STOBO: SCALING BATCH BAYESIAN OPTIMIZATION TO HIGH DIMENSIONS THROUGH CAUSAL DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

In this work, we develop a unified framework for high-dimensional Bayesian optimization in settings where the optimization variables are causally related but with unknown causal structure. We exploit the synergy between stochastic (hard) interventions in causal discovery and batch Bayesian optimization to design a stochastic Bayesian optimization algorithm that simultaneously targets two goals: efficient black-box optimization and causal structure learning. To guide intervention selection, we develop a gradient-based acquisition function that exploits regions with large expected gradient magnitude while exploring regions with high gradient uncertainty. After each intervention batch, we perform independence tests to update the estimated causal structure. Unlike traditional causal discovery methods, our approach relies only on marginal, rather than conditional, independence tests, and is therefore more sample-efficient. We derive both the asymptotic behavior and non-asymptotic error bounds of the test statistic. Experiments on a broad collection of synthetic benchmark functions and real-world optimization problems demonstrate that StoBO achieves competitive optimization performance while accurately recovering causal parent variables in high-dimensional settings.

1 INTRODUCTION

Bayesian optimization (BO) is a global optimization framework for black-box functions that are expensive or time-consuming to evaluate, and is widely used in applications such as hyperparameter tuning, experimental design, and policy optimization (Turner et al., 2021; Zhang et al., 2023). Classical BO typically assumes that any configuration of the input variables can be queried independently, i.e., the optimizer is free to set each variable to any value within its domain without affecting the others. However, this assumption can be violated in real-world settings: (1) interventions on some variables causally influence others (e.g., increasing advertising affects both sales and product visibility); (2) some variables cannot be directly controlled (e.g., weather influences sales but not vice versa); and (3) causal constraints may govern how changes propagate (e.g., a perturbation in X_1 must flow through X_2 before affecting Y). When the variables in the black-box function are causally dependent, standard BO can be suboptimal because it ignores this underlying structure. Exploration-exploitation policies that ignore causal relationships are typically not sample-efficient: with Gaussian process (GP) surrogates, regret bounds and sample complexity depend on the effective input dimension, so achieving a given accuracy may require thousands of evaluations even in moderate dimensions. Since GP inference scales as $\mathcal{O}(n^3)$, in the number of observations n, such large evaluation budgets also make each BO iteration computationally expensive. In this work, we develop an exploration-exploitation pipeline that jointly optimizes the target outcome and identifies the direct causal factors driving that outcome.

When causal discovery is based on observational data alone, classical constraint-based algorithms can identify the underlying directed acyclic graph (DAG) only up to its Markov equivalence class under the faithfulness assumption (Verma & Pearl, 1990; Spirtes et al., 1995). By incorporating interventional data, related constraint-based methods refine this to an interventional Markov equivalence class, which is a subset of the observational Markov equivalence class, and with a sufficiently rich set of interventions, the DAG can be identified exactly (Hauser & Bühlmann, 2012; Yang et al., 2018). In the score-based paradigm, traditional approaches rely on combinatorial search over DAGs, whereas more recent methods relax this to a continuous optimization problem, where a differentiable score is optimized over a continuously parameterized adjacency matrix under an acyclicity constraint

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(Zheng et al., 2018; Brouillard et al., 2020; Lippe et al., 2022). This framework allows the graph structure and the data-generating mechanisms to be learned jointly with gradient-based methods. However, in both constraint-based and score-based algorithms, the intervention objective is purely causal discovery, rather than jointly optimizing the underlying black-box function.

The two objectives, causal discovery and outcome optimization, complement each other. Identifying the direct causal factors of the outcome variable allows us to restrict optimization to a low-dimensional subspace, making the search more efficient; conversely, by intelligently choosing exploratory queries (i.e., interventions), we generate informative perturbations that help test (conditional) independences among variables. A recent line of work on BO, known as causal Bayesian optimization, incorporates causal relationships among the variables into the sequential optimization procedure (Aglietti et al., 2020; Sussex et al., 2023). These approaches assume that the causal dependencies (i.e., the underlying DAG) are known, while the data-generating mechanisms remain unknown. By contrast, in this work, we address the more challenging setting in which both the DAG and the data-generating mechanisms are unknown. In Duong et al. (2025), BO is used not to design interventions, but as a black-box optimizer over a continuous low-rank parameterization of DAGs, with each BO proposal decoded into a graph and scored on purely observational data. Toth et al. (2022) place a Bayesian prior over both DAGs and structural functions. They use DiBS (Brouillard et al., 2020) to approximate the posterior over graphs, and at each step select interventions myopically by maximising the mutual information between the potential interventional outcome and the target query. To optimise over continuous intervention values, they place a GP surrogate over this mutual-information utility and run a standard BO loop. However, the procedure is computationally demanding and relies on several approximations. A related strategy is adopted by Branchini et al. (2023), who instead place a discrete uniform prior over a finite set of DAGs; effectively, their method is not a full structure learner over the DAG space, but an optimiser over a predefined catalogue of candidate structures.

Existing methods for high-dimensional BO (that do not account for causal structure) can broadly be grouped into embedding-based approaches and variable-selection approaches. Embedding methods either randomly embed the high-dimensional input space into a lower-dimensional subspace (Nayebi et al., 2019; Letham et al., 2020) or learn a low-dimensional embedding map from data (Tripp et al., 2020; Zhang et al., 2019). In the variable-selection paradigm, it is assumed that only a subset of input dimensions meaningfully influences the objective function, while the remaining variables are effectively irrelevant (Li et al., 2017; Song et al., 2022). In this work, we explicitly account for causal relations among the variables and develop an efficient algorithm that leverages this hidden structural information. Our main contributions are:

A synergistic intervention principle. Property 1 reveals that intervening on all input variables (i) reduces identification of direct parents to marginal independence testing and (ii) is naturally compatible with BO, which queries the configurations of all input variables.

A dual-purpose batch acquisition strategy. In addition to a conventional acquisition function for identified parent variables, we develop a gradient-guided acquisition function for non-parents to empower the following independence tests, while maintaining exploration.

An independence test for heterogeneous batches. We develop an independence test statistic that aggregates batches while allowing each batch to follow a different distribution. Despite distributional heterogeneity, test power increases with the number of batches.

Definitions and background on causal discovery are summarized in Section 2. Section 3 describes the stochastic Bayesian optimization algorithm (StoBO) in detail. Section 4 introduces our independence test and establishes its properties. In Section 5, we present a comprehensive numerical study. All proofs are collected in the appendix.

2 THE DUAL-OBJECTIVE PROBLEM

We consider a random vector $\mathbf{X} = (X_0, X_1, \ldots, X_d)^\top$ with a joint distribution $P_{\mathbf{X}}$ and a density function $p_{\mathbf{X}}$. The variable X_0 is the target of interest and is expressed as a function of the remaining d variables: $X_0 = f_0(X_1, \ldots, X_d, U_0)$, where f_0 is an unknown (black-box) function, and U_0 is an exogenous variable capturing random noise or unobserved effects. Our objective is to determine the

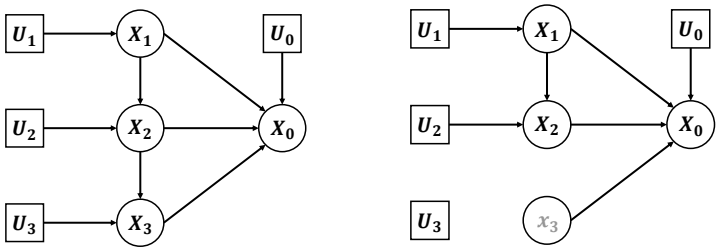


Figure 1: Left: A DAG with nodes $\{X_0, X_1, X_2, X_3\}$ and exogenous variables $\{U_0, U_1, U_2, U_3\}$. Right: An intervention on X_3, $do(X_3 = x_3)$, breaks the directed edges $X_2 \to X_3$ and $U_3 \to X_3$.

value of $\mathbf{X}^0 = (X_1, \ldots, X_d)^\top$ that maximizes the expected value of X_0:

$$\max_{(x_1,\ldots,x_d)\in D(\mathbf{X}^0)} E_{U_0}[f_0(x_1, \ldots, x_d, U_0)], \tag{1}$$

where $D(\mathbf{X}^0)$ represents the (compact) domain of $\mathbf{X}^0$. Unlike standard BO methods, in which the variables $\{X_1, \ldots, X_d\}$ are mutually independent or weakly associated, our approach explicitly accounts for underlying cause-and-effect relationships among the variables. If only a subset of $\{X_j\}_{j=1}^d$ directly influences X_0, optimization can be restricted to these direct causes rather than considering all d variables, leading to a more efficient and interpretable solution.

Graphical Modelling We model causal dependencies using a DAG $\mathcal{G} = (V, E)$, where the nodes $V = \{X_0, X_1, \ldots, X_d\}$ represent the random variables, and the edges $E = \{(X_k, X_j)\}$ encode causal relationships (Pearl, 2010; Scutari & Strimmer, 2011). An edge $(X_k, X_j) \in E$ corresponds to the arrow $X_k \to X_j$, indicating that X_k is a direct cause of X_j. The set of direct parents of X_j is denoted as $pa(X_j) := \{X_k : (X_k, X_j) \in E\}$.

To model the data-generating process, we leverage the framework of structural causal model (SCM). In particular, given the DAG $\mathcal{G}$, the causal mechanism between X_j and its parents $pa(X_j)$ is

$$X_j = f_j(pa(X_j), U_j), \tag{2}$$

for $0 \le j \le d$, where $\{U_j\}_{j=0}^d$ are exogenous (background) variables, and $\{f_j\}_{j=0}^d$ are unknown structural equations; see Figure 1. Each exogenous variable U_j is independent of $pa(X_j)$, i.e., $U_j \perp\!\!\!\perp pa(X_j)$. The joint distribution $P_{\mathbf{X}}$ of the endogenous variables $\mathbf{X}$ is then uniquely determined by the distribution $P_{\mathbf{U}}$ of the exogenous variables $\mathbf{U} = (U_0, U_1, \ldots, U_d)^\top$. The entailed distribution $p_{\mathbf{X}}$ satisfies the Markov condition with respect to $\mathcal{G}$, meaning that each variable is conditionally independent of its non-descendants given its parents. In particular, the conditional density follows

$$p_{X_j}(X_j|pa(X_j)) = \mathrm{E}_{U_j}[\mathbb{1}(X_j = f_j(pa(X_j), U_j))|pa(X_j)],$$

where $\mathbb{1}(\cdot)$ is the 0-1 function. The joint density factorizes as $p_{\mathbf{X}}(\mathbf{X}) = \prod_{j=0}^d p_{X_j}(X_j|pa(X_j))$. We assume that the exogenous variables $\{U_j\}_{j=0}^d$ are mutually independent, i.e., $U_j \perp\!\!\!\perp U_k$ for any $0 \le k < j \le d$. This assumption guarantees the absence of hidden confounders, a property known as *causal sufficiency*[1]. We further assume that the entailed distribution $p_{\mathbf{X}}$ is *faithful* to $\mathcal{G}$; that is, there exists a one-to-one relationship between the conditional independence statements in $p_{\mathbf{X}}$ and the d-separation statements in $\mathcal{G}$.

Interventions Let π be the index set of the true direct parents of X_0: $pa(X_0) = \mathbf{X}_\pi$. Under the causal sufficiency and faithfulness assumptions, the optimization problem (1) reduces to a lower-dimensional problem where the optimization variables are the direct causal parents of X_0:

$$\max_{\mathbf{x}\in D(\mathbf{X}_\pi)} g(\mathbf{x}) = E_{U_0}[f_0(\mathbf{X}_\pi = \mathbf{x}, U_0)]. \tag{3}$$

However, both the structural function f_0 and the causal graph $\mathcal{G}$ – including the set of direct causal parents $pa(X_0)$ – are unknown. This leads to a dual-objective BO:

[1]For simplicity, we present our main results under this assumption, but causal sufficiency is not essential for our StoBO algorithm, as explained in Appendix A.

(O1) approximating and maximizing the black-box function g, and

(O2) identifying the direct causal parents of X_0.

For objective (O1), if the variables $\{X_j\}_{j=1}^d$ were independent, evaluating f_0 at a query point would not alter the underlying data-generating process. However, due to causal dependencies among the variables, fixing a variable at specific values disrupts its dependence on its parents, thereby altering the causal mechanism. For objective (O2), causal models are generally not identifiable from observational data alone. Learning the local causal structure around X_0 requires actively intervening on variables, which explicitly modifies the causal mechanism.

An (atomic) intervention on a manipulable variable X_k, denoted by $do(X_k = x_k)$, replaces the structural assignment $X_k = f_k(pa(X_k), U_k)$ with the constant $X_k = x_k$. This results in a modified distribution, called the *interventional distribution*, given by the truncated factorization:

$$p_{\mathbf{X}}^{do(X_k=x_k)}(\mathbf{X}) = \prod_{j=0, j\neq k}^{d} p_{X_j}(X_j|pa(X_j))|_{X_k=x_k}, \tag{4}$$

$$\text{where} \quad p_{X_j}(X_j|pa(X_j))|_{X_k=x_k} = \begin{cases} p_{X_j}(X_j|pa(X_j)\setminus\{X_k\}, X_k = x_k), & \text{if } X_k \in pa(X_j); \\ p_{X_j}(X_j|pa(X_j)), & \text{otherwise.} \end{cases}$$

That is, an experiment in the context of causal modeling consists of performing $do(X_k = x_k)$ and observing a realization of $\{X_j\}_{j\neq k}$ from the resulting interventional distribution. In terms of the underlying DAG, an intervention on X_k removes all incoming edges to X_k in $\mathcal{G}$, resulting in a mutilated graph denoted as $\mathcal{G}^{do(X_k=x_k)}$. Figure 1 illustrates this process for an intervention on X_3, where the directed edges $X_2 \to X_3$ and $U_3 \to X_3$ are deleted.

3 Batch Bayesian optimization with stochastic queries

Given observational data alone, without additional assumptions on the structural assignments $f_j(pa(X_j), U_j)$, the true underlying causal structure $\mathcal{G}$ can be identified only up to its Markov equivalence class (Spirtes et al., 2001; Hoyer et al., 2008). Interventions offer greater sample efficiency than passive observations in causal discovery by actively manipulating variables, directly revealing causal relationships that would otherwise require significantly more data to infer (Hauser & Bühlmann, 2012). Moreover, in the BO framework, interventions align with the active learning principle. Rather than the atomic intervention $do(X_j = x_j)$, we here adopt the stochastic intervention strategy: $do(X_j = W_j)$, where W_j is a random variable with known distribution (Korb et al., 2004; He & Geng, 2008). More specifically, intervening on X_j involves assigning to it a sequence of values $\{w_{j1}, w_{j2}, \ldots\}$, randomly drawn from the distribution of W_j. Consequently, in the interventional distribution factorization (4), the conditional distribution of X_j changes from $p_{X_j}(X_j|pa(X_j))$ to the distribution of W_j, denoted by p_{W_j}.

If there is a direct causal relationship $X_j \to X_0$ in $\mathcal{G}$, then X_j and X_0 remain dependent in the post-intervention graph after intervening on X_j alone. However, the converse is not necessarily true: if X_j is an ancestor of X_0 through a directed path (e.g., $X_j \to X_k \to X_0$), intervening on X_j alone may still result in dependence between X_j and X_0 in the post-intervention graph. To identify the direct parents of X_0, *we propose intervening on all the variables* $\{X_j\}_{j=1}^d$, as justified by the following property:

Property 1. *Let $\mathcal{G}$ be a DAG with nodes $V = \{X_j\}_{j=0}^d$, and assume post-interventional faithfulness for the hard intervention $do(\mathbf{X}^0 = \mathbf{w})$. Let $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{w})}(\mathbf{X})$ be the interventional distribution, and $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$ the post-intervention graph. Then for any $1 \leq j \leq d$, $X_0 \not\perp\!\!\!\perp X_j$ under $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{w})}(\mathbf{X})$, if and only if there is a directed edge $X_j \to X_0$ in $\mathcal{G}$.*

In Property 1, we assume post-interventional faithfulness. For an SCM whose associated DAG contains no unobserved confounders, the resulting interventional distributions satisfy this assumption under mild regularity conditions (Squires & Uhler, 2023). Property 1 holds for both atomic and stochastic interventions. In both cases, the post-intervention graph is identical; the only difference is that, under stochastic interventions, the intervened variables X_j's have non-degenerate distributions.

The stochastic intervention on $\mathbf{X}^0 = (X_1, \ldots, X_d)^\top$ is denoted by $do(\mathbf{X}^0 = \mathbf{W})$, where an intervention $do(\mathbf{X}^0 = \mathbf{w})$ assigns $\mathbf{X}^0$ the value $\mathbf{w}$ randomly drawn from the distribution of $\mathbf{W} = (W_1, \ldots, W_d)^\top$. The variables W_j are mutually independent: $W_j \perp\!\!\!\perp W_k$ for any $1 \leq k < j \leq d$. The interventional distribution is given by $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{W})} = p_{X_0}(X_0|\mathbf{X}_\pi = \mathbf{W}_\pi) \prod_{j=1}^d p_{W_j}$. To effectively balance the two objectives – O1 (efficient exploration-exploitation of g) and O2 (causal discovery) – we adopt a batch-mode BO approach that selects m query points, denoted by $\{\mathbf{w}_i\}_{i=1}^m$, at each iteration. Conventional batch BO selects queries solely to achieve O1 (Gonzalez et al., 2016). In contrast, *we recast each query as a controlled intervention* and design each batch to be informative for both optimizing g and detecting causal dependence. In particular, in Algorithm 1, $p_{\mathbf{W}}$ is a multivariate Gaussian distribution with diagonal covariance. We optimize the mean vector of $p_{\mathbf{W}}$ to concentrate the interventions $\{\mathbf{w}_i\}_{i=1}^m$ in regions with high acquisition-function values (Block 2) and large posterior expected gradient magnitudes (Block 3). The former promotes efficient optimization of g, while the latter enhances the statistical power of the independence test for causal discovery. Figure 2 gives an overview of the four blocks in Algorithm 1.

