ASME Journal of Mechanical Design
Online journal at:
https://asmedigitalcollection.asme.org/mechanicaldesign

Zahra Zanjani Foumani
Department of Mechanical and Aerospace Engineering,
University of California,
Irvine, CA 92697
e-mail: zzanjani@uci.edu

Amin Yousefpour
Department of Mechanical and Aerospace Engineering,
University of California,
Irvine, CA 92697
e-mail: yousefpo@uci.edu

Mehdi Shishehbor
Department of Mechanical and Aerospace Engineering,
University of California,
Irvine, CA 92697
e-mail: mshisheh@uci.edu

Ramin Bostanabad[1]
Department of Mechanical and Aerospace Engineering,
University of California,
Irvine, CA 92697
e-mail: raminb@uci.edu

Safeguarding Multi-Fidelity Bayesian Optimization Against Large Model Form Errors and Heterogeneous Noise

Bayesian optimization (BO) is a sequential optimization strategy that is increasingly employed in a wide range of areas such as materials design. In real-world applications, acquiring high-fidelity (HF) data through physical experiments or HF simulations is the major cost component of BO. To alleviate this bottleneck, multi-fidelity (MF) methods are used to forgo the sole reliance on the expensive HF data and reduce the sampling costs by querying inexpensive low-fidelity (LF) sources whose data are correlated with HF samples. However, existing multi-fidelity BO (MFBO) methods operate under the following two assumptions that rarely hold in practical applications: (1) LF sources provide data that are well correlated with the HF data on a global scale, and (2) a single random process can model the noise in the MF data. These assumptions dramatically reduce the performance of MFBO when LF sources are only locally correlated with the HF source or when the noise variance varies across the data sources. In this paper, we view these two limitations and uncertainty sources and address them by building an emulator that more accurately quantifies uncertainties. Specifically, our emulator (1) learns a separate noise model for each data source, and (2) leverages strictly proper scoring rules in regularizing itself. We illustrate the performance of our method through analytical examples and engineering problems in materials design. The comparative studies indicate that our MFBO method outperforms existing technologies, provides interpretable results, and can leverage LF sources which are only locally correlated with the HF source.
[DOI: 10.1115/1.4064160]

Keywords: Bayesian optimization, multi-fidelity modeling, emulation, Gaussian process, interval score, data-driven design, design of engineered materials system, design optimization, design representation

1 Introduction

Bayesian optimization (BO) is a sequential and sample-efficient global optimization technique that is increasingly used in the optimization of expensive-to-evaluate (and typically black-box) functions [1–3]. BO has two main ingredients: an emulator which is typically a Gaussian process (GP) and an acquisition function (AF) [4,5]. The first step in BO is to train an emulator on some initial data. Then, an auxiliary optimization is solved to determine the new sample that should be added to the training data. The objective function of this auxiliary optimization is the AF whose evaluation relies on the emulator. Given the new sample, the training data are updated and the entire emulation-sampling process is repeated until the convergence conditions are met [6–9].

Although BO is a highly efficient technique, the total cost of optimization can be substantial if it solely relies on the accurate but expensive high-fidelity (HF) data source. To mitigate this issue, multi-fidelity (MF) techniques are widely adopted [10–14] where one uses multiple data sources of varying levels of accuracy and cost in BO. The fundamental principle behind MF techniques is to exploit the correlation between low-fidelity (LF) and HF data to decrease the overall sampling costs [11,15,16]. Compared to single-fidelity BO (SFBO), the choice of the emulator is more important in multi-fidelity BO (MFBO) due to the MF nature of the data. In this regard, we note that most works on SFBO or MFBO leverage variations of Gaussian processes (GPs) for emulations since in scarce data regimes other methods such as probabilistic neural networks suffer overfitting, over confidence, or long training times [6,17–21].

Over the past two decades, many MFBO strategies have been proposed. However, each of these methods has some major drawbacks that are mainly rooted in their emulation strategy which fails to consider some features of MF data sets. For example, many existing MF techniques require prior knowledge about the hierarchy of LF data sources [22–26] and hence they break down

[1]Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received June 15, 2023; final manuscript received November 16, 2023; published online December 18, 2023. Assoc. Editor: Douglas Allaire.

when one does not know the relative accuracy of the LF sources. The MF modeling methods defined in Refs. [26–29] struggle to capture intricate correlations among different fidelities as separate emulators are trained for each data source. Kennedy and O'Hagan's bi-fidelity approach and its extensions [30–33] are limited to bi-fidelity cases and presume simple bias forms (e.g., an additive function [30,34–36]) for the LF sources. Co-Kriging and its extensions [23,37–42] fail to accurately capture cross-source correlations. `Botorch` [43] which is a widely popular MFBO packages is sensitive to the sampling costs (where highly inexpensive LF sources are heavily sampled which, in turn, causes numerical and convergence issues) and also requires a prior knowledge about the hierarchy of data sources. The above-reviewed methods also fail to directly handle categorical variables which frequently arise in applications such as materials design.

The recent work in Ref. [44] addresses most of the above limitations with two contributions. First, it achieves MF emulation via latent map Gaussian processes (LMGPs) which can simultaneously fuse any number of data sources, do not require any prior knowledge about the hierarchy of the data sources, can handle categorical variables, and do not use any simplification assumptions (e.g., linear correlation among sources, additive biases, etc.) while fusing MF data sets. Second, it quantifies the information value of LF and HF samples differently to consider the MF nature of the data while exploring the search space. The AF used in Ref. [44] is cost-aware in that it considers the sampling cost in quantifying the value of HF and LF data points. Henceforth, we refer to this method as `MFBO`.

While `MFBO` performs quite well, it shares two limitations with other MFBO methods. To demonstrate the first one, we consider a simple 1D example in Fig. 1 where each of the two LF sources is more correlated with the HF function in half of the domain. Specifically, LF1 and LF2 are only correlated with the HF source in the left and right regions, respectively, see Fig. 1(*a*). Before starting BO, the first step in `MFBO` is excluding highly biased LF sources from BO with the rationale that they can steer the search process in the wrong direction. This decision is made based on the fidelity manifold that LMGP learns using the initial data. In this manifold, each data source is encoded with a point and distances between these points are inversely related to the *global* correlations between the corresponding data sources, see Fig. 1(*b*) and Ref. [44]. So, based on Fig. 1(*b*), the HF source is barely correlated with the LF sources even though they are close to the HF function in half of the domain (since eliminating both sources converts the BO into a single-fidelity process in this example, we assume that `MFBO` only excludes LF2 and samples from the other two sources as shown in Fig. 1(*a*)). This is obviously a sub-optimal decision as it precludes the possibility of leveraging an LF source that is valuable in a small portion of the search space which may include the global optimum of the HF source (in the example of Fig. 1(*a*), `MFBO` should ideally leverage LF2 but *mostly* sample from it in the $x>5$ region). Hence, the first limitation of existing methods is their inability to leverage LF sources which are only locally correlated with the HF source.

The second limitation of existing MFBO methods (including `MFBO`) is that they assume all sources are corrupted with the same noise process (with unknown noise variance). However, MF datasets typically have different levels of noise especially if some sources represent deterministic computer simulations while others are physical experiments [45,46]. In such applications, the emulators used in existing MFBO methods overestimate the uncertainties associated with the noise-free data sources which, in turn, adversely affects the exploration property of BO.

In this paper, we introduce $\texttt{MFBO}_{\texttt{UQ}}$ which addresses the two aforementioned limitations of existing technologies because it (1) never discards an LF source (regardless of the magnitude of its global bias with respect to the HF source) to leverage it in parts of the domain where its samples are locally correlated with the

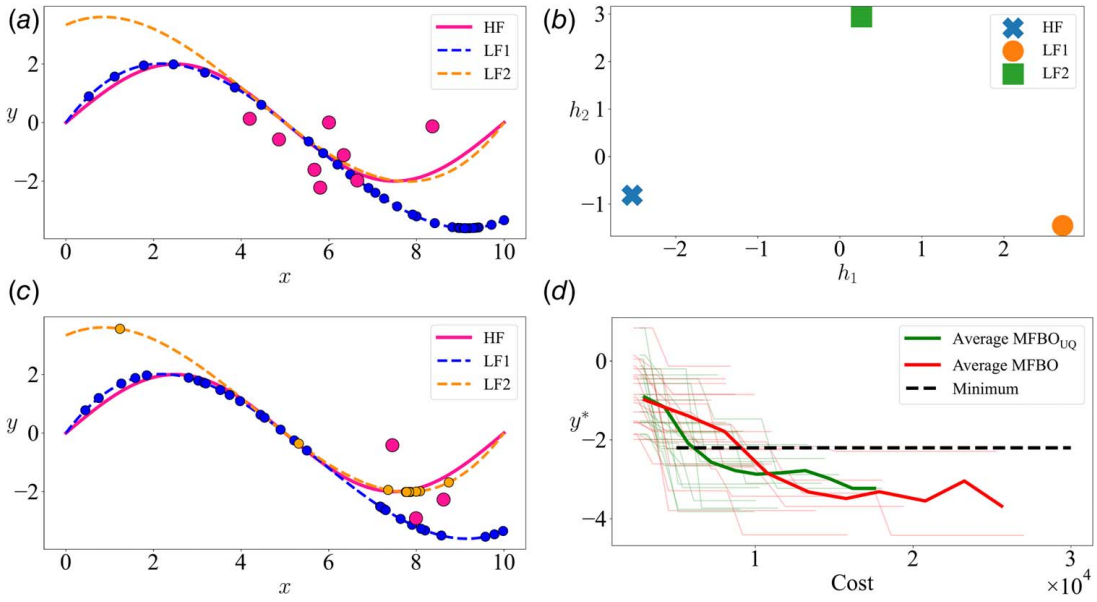

Fig. 1 Comparison between `MFBO` and $\texttt{MFBO}_{\texttt{UQ}}$: HF data are noisy ($\sigma_{noise}=1$) and expensive while the LF data are deterministic and cheap (sampling costs from the HF and two LF sources are 10/1/1). In this example, LF1 is more correlated with the HF source for $x<5$ while LF2 has a higher fidelity for $x>5$: (*a*) demonstrates the sampling history of `MFBO` assuming LF2 is excluded from the sampling process (see the text for the reason), (*b*) visualizes the fidelity manifold learnt by an LMGP that is trained on the initial data. Each point in this manifold encodes a data source and the distances among these points quantify global correlations among the corresponding data sources, (c) $\texttt{MFBO}_{\texttt{UQ}}$ is proposed in this paper and effectively explores the space via LF samples which are either highly correlated with HF data, or provide small function values, and (*d*) $\texttt{MFBO}_{\texttt{UQ}}$ outperforms `MFBO` in finding the optimum of HF source (y^*) for various initial conditions (the large noise variance of HF data causes both approaches to have some errors upon convergence). Initial data are not shown in (*a*) and (c).

