

NON-MYOPIC MULTIFIDELITY BAYESIAN OPTIMIZATION

Francesco Di Fiore

Politecnico di Torino

francesco.difiore@polito.it

Laura Mainini

Massachusetts Institute of Technology

Politecnico di Torino

laura.mainini@polito.it

July 20, 2022

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 at a specific input, precluding greater informative gains that might be obtained looking ahead more steps. This paper proposes a non-myopic multifidelity Bayesian framework 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 in the solution over two steps ahead. We demonstrate that the proposed algorithm outperforms a standard multifidelity Bayesian framework on popular benchmark optimization problems.

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 costly.

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 at this new training sample, and 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 produce 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, 4, 5, 6]. Multifidelity Bayesian optimization (MFBO) combines the function outputs at different levels of fidelity into a single probabilistic model, and use this information to compute a multifidelity acquisition function that allows to jointly select the new input location and the associated level of fidelity to query [7, 8, 9, 10]. 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 [11, 12, 13, 14, 15]. 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

arXiv:2207.06325v2 [cs.LG] 19 Jul 2022

MFBO
but "look
ahead" more

ask of
possible
applications?

How are they
defining these
fidelity measures?

* Most MFBO
is greedy

there isn't ...
DP? D+a?
other?

↑
general non-myopic
algor.

non-myopic AF

① MDP? Too many computations, slow?

* Learn Markov decision process.

A PREPRINT - JULY 20, 2022

determined through a partially observable Markov decision process (POMDP) [16, 17, 18]. However, the solution of the POMDP is computationally intense, limiting this approach to small-scale optimization problems [19]. Several works proposed approximations of the POMDP to formulate two-step lookahead acquisition functions capable to enhance the performance of BO, while containing the computational expense that would arise looking more steps ahead [20, 16, 21, 17, 18]. 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. *Sounds familiar*

To address this issue, we propose a strategy for non-myopic multifidelity Bayesian optimization to maximize the long-term reward as an improvement of the optimal solution. 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. ** is a DP problem*
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. ** Use Monte-Carlo to get closed-form*
3. Illustrate and discuss the proposed Non-myopic MFBO in comparison with standard MFBO over benchmark problems specifically conceived to stress-test multifidelity methods.

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 non-myopic MFBO is numerically investigated and compared to standard MFBO over a variety of benchmarks. 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 $f: \mathcal{X} \rightarrow \mathbb{R}$ from noisy observations at iteratively selected sample points $\mathbf{x} \in \mathcal{X}$ in the input space \mathcal{X} . More formally, we can define our optimization problem as follows:

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \quad (1)$$

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 Equation (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 the prediction. This information is used to compute an acquisition function that defines a policy to measure the improvement of the solution at different input locations, which usually encodes a trade-off between the exploration of the input space and the exploitation toward 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 [22]. The GP distribution $GP(\mu(\mathbf{x}), \kappa(\mathbf{x}, \mathbf{x}'))$ is a joint Gaussian distribution over any finite set of N random input points $\{\mathbf{x}_n\}_{n=1}^N$ completely specified by its mean $\mu(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$ and covariance (also referred as kernel) function $\kappa(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(\mathbf{x}'))]$. We assume that $f \sim GP(0, \kappa(\mathbf{x}, \mathbf{x}'))$ is the prior belief about the objective and that we have collected observations in the form $\{\mathbf{x}_n, y_n\}_{n=1}^N$, where $y_n \sim \mathcal{N}(f(\mathbf{x}_n), \sigma_\epsilon(\mathbf{x}_n))$ and σ_ϵ is the standard deviation of the normally distributed noise. *info. familiar with*

The acquisition function $U: \mathcal{X} \rightarrow \mathbb{R}^+$ under the GP prior depends on the previous observations of the objective and determines which point in \mathcal{X} should be evaluated in the next optimization step via maximization $\mathbf{x}_{new} \in \max_{\mathbf{x} \in \mathcal{X}} U(\mathbf{x})$. There is a variety of acquisition functions commonly adopted in literature, such as the expected improvement (EI) [23],

When say AF is greedy, uses all previous observation to determine next?

