A Differentiable Bayesian Optimization Framework via Variational Mutual Information Estimation

Anonymous Authors[1]

Abstract

Bayesian optimization (BO) of expensive black-box functions is traditionally addressed with Gaussian processes (GPs), which scale cubically with observations, or Bayesian neural networks (BNNs), which incur costly posterior sampling and inner-loop acquisition optimization. We propose VBO-MI (Variational Bayesian Optimization with Mutual Information), a fully gradient-based BO framework that requires no explicit GP prior or fixed parametric posterior family over objective function and treats it as a strict black box. An actor-critic architecture pairs an action-net with a variational critic that estimates information gain, eliminating the acquisition optimization bottleneck and achieving up to $10^2\times$ fewer FLOPs than BNN-BO baselines. A lightweight surrogate network further reduces real function queries to one batch per iteration. We establish consistency guarantees and evaluate VBO-MI on synthetic benchmarks (Ackley, Levy, Griewank) and real-world tasks (Rover Trajectory, Lunar Lander, Pest Control), demonstrating competitive or superior performance over the baselines.

1. Introduction

Bayesian optimization (BO) has matured into a de-facto tool for expensive global black-box optimization with applications in hyperparameter tuning, materials design, and experimental sciences (Frazier, 2018; Garnett, 2023). In Bayesian Optimization, the algorithm treats f as a black-box function and aims to optimize it sequentially under a finite budget T—with access to only T evaluations (input–output pairs) of the function (Cox & John, 1992). At each evaluation, the algorithm builds a probabilistic model of f using the observed input–output pairs, and uses this model to guide the selection of the next evaluation points by maximizing an acquisition function that trades off exploration and exploitation.

Traditionally, GP assumptions for the surrogate coupled with acquisition functions such as expected improvement (Hennig & Schuler, 2012; Hennig et al., 2022) and UCB (Srinivas et al., 2010) have driven significant progress (Rudner et al., 2022). However, GPs require inverting a covariance matrix (cubic scaling in observations) and become computationally prohibitive for large datasets. Although approximate GPs have been proposed (McIntire et al., 2016; Jimenez & Katzfuss, 2023), they still face trade-offs between accuracy, local fidelity, and uncertainty quantification. Recent works replace GPs with BNNs (Li et al., 2024; Springenberg et al., 2016), using variational inference with Sparse Gaussian assumptions (Wei et al., 2024), Gamma variational distributions (Cheng et al., 2025), or HMC (Li et al., 2024). However, all these methods rely on specific distributional forms for posterior sampling, limiting their capability in complex, high-dimensional settings, and incurring significant computational cost (Cheng et al., 2025; Li et al., 2024).

To address these limitations, this work develops a framework that models f without assuming a specific parametric family. By leveraging a flexible variational representation, the proposed approach enables more expressive uncertainty modeling and improved exploration. Our method enables full gradient flow through the internal optimization process while maintaining strict zeroth-order access to f. We leverage recent advances in variational mutual information estimation (Song & Ermon, 2020; Belghazi et al., 2018) to design a novel acquisition function approximated directly from data samples, jointly trained with an action-net that explores the input space. Our main contributions are:

1. A general BO framework that does not assume any specific variational distribution form for the surrogate, allowing greater flexibility in modeling complex objective functions.

2. A novel variational acquisition function derived from a variational MI formulation, providing a principled exploration–exploitation trade-off.

3. A lightweight surrogate network that replaces real oracle

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

calls inside the inner loops, reducing oracle queries to one batch per iteration (a 16× reduction).