HF data, and (2) estimates a separate noise process for each data source. More specifically, we view these two limitations as uncertainty sources and we address them by reformulating the training process of $\texttt{MFBO}_{\texttt{UQ}}$'s emulator to improve it's uncertainty quantification (UQ) ability. We argue that this improvement in the emulator helps MFBO in balancing exploration and exploitation more effectively. Figure 1(*c*) schematically demonstrates the advantages of $\texttt{MFBO}_{\texttt{UQ}}$ over MFBO in a 1D example where there are one HF and two LF sources. As it can be observed the LF sources are mostly sampled where they either are well correlated with the HF source, or provide attractive function values (e.g., very small values in minimization). The advantages of $\texttt{MFBO}_{\texttt{UQ}}$ over MFBO in finding the optimum of HF (y^*) hold over various initializations, see Fig. 1(*d*).

The rest of the paper is organized as follows. We provide the methodological details in Sec. 2 and then evaluate the performance of $\texttt{MFBO}_{\texttt{UQ}}$ via multiple ablation studies in Sec. 3. In Sec. 3, we also visualize how strategic sampling in $\texttt{MFBO}_{\texttt{UQ}}$, driven by accurate uncertainty quantification and effective handling of biased data sources, results in its superior performance compared to MFBO. This is demonstrated through two real-world high-dimensional material design examples with noisy and highly biased data sources. We conclude the paper in Sec. 4 by summarizing our contributions and providing future research directions.

2 Methods

In this section, we first provide some background on LMGP and MF modeling with LMGP in Secs. 2.1 and 2.2, respectively. We then propose our efficient mechanism for inversely learning a noise process for each data source in Sec. 2.3. Next, we introduce the cost-aware AF of $\texttt{MFBO}_{\texttt{UQ}}$ in Sec. 2.4. Finally, in Sec. 2.5 we elaborate on our idea that improves the UQ capabilities of LMGPs and, in turn, benefits MFBO.

2.1 Latent Map Gaussian Process. GPs are emulators which assume the responses or outputs in the training data come from a multivariate normal distribution with parametric mean and covariance functions that depend on the inputs. Based on this assumption, the following equation can be written:

$$y(\boldsymbol{x}) = \beta + \xi(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_{\mathrm{dx}}]^T$ is the input vector, $y(\boldsymbol{x})$ is the output, β is an unknown coefficient, and $\xi(\boldsymbol{x})$ is a zero-mean GP with the covariance function:

$$\mathrm{cov}(\xi(\boldsymbol{x}),\ \xi(\boldsymbol{x}')) = c(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 r(\boldsymbol{x}, \boldsymbol{x}') \tag{2}$$

where σ^2 is the variance of the process and $r(\cdot, \cdot)$ is the parametric correlation function which measures the distance between any two input vectors. In this paper, we use the Gaussian correlation function defined as

$$r(\boldsymbol{x}, \boldsymbol{x}') = \exp\left\{-\sum_{i=1}^{\mathrm{dx}} 10^{\omega_i}\ (x_i - x_i')^2\right\} \tag{3}$$

where $\boldsymbol{\omega} = [\omega_1, \omega_2, \ldots, \omega_{\mathrm{dx}}]^T$ are the scale parameters. To directly use GPs in MF modeling, we follow Ref. [47] who convert MF modeling to a manifold learning problem via LMGPs which are extensions of GPs that can handle categorical data [48] while providing a visualizable manifold that can be used to interpret the global correlations among the data sources.

Denoting the categorical inputs by $\boldsymbol{t} = [t_1, t_2, \ldots, t_{\mathrm{dt}}]^T$ where variable t_i has l_i distinct levels, LMGP maps each combination of the categorical levels to a point in a learned quantitative manifold. To this end, LMGP assigns a unique vector to each combination of the categorical variables and then uses a parametric function to map these unique vectors into a compact manifold with dimensionality dz. Assuming a linear transformation is used in LMGP, the mapping operation reads as

$$\boldsymbol{z}(\boldsymbol{t}) = \boldsymbol{\zeta}(\boldsymbol{t})\boldsymbol{A} \tag{4}$$

where $\boldsymbol{t}$ denotes a specific combination of the categorical variables, $\boldsymbol{z}(\boldsymbol{t})$ is the $1 \times \mathrm{d}z$ posterior latent representation of $\boldsymbol{t}$, $\boldsymbol{\zeta}(\boldsymbol{t})$ is a unique prior vector representation of $\boldsymbol{t}$, and $\boldsymbol{A}$ is a rectangular matrix that maps $\boldsymbol{\zeta}(\boldsymbol{t})$ to $\boldsymbol{z}(\boldsymbol{t})$. In this paper, grouped one-hot encoding is used to generate the prior vectors and hence the dimensionality of $\boldsymbol{\zeta}(\boldsymbol{t})$ and $\boldsymbol{A}$ are $1 \times \sum_{i=1}^{\mathrm{dt}} l_i$ and $\sum_{i=1}^{\mathrm{dt}} l_i \times \mathrm{d}z$, respectively. These mapped points can now be directly embedded in the correlation function as

$$r(\boldsymbol{u}, \boldsymbol{u}') = \exp\left\{-\sum_{i=1}^{\mathrm{dx}} 10^{\omega_i}\ (x_i - x_i')^2 - \sum_{i=1}^{\mathrm{dz}} (z_i(\boldsymbol{t}) - z_i(\boldsymbol{t}'))^2\right\} \tag{5}$$

where $\boldsymbol{u} = [\boldsymbol{x};\ \boldsymbol{t}]$ and $\boldsymbol{z}(\boldsymbol{t}) = [z_1(\boldsymbol{t}),\ z_2(\boldsymbol{t}), \ldots, z_{\mathrm{dz}}(\boldsymbol{t})]$ is the location in the learned latent space corresponding to the specific combination of the categorical variables denoted by $\boldsymbol{t}$.

LMGP estimates the hyperparameters (β, $\boldsymbol{A}$, $\boldsymbol{\omega}$, σ^2) via maximum a posteriori (MAP) which, assuming $\mathrm{d}z = 2$, provides point-estimates for $\mathrm{d}x + 2 \times \sum_{i=1}^{\mathrm{dt}} l_i + 2$ variables. Upon parameter estimation, LMGP uses the conditional distribution formulas to predict the response distribution at the arbitrary point $\boldsymbol{u}$ with the following mean and variance:

$$\mathbb{E}[y(\boldsymbol{u})] = \mu(\boldsymbol{u}) = \hat{\beta} + \boldsymbol{r}^T(\boldsymbol{u})\boldsymbol{R}^{-1}(\boldsymbol{y} - \boldsymbol{1}_{n\times 1}\hat{\beta}) \tag{6}$$

$$c(y(\boldsymbol{u}), y(\boldsymbol{u})) = \sigma^2(\boldsymbol{u}) = \hat{\sigma}^2\left(1 - \boldsymbol{r}^T(\boldsymbol{u})\boldsymbol{R}^{-1}\boldsymbol{r}(\boldsymbol{u})\right) \tag{7}$$

where n is number of training samples, $\mathbb{E}$ denotes expectation, $\boldsymbol{1}_{a\times b}$ is an $a \times b$ matrix of ones, $\boldsymbol{r}(\boldsymbol{u})$ is an $n \times 1$ vector with the ith element $r(\boldsymbol{u}^i, \boldsymbol{u})$, and $\boldsymbol{R}$ is an $n \times n$ matrix with $R_{ij} = r(\boldsymbol{u}^i, \boldsymbol{u}^j)$.

2.2 Multi-Fidelity Emulation Via LMGP. The first step to MF emulation with LMGP is to augment the inputs with the additional *categorical* variable s that indicates the source of a sample, i.e., $s = \{'1', '2', \ldots, 'ds'\}$ where the jth element corresponds to source j for $j = 1, \ldots, ds$. Subsequently, the training data from all sources are concatenated and used in LMGP to build an MF emulator. Upon training, to predict the objective value of at point $\boldsymbol{x}$ from source j, $\boldsymbol{x}$ is concatenated with the categorical variable s that corresponds to source j and fed into the trained LMGP. We refer the readers to Ref. [44] for more detail but note here that in case the input variables already contain some categorical features (see Sec. 3.2 for an example), we endow LMGP with two manifolds where one encodes the fidelity variable s while the other manifold encodes the rest of the categorical variables. While this choice does not noticeably affect the accuracy of LMGP during test time, it increases interpretability. For instance, we use the learned manifold for the categorical variables in Sec. 3.2 to show the trajectory of BO in the design space.

