Cost-aware learning of relevant contextual variables within Bayesian optimization

Julien Martinelli[*], **Ayush Bharti, S.T. John, Armi Tiihonen, Samuel Kaski**
Aalto University

Louis Filstroff
ENSAI, CREST

Sabina Sloman
University of Manchester

Abstract

Contextual Bayesian Optimization (CBO) is a powerful framework for optimizing black-box, expensive-to-evaluate functions with respect to design variables, while simultaneously efficiently integrating *relevant contextual* information regarding the environment, such as experimental conditions. However, in many practical scenarios, the relevance of contextual variables is not necessarily known beforehand. Moreover, the contextual variables can sometimes be optimized themselves, a setting that current CBO algorithms do not take into account. Optimizing contextual variables may be costly, which raises the question of determining a minimal relevant subset. In this paper, we frame this problem as a cost-aware model selection BO task and address it using a novel method, Sensitivity-Analysis-Driven Contextual BO (`SADCBO`). We learn the relevance of context variables by sensitivity analysis of the posterior surrogate model at specific input points, whilst minimizing the cost of optimization by leveraging recent developments on early stopping for BO. We empirically evaluate our proposed `SADCBO` against alternatives on synthetic experiments together with extensive ablation studies, and demonstrate a consistent improvement across examples.

1 Introduction

Bayesian optimization (BO) is a sample-efficient black-box optimization method, typically used when the expense of computing the objective function makes the problem intractable [Jones et al., 1998, Brochu et al., 2010]. Given an objective function that can be evaluated pointwise over a a set of *design variables*, BO combines surrogate modeling with a pre-specified policy of evaluation over the design space (the so-called acquisition function) in order to locate the global optimum of the function efficiently. BO has been especially useful as a mechanism for automatic discovery of materials [Zhang et al., 2020] and pharmaceutical compounds [Gómez-Bombarelli et al., 2018, Korovina et al., 2020], problem domains in which evaluating the performance of a candidate design requires performing a costly experiment. Despite the success of BO and the recent algorithmic advancements, open challenges still remain for its practical use.

A key implicit assumption in BO is that the objective function only depends on the design variables. This assumption is violated in many practical scenarios, wherein various environmental factors and experimental settings, referred to as *contextual variables* [Krause and Ong, 2011, Kirschner et al., 2020, Arsenyan et al., 2023], also affect the objective function. For instance, ambient humidity was found to influence the experiments in robot-assisted material design [Nega et al., 2021], leading to a changing optimal design under different humidity conditions. Moreover, in practice, the domain

[*]contact: `julien.martinelli@aalto.fi`

Preprint. Under review.

experts themselves might not know *a priori* which contextual variables are relevant, and would observe their confounding effect only during the course of the optimization process. Identifying the relevant contextual variables is therefore critical not only to guarantee reliable optimization results, but also for the practitioners to reliably reproduce experimental results.

Variants of BO have therefore been developed to deal with the uncertainty related to the contextual variables. In particular, Krause and Ong [2011] introduced the Contextual Bayesian optimization (CBO) framework, which enables the inclusion of uncontrollable contextual information (such as environment conditions during the experiment) in the surrogate model. Alternatively, several works have proposed to alter the simple optimization objective to make it robust in some sense, such as taking the expectation w.r.t. the contextual variables [Toscano-Palmerin and Frazier, 2018], or considering distributionally-robust scenarios [Bogunovic et al., 2018, Kirschner et al., 2020]. However, in some applications, contextual variables *can* be controlled. For instance, synthesis conditions of material samples, such as sintering temperature or the used solvents, or certain environment conditions, such as experiment room temperature or ambient humidity [Higgins et al., 2021, Nega et al., 2021], are principally controllable during the course of an experiment, but it may not be straigthforward to predict whether they are relevant to include [Abolhasani and Brown, 2023]. While gains in BO performance can potentially be obtained by optimizing over all the potential contextual variables, or by determining the relevant ones and optimizing over them, intervening on such variables is most of the time costly, thus invoking a cost versus efficiency trade-off. To the best of our knowledge, this trade-off has not been considered so far in the literature.

Contributions. In this paper, we extend the CBO framework to settings in which the relevant contextual variables are (i) not known beforehand, and (ii) can be intervened on at some cost. Our first contribution is to cast the identification of relevant contextual variables as a cost-aware model selection problem. We then propose a Sensitivity-Analysis-Driven CBO (`SADCBO`) algorithm for the simultaneous optimization and identification of relevant contextual variables in a cost-effective manner. The proposed `SADCBO` leverages recent advances in sensitivity-analysis-driven variable selection [Sebenius et al., 2022] and early stopping criteria for BO [Ishibashi et al., 2023]. We emphasize that `SADCBO` combines the *contextual observational* setting, where the context information is only observed, and the *contextual interventional* setting, where contextual variables are intervened on (similar to design variables), into a sequential algorithm. We provide a thorough evaluation of the performance of `SADCBO`, comparing it against methods from the CBO and high-dimensional BO literatures, on synthetic examples. Our results demonstrate that `SADCBO` consistently achieves the optimum value at a lower cost compared to existing methods.

2 Contextual Bayesian optimization (CBO)

The CBO framework [Krause and Ong, 2011] deals with a black-box function $f : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ defined on the joint space of both the *design variables* $\mathcal{X} \subset \mathbb{R}^d$ and *contextual variables* $\mathcal{Z} \subset \mathbb{R}^c$. We assume that we get noisy evaluations of f, that is we get $y = f(\mathbf{x}, \mathbf{z}) + \varepsilon$ with $\varepsilon \sim \mathcal{N}(0, \sigma^2_{\text{noise}})$.

A Gaussian process (GP) prior [Rasmussen and Williams, 2006] is placed on f; with the notation $\mathbf{v} = [\mathbf{x}, \mathbf{z}]$, we write $f(\mathbf{v}) \sim \mathcal{GP}(0, k(\mathbf{v}, \mathbf{v}'))$. A GP is a stochastic process fully characterized by its mean function (taken here to be zero without loss of generality), and its kernel function $k(\mathbf{v}, \mathbf{v}') = \text{cov}[f(\mathbf{v}), f(\mathbf{v}')]$. This means that, for any finite-dimensional collection of inputs $[\mathbf{v}_1, \dots, \mathbf{v}_t]$, the function values $\mathbf{f} = [f(\mathbf{v}_1), \dots, f(\mathbf{v}_t)]^\top \in \mathbb{R}^t$ follow a multivariate normal distribution $\mathbf{f} \sim \mathcal{N}(\mathbf{0}, \mathbf{K})$, where $\mathbf{K} \in \mathbb{R}^{t \times t} = (k(\mathbf{v}_i, \mathbf{v}_j))_{1 \leq i,j \leq t}$ is called the kernel matrix. Lastly, given a dataset $\mathcal{D}_t = \{(\mathbf{x}_i, \mathbf{z}_i, y_i)\}_{i=1}^t = \{(\mathbf{v}_i, y_i)\}_{i=1}^t$, the posterior distribution of $f(\mathbf{v})$ given $\mathcal{D}_t$ is Gaussian for all $\mathbf{v}$ with closed-form expressions for the mean $\mu_t(\mathbf{v}|\mathcal{D}_t)$ and variance $\sigma_t^2(\mathbf{v}|\mathcal{D}_t)$:

$$\mu_t(\mathbf{v}|\mathcal{D}_t) = \boldsymbol{k}_{\mathbf{v}}(\mathbf{K} + \sigma^2_{\text{noise}}\mathbf{I})^{-1}\mathbf{y}, \qquad \sigma_t^2(\mathbf{v}|\mathcal{D}_t) = k(\mathbf{v}, \mathbf{v}) - \boldsymbol{k}_{\mathbf{v}}(\mathbf{K} + \sigma^2_{\text{noise}}\mathbf{I})^{-1}\boldsymbol{k}_{\mathbf{v}},$$

where $\mathbf{y} = [y_1, \dots, y_t]^\top \in \mathbb{R}^t$ and $\boldsymbol{k}_{\mathbf{v}} = [k(\mathbf{v}, \mathbf{v}_1), \dots, k(\mathbf{v}, \mathbf{v}_t)]^\top \in \mathbb{R}^t$.

In the CBO setting, we sequentially observe the context variables, and choose the design variables in response to this observation. More precisely, at iteration $t + 1$, a context vector $\mathbf{z}_{t+1}$ is observed (assumed to have been drawn from a distribution $p(\mathbf{z})$), and the optimal design $\mathbf{x}^\star_{t+1}$ is such that

$$\mathbf{x}^\star_{t+1} = \arg\max_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}, \mathbf{z}_{t+1}). \tag{1}$$

Given $\mathbf{z}_{t+1}$ and the previous t observations $\mathcal{D}_t$, the next candidate design point $\mathbf{x}_{t+1}$ is selected using the Upper Confidence Bound (UCB) acquisition function [Srinivas et al., 2012] :

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}\in\mathcal{X}} \mu_t(\mathbf{x}, \mathbf{z}_{t+1}|\mathcal{D}_t) + \beta_t^{1/2}\sigma_t(\mathbf{x}, \mathbf{z}_{t+1}|\mathcal{D}_t), \tag{2}$$

where $(\beta_t)_{t\geq 1}$ is a sequence balancing exploitation (high values of μ_t) and exploitation (high values of σ_t). This incurs a design query cost $\lambda_{\mathbf{x}}$ and leads to a regret $R_{t+1} = f(\mathbf{x}^\star_{t+1}, \mathbf{z}_{t+1}) - f(\mathbf{x}_{t+1}, \mathbf{z}_{t+1})$ at iteration $t+1$, with the goal being to minimize the cumulative regret.

Extending the CBO problem setup. We extend the problem setting of CBO in two ways. Firstly, we assume that only a subset of the contextual variables truly affect the function f. Let $\mathbf{z} = [z^{(1)}, \ldots, z^{(c)}]$ be the vector of all contextual variables. For any set J belonging to the power set of $\{1, \ldots, c\}$, we denote by $\mathbf{z}^{(J)} \in \mathbb{R}^{|J|}$ the vector of reduced dimension whose variables are indexed by J. For instance, if $J = \{1, 3\}$, then $\mathbf{z}^{(J)} = [z^{(1)}, z^{(3)}]$. We assume that there exists a set $J^\star$, where $|J^\star| \ll c$, such that $f(\mathbf{x}, \mathbf{z}) = f(\mathbf{x}, \mathbf{z}^{(J^\star)})\ \forall(\mathbf{x}, \mathbf{z})$. Secondly, we include the possibility to set the value of any of the contextual variables at some cost, besides the usual design query cost $\lambda_{\mathbf{x}}$. This means that for all $j \in \{1, \ldots, c\}$, the context variable $z^{(j)}$ can be intervened on for a cost λ_j. With these two additional assumptions, our aim is to maximize the function f in a cost-efficient manner, while identifying the optimal set $J^\star$.

