows to assess the effectiveness of the optimal shape configuration identified by the competing methods. We consider a statistics over 25 experiments starting from different initial samples determined through a Latin hypercube design of experiments [46]. This allows to quantify and compensate the influence of the random initial sampling process in the assessment of the performance of the competing algorithms. In particular, the multifidelity searches are initialized with 30 initial design configurations, of which $n_0^{(1)} = 25$ samples are observed with the low-fidelity aerodynamic model and $n_0^{(2)} = 5$ are computed with the high-fidelity aerodynamic model.

Fig. 8 illustrates the values of the minimum drag coefficient C_d^* as a function of the computational budget B to compare the optimization outcomes achieved by the competing algorithms. The results are reported in terms of median values of the minimum drag coefficient together with the associated statistics in between the 25th and 75th percentiles. At the beginning of the optimization procedure, all the starting samples score worse than the baseline solution (C_d^* >0.020475) and the algorithms progressively improve the design solution through the evaluation of the physics-based models. We can observe that the competing methods are capable to achieve design solutions superior to the baseline unmodified RAE 2822 airfoil. However, our non-myopic multifidelity strategy permits to identify superior design solutions if compared with baseline MFBO algorithms, providing larger reduction of the drag coefficient with a fraction of the computational cost.

Specific median values that characterize the distribution of the multifidelity experiments at $B = 15$, 25, 50 and 100 are reported in Table 2. In particular, the NM2-BO algorithm achieves a median

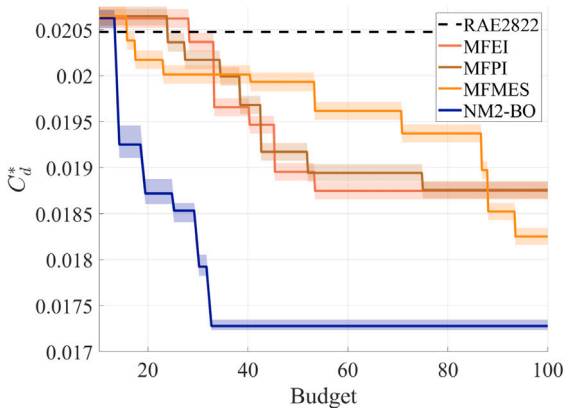


Fig. 8. Minimum drag coefficient C_d^* obtained with the NM2-BO algorithm and the baseline MFBO algorithms.

Table 2
Comparison between NM2-BO and standard MFBO algorithms median values of the minimum drag coefficient C_d^*.

B	C_d^* MFEI	C_d^* MFPI	C_d^* MFMES	C_d^* NM2-BO
15	0.02062	0.02064	0.02065	0.01925
25	0.02062	0.02036	0.02001	0.01872
50	0.01895	0.01917	0.01993	0.01727
100	0.01875	0.01876	0.01825	0.01727

Table 3
Comparison between the best design solutions evaluated with the NM2-BO and the standard MFBO algorithms.

Method	w_1	w_2	w_3	w_4	w_5	w_6
NM2-BO	−0.9875	0.9755	−0.1987	0.3927	0.8424	−0.9451
MFEI	−0.3785	−0.1015	−0.4479	0.3031	0.2993	−0.4426
MFPI	−0.4580	0.4574	0.2391	0.4266	0.4096	−0.2365
MFMES	−0.4752	0.4857	0.2124	−0.1824	0.3324	−0.1692

value of the drag coefficient equal to $C_d^* = 0.01872$ for $B = 25$ which corresponds to a design improvement of the 8.6% if compared with the baseline solution, and converges to an optimal median value of $C_d^* = 0.01727$ with a design upgrade of the 15.7% consuming a budget of just $B = 32.7$. The second best performing algorithm is the MFMES, which scores $C_d^* = 0.01825$ and provides a design improvement of the 10.9% with respect to the baseline airfoil adopting a computational budget $B = 93.6$ approximately three times larger than NM2-BO.