It has been recently shown [48] that LMGPs have the following primary advantages over other MF emulators: (1) they provide a more flexible and accurate mechanism to build MF emulators since they learn the relations between the sources in a nonlinear manifold, (2) they learn all the sources quite accurately rather than just emulating the HF source, and (3) they provide a visualizable global metric for comparing the relative discrepancies/similarities among the data sources.

2.3 Source-Dependent Noise Modeling. The presence of noise significantly affects the performance of BO and incorrectly modeling it can cause over-exploration or under-exploration of the search space. To mitigate the effects of noise in BO, we reformulate LMGPs to independently model a noise process for each data source. This reformulation improves emulation accuracy and, in turn, improves the search process when LMGP is deployed in MFBO.

To model noise in GPs, the nugget or jitter parameter, δ, is used [49] to replace $\boldsymbol{R}$ with $\boldsymbol{R}_\delta = \boldsymbol{R} + \delta \boldsymbol{I}$ where $\boldsymbol{I}$ is an $n \times n$ identity matrix. With this approach, the estimated stationary noise variance in the data is $\delta\sigma^2$ and the mean and variance formulations in Eqs. (6) and (7) are modified by using $\boldsymbol{R}_\delta$ instead of $\boldsymbol{R}$.

Although incorporating this modification in the correlation matrix can enhance the performance of the emulator and BO in single-fidelity (SF) problems, it does not yield the same benefits in MF optimization. This is likely because of the dissimilar nature of the data sources and their corresponding noises. When dealing with multiple sources of data, each source may suffer from different levels and types of noise. Consider a bi-fidelity dataset where the HF data come from an experimental setup and are subject to measurement noise, while the LF data are generated by a deterministic computer code which has a systematic bias due to missing physics. In this case, using only one nugget parameter in LMGP for MF emulation is obviously not an optimum choice.

To address this issue effectively, we propose to use multiple nugget parameters in the emulator. Specifically, we define the nugget vector $\boldsymbol{\delta} = [\delta_1, \ \delta_2, \ \ldots, \ \delta_{\mathrm{d}s}]$ and update the correlation matrix as follows:

$$\boldsymbol{R}_\delta = \boldsymbol{R} + \boldsymbol{N}_\delta \tag{8}$$

where $\boldsymbol{N}_\delta$ denotes an $n \times n$ diagonal matrix whose (i, i)th element is the nugget element corresponding to the data source of the ith sample. For instance, suppose the ith sample ($\boldsymbol{u}^i$) is generated by source ds. Then, (i, i)th element of $\boldsymbol{N}_\delta$ is $\delta_{\mathrm{d}s}$. With this modification, the estimated stationary noise variance for the samples in each data source is $\delta_i\sigma^2$.

Then, we use Eq. (8) to build the correlation matrix of LMGP and jointly estimate all the parameters via MAP as

$$[\widehat{\beta}, \ \hat{\sigma}^2, \ \widehat{\boldsymbol{\omega}}, \ \widehat{\boldsymbol{A}}, \ \widehat{\boldsymbol{\delta}}] = \underset{\beta, \sigma^2, \boldsymbol{\omega}, \boldsymbol{A}}{\operatorname{argmin}} \frac{n}{2}\log(\sigma^2) + \frac{1}{2}\log(|\boldsymbol{R}_\delta|) + \frac{1}{2\sigma^2}(\boldsymbol{y} - \boldsymbol{M}\beta)^T \boldsymbol{R}_\delta^{-1}(\boldsymbol{y} - \boldsymbol{M}\beta) + \log \underset{\beta, \sigma^2, \boldsymbol{\omega}, \boldsymbol{A}, \boldsymbol{\delta}}{(\mathrm{P}(\cdot))} \tag{9}$$

where $p(\cdot)$ is the prior of the hyperparameters. We define independent priors for each parameter where $\omega_i \sim N(-3, 3)$, $\beta \sim N(0, 1)$, $A_{ij} \sim N(0, 1)$, $\sigma^2 \sim LN(0, 1)$,[2] and $\delta_i \sim LHS(0, 0.01)$[3] [50]. Our multi-noise approach increases the number of LMGP's hyperparameters to $\mathrm{d}x + 2 \times \sum_{i=1}^{\mathrm{d}t} l_i + 2 + \mathrm{d}s$. We highlight that the above formulations cannot learn an input-dependent noise variance which requires the nugget to be a function of $\boldsymbol{x}$. We make this choice due to the fact that our emulator is used in small-data applications where modeling an input-dependent noise can result into overfitting since it increases the number of hyperparameters by at least $\mathrm{d}s \times \mathrm{d}x$.

2.4 Multi-Source Cost-Aware Acquisition Function. The choice of AF is crucial in BO as it guides the sampling process by balancing exploration and exploitation. Exploration involves searching unseen regions (where the emulator naturally provides large prediction intervals) while exploitation focuses on regions of the input space where good designs are already observed. A mere focus on exploitation causes convergence to local optima while excessive exploration increases the sampling costs and delays the convergence. The choice of AF is even more important in MFBO, since the AF must consider the biases of LF data and source-dependent sampling costs in addition to balancing exploration and exploitation. To capture these goals, separate AFs are defined in Ref. [44] for LF and HF sources where the cheap LF sources are primarily used for exploration while the expensive HF samples are maximally exploited. Following this idea, the AF of the jth LF source ($j \neq l$, l denotes the HF source) is defined as the exploration part of the expected improvement (EI) in MFBO

$$\gamma_{LF}(\boldsymbol{u}; j) = \sigma_j(\boldsymbol{u})\phi\left(\frac{y_j^* - \mu_j(\boldsymbol{u})}{\sigma_j(\boldsymbol{u})}\right) \tag{10}$$

where y_j^* is the best function value obtained so far from source j and $\phi(\cdot)$ denotes the probability density function of the standard normal variable. $\sigma_j(\boldsymbol{u})$ and $\mu_j(\boldsymbol{u})$ are the standard deviation and mean, respectively, of point $\boldsymbol{u}$ from source j which we estimate via

$$\mu(\boldsymbol{u}) = \hat{\beta} + \boldsymbol{r}^T(\boldsymbol{u})\boldsymbol{R}_\delta^{-1}(\boldsymbol{y} - \boldsymbol{1}_{n\times 1}\hat{\beta}) \tag{11}$$

$$c\big(y(\boldsymbol{u}), y(\boldsymbol{u})\big) = \sigma^2(\boldsymbol{u}) = \hat{\sigma}^2\big(1 - \boldsymbol{r}^T(\boldsymbol{u})\boldsymbol{R}_\delta^{-1}\boldsymbol{r}(\boldsymbol{u})\big) + \hat{\delta}_j \tag{12}$$

MFBO utilizes improvement as the AF for the HF data source since it is computationally efficient and emphasizes exploitation. Accordingly, $\mathrm{MFBO}_{\mathrm{UQ}}$ uses improvement for the HF source (source l) with the new mean calculated based on the Eq. (11)

$$\gamma_{HF}(\boldsymbol{u}; l) = \mu_l(\boldsymbol{u}) - y_l^* \tag{13}$$

In each iteration of BO, we first use the mentioned AFs to solve ds auxiliary optimizations to find the candidate points with the highest acquisition value from each source. We then scale these values by the corresponding sampling costs to obtain the following composite AF:

$$\gamma_{\mathrm{MFBO_{UQ}}}(\boldsymbol{u}; j) = \begin{cases} \dfrac{\gamma_{LF}(\boldsymbol{u}; j)}{O(j)} & j = [1, \ldots, \mathrm{d}s] \ \& \ j \neq l \\ \dfrac{\gamma_{HF}(\boldsymbol{u}; l)}{O(l)} & j = l \end{cases} \tag{14}$$

where $O(j)$ is the cost of acquiring one sample from source j. We determine the final candidate point (and the source that it should be sampled from) at iteration $k+1$ via

$$[\boldsymbol{u}^{k+1}, j^{k+1}] = \underset{\boldsymbol{u}, j}{\operatorname{argmax}} \gamma_{\mathrm{MFBO_{UQ}}}(\boldsymbol{u}; j) \tag{15}$$

2.5 Emulation for Exploration. The composite AF in Eq. (14) quantifies the information value of LF samples via Eq. (10) whose value scales with the prediction uncertainties, i.e., $\sigma(\boldsymbol{u})$. The source-dependent noise modeling of Sec. 2.3 improves LMGP's ability in learning the uncertainty by introducing a few more hyperparameters. However, the added hyperparameters may result in overfitting and, in turn, deteriorate the predicted uncertainties [51,52]. A related issue is the effect of large local biases of LF sources which can inflate the uncertainty quite substantially and, as a result, increase $\gamma_{LF}(\boldsymbol{u}; j)$. This increase causes MFBO to repeatedly sample from the biased LF sources. Such repeated samplings reduce the efficiency of MFBO and may cause numerical issues (due to ill-conditioning of the covariance matrix) or even convergence to a sub-optimal solution.

To address the above issues simultaneously, we argue that the training process of the emulator should increase the importance of UQ which directly affects the exploration part of MFBO. To this end, we leverage strictly proper scoring rules while training LMGPs.