Algorithm 1 Stochastic Bayesian optimization (StoBO).

Input: Batches from previous iterations: $\mathcal{B}_{1:t} = \{\mathcal{B}_1, \ldots, \mathcal{B}_t\}$.
Output: New stochastic-intervention batch $\mathcal{B}_{t+1}$.

B1
1: Apply independence tests to update the estimated parent set of X_0:
$$\hat{\pi} = \{j : X_j \not\perp\!\!\!\perp X_0 \mid \mathcal{B}_{1:t},\ 1 \leq j \leq d\}.$$

B2
2: Let $\mathcal{D}_t^{\mathrm{pa}}$ contain all observations in $\mathcal{B}_{1:t}$, retaining only the components corresponding to $\mathbf{X}_{\hat{\pi}}$ and X_0.
3: Fit a parent-variable GP surrogate model to $\mathcal{D}_t^{\mathrm{pa}}$, denoted by $\mathcal{GP}_t^{\mathrm{pa}}$, and compute
$$\mathbf{w}_*^{\mathrm{pa}} = \underset{\mathbf{x} \in D(\mathbf{X}_{\hat{\pi}})}{\arg\max}\, \alpha(\mathbf{x}; \mathcal{GP}_t^{\mathrm{pa}}),$$
where $\alpha(\mathbf{x}; \mathcal{GP}_t^{\mathrm{pa}})$ is the BO acquisition function.

B3
4: Fit a GP model to the whole data $\mathcal{B}_{1:t}$ to approximate the conditional mean $g(\mathbf{x}) = \mathbb{E}[X_0|\mathbf{X}^0 = \mathbf{x}]$, and maximize the total posterior expected gradient magnitude:
$$\mathbf{u}_{t+1} = \underset{\substack{\mathbf{x} \in D(\mathbf{X}^0) \\ \mathbf{x}_{\hat{\pi}} = \mathbf{w}_*^{\mathrm{pa}}}}{\arg\max} \sum_{i \in [d] \setminus \hat{\pi}} \mathbb{E}\big[\big|\frac{\partial g(\mathbf{x})}{\partial x_i}\big| \,\big|\, \mathcal{B}_{1:t}\big],$$
where the maximization is over $\mathbf{x}_{[d]\setminus\hat{\pi}}$, with $\mathbf{x}_{\hat{\pi}}$ fixed at $\mathbf{w}_*^{\mathrm{pa}}$.

B4
5: Randomly sample m stochastic interventions from $\mathbf{W} \sim \mathcal{N}(\mathbf{u}_{t+1}, \mathrm{diag}(\boldsymbol{\sigma}^2))$, and denote them by $\{\mathbf{w}_i^{t+1}\}_{i=1}^m$.
6: Perform the interventions $\{do(\mathbf{X}^0 = \mathbf{w}_i^{t+1})\}_{i=1}^m$ and collect the values of X_0: $\{\chi_i^{t+1}\}_{i=1}^m$.
7: $\mathcal{B}_{t+1} \leftarrow \{(\mathbf{w}_i^{t+1}, \chi_i^{t+1})\}_{i=1}^m$.

Unlike conventional BO algorithms, StoBO incorporates a causal discovery step (Block 1). In Section 4, we introduce an independence-testing method capable of detecting all forms of dependence rather than only linear or monotone relations.

The stochastic intervention distribution is a multivariate Gaussian $\mathbf{W} \sim \mathcal{N}(\mathbf{u}_{t+1}, \mathrm{diag}(\boldsymbol{\sigma}^2))$ with a fixed diagonal covariance matrix, $\mathrm{diag}(\boldsymbol{\sigma}^2)$, which controls the sampling variability along each dimension; The standard-deviation vector $\boldsymbol{\sigma}$ is determined from the data, as detailed in Appendix D. The mean vector $\mathbf{u}_{t+1}$ is optimized separately for each batch: Block 2 determines the components in $\mathbf{u}_{t+1}$ corresponding to the identified parent variables $\mathbf{X}_{\hat{\pi}}$, while Block 3 determines the remaining components. The parent and non-parent components of the mean vector are chosen to serve different objectives: for $\mathbf{X}_{\hat{\pi}}$, the Gaussian mean is the maximizer of the acquisition function $\alpha(\mathbf{x}; \mathcal{GP}_t^{\mathrm{pa}})$; for $\mathbf{X}^0_{-\hat{\pi}}$, the Gaussian mean is placed in a region where the latent function has a sufficiently large expected ℓ_1-norm of the gradient. We next explain the motivation behind the gradient-guided acquisition function; technical details are provided in Appendix E.

Theorem 1 establishes non-asymptotic error bounds for the independence test statistic. Its proof indicates that the test has high power to detect dependence between X_i and X_0 when two conditions hold: (1) $|\frac{\partial g(\mathbf{x})}{\partial x_i}|$ is sufficiently large within a local region (i.e., λ-strong marginal faithfulness), and (ii) the intervention distribution p_{W_i} assigns sufficient probability mass to that region. This motivates our use of $\mathcal{N}(\mathbf{u}_{t+1}, \text{diag}(\boldsymbol{\sigma}^2))$ as the stochastic intervention distribution. Its fixed variance controls the concentration of the interventions around $\mathbf{u}_{t+1}$, while the mean components corresponding to the estimated non-parent variables are selected to target regions with large gradient magnitudes. Because the gradient of g is itself uncertain under the GP prior, we cannot evaluate $|\frac{\partial g(\mathbf{x})}{\partial x_i}|$ directly and therefore score each coordinate by its posterior expectation, $\mathbb{E}\big[|\frac{\partial g(\mathbf{x})}{\partial x_i}| \;\big|\; \mathcal{B}_{1:t}\big]$. It is ready to prove that $\frac{\partial g(\mathbf{x})}{\partial x_i}$ is normally distributed a posteriori with mean $\hat{u}_i(\mathbf{x})$ and standard deviation $\hat{\sigma}_i(\mathbf{x})$. Its absolute value follows a folded normal distribution, and the resulting expectation admits the closed form given in Eq. (11). This score is large under either of two conditions: when the posterior mean slope $\hat{u}_i(\mathbf{x})$ is large in magnitude, indicating a confidently steep local trend, or when the posterior standard deviation $\hat{\sigma}_i(\mathbf{x})$ is large, indicating that a steep slope remains plausible even though the current estimate is near zero. This joint sensitivity to both the estimated slope and its uncertainty allows the acquisition function to target regions of potentially large gradient magnitude while accounting for posterior gradient uncertainty.

For the initial batch $\mathcal{B}_1$, the m intervention values are sampled from $\mathcal{N}(\mathbf{u}_1, \text{diag}(\boldsymbol{\sigma}^2))$, where the mean vector $\mathbf{u}_1$ is randomly selected. Once all query points have been selected, we perform interventions on the causal model and observe the corresponding responses of X_0 (Block 4), thereby completing the batch selection. StoBO terminates when either a given function-evaluation budget is exhausted or when the estimated parent set $\hat{\pi}$ remains unchanged across two consecutive batches.

4 INDEPENDENCE TEST

According to Property 1, *we only need to perform marginal, rather than conditional, independence tests* between X_j and X_0, for $j = 1, \ldots, d$. A key challenge in developing an independence-test statistic for the data $\{\mathcal{B}_b\}_{b=1}^t$ is that the marginal distribution of each variable X_j changes across batches. In particular, the mean of the Gaussian interventional distribution p_{W_j} is governed by the traditional and gradient-guided acquisition functions, both evolving over batches. This heterogeneity prevents the direct application of an independence test to the samples $\{w_{ji}^b : 1 \le i \le m, 1 \le b \le t\}$ and $\{\chi_i^b : 1 \le i \le m, 1 \le b \le t\}$ for assessing the independence between X_j and X_0. To address this, we define X_j^b and X_0^b as the random variables corresponding to the bth batch: $\{w_{ji}^b\}_{i=1}^m \overset{d}{=} X_j^b$ and $\{\chi_i^b\}_{i=1}^m \overset{d}{=} X_0^b$, where $\overset{d}{=}$ denotes equality in distribution. By treating the multivariate data $\{\mathbf{w}_{ji}^{[t]} = (w_{ji}^1, \ldots, w_{ji}^t)^\top\}_{i=1}^m$ and $\{\boldsymbol{\chi}_i^{[t]} = (\chi_i^1, \ldots, \chi_i^t)^\top\}_{i=1}^m$ as samples from the random vectors $\mathbf{X}_j^{[t]} = (X_j^1, \ldots, X_j^t)^\top$ and $\mathbf{X}_0^{[t]} = (X_0^1, \ldots, X_0^t)^\top$, we reformulate the problem of testing $X_j \perp\!\!\!\perp X_0$ as testing the independence between the t-dimensional random vectors $\mathbf{X}_j^{[t]}$ and $\mathbf{X}_0^{[t]}$. Notably, $\mathbf{X}_j^{[t]} \perp\!\!\!\perp \mathbf{X}_0^{[t]}$ if and only if $X_j \perp\!\!\!\perp X_0$. Throughout, for a matrix $\boldsymbol{\Lambda}$, $\Lambda_{ij} = [\boldsymbol{\Lambda}]_{ij}$ refers to the (i, j)th element of $\boldsymbol{\Lambda}$.

The distance covariance introduced by Székely et al. (2007) provides a measure of dependence between two random vectors $\mathbf{X}$ and $\mathbf{Y}$, which may have different dimensions, as long as they have finite first moments. It quantifies a weighted distance between the joint characteristic function of $\mathbf{X}$ and $\mathbf{Y}$ and the product of their marginal characteristic functions. Unlike Pearson correlation, which captures only linear relationships, the population distance covariance $\text{dCov}(\mathbf{X}, \mathbf{Y})$ is zero if and only if $\mathbf{X}$ and $\mathbf{Y}$ are independent, making it sensitive to all forms of dependence. Székely & Rizzo (2014) later developed an unbiased estimator of the population distance covariance. In this work, we adopt the original formulation, as the two versions exhibit very similar type-I error rates and power in our simulations. Given m observations of the random vector $\mathbf{X}$, denoted by $\{\mathbf{x}_i\}_{i=1}^m$, the "distance" matrix $\boldsymbol{\Lambda}(\mathbf{X}) \in \mathbb{R}^{m \times m}$ is

$$\Lambda_{ik}(\mathbf{X}) = \|\mathbf{x}_i - \mathbf{x}_k\| - \frac{1}{m}\sum_{l=1}^m \|\mathbf{x}_l - \mathbf{x}_k\| - \frac{1}{m}\sum_{l=1}^m \|\mathbf{x}_i - \mathbf{x}_l\| + \frac{1}{m^2}\sum_{r,l=1}^m \|\mathbf{x}_r - \mathbf{x}_l\|.$$

The matrix $\mathbf{\Lambda}(\mathbf{Y})$ is defined analogously. Here, $\|\cdot\|$ denotes the ℓ_2-norm. The squared sample distance covariance is then given by

$$\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y}) = \frac{1}{m^2} \sum_{i=1}^{m} \sum_{k=1}^{m} [\mathbf{\Lambda}(\mathbf{X}) \odot \mathbf{\Lambda}(\mathbf{Y})]_{ik},$$

where $\odot$ denotes the Hadamard (element-wise) product of matrices.

While $\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y})$ captures departures from independence between the full vectors $\mathbf{X}$ and $\mathbf{Y}$, in our setting dependence can only arise coordinate-wise (namely, batch-wise): for each $b = 1, \ldots, t$, dependence may occur only between X_0^b and X_j^b, and X_0^b is independent of X_j^r for any $1 \le b \ne r \le t$. Moreover, although the random variables X_j^b and X_j^r in different batches follow different distributions, the underlying causal relationship $X_j \to X_0$, if present, remains persistent. Consequently, all pairs (X_j^b, X_0^b) for $1 \le b \le t$ should exhibit a statistically comparable degree of dependence. Based on these observations, we define a coordinate-wise squared sample distance covariance:

$$\overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]}) = \sum_{b=1}^{t} \mathrm{dCov}_m^2(X_j^b, X_0^b). \tag{5}$$

The following properties show that the statistic $m\ \overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ has a well-defined limiting distribution under independence, and that independence tests based on the coordinate-wise statistic $\overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ enjoy higher asymptotic power than those based on $\mathrm{dCov}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$.

Property 2. *Let X_j^b and X_0^b have finite first moments for all $b = 1, \ldots, t$. If X_j and X_0 are independent, then*

$$\lim_{m \to \infty} m\ dCov_m^2(X_j^b, X_0^b) \overset{d}{=} \|\zeta^b\|^2, \tag{6}$$

where $\zeta^b(r, s)$ is a zero-mean GP, and $\|\zeta^b\|^2 \overset{d}{=} \sum_{k=1}^{\infty} \lambda_k^b Z_k^2$. Here, Z_k are independent standard normal random variables, and λ_k^b are the non-negative eigenvalues of the covariance operator. Then the statistic $m\ \overline{dCov}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ converges in distribution to $\sum_{b=1}^{t} \|\zeta^b\|^2$.

Property 3. *Independence tests based on the coordinate-wise squared sample distance covariance $\overline{dCov}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ are asymptotically more powerful than tests based on the full squared sample distance covariance $dCov_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$.*

For finite samples, its empirical distribution often deviates from the asymptotic counterpart. Therefore, we employ a random permutation test, shuffling the observations $\{\chi_i^b\}_{i=1}^m$ of X_0^b while keeping the observations $\{w_{ji}^b\}_{i=1}^m$ of X_j^b fixed. Computationally, this involves generating a permutation matrix $\mathbf{P}$ and computing the permuted sample distance covariance

$$\mathrm{dCov}_m^2(X_j^b, X_0^b; \mathbf{P}) = \frac{1}{m^2} \sum_{i=1}^{m} \sum_{k=1}^{m} [\mathbf{\Lambda}(X_j^b) \odot (\mathbf{P}\mathbf{\Lambda}(X_0^b))]_{ik}.$$

The corresponding permuted coordinate-wise sample distance covariance is given by

$$\overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]}; \mathbf{P}) = \sum_{b=1}^{t} \mathrm{dCov}_m^2(X_j^b, X_0^b; \mathbf{P}). \tag{7}$$

The permutation test is repeated (e.g., 1000 times), and the p-value is computed as the proportion of permuted test statistics that are greater than or equal to the observed test statistic. When $\mathbf{X}_j^{[t]}$ and $\mathbf{X}_0^{[t]}$ are dependent, performing a permutation test on sample distance covariance yields an asymptotic p-value of 0, as the sample size m increases. Conversely, when $\mathbf{X}_j^{[t]}$ and $\mathbf{X}_0^{[t]}$ are independent, the p-value follows a uniform distribution in the range [0, 1].

Computing $\overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ is highly amenable to parallelization. Moreover, we can leverage the $\mathcal{O}(m \log(m))$ algorithm of Huo & Székely (2016) to achieve efficient implementation. Finally, we have the following theorem on the non-asymptotic sample complexity of the independence test:

Theorem 1. *Assume that the structural function f_0 is continuously differentiable, and for any $X_j \in \mathbf{X}_\pi$, there is a batch index b and a bounded sampling region $\mathcal{R}_b \subset D(\mathbf{X}_\pi)$ and $S \subset D(U_0)$ such that **(1)** the partial derivative w.r.t. x_j is bounded away from zero on $\mathcal{R}_b \times S$: for all $(\mathbf{x}_\pi, u_0) \in \mathcal{R}_b \times S$, $|\frac{\partial f_0}{\partial x_j}(\mathbf{x}_\pi, u_0)| \geq c_j > 0$, and **(2)** the design places non-negligible mass on $\mathcal{R}_b$: $\Pr(\mathbf{X}_\pi \in \mathcal{R}_b) \times \Pr(U_0 \in S) \geq p_j > 0$. Then for the hypothesis test H_0 : $dCov^2(X_j^b, X_0^b) = 0$ vs. H_1 : $dCov^2(X_j^b, X_0^b) \geq \lambda_j > 0$, there exists a universal constant $c > 0$ such that, for any significance level $\alpha \in (0, 1)$ and any upper bound K on the Orlicz ψ_1-norm of the kernel of the squared sample distance covariance $dCov_m^2(X_j^b, X_0^b)$, choosing the sample size $m \geq c\frac{K^2}{\lambda_j^2}\log(\frac{1}{\alpha})$ yields a test whose type I and type II error probabilities are both bounded by α.*

5 EXPERIMENTS

Simulation study We evaluate StoBO on ten test functions that have been used in previous studies on optimization: Michalewicz, Hartmann6, Ackley, Rastrigin, Levy, Griewank, Rosenbrock, Dixon-Price, Powell, and Styblinski-Tang. Details of these functions are provided in Appendix F.1. For all experiments in Table 1, we set $d = 100$, corresponding to (randomly generated) DAGs with 101 nodes; the number of true parents is fixed at five, except for Hartmann6 (six) and Powell (four). Figure 3 shows an example of such a DAG. The batch size is $m = 50$, and the upper confidence bound with $\beta = 2$ is used as the acquisition function in Block 2. In the independence test, a node is identified as a direct parent if the corresponding p-value is smaller than 0.05. StoBO is stopped when $\hat{\pi}$ from two consecutive iterations are identical. The second column of Table 1 reports the number of iterations required to meet this stopping criterion. Columns 3-5 report, respectively, the number of true parents missed (FN), the total number of nodes identified as parents, including both true positives (TP) and false positives (FP), and the absolute/relative gap between the current best value estimated by StoBO and the true global optimum. If not all true parents have been identified at stopping, we continue running the algorithm for 10 more batches; the final column of Table 1 reports the final number of true parents detected. Each experiment is repeated 100 times with different random seeds, and we report the mean and standard deviation of the metrics.