Table 3 provides details about the best design solutions identified by the optimization algorithms. Our NM2-BO framework identifies an optimal design configuration that permits to achieve a drag coefficient of $C_d^* = 0.01723$ which corresponds to an upgrade of the 15.9% with respect to the baseline solution. These outcomes suggest that our NM2-BO algorithm is capable to capitalize the two-step ahead informative gains quantified through our original acquisition function (Section 3.2), enhancing the wise selection of the aerodynamic model to query while balancing computational cost and effectiveness of the design solution.

Fig. 9 illustrates the best optimal airfoil geometries obtained with the competing algorithms compared with the baseline RAE2822 airfoil, and the related distributions of the pressure coefficient. In particular, the optimal airfoil computed with our non-myopic algorithm allows to efficiently expand the flow at the upper surface leading edge while reducing the intensity of the shock waves that occur at transonic speed; indeed, the corresponding distribution of the pressure coefficient (Fig. 9(e)) shows that the shock wave on the upper surface is reduced in intensity, and the supersonic bubble evolves with an almost constant pressure plateau. These outcomes are achieved through the reduction of the adverse pressure gradient which results in the substantial decrease of the drag coefficient.

5. Conclusions

This paper presented a Non-Myopic Multifidelity Bayesian Optimization framework (NM2-BO) to accelerate the optimization of expensive objective functions using multiple representations at different levels of fidelity and evaluating costs. This is achieved by measuring the informative gains acquired from future steps of the optimization procedure through a sampling scheme based on an original two-step lookahead multifidelity acquisition function. Our approach can handle an unlimited finite number of fidelity levels, ensuring its applicability to a variety of real-world optimization problems. We demonstrate the performances of our NM2-BO algorithm against standard multifidelity Bayesian frameworks on a large set of benchmark optimization problems. These analytical experiments have revealed that the proposed non-myopic multifidelity scheme is capable to outperform the standard algorithms leading to a superior solution of the optimization procedure with a fraction of the computational cost, investigating over a variety of mathematical properties that are frequently encountered in real-world applications in science and engineering – such as non linearity, multimodality, oscillatory nature, non-convexity and presence of noise. In addition, we demonstrated our approach for the physics-based aerodynamic optimization problem of a RAE-2822 airfoil subject to aerodynamic and geometric constraints. The NM2-BO permits to improve the aerodynamic design of the 15.7% requiring a 65% lower computational cost if compared with the best performing standard algorithm, which achieved a design improvement of the 10.9%. Across all the demonstration campaign, the non-myopic multifidelity scheme allows to effectively capitalize from the use of limited high-fidelity information to identify improved optimal solutions, while targeting a trade-off between computational cost and future information gains as prescribed by our two-step lookahead multifidelity acquisition function.

CRediT authorship contribution statement

Francesco Di Fiore: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Laura Mainini:** Writing – review & editing, Visualization, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

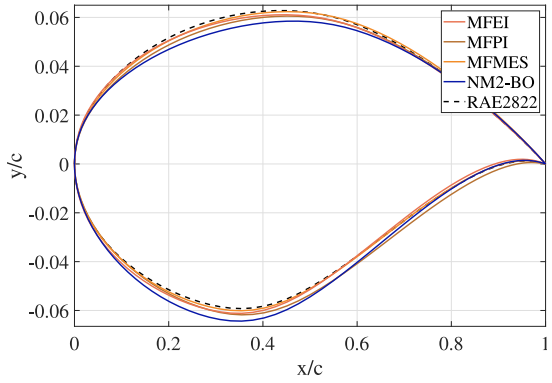
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