So non-myopic uses all prev. observation to predict next few?

* Could be useful in your BO implementation

upper confidence bound (UCB) [24], entropy search (ES) [25], max-value entropy search [26] and predictive entropy search (PES) [27]. In this work we focus on GP based Bayesian optimization with the EI acquisition function, given its simple implementation and reasonable performances in practice:

$$EI(\mathbf{x}) = \sigma(\mathbf{x})(\gamma(\mathbf{x})\Phi(\gamma(\mathbf{x})) + \mathcal{N}(\gamma(\mathbf{x}); 0, 1)) \quad (2)$$

where $\gamma(\mathbf{x}) = (f(\mathbf{x}^*) - \mu(\mathbf{x}))/\sigma(\mathbf{x})$, \mathbf{x}^* is the current location of the best value of the objective and $\Phi(\cdot)$ is the cumulative distribution function of a standard normal distribution.

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 $f^{(l)}(\mathbf{x})$. 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 choose the location of the next input and the level of fidelity to query through a multifidelity acquisition function. The probabilistic surrogate model is defined extending the Gaussian process to a multifidelity scenario through an autoregressive scheme [3]:

$$f^{(l)} = \rho f^{(l-1)}(\mathbf{x}) + \delta^{(l)}(\mathbf{x}), \quad l = 2, \dots, L \quad (3)$$

where $\delta^{(l)} \sim GP(0, \kappa^{(l)}(\mathbf{x}, \mathbf{x}'))$ models the discrepancy between two adjoining levels of fidelity, and ρ is a constant scaling factor. Hence, we can obtain the posterior mean $\mu^{(l)}(\mathbf{x})$ and variance $\sigma^{2(l)}(\mathbf{x})$ of the objective through the covariance matrix $\mathbf{K}(i, j) = \kappa((\mathbf{x}_i, l_i), (\mathbf{x}_j, l_j)) + \delta_{ij}\sigma_\epsilon(\mathbf{x}_i)$ and $\kappa_N^{(l)}(\mathbf{x}) = \kappa((\mathbf{x}_i, l_i), (\mathbf{x}, l))$:

$$\mu^{(l)}(\mathbf{x}) = \kappa_N^{(l)}(\mathbf{x})^T \mathbf{K}^{-1} \mathbf{y} \quad (4)$$

$$\sigma^{2(l)}(\mathbf{x}) = \kappa((\mathbf{x}, l), (\mathbf{x}, l)) - \kappa_N^{(l)}(\mathbf{x})^T \mathbf{K}^{-1} \kappa_N^{(l)}(\mathbf{x}) \quad (5)$$

where δ_{ij} is the Kronecker delta function.

The multifidelity acquisition function is computed starting from the information of the multifidelity GP. Multifidelity acquisition functions include the multifidelity expected improvement (MFEI) [7], Multifidelity Predictive Entropy Search (MFES) [9], or Multifidelity Max-Value Entropy Search [10]. We base our multifidelity formulation on the MFEI, which derives from the Expected improvement acquisition function:

$$MFEI(\mathbf{x}, l) = EI(\mathbf{x})\alpha_1(\mathbf{x}, l)\alpha_2(\mathbf{x}, l)\alpha_3(\mathbf{x}, l) \quad (6)$$

where $EI(\mathbf{x})$ is the expected improvement depicted in Equation 2 evaluated at the highest level of fidelity. The utility functions α_1 , α_2 and α_3 are defined as follows:

$$\alpha_1(\mathbf{x}, l) = \text{corr}[f^{(l)}, f^{(L)}] \quad (7)$$

$$\alpha_2(\mathbf{x}, l) = 1 - \frac{\sigma_\epsilon}{\sqrt{\sigma^{2(l)}(\mathbf{x}) + \sigma_\epsilon^2}} \quad (8)$$

$$\alpha_3(l) = \frac{\lambda^{(L)}}{\lambda^{(l)}} \quad (9)$$

$\alpha_1(\mathbf{x}, l)$ accounts the increase of the objective uncertainty as the 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. $\alpha_2(\mathbf{x}, l)$ 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 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.