3 Methodology

This section introduces our method to solve the aforementioned extended CBO problem. The method relies on a variable selection technique from the GP literature [Sebenius et al., 2022] based on sensitivity analysis to handle the presence of irrelevant contextual variables. Section 3.1 describes the adaption of this method to the optimization setting by restricting the dataset to high function values. Section 3.2 then presents our algorithm, coined `SADCBO`, which employs that variable selection method in a sequential algorithm to solve the optimization problem.

3.1 Variable selection for CBO via sensitivity analysis

One approach for handling the presence of contextual variables that can be intervened on is to include them in the design space. However, such a strategy can become infeasible when their relevance is not known *a priori* and the domain experts can only provide a candidate set of potentially relevant contextual variables, leading to an exponential expansion of the search space. In such cases, identifying the relevance of the contextual variables is key, not only for efficient optimization of the function, but also as additional information to the experts about the experiment.

To that end, we adapt the Feature Collapsing (FC) method [Sebenius et al., 2022] from the sensitivity analysis literature to identify the relevant contextual variables. The FC method applies a perturbation to a training point (namely, setting one feature to zero), and measures the induced shift in the posterior predictive distribution in terms of KL divergence. Given a dataset $\mathcal{D}_t = \{(\mathbf{x}_i, \mathbf{z}_i, y_i)\}_{i=1}^t$, the relevance $r(i, j)$ of the i^{th} sample of the j^{th} contextual variable $z_i^{(j)}$ is computed as

$$r(i, j) = \mathrm{KL}\left(p(y_\star|\mathbf{x}_i, \mathbf{z}_i, \mathcal{D}_t)\ ||\ p(y_\star|\mathbf{x}_i, \mathbf{z}_i \odot \boldsymbol{\xi}[j], \mathcal{D}_t)\right), \tag{3}$$

where $\boldsymbol{\xi}[j] = [\xi^{(1)}, \ldots, \xi^{(c)}]$ is a vector s.t. $\xi^{(j)} = 0$, and $\xi^{(j')} = 1$, for $j' \neq j$, and $\odot$ is the element-wise multiplication. The relevance score of the j^{th} contextual variable is then computed as

$$\mathrm{FC}(j) = \frac{1}{|\mathcal{D}_t|}\sum_{i=1}^{|\mathcal{D}_t|}\left(\frac{r(i, j)}{\sum_{j=1}^{c} r(i, j)}\right). \tag{4}$$

Here, we normalize the relevance r before averaging over the dataset in order to determine a universal variable selection threshold.

The FC scores computed in this manner reveal the variables that are relevant for output prediction across the dataset $\mathcal{D}_t$. However, as our goal is to maximize f, we are interested in identifying contextual variables that are relevant for high function values. Hence, we modify the dataset over which the FC scores are averaged in Equation (4) to adapt them to the BO setting. We denote that

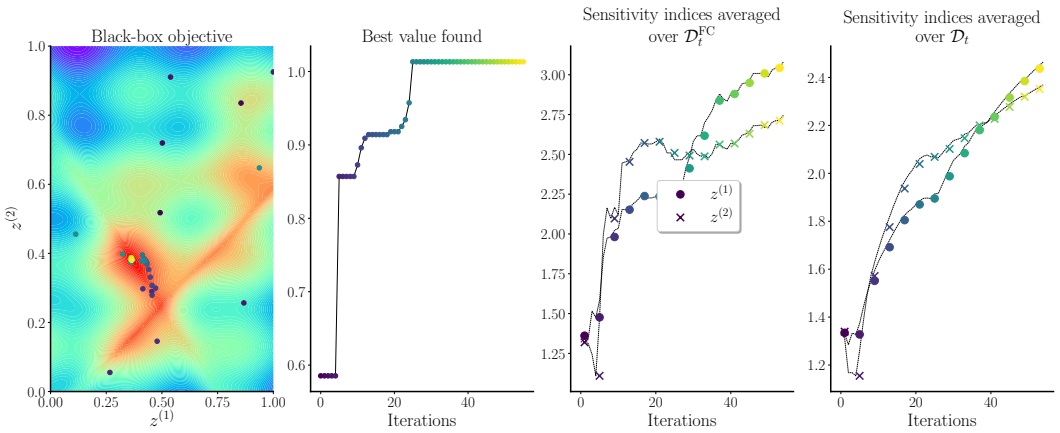

Figure 1: **Performing sensitivity analysis on $\mathcal{D}_t^{\text{FC}}$ characterizes variable importance at the optimum faster than $\mathcal{D}_t$.** *Left*: 2D black-box objective together with the queries produced along a BO trajectory. The initial dataset is represented by dark-colored dots. Newly obtained samples display an increasingly lighter color. *Middle left*: Best value found during the optimization trial. *Middle right*: Sensitivity indices for $z^{(1)}$ and $z^{(2)}$ averaged over $\mathcal{D}_t^{\text{FC}}$. As we converge to the optimum, $\mathcal{D}_t^{\text{FC}}$ mainly involves samples close to the optimum, leading to a different variable relevance ranking (iteration 30 to the end) compared to that of the early iterations (10 to 30). *Right*: Sensitivity indices computed on the whole dataset $\mathcal{D}_t$, showing a different trend than the previous one due to the presence of samples with low values.

dataset by $\mathcal{D}_t^{\text{FC}}$. We define it to be the union of two datasets : $\mathcal{D}_t^{\text{FC}} = \mathcal{D}_t^{\gamma_t} \cup \mathcal{D}_t^{Q}$. Here, $\mathcal{D}_t^{\gamma_t}$ is a subset of $\mathcal{D}_t$, comprised of only high output values, defined as

$$\mathcal{D}_t^{\gamma_t} = \{(\mathbf{x}_i, \mathbf{z}_i, y_i) \in \mathcal{D}_t \mid y_i / y_{\text{best}} \geq \gamma_t\}, \tag{5}$$

where $y_{\text{best}} = \max_{1 \leq i \leq t} y_i$ is the current observed maximum. For instance, using $\gamma_t = 0.8 \; \forall t$ would yield a $\mathcal{D}_t^{\gamma_t}$ that consists of the highest 20% observations obtained so far. As for $\mathcal{D}_t^{Q} := \{(\mathbf{x}_q^\star, \mathbf{z}_{t+1})\}_{q=1}^{Q}$, it is composed of a batch of Q promising design points $\{\mathbf{x}_q^\star\}_{q=1}^{Q}$ and the current context $\mathbf{z}_{t+1}$ with respect to an acquisition function α as

$$\{\mathbf{x}_q^\star\}_{q=1}^{Q} = \underset{\{\mathbf{x}_q\}_{q=1}^{Q} \in \mathcal{X}^Q}{\arg\max} \; \alpha(\{(\mathbf{x}_q, \mathbf{z}_{t+1})\}_{q=1}^{Q} | p(f|\mathcal{D}_t)). \tag{6}$$

Figure 1 illustrates the pertinence of working with $\mathcal{D}_t^{\text{FC}}$ instead of $\mathcal{D}_t$ on a toy example.

Once the FC scores are computed and sorted in descending order, we select the indices of those contextual variables whose cumulative FC scores is greater than $\eta \in [0, 1]$, meaning that the selected variables explain the fraction η of the output sensitivity amongst all contextual variables. Let J_η denote the set of indices of the selected contextual variables. We train a GP surrogate based on $\{(\mathbf{x}_i, \mathbf{z}_i^{(J_\eta)}, \mathbf{y}_i\}_{i=1}^{t}$ and select the designs through maximization of the UCB acquisition function:

$$\mathbf{x}_{t+1} = \underset{\mathbf{x} \in \mathcal{X}}{\arg\max} \; \mu_t(\mathbf{x}, \mathbf{z}_{t+1}^{(J_\eta)} | \mathcal{D}_t) + \beta_t^{1/2} \sigma_t(\mathbf{x}, \mathbf{z}_{t+1}^{(J_\eta)} | \mathcal{D}_t). \tag{7}$$

Note that any measure of variable relevance could have been applied here, such as the method proposed in Spagnol et al. [2019] based on the maximum mean discrepancy [Gretton et al., 2012]. However, we found the FC method to perform better, see Supplementary Section A for a comparison.

3.2 Sensitivity-Analysis Driven CBO (SADCBO)

We now present SADCBO, a sequential method for performing BO in the presence of irrelevant contextual variables, summarized in Algorithm 1. SADCBO utilizes the variable selection method of Section 3.1 and proceeds in two phases. In the first phase, termed *observational phase*, we choose to

only observe the values of the contextual variables without optimizing over them. This is to ensure that we do not spend budget optimizing the contextual variables when their relevance is computed based on a limited amount of data, and hence, can be noisy. Instead, we perform the vanilla CBO method described in Section 2 after selecting the contextual variables based on their FC relevance. Thus, in this phase, we leverage the available contextual information to guide design selection. This information, however, will saturate at some point, leading to diminishing simple regret differences. This is when we begin the second phase.

In the second phase, we begin to intervene on the contextual variables selected at each iteration based on their FC relevance. Hence, we call this the *interventional phase*. As there is a cost λ_j associated with intervening on the context variable $z^{(j)}$,we modify the FC relevance in Equation (4) to be $\text{FC}(j)/\lambda_j$, $j = 1, \dots, c$. Our variable selection criterion can then be interpreted as the degree of sensitivity *per unit cost*. This allows SADCBO to automatically trade off a variable's potential to substantially affect the optimum with the associated cost of intervention. As previously, once the contextual variables $\mathbf{z}^{(J_\eta)}$ have been selected, we train a GP surrogate based on $\{(\mathbf{x}_i, \mathbf{z}_i^{(J_\eta)}, \mathbf{y}_i\}_{i=1}^t$ and select the next design and contextual variables to query as

$$(\mathbf{x}_{t+1}, \mathbf{z}_{t+1}^{(J_\eta)}) = \underset{(\mathbf{x}, \mathbf{z}^{(J_\eta)}) \in \mathcal{X} \times \prod_{j \in J_\eta} \mathcal{Z}_j}{\arg\max} \mu_t(\mathbf{x}, \mathbf{z}^{(J_\eta)} | \mathcal{D}_t) + \beta_t^{1/2} \sigma_t(\mathbf{x}, \mathbf{z}^{(J_\eta)} | \mathcal{D}_t). \tag{8}$$

Note that our acquisition function is not cost-weighted, as is traditionally done when incorporating cost in the optimization. However, cost-weighted acquisition functions can dramatically underperform compared to their vanilla counterparts [Lee et al., 2021], specifically for non-continuous cost models. Including the cost at the model selection level avoids this issue.

Going from observational to interventional phase. To switch from the observational to the interventional phase in SADCBO, we use the criterion proposed by Ishibashi et al. [2023] for determining the stopping time in BO. Using this criterion, we detect the point at which the gain in the optimization from purely observing the contextual variables diminishes, following which the interventional phase begins. We now briefly describe the details of this switching criterion.