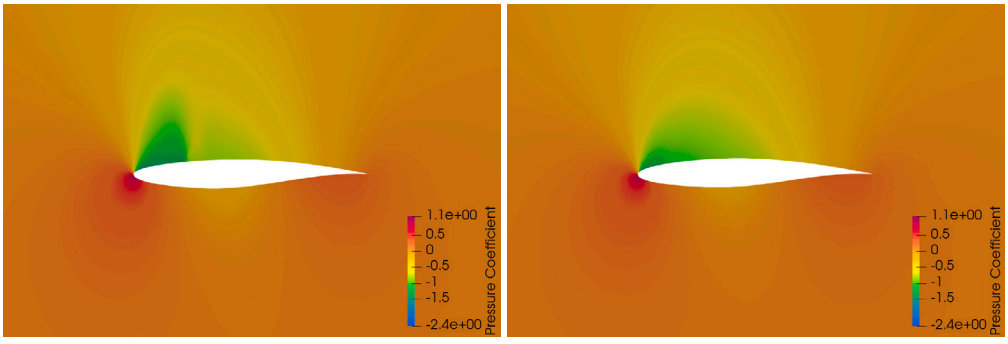
Data will be made available on request.

Acknowledgments

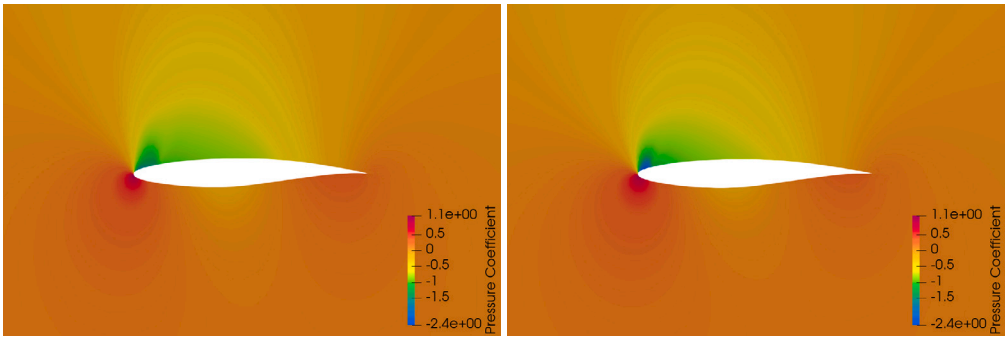
This work was supported by project Multisource Frameworks to Support Real-time Structural Assessment and Autonomous Decision Making under the Visiting Professor program of Politecnico di Torino. Additional acknowledgments to the Univeristy's Doctoral Scholarship supporting Mr. Francesco Di Fiore. The authors thank Prof. Paolo Maggiore at Politecnico di Torino for his support.



(a) Optimal airfoils



(b) MFEI (c) MFPI



(d) MFMES (e) NM2-BO

Fig. 9. (a) Optimal airfoils geometry and associated pressure coefficient contours obtained with (b) MFEI, (c) MFPI, (d) MFMES, and (e) our NM2-BO algorithm.

References

[1] Jonas Mockus, Vytautas Tiesis, Antanas Zilinskas, The Application of Bayesian Methods for Seeking the Extremum, Towards Global Optimization 2 (117-129) (1978) 2.

[2] Jasper Snoek, Hugo Larochelle, Ryan P Adams, Practical Bayesian Optimization of Machine Learning Algorithms, in: F. Pereira, C.J. Burges, L. Bottou, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems, 25 (2012).

[3] MC Kennedy, A O'Hagan, Predicting the output from a complex computer code when fast approximations are available, Biometrika (ISSN: 0006-3444) 87 (1) (2000) 1–13, http://dx.doi.org/10.1093/biomet/87.1.1.

[4] A.I. Forrester, A. Sóbester, A.J. Keane, Multi-fidelity optimization via surrogate modelling, Proc. R. Soc. A 463 (2088) (2007) 3251–3269, http://dx.doi.org/10.1098/rspa.2007.1900.

[5] Benjamin Peherstorfer, Karen Willcox, Max Gunzburger, Survey of Multifidelity Methods in Uncertainty Propagation, Inference, and Optimization, SIAM Review 60 (3) (2018) 550–591, http://dx.doi.org/10.1137/16M1082469.