Scoring rules [53] evaluate a probabilistic prediction by assigning a numerical score to it. The scoring rule of an emulator is (strictly) proper if matching the predicted distribution with the underlying sample distribution (uniquely) maximizes the expected score for any sample. The probabilistic nature of LMGP's prediction motivates us to use the negatively oriented interval score (hereafter denoted by *IS*) to evaluate the UQ capabilities of LMGPs. We choose *IS* since it is robust to outliers, rewards narrow prediction intervals, and is flexible in the choice of desired coverage levels [54,55]. *IS* is a special case of quantile prediction that penalizes the model for each observation that is not inside the $(1 - v) \times$

[2]Log-Normal.
[3]Log-Half-Horseshoe with zero lower bound and scale parameter 0.01.

100% prediction interval. The lower ($\mathcal{L}^i$) and upper $\mathcal{U}^i$ endpoints of this prediction interval for the ith observation are their predictive quantiles at levels $v/2$ and $1-v/2$, respectively. So, we calculate the *IS* as

$$IS_v = \frac{1}{n}\sum_{i=1}^{n}(\mathcal{U}^i - \mathcal{L}^i) + \frac{2}{v}(\mathcal{L}^i - y(\boldsymbol{u}^i))\mathbb{1}\{y(\boldsymbol{u}^i) < \mathcal{L}^i\} + \frac{2}{v}(y(\boldsymbol{u}^i) - \mathcal{U}^i)\mathbb{1}\{y(\boldsymbol{u}^i) > \mathcal{U}^i\} \quad (16)$$

where $\mathbb{1}\{\cdot\}$ is an indicator function which is 1 if its condition holds and zero otherwise [56,57]. We use $v = 0.05$ (95% prediction interval), so $\mathcal{U}^i = \mu(\boldsymbol{u}^i) + 1.96\sigma(\boldsymbol{u}^i)$ and $\mathcal{L}^i = \mu(\boldsymbol{u}^i) - 1.96\sigma(\boldsymbol{u}^i)$.

Having defined the *IS*, we now formulate the new objective function for training LMGPs where $IS_{0.05}$ is used as a penalty term during hyperparameter estimation to increase the focus on UQ. Since the effectiveness of this penalization mechanism depends on the value of the posterior, we introduce an adaptive coefficient whose magnitude depends on the posterior value. With this penalty term, we estimate the hyperparameters of LMGP via

$$[\hat{\beta}, \widehat{\sigma^2}, \widehat{\boldsymbol{\omega}}, \widehat{\boldsymbol{A}}, \widehat{\boldsymbol{\delta}}] = \underset{\beta,\sigma^2,\boldsymbol{\omega},\boldsymbol{A},\boldsymbol{\delta}}{\operatorname{argmin}} \; L_{MAP} + \varepsilon|L_{MAP}| \times IS_{0.05} \quad (17)$$

where $|\cdot|$ denotes the absolute function and ε is a user-defined scaling parameter. In this paper, we use $\varepsilon = 0.08$ for all of our examples.

3 Results and Discussion

We demonstrate the performance of $\texttt{MFBO}_{\texttt{UQ}}$ on two analytic examples (see Table 1 in the Supplementary Information available in the Supplemental Materials on the ASME Digital Collection for details on functional forms, size of initial data, sampling costs, and number of LF sources and their accuracy with respect to the HF source) and two real-world problems. For analytic examples, we compare the results against `Botorch`, `MFBO`, and SFBO. $\texttt{MFBO}_{\texttt{UQ}}$ and `MFBO` use the AFs introduced in Sec. 2.4 while SFBO uses EI as its AF and LMGP as its emulator. `Botorch` employs single-task multi-fidelity GP and knowledge gradient as its emulator and AF, respectively [58,59]. All the baselines except `Botorch` are also used for engineering examples. `Botorch` is not applicable to them since it cannot handle categorical variables and also the y_l^* values determined by `Botorch` are not obtained through direct sampling from the available data sets (rather, the samples are obtained by optimizing the learned posterior).

We assume that the cost of querying any of the data sources is much higher than the computational costs of BO (i.e., fitting LMGP and solving the auxiliary optimization problem). Therefore, we compare the methods based on their capability to identify the global optimum of the HF source and the overall data collection cost. By comparing these methods, we aim to demonstrate: (1) the advantages of estimating noise process for each data source, (2) that using IS improves the accuracy of LMGP and, in turn, enhances the convergence of BO (since our defined AFs highly rely on the quality of the prediction), and (3) that deploying IS eliminates the need for excluding highly biased LF sources from BO.

We use the same stop conditions across all the baselines to clearly demonstrate the benefits of our two contributions. In particular, the optimization is stopped when either of the following happens: (1) the overall sampling cost exceeds a pre-determined maximum budget, or (2) the best HF sample does not change over 50 iterations. The maximum budget for the analytical examples is 40,000 units, while it is 1000 and 1800 for the two real-world examples. These budgets are chosen based on the data collection costs.

3.1 Analytical Examples.

We consider two analytical examples, `Wing` [60] and `Borehole` [61], whose input dimensionality is 10 and 8, respectively. To challenge the convergence and better illustrate the power of separate noise estimation, we only add noise to the HF data (the noise variance is defined based on the range of each function). The added noise variance to the HF source of `Wing` and `Borehole` are 9 and 16, respectively. Both examples are single response and details regarding their formulation, initialization, and sampling cost is presented in SI A available in the Supplemental Materials). To assess the robustness of the results and quantify the effect of random initial data, we repeat the optimization process 20 times for each example with each of the baselines (all initial data are generated via Sobol sequence).

In each example, the relative root mean squared error is calculated between LF sources and their corresponding HF source based on 10,000 samples to show the relative accuracy of the LF sources (presented in Table 1 in SI available in the Supplemental Materials). Based on these ground truth numbers (which are *not* used in BO), in the case of `Borehole` the source ID, true fidelity level, and sampling costs are not related (e.g., although the first LF source is the most expensive one, it has the least accuracy compared to the HF source). In the case of `Wing`, however, these numbers match (e.g., LF1 is the most accurate and expensive LF source and is followed by LF2 and then LF3).

`MFBO` excludes the highly biased LF sources from BO before any new samples are obtained (also, during BO, the initial samples from highly biased LF sources are not used in emulation). This exclusion is done based on the latent map of the LMGP model that is trained on the initial data. Figure 2 shows the latent maps of `Wing` and `Borehole` examples. As shown in Fig. 2, while all the fidelity sources of `Wing` are beneficial (since the points encoding the LF sources are very close to the HF point), the first two LF sources of `Borehole` are not correlated enough with the HF (their latent positions are distant from that of the HF) and hence are excluded in `MFBO`. However, $\texttt{MFBO}_{\texttt{UQ}}$ does not require this exclusion because it leverages the biased LF sources merely in the regions that they are correlated with the HF source. We also keep the biased sources in `Botorch` since there is no explicit requirement to exclude them within the package's documentation. In this paper, we do not exclude the biased sources from `MFBO` to have a comprehensive comparison with other baseline methods and, most notably, to effectively illustrate the impacts of our contributions.

Figure 3 summarizes the convergence history of each example by depicting the best HF sample (y^*) found by each method versus its accumulated sampling cost. We note that the initialization process is identical for all MFBO methods and the reason for observing different starting points for them is that we report y^* versus cumulative cost. More specifically, a method may take samples from any of the sources but this action may not improve y^* in which case the cumulative cost increases while y^* does not. We also note that SFBO has a different initialization since we must use more initial HF samples in SFBO to ensure its starting cost is comparable to the costs of MF methods which use both HF and LF data.

As we expect, $\texttt{MFBO}_{\texttt{UQ}}$ and `MFBO` outperform SFBO in `Wing` (Fig. 3(*a*)) by leveraging the inexpensive LF sources that are globally correlated with the HF source. However, the large added noise adversely affects the performance of `Botorch` in estimating the correlation among sources. This inaccurate correlation estimation combined with large cost differences among the data sources prevents `Botorch` from leveraging the correlated LF sources and causes convergence to the sub-optimal solution $y^* = 183.72$ while the ground truth is 123.25. The superiority of $\texttt{MFBO}_{\texttt{UQ}}$ is more evident in the `Borehole` example where there are highly biased LF sources. In `Borehole` (Fig. 3(*b*)), all the thin red curves (`MFBO`) are straight lines, except for two curves. This means that for 18 repetitions, the optimization process fails to improve. The reason behind this failure is that `MFBO` cannot handle the large bias of the LF sources and samples points that steer the optimization in the wrong direction. Consequently, `MFBO` cannot find any efficient HF sample with large enough information value (that justifies its high sampling cost) which results in the lack of improvement in

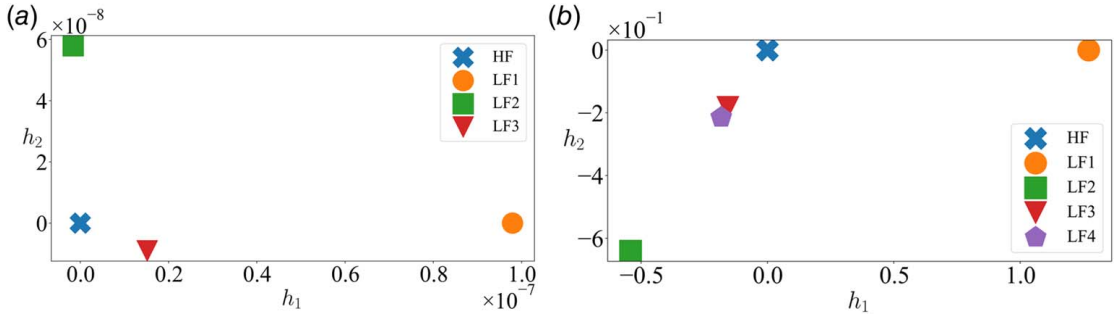

Fig. 2 Fidelity manifolds in analytic examples: The plots in (*a*) and (*b*) are obtained by fitting an LMGP to the initial data in the `Wing` and `Borehole` examples, respectively. Due to the consistency across the 20 repetitions, the plots are randomly chosen among them. In (*b*), the HF source is encoded far from LF1 and LF2 which indicates that these two sources have large biases with respect to the HF source. `MFBO` excludes these two sources from the BO while $\texttt{MFBO}_{\texttt{UQ}}$ does not.

y^*. Conversely, all the thin green curves ($\texttt{MFBO}_{\texttt{UQ}}$) converge to a value very close to the ground truth. Additionally, while efficient sampling from LF sources improves the performance of $\texttt{MFBO}_{\texttt{UQ}}$, the large added noise to the HF source adversely affects the performance of SFBO and results in a sub-optimal convergence.