Table 1: The metrics below quantify the number of batches required (t: $\hat{\pi}_t = \hat{\pi}_{t-1}$), and, at termination, the number of true parents missed (FN), the total number of variables identified as parents (TP+FP), and the optimality gap from the global optimum (Gap). The table reports the mean and standard deviation of the metrics over 100 repetitions.

Function (parents)	t	FN	TP+FP	Gap	TP (t+10)
Michalewicz (5)	14.45 (8.70)	2.48 (1.28)	8.48 (2.58)	23.96% (13.42%)	3.34 (1.00)
Hartmann6 (6)	14.74 (7.82)	0.79 (1.03)	9.43 (2.22)	2.20% (4.42%)	5.38 (0.81)
Ackley (5)	17.04 (8.22)	0.19 (0.94)	9.68 (2.59)	1.15 (3.01)	5.00 (0.00)
Rastrigin (5)	12.81 (6.90)	0.34 (1.01)	10.42 (2.19)	16.39 (7.32)	5.00 (0.00)
Levy (5)	12.57 (6.44)	0.84 (0.51)	8.95 (2.06)	1.23 (0.83)	4.45 (0.50)
Griewank (5)	16.46 (9.30)	0.03 (0.03)	9.54 (2.01)	0.59 (1.06)	5.00 (0.00)
Rosenbrock (5)	11.88 (7.83)	0.93 (0.26)	8.94 (2.36)	817.40 (828.78)	4.19 (0.39)
Dixon-Price (5)	12.39 (7.89)	1.08 (0.39)	8.46 (2.09)	196.95 (183.91)	4.12 (0.33)
Powell (4)	10.30 (5.99)	0.10 (0.36)	9.11 (2.31)	21.35 (19.33)	4.00 (0.00)
Styblinski-Tang (5)	14.43 (7.24)	0.23 (0.80)	9.70 (2.11)	4.90% (7.07%)	5.00 (0.00)

Table 1 shows that StoBO typically satisfies the stopping criterion within 10-17 batches. Parent recovery is strongest for Ackley, Rastrigin, Griewank, Powell, and Styblinski-Tang, with fewer than 0.35 true parents missed on average at stopping. Michalewicz is the most challenging function for structure identification, with 2.48 of the five true parents missed at stopping and incomplete recovery even after 10 additional batches. For each test function, the missed parent variables are highly consistent across the 100 repetitions. Parent recovery and optimization accuracy are not perfectly aligned. For example, Hartmann6 achieves a small optimality gap despite missing one or two true parents. This suggests that the missed variables may have limited influence on the response. Overall, StoBO provides effective variable screening, reducing 100 candidate variables to approximately 9 while retaining most true parents, although some false positives remain. Figure

4 in Appendix F.2 shows the evolution of the parent set $\hat{\pi}_t$ (left) and the minimal value of StoBO (right) for a representative Ackley-function run. Sensitivity analyses of the dimension d, the number of parent variables, the batch size m, the number of batches t, and the standard-deviation vector $\boldsymbol{\sigma}$, together with an ablation study of Block 2, are reported in Appendix F.2.

We benchmark StoBO against eight sequential methods for high-dimensional BO: HeSBO (Nayebi et al., 2019), TuRBO (Eriksson et al., 2019), ALEBO (Letham et al., 2020), SAASBO (Eriksson & Jankowiak, 2021), BAxUS (Papenmeier et al., 2022), LassoBO (Hoang et al., 2025), standard BO (Xu et al., 2025) and GIT-BO (Yu et al., 2026). HeSBO, ALEBO, BAxUS and GIT-BO exploit low-dimensional subspace embeddings; SAASBO uses a sparsity-inducing prior to identify axis-aligned structure; LassoBO performs variable selection; and TuRBO conducts local optimization within adaptive trust regions. Implementation and configuration details are provided in Appendix F.3. Figure 10 compares the trajectories of the lowest objective values attained by the nine algorithms across the ten test functions, evaluated at budgets of 250, 500, 750, and 1000 function evaluations. Owing to their high computational costs, ALEBO and SAASBO were evaluated only up to a budget of 500 function evaluations. Both Table 1 and Figure 10 demonstrate that StoBO prioritizes parent-variable discovery during the first 10 to 15 batches and typically identifies a high-quality local optimum after approximately 15 batches. Because StoBO updates its surrogate model only once per batch of m queries, it is approximately m times faster than all the competing methods. This substantially reduces computational cost, although it results in slightly higher best-found objective values.

Real-world problems We evaluate seven algorithms on five real-world problems, which are described in Appendix F.3. ALEBO and SAASBO were excluded owing to their prohibitive computational costs. The true parent variables are unknown. Table 2 reports the mean minimal objective value attained after 1000 function evaluations, averaged over 20 random seeds, while Figure 11 shows the corresponding optimization trajectories. The final row gives the cardinality of $\hat{\pi}_t$, i.e., the number of variables identified as "parents" by StoBO. Overall, StoBO is competitive on all datasets,

Table 2: Mean minimum objective value (standard deviation) attained over 20 random seeds.

	DNA	Mazda	SVM	MOPTA08	CEC2020-38
HeSBO	0.323(0.007)	18.780(2.597)	0.191(0.002)	304.684(4.093)	10.093(1.553)
TuRBO	0.287(0.001)	1.686(0.370)	0.097(0.004)	240.652(3.414)	2.808(0.237)
BAxUS	0.289(0.001)	2.447(0.611)	0.097(0.010)	233.429(2.385)	2.474(0.692)
LassoBO	0.291(0.002)	1.397(0.731)	0.039(0.004)	216.371(1.173)	1.894(0.494)
Standard BO	0.283(0.001)	-10.499(0.553)	0.036(0.003)	218.724(3.477)	2.536(0.508)
GIT-BO	0.306(0.001)	25.876(2.519)	0.197 (0.002)	315.865(3.390)	4.804(0.200)
StoBO	0.296(0.002)	-1.488(0.933)	0.100(0.008)	238.572(4.231)	6.719(0.480)
$\lvert\hat{\pi}_t\rvert$	34.80 (3.66)	44.30 (4.12)	46.90 (3.96)	28.40 (3.23)	23.30 (4.15)

with the strongest relative performance on Mazda. Although it does not attain the lowest objective value, it is substantially faster than all the sequential benchmarks (Table 7). StoBO also identifies a subset of most influential variables, which will help guide subsequent engineering analysis.

6 Conclusion

This paper recasts queries in batch Bayesian optimization as controlled interventions and introduces a stochastic-intervention strategy that reduces identification of the response variable's causal parents to marginal independence tests between the response and each input variable. Although our independence test builds on prior work, we introduce a non-trivial modification to the test statistic that not only accommodates distributional heterogeneity across batches but also provably improves statistical power. The properties of the test statistic motivate an additional gradient-guided acquisition function that complements conventional acquisition and likewise balances exploration and exploitation, while increasing the power of subsequent independence tests. Experiments on synthetic functions and real-word problems demonstrate StoBO's ability to discover influential variables while achieving competitive optimization performance. Its batch formulation reduces the frequency of surrogate updates and allows parallel function evaluations, offering a practical trade-off between optimization accuracy and computational efficiency. These capabilities make StoBO relevant to hyperparameter tuning, simulation-based engineering design, and experimental optimization.

AI USE STATEMENT

In this work, we used generative AI tools (1) for language editing, including grammar, spelling, clarity, and word choice, (2) to support our understanding of technical details in relevant works, (3) to assist with code development, including debugging, syntax clarification, and providing guidance on implementation. We did not use generative AI tools for any tasks requiring mandatory disclosure. We reviewed all AI-assisted work and independently verified the technical explanations and factual information. We take responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

The data and code used in our experiments are all available on GitHub: https://anonymous.4open.science/r/StoBO. Detailed configurations of all algorithms are given in the Appendix.

REFERENCES

Virginia Aglietti, Xiaoyu Lu, Andrei Paleyes, and Javier González. Causal bayesian optimization. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3155–3164, 26–28 Aug 2020.

Nicola Branchini, Virginia Aglietti, Neil Dhir, and Theodoros Damoulas. Causal entropy optimization. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 8586–8605. PMLR, 25–27 Apr 2023.

Philippe Brouillard, Sébastien Lachapelle, Alexandre Lacoste, Simon Lacoste-Julien, and Alexandre Drouin. Differentiable causal discovery from interventional data. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2020. Curran Associates Inc.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3), 2011. doi: 10.1145/1961189.1961199.

Bao Duong, Sunil Gupta, and Thin Nguyen. Causal discovery via bayesian optimization. In *International Conference on Representation Learning*, volume 2025, pp. 54318–54351, 2025.

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*, volume 161 of *Proceedings of Machine Learning Research*, pp. 493–503. PMLR, 2021.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Robin J. Evans. Graphs for margins of bayesian networks. *Scandinavian Journal of Statistics*, 43 (3):625 – 648, 2016. doi: 10.1111/sjos.12194.

Javier Gonzalez, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch bayesian optimization via local penalization. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 648–657. PMLR, 2016.

Alain Hauser and Peter Bühlmann. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *J. Mach. Learn. Res.*, 13(1):2409–2464, 2012.

Yang-Bo He and Zhi Geng. Active learning of causal networks with intervention experiments and optimal designs. *Journal of Machine Learning Research*, 9(84):2523–2547, 2008.

Vu Viet Hoang, Hung The Tran, Sunil Gupta, and Vu Nguyen. High dimensional bayesian optimization using lasso variable selection. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 3106–3114. PMLR, 2025.

Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2008.

Xiaoming Huo and Gábor J. Székely. Fast computing for distance covariance. *Technometrics*, 58 (4):435–447, 2016. doi: 10.1080/00401706.2015.1054435.

Donald R Jones. Large-scale multi-disciplinary mass optimization in the auto industry. In *MOPTA 2008 Conference (20 August 2008)*, volume 64, 2008.

Takehisa Kohira, Hiromasa Kemmotsu, Oyama Akira, and Tomoaki Tatsukawa. Proposal of benchmark problem based on real-world car structure design optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '18, pp. 183–184, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450357647. doi: 10.1145/3205651.3205702. URL https://doi.org/10.1145/3205651.3205702.

Kevin Burt Korb, Lucas Ryan Hope, Ann Elizabeth Nicholson, and Karl Axnick. Varieties of causal intervention. In *Proceedings of the 8th Pacific Rim International Conference on Artificial Intelligence (PRICAI 2004)*, Lecture Notes in Computer Science, pp. 322 – 331. Springer, 2004.

Abhishek Kumar, Guohua Wu, Mostafa Z. Ali, Rammohan Mallipeddi, Ponnuthurai Nagaratnam Suganthan, and Swagatam Das. A test-suite of non-convex constrained optimization problems from the real-world and some baseline results. *Swarm and Evolutionary Computation*, 56:100693, 2020. ISSN 2210-6502. doi: https://doi.org/10.1016/j.swevo.2020.100693. URL https://www.sciencedirect.com/science/article/pii/S2210650219308946.

Ben Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy. Re-examining linear embeddings for high-dimensional bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1546–1558. Curran Associates, Inc., 2020.

Cheng Li, Sunil Gupta, Santu Rana, Vu Nguyen, Svetha Venkatesh, and Alistair Shilton. High dimensional bayesian optimization using dropout. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 2096–2102. AAAI Press, 2017.

Phillip Lippe, Taco Cohen, and Efstratios Gavves. Efficient neural causal discovery without acyclicity constraints. In *International Conference on Learning Representations*, 2022.

Amin Nayebi, Alexander Munteanu, and Matthias Poloczek. A framework for Bayesian optimization in embedded subspaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4752–4761. PMLR, 09–15 Jun 2019.

Leonard Papenmeier, Luigi Nardi, and Matthias Poloczek. Increasing the scope as you learn: Adaptive bayesian optimization in nested subspaces. In *Advances in Neural Information Processing Systems*, volume 35, pp. 11586–11601. Curran Associates, Inc., 2022.

Judea Pearl. An introduction to causal inference. *The International Journal of Biostatistics*, 6(2), 2010. doi: doi:10.2202/1557-4679.1203.

Aaditya Ramdas, Sashank J. Reddi, Barnabás Póczos, Aarti Singh, and Larry Wasserman. On the decreasing power of kernel and distance based nonparametric hypothesis tests in high dimensions. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pp. 3571–3577. AAAI Press, 2015.

Thomas Richardson and Peter Spirtes. Ancestral graph markov models. *Annals of Statistics*, 30(4): 962 – 1030, 2002. doi: 10.1214/aos/1031689015.

Marco Scutari and Korbinian Strimmer. Introduction to graphical modelling, 2011. URL https://arxiv.org/abs/1005.1036.

Yihang Shen and Carl Kingsford. Computationally efficient high-dimensional bayesian optimization via variable selection. In Aleksandra Faust, Roman Garnett, Colin White, Frank Hutter, and Jacob R. Gardner (eds.), *Proceedings of the Second International Conference on Automated Machine Learning*, volume 224 of *Proceedings of Machine Learning Research*, pp. 15/1–27. PMLR, 12–15 Nov 2023. URL https://proceedings.mlr.press/v224/shen23a.html.

Lei Song, Ke Xue, Xiaobin Huang, and Chao Qian. Monte carlo tree search based variable selection for high dimensional bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 35, pp. 28488–28501. Curran Associates, Inc., 2022.

Peter Spirtes, Christopher Meek, and Thomas Richardson. Causal inference in the presence of latent variables and selection bias. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, pp. 499–506, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. The MIT Press, 01 2001. doi: 10.7551/mitpress/1754.001.0001.

Chandler Squires and Caroline Uhler. Causal structure learning: A combinatorial perspective. *Foundations of Computational Mathematics*, 23(5):1781 – 1815, 2023. doi: 10.1007/s10208-022-09581-9.

S. Surjanovic and D. Bingham. Virtual library of simulation experiments: Test functions and datasets. Retrieved August 11, 2026, from `http://www.sfu.ca/~ssurjano`.

Scott Sussex, Anastasia Makarova, and Andreas Krause. Model-based causal bayesian optimization. In *The Eleventh International Conference on Learning Representations*, 2023.

Gábor J. Székely and Maria L. Rizzo. Partial distance correlation with methods for dissimilarities. *The Annals of Statistics*, 42(6):2382 – 2412, 2014. doi: 10.1214/14-AOS1255.

Gábor J. Székely and Maria L. Rizzo. The distance correlation t-test of independence in high dimension. *Journal of Multivariate Analysis*, 117:193–213, 2013. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2013.02.012.

Gábor J. Székely, Maria L. Rizzo, and Nail K. Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007.

Christian Toth, Lars Lorch, Christian Knoll, Andreas Krause, Franz Pernkopf, Robert Peharz, and Julius von Kügelgen. Active bayesian causal inference. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16261–16275. Curran Associates, Inc., 2022.

Austin Tripp, Erik Daxberger, and José Miguel Hernández-Lobato. Sample-efficient optimization in the latent space of deep generative models via weighted retraining. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11259–11272. Curran Associates, Inc., 2020.

Ryan Turner, David Eriksson, Michael McCourt, Juha Kiili, Eero Laaksonen, Zhen Xu, and Isabelle Guyon. Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pp. 3–26. PMLR, 06–12 Dec 2021.

Thomas Verma and Judea Pearl. Equivalence and synthesis of causal models. In *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*, UAI '90, pp. 255–270, USA, 1990. Elsevier Science Inc.

Kenan Šehić, Alexandre Gramfort, Joseph Salmon, and Luigi Nardi. Lassobench: A high-dimensional hyperparameter optimization benchmark suite for lasso. In Isabelle Guyon, Marius Lindauer, Mihaela van der Schaar, Frank Hutter, and Roman Garnett (eds.), *Proceedings of the First International Conference on Automated Machine Learning*, volume 188 of *Proceedings of Machine Learning Research*, pp. 2/1–24. PMLR, 25–27 Jul 2022. URL `https://proceedings.mlr.press/v188/sehic22a.html`.

Zhitong Xu, Haitao Wang, Jeff Phillips, and Shandian Zhe. Standard gaussian process is all you need for high-dimensional bayesian optimization. In *International Conference on Learning Representations*, volume 2025, pp. 94842–94862, 2025.

Karren Yang, Abigail Katcoff, and Caroline Uhler. Characterizing and learning equivalence classes of causal DAGs under interventions. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5541–5550. PMLR, 10–15 Jul 2018.

Rosen Yu, Cyril Picard, and Faez Ahmed. Git-bo: High-dimensional bayesian optimization with tabular foundation models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 111636–111663, 2026. URL `https://proceedings.iclr.cc/paper_files/paper/2026/file/b54bafb92da25f09ac367cff46678992-Paper-Conference.pdf`.

Jiji Zhang and Peter Spirtes. Strong faithfulness and uniform consistency in causal inference. In *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, UAI'03, pp. 632–639. Morgan Kaufmann Publishers Inc., 2002.

Miao Zhang, Huiqi Li, and Steven Su. High dimensional bayesian optimization via supervised dimension reduction. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, pp. 4292–4298. AAAI Press, 2019.