[6] S. Choi, J.J. Alonso, I.M. Kroo, M. Wintzer, Multifidelity design optimization of low-boom supersonic jets, J. Aircr. 45 (1) (2008) 106–118, http://dx.doi.org/10.2514/1.28948.

[7] P.S. Beran, D. Bryson, A.S. Thelen, M. Diez, A. Serani, Comparison of multi-fidelity approaches for military vehicle design, in: AIAA Aviation 2020 forum and exposition, 2020, p. 3158, http://dx.doi.org/10.2514/6.2020-3158.

[8] D. Huang, T.T. Allen, W.I. Notz, R.A. Miller, Sequential kriging optimization using multiple-fidelity evaluations, Struct. Multidiscip. Optim. 32 (5) (2006) 369–382, http://dx.doi.org/10.1007/s00158-005-0587-0.

[9] Kirthevasan Kandasamy, Gautam Dasarathy, Junier B Oliva, Jeff Schneider, Barnabas Poczos, Gaussian Process Bandit Optimisation with Multi-fidelity Evaluations, in: Advances in Neural Information Processing Systems, 29, Curran Associates, Inc., 2016.

[10] Y. Zhang, T.N. Hoang, B.K.H. Low, M. Kankanhalli, Information-based multi-fidelity bayesian optimization, in: Advances in Neural Information Processing Systems Workshop on Bayesian Optimization, 2017, pp. 1–5.

[11] X. Ruan, P. Jiang, Q. Zhou, J. Hu, L. Shu, Variable-fidelity probability of improvement method for efficient global optimization of expensive black-box problems, Struct. Multidiscip. Optim. 62 (2020) 3021–3052, http://dx.doi.org/10.1007/s00158-020-02646-9.

[12] S. Takeno, H. Fukuoka, Y. Tsukada, T. Koyama, M. Shiga, I. Takeuchi, M. Karasuyama, Multi-fidelity bayesian optimization with max-value entropy search and its parallelization, in: International Conference on Machine Learning, PMLR, 2020, pp. 9334–9345.

[13] M.G. Fernández-Godino, C. Park, N.-H. Kim, R.T. Haftka, Review of multi-fidelity models, 2016, arXiv preprint arXiv:1609.07196.

[14] C. Park, R.T. Haftka, N.H. Kim, Remarks on multi-fidelity surrogates, Struct. Multidiscip. Optim. 55 (3) (2017) 1029–1050, http://dx.doi.org/10.1007/s00158-016-1550-y.