As detailed in SI B.1 available in the Supplemental Materials, we note that unlike `Botorch`, the performance of $\texttt{MFBO}_{\texttt{UQ}}$ is robust to the sampling costs and local correlations. For instance, in `Borehole` (Fig. 3(*b*)), `Botorch` estimates the optimum as $y_l^* = 7.29$ while this value is $y_l^* = 4.14$ for $\texttt{MFBO}_{\texttt{UQ}}$ (the ground truth is 3.98). The reason behind this inaccuracy is that `Botorch` fails to find an HF sample whose information value is large enough to justify its high sampling cost and, as a result, cheap LF sources are largely queried. Additionally, due to the strong bias in two of the LF sources, `Botorch` fails to effectively sample them within the correlated domain. So, LF queries do not improve y_l^* and `Botorch` stops without finding the optimum.

3.2 Real-World Datasets. In this section, we study two materials design problems where the aim is to find the composition that best optimizes the property of interest. We do not add noise to these two examples as they are inherently noisy. The design space of both examples has categorical inputs (denoted by $\boldsymbol{t}$) and we add one more categorical variable (denoted by s) to enable data fusion as described in Sec. 2.2. We design our LMGP to map the categorical inputs onto two 2D manifolds (one for $\boldsymbol{t}$ and the other for s) to help with the visualization of the exploration–exploitation behavior of BO in the design space. The HF and LF data are obtained via simulations (based on the density functional theory) with different fidelity levels.

The first example is on designing a nanolaminate ternary alloy (`NTA`) which is used in applications such as high-temperature structural materials [62]. `NTA` is in the form of M_2AX where M is an early transition metal, A is a main group element, and X is either carbon or nitrogen. This problem is bi-fidelity where the goal is to find the member of `NTA` family with the largest bulk modulus. The HF and LF datasets have 224 samples each and are 10-dimensional (7 quantitative and 3 categorical where the latter have 10, 12, and 2 levels). The cost ratio between the HF and LF sources is 10/1 and we initialize the BO with 30 HF and 30 LF samples (the composition with the largest bulk modulus is never in the initial data). To quantify the sensitivity of the results to the random initial data, we repeat this process 20 times for each BO method.

Our second problem is on designing hybrid organic–inorganic perovskite (HOIP) crystals in the form of ABX_3 where B is occupied by metal cation, A can be organic or inorganic cation, and X denotes a choice of halide [63]. In this example, our goal is to find the compound with the smallest inter-molecular binding energy. There are three datasets (one from HF and two from LF sources) which have the same dimensionality (1 output and 3 categorical inputs with 10, 3, and 16 levels) but different sizes. The HF dataset has 480 samples while the first and second LF datasets have 179 and 240 samples, respectively. The cost ratio between the three sources is 15/10/5 (where the HF and LF2 sources are the most expensive and cheapest, respectively) and we initialize the BO with (15, 20, 15) samples for the HF and LF sources (the best compound is excluded from the initial data). We repeat the BO process 20 times to assess the sensitivity of the results to the initial data. As mentioned before, the first step in `MFBO` is to train an LMGP to the initial data in each problem to exclude the highly biased sources. As `NTA` has categorical variables, LMGP learns two manifolds. Based on Fig. 4(*a*), the latent points of the fidelity sources of `NTA` are very close in the learned fidelity manifold which indicates that there

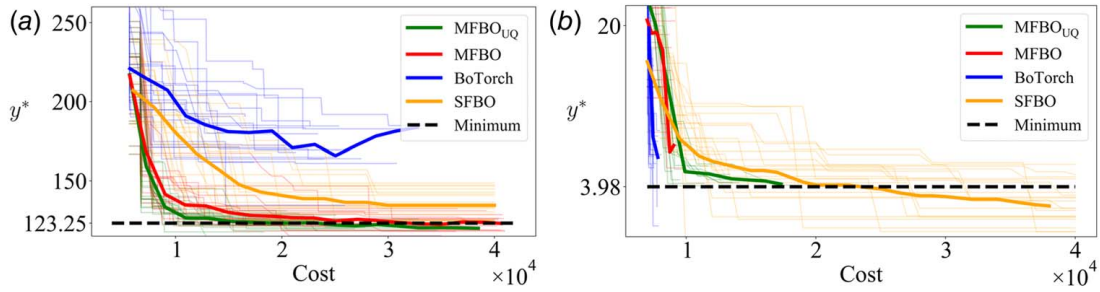

Fig. 3 Convergence histories in analytic examples: The plots depict the best HF sample found by each approach (y^*) versus their sampling costs accumulated during the BO iterations (the cost of initial data is included). (*a*) and (*b*) summarize the results for the `Wing` and `Borehole` examples, respectively. The thin curves show the convergence history of each repetition and the solid thick ones indicate the average behavior across the 20 repetitions. In both examples, $\texttt{MFBO}_{\texttt{UQ}}$ outperforms all other methods. The ground truth is represented by the black dashed line. (Color version online.)

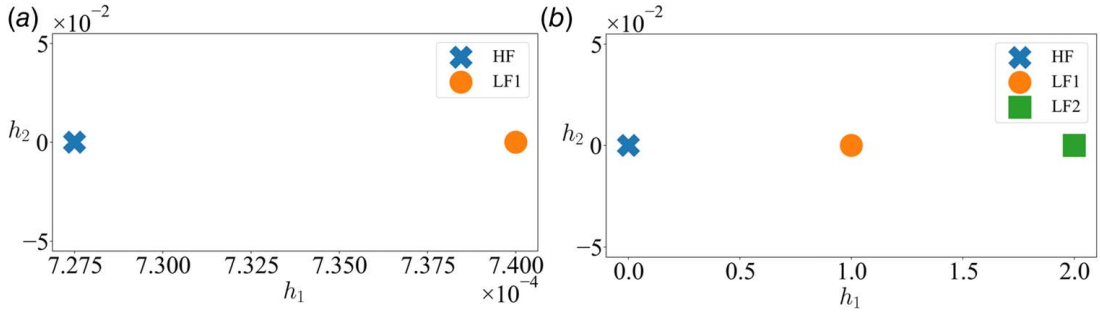

Fig. 4 Fidelity manifolds in real-world examples: The plots in (*a*) and (*b*) are obtained by fitting an LMGP to the initial data in the NTA and HOIP examples, respectively. Due to the consistency across the 20 repetitions, the plots are randomly chosen among them. In (*b*), the HF source is encoded far from LF sources which indicates that these two sources have large biases with respect to the HF source and MFBO should exclude them. However, by excluding these sources, the problem transforms into SF, rendering it incomparable to MFBO_{UQ}. To maintain comparability with MFBO_{UQ}, we retain the LF sources in MFBO.

is a high correlation between the corresponding two data sources. However, both latent points of LF sources in HOIP are far from the HF one so they both should be excluded due to their large global bias. By excluding both LF sources the MF problem in HOIP reduces to an SF one so we do not exclude the biased LF sources from HOIP to be able to compare the performance of MFBO_{UQ} with MFBO.

A summary of the convergence history of NTA and HOIP is depicted in Fig. 5 by showing the best HF sample (y^*) found by each method versus its accumulated sampling cost. The initialization is the same for all the MFBO methods and observing different starting points follows the same rationale mentioned for Fig. 3. In Fig. 5(*a*), the LF sources are globally correlated with the HF source and hence both MF methods perform better than SFBO by using inexpensive and informative LF data. Additionally, the higher prediction accuracy of the emulator of MFBO_{UQ} results in a more efficient sampling and faster convergence of BO in MFBO_{UQ} compared to MFBO. Regarding the spike in the convergence plot of MFBO in Fig. 5(*a*), we note that 18 repetitions converge at costs below 500. Consequently, the thick red line (which is the average across the 20 repetitions) becomes highly sensitive to the convergence values after cost exceeds 500 since it is an average of only two values. Specifically, in one of these two repetitions the best sample found is 237 for many iterations until the cost reaches 544 when MFBO suddenly converges to the ground truth (i.e., 255). This sudden convergence results in the spike in the corresponding history and, in turn, the average behavior captured by the thick red line.

The superiority of MFBO_{UQ} is more obvious in HOIP (see Fig. 5(*b*)) which has two highly biased LF sources. In this example, MFBO expectedly converges to a sub-optimal compound since both LF sources are only locally correlated with the HF source. So, the AFs fail to sample valuable points to improve the optimization as they cannot find the region where the LF sources are beneficial and informative. Additionally, each data source is obtained from a distinct process so it suffers from different types and levels of noise. Therefore, estimating a single noise for all the data sources in MFBO reduces the emulation accuracy and further exacerbates the performance of AFs. MFBO_{UQ} overcomes these issues by focusing more on UQ and estimating separate noise processes; resulting in a better performance compared to SFBO and especially to MFBO.