Let $\mathbf{v}_t^\star = \arg\max_{\mathbf{v} \in \mathcal{D}_t} f(\mathbf{v})$ be the current best candidate point in the dataset up to time t, where $\mathbf{v} = [\mathbf{x}, \mathbf{z}]$. and denote $f^* := \max_{\mathbf{v} \in \mathcal{V}} f(\mathbf{v})$. $R_t = f^\star - \mathbb{E}_{\hat{f} \sim p(f|\mathcal{D}_t)}[\max_{\mathbf{v} \in \mathcal{V}} \hat{f}(\mathbf{v})]$ be the expected minimum simple regret. Then, $\Delta R_t := |R_t - R_{t-1}|$ can be upper bounded by

$$\Delta R_t \leq v(\phi(g) + g\Phi(g)) + |\Delta \mu_t^\star| + \kappa_{\delta, t-1} \sqrt{\frac{1}{2} \text{KL}(p(f|\mathcal{D}_t) || p(f|\mathcal{D}_{t-1}))} := \Delta \tilde{R}_t,$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the p.d.f. and c.d.f. of a standard Gaussian distribution, respectively, $\Delta \mu_t^\star := \mu_{t-1}(\mathbf{v}_{t-1}^\star) - \mu_t(\mathbf{v}_t^\star)$, $v := \sqrt{\sigma_t^2(\mathbf{v}_t^\star) - 2\sigma_t^2(\mathbf{v}_t^\star, \mathbf{v}_{t-1}^\star)) + \sigma_t^2(\mathbf{v}_{t-1}^\star)}$, $g := -\Delta \mu_t^\star / v$, and $\kappa_{\delta, t-1}$ is a sequence indexed by t and depending on δ. Then, we switch from the observational to the interventional phase in SADCBO when $\Delta R_t \leq s_t$, where

$$s_t := \frac{(\sigma_{t-1}^2(\mathbf{v}_t^\star) + \kappa_{\delta, t-1}/2)\sigma_{t-1}^2(\mathbf{v}_t) \sqrt{-2 \log \delta}}{\sqrt{\sigma_{\text{noise}}}(\sigma_{t-1}^2(\mathbf{v}_t) + \sigma_{\text{noise}}^{-1})}.$$

4 Related work

Robust BO. Bogunovic et al. [2018] and Kirschner et al. [2020] perform worst-case optimization under fluctuations of the contextual variables. In particular, Distributionally-Robust BO (DRBO) tries to maximize the expected black-box function value under the worst-case distribution of the contextual variables. This worst-case distribution belongs to an uncertainty set of distributions, typically a ball centered around a reference distribution that is gradually learnt. However, as in Krause and Ong [2011], these works assume that the relevant contextual variables are known *a priori*, and can only be observed and not controlled. Moreover, DRBO relies on discretization of the input space, and can therefore be computationally challenging even in moderately high dimensions.

High-dimensional BO. Due to the curse of dimensionality, the performance of standard BO is severely degraded when applied in high-dimensional input spaces. To tackle this problem, most proposed approaches either aim at carrying out BO in a lower-dimensional space instead of the

Algorithm 1 Sensitivity-Analysis-Driven Contextual BO (`SADCBO`)

Input: initial dataset $\mathcal{D}_0$, hyperparameters η and γ, batch size Q, budget Λ, costs $\lambda_{\mathbf{x}}, \lambda_1, \ldots, \lambda_c$
Initialize GP using all variables $[\mathbf{x}, \mathbf{z}]$. `phase` = observational
while $\Lambda \geq \lambda_{\mathbf{x}} + \min_j \lambda_j$ **do**
  Receive context $\mathbf{z}_t \sim p(\mathbf{z})$
  Assemble dataset $\mathcal{D}_t^{\text{FC}}$ (Equations (5) and (6))
  Compute sensitivity measure $\text{FC}(j)$ based on $\mathcal{D}_t^{\text{FC}}$ Equation (3)
  In descending order, add indices to J_η until $\sum_{j \in J_\eta} \text{FC}(j) > \eta$
  Train reduced GP on $[\mathbf{x}, \mathbf{z}^{(J_\eta)}]$
  Get $\mathbf{x}_t$ Equation (7) (and $\mathbf{z}_t$ (Equation (8)) if `phase` = interventional)
  $y_t \leftarrow f(\mathbf{x}_t, \mathbf{z}_t) + \varepsilon_t$
  $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(\mathbf{x}_t, \mathbf{z}_t, y_t)\}$
  Retrain full GP
  if `phase` = observational and $\Delta \tilde{R}_t \leq s_t$ [based on $p(f|\mathcal{D}_t)$] (Section 3.2) **then**
    `phase` = interventional `// Never check again once criterion satisfied`
  end if
  $\Lambda \leftarrow \Lambda - \lambda_{\mathbf{x}} + \sum_{j \in J_\eta} \lambda_j,\ t \leftarrow t + 1$
end while

original, or work with a structured GP surrogate. The former can be achieved for instance in a data-agnostic manner, by randomly dropping dimensions of the problem [Li et al., 2018] or considering tree-like random decompositions [Ziomek and Bou-Ammar, 2023]. Data-driven methods based on various measures of feature relevance have also been proposed [Spagnol et al., 2019, Shen and Kingsford, 2021]. These methods have the drawback of having to assign a value to the dropped variables for evaluation, which the proposed method alleviates, as the non-selected contextual variables are sampled from their reference distribution. As for the structured surrogate methods, they encode structural information about the objective, for instance using an additive kernel, together with an acquisition function which is additive under the provided decomposition [Rolland et al., 2018]. Finally, a recent line of work proposed using a sparsity-enforcing GP surrogate, equipped with a horseshoe prior on the squared inverse lengthscales [Eriksson and Jankowiak, 2021, Liu et al., 2023].

Cost-aware BO. In most methods, the BO budget is given in iterations, implicitly assuming that each evaluation has the same cost, whereas they may vary significantly across different space regions [Lee et al., 2020], or depend on the number of variables we intervene over. Cost-aware BO integrates the cost-constrained nature of the problem, usually through the acquisition function, but also with more involved strategies like constrained Markov decision processes [Lee et al., 2021]. To the best of our knowledge, the cost of intervening on contextual variables has not been studied before.

5 Experimental results

In this section, we evaluate our method on several synthetic examples in Section 5.1. We compare the performance of our method with a number of baselines listed in Table 1. Finally, ablation studies with respect to the hyperparameters of our method is presented in Section 5.2. Source code reproducing the experiments is available at `Link removed for anonymity`.

Baselines. Benchmarked baselines are referenced in Table 1. Our approach, coined `SADCBO`, is expected to compete with or outperform Oracle Vanilla BO (`OVBO`) upon identification of the relevant contextual variables. We also report `SADBO`, an analogue to our method but without the first phase, which amounts to performing BO with variable selection at each step, i.e., the interventional phase.

Implementation details. We fix the hyperparameter of `SADCBO` and `SADBO` to $\eta = 0.8, Q = 10, \gamma_t = 0.8\ \forall t$. A min-max transformation is performed on the input data, scaling it to the unit cube: $\mathcal{X} \times \mathcal{Z} = [0, 1]^{d+c}$. Similarly, the output is scaled between $[0, 1]$ and a noise term $\varepsilon \sim \mathcal{N}(0, \sigma^2_{\text{noise}})$ is added with $\sigma^2_{\text{noise}} = 0.001$. The contextual variable distribution is $p(\mathbf{z}) = \mathcal{U}([0, 1]^c)$. The query cost of any contextual variable is set to 3, and that of any design variable to 1. We use an RBF kernel together with the UCB acquisition strategy, as well as Q-UCB for computing $\mathcal{D}_t^Q$ (Eq. (6)) [Wilson et al., 2017]. Our algorithm is implemented using the `BoTorch` framework [Balandat et al., 2020]. Experiments were run on a 8 cores with 3.2GHz 13" 2020 M1 Macbook Pro.

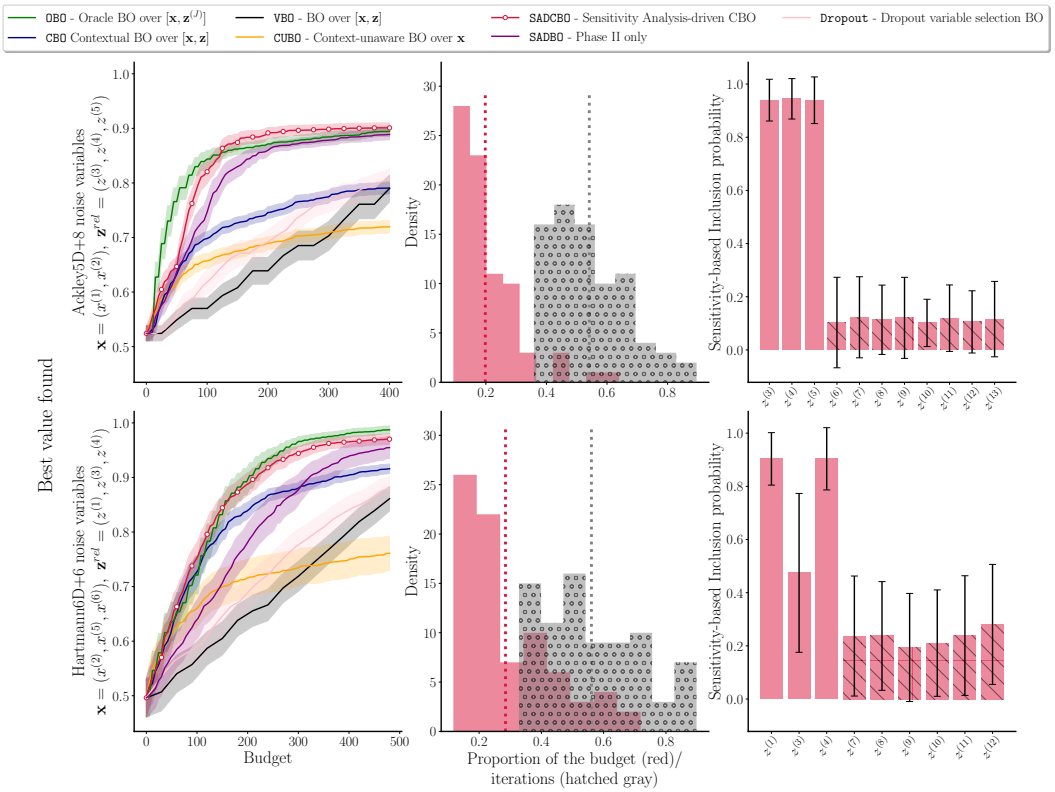

Figure 2: Synthetic functions benchmark. Each line shows the mean across $N = 80$ simulated trials. Left: Best value found. SADCBO (red curve with white markers) **and** SADBO (purple line) **outperform other baselines and compare favorably with the oracle method** OBO. SADCBO improves over SADBO due to its sequential nature. Middle: Histograms of early stopping criterion hitting time for SADCBO, in proportion of the budget (iterations) colored in red (hatched, gray). Vertical lines refer to the mean of each distribution. Observational and interventional phases roughly last the same number of iterations. Right: Inclusion probability of each contextual variable for SADCBO. Irrelevant contextual variables are hatched. SADCBO associates high importance to relevant variables as expected.