Mimi Zhang, Andrew Parnell, Dermot Brabazon, and Alessio Benavoli. Bayesian optimisation for sequential experimental design with applications in additive manufacturing, 2023. URL https://arxiv.org/abs/2107.12809.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Changbo Zhu, Xianyang Zhang, Shun Yao, and Xiaofeng Shao. Distance-based and rkhs-based dependence metrics in high dimension. *The Annals of Statistics*, 48(6):pp. 3366–3394, 2020.

A MARKOV PROPERTY AND FAITHFULNESS IN THE PRESENCE OF LATENT CONFOUNDERS

Let $V = X \cup L$ with $X = \{X_0, \dots, X_d\}$ the observed variables and $L = \{L_1, \dots, L_q\}$ latent variables. Let $\mathcal{G}_+$ be a DAG on V and suppose the data are generated by a Markovian SCM on $\mathcal{G}_+$ with jointly independent exogenous noises and no selection bias. For an intervention target set $I \subseteq \{1, \dots, d\}$, let $do(\mathbf{X}_I)$ be a surgical hard intervention on the variables $\{X_i : i \in I\}$, let $\mathcal{G}_+^{do(\mathbf{X}_I)}$ denote the corresponding mutilated DAG on V, and let $\mathcal{G}^{do(\mathbf{X}_I)}$ denote the latent-projection ADMG/MAG of $\mathcal{G}_+^{do(\mathbf{X}_I)}$ onto X (i.e. after marginalising out L). We have

$p_{\mathbf{X},\mathbf{L}}^{do(\mathbf{X}_I)}$ is Markov (via d-separation) to $\mathcal{G}_+^{do(\mathbf{X}_I)}$: By assumption, for each node $V_j \in V$ there is a structural equation $V_j = f_j\big(\mathrm{pa}_{\mathcal{G}_+}(V_j), U_j\big)$, where the exogenous noises U_j's are mutually independent, and the system of equations admits a unique solution for every noise realisation. A surgical hard intervention $do(\mathbf{X}_I)$ replaces, for each $i \in I$, the original structural equations of X_i by new equations of the form $X_i = g_i(\tilde{U}_i)$, where $\tilde{U}_i$ is a new exogenous noise, independent of all other noises. Graphically, this corresponds to deleting all incoming edges into the nodes $\{X_i : i \in I\}$, yielding a new DAG $\mathcal{G}_+^{do(\mathbf{X}_I)}$ on V. The resulting post-intervention system is again an SCM on V, with structural equations

$$V_j = \begin{cases} f_j\big(\mathrm{pa}_{\mathcal{G}_+}(V_j), U_j\big), & V_j \notin \{X_i : i \in I\}, \\ g_j(\tilde{U}_j), & V_j = X_j \text{ with } j \in I, \end{cases}$$

and exogenous noises $(U_j)_{j \notin I}$ together with $(\tilde{U}_i)_{i \in I}$, which remain jointly independent by construction. By the standard result for Markovian SCMs on DAGs (global directed Markov property), the induced interventional distribution $p_{\mathbf{X},\mathbf{L}}^{do(\mathbf{X}_I)}$ is Markov with respect to the truncated DAG $\mathcal{G}_+^{do(\mathbf{X}_I)}$, i.e. every d-separation in $\mathcal{G}_+^{do(\mathbf{X}_I)}$ corresponds to a conditional independence in $p_{\mathbf{X},\mathbf{L}}^{do(\mathbf{X}_I)}$.

$p_{\mathbf{X}}^{do(\mathbf{X}_I)}$ is Markov (via m-separation) to the latent-projection $\mathcal{G}^{do(\mathbf{X}_I)}$: The latent-projection $\mathcal{G}^{do(\mathbf{X}_I)}$ is the latent-projection ADMG/MAG obtained from $\mathcal{G}_+^{do(\mathbf{X}_I)}$ by marginalising out the latent nodes L. By the theory of ancestral graphs / MAGs (Richardson & Spirtes, 2002; Evans, 2016), there exists a mixed graph $\mathcal{G}^{do(\mathbf{X}_I)}$ on the observed node set X such that:

(i) $p_{\mathbf{X},\mathbf{L}}^{do(\mathbf{X}_I)}$ is Markov to $\mathcal{G}_+^{do(\mathbf{X}_I)}$ if and only if $p_{\mathbf{X}}^{do(\mathbf{X}_I)}$ is Markov to $\mathcal{G}^{do(\mathbf{X}_I)}$ in the sense of the global Markov property for ADMGs (with m-separation); and

(ii) the conditional independences among the observed variables X implied by m-separation in $\mathcal{G}^{do(\mathbf{X}_I)}$ coincide with exactly those implied by d-separation in $\mathcal{G}_+^{do(\mathbf{X}_I)}$ after marginalising out L.

For *post-interventional faithfulness*, we impose the following assumption: for (almost) all choices of structural parameters, the interventional distribution $p_{\mathbf{X}}^{do(\mathbf{X}_I)}$ is faithful (via m-separation) to $\mathcal{G}^{do(\mathbf{X}_I)}$ for every intervention set I. That is, all and only the conditional independences in $p_{\mathbf{X}}^{do(\mathbf{X}_I)}$ are implied by m-separation in the post-intervention ADMG/MAG $\mathcal{G}^{do(\mathbf{X}_I)}$. In parametric families such as linear-Gaussian or suitably regular exponential-family Bayesian networks, the set of parameters violating (post-)interventional faithfulness has Lebesgue measure zero; we therefore view this as a mild genericity assumption.

In terms of our particular intervening strategy (hard interventions on all $X_1, \dots, X_d$), any latent confounder L that affects both X_j and X_0 has its arrow into X_j cut by the hard intervention, and hence the back-door path $X_j \leftarrow L \rightarrow X_0$ disappears in the post-intervention graph. The only remaining open paths from X_j to X_0 are directed paths starting at X_j; these exist if and only if X_j is an ancestor of X_0 in the mutilated DAG. Under post-interventional faithfulness as above, this implies: (1) If there is no directed path $X_j \rightarrow \cdots \rightarrow X_0$ in the post-intervention graph $\mathcal{G}^{do(\mathbf{X}_I)}$, then $X_j \perp\!\!\!\perp X_0$ in the interventional distribution $p_{\mathbf{X}}^{do(\mathbf{X}_I)}$. (2) If there is such a directed path, then generically X_j and X_0 remain dependent. Therefore, with hidden confounders, the post-interventional faithfulness assumption is still justified, and Property 1 continues to hold.

B PROOF OF PROPERTIES

Proof of Property 1 By the definition of a hard (surgical) intervention on $\{X_j\}_{j=1}^d$, all incoming edges into these nodes are removed, while outgoing edges are kept. Hence, in the post-intervention graph $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$: (1) every X_j with $j \geq 1$ has no parents; (2) there are no edges between variables in $\{X_j\}_{j=1}^d$ because any such edge would be incoming into an intervened node, and would be removed by graph surgery; (3) the only possible remaining edges incident to X_0 from $\{X_j\}_{j=1}^d$ are of the form $X_j \to X_0$, and such an edge is present in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$ if and only if it was present in $\mathcal{G}$.

Any path from X_j to X_0 in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$ must have length 1, because any longer path would involve an intermediate node in $\{X_j\}_{j=1}^d$, and the edge entering that node would have been deleted by the intervention. Therefore, if $X_j \to X_0$ is an edge in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$, there is exactly one path $X_j \to X_0$; if there is no edge $X_j \to X_0$ in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$, then there is no path at all between X_j and X_0. Therefore, X_j and X_0 are d-connected in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$ if and only if $X_j \to X_0$ is an edge in $\mathcal{G}$.

By post-interventional faithfulness of $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{w})}(\mathbf{X})$ to $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$, $X_j \perp\!\!\!\perp X_0$ under $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{w})}(\mathbf{X})$ if and only if X_j and X_0 are d-separated in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$, and hence $X_j \not\perp\!\!\!\perp X_0$ under $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{w})}(\mathbf{X})$ if and only if X_j and X_0 are d-connected in $\mathcal{G}^{do(\mathbf{X}^0=\mathbf{w})}$.

Combining the above observations, we conclude that $X_j \not\perp\!\!\!\perp X_0$ under $p_{\mathbf{X}}^{do(\mathbf{X}^0=\mathbf{w})}(\mathbf{X})$ if and only if $X_j \to X_0$ is an edge in $\mathcal{G}$.

Remark In Algorithm 1, the variables W_j need be mutually independent. The reason is that a stochastic intervention $do(\mathbf{X}^0 = \mathbf{W})$ does not only remove incoming edges into $\{X_1, ..., X_d\}$. It also injects the joint distribution of $\mathbf{W}$ into the intervened variables. If the components of $\mathbf{W}$ are dependent, then the intervened variables $\{X_1, ..., X_d\}$ can remain statistically dependent even though all causal edges among them have been surgically removed. Then, marginal dependence between X_j and X_0 may arise through correlation in the intervention distribution, not through a causal edge $X_j \to X_0$.

Proof of Property 2 Eq. (6) follows directly from Theorem 5 in (Székely et al., 2007). Note that, each squared sample distance covariance $\mathrm{dCov}_m^2(X_j^b, X_0^b)$ is computed from an independent batch of samples, and the joint interventional distribution $p_{\mathbf{X}}^{do(\mathbf{X}^0)}$ changes across batches. Therefore, the statistics $\{\mathrm{dCov}_m^2(X_j^b, X_0^b)\}_{b=1}^t$ are mutually independent. Because the summation operator is a continuous map, we obtain, by the continuous mapping theorem,

$$\lim_{m\to\infty} m\, \overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]}) \overset{d}{=} \sum_{b=1}^{t} \|\zeta^b\|^2.$$

On the other hand, by Corollary 2 in (Székely et al., 2007), if X_j and X_0 are dependent, then $m\, \overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ diverges to infinity as $m \to \infty$.

Proof of Property 3 Let $\mathbf{X}$ and $\mathbf{Y}$ be two t-dimensional random vectors. The fact that, for fixed m, the power of independence tests based on $\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y})$ drops with increasing t has been reported by both Ramdas et al. (2015) and Zhu et al. (2020).

Ramdas et al. (2015) proved that the population distance covariance $\mathrm{dCov}(\mathbf{X}, \mathbf{Y})$ shrinks to zero polynomially in the dimension t. Theoretically, the loss of power of distance-covariance-based tests in high dimensions arises in part because the kernel moment terms in the Hoeffding decomposition scale with the dimension, causing the population distance covariance (the signal) to decay polynomially in t while the estimator's variance remains of order $1/m$. Zhu et al. (2020) proved that in high dimension, the sample squared distance covariance $\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y})$ effectively reduces to the sum of squared linear covariances between components: $\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y}) \approx c_1 \sum_{i,k} \mathrm{Cov}_m^2(X_i, Y_k)$, where c_1 is a constant. Hence, in high dimensions, $\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y})$ detects mainly linear dependence, not

general nonlinear dependence. Both Ramdas et al. (2015) and Zhu et al. (2020) demonstrated via simulation that the power of tests based on $\mathrm{dCov}_m^2(\mathbf{X}, \mathbf{Y})$ decreases dramatically with increasing dimension.

Zhu et al. (2020) further showed that the marginally aggregated sample squared distance covariance $c_2 \sum_{i,k} \mathrm{dCov}_m^2(X_i, Y_k)$ maintains high power with increasing dimension, where c_2 is another constant. Note that when $X_i \perp\!\!\!\perp Y_k$ for any $i \neq k$, we have $c_2 \sum_{i,k} \mathrm{dCov}_m^2(X_i, Y_k) = c_2 \sum_{i=1}^t \mathrm{dCov}_m^2(X_i, Y_i)$, and hence the marginally aggregated sample squared distance covariance coincides (up to a normalizing constant) with our coordinate-wise distance covariance $\overline{\mathrm{dCov}}_m^2(\mathbf{X}, \mathbf{Y})$. Therefore, we here claim that our coordinate-wise distance covariance $\overline{\mathrm{dCov}}_m^2(\mathbf{X}, \mathbf{Y})$ maintains high power with increasing dimension.

C Proof of Theorem 1

Recall that the structural equation is $X_0 = f_0(\mathbf{X}_\pi, U_0)$. We might assume $X_1 \in \mathbf{X}_\pi$, and for ease of explanation, write

$$X_0 = f_0(X_1, X_2, U_0);$$

that is, X_2 collects all the remaining parental influence on X_0.

Regularity on the joint distribution of (X_1, X_2, U_0) and on f_0 to have λ-strong marginal faithfulness: Recall that X_1^b, X_2^b and X_0^b are the random variables associated with the b-th batch such that $\{w_{1i}^b\}_{i=1}^m \overset{d}{=} X_1^b$, $\{w_{2i}^b\}_{i=1}^m \overset{d}{=} X_2^b$, and $\{\chi_i^b\}_{i=1}^m \overset{d}{=} X_0^b$, where $\overset{d}{=}$ denotes equality in distribution. The strength of dependence between X_1^b and $X_0^b = f_0(X_1^b, X_2^b, U_0)$, as measured by the population distance covariance $\mathrm{dCov}(X_1^b, X_0^b)$, is determined by the combination of f_0 and the joint distribution of the inputs (X_1^b, X_2^b, U_0). (More accurately, the strength of dependence is determined by f_0, the joint distribution of (X_1^b, X_2^b), and the distribution of U_0, because $U_0 \perp\!\!\!\perp X_1^b$ and $U_0 \perp\!\!\!\perp X_2^b$.) For a fixed f_0, different experimental designs (i.e., different distributions of X_1 and X_2 across batches) can induce widely varying dependence strengths – including arbitrarily small ones – even when X_1 is a direct causal parent of X_0.

To guarantee a non-trivial dependence signal in at least one of the batches, we therefore impose regularity conditions on both the structural function and the design distributions:

> Among the t batches, there exists at least one batch for which X_1 has a non-degenerate effect on X_0 over the sampled region. Mathematically, we assume that f_0 is continuously differentiable, and the partial derivative is bounded away from zero on a sampling region $\mathcal{R}$:
>
> $$\left| \frac{\partial f_0}{\partial x_1}(x_1, x_2, u_0) \right| \geq c_1 > 0 \quad \text{for all } (x_1, x_2) \in \mathcal{R} \text{ and } u_0 \in S \subset D(U_0),$$
>
> and that the design distributions place non-negligible mass on $\mathcal{R}$: $\Pr((X_1^b, X_2^b) \in \mathcal{R}) \times \Pr(u_0 \in S) \geq p_1 > 0$.

Here, S is a subset of the domain of U_0. Let b^* denote the batch with the regularized design. Then, conditional on $(X_2^{b^*}, U_0)$, landing in the projection of $\mathcal{R}$, the map $x_1 \mapsto x_0 = f_0(x_1, X_2^{b^*}, U_0)$ is strictly monotone with slope bounded away from zero on an interval that the design visits with a positive probability p_1.

Consider the class $\mathcal{F}$ of all joint distributions of $(X_1^{b^*}, X_2^{b^*}, U_0)$ that satisfy: (1) the derivative bound $|\frac{\partial f_0}{\partial x_1}| \geq c_1$ on $\mathcal{R} \times S$, (2) the "non-negligible mass" condition $\Pr((X_1^{b^*}, X_2^{b^*}, U_0) \in \mathcal{R} \times S) \geq p_1$, and (3) finite first moments for all X_0^b, X_1^b, X_2^b. Under these conditions, $\mathcal{F}$ is tight and closed under weak convergence, hence compact in the weak topology. The mapping $P \in \mathcal{F} \mapsto \mathrm{dCov}(X_1^{b^*}, X_0^{b^*})$ is a continuous functional of the joint law P (distance covariance is defined via integrals of characteristic functions or expectations of bounded kernels, both continuous in the weak topology under moment bounds). Because none of the distributions in $\mathcal{F}$ has $X_1^{b^*}$ and $X_0^{b^*}$ independent, for

all $P \in \mathcal{F}$, $\mathrm{dCov}(X_1^{b^*}, X_0^{b^*}) > 0$. Let $\lambda = \inf_{P \in \mathcal{F}} \mathrm{dCov}(X_1^{b^*}, X_0^{b^*})$. Since $\mathcal{F}$ is compact and dCov is continuous, the extreme value theorem guarantees that the infimum is achieved: there exists $P^* \in \mathcal{F}$ with $\lambda = \mathrm{dCov}(X_1^{b^*}, X_0^{b^*})|_{P^*}$. Because every $P \in \mathcal{F}$ gives strictly positive distance covariance, so in particular $\lambda > 0$. We hence conclude that there exists a constant $\lambda > 0$ such that $\mathrm{dCov}(X_1^{b^*}, X_0^{b^*}) \geq \lambda > 0$.

In line with Zhang & Spirtes (2002), we here interpret the inequality $\mathrm{dCov}(X_1^{b^*}, X_0^{b^*}) \geq \lambda > 0$ as λ-strong *marginal* faithfulness that the statistical dependency between variables that should be dependent (d-connected) is bounded away from zero, not just close to zero.

Non-asymptotic sample complexity. Fix a batch b, and let $\Delta := \mathrm{dCov}^2(X_1^b, X_0^b)$ denote the squared *population* distance covariance. We work with the unbiased estimator of Δ introduced by Székely & Rizzo (2014). By Székely & Rizzo (2013) and Székely & Rizzo (2014), the statistic $T_m := \mathrm{dCov}_m^2(X_1^b, X_0^b)$ is a U-statistic with symmetric kernel

$$\kappa(Z_i, Z_k) = \Lambda_{ik}(X_1^b)\,\Lambda_{ik}(X_0^b), \qquad Z_i := (w_{1i}^b, \chi_i^b),$$

where $\Lambda_{ik}(\cdot)$ denotes the doubly centered distance terms.