The 2D manifolds in Figs. 6 and 7 demonstrate the trajectory of BO in the categorical design space of each data source in NTA and HOIP, respectively. The top and bottom rows of these figures correspond to MFBO and MFBO_{UQ}. In these manifolds, each latent point indicates a compound and is color-coded based on the ground truth response value (i.e., the bulk modulus) from each source. The marker shapes in these manifolds indicate whether a compound is part of the initial data, sampled during BO, or never seen by LMGP. As expected, most markers are triangles which indicates that most combinations are never tested by either MFBO or MFBO_{UQ}. The red arrows next to the legend mark the response ranges in each data set which indicate that, unlike in Fig. 6 for NTA, the response ranges across the three sources are quite different in the HOIP problem.

To benefit any MFBO approach, LF sources should be sampled in two primary regions of their input space: (1) the region that contains their own optima since each data source is analyzed separately in the auxiliary optimization problems (see Sec. 2.4 for details), and (2) the region where the LF sources are correlated with the HF source. These two regions may overlap with each other (as is the case in NTA) or not (as is the case in HOIP or the 1D example in Fig. 1(*b*) where MFBO_{UQ} only samples LF2 once when $x<5$). We

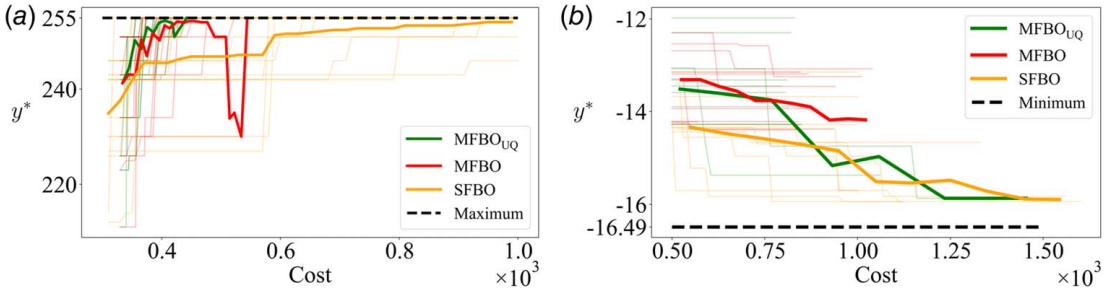

Fig. 5 Convergence histories in real-world examples: The plots depict the best HF sample found by each approach (y^*) versus their sampling costs accumulated during the BO iterations (the cost of initial data is included): (*a*) and (*b*) summarize the results for the NTA and HOIP, respectively. The thin curves show the convergence history of each repetition and the solid thick ones indicate the average behavior across the 20 repetitions. In (*b*), MFBO fails to find the optimum due to it disability in handling biased LF sources. In both examples, MFBO_{UQ} outperforms other methods. (Color version online.)

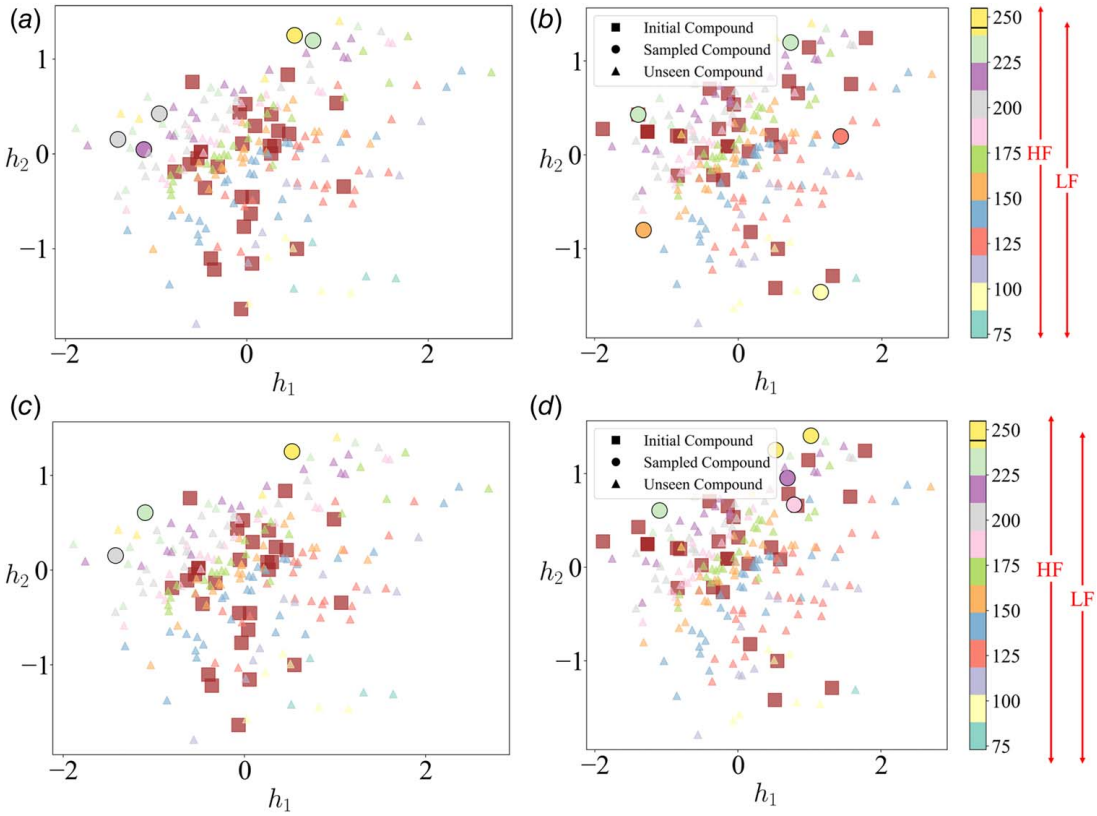

Fig. 6 BO sampling history in the encoded categorical design space of NTA: The plots in the top and bottom row illustrate the exploration–exploitation behavior of BO in MFBO and $MFBO_{UQ}$, respectively. The left and right columns correspond to the space of HF and LF sources, respectively. All latent points are color-coded based on the ground truth bulk modulus from each source and the marker shapes indicate whether the compound is part of the initial data, sampled during BO, or never seen by LMGP. The red arrows next to the legend indicate the range of response in the two data sources. This figure effectively demonstrates how strategic sampling in $MFBO_{UQ}$ leads to faster convergence compared to MFBO (see text for more detailed explanations): (*a*) HF MFBO, (*b*) LF MFBO, (*c*) HF $MFBO_{UQ}$, and (*d*) LF $MFBO_{UQ}$. (Color version online.)

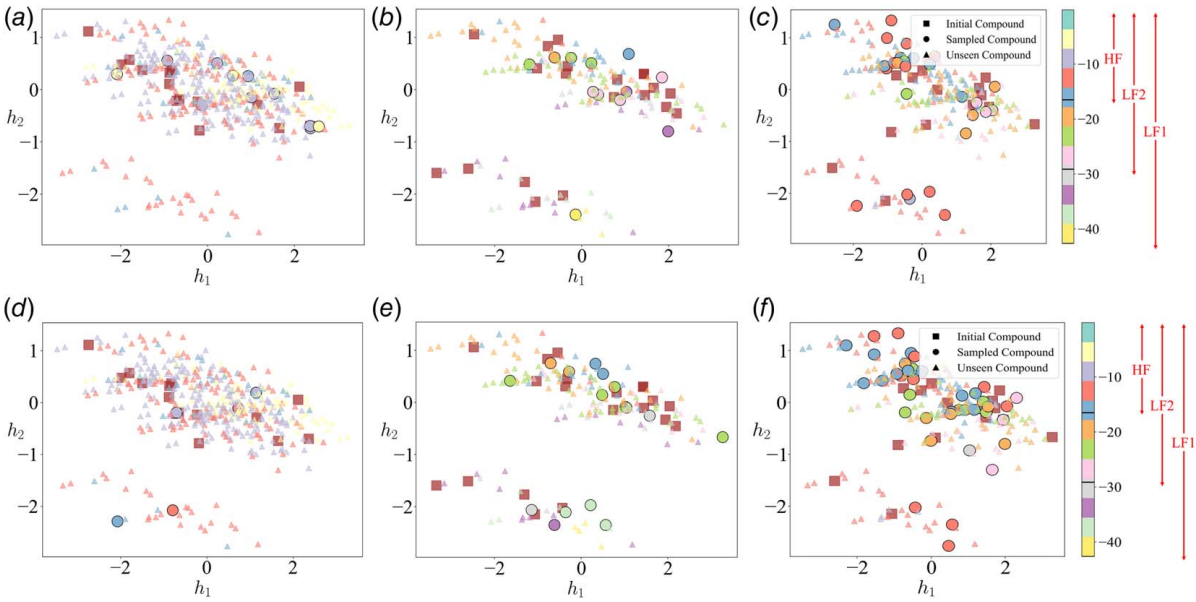

Fig. 7 BO sampling history in the encoded categorical design space of HOIP: The plots in the top and bottom row illustrate the exploration–exploitation behavior of BO in MFBO and $MFBO_{UQ}$, respectively. The left, middle, and right columns correspond to the space of HF, LF1, and LF2 sources, respectively. All latent points are color-coded based on the ground truth binding energy from each source and the marker shapes indicate whether the compound is part of the initial data, sampled during BO, or never seen by LMGP. The red arrows next to the legend indicate the range of responses in the data sources. This figure demonstrates how the strategic sampling in $MFBO_{UQ}$ enables it to find the optimum while MFBO fails: (*a*) HF MFBO, (*b*) LF1 MFBO, (*c*) LF2 MFBO,, (*d*) HF $MFBO_{UQ}$, (e) LF1 $MFBO_{UQ}$, and (*f*) LF2 $MFBO_{UQ}$. (Color version online.)

note that exploring the correlation region (if it exists!) is crucial for capturing the relationship between the LF and HF sources and as shown below the effectiveness of this exploration highly depends on the accuracy of the emulator in surrogating each source, estimating uncertainties, and identifying the correlation patterns among different data sources.