* Learn Monte-Carlo method

3 Non-Myopic Multifidelity Bayesian Optimization

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

Use M.C. to predict function value?

Discovering fidelity function as go along?

3.1 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 systems, where its dynamics characterize the sequence of decisions to achieve a specific goal [28, 19]. 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 at 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 determine 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.

Let us consider the generic stage z of MFBO, and define the input \mathbf{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 $S_z = \{\mathbf{x}_n, y_n^{(l_n)}, l_n\}_{n=1}^N$ of N observations, and a control $c_z = \{\mathbf{x}_{z+1}, l_{z+1}\}$ that activates the dynamics of the system. Thus, a policy π_z is defined as a function that maps the state s_z to a control $c_z = \pi_z(s_z)$.

Given the state and the control of the system, we introduce the disturbances to represent a simulated value of the objective function at $\{\mathbf{x}_{z+1}, l_{z+1}\}$, defined as a random variable $d_z^{(l)} \sim \mathcal{N}(\mu_z^{(l)}(\mathbf{x}_{z+1}), \sigma_z^{2(l)}(\mathbf{x}_{z+1}))$ characterized with the mean (Equation (4)) and variance (Equation (5)) 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 $S_{z+1} = S_z \cup \{\mathbf{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}\}$:

$$S_{z+1} = \mathcal{F}(\mathbf{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z) \quad (10)$$

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 control 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 :

$$r_z(\mathbf{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z) = (f_z^{*(L)} - f_{z+1}^{(L)})^+ \quad (11)$$

where $f_z^{*(L)}$ is the minimum value of the objective function at z evaluated at the highest level of fidelity. Thus, we can formulate the two-step lookahead multifidelity acquisition function at a generic stage z as the expected reward:

$$U_z^\pi(\mathbf{x}_{z+1}, l_{z+1}, S_z) = \mathbb{E}[r_z(\mathbf{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z) + J_{z+1}(\mathcal{F}(\mathbf{x}_{z+1}, y^{(l_{z+1})}, l_{z+1}, S_z))] \quad (12)$$

where the expectation is taken with respect to the disturbances, $\mathbb{E}[r_z(\cdot)] = MFEI(\mathbf{x}_{z+1}, l_{z+1})$ is the multifidelity expected improvement (Equation (6)), and $J_{z+1}(\mathcal{F}(\cdot))$ is the long-term expected reward.

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} :

$$J_{z+1} = \max(MFEI(\mathbf{x}_{z+2}, l_{z+2})) \quad (13)$$

Combining Equation (12) with Equation (13), we formalize the two-step lookahead multifidelity acquisition function:

$$U_z^{\pi^*}(\mathbf{x}_{z+1}, l_{z+1}, S_z, S_{z+1}) = MFEI(\mathbf{x}_{z+1}, l_{z+1}) + \mathbb{E}[\max(MFEI(\mathbf{x}_{z+2}, l_{z+2}))] \quad (14)$$

3.2 Estimating the Two-Step Lookahead Multifidelity Acquisition Function

The solution of the two-step lookahead multifidelity acquisition function (Equation (14)) cannot be computed in closed-form, as requires to evaluate the nested maximizations and expectations.