5.1 Synthetic experiments

Two synthetic test functions are considered:

- The **Ackley-5D function** has two design variables $(x^{(1)}, x^{(2)})$ and 3 relevant contextual variables $(z^{(3)}, z^{(4)}, z^{(5)})$. This function serves as a sanity check: every relevant variable involved contributes equally to the output function. This can be seen from the definition (Supplementary Section E). 8 irrelevant contextual variables are added to the input space.
- The **Hartmann-6D function** has three design variables $(x^{(2)}, x^{(5)}, x^{(6)})$ and three contextual variables $(z^{(1)}, z^{(3)}, z^{(4)})$. Here, the baselines that integrate contextual information are expected to perform better, according to a global Sobol sensitivity analysis [Sobol, 2001] (Table 2). 6 irrelevant contextual variables are added to the input space.

Figure 2 (left panel) displays the best value found by each baseline. SADCBO (in red) closely follows the Oracle vanilla BO OVBO (in green), specifically when considering Hartmann-6D (lower panel). SADBO yields slighty worse performance, as it does not leverage contextual information available for free in the first observational phase, and spends all its budget in the interventional phase. This difference is particularly striking for the Hartmann case. This behavior is expected, as in this example, the contextual variables' importance outweigh that of the design variables. As a result, observational contextual data is both highly informative and free. Finally, VBO does a poor job at optimizing the function as it considers all variables, suffering from both the curse of dimensionality and high costs.

Table 1: Baselines used in the experiments. We expect SADCBO to outperform both baselines that do and do not incorporate variable selection, and compare favorably to an oracle method that has access to the true set of relevant contextual variables.

	Name	Description	Reference
No variable selection	CUBO	Context-Unaware BO over designs $\mathbf{x}$	-
	CBO	Contextual BO using all contexts $\mathbf{z}$	[Krause and Ong, 2011]
	VBO	Vanilla BO over $[\mathbf{x}, \mathbf{z}]$	-
Variable selection	DBO	Randomly keep only 5 context variables	[Li et al., 2018]
	MMDBO/MMDCBO	Maximum mean discrepancy-driven BO (Supplemenary Section A)	[Spagnol et al., 2019]
Our method	SADBO/SADCBO	**Sensitivity analysis-driven BO/CBO**	-
Oracle	OVBO	Oracle Vanilla BO optimising only $[\mathbf{x}, \mathbf{z}^{(J)}]$	-

Next, the middle panel reports the moment at which the stopping criterion kicks in for SADCBO, in proportion of the total budget (red) or iterations (gray), demonstrating that both phases are successfully leveraged in our approach, as they roughly last the same number of iterations. We further ensure that the criterion is well-behaved in the sense that the more information about the output the contextual variables contain, the later the early stopping criterion triggers (Figure S2).

Lastly, the right panel reports the sensitivity indices computed at each iteration for each contextual variable, averaged across the trajectories of multiple BO trials. For Ackley-5D (top row), we recover equal importance for the relevant context variable, as expected. For Hartmann-6D, the low relevance attributed to $z^{(3)}$ matches the global sensitivity analysis results (Table 2), even though global sensitivity indices might differ from sensitivity indices with respect to the function optimum.

5.2 Ablation study

We here report the results of ablation studies that show that SADCBO 's performance is robust to variation in query cost and the contextual variable distribution. Additional results described in Supplementary Section C further demonstrate its robustness to the number of irrelevant contextual variables and to the hyperparameter settings. Results are summarized in Figure 3.

Query cost. Many different scenarios will show up in a real-world setting. We now investigate four different query cost models, described as column headings in Figure 3, and show that SADCBO performs well for reasonably high enough contextual variable cost. For potentially expensive contextual variables (first and second columns), SADCBO significantly improves over SADBO, even when relevant context variables are costly compared to their irrelevant counterpart. This highlights the importance of the contextual observational phase in careful determination of which contextual variables justify the expense. Unsurprisingly however, when contextual and design variables share the same cost (Figure 3, first column), SADBO (purple) slightly outperforms SADCBO (red). This is confirmed in the second column when $\lambda_j \sim \mathcal{U}(0.5, 2)$. In other words, there exist low enough cost values below which one should skip the observational phase and start by directly optimizing contextual variables.

Contextual variables distribution. The distribution of the contextual variables affects the probability of sampling contextual regions associated with high function values, and thus the importance of the contextual observational phase. Here, we consider three different contextual variable models, described as column headings of the bottom two rows of Figure 3. Again, SADCBO performs well in almost each setting.

The impact of the contextual variables distribution is best shown on the Ackley function. When scaled in $[0, 1]$, the maximizer with respect to the 3 relevant contextual variables $(z^{(1)}, z^{(3)}, z^{(4)})$ lies at $(0.5, 0.5, 0.5)$. Under a Beta(2,2) distribution, the maximizer neighbourhood is more susceptible to be sampled than any other region of the space. Under a Beta(0.5,0.5) law, sampling this region is roughly less than two times likely. As a result, the performance of SADCBO is as good as one could hope for under Beta(2,2), and virtually revert back to SADBO under Beta(0.5,0.5), as getting a context vector in the neighbourhood of the optimum is unlikely. This also translates in terms of stopping time: as the maximum is more likely to be sampled, the observational phase quickly converges, leading to an earlier stopping time (Figure S3).

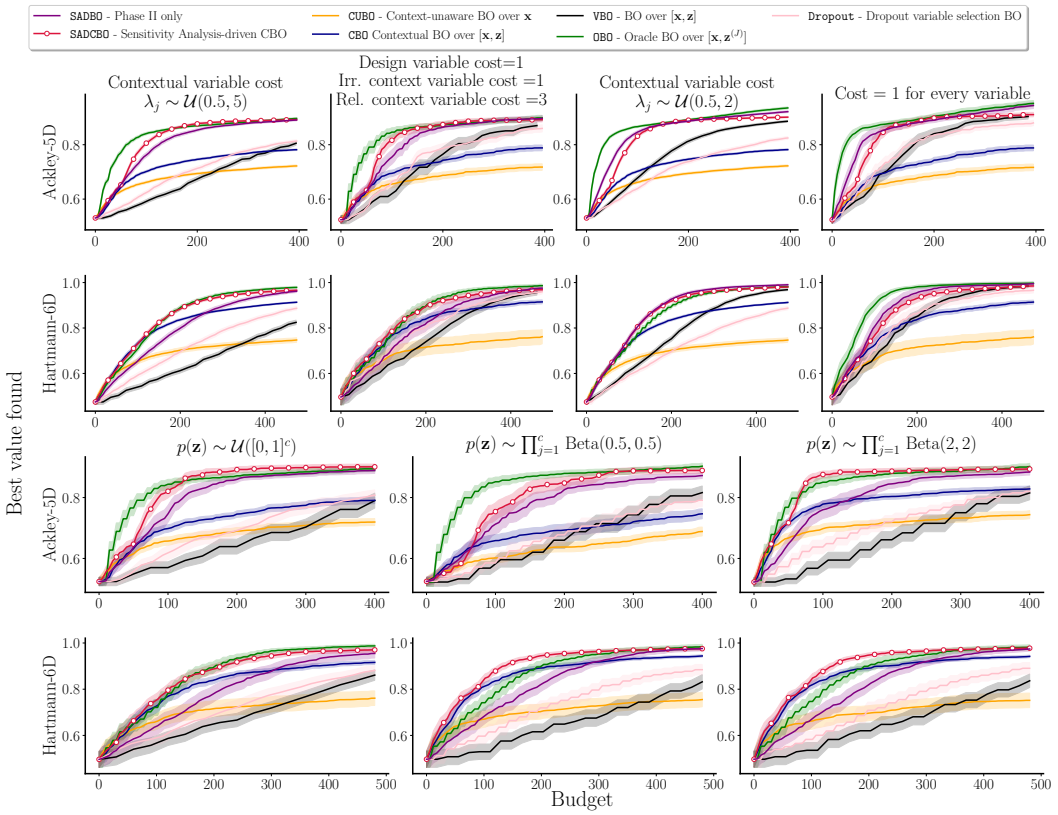

Figure 3: Ablation study on query cost and contextual variable distribution. The comparative performance of our method is robust to each ablation: in each panel, SADCBO (red line with white markers) and SADBO (purple line) outperform other baselines and compare favorably with the oracle method (green line). **Rows 1–2:** Variation in query cost (see column labels). For random costs, we sample 10 different cost models, leading to 10 cost models $\times$ 80 experiments per model = 800 experiments in total. **Rows 3–4:** Variation in distribution of contextual variables (see column labels).

6 Conclusion

In this paper, we extended Contextual BO [Krause and Ong, 2011] to settings in which the contextual variables can not only be observed, but also intervened on at a cost. We introduced SADCBO, an algorithm designed to sort out relevant context variables affecting the experimental outcomes by efficiently leveraging information present in both the observational and the interventional data. SADCBO results in more adequate surrogate models, and ensures the reproducibility of experiments by controlling for such relevant variables. In that respect, SADCBO should be used for practical applications where contextual variables can have an influence while being controllable. This would include for example high-throughput materials and molecule exploration loops that are being increasingly utilized in both academic and industrial laboratories for drug design and new material development [Zhang et al., 2020, Gómez-Bombarelli et al., 2018]. Our variable selection approach can also be combined with any GP surrogate. Thus, if a practitioner feels that a specific contextual variable should be present, this can be achieved in a straightforward manner. Conversely, the variable selection procedure could be generalized to discard design variables as well.

Limitations and future work. To achieve cost-efficiency, SADCBO integrates the query cost at the variable selection level and employs an early stopping criterion. The latter only depends on an upper bound on the instantaneous regret difference, and is therefore not cost-aware. Adding a notion of remaining budget to this criterion would certainly benefit our approach, in line with the cost-aware BO method developed by Lee et al. [2021]. Furthermore, a very recent work [Branchini et al., 2023] proposed to perform BO under the assumption that the input variables and the output are linked by a

causal directed acyclic graph, learning the graph whilst maximizing the objective function. Despite its high computational complexity, applying this technique to our particular problem might be promising.

References

Milad Abolhasani and Keith A. Brown. Role of AI in experiment materials science. *MRS Bulletin*, 2023.

Vahan Arsenyan, Antoine Grosnit, and Haitham Bou-Ammar. Contextual causal Bayesian optimisation. *arXiv preprint arXiv:2301.12412*, 2023.

Maximilian Balandat, Brian Karrer, Daniel Jiang, Samuel Daulton, Ben Letham, Andrew G Wilson, and Eytan Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Ilija Bogunovic, Jonathan Scarlett, Stefanie Jegelka, and Volkan Cevher. Adversarially robust optimization with Gaussian processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Nicola Branchini, Virginia Aglietti, Neil Dhir, and Theodoros Damoulas. Causal entropy optimization. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2023.

Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

David Eriksson and Martin Jankowiak. High-dimensional Bayesian optimization with sparse axis-aligned subspaces. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence (UAI)*, 2021.

Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, 4(2):268–276, 2018.