The sampling distributions $\{p_{W_j}\}_{j=1}^d$ (for the stochastic interventions) are Gaussian on bounded intervals, and hence are sub-Gaussian. The noise U_0 is Gaussian and therefore sub-Gaussian as well. If f_0 is Lipschitz, then $X_0 = f_0(\mathbf{X}_\pi, U_0)$ is also sub-Gaussian (as a Lipschitz transform of sub-Gaussian inputs).

Distances and their centered versions $\Lambda_{ik}(X_1^b)$ and $\Lambda_{ik}(X_0^b)$ are Lipschitz functions and linear combinations of sub-Gaussian variables, and are therefore sub-Gaussian. Consequently, their product $\kappa(Z_i, Z_k) = \Lambda_{ik}(X_1^b)\,\Lambda_{ik}(X_0^b)$ is (at least) sub-exponential. Let K denote any upper bound on the sub-exponential norm (Orlicz ψ_1-norm) of κ:

$$K \geq \|\kappa\|_{\psi_1} := \inf\{c > 0 : \mathbb{E}[\exp(|\kappa|/c)] \leq 2\}.$$

The bound $\|\kappa\|_{\psi_1} \leq K$ implies a tail inequality of the form $\Pr(|\kappa| > t) \leq 2\exp(-ct/K)$ for some universal constant $c > 0$.

Utilizing the Bernstein-type concentration inequality for U-statistics with sub-exponential kernels, there exists a universal constant $c > 0$ such that, for all $\varepsilon > 0$,

$$\Pr\left(|T_m - \Delta| > \varepsilon\right) \;\leq\; 2\exp\left(-c\,m\,\min\left\{\frac{\varepsilon}{K}, \frac{\varepsilon^2}{K^2}\right\}\right). \tag{8}$$

(If κ is in fact sub-Gaussian, then one obtains the simpler bound $\Pr(|T_m - \Delta| > \varepsilon) \leq 2\exp(-cm\varepsilon^2/K^2)$, where K is an upper bound on the ψ_2-norm of κ.)

We now consider the hypothesis test

$$H_0 : \Delta = 0 \qquad \text{vs.} \qquad H_1 : \Delta \geq \lambda > 0,$$

corresponding to a strong marginal faithfulness condition. Given a threshold $\theta > 0$, we use the test that rejects H_0 whenever $T_m \geq \theta$. Under H_0, we have $\Delta = 0$, and the type I error is

$$\Pr(T_m \geq \theta \mid H_0) = \Pr(|T_m - 0| \geq \theta \mid H_0) \;\leq\; 2\exp\left(-c\,m\,\min\left\{\frac{\theta}{K}, \frac{\theta^2}{K^2}\right\}\right).$$

Under H_1, we have $\Delta \geq \lambda$, and we fail to reject H_0 if $T_m \leq \theta$. The type II error is therefore

$$\Pr(T_m \leq \theta \mid H_1) \;\leq\; \Pr\left(|T_m - \Delta| \geq \Delta - \theta \mid H_1\right) \;\leq\; \Pr\left(|T_m - \Delta| \geq \lambda - \theta\right),$$

and by Eq. (8),

$$\Pr(T_m \leq \theta \mid H_1) \;\leq\; 2\exp\left(-c\,m\,\min\left\{\frac{\lambda - \theta}{K}, \frac{(\lambda-\theta)^2}{K^2}\right\}\right).$$

Define

$$\gamma_1 := \min\left\{\frac{\theta}{K}, \frac{\theta^2}{K^2}\right\}, \qquad \gamma_2 := \min\left\{\frac{\lambda - \theta}{K}, \frac{(\lambda-\theta)^2}{K^2}\right\}.$$

To ensure that both error probabilities are bounded by a prescribed significance level $\alpha \in (0,1)$, it suffices to choose m such that

$$2\exp(-cm\gamma_1) \le \alpha \quad \text{and} \quad 2\exp(-cm\gamma_2) \le \alpha.$$

Equivalently,

$$m \;\ge\; \frac{1}{c\,\gamma_1}\log(\frac{2}{\alpha}) \quad \text{and} \quad m \;\ge\; \frac{1}{c\,\gamma_2}\log(\frac{2}{\alpha}),$$

so that a sufficient non-asymptotic sample size is

$$m \;=\; \mathcal{O}\left(\frac{1}{c\,\min\{\gamma_1,\gamma_2\}}\,\log(\frac{1}{\alpha})\right).$$

A natural and convenient choice of threshold is $\theta = \lambda/2$, which balances the deviations under H_0 and H_1, i.e., $\lambda - \theta = \theta = \lambda/2$. In that case $\gamma_1 = \gamma_2$, and in the regime $\lambda \lesssim K$ the quadratic term dominates, yielding the simplified scaling

$$m = \mathcal{O}\left(\frac{K^2}{\lambda^2}\,\log(\frac{1}{\alpha})\right),$$

which makes explicit the dependence of the sample complexity on the signal strength λ, the sub-exponential scale K, and the significance level α.

D STOCHASTIC INTERVENTION DISTRIBUTIONS

The stochastic intervention distribution p_{W_j} for X_j can be either a Gaussian distribution, $\mathcal{N}(u_j, \sigma_j^2)$, or a uniform distribution, $\mathcal{U}(u_j - 0.5h_j, u_j + 0.5h_j)$. Both distributions are centred at u_j as determined in Blocks 2 and 3 of the StoBO algorithm. We next explain how the scale parameters σ_j and h_j are determined from the data. Note that if the centred interval $[u_j - 0.5h_j, u_j + 0.5h_j]$ extends beyond the domain $D(X_j)$, we shift it by the minimum amount required to lie entirely within the domain, while keeping its width h_j unchanged. For example, the interval may be shifted to $[u_j - 0.2h_j, u_j + 0.8h_j]$ when this interval is contained in $D(X_j)$.

Let X_j^1 and X_j^2 be two independent draws from the intervention distribution for X_j. If X_j^1 and X_j^2 follow a Gaussian distribution with standard deviation σ_j, then

$$\Pr\left(\left|X_j^1 - X_j^2\right| \le \frac{\hat{\ell}_j}{2}\right) = 2\Phi\left(\frac{\hat{\ell}_j/2}{\sqrt{2}\sigma_j}\right) - 1,$$

where $\hat{\ell}_j$ is the estimated length scale for X_j in the GP model. Alternatively, if X_j^1 and X_j^2 are independently drawn from a uniform distribution over an interval of length h_j, where $h_j \ge \hat{\ell}_j/2$, then

$$\Pr\left(\left|X_j^1 - X_j^2\right| \le \frac{\hat{\ell}_j}{2}\right) = 1 - \left(1 - \frac{\hat{\ell}_j/2}{h_j}\right)^2.$$

Let $\rho = \Pr(|X_j^1 - X_j^2| \le \frac{\hat{\ell}_j}{2})$ denote the probability that the distance between two random points is shorter than $\hat{\ell}_j/2$. Note that we evaluate the probability that the distance between two random samples is shorter than half of the estimated length scale, rather than the full estimated length scale. This more conservative threshold is used because, when only a limited number of observations are available, the estimated length scale can be substantially larger than the true length scale. Solving the two probability expressions gives

$$\sigma_j = \frac{\hat{\ell}_j/2}{\sqrt{2}\Phi^{-1}((1+\rho)/2)},$$

for the Gaussian distribution, and

$$h_j = \frac{\hat{\ell}_j/2}{1 - \sqrt{1-\rho}},$$

Table 3: Relationship between the distribution parameters σ_j and h_j and the estimated length scale $\hat{\ell}_j$ for different values of $\rho = \Pr(|X_j^1 - X_j^2| \leq \hat{\ell}_j/2)$.

ρ	Gaussian (σ_j)	Uniform (h_j)
30%	$0.918 \times \hat{\ell}_j$	$3.061 \times \hat{\ell}_j$
40%	$0.674 \times \hat{\ell}_j$	$2.218 \times \hat{\ell}_j$
50%	$0.524 \times \hat{\ell}_j$	$1.707 \times \hat{\ell}_j$
60%	$0.420 \times \hat{\ell}_j$	$1.361 \times \hat{\ell}_j$
70%	$0.341 \times \hat{\ell}_j$	$1.106 \times \hat{\ell}_j$

for the uniform distribution. Table 3 provides the specific values of σ_j and h_j for different values of ρ.

In our Python package, when Gaussian sampling is the intervention distribution, the default value of ρ is 0.80 for estimated parent variables ($j \in \hat{\pi}$) and 0.65 for estimated non-parent variables ($j \notin \hat{\pi}$). These values are selected primarily to support accurate parent discovery rather than global minimization or maximization. Nevertheless, the sensitivity study in Appendix F.2, which examines the effect of $\rho = \Pr(|X_j^1 - X_j^2| \leq \hat{\ell}_j/2)$ for $j \in \hat{\pi}$, shows that $\rho = 0.80$ also provides the best overall optimization performance under the tested settings. Once the estimated parent set remains unchanged for two consecutive iterations, the algorithm terminates the parent-discovery stage and switches to conventional sequential BO for global optimization.

E Implementation details of Algorithm 1

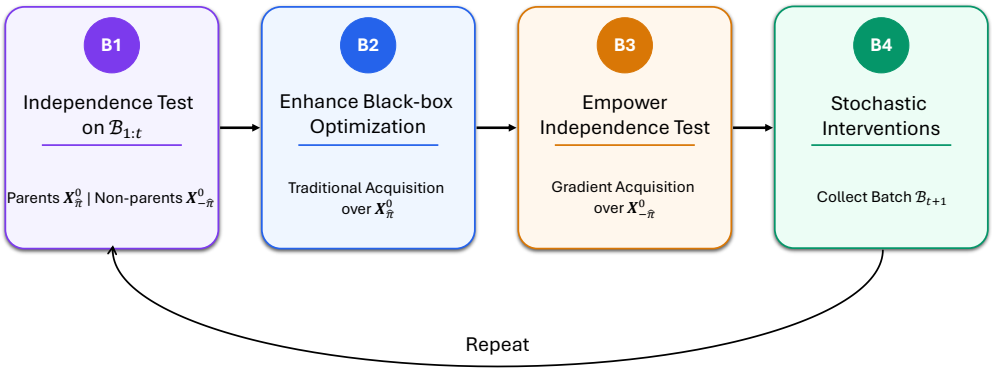


Figure 2: Overview of the StoBO algorithm. Marginal independence tests partition the variables into estimated parents $\mathbf{X}^0_{\hat{\pi}}$ and non-parents $\mathbf{X}^0_{-\hat{\pi}}$ (B1). A conventional acquisition function selects a local batch over the estimated parents to optimize the black-box objective (B2), while a gradient-guided acquisition function selects a local batch over the estimated non-parents to strengthen subsequent independence tests (B3). The selected batches are implemented as stochastic interventions (B4), and the procedure is repeated.

We here explain the optimization problem in Block 3 of Algorithm 1, drawing on established results from the literature.

Gaussian process regression The GP is characterized by a mean function $\mu(\mathbf{x})$ and a kernel function $k(\mathbf{x}, \mathbf{x}')$: $g(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$. For notational convenience, we henceforth use Y, rather than X_0, to denote the response variable. Suppose we have a training set $\mathcal{B} = \mathcal{B}_{1:t}$ comprising $n = tm$ observations, denoted by $\mathcal{B} = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n\}$. In general, the function evaluation $g(\mathbf{x})$ could be subject to random noise, and the output at the input $\mathbf{x}_i$ is $y_i = g(\mathbf{x}_i) + \epsilon(\mathbf{x}_i)$, where the noisy term $\epsilon(\mathbf{x})$ is usually assumed to be normally distributed with mean 0 and variance σ_ϵ^2.

We write $\mathbf{x}_{1:n}$ for the collection of n inputs and $y_{1:n}$ for the column vector of observations, and adopt compact notation for functions applied to collections of points:

$$g(\mathbf{x}_{1:n}) = [g(\mathbf{x}_1), \ldots, g(\mathbf{x}_n)]^\top, \quad \mu(\mathbf{x}_{1:n}) = [\mu(\mathbf{x}_1), \ldots, \mu(\mathbf{x}_n)]^\top,$$

with $k(\mathbf{x}_{1:n}, \mathbf{x}_{1:n})$ denoting the $n \times n$ covariance matrix whose (i, j)-th entry is $k(\mathbf{x}_i, \mathbf{x}_j)$. Let $\{\hat{\mu}(\cdot), \hat{k}(\cdot, \cdot), \hat{\sigma}_\epsilon\}$ denote the estimates obtained from the training data $\mathcal{B}$.

By the standard conditioning rule for Gaussian vectors, the predictive distribution at a new input $\mathbf{x}$ is $[g(\mathbf{x})|\mathcal{B}, \mathbf{x}] \sim \mathcal{N}(\mu_n(\mathbf{x}), \sigma_n^2(\mathbf{x}))$, where

$$\mu_n(\mathbf{x}) = \hat{\mu}(\mathbf{x}) + \hat{k}(\mathbf{x}, \mathbf{x}_{1:n})[\hat{k}(\mathbf{x}_{1:n}, \mathbf{x}_{1:n}) + \hat{\sigma}_\epsilon^2 \mathbf{I}]^{-1}(y_{1:n} - \hat{\mu}(\mathbf{x}_{1:n})), \tag{9}$$

and

$$\sigma_n^2(\mathbf{x}) = \hat{k}(\mathbf{x}, \mathbf{x}) - \hat{k}(\mathbf{x}, \mathbf{x}_{1:n})[\hat{k}(\mathbf{x}_{1:n}, \mathbf{x}_{1:n}) + \hat{\sigma}_\epsilon^2 \mathbf{I}]^{-1}\hat{k}(\mathbf{x}_{1:n}, \mathbf{x}). \tag{10}$$

Here, $\hat{k}(\mathbf{x}_{1:n}, \mathbf{x}) = [\hat{k}(\mathbf{x}_1, \mathbf{x}), \ldots, \hat{k}(\mathbf{x}_n, \mathbf{x})]^\top$ is a column vector, and $\hat{k}(\mathbf{x}, \mathbf{x}_{1:n}) = \hat{k}(\mathbf{x}_{1:n}, \mathbf{x})^\top$.

Gradient of the posterior mean Let $G(\mathbf{x}) = \nabla_{\mathbf{x}} \hat{k}(\mathbf{x}, \mathbf{x}_{1:n})$ denote the $d \times n$ matrix whose i-th column is $\nabla_{\mathbf{x}} \hat{k}(\mathbf{x}, \mathbf{x}_i)$. Differentiating (9) with respect to $\mathbf{x}$ gives the gradient of the posterior mean function

$$\nabla_{\mathbf{x}} \mu_n(\mathbf{x}) = \nabla_{\mathbf{x}} \hat{\mu}(\mathbf{x}) + G(\mathbf{x})[\hat{k}(\mathbf{x}_{1:n}, \mathbf{x}_{1:n}) + \hat{\sigma}_\epsilon^2 \mathbf{I}]^{-1}(y_{1:n} - \hat{\mu}(\mathbf{x}_{1:n})).$$

The proof of Theorem 1 shows that the independence test has the greatest statistical power in regions where the gradient magnitude of g is large. Accordingly, Block 4 selects local regions with large gradient magnitudes while accounting for predictive uncertainty, in a manner analogous to the acquisition function.

Posterior distribution of the gradient Provided the kernel admits the required mixed second-order derivatives, the gradient of the latent function is itself jointly Gaussian a posteriori:

$$[\nabla_{\mathbf{x}} g(\mathbf{x})|\mathcal{B}] \sim \mathcal{N}(\nabla_{\mathbf{x}} \mu_n(\mathbf{x}), \Sigma_n(\mathbf{x})),$$

where

$$\Sigma_n(\mathbf{x}) = \nabla_{\mathbf{x}} \nabla_{\mathbf{x}'}^\top \hat{k}(\mathbf{x}, \mathbf{x}')|_{\mathbf{x}'=\mathbf{x}} - G(\mathbf{x})[\hat{k}(\mathbf{x}_{1:n}, \mathbf{x}_{1:n}) + \hat{\sigma}_\epsilon^2 \mathbf{I}]^{-1} G(\mathbf{x})^\top.$$

Let $\mathbf{G} = (G_1, \ldots, G_d)^\top := [\nabla_{\mathbf{x}} g(\mathbf{x})|\mathcal{B}]$ denote the gradient variable. Since every component of a multivariate normal vector is itself univariate normal, with mean and variance given by the corresponding entries of the mean vector and covariance matrix, we have

$$G_i \sim \mathcal{N}(\hat{u}_i(\mathbf{x}), \hat{\sigma}_i^2(\mathbf{x})), \;\; \hat{u}_i(\mathbf{x}) = \frac{\partial \mu_n(\mathbf{x})}{\partial x_i}, \;\; \hat{\sigma}_i^2(\mathbf{x}) = [\Sigma_n(\mathbf{x})]_{ii}, \;\; i = 1, \ldots, d.$$

Then the distibution of $|G_i|$ is the folded normal distribution: $|G_i| \sim \text{FoldedNormal}(\hat{u}_i(\mathbf{x}), \hat{\sigma}_i^2(\mathbf{x}))$, with expectation given by the standard folded-normal mean formula:

$$\mathbb{E}[|G_i|] = \hat{\sigma}_i(\mathbf{x})\sqrt{\frac{2}{\pi}} \exp(-\frac{\hat{u}_i^2(\mathbf{x})}{2\hat{\sigma}_i^2(\mathbf{x})}) + \hat{u}_i(\mathbf{x}) \left[1 - 2\Phi(-\frac{\hat{u}_i(\mathbf{x})}{\hat{\sigma}_i(\mathbf{x})})\right]. \tag{11}$$

where $\Phi(\cdot)$ is the standard normal CDF. Equation (11) provides a closed-form criterion that jointly reflects the posterior gradient mean $\hat{u}_i(\mathbf{x})$ and posterior gradient uncertainty $\hat{\sigma}_i(\mathbf{x})$, and is used directly in 4 to identify regions of X_i where the independence test is expected to have high power.