[15] Matthias Poloczek, Jialei Wang, Peter Frazier, Multi-Information Source Optimization, in: Advances in Neural Information Processing Systems, 30, Curran Associates, Inc., 2017.
[16] M. Meliani, N. Bartoli, T. Lefebvre, M.-A. Bouhlel, J.R. Martins, J. Morlier, Multi-fidelity efficient global optimization: Methodology and application to airfoil shape design, in: AIAA Aviation 2019 Forum, 2019, p. 3236, http://dx.doi.org/10.2514/6.2019-3236.
[17] M. Imani, S.F. Ghoreishi, D. Allaire, U.M. Braga-Neto, Mfbo-ssm: Multi-fidelity bayesian optimization for fast inference in state-space models, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 7858–7865, http://dx.doi.org/10.1609/aaai.v33i01.33017858.
[18] D. Ginsbourger, R. Le Riche, Towards gaussian process-based optimization with finite time horizon, in: MODa 9–Advances in Model-Oriented Design and Analysis, Springer, 2010, pp. 89–96.
[19] Remi Lam, Karen Willcox, David H. Wolpert, Bayesian Optimization with a Finite Budget: An Approximate Dynamic Programming Approach, in: Advances in Neural Information Processing Systems, 29 (2016).
[20] Remi Lam, Karen Willcox, Lookahead Bayesian Optimization with Inequality Constraints, in: Advances in Neural Information Processing Systems, 30, Curran Associates, Inc., 2017.
[21] W.B. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, Vol. 703, John Wiley & Sons, 2007.
[22] M.A. Osborne, R. Garnett, S.J. Roberts, Gaussian processes for global optimization, in: 3rd International Conference on Learning and Intelligent Optimization, LION3, Citeseer, 2009, pp. 1–15.
[23] J. González, M. Osborne, N. Lawrence, Glasses: Relieving the myopia of bayesian optimisation, in: Artificial Intelligence and Statistics, PMLR, 2016, pp. 790–799.
[24] F. Di Fiore, L. Mainini, NM-MF: Non-myopic multifidelity framework for constrained multi-regime aerodynamic optimization, AIAA J. (2022) 1–11, http://dx.doi.org/10.2514/1.J062219.
[25] L. Mainini, A. Serani, M.P. Rumpfkeil, E. Minisci, D. Quagliarella, H. Pehlivan, S. Yildiz, S. Ficini, R. Pellegrini, F. Di Fiore, et al., Analytical benchmark problems for multifidelity optimization methods, in: NATO STO AVT-354 Multi-Fidelity Methods for Military Vehicle Design Workshop, 2022, Codes available at: https://gitlab.com/qudo046/avt-331-l1-benchmarks, Accessed: 1 December 2022.
[26] D. Quagliarella, M. Diez, An open-source aerodynamic framework for benchmarking multi-fidelity methods, in: AIAA Aviation 2020 Forum and Exposition, 2020, p. 3179, http://dx.doi.org/10.2514/6.2020-3179.
[27] C.E. Rasmussen, Gaussian processes in machine learning, in: Summer School on Machine Learning, Springer, 2003, pp. 63–71.
[28] A. Pepelyshev, The role of the nugget term in the Gaussian process method, in: MODa 9–Advances in Model-Oriented Design and Analysis: Proceedings of the 9th International Workshop in Model-Oriented Design and Analysis, Springer, 2010, pp. 149–156.
[29] D.R. Jones, M. Schonlau, W.J. Welch, Efficient global optimization of expensive black-box functions, Journal of Global Optimization 13 (4) (1998) 455–492, http://dx.doi.org/10.1023/A:1008306431147.
[30] Niranjan Srinivas, Andreas Krause, Sham Kakade, Matthias Seeger, Gaussian process optimization in the bandit setting: no regret and experimental design, in: Proceedings of the 27th International Conference on International Conference on Machine Learning, Omnipress, 2010, pp. 1015–1022.
[31] Philipp Hennig, Christian J. Schuler, Entropy search for information-efficient global optimization, J. Mach. Learn. Res. (ISSN: 1532-4435) 13 (2012) 1809–1837.
[32] Z. Wang, S. Jegelka, Max-value entropy search for efficient bayesian optimization, in: International Conference on Machine Learning, PMLR, 2017, pp. 3627–3635.
[33] José Miguel Henrández-Lobato, Matthew W. Hoffman, Zoubin Ghahramani, Predictive entropy search for efficient global optimization of black-box functions, in: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1, MIT Press, 2014, pp. 918–926.