A Gretton, K Borgwardt, M J Rasch, and B Scholkopf. A kernel two-sample test. *J. of Mach. Learn. Res.*, 13:723–773, 2012.

Kate Higgins, Maxim Ziatdinov, Sergei Kalinin, and Mahshid Ahmadi. High-throughput study of antisolvents on the stability of multicomponent metal halide perovskites through robotics-based synthesis and machine learning approaches. *Journal of the American Chemical Society*, 2021.

Hideaki Ishibashi, Masayuki Karasuyama, Ichiro Takeuchi, and Hideitsu Hino. A stopping criterion for Bayesian optimization by the gap of expected minimum simple regrets. In *Proceedings of The International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2023.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization*, 13:455–492, 1998.

Johannes Kirschner, Ilija Bogunovic, Stefanie Jegelka, and Andreas Krause. Distributionally Robust Bayesian Optimization. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

Ksenia Korovina, Sailun Xu, Kirthevasan Kandasamy, Willie Neiswanger, Barnabas Poczos, Jeff Schneider, and Eric Xing. ChemBO: Bayesian Optimization of Small Organic Molecules with Synthesizable Recommendations. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

Andreas Krause and Cheng Ong. Contextual Gaussian Process Bandit Optimization. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

Eric Hans Lee, Valerio Perrone, Cedric Archambeau, and Matthias Seeger. Cost-aware bayesian optimization, 2020.

Eric Hans Lee, David Eriksson, Valerio Perrone, and Matthias Seeger. A Nonmyopic Approach to Cost-Constrained Bayesian Optimization. *arXiv preprint arXiv:2106.06079*, 2021.

Cheng Li, Sunil Gupta, Santu Rana, Vu Nguyen, Svetha Venkatesh, and Alistair Shilton. High Dimensional Bayesian Optimization Using Dropout. *arXiv preprint arXiv:1802.05400*, 2018.

Sulin Liu, Qing Feng, David Eriksson, Benjamin Letham, and Eytan Bakshy. Sparse Bayesian Optimization. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2023.

Philip W. Nega, Zhi Li, Victor Ghosh, Janak Thapa, Shijing Sun, Noor Titan Putri Hartono, Mansoor Ani Najeeb Nellikkal, Alexander J. Norquist, Tonio Buonassisi, Emory M. Chan, and Joshua Schrier. Using automated serendipity to discover how trace water promotes and inhibits lead halide perovskite crystal formation. *Applied Physics Letters*, 119(4), 07 2021.

C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Paul Rolland, Jonathan Scarlett, Ilija Bogunovic, and Volkan Cevher. High-Dimensional Bayesian Optimization via Additive Models with Overlapping Groups. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.

Isaac Sebenius, Topi Paananen, and Aki Vehtari. Feature collapsing for gaussian process variable ranking. In *Proceedings of The International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

Yihang Shen and Carl Kingsford. Computationally Efficient High-Dimensional Bayesian Optimization via Variable Selection. *arXiv preprint arXiv:2109.09264*, 2021.

I.M Sobol. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and Computers in Simulation*, 55(1):271–280, 2001.

Adrien Spagnol, Rodolphe Le Riche, and Sébastien Da Veiga. Bayesian optimization in effective dimensions via kernel-based sensitivity indices. In *Proceedings of the International Conference on Applications of Statistics and Probability in Civil Engineering (ICASP)*, 2019.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, may 2012. doi: 10.1109/tit.2011.2182033.

Saul Toscano-Palmerin and Peter I Frazier. Bayesian optimization with expensive integrands. *arXiv preprint arXiv:1803.08661*, 2018.

James T. Wilson, Riccardo Moriconi, Frank Hutter, and Marc Peter Deisenroth. The reparameterization trick for acquisition functions. *arXiv preprint arXiv:1712.00424*, 2017.

Yichi Zhang, Daniel W Apley, and Wei Chen. Bayesian optimization for materials design with mixed quantitative and qualitative variables. *Scientific Reports*, 10(1):1–13, 2020.

Juliusz Ziomek and Haitham Bou-Ammar. Are Random Decompositions all we need in High Dimensional Bayesian Optimisation?, 2023.

Appendix — Cost-aware learning of relevant contextual variables within Bayesian Optimization

Outline. The Appendix is organized as follows. In Section A, we describe an alternative variable relevance measure based on maximum mean discrepancy, and provide empirical evidence that it performs worse than the measure based on predictive posterior sensitivity analysis implemented in SADCBO. Section B contains additional experimental results regarding the distribution of early stopping time for SADCBO under different conditions. Section C reports further ablation studies, on varying the number of irrelevant contextual variables present, the hyperparameters of SADCBO and the GP surrogate kernel structure. Section D presents a forward selection approach to perform variable selection once the sensitivity indices have been computed and evaluates this approach on synthetic examples. Finally, Section E contains the analytical expressions of the synthetic examples used throughout the paper.

A SADCBO outperforms Maximum Mean Discrepancy-based variable selection

Spagnol et al. [2019] introduced a BO algorithm with a variable selection procedure based on the Hilbert Schmidt Independence Criterion (HSIC). This measure can be used in our setting as well. We now briefly describe how it is defined.

As introduced in the main text, let $\mathcal{Z} \subset \mathbb{R}^c$ be the space of contextual variables, and $\mathcal{H}$ be a Hilbert space of $\mathbb{R}$-valued functions on $\mathcal{Z}$. Assume that $k : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ is the unique positive definite kernel associated with the Reproducing Kernel Hilbert Space $\mathcal{H}$. Let $\mu_{\mathbb{P}_Z}$ be the kernel mean embedding of the distribution $\mathbb{P}_Z$, $\mu_{\mathbb{P}_Z} := \mathbb{E}_Z[k(Z, \cdot)] = \int_{\mathcal{Z}} k(\mathbf{z}, \cdot) \mathrm{d}\mathbb{P}_Z$. Kernel embeddings of probability measures provide a distance between distributions between their embeddings in the Hilbert Space $\mathcal{H}$, named Maximum Mean Discrepancy (MMD, [Gretton et al., 2012]):

$$\mathrm{MMD}(\mathbb{P}_Z, \mathbb{P}_Y) = \|\mu_{\mathbb{P}_Z} - \mu_{\mathbb{P}_Y}\|^2_{\mathcal{H}}. \tag{S1}$$

For two random variables $Z \sim \mathbb{P}_Z$ on $\mathcal{H}$ and $Y \sim \mathbb{P}_Y$ on $\mathcal{G}$, the HSIC is the squared MMD between the product distribution $\mathbb{P}_{ZY}$ and the product of its marginals $\mathbb{P}_Z\mathbb{P}_Y$,

$$\begin{aligned}
\mathrm{HSIC}(Z, Y) &= \mathrm{MMD}^2(\mathbb{P}_{ZY}, \mathbb{P}_Z\mathbb{P}_Y) && \text{(S2)}\\
&= \|\mu_{\mathbb{P}_{ZY}} - \mu_{\mathbb{P}_Z\mathbb{P}_Y}\|^2_{\mathcal{H}\otimes\mathcal{G}} && \text{(S3)}\\
&= \mathbb{E}_{Z,Y}\mathbb{E}_{Z',Y'}[k(Z, Z')l(Y, Y')] && \text{(S4)}\\
&\quad + \mathbb{E}_Z\mathbb{E}_Y\mathbb{E}_{Z'}\mathbb{E}_{Y'}[k(Z, Z')l(Y, Y')]\\
&\quad - 2\mathbb{E}_{Z,Y}\mathbb{E}_{Z'}\mathbb{E}_{Y'}[k(Z, Z')l(Y, Y')].
\end{aligned}$$

To determine the relevance of a variable $Z^{(i)}$, Spagnol et al. [2019] introduce

$$S^{\mathrm{HSIC}}(Z^{(i)}) = \mathrm{HSIC}(Z^{(i)}, \mathbb{I}(Z \in \mathcal{L}_\gamma)), \tag{S5}$$

with $\mathcal{L}_\gamma$ a region of interest: the locations where the objective function value is above a threshold γ. This measure reflects how important $Z^{(i)}$ is to reach $\mathcal{L}_\gamma$.

We implemented this measure, substituting expectations for empirical means over the dataset $\mathcal{D}$. We use $\gamma = 0.8$, a threshold identical to the one used for SADCBO in Equation (5). The kernel k is chosen to be a RBF kernel, and l is a linear kernel $l(y, y') = yy'$, a common choice for binary data. We create two baselines: MMDCBO, the analogue of SADCBO but using the MMD-based variable relevance measure, and likewise, MMDBO, the analogue of SADBO. Unlike SADCBO, the MMD-based strategy cannot use candidates samples obtained from a batch acquisition function, as no output value is known for these candidates, which prevents the computation of the expectations in Equation (S4). Therefore, for fair comparison, we set $Q = 0$ for SADCBO and SADBO. This means that the dataset on which SADCBO and SADBO perform sensitivity analysis is restricted to $\mathcal{D}^\gamma$ only, as introduced in Equation (5).

The results displayed in Figure S1 show that both SADCBO and SADBO consistently outperform their MMD-based counterparts for the Hartmann6D and Ackley5D functions, even when increasing the number of irrelevant variables. Thus, we decided not to include this measure in our benchmarks presented in the main text.