F Supplementary experimental details and results

F.1 Details on the benchmark functions

The test functions below are all non-convex and are selected from the online resource (Surjanovic & Bingham), which provides a range of benchmark functions for comparing optimization algorithms. In terms of generic global black-box optimization, Michalewicz, Hartmann6, Ackley, and Rastrigin are expected to be among the most challenging because of their narrow or numerous local basins. Levy and Griewank represent intermediate multimodal difficulty, whereas Rosenbrock,

Dixon-Price, and Powell primarily test non-separability and challenging local geometry. Styblinski-Tang is generally the least difficult because of its smooth and separable structure. Accordingly, we list the functions in decreasing order of expected optimization difficulty to assess whether StoBO's performance varies with problem difficulty. For functions with a global minimum value of zero, we report the absolute optimality gap, $|f(\hat{\mathbf{x}}^*) - f(\mathbf{x}^*)| = |f(\hat{\mathbf{x}}^*)|$. For all other functions, we report the relative optimality gap, $|\frac{f(\hat{\mathbf{x}}^*)-f(\mathbf{x}^*)}{f(\hat{\mathbf{x}}^*)}|$.

Michalewicz The Michalewicz function is a difficult multimodal benchmark with $d!$ local optima. Its landscape becomes increasingly deceptive as the dimensionality grows, making it useful for evaluating whether a method can perform reliable global search under severe local trapping. Define

$$f(\mathbf{x}) = -\sum_{i=1}^{d} \sin(x_i)\, \sin^{2m}\left(\frac{ix_i^2}{\pi}\right).$$

The parameter m defines the steepness of the valleys and ridges. Here we set $m = 10$, the recommended value. The function domain is $[0, \pi]$ for each X_j. For $d = 5$, the global minimum value is $f(\mathbf{x}^*) = -4.6877$ at $\mathbf{x}^* = (2.2029, 1.5708, 1.2850, 1.9231, 1.7205)^\top$. For $d = 10$, the global minimum value is $f(\mathbf{x}^*) = -9.66015$ at

$$\mathbf{x}^* = (2.2029, 1.5708, 1.2850, 1.9231, 1.7205, 1.5708, 1.4544, 1.7561, 1.6557, 1.5708)^\top.$$

Hartmann6 The Hartmann6 function has 6 local minima. It is widely used in Bayesian optimization because it is low-dimensional enough to be interpretable, while still being sufficiently nontrivial and highly non-convex. This mode is used only when $d = 6$. Define

$$f(\mathbf{x}) = -\sum_{i=1}^{4} \alpha_i \exp\left(-\sum_{j=1}^{6} A_{ij}(x_j - P_{ij})^2\right),$$

where

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

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

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

The function domain is $(0, 1)$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = -3.3224$ at

$$\mathbf{x}^* = (0.2017, 0.1500, 0.4769, 0.2753, 0.3117, 0.6573)^\top.$$

Ackley The Ackley function is characterized by a nearly flat outer region, and a large hole at the centre. It is widely used to evaluate whether a method can balance global exploration and local exploitation. Define

$$f(\mathbf{x}) = -a \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d} \cos(cx_i)\right) + a + \exp(1),$$

where $a = 20$, $b = 0.2$ and $c = 2\pi$. The function domain is $[-32.768, 32.768]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $\mathbf{x}^* = \mathbf{0}$.

Rastrigin The Rastrigin function is a highly multimodal benchmark with a large number of regularly distributed local optima. Define

$$f(\mathbf{x}) = 10d + \sum_{i=1}^{d} \left(x_i^2 - 10\cos(2\pi x_i)\right).$$

The function domain is $[-5.12, 5.12]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $\mathbf{x}^* = \mathbf{0}$.

Levy The Levy function is a multimodal non-convex benchmark with oscillatory structure and multiple local optima. Define

$$f(\mathbf{x}) = \sin^2(\pi w_1) + \sum_{i=1}^{d-1} (w_i - 1)^2 \left[1 + 10\sin^2(\pi w_i + 1)\right] + (w_d - 1)^2 \left[1 + \sin^2(2\pi w_d)\right],$$

where

$$w_i = 1 + \frac{x_i - 1}{4}, \qquad i = 1, \ldots, d.$$

The function domain is $[-10, 10]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $\mathbf{x}^* = \mathbf{1}$.

Griewank The Griewank function has many widespread local minima, which are regularly distributed. Define

$$f(\mathbf{x}) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1.$$

The function domain is $[-600, 600]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $\mathbf{x}^* = \mathbf{0}$.

Rosenbrock The Rosenbrock function is a standard benchmark with strong interactions among variables and a narrow curved valley leading to the global optimum. It is a popular test problem for gradient-based optimization algorithms. Define

$$f(\mathbf{x}) = \sum_{i=1}^{d-1} \left[100\big(x_{i+1} - x_i^2\big)^2 + (x_i - 1)^2\right].$$

The function domain is $[-5, 10]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $\mathbf{x}^* = \mathbf{1}$.

Dixon-Price The Dixon-Price function is a non-separable benchmark with recursive dependence across dimensions. It is often used to test whether a method can handle strong variable interactions and chain-like coupling effects in high-dimensional optimization. Define

$$f(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^{d} i\left(2x_i^2 - x_{i-1}\right)^2.$$

The function domain is $[-10, 10]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $x_j^* = 2^{-\frac{2^j - 2}{2^j}}$, for $j = 1, \ldots, d$.

Powell The Powell function is a block-structured non-separable benchmark with strong variable interactions and high-order terms. It is useful for testing whether a method can deal with coupled variables and challenging local geometry. Here, the dimension d is a multiple of 4. Define

$$f(\mathbf{x}) = \sum_{i=1}^{d/4} \Big[(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4\Big].$$

The function domain is $[-4, 5]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = 0$ at $\mathbf{x}^* = \mathbf{0}$.

Styblinski-Tang Compared with the above nine test functions, the Styblinski-Tang function is relatively smoother. Although it is less rugged, it still contains non-trivial curvature and is useful for testing whether a method remains effective beyond highly irregular landscapes. Define

$$f(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^{d} \left(x_i^4 - 16x_i^2 + 5x_i\right).$$

The function domain is $[-5, 5]$ for each X_j. The global minimum value is $f(\mathbf{x}^*) = -d \times 39.1660$ at $x_j^* = -2.9035$, for $j = 1, \ldots, d$.

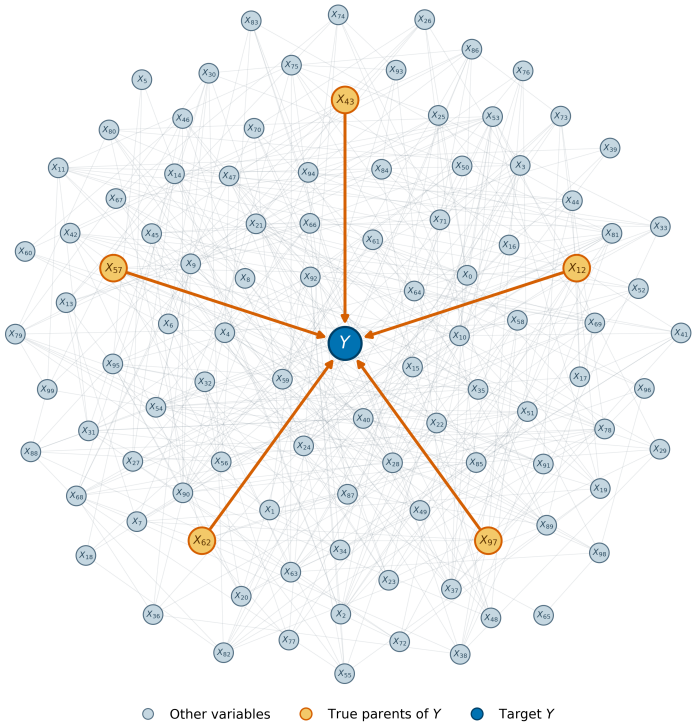
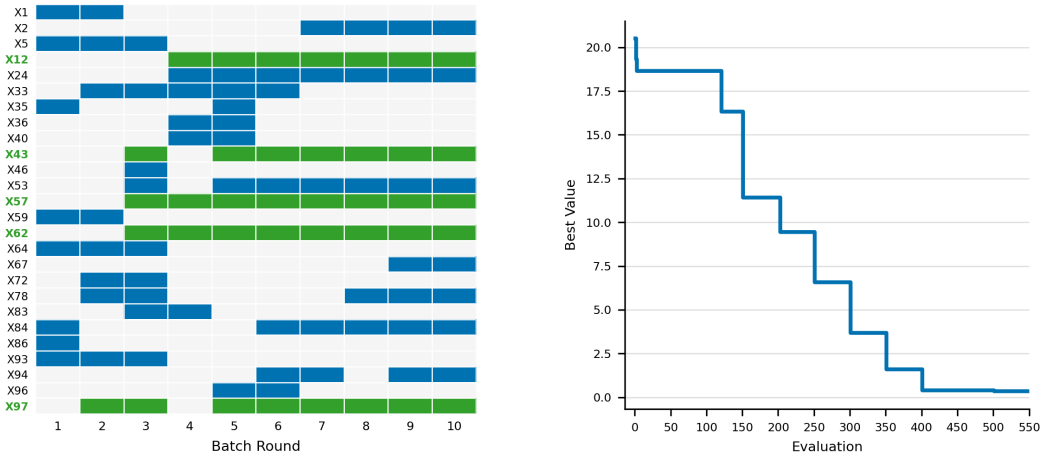


Figure 3: Example of a randomly generated DAG with 101 nodes. The direct parents of the response variable Y are selected at random from the input variables, after which edges among the input variables are generated subject to the acyclicity constraint following the Erdös–Rényi style. The structural functions governing dependencies among the input variables are specified by randomly parameterized multilayer perceptrons. The problem generator supports both observational and interventional sampling for causal discovery tasks.



(a) Evolution of the Estimated Parent Set

(b) Evolution of the Optimal Value

Figure 4: Evolution of the parent set $\hat{\pi}_t$ (left) and minimal value of the parent-variable GP surrogate (right) for a representative Ackley-function run. Green variables $\{X_{12}, X_{43}, X_{57}, X_{62}, X_{97}\}$ are true parents, and blue variables are false positives. True parents enter and leave the estimated parent set $\hat{\pi}_t$ less frequently than non-parent variables.

F.2 SENSITIVITY AND ABLATION STUDIES

The effects of the dimension d. We vary d in $\{50, 100, 150, 200\}$, corresponding to $\{50, 100, 150, 200\}$ input features, and report the mean numbers of missed parent variables (FN) and total selected variables (TP + FP) after 750 function evaluations (i.e., $t = 15$). All other parameters are kept at their default values. In particular, the number of true parents is fixed at five, except for Hartmann6 and Powell, which have six and four true parents, respectively. Each configuration is evaluated over 20 random seeds. Figure 5 shows how the two parent-selection metrics vary with the

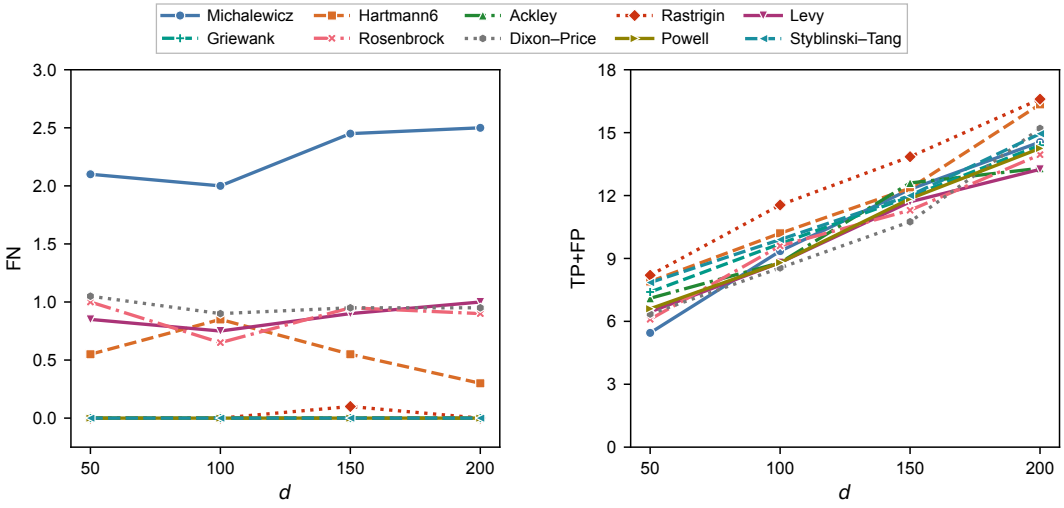


Figure 5: Effect of input dimensionality on parent discovery performance across ten test functions. Left: mean number of missed parent variables (FN). Right: mean total number of selected variables (TP + FP). The horizontal axis shows the number of input features.

number of input features across the ten test functions. The left panel reports the mean FN, while the right panel reports the mean TP + FP. The left panel shows that the mean FN does not increase with the number of input features. The mean FN remains broadly stable as the number of input features increases, although modest function-specific fluctuations are observed. This suggests that the difficulty of identifying all true parents is influenced more strongly by the structure of the test function than by the number of additional non-parent variables. By contrast, the total number of selected variables increases approximately linearly with the input dimension. Because the number of true parents is fixed and FN remains relatively stable, this increase is driven primarily by the selection of additional false positives.

The effects of the batch size m. We fix d at 100, and vary m in $\{25, 50, 75, 100\}$, and report the mean numbers of missed parent variables (FN) and total selected variables (TP + FP) after 10 batches (i.e., t=10). All other parameters are kept at their default values. Figure 6 shows that increasing the batch size generally reduces the number of missed parent variables, with the largest improvement occurring between $m = 25$ and $m = 50$. Further increases provide smaller and function-dependent gains, indicating diminishing returns beyond $m = 50$. By contrast, the total number of selected variables remains broadly stable across the tested batch sizes. Because the number of true parents is fixed, the combination of lower FN and stable TP + FP suggests that larger batches improve the composition of the selected set by replacing false positives with true parents, rather than simply selecting more variables. However, since the number of batches is fixed at $t = 10$, increasing m also increases the total number of function evaluations from 250 to 1000. The observed improvement therefore reflects the combined effects of batch size and evaluation budget and should not be attributed solely to batch size.

The effects of the number of batches t. We vary t in $\{5, 10, 15, 20\}$, with the batch size m fixed at 50. All other parameters are kept at their default values. Figure 7 shows that increasing the number of batches generally reduces the number of missed parent variables, with the largest improvement

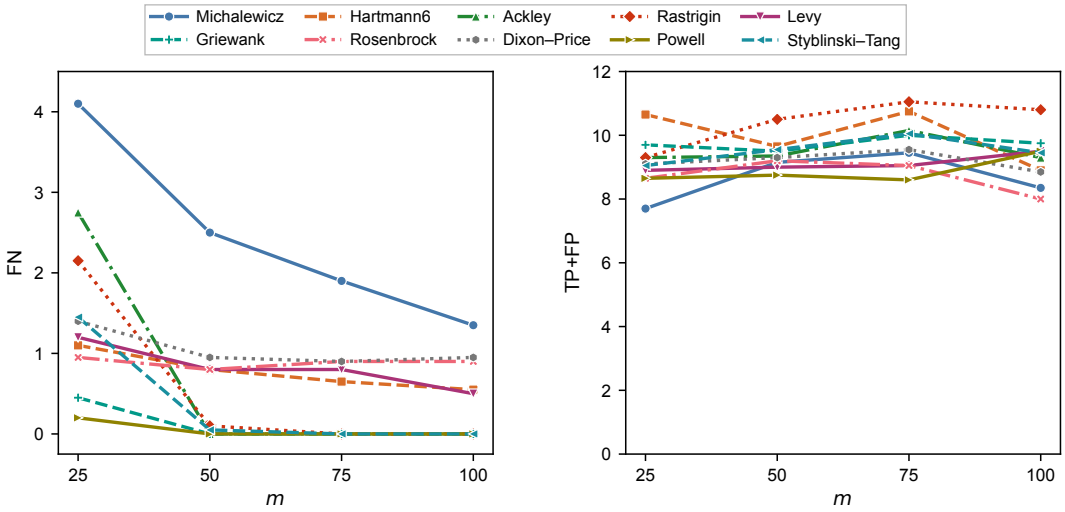


Figure 6: Effect of batch size m on parent-selection performance across ten test functions after $t = 10$ batches. Left: mean number of missed parent variables (FN). Right: mean total number of selected variables (TP + FP).