We adopt the Monte Carlo approach to estimate $U_z^{\pi^*}$, using the reparameterization strategy proposed by Wilson et al. [29] to formulate the value of the objective function at the first step ahead with l -th level of fidelity:

$$f^{(l)}(\mathbf{x}_{z+1}) = \mu_z^{(l)} + \mathbf{C}_z^{(l)}(\mathbf{x}_{z+1}) \mathbf{Z} \quad (15)$$

Algorithm 1 Non-myopic multifidelity Bayesian optimization

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

Output: $f^* = \min f(\mathbf{x})$

- 1: $S_0 \leftarrow \{\mathbf{x}_n, y^{(l_n)}, l_n\}_{n=1}^{N_0}$ collect initial observations
- 2: $\mu_0^{(l)}, \sigma_0^{2(l)} \leftarrow$ learn the initial multifidelity GP
- 3: $i \leftarrow 1$
- 4: **repeat**
- 5: Load \mathbf{x}_i and associated l_i
- 6: Evaluate $y^{(l_i)}(\mathbf{x}_i)$
- 7: $S_i \leftarrow S_{i-1} \cup \{\mathbf{x}_i, y^{(l_i)}, l_i\}$
- 8: $\mu_i^{(l)}, \sigma_i^{2(l)} \leftarrow$ update the multifidelity GP
- 9: $z \leftarrow i$
- 10: Compute $MFEI(\mathbf{x}_{z+1}, l_{z+1})$
- 11: **for** $j \leftarrow 1, N_{MC}$ **do**
- 12: $Z_j \leftarrow \mathcal{N}(0, 1)$
- 13: $\mu_{z+1}^{(l)}, \sigma_{z+1}^{2(l)} \leftarrow$ estimate the multifidelity GP
- 14: Compute $\widehat{MFEI}(\mathbf{x}_{z+2}, l_{z+2}, Z_j)$
- 15: **end for**
- 16: **return** $\widehat{U}_z^{\pi^*} = \{\widehat{U}_{zj}^{\pi^*}\}_{j=1}^{N_{MC}}$
- 17: $U_i^{\pi^*} = \mathbb{E}[\widehat{U}_z^{\pi^*}]$
- 18: $[\mathbf{x}_{i+1}, l_{i+1}] \leftarrow \max(U_i^{\pi^*})$
- 19: $i + 1 \leftarrow i$
- 20: **until** $B_i \leq B_{max}$
- 21: **return** \mathbf{x}^* that minimize $f(\mathbf{x})$ over S_i

where $\mathbf{C}_z^{(l)}$ is the Cholesky decomposition of the covariance matrix \mathbf{K}_z , and Z is an independent standard normal random variable. Then, we use Equation (15) to compute the mean and variance of the multifidelity GP at the step $z + 1$ for the generic input \mathbf{x} :

$$\mu_{z+1}^{(l)}(\mathbf{x}) = \mu_z^{(l)}(\mathbf{x}) + \mathbf{H}_z^{(l)}(\mathbf{x})Z \quad (16)$$

$$\sigma_{z+1}^{(l)}(\mathbf{x}) = \sigma_z^{(l)}(\mathbf{x}) - \mathbf{H}_z^{(l)}(\mathbf{x})\mathbf{H}_z^{(l)}(\mathbf{x})^T \quad (17)$$

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

Equation (16) and Equation (17) are used to estimate the multifidelity expected improvement at the step $z + 1$:

$$MFEI(\mathbf{x}_{z+2}, l_{z+2}) \sim \widehat{MFEI}(\mathbf{x}_{z+2}, l_{z+2}, Z) \quad (18)$$

To approximate the expectation term of Equation (14), we sample the random variable Z and compute $\mathbb{E}[\widehat{U}_z^{\pi^*}(\mathbf{x}_{z+1}, l_{z+1}, Z)]$ averaging over many realizations of the two-step lookahead multifidelity acquisition $\widehat{U}_z^{\pi^*}$ evaluated using Equation (18).

Although the need of numerous evaluations of $\widehat{U}_z^{\pi^*}$, we emphasize that our Monte Carlo approach requires the evaluation of Equation (16) and Equation (17) 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.