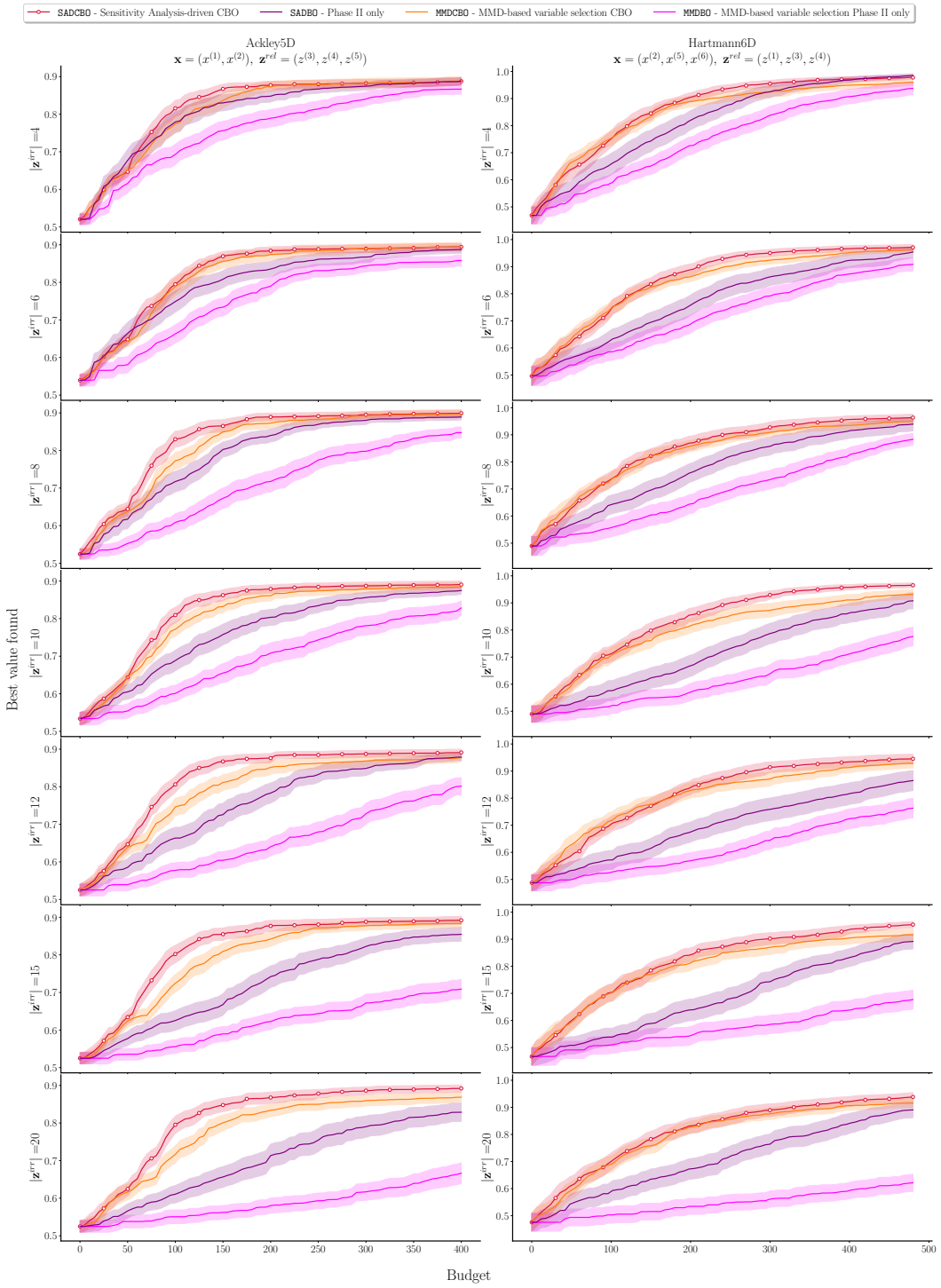

Figure S1: Comparison of predictive posterior and MMD-based sensitivity analysis variable selection techniques when increasing the number of irrelevant contextual variables. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1), and $p(\mathbf{z}) = \mathcal{U}([0,1]^c)$. For both test functions, `SADCBO` and `SADBO` outperform `MMDCBO` and `MMDBO`, even in high dimensional settings.

B Additional experimental results

This section provides additional insights on the hitting time of the early stopping criterion when i) the ratio of relevant contextual variables over design variables increases (Figure S2, mentioned in the text in Section 5.1) and ii) when the contextual variable distribution allocates a different amount of mass to the optimum region (Figure S3, mentioned in the text in Section 5.2).

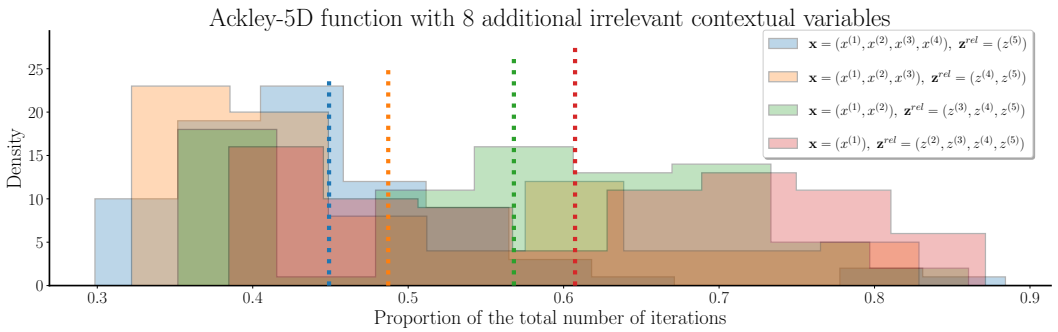

Figure S2: Distribution of early stopping time for SADCBO across 80 different BO trials. We consider the Ackley-5D function with an increasingly larger ratio of relevant contextual variables over design variables, and 8 irrelevant contextual variables. $p(\mathbf{z}) = \mathcal{U}([0,1]^c)$. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). As the impact of contextual variables on the output function grows, the proportion of iterations spent in the observational phase grows as well.

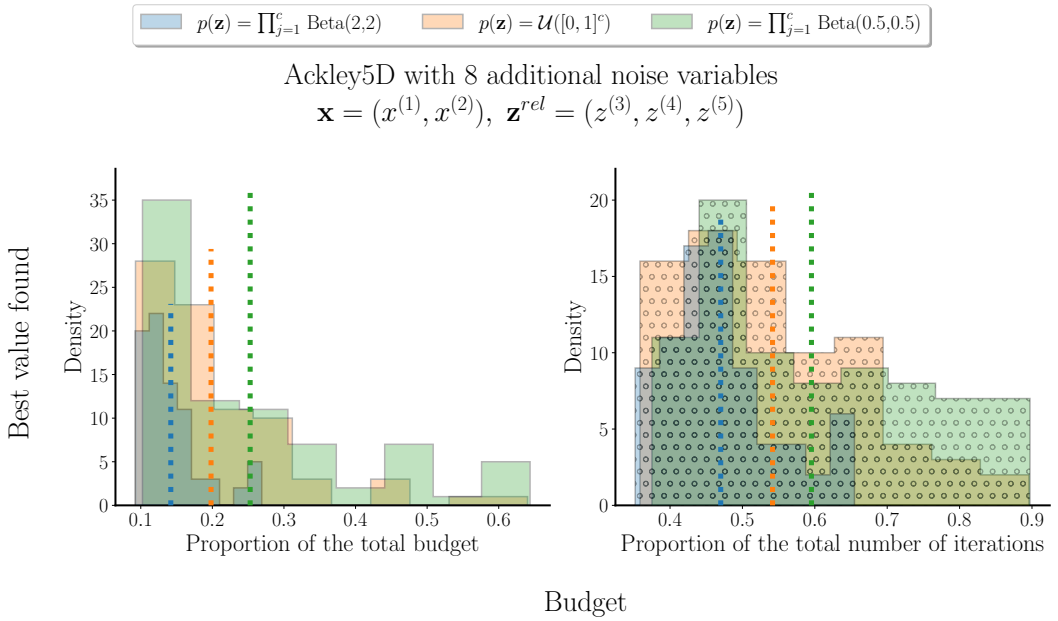

Figure S3: Distribution of early stopping time for SADCBO across 80 different trials. We consider the Ackley-5D function with different contextual distributions $p(\mathbf{z})$. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). From $p(\mathbf{z}) = \prod_{j=1}^{c} \text{Beta}(2,2)$ to $p(\mathbf{z}) = \mathcal{U}([0,1]^c)$ and finally to $p(\mathbf{z}) = \prod_{j=1}^{c} \text{Beta}(0.5, 0.5)$, a decreasing amount of probability mass is allocated to the optimum region $[0.5 - \mathrm{d}\mathbf{z}, 0.5 + \mathrm{d}\mathbf{z}]^c$, leading to an increased early stopping time.

C Further ablation studies

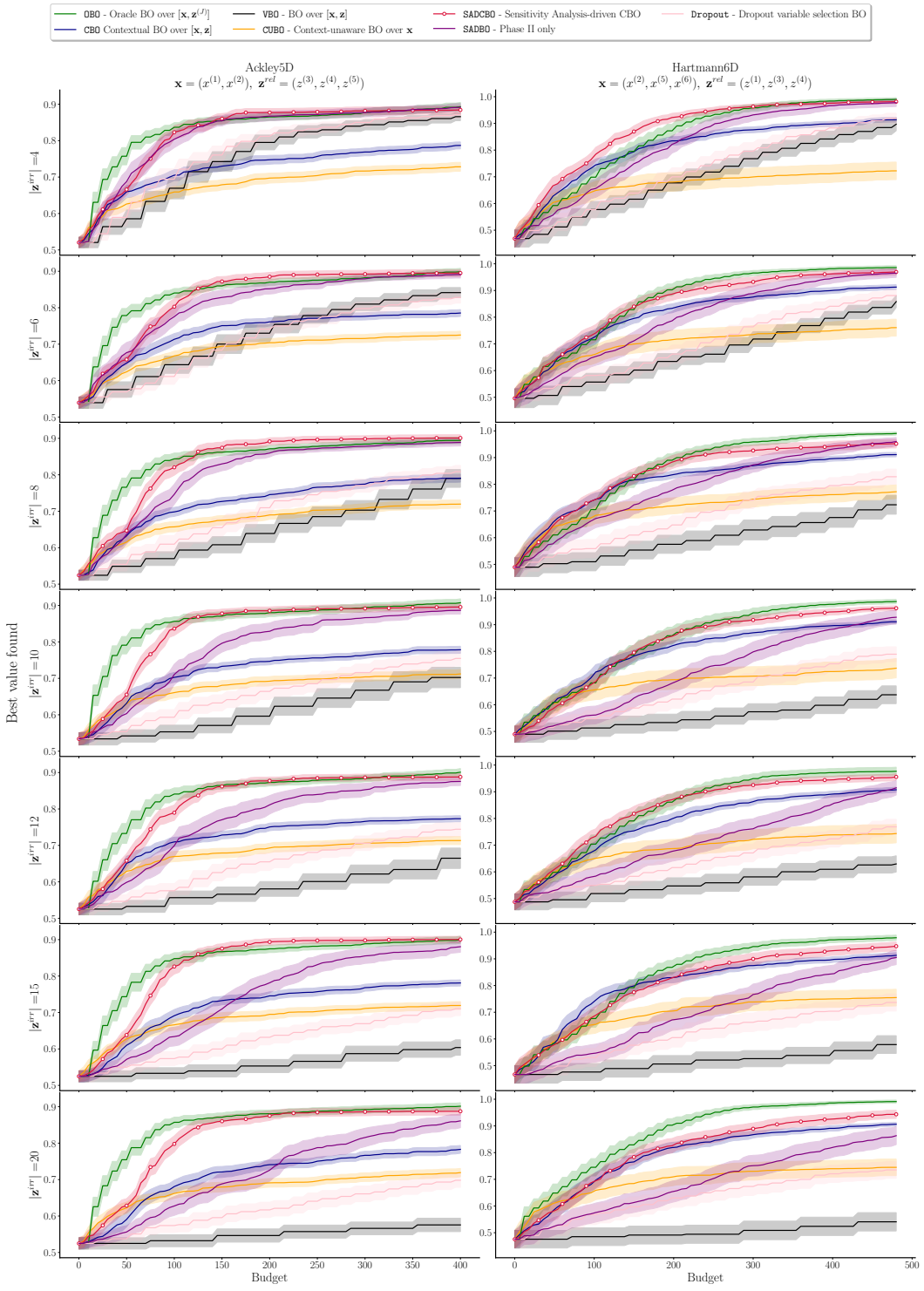

Figure S4: Varying the number of irrelevant contextual variables. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). $p(\mathbf{z}) = \mathcal{U}([0,1]^c)$. For both test functions, SADCBO (red curve) remains competitive compared to the oracle OBO, even in high dimensions.