As shown in Fig. 6, for both MFBO and MFBO_{UQ} manifolds with very similar structures are learnt by LMGP for HF and LF data (this was expected per Fig. 4 which indicates that the two sources are highly correlated). For instance, for both LF and HF data, the optimum compound is located at the top-right corner of the manifold and their values are also quite close (255 for HF and 244 for LF). This similarity indicates that MFBO and MFBO_{UQ} are both able to learn about the HF source by sampling the space of the LF source. However, this sampling is more effective in the case of MFBO_{UQ} since its emulator quantifies the uncertainties more accurately. In particular, MFBO_{UQ} correctly samples compounds from the LF source that are mostly encoded in the top-right corner of the manifold (see Fig. 6(*d*)) while MFBO tests compounds that explore the entire design space (see Fig. 6(*b*)).

As shown in Fig. 7, for any of the sources and with either MFBO or MFBO_{UQ}, the compounds in the HOIP example are encoded by LMGP into two major clusters where the smaller one contains the optimum design. By examining these two clusters we observe that all the compounds in the smaller cluster have dimethylformamide (DMF) solvent. These observations are quite interesting in that they provide engineers with insights into the most important design variables that affect the materials properties (e.g., DMF solvent which decreases the binding energy in this example).

The initial HF dataset used in either MFBO or MFBO_{UQ} (see Figs. 7(*a*) and 7(*d*)) is very small and does not have any compounds from the small cluster that contains the optimum. However, there are some initial samples from LF1 and LF2 in this cluster and so we should expect BO to leverage these samples (and the fact that they have some correlation with the unseen HF compounds) in emulating the HF source and sampling compounds from it that belong to the small cluster. While this expectation is met by MFBO_{UQ}, MFBO fails to explore the (encoded) design space that contains the optimum HF sample. This failure is because (1) both LF sources (especially LF1) provide smaller binding energies than the HF source, and (2) the emulator of MFBO overestimates the uncertainties in LF sources. The combination of these two factors prevents MFBO to find an HF sample that is valuable enough to be selected in Eq. (15). We refer readers to SI B.2 available in the Supplemental Materials for more analysis on the performance of MFBO_{UQ} in these two examples.

4 Conclusion

In this paper, we develop a novel method to improve the performance of multi-fidelity cost-aware BO techniques. Our method enhances the accuracy and convergence rate of MFBO through two main contributions. First, we enable the emulator to estimate separate noise processes for each source of data. This feature increases the accuracy of the trained model since different data sources may exhibit different types and levels of noise. Second, we define a new objective function penalized by strictly proper scoring rules to (1) improve the prediction, (2) increase the focus on UQ, and (3) forgo the need to exclude highly biased data sources from BO. Our BO method, MFBO_{UQ}, accommodates any number of data sources with any levels of noise, does not require any prior knowledge about the relative accuracy of (or relation between) these sources, and can handle both continuous and categorical variables. In this paper, we illustrate these features via both analytic and engineering problems.

In this work, we use two fixed AFs in each iteration. However, one can also customize the choice of AFs for different iterations using adaptive approaches. Additionally, the examples presented in this paper are limited to single-objective problems and we do not aim to exclude the effect of noise in the final solution (i.e., the best HF sample found is noisy). We intent to study these directions in our future works.

Acknowledgment

We appreciate the support from National Science Foundation (award number *CMMI* – 2238038), the Early Career Faculty grant from NASA's Space Technology Research Grants Program (award number 80*NSSC*21*K*1809), and the UC National Laboratory Fees Research Program of the University of California (Grant No. *L*22*CR*4520).

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

References

[1] Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N., 2015, "Taking the Human Out of the Loop: A Review of Bayesian Optimization," Proc. IEEE, **104**(1), pp. 148–175.

[2] Brochu, E., Cora, V. M., and De Freitas, N., 2010, "A Tutorial on Bayesian Optimization of Expensive Cost Functions, With Application to Active User Modeling and Hierarchical Reinforcement Learning," arXiv preprint arXiv:1012.2599.

[3] Adams, R. P., 2014, "A Tutorial on Bayesian Optimization for Machine Learning," Harvard University, Cambridge, MA.

[4] Frazier, P. I., 2018, "A Tutorial on Bayesian Optimization," arXiv preprint.

[5] Nguyen, L., 2023, "Tutorial on Bayesian Optimization".

[6] Li, S., Xing, W., Kirby, R., and Zhe, S., 2020, "Multi-fidelity Bayesian Optimization Via Deep Neural Networks," Adv. Neural Inf. Process. Syst., **33**, pp. 8521–8531.

[7] Couckuyt, I., Gonzalez, S. R., and Branke, J., 2022, "Bayesian Optimization: Tutorial," Proceedings of the Genetic and Evolutionary Computation Conference Companion, Boston, MA, pp. 843–863.

[8] Frazier, P. I., and Wang, J., 2015, "Bayesian Optimization for Materials Design," *Information Science for Materials Discovery and Design*, T. Lookman, F. J. Alexander, and K. Rajan, eds., Springer, New York City, pp. 45–75.

[9] Turner, R., Eriksson, D., McCourt, M., Kiili, J., Laaksonen, E., Xu, Z., and Guyon, I., 2020, "Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020," 2020 Conference on Neural Information Processing Systems, Chicago, IL, Dec. 6–12, PMLR, pp. 3–26.

[10] Song, J., Chen, Y., and Yue, Y., 2019, "A General Framework for Multi-fidelity Bayesian Optimization With Gaussian Processes," Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, Naha, Okinawa, Japan, Apr. 16–18, PMLR, pp. 3158–3167.

[11] Takeno, S., Fukuoka, H., Tsukada, Y., Koyama, T., Shiga, M., Takeuchi, I., and Karasuyama, M., 2020, "Multi-fidelity Bayesian Optimization With Max-Value Entropy Search and Its Parallelization," Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, July 13–18, PMLR, pp. 9334–9345.

[12] Zhang, S., Lyu, W., Yang, F., Yan, C., Zhou, D., Zeng, X., and Hu, X., 2019, "An Efficient Multi-fidelity Bayesian Optimization Approach for Analog Circuit Synthesis," Proceedings of the 56th Annual Design Automation Conference 2019, New York, June 2–6, pp. 1–6.

[13] Kandasamy, K., Dasarathy, G., Schneider, J., and Póczos, B., 2017, "Multi-Fidelity Bayesian Optimisation With Continuous Approximations," Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, Aug. 6–11, PMLR, pp. 1799–1808.

[14] Zhang, Y., Hoang, T. N., Low, B. K. H., and Kankanhalli, M., 2017, "Information-Based Multi-fidelity Bayesian Optimization," Neural Information Processing Systems, Long Beach, CA, Dec. 4–9, p. 49.

[15] Shu, L., Jiang, P., and Wang, Y., 2021, "A Multi-fidelity Bayesian Optimization Approach Based on the Expected Further Improvement," Struct. Multidiscipl. Optim., **63**, pp. 1709–1719.

[16] Tran, A., Wildey, T., and McCann, S., 2020, "sMF-BO-2CoGP A Sequential Multi-fidelity Constrained Bayesian Optimization Framework for Design Applications," ASME J. Comput. Inf. Sci. Eng., **20**(3), p. 031007.