3.3 Non-Myopic Multifidelity Optimization Algorithm

Algorithm 1 details the steps of our non-myopic multifidelity optimization 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 [30], together with the objective function at different levels of fidelity $f^{(l)}$, and the multifidelity Gaussian process prior

important
computation
to understand

$GP(0, \kappa^{(l)}(\mathbf{x}, \mathbf{x}'))$. We draw an initial subset of feasible input points $\{\mathbf{x}_n\}_{n=1}^{N_0}$ 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 $S_0 = \{\mathbf{x}_n, y^{(l_n)}, l_n\}_{n=1}^{N_0}$. The multifidelity GP prior together with the initial state S_0 induce the multifidelity Gaussian process posterior with mean $\mu^{(l)}$ (Equation (4)) and variance $\sigma^{2(l)}$ (Equation (5)), representing the first surrogate model of the objective function.

Let us now consider the generic iteration i of the optimization search. The algorithm selects the input \mathbf{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 $y^{(l_i)}(\mathbf{x}_i)$. This information is used to augment the state $S_i = S_{i-1} \cup \{\mathbf{x}_i, y^{(l_i)}, l_i\}$ 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 \mathbf{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 (Equation (14)) is computed on the current state $S_z = S_i$, while the second term is estimated using the Monte-Carlo approach discussed in Section 3.2. 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 Equation (16) and Equation (17). Then, we evaluate the second term of the acquisition function through Equation (18) and compute $\hat{U}_{z,j}^{\pi^*}$. The Monte Carlo algorithm iterates till 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 \mathbf{x}_{i+1} and the associated level of fidelity l_{i+1} . The optimization 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

** Important in finding applications*

In this section, we illustrate and discuss the proposed non-myopic MFBO in comparison with MFBO over a set of **well accepted benchmark** problems, specifically conceived to stress-test multifidelity methods. As the standard MFBO algorithm, we implemented the method based on multifidelity expected improvement proposed by Huang et al. [7].

For all the results presented in this section, we impose the same initial settings for both the non-myopic and the standard MFBO algorithm. 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 [30] and are used to compute the first surrogate model as depicted in Section 3.3. We use the square exponential kernels for all the GP covariances, and optimize the hyperparameters of the kernel and mean functions of the GP via Maximum Likelihood Estimation [31].

The set of multifidelity benchmark functions we used to validate our methodology is proposed by Mainini et al. [32] and includes the Forrester function, the Rosenbrock function, the Rastrigin function shifted and rotated and a mass spring system optimization problem. The multifidelity Borehole function test-case [33] is also included to further expand the experimentation. This set of functions is specifically selected to emulate mathematical characteristics that are frequent in physics-based optimization problems, keeping contained the computational expense associated with their evaluation. Table 1 summarizes the experimental setup for each test function.

Table 1: Summary of the experiments setup

Function	$n_0^{(2)}$	$n_0^{(1)}$	B_{max}	$\lambda^{(2)}$	$\lambda^{(1)}$
Forrester	2	5	100	1	0.05
Rosenbrock 2D	5	10	200	1	0.5
Rosenbrock 5D	15	30	500	1	0.5
Rosenbrock 10D	50	250	1000	1	0.5
Rastrigin	10	20	200	1	0.0039
Mass Spring	4	10	400	1	1/60
Borehole	100	500	800	1	0.5

example function for toy problem

4.1 Results and Discussion

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

$$\Delta f = \frac{f(\mathbf{x}_i^*) - f^*}{f_{\max} - f^*} \quad (19)$$

where $f(\mathbf{x}_i^*)$ is the best value of the objective function determined by the algorithm at the i -th iteration, f^* is the analytical solution of Equation (1) and $f_{\max} = \max_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})$. 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 5 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 25-th and 75-th percentiles.

Figure 1 summarizes the results we obtained from the experiments discussed in Section 4. In all the benchmark problems, the non-myopic multifidelity Bayesian optimization (Non-Myopic MFBO) outperforms the standard multifidelity Bayesian framework (MFBO) 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 (figure 1(a)), our algorithm permits to determine the optimal solution with a significant reduction of the expended budget if compared with the baseline MFBO algorithm.