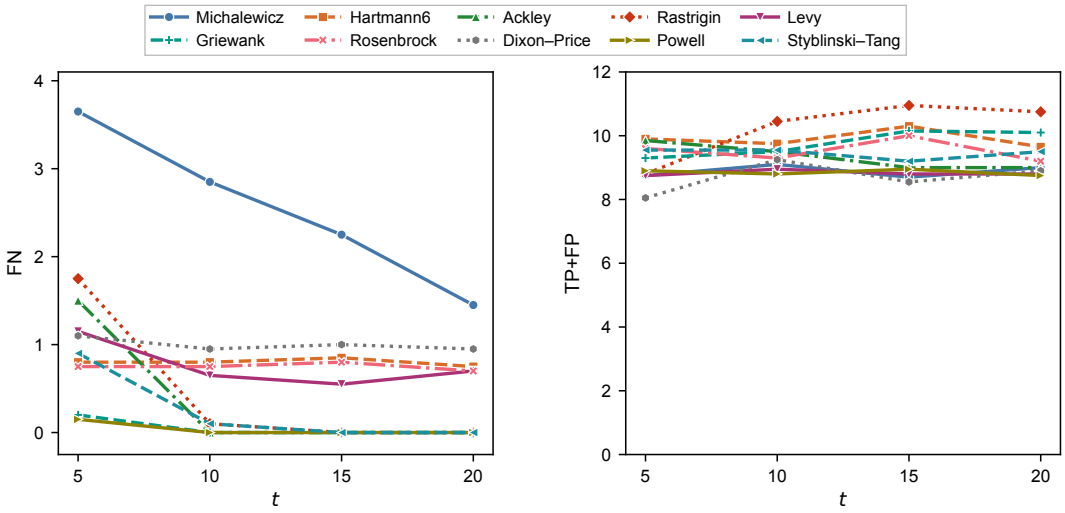


Figure 7: Effect of the number of batches t on parent-selection performance across ten test functions, with the batch size fixed at $m = 50$. Left: mean number of missed parent variables (FN). Right: mean total number of selected variables (TP + FP). Values $t \in 5, 10, 15, 20$ correspond to total evaluation budgets of $250, 500, 750, 1000$, respectively.

occurring between $t = 5$ and $t = 10$. For most test functions, FN changes only modestly beyond $t = 10$, indicating diminishing returns from additional batches. Michalewicz is a notable exception, for which FN continues to decrease as t increases. Meanwhile, the total number of selected variables remains broadly stable across the tested values of t. Because the number of true parents is fixed, the combination of decreasing FN and stable TP + FP suggests that additional batches improve the composition of the selected set, replacing false positives with true parents rather than simply increasing the number of selected variables. Overall, $t = 10$ provides a reasonable balance between parent-discovery performance and evaluation cost for most functions, whereas more batches may benefit structurally more challenging functions. Importantly, the results in Figure 7 are consistent with Property 3, which establishes that independence tests based on the coordinate-wise squared

sample distance covariance $\overline{\mathrm{dCov}}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$ are asymptotically more powerful than those based on the full squared sample distance covariance $\mathrm{dCov}_m^2(\mathbf{X}_j^{[t]}, \mathbf{X}_0^{[t]})$.

The effects of the number of parent variables. We fix d at 100, and vary the number of true parents in $\{10, 15, 20, 25, 30\}$. Increasing the number of parent variables will dilute the marginal signal associated with each individual parent, as variation in the response increasingly reflects contributions from other parents. To accommodate this more challenging setting, we set $p = 0.1$ and the batch size to $m = 75$, using 10 batches for a total of 750 function evaluations. All other parameters are kept at their default values. Figure 8 shows that the number of missed parent variables increases

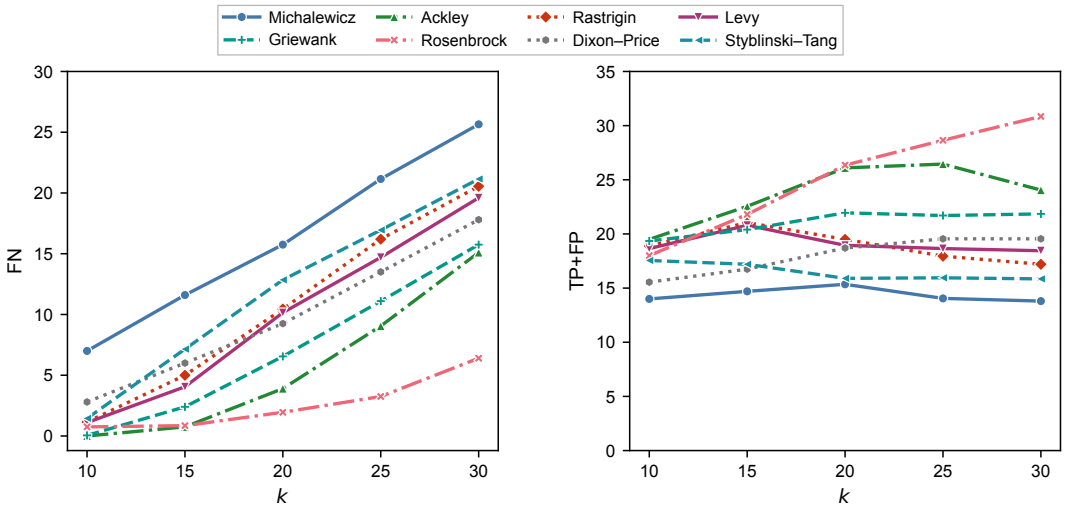


Figure 8: Effect of the number of true parent variables on parent-discovery performance across eight benchmark functions, with $d = 100$, $p = 0.1$, and batch size $m = 75$. Results are averaged over 20 random seeds after 10 batches (750 function evaluations). Left: mean number of missed parents (FN). Right: mean total number of selected variables (TP+FP).

with the true parent count k across all benchmark functions, although the magnitude varies considerably. (Hartmann6 and Powell are excluded because their standard formulations do not support all the parent counts considered: Hartmann6 has six input dimensions, while Powell requires the dimension to be a multiple of four.) Rosenbrock exhibits the strongest parent recovery, whereas Michalewicz consistently has the highest FN. Meanwhile, the total number of selected variables remains broadly stable for most functions, with Rosenbrock showing a sustained increase. These results indicate that parent discovery becomes more challenging as the number of true parents increases under a fixed evaluation budget. This pattern is consistent with dilution of individual marginal signals by contributions from other parents, which may reduce the power of marginal independence tests. We did not include the test functions

The effects of different probabilities $\rho = \Pr(|X_j^1 - X_j^2| \leq \frac{\hat{\ell}_j}{2})$, where $j \notin \hat{\pi}$. The ρ value for estimated non-parent variables is varied in $\{0.55, 0.60, 0.65, 0.70, 0.75\}$. The ρ value for estimated parent variables is fixed at 0.80. Figure 9 shows that parent-discovery performance is generally insensitive to the choice of ρ over the range 0.55-0.75. For both Gaussian and uniform intervention distributions, neither the mean FN nor the mean TP+FP exhibits a consistent monotonic relationship with ρ, and the best-performing value varies across the test functions. In particular, values between 0.60 and 0.70 produce broadly comparable results across both metrics. We therefore set $\rho = 0.65$ as a robust central value for estimated non-parent variables, rather than claiming that it is optimal for every function or sampling distribution.

The effects of different probabilities $\rho = \Pr(|X_j^1 - X_j^2| \leq \frac{\hat{\ell}_j}{2})$, where $j \in \hat{\pi}$. The ρ value for estimated parent variables is varied in $\{0.50, 0.60, 0.70, 0.80\}$. The ρ value for estimated non-parent

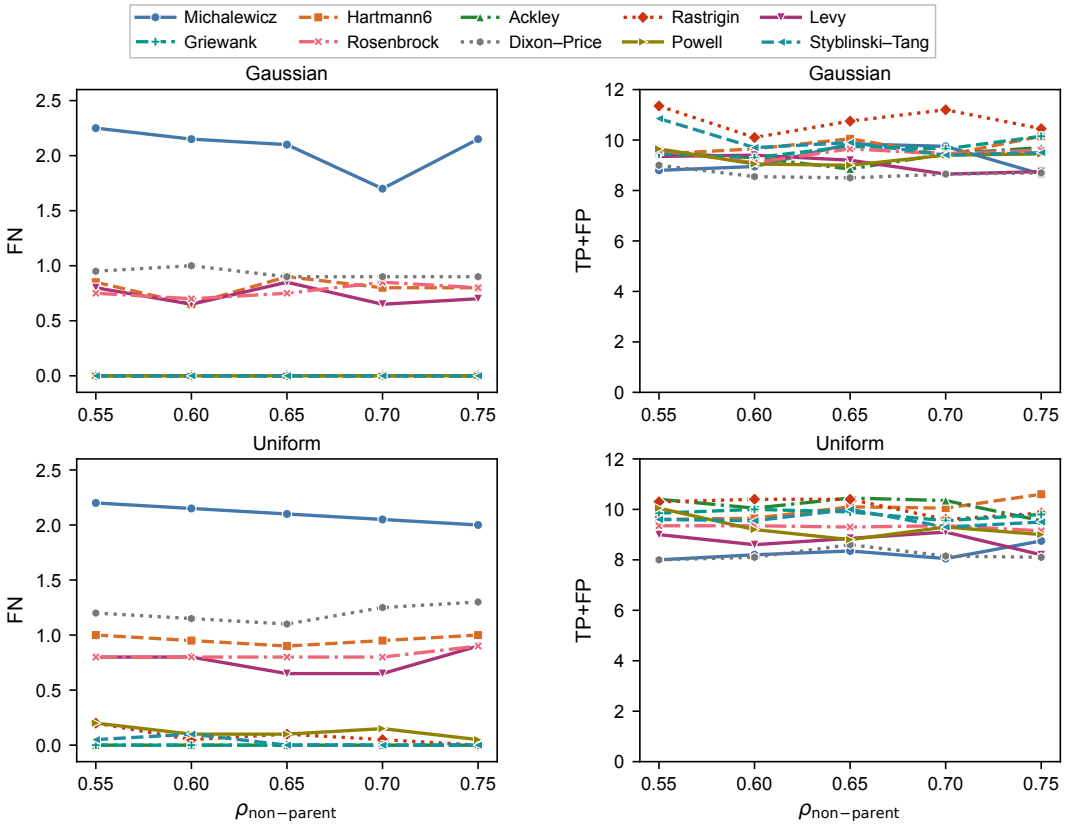


Figure 9: Sensitivity of parent-discovery performance to the probability $\rho = \Pr(|X_j^1 - X_j^2| \leq \frac{\hat{\ell}_j}{2})$, where $j \notin \hat{\pi}$. Top: Gaussian sampling; bottom: uniform sampling. Left: mean number of missed parent variables (FN); right: mean total number of selected variables (TP+FP). Results are reported across ten test functions, with the parent-variable probability fixed at 0.80.

variables is fixed at 0.65. Table 4 reports the absolute/relative gap between the global optimum and the best objective value identified by StoBO at the end of the experimental budget (750). Table

Table 4: Mean absolute/relative gap to the global optimum over 20 random seeds. The experimental budget is fixed at 750. The ρ value for non-parent variables is fixed at 0.65.

	Gaussian				Uniform			
ρ	50%	60%	70%	80%	50%	60%	70%	80%
Michalewicz	23.4%	22.2%	21.1%	18.8%	23.7%	22.2%	20.1%	21.3%
Hartmann6	1.97%	2.13%	2.06%	2.02%	2.82%	2.18%	2.13%	2.07%
Ackley	1.70	0.883	0.655	0.472	1.80	1.12	0.821	0.485
Rastrigin	16.5	16.7	14.3	12.5	17.0	15.3	14.0	14.9
Levy	1.10	1.10	0.807	0.923	1.39	1.34	1.15	1.17
Griewank	0.506	0.484	0.550	0.549	0.505	0.509	0.516	0.573
Rosenbrock	719	700	651	514	882	813	900	860
Dixon–Price	177	157	163	123	212	234	185	208
Powell	16.3	14.0	11.8	15.0	13.0	14.7	12.8	13.1
Styblinski–Tang	3.19%	6.29%	3.44%	2.38%	9.51%	6.56%	4.90%	3.56%

4 shows that the optimal ρ value that yields the smallest gap varies across the ten test functions. Nevertheless, $\rho = 0.8$ provides the best overall performance under Gaussian sampling, and therefore is adopted as the default. Performance is generally stable for ρ between 0.60 and 0.80, indicating that StoBO is not highly sensitive to the precise choice of this parameter. Under the specific settings

considered here, particularly with ρ fixed at 0.65 for the estimated non-parent variables, the Gaussian sampling distribution performs better overall than the uniform distribution. This result is specific to the present experimental configuration and does not imply that Gaussian sampling is universally superior.

StoBO with and without Block 2. Here we investigate whether Block 2 improves black-box optimization. When Block 2 is removed, each batch of queries is selected by optimizing the gradient-guided acquisition function over all input variables, regardless of whether they have been identified as parents. We compare the best objective values identified by the full StoBO algorithm and its ablated version without Block 2 under the same experimental budget. Upon termination, we record the absolute/relative gap between the best objective value identified by each algorithm and the global optimum. Each experiment is repeated over 20 random seeds, and Table 5 reports the mean absolute/relative gaps. Table 5 shows that incorporating Block 2 consistently improves the optimization

Table 5: Comparison of StoBO with and without Block 2. Entries report the mean absolute or relative gap between the best objective value attained and the known global optimum over 20 random seeds at evaluation budgets of 250, 500, 750, and 1000. Lower values indicate better optimization performance.

	With B2				Without B2			
Budget	250	500	750	1000	250	500	750	1000
Michalewicz (%)	43.04	27.05	19.38	15.46	43.33	36.25	32.71	28.10
Hartmann6 (%)	8.23	2.10	2.02	2.01	8.30	2.73	2.41	2.31
Ackley	13.40	1.26	0.43	0.32	14.02	4.27	2.43	1.78
Rastrigin	26.87	16.43	12.11	11.29	29.35	25.06	23.30	22.41
Levy	2.42	1.37	0.93	0.58	3.73	3.24	2.77	2.71
Griewank	1.16	0.82	0.55	0.42	1.04	0.94	0.92	0.83
Rosenbrock	1726	794.5	513.8	417.5	3274	2494	2346	2193
Dixon-Price	424.4	195.1	123.3	97.46	1249	853.5	747.8	666.6
Powell	29.85	19.20	15.06	9.73	41.81	40.10	35.25	34.28
Styblinski-Tang (%)	17.72	7.71	2.38	0.11	19.39	11.64	6.92	6.15

performance of StoBO. The full algorithm achieves a smaller gap to the global optimum for all test functions and evaluation budgets, except for Griewank at a budget of 250, where the ablated version performs marginally better. The advantage of Block 2 generally becomes more pronounced as the evaluation budget increases. At a budget of 1000, including Block 2 reduces the final optimality gap by approximately 13%-98% relative to the ablated version, with particularly substantial improvements for Ackley, Levy, Rosenbrock, Dixon-Price, Powell, and Styblinski-Tang. These results indicate that exploiting the estimated parent structure through Block 2 concentrates the search on influential variables, leading to faster convergence and better final objective values.

F.3 Details on the benchmark algorithms and real-world problems

Unlike StoBO, which performs batch optimization, all eight benchmark methods operate sequentially, proposing one evaluation point at a time. For all optimization algorithms, the number of initial samples was fixed at 50.

- HeSBO (GitHub: aminnayebi/HesBO): We used the default HeSBO-EI implementation. For the synthetic simulation, the dimension of the embedding space was fixed at 10. For the real-world problems, the dimension of the embedding space was fixed at default 20. All other algorithmic hyperparameters were kept at their default values.
- TuRBO (GitHub: uber-research/TuRBO): We used TuRBO-1 with the default ARD implementation. All algorithmic hyperparameters were kept at their default values.
- ALEBO (GitHub: facebookresearch/alebo): We used the default implementation of ALEBO, with the embedding dimension fixed at 10 and all other algorithmic hyperparameters set to their default values.
- SAASBO (GitHub: martinjankowiak/saasbo): The algorithm is extremely slow and hence we implemented the reduced-computation variant provided in the original paper.

- BAxUS (GitHub: lpapenme/BAxUS): We used the default BAxUS configuration. We set the initial target dimension to one and enabled automatic adjustment of the target dimension. All remaining algorithmic hyperparameters were kept at their default values.
- LassoBO (GitHub: Fsoft-AIC/LassoBO): We used the default implementation of LassoBO, with all algorithmic hyperparameters set to their default values.
- Standard BO (GitHub: XZT008/Standard-GP-is-all-you-need-for-HDBO): We used SBO-Matérn (RI) with an ARD Matérn-(5/2) kernel and robust length-scale initialization ($\ell_0 = \sqrt{d}$). All algorithmic hyperparameters were kept at their default values except that the UCB exploration parameter was changed from $\beta = 1.5$ to $\beta = 2.0$.
- GIT-BO (GitHub: rosenyu304/GITBO): We used GIT-BO with its default configuration and the pretrained TabPFN-v2 regression checkpoint provided by the repository. For the synthetic simulation, the subspace rank was set to 10. For real-world problems, the subspace rank was set to default 15.
- StoBO (https://anonymous.4open.science/r/StoBO): For all experiments, unless otherwise stated, we configured StoBO with batch size $m = 50$. Independence test used a significance level of $\alpha = 0.05$ with 1000 permutations. The surrogate model was an ARD Matern-$5/2$ GP, and the acquisition function was UCB with $\beta = 2.0$. Following Xu et al. (2025), we initialize every ARD length scale to $\ell_0 = \sqrt{p}$, where p is the number of input features. The stochastic intervention distribution is Gaussian with pairwise locality probabilities of 0.80 and 0.65 for parent and non-parent variables, respectively. For the real-problems, we set the batch size to $m = 75$ and $p = 0.1$, guided by prior knowledge that many input features may influence the response.