4. Theoretical consistency guarantees for the joint optimization of all three networks.

5. Competitive or superior performance over diverse baselines on benchmarks from $d = 10$ to $d = 60$.

Notation. Random vectors are denoted by bold lowercase symbols (e.g., $\mathbf{x}_t \in \mathbb{R}^d$). The sequence up to step t is $\mathbf{X}_t = [\mathbf{x}_1, \ldots, \mathbf{x}_t]$. A realization uses superscript (i), so $\mathbf{x}_t^{(i)} \in \mathbb{R}^d$. Calligraphic letters (e.g. $\mathcal{D}$) denote sets. $I(\mathbf{x}; y)$ denotes mutual information between a vector and scaler random variables; $\mathrm{KL}(P\|Q)$ denotes KL divergence between two probability measures P, and Q.

2. Related Work

Scalable Bayesian optimization. GPs remain the canonical choice for BO due to their well-calibrated uncertainty (Srinivas et al., 2010), but their $O(T^3)$ complexity has spurred scalable alternatives: Vecchia approximations (Jimenez & Katzfuss, 2023), Trust-Region BO (Eriksson et al., 2019), and Focalized Sparse GPs (Wei et al., 2024). While effective, these methods rely on stationary kernel assumptions that struggle with complex, non-linear landscapes.

Variational and Sampling-Based BNNs. BNNs offer superior scalability and representation power. HMC (Springenberg et al., 2016; Li et al., 2024) provides high-quality posteriors at prohibitive cost. Variational Inference via VBLL (Harrison et al., 2024) and PFNs (Müller et al., 2023) reduce sampling costs but still rely on parametric posterior families that may cause model misspecification.

Information-Theoretic acquisition functions. PES (Hernandez-Lobato et al., 2015) and MES (Wang & Jegelka, 2017) offer principled non-myopic exploration. JES (Hvarfner et al., 2022) models the joint density but reverts to GP assumptions for tractability. A recent unified framework (Cheng et al., 2025) introduces acquisition functions constrained to specific variational families such as Gamma. Closest to our work, Ishikura & Karasuyama (2025) propose a variational lower bound for entropy search, but remain rooted in the GP framework. In contrast, our actor-critic formulation makes no GP assumptions on f and eliminates the inner-loop optimization bottleneck entirely.

3. Problem Setting and Background

3.1. Problem Setting

We consider maximizing an unknown scalar-valued objective $f : \mathcal{D} \to \mathbb{R}$ over compact $\mathcal{D} \subseteq \mathbb{R}^d$:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{D}} f(\mathbf{x}). \tag{1}$$

The function is a strict black box: no analytical form or gradients are available. At iteration t, the optimizer selects $\mathbf{x}_t$ and receives noisy observation $y_t = f(\mathbf{x}_t) + \varepsilon_t$, where ε_t is independent zero-mean noise. We denote $N = T \cdot B$ the total function evaluations, with each of T main iterations querying one batch of B points.[1]

3.2. Bayesian Optimization and Information Gain

BO maintains a probabilistic surrogate over f and uses an acquisition function $\alpha(\mathbf{x} \mid \mathbf{X}_t, \mathbf{Y}_t)$ to select the next input. The information gain (Srinivas et al., 2010) quantifies information about f from observations $\mathbf{Y}_T$:

$$I(\mathbf{Y}_T; f) = H(\mathbf{Y}_T) - H(\mathbf{Y}_T \mid f). \tag{2}$$

Under a GP assumption, sequentially maximising this gives the GP-UCB rule (Srinivas et al., 2010):

$$\mathbf{x}_t = \arg\max_{\mathbf{x}\in\mathcal{D}} \sqrt{\beta}\,\sigma_{t-1}(\mathbf{x}) + \mu_{t-1}(\mathbf{x}), \tag{3}$$

where μ_{t-1} and σ_{t-1} are the GP posterior mean and standard deviation, and β controls the exploration–exploitation trade-off.

3.3. Variational Mutual Information

To incorporate Mutual Information (MI) into our optimization strategy without relying on restrictive assumptions (e.g., Gaussianity), we employ a variational lower bound based on the Donsker–Varadhan (DV) representation (Belghazi et al., 2018; Donsker & Varadhan, 1975) of the Kullback–Leibler (KL) divergence.