A significant performance of our non-myopic framework can be observed in the optimization of the Rastrigin function (figure 1(b)), where the step-like behaviour 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 MFBO while is almost not appreciable for the non-myopic MFBO, suggesting 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, leading to an effective exploitation toward the optimum. This is in agreement with what observed by Ginsbourger and Le Riche [16] for their non-myopic single-fidelity optimization algorithm.

The experiments on the Rosenbrock function are conducted for input spaces of various dimensionality ($d = 2, 5, 10$). For $d = 2$ (figure 1(c)), the non-myopic MFBO outperforms the standard MFBO achieving a much smaller error with a fraction of the cost required by the standard framework. As the dimension increases to $d = 5$ (figure 1(d)), although the MFBO is able to reduce the error in the initial stage, it eventually converges to a sub-optimal solution compared with the non-myopic strategy. At $d = 10$ (figure 1(e)), 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.

In addition, the non-myopic MFBO reaches the minimum function values with a smaller cost than the competing standard MFBO for both the optimization problem of the mass spring system (figure 1(f)) and of the Borehole function (figure 1(g)), consistently with the results obtained for the other benchmark functions.

The results obtained for all the benchmark problems reveal that the non-myopic multifidelity strategy leads to overall better performance with respect to the standard multifidelity setting, demonstrating the ability to efficiently explore the input space with a contained budget and take advantages from the acquired information to locally exploit toward the optimum.

All the experiments presented in this paper are obtained running each test on a single core of a desktop PC with Intel Core i7-8700 (3.2 GHz) and 32 GB of RAM. The computational time required to compute a single iteration of the proposed non-myopic MFBO algorithm ranges from seconds to minutes.

5 Conclusions

In this paper, we proposed a non-myopic multifidelity Bayesian optimization framework based on two-step lookahead multifidelity acquisition function. Our algorithm allows to improve the solution of the optimization by unlocking the informative gain acquired from future steps of the optimization, reducing the total computational expense. The non-myopic framework can handle an unlimited finite number of fidelities levels, ensuring its applicability to a variety of real-world optimization problems. We evaluated the performances of our strategy against a standard multifidelity Bayesian framework on several benchmark optimization problems, revealing that the lookahead property allows to outperform it.