[17] Li, S., Kirby, R., and Zhe, S., 2021, "Batch Multi-fidelity Bayesian Optimization With Deep Auto-Regressive Networks," Adv. Neural Inf. Process. Syst., **34**, pp. 25463–25475.
[18] Zhang, X., Xie, F., Ji, T., Zhu, Z., and Zheng, Y., 2021, "Multi-fidelity Deep Neural Network Surrogate Model for Aerodynamic Shape Optimization," Comput. Meth. Appl. Mech. Eng., **373**, p. 113485.
[19] Li, Z., Zhang, S., Li, H., Tian, K., Cheng, Z., Chen, Y., and Wang, B., 2022, "On-Line Transfer Learning for Multi-fidelity Data Fusion With Ensemble of Deep Neural Networks," Adv. Eng. Inform., **53**, p. 101689.
[20] Liu, D., and Wang, Y., 2019, "Multi-Fidelity Physics-Constrained Neural Network and Its Application in Materials Modeling," ASME J. Mech. Des., **141**(12), p. 121403.
[21] Sarkar, S., Mondal, S., Joly, M., Lynch, M. E., Bopardikar, S. D., Acharya, R., and Perdikaris, P., 2019, "Multifidelity and Multiscale Bayesian Framework for High-Dimensional Engineering Design and Calibration," ASME J. Mech. Des., **141**(12), p. 121001.
[22] Huang, D., Allen, T. T., Notz, W. I., and Miller, R. A., 2006, "Sequential Kriging Optimization Using Multiple-Fidelity Evaluations," Struct. Multidiscipl. Optim., **32**, pp. 369–382.
[23] Forrester, A. I., Sóbester, A., and Keane, A. J., 2007, "Multi-fidelity Optimization Via Surrogate Modelling," Proc. R. Soc. A: Math. Phys. Eng. Sci., **463**(2088), pp. 3251–3269.
[24] Le Gratiet, L., and Cannamela, C., 2015, "Cokriging-Based Sequential Design Strategies Using Fast Cross-Validation Techniques for Multi-fidelity Computer Codes," Technometrics, **57**(3), pp. 418–427.
[25] Picheny, V., Ginsbourger, D., Richet, Y., and Caplin, G., 2013, "Quantile-Based Optimization of Noisy Computer Experiments With Tunable Precision," Technometrics, **55**(1), pp. 2–13.
[26] Kandasamy, K., Dasarathy, G., Oliva, J. B., Schneider, J., and Póczos, B., 2016, "Gaussian Process Bandit Optimisation With Multi-fidelity Evaluations," Adv. Neural Inf. Process. Syst., **29**.
[27] Sun, Q., Chen, T., Liu, S., Chen, J., Yu, H., and Yu, B., 2022, "Correlated Multi-objective Multi-fidelity Optimization for Hls Directives Design," ACM Trans. Des. Autom. Electron. Syst. (TODAES), **27**(4), pp. 1–27.
[28] Lam, R., Allaire, D. L., and Willcox, K. E., 2015, "Multifidelity Optimization Using Statistical Surrogate Modeling for Non-Hierarchical Information Sources," 56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Kissimmee, FL, Jan. 5–9, p. 0143.
[29] Winkler, R. L., 1981, "Combining Probability Distributions From Dependent Information Sources," Manage. Sci., **27**(4), pp. 479–488.
[30] Kennedy, M. C., and O'Hagan, A., 2001, "Bayesian Calibration of Computer Models," J. R. Stat. Soc. Ser. B (Stat. Methodol.), **63**(3), pp. 425–464.
[31] Raissi, M., and Karniadakis, G., 2016, "Deep Multi-Fidelity Gaussian Processes," arXiv preprint arXiv:1604.07484.
[32] Son, S.-H., Park, D.-H., Cha, K.-J., and Choi, D.-H., 2013, "Constrained Global Design Optimization Using a Multi-fidelity Model," 10th World Congress on Structural and Multidisciplinary Optimization.
[33] Le Gratiet, L., 2013, "Bayesian Analysis of Hierarchical Multifidelity Codes," SIAM/ASA J. Uncertain. Quantif., **1**(1), pp. 244–269.
[34] Eldred, M. S., Ng, L. W., Barone, M. F., and Domino, S. P., 2015, "Multifidelity Uncertainty Quantification Using Spectral Stochastic Discrepancy Models," Technical Report, Sandia National Laboratory (SNL-NM), Albuquerque, NM.
[35] Olleak, A., and Xi, Z., 2020, "Calibration and Validation Framework for Selective Laser Melting Process Based on Multi-fidelity Models and Limited Experiment Data," ASME J. Mech. Des., **142**(8), p. 081701.
[36] Kleiber, W., Sain, S. R., Heaton, M. J., Wiltberger, M., Reese, C. S., and Bingham, D., 2013, "Parameter Tuning for a Multi-fidelity Dynamical Model of the Magnetosphere," Ann. Appl. Stat., **7**(3), pp. 1286–1310.
[37] Xiao, M., Zhang, G., Breitkopf, P., Villon, P., and Zhang, W., 2018, "Extended Co-Kriging Interpolation Method Based on Multi-fidelity Data," Appl. Math. Comput., **323**, pp. 120–131.
[38] Perdikaris, P., Venturi, D., Royset, J. O., and Karniadakis, G. E., 2015, "Multi-fidelity Modelling Via Recursive Co-Kriging and Gaussian–Markov Random Fields," Proc. R. Soc. A: Math. Phys. Eng. Sci., **471**(2179), p. 20150018.
[39] Zhou, Q., Wu, Y., Guo, Z., Hu, J., and Jin, P., 2020, "A Generalized Hierarchical Co-Kriging Model for Multi-fidelity Data Fusion," Struct. Multidiscipl. Optim., **62**, pp. 1885–1904.
[40] Chen, C., Ran, D., Yang, Y., Hou, H., and Peng, C., 2023, "Topsis Based Multi-fidelity Co-Kriging for Multiple Response Prediction of Structures With Uncertainties Through Real-Time Hybrid Simulation," Eng. Struct., **280**, p. 115734.
[41] Ruan, X., Jiang, P., Zhou, Q., and Yang, Y., 2019, "An Improved Co-Kriging Multi-fidelity Surrogate Modeling Method for Non-Nested Sampling Data," Int. J. Mech. Eng. Rob. Res., **8**(4), pp. 1–9.
[42] Shi, R., Liu, L., Long, T., Wu, Y., and Gary Wang, G., 2020, "Multi-fidelity Modeling and Adaptive Co-Kriging-Based Optimization for All-Electric Geostationary Orbit Satellite Systems," ASME J. Mech. Des., **142**(2), p. 021404.
[43] Gardner, J., Pleiss, G., Weinberger, K. Q., Bindel, D., and Wilson, A. G., 2018, "Gpytorch: Blackbox Matrix-Matrix Gaussian Process Inference With GPU Acceleration," Adv. Neural Inf. Process. Syst., **31**.
[44] Zanjani Foumani, Z., Shishehbor, M., Yousefpour, A., and Bostanabad, R., 2023, "Multi-Fidelity Cost-Aware Bayesian Optimization," Comput. Meth. Appl. Mech. Eng., **407**, p. 115937.
[45] Escamilla-Ambrosio, P. J., and Mort, N., 2003, "Hybrid Kalman Filter-Fuzzy Logic Adaptive Multisensor Data Fusion Architectures," Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, HI, Dec. 9–12, Vol. 5, IEEE, pp. 5215–5220.
[46] Kreibich, O., Neuzil, J., and Smid, R., 2013, "Quality-Based Multiple-Sensor Fusion in an Industrial Wireless Sensor Network for MCM," IEEE. Trans. Ind. Electron., **61**(9), pp. 4903–4911.
[47] Eweis-Labolle, J. T., Oune, N., and Bostanabad, R., 2022, "Data Fusion With Latent Map Gaussian Processes," ASME J. Mech. Des., **144**(9), p. 091703.
[48] Oune, N., and Bostanabad, R., 2021, "Latent Map Gaussian Processes for Mixed Variable Metamodeling," Comput. Meth. Appl. Mech. Eng., **387**, p. 114128.
[49] Bostanabad, R., Kearney, T., Tao, S., Apley, D. W., and Chen, W., 2018, "Leveraging the Nugget Parameter for Efficient Gaussian Process Modeling," Int. J. Numer. Meth. Eng., **114**(5), pp. 501–516.
[50] Carvalho, C. M., Polson, N. G., and Scott, J. G., 2010, "The Horseshoe Estimator for Sparse Signals," Biometrika, **97**(2), pp. 465–480.
[51] Gal, Y., Van Der Wilk, M., and Rasmussen, C. E., 2014, "Distributed Variational Inference in Sparse Gaussian Process Regression and Latent Variable Models," Adv. Neural Inf. Process. Syst., **27**.
[52] Mohammed, R. O., and Cawley, G. C., 2017, "Over-Fitting in Model Selection With Gaussian Process Regression," Machine Learning and Data Mining in Pattern Recognition: 13th International Conference, MLDM 2017, New York, NY, July 15–20, Proceedings 13, Springer, pp. 192–205.
[53] Lindley, D. V., 1982, "Scoring Rules and the Inevitability of Probability," Int. Stat. Rev./Revue Int. Stat., **50**(1), pp. 1–11.
[54] Bracher, J., Ray, E. L., Gneiting, T., and Reich, N. G., 2021, "Evaluating Epidemic Forecasts in an Interval Format," PLoS Comput. Biol., **17**(2), p. e1008618.
[55] Mitchell, K., and Ferro, C., 2017, "Proper Scoring Rules for Interval Probabilistic Forecasts," Q. J. R. Metereol. Soc., **143**(704), pp. 1597–1607.
[56] Gneiting, T., and Raftery, A. E., 2007, "Strictly Proper Scoring Rules, Prediction, and Estimation," J. Am. Stat. Assoc., **102**(477), pp. 359–378.
[57] Mora, C., Eweis-Labolle, J. T., Johnson, T., Gadde, L., and Bostanabad, R., 2023, "Data-Driven Calibration of Multifidelity Multiscale Fracture Models Via Latent Map Gaussian Process," ASME J. Mech. Des., **145**(1), p. 011705 .
[58] Poloczek, M., Wang, J., and Frazier, P., 2017, "Multi-Information Source Optimization," Adv. Neural Inf. Process. Syst., **30**.
[59] Wu, J., Toscano-Palmerin, S., Frazier, P. I., and Wilson, A. G., 2020, "Practical Multi-fidelity Bayesian Optimization for Hyperparameter Tuning," Proceedings of the 35th Uncertainty in Artificial Intelligence Conference, Toronto, Canada, PMLR, pp. 788–798.
[60] Moon, H., 2010, "Design and Analysis of Computer Experiments for Screening Input Variables," Ph.D. thesis, The Ohio State University, Columbus, OH.
[61] Morris, M. D., Mitchell, T. J., and Ylvisaker, D., 1993, "Bayesian Design and Analysis of Computer Experiments: Use of Derivatives in Surface Prediction," Technometrics, **35**(3), pp. 243–255.
[62] Cover, M., Warschkow, O., Bilek, M., and McKenzie, D., 2009, "A Comprehensive Survey of M2ax Phase Elastic Properties," J. Phys.: Condens. Matter., **21**(30), p. 305403.
[63] Herbol, H. C., Hu, W., Frazier, P., Clancy, P., and Poloczek, M., 2018, "Efficient Search of Compositional Space for Hybrid Organic–Inorganic Perovskites Via Bayesian Optimization," npj Comput. Mater., **4**(1), p. 51.