Among the eight benchmark methods, LassoBO is the only one that explicitly identifies relevant variables. We therefore compare the variable-selection performance of StoBO and LassoBO. Table 6 reports the mean numbers of missed relevant variables (FN) and total selected variables (TP + FP) for each algorithm at the end of each evaluation budget. Both StoBO and LassoBO perform well on

Table 6: Mean numbers of false negatives (FN; top) and selected variables (TP + FP; bottom) across 20 random seeds at the end of each evaluation budget.

	StoBO				LassoBO			
Budget	250	500	750	1000	250	500	750	1000
Michalewicz (5)	3.50	2.90	2.50	2.10	0.80	0.30	0.20	0.10
	8.50	9.00	9.00	9.50	8.60	7.00	4.80	4.90
Hartmann6 (6)	0.80	0.70	0.80	0.80	1.10	1.20	1.10	1.20
	10.40	9.70	9.20	8.90	5.00	4.80	4.90	4.80
Ackley (5)	3.00	0.40	0.40	0.40	0.00	0.00	0.00	0.00
	8.00	10.30	9.70	9.10	12.20	15.70	15.50	13.10
Rastrigin (5)	2.10	0.50	0.40	0.40	2.40	0.70	0.90	0.40
	9.30	11.70	10.90	10.40	17.40	22.90	22.30	23.90
Levy (5)	1.70	1.00	1.10	1.00	0.50	0.40	0.60	0.80
	8.40	8.90	8.60	8.50	8.50	10.80	8.10	5.60
Griewank (5)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	9.90	10.30	9.50	9.80	5.00	5.00	5.00	5.00
Rosenbrock (5)	0.70	0.80	0.80	0.80	1.00	1.00	1.00	1.00
	9.80	8.50	8.70	8.50	4.00	4.00	4.00	4.00
Dixon-Price (5)	1.40	1.20	1.20	1.20	1.00	1.00	1.00	1.00
	8.10	8.60	8.20	8.10	4.00	4.00	4.00	4.00
Powell (4)	0.30	0.10	0.10	0.10	0.00	0.00	0.00	0.00
	8.60	9.10	9.60	9.40	4.00	4.00	4.00	4.00
Styblinski-Tang (5)	1.90	0.00	0.00	0.00	0.20	0.00	0.00	0.00
	9.50	10.70	9.80	9.30	5.70	5.00	5.00	5.00

the Griewank, Powell, and Styblinski-Tang functions, with neither method missing any true parents, even under small evaluation budgets. However, LassoBO selects fewer false positives. LassoBO also outperforms StoBO on the more challenging Michalewicz function, whereas StoBO performs

better on Ackley and Rastrigin. The two methods achieve comparable results on the remaining four functions. Across all test functions, however, StoBO is substantially faster than LassoBO.

We here explain the five real-world high-dimensional optimization problems.

- **DNA** ($d = 181$): We used the dna.scale training partition distributed through LIBSVM, comprising 2,000 DNA sequences represented by 180 binary attributes and assigned to three splice-junction classes (Chang & Lin, 2011). The dataset is a processed version of the UCI Molecular Biology Splice-junction Gene Sequences dataset (DOI: 10.24432/C5M888). Šehić et al. (2022) reported that the effective dimensionality is approximately 43.
- **Mazda** ($d = 223$): The Mazda CdMOBP (Kohira et al., 2018) is a constrained discrete multi-objective optimization problem in which the plate thicknesses of 74 structural parts per vehicle are optimized simultaneously for three car models, resulting in 222 design variables. The overall objectives (minimizing total weight and maximizing the number of common parts that share the same plate thickness across the three cars) depend on all 222 variables, whereas each vehicle-specific performance constraint depends only on its corresponding 74-variable subset, leaving the other 148 variables inactive for that constraint. Following Yu et al. (2026), we convert the problem into an unconstrained single-objective minimization task. Because the number of common parts enters the objective with a negative sign, the resulting objective value can be negative.
- **SVM** ($d = 389$): Eriksson & Jankowiak (2021) used the UCI CT-slice dataset (DOI: 10.24432/C5CP6G), which contains 385 predictors and a target describing each slice's relative position along the human body. They randomly selected 5,000 observations for training and 5,000 for testing and trained a kernel support-vector regressor to predict slice position. This created a 388-dimensional hyperparameter-tuning problem comprising an individual kernel length scale for each predictor and three global SVR hyperparameters, ϵ, C, and γ. Eriksson & Jankowiak (2021) identified these three global hyperparameters as the most influential, while only a small subset of the feature-specific length scales showed secondary importance.
- **MOPTA08** ($d = 125$): Eriksson & Jankowiak (2021) considered the MOPTA08 vehicle-design benchmark, a 124-dimensional problem derived from a real General Motors multidisciplinary vehicle-mass optimization study (Jones, 2008). The design variables describe component gauges, materials and vehicle shape. The problem also includes 68 engineering performance constraints. The public benchmark evaluates candidate designs using Kriging response surfaces fitted to automotive simulation results rather than the original computationally expensive vehicle models. Eriksson & Jankowiak (2021) converted the constrained problem into a scalar objective by penalizing constraint violations. Shen & Kingsford (2023) reported that the variables have varying levels of influence on the objective.
- **CEC2020-38** ($d = 127$): The CEC 2020 power-system benchmark suite (Kumar et al., 2020) is a collection of 57 deterministic, constrained engineering optimization models. We here focus on the 30-bus optimal-power-flow problem that minimizes electricity-generation fuel cost. The numerical representation has 126 variables: 116 describe bus-voltage components and active/reactive power quantities, and ten are explicit controls representing the active and reactive outputs of five generators; the output of the slack generator is determined by the power-flow solution. The 116 equality constraints enforce active- and reactive-power balance throughout the network. Therefore, CEC2020-38 is numerically 126-dimensional, but its feasible solutions have approximately ten independent control degrees of freedom. As with Yu et al. (2026), we incorporate the constraints into the objective as penalty terms.

Figure 10 compares the running minimum objective values obtained by the nine BO algorithms. Of the eight sequential BO algorithms, six were run for 1000 iterations, while ALEBO and SAASBO were limited to 500 iterations because of their higher computational cost. Each trajectory represents the mean over 20 random seeds. Figure 10 shows that StoBO achieves competitive optimization performance across the benchmark functions. It approaches the known global optimum on Hartmann6, Griewank and Styblinski-Tang, and performs favourably against several sequential BO methods on the remaining functions. However, its performance is function-dependent, and the strongest sequential methods converge faster or attain better final objective values on several benchmarks, particularly Michalewicz, Rastrigin and Dixon-Price.

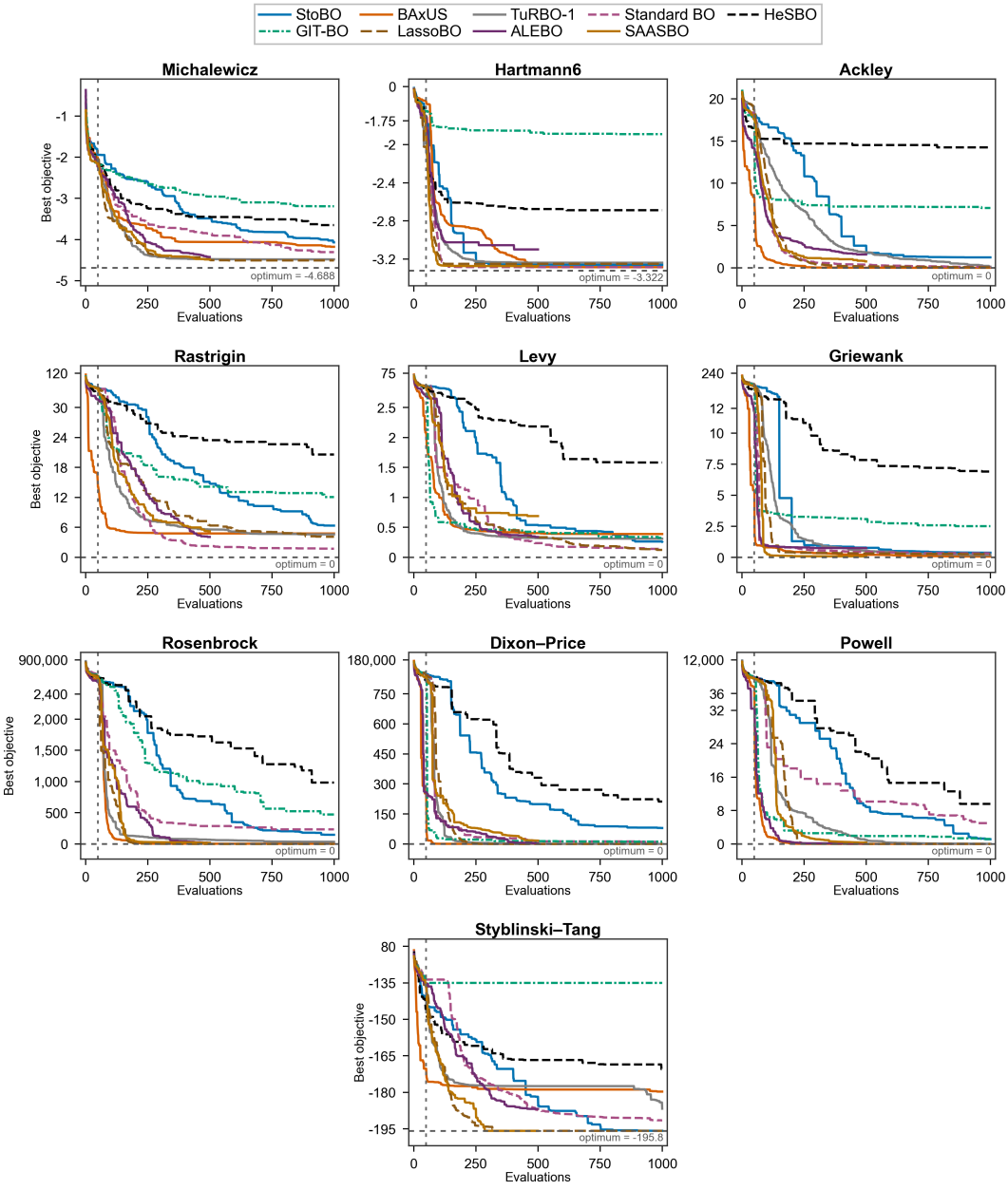


Figure 10: Trajectories of the lowest objective value obtained by each of the nine BO algorithms, with each panel representing one test function. Owing to their high computational cost, ALEBO and SAASBO were run for 500 iterations (i.e., 500 evaluations), whereas the other six benchmarking methods were run for 1000 iterations.

Figure 11 compares the running minimum objective values obtained by the seven BO algorithms in Table 2 on the five real-world problems. Each trajectory is averaged over 20 random seeds. The trajectories show that StoBO improves steadily and generally avoids the early stagnation observed for HeSBO and GIT-BO. Its strongest relative performance occurs on Mazda, where it achieves the second-lowest final objective value. It remains competitive on DNA, SVM, and MOPTA08, but converges more slowly on CEC2020-38. Overall, StoBO provides competitive performance and efficient batch optimization with the additional advantage of directly identifying the most influential (parent) variables.

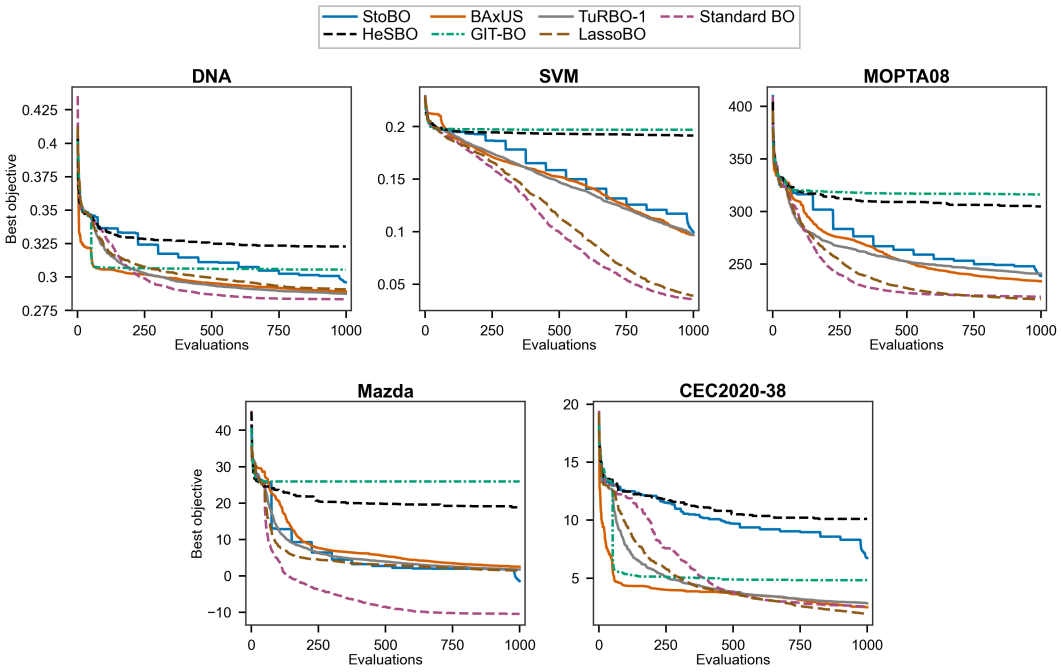


Figure 11: Trajectories of the lowest objective value obtained by each of the seven BO algorithms, with each panel representing one real-world problem. All methods were run for 1000 function evaluations.

Table 7 compares the runtime of the nine BO algorithms on the Ackley test function with $d = 100$ variables and five true parents. Each algorithm is evaluated over 20 random seeds with a budget of 500 function evaluations. All sequential benchmarks use the configurations described in Appendix F.1. For StoBO, we set the batch size to $m = 50$, and $p = 0.05$; StoBO terminates when the estimated parent set $\hat{\pi}$ remains unchanged across two consecutive batches. All experiments are conducted on a system with one NVIDIA A100 40 GB GPU, 11 CPU cores, and 90 GB RAM. GPU acceleration is used where supported by the original implementation; otherwise, the corresponding CPU implementation is used. StoBO is approximately 1.6 times faster than HeSBO and TuRBO-1

Table 7: Mean runtime per seed on the Ackley test function.

Algorithm	Runtime (s)
StoBO	116.619
BAxUS	4955.462
Standard BO	2884.863
LassoBO	3817.448
HeSBO	184.900
ALEBO	41844.893
SAASBO	23504.103
TuRBO-1	181.958
GITBO	758.239

and substantially faster than all other benchmarks. In particular, StoBO is over 20 times faster than Standard BO and several hundred times faster than ALEBO, demonstrating its strong computational efficiency in high-dimensional optimization.

G LIMITATIONS

The StoBO algorithm can be readily applied to problems with mixed variables: continuous and discrete, by using suitable surrogate models and acquisition functions. However, when some variables are non-intervenable, Property 1 may no longer hold in general. A sufficient condition under which the property remains valid is that the non-intervenable variables are root variables, i.e., they have no incoming edges. More generally, non-intervenability does not invalidate the property by itself. The property remains valid as long as the non-intervenable variables do not induce any remaining indirect or backdoor paths to X_0 after intervention. Equivalently, Property 1 continues to hold if the post-intervention graph is star-shaped into X_0, meaning that all remaining edges are direct incoming edges into X_0.

The main computational bottleneck of StoBO is Block 3, which involves GP fitting and gradient-guided acquisition optimization. Let $n = mt$ be the number of training observations. GP fitting costs $\mathcal{O}(n^3 + n^2 d)$, where $\mathcal{O}(n^2 d)$ comes from constructing the kernel matrix and $\mathcal{O}(n^3)$ from its Cholesky decomposition. After fitting the GP, evaluating the posterior mean and variance at one candidate point costs approximately $\mathcal{O}(n^2 + nd)$. If a gradient-based optimizer is used, its additional overhead is roughly $\mathcal{O}(d)$ per iteration, assuming fixed memory. The independence test (Block 1) costs $\mathcal{O}(d \times t \times m \log(m))$=$\mathcal{O}(dn \log(m))$. Therefore, the per-iteration complexity of StoBO is dominated by GP fitting, scaling approximately as $\mathcal{O}(n^3)$ when n is large. In contrast to the eight state-of-the-art algorithms in Table 2, which update their models after every new observation, StoBO performs an update only after every m new observations. It therefore requires approximately $1/m$ as many model updates, resulting in an approximately m-fold reduction in update-related computational time.

In Appendix D, we explain the rationale for dividing the estimated length scale by two in the probability $\rho = \Pr(|X_j^1 - X_j^2| \leq \frac{\hat{\ell}_j}{2})$. With small sample sizes, the length scale tends to be overestimated; the factor of two therefore provides a conservative correction. As $\hat{\ell}_j$ is updated after each new batch, however, a fixed correction factor may become less appropriate. An adaptive formulation could instead be used, for example,

$$\rho = \Pr(|X_j^1 - X_j^2| \leq \frac{\hat{\ell}_{j,t}}{2/\log(t+1)}), \;\; t \geq 2$$

where t denotes the number of batches. We plan to investigate this adaptive approach in future work.

Finally, StoBO's parent-discovery performance may deteriorate as the number of true parents increases. Under a fixed evaluation budget, contributions from other parents can weaken individual marginal signals, reducing the power of marginal independence tests. Sensitivity to increasing effective dimensionality also affects other variable-selection-based BO methods, although the mechanisms differ. For example, LassoBO (Hoang et al., 2025) relies on sparse estimation of GP inverse length scales, and its authors acknowledge potential performance degradation when the proportion of relevant variables is high.