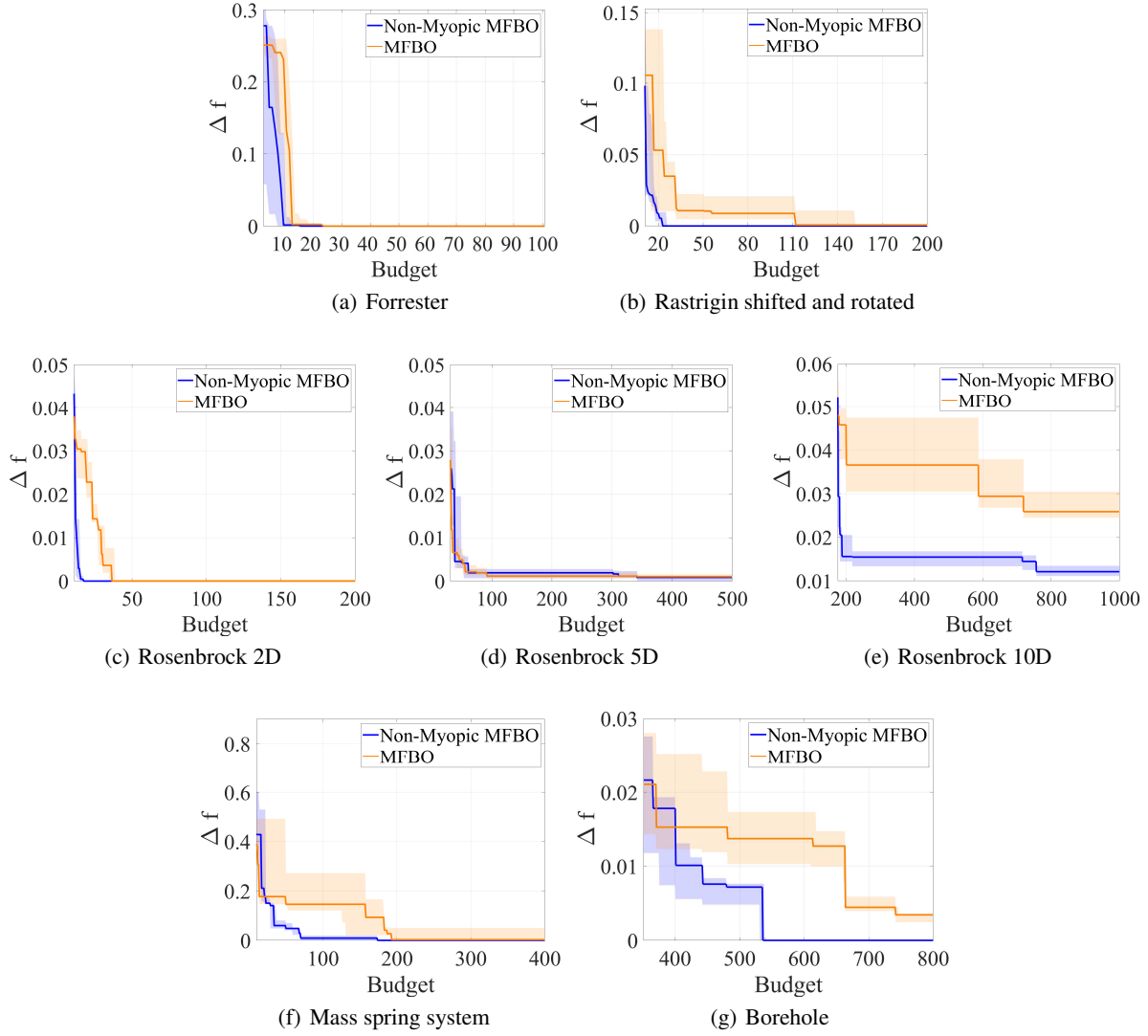


Figure 1: Performances of the Non-Myopic MFBO compared to the standard MFBO.

Acknowledgments

The work has been partly conducted under the Visiting Professor Program of Politecnico di Torino. Additional acknowledgments to the University's Doctoral Scholarship supporting Mr. Francesco Di Fiore. The authors thank Prof. Paolo Maggiore at Politecnico di Torino for the support.

References

- [1] Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of bayesian methods for seeking the extremum. *Towards global optimization*, 2(117-129):2, 1978.
- [2] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.
- [3] Marc C Kennedy and Anthony O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000.