Number of irrelevant contextual variables. Next, we add an increasingly larger number of irrelevant contextual variables to the input space. Results are reported in Figure S4. Except for `OBO` and `CUBO`, by definition, this induces worse performances for every baseline. Concerning the Ackley-5D function, as dimensionality increases one can observe a signifcant gap between `SADBO` and `SADCBO`. The latter only mildly suffers from dimensionality increase. The story is more or less the same for the Hartmann-6D function, although for this example, pure contextual BO performs on par with `SADCBO` when 15 or more irrelevant variables are added.

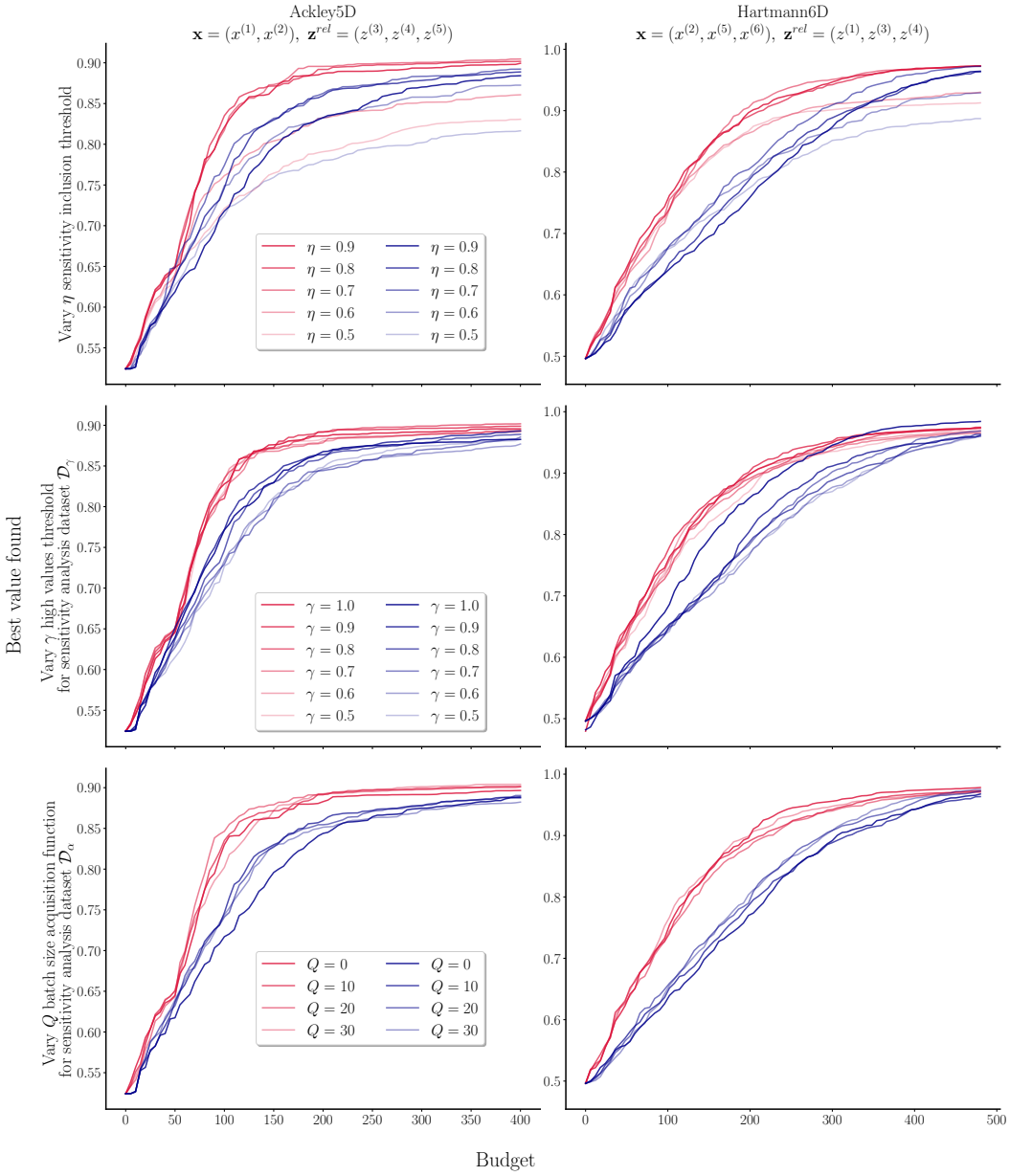

Figure S5: Varying hyperparameters for `SADCBO` and `SADBO`. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). Top: varying η, the contextual variable inclusion threshold over the cumulative sum of sensitivity indices. Middle: varying γ, the threshold used in the creation of the truncated dataset $\mathcal{D}^\gamma$ from Equation (5). Bottom: varying Q, the size of the dataset $\mathcal{D}^Q$ from Equation (6). η is the most sensitive hyperparameter here.

Hyperparameters. We vary the 3 hyperparameters of SADCBO: $\eta \in [0, 1]$ the threshold based over the cumulative sum of sensitivity indices, which in turn regulates how many variables are selected every iteration; $\gamma \in [0, 1]$, a threshold upon which a value is considered high enough to have its input added to dataset $\mathcal{D}^\gamma$ Equation (5), used for sensitivity analysis; and Q the size of the dataset $\mathcal{D}^Q$ Equation (6).

Figure S5 reports the performances both for SADCBO (shades of red) and SADBO (shades of blue). Unsurprisingly, η stands out as the most stringent parameter: as its value decreases, less variables are included, at which point not all relevant ones are selected, leading to reduced performances. Note that in a setting where there are no relevant contextual variable, lower values of η will actually lead to better performances. This is investigated in Section D. Then, varying $\gamma \in [0, 1]$ slightly affects the results: γ increasing means that more samples are collected for sensitivity analysis, but these are less relevant for producing a reliable set of variables accounting for the fluctuations at the optimum. Finally, for the examples considered, Q has only a limited effect, close to that of varying γ. This might stem from the fact that batched acquisition functions are notoriously difficult to optimize and may sometimes struggle at enforcing diversity.

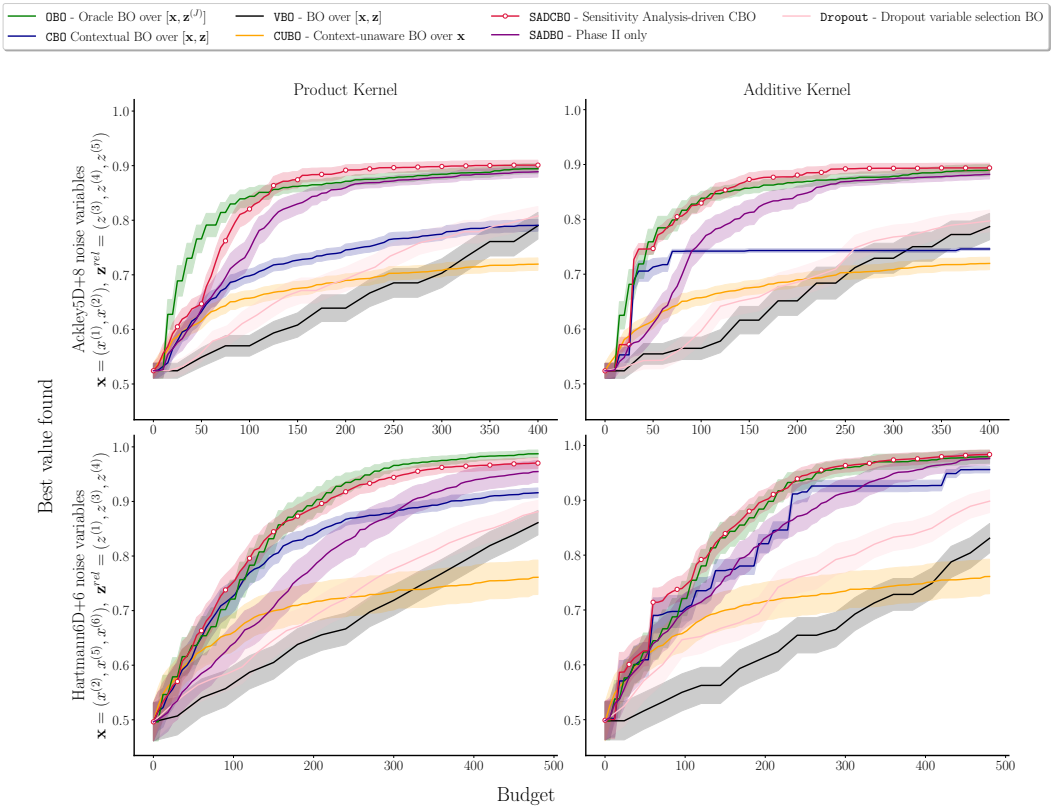

Figure S6: Using different kernel structure for the GP surrogate. Left: product structure over design and contextual variables. Right: additive kernel over design and contextual variables. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). $p(\mathbf{z}) = \mathcal{U}([0, 1]^c)$.

Kernel structure. In every experiment of the paper, we considered a product kernel,

$$k((\mathbf{x}, \mathbf{z}), (\mathbf{x}', \mathbf{z}')) = k_X(\mathbf{x}, \mathbf{x}')k_Z(\mathbf{z}, \mathbf{z}'). \tag{S6}$$

Note that many classical kernels satisfy this structure, e.g. RBF and Matern. Thus, two context-design pairs are similar if the contexts are similar *and* if the designs are similar. As mentioned by Krause and Ong [2011], one can also consider the additive combination

$$k((\mathbf{x}, \mathbf{z}), (\mathbf{x}', \mathbf{z}')) = k_X(\mathbf{x}, \mathbf{x}') + k_Z(\mathbf{z}, \mathbf{z}'), \tag{S7}$$

such that similar context-design pairs can be found to be similar even if designs are not, as soon as contexts are highly similar. We here report performances for both kernels side-by-side in Figure S6, using RBF kernels both k_X and k_Z. Note that CBO alone seems quite sensitive to the specific kernel structure, whereas the baselines which also optimize for the context are less.

D Forward variable selection further improves the performance of SADCBO

In the SADCBO method, once the FC variable relevance indices have been computed using Equation (4) and sorted in descending order, the contextual variables selected are those whose cumulative FC scores is greater than $\eta \in [0, 1]$. This selected set explains the fraction η of the output sensitivity amongst all contextual variables. This can prove problematic in the extreme case where none of the contextual variables at hand have any impact on the function, as we will still have to select sufficiently enough variables to reach $100\eta\%$ of the output sensitivity.