[34] Y.W. Teh, M. Seeger, M.I. Jordan, Semiparametric latent factor models, in: International Workshop on Artificial Intelligence and Statistics, PMLR, 2005, pp. 333–340.
[35] Edwin V Bonilla, Kian Chai, Christopher Williams, Multi-task Gaussian Process Prediction, in: Advances in Neural Information Processing Systems, 20 (2007).
[36] M.A. Alvarez, N.D. Lawrence, Computationally efficient convolved multiple output gaussian processes, J. Mach. Learn. Res. 12 (2011) 1459–1500.
[37] L. Brevault, M. Balesdent, A. Hebbal, Overview of gaussian process based multi-fidelity techniques with variable relationship between fidelities, application to aerospace systems, Aerospace Science and Technology 107 (2020) 106339, http://dx.doi.org/10.1016/j.ast.2020.106339.
[38] F. Grassi, G. Manganini, M. Garraffa, L. Mainini, Raal: Resource aware active learning for multifidelity efficient optimization, AIAA J. 61 (6) (2023) 2744–2753, http://dx.doi.org/10.2514/1.J061383.
[39] A. Forrester, A. Sobester, A. Keane, Engineering Design Via Surrogate Modelling: A Practical Guide, John Wiley & Sons, 2008.
[40] P.H. Reisenthel, T.T. Allen, Application of multifidelity expected improvement algorithms to aeroelastic design optimization, in: 10th AIAA Multidisciplinary Design Optimization Conference, 2014, p. 1490, http://dx.doi.org/10.2514/6.2014-1490.
[41] Y. Liu, S. Chen, F. Wang, F. Xiong, Sequential optimization using multi-level cokriging and extended expected improvement criterion, Struct. Multidiscip. Optim. 58 (2018) 1155–1173, http://dx.doi.org/10.1007/s00158-018-1959-6.
[42] M.-L. Puterman, Markov.decision. processes, Handbooks in Operations Research and Management Science, No. 2, 1990, pp. 331–434.
[43] D. Bertsekas, Dynamic Programming and Optimal Control: Volume I, Athena scientific, 1995.
[44] J.T. Wilson, F. Hutter, M.P. Deisenroth, Maximizing acquisition functions for bayesian optimization, 2018, arXiv preprint arXiv:1805.10196.
[45] M.D. McKay, Latin hypercube sampling as a tool in uncertainty analysis of computer models, in: Proceedings of the 24th Conference on Winter Simulation, 1992, pp. 557–564.
[46] F.A. Viana, A tutorial on latin hypercube design of experiments, Quality and Reliability Engineering International 32 (5) (2016) 1975–1985, http://dx.doi.org/10.1002/qre.1924.
[47] S. Lee, F. Dietrich, G.E. Karniadakis, I.G. Kevrekidis, Linking gaussian process regression with data-driven manifold embeddings for nonlinear data fusion, Interface Focus 9 (3) (2019) 20180083, http://dx.doi.org/10.1098/rsfs.2018.0083.
[48] H. Wang, Y. Jin, J. Doherty, A generic test suite for evolutionary multifidelity optimization, IEEE Transactions on Evolutionary Computation 22 (6) (2017) 836–850, http://dx.doi.org/10.1109/TEVC.2017.2758360.
[49] D.L. Clark Jr., H.-R. Bae, K. Gobal, R. Penmetsa, Engineering design exploration using locally optimized covariance kriging, AIAA J. 54 (10) (2016) 3160–3175, http://dx.doi.org/10.2514/1.J054860.
[50] H. Rosenbrock, An automatic method for finding the greatest or least value of a function, The Computer Journal 3 (3) (1960) 175–184, http://dx.doi.org/10.1093/comjnl/3.3.175.
[51] D.J. Toal, Some considerations regarding the use of multi-fidelity kriging in the construction of surrogate models, Struct. Multidiscip. Optim. 51 (6) (2015) 1223–1245, http://dx.doi.org/10.1007/s00158-014-1209-5.
[52] S. Xiong, P.Z. Qian, C.J. Wu, Sequential design and analysis of high-accuracy and low-accuracy computer codes, Technometrics 55 (1) (2013) 37–46.
[53] T.D. Economon, F. Palacios, S.R. Copeland, T.W. Lukaczyk, J.J. Alonso, Su2: An open-source suite for multiphysics simulation and design, AIAA J. 54 (3) (2016) 828–846, http://dx.doi.org/10.2514/1.J053813.
[54] C. Geuzaine, J.-F. Remacle, Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities, Internat. J. Numer. Methods Engrg. 79 (11) (2009) 1309–1331, http://dx.doi.org/10.1002/nme.2579.