- [4] Alexander IJ Forrester, András Sóbester, and Andy J Keane. Multi-fidelity optimization via surrogate modelling. *Proceedings of the royal society a: mathematical, physical and engineering sciences*, 463(2088):3251–3269, 2007.
- [5] Benjamin Peherstorfer, Karen Willcox, and Max Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *Siam Review*, 60(3):550–591, 2018.
- [6] Philip S Beran, Dean Bryson, Andrew S Thelen, Matteo Diez, and Andrea Serani. Comparison of multi-fidelity approaches for military vehicle design. In *AIAA AVIATION 2020 FORUM*, page 3158, 2020.
- [7] Deng Huang, Theodore T Allen, William I Notz, and R Allen Miller. Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization*, 32(5):369–382, 2006.
- [8] Kirthivasan Kandasamy, Gautam Dasarathy, Junier Oliva, Jeff Schneider, and Barnabás Póczos. Gaussian process optimisation with multi-fidelity evaluations. In *Proceedings of the 30th/International Conference on Advances in Neural Information Processing Systems (NIPS’30)*, 2016.
- [9] Yehong Zhang, Trong Nghia Hoang, Bryan Kian Hsiang Low, and Mohan Kankanhalli. Information-based multi-fidelity bayesian optimization. In *NIPS Workshop on Bayesian Optimization*, 2017.
- [10] Shion Takeno, Hitoshi Fukuoka, Yuhki Tsukada, Toshiyuki Koyama, Motoki Shiga, Ichiro Takeuchi, and Masayuki Karasuyama. Multi-fidelity bayesian optimization with max-value entropy search and its parallelization. In *International Conference on Machine Learning*, pages 9334–9345. PMLR, 2020.
- [11] M Giselle Fernández-Godino, Chanyoung Park, Nam-Ho Kim, and Raphael T Haftka. Review of multi-fidelity models. *arXiv preprint arXiv:1609.07196*, 2016.
- [12] Chanyoung Park, Raphael T Haftka, and Nam H Kim. Remarks on multi-fidelity surrogates. *Structural and Multidisciplinary Optimization*, 55(3):1029–1050, 2017.
- [13] Matthias Poloczek, Jialei Wang, and Peter I Frazier. Multi-information source optimization. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 4291–4301, 2017.
- [14] Jian Wu, Saul Toscano-Palmerin, Peter I Frazier, and Andrew Gordon Wilson. Practical multi-fidelity bayesian optimization for hyperparameter tuning. In *Uncertainty in Artificial Intelligence*, pages 788–798. PMLR, 2020.
- [15] Shibo Li, Wei Xing, Robert Kirby, and Shandian Zhe. Multi-fidelity bayesian optimization via deep neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.
- [16] David Ginsbourger and Rodolphe Le Riche. Towards gaussian process-based optimization with finite time horizon. In *mODA 9—Advances in Model-Oriented Design and Analysis*, pages 89–96. Springer, 2010.
- [17] Remi Lam, Karen Willcox, and David H Wolpert. Bayesian optimization with a finite budget: An approximate dynamic programming approach. *Advances in Neural Information Processing Systems*, 29:883–891, 2016.
- [18] Remi Lam and Karen Willcox. Lookahead bayesian optimization with inequality constraints. In *NIPS*, pages 1890–1900, 2017.
- [19] Warren B Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality*, volume 703. John Wiley & Sons, 2007.
- [20] Michael A Osborne, Roman Garnett, and Stephen J Roberts. Gaussian processes for global optimization. In *3rd international conference on learning and intelligent optimization (LION3)*, pages 1–15. Citeseer, 2009.
- [21] Javier González, Michael Osborne, and Neil Lawrence. Glasses: Relieving the myopia of bayesian optimisation. In *Artificial Intelligence and Statistics*, pages 790–799. PMLR, 2016.
- [22] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer school on machine learning*, pages 63–71. Springer, 2003.
- [23] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.
- [24] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.
- [25] Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(6), 2012.
- [26] Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient bayesian optimization. In *International Conference on Machine Learning*, pages 3627–3635. PMLR, 2017.
- [27] José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. *arXiv preprint arXiv:1406.2541*, 2014.

- [28] Dimitri Bertsekas. *Dynamic programming and optimal control: Volume I*, volume 1. Athena scientific, 1995.
- [29] James T Wilson, Frank Hutter, and Marc Peter Deisenroth. Maximizing acquisition functions for bayesian optimization. *arXiv preprint arXiv:1805.10196*, 2018.
- [30] Michael D McKay. Latin hypercube sampling as a tool in uncertainty analysis of computer models. In *Proceedings of the 24th conference on Winter simulation*, pages 557–564, 1992.
- [31] Alexander Forrester, Andras Sobester, and Andy Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.
- [32] L Mainini, A Serani, MP 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. *arXiv preprint arXiv:2204.07867*, 2022.
- [33] Shifeng Xiong, Peter ZG Qian, and CF Jeff Wu. Sequential design and analysis of high-accuracy and low-accuracy computer codes. *Technometrics*, 55(1):37–46, 2013.