An alternative approach described by Shen and Kingsford [2021] can be employed to tackle this issue. Using the sorted sensitivity indices, one performs forward variable selection by fitting GP surrogates which include an increasingly larger number of (highest ranked) contextual variables. The stopping criterion on the addition of contextual variables to the surrogate is computed based on a comparison of the negative Marginal Log-Likelihood (MLL) between nested models. This introduces an hyperparameter β, which we set to 10, similarly as Shen and Kingsford [2021]. Note that the hyperparameter η is no longer necessary with this approach. Algorithm S2 summarizes the forward variable selection process performed at each BO iteration. We report the performance of this baseline, coined SADCBO + Forward selection on two experiments. In the first scenario (Figure S7), where the number of irrelevant contextual variables is varied, performing forward variable selection based on the sensitivity indices leads to faster convergence to the optimum, specifically as the number of irrelevant contexts grows large ($|\mathbf{z}^{irr}| \geq 12$). In the second scenario (Figure S8), we consider the Ackley5D (resp. Hartmann6D) function, but this time there are no relevant contextual variable: all relevant dimensions are associated to design variables, and there are 8 (resp. 6) additional irrelevant contextual variables. Performing forward variable selection leads to a marginally faster convergence to the optimum.

Algorithm S2 Forward variable selection

Input: Dataset $\mathcal{D}$, contextual variables $[\mathbf{z}_{\text{sort}}^{(1)}, \dots, \mathbf{z}_{\text{sort}}^{(c)}]$ sorted in descending order of relevance according to Equation (4).
Let $L_{\mathbf{z}_{\text{sort}}^{(0)}}$ be the negative MLL of the GP fitted on $\mathcal{D}$ using only design variables $\mathbf{x}$
$j^\star \leftarrow c$
for $j = 1, \dots, c$ **do**
  Fit a GP on $\mathcal{D}$ using $[\mathbf{x}, \mathbf{z}_{\text{sort}}^{(1)}, \dots, \mathbf{z}_{\text{sort}}^{(j)}]$, with negative MLL $L_{\mathbf{z}_{\text{sort}}^{(j)}}$
  if $j = 1$ and $L_{\mathbf{z}_{\text{sort}}^{(j)}} < L_{\mathbf{z}_{\text{sort}}^{(j-1)}}$ **then**
    $j^\star \leftarrow 1$
    break
  else if $L_{\mathbf{z}_{\text{sort}}^{(j)}} < L_{\mathbf{z}_{\text{sort}}^{(j-1)}}$ or $L_{\mathbf{z}_{\text{sort}}^{(j)}} - L_{\mathbf{z}_{\text{sort}}^{(j-1)}} < (L_{\mathbf{z}_{\text{sort}}^{(j-1)}} - L_{\mathbf{z}_{\text{sort}}^{(j-2)}})/\beta$ **then**
    $j^\star \leftarrow j$
    break
  end if
end for
return $[\mathbf{z}_{\text{sort}}^{(1)}, \dots, \mathbf{z}_{\text{sort}}^{(j^\star)}]$

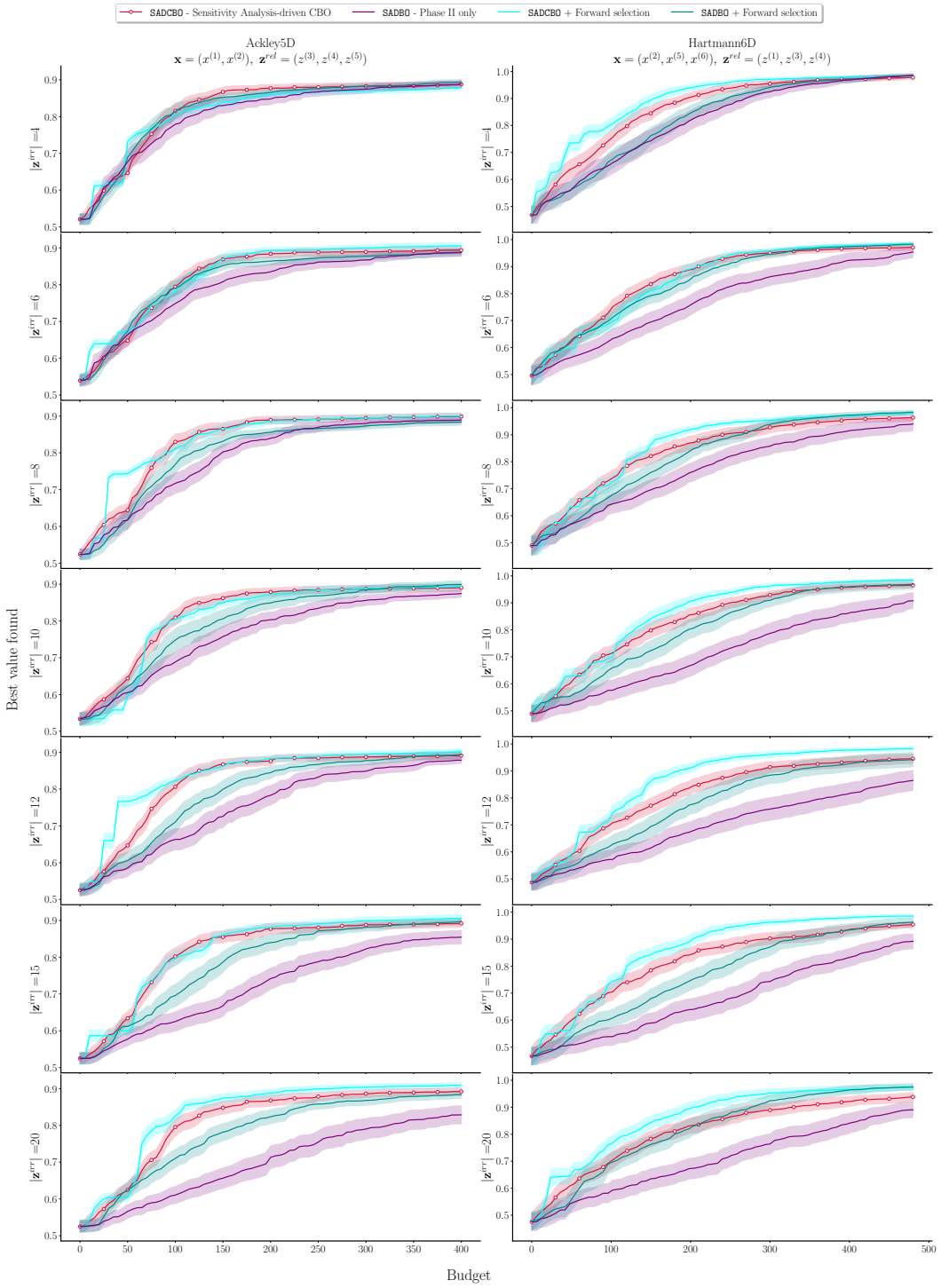

Figure S7: Comparison of SADCBO and SADCBO + forward selection, when increasing the number of irrelevant contextual variables. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). $p(\mathbf{z}) = \mathcal{U}([0,1]^c)$. For both test functions, Sensitivity Analysis-driven CBO (red curve) remains competitive, even in high dimensions. Forward variable selection leads to faster convergence, specifically starting when the number of irrelevant variables $|\mathbf{z}^{irr}|$ reaches 12 or more.

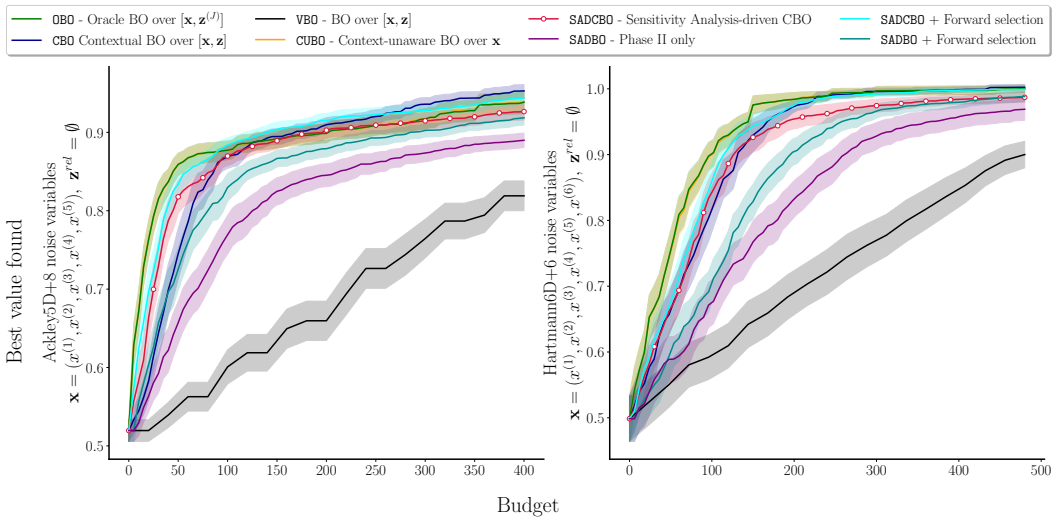

Figure S8: Synthetic examples where no relevant contextual variable are present. For any contextual (resp. design) variable, the associated query cost is 3 (resp. 1). $p(\mathbf{z}) = \mathcal{U}([0,1]^c)$. The baseline performing forward variable selection on top of SADCBO (cyan curve) provides a slight but consistent improvement over SADCBO (red curve), and likewise for SADBO (dark cyan versus purple curve).

E Experiment details

Hartmann-6D function:

$$f(\mathbf{v}) = -\sum_{i=1}^{4} \alpha_i \exp\left(-\sum_{j=1}^{6} A_{ij}(v^{(j)} - P_{ij})\right)$$

$$\alpha = (1.0, 1.2, 3.0, 3.2)^T$$

$$\mathbf{A} = \begin{pmatrix} 10 & 3 & 17 & 3.5 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.05 & 10 & 0.1 & 14 \end{pmatrix}$$

$$\mathbf{P} = 10^{-4} \begin{pmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{pmatrix}$$

defined over $\mathcal{V} = [0,1]^6$. Table 2 provides the results of a Sobol global sensitivity analysis performed using evaluations of the functions collected over a grid of $N = 917504$ samples [Sobol, 2001]. Adding up the first order indices for design and contextual variables separately leads to $S_{\mathbf{x}} \approx 0.124$ and $S_{\mathbf{z}} \approx 0.196$. This means that with respect to first order interactions, contextual variables have more impact than design variables, in this synthetic example. One should notice however that these indices are computed across the whole search space and not specifically at the optimum.

Table 2: Sobol global sensitivity analysis for the Hartmann-6D function using $N = 917504$ samples.

Variable	First order sensitivity indices	Total order sensitivity indices
$z^{(1)}$	0.107	0.343
$x^{(2)}$	0.006	0.399
$z^{(3)}$	0.007	0.052
$z^{(4)}$	0.082	0.379
$x^{(5)}$	0.106	0.297
$x^{(6)}$	0.012	0.482

Ackley 5D function:

$$f(\mathbf{v}) = -20\exp\left(-0.2\sqrt{\frac{1}{5}\sum_{j=1}^{5}(v^{(j)})^2}\right) - \exp\left(\frac{1}{5}\sum_{j=1}^{5}\cos(2\pi v^{(j)})\right) + 20 + e^1$$

defined over $[-5, 5]^5$.