Let P and Q be two probability measures over the same measurable space. The DV representation expresses the KL divergence as

$$D_{\mathrm{KL}}(P \,\|\, Q) = \sup_{T_c} \mathbb{E}_P[T_c] - \log \mathbb{E}_Q[e^{T_c}], \tag{4}$$

where the supremum is taken over all measurable functions T_c such that the expectations are finite (Belghazi et al., 2018; Donsker & Varadhan, 1975). Now, consider random vector $\mathbf{x}$ and random variable y with joint distribution $P_{\mathbf{x},y}$, and let $Q = P_{\mathbf{x}} \times P_y$ denote the product of marginals. Then, the MI

[1] We also perform W warm-up iterations; see Algorithm 1.

between $\mathbf{X}$ and y can be written as

$$I(\mathbf{x}; y) = D_{\mathrm{KL}}\big(P_{\mathbf{x},y} \,\|\, P_{\mathbf{x}\times y}\big) = \tag{5}$$

$$\sup_{T\in\mathcal{T}} \mathbb{E}_{P_{\mathbf{x},y}}[T(\mathbf{x}, y)] - \log \mathbb{E}_{P_{\mathbf{x}}\times P_y}[e^{T(\mathbf{x},y)}]. \tag{6}$$

In practice, we parameterize $D(\mathbf{x}, y)$ using a Neural Network $D_\theta(\mathbf{x}, y)$ with parameters θ, and estimate the expectations via sampling. This yields the following variational lower bound:

$$I(\mathbf{x}; y) \geq \max_{\theta\in\Theta} \mathbb{E}_{P_{\mathbf{x},y}}[D_\theta(\mathbf{x}, y)] - \log \mathbb{E}_{P_{\mathbf{x}}\times P_y}[e^{D_\theta(\mathbf{x},y)}]. \tag{7}$$

Given samples of distributions $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{n=1}^{N}$, the Neural Network T_θ is trained by minimizing the following loss function:

$$L_{D_\theta} = -\frac{1}{B}\sum_{i=1}^{B} D_\theta(\mathbf{x}^{(i)}, y^{(i)}) + \log\left(\frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B} e^{D_\theta(\mathbf{x}^{(i)}, y^{(j)})}\right), \tag{8}$$

where B denotes the minibatch size. This formulation provides a scalable, sample-based estimator of MI, and integrate into our Bayesian Optimization framework to guide exploration.

4. VBO-MI: Proposed Framework

4.1. Exploration and Mutual Information Gain Estimation

Using results of previous section, our goal here is to find a variational bound for exploring the function f. Based on the defined setting, at time step t we have access to $\mathbf{X}_t = [\mathbf{x}_1, ..., \mathbf{x}_t]$, of current chosen actions. We also, assume that these actions are chosen according to $P_{\mathbf{X}_t}$. Moreover, corresponding observations are $\mathbf{Y}_t = [y_1, .., y_t]$, where $y_t = f(\mathbf{x}_t) + \epsilon_t$, and ϵ_t is zero mean, independent, identically distributed (i.i.d.) noise. Our goal in this section is to find a lower bound on the information gain quantity which corresponds to the exploration term in (3), which is:

$$I_G = I(f(\mathbf{X}_t), \mathbf{Y}_t), \tag{9}$$

where we have[2] $\mathbf{Y}_t = f(\mathbf{X}_t) + \mathcal{E}_t$, $\mathcal{E}_t = [\epsilon_i]_{i=1}^t$, and $f(\mathbf{X}_t) = [f(\mathbf{x}_i)]_{i=1}^t$.[3] To find a simple bound on this quantity using the bound of (Belghazi et al., 2018), and (6), we first note that we have the following Markov chain (Cover & Thomas, 2012):

$$\mathbf{X}_t \longleftrightarrow f(\mathbf{X}_t) \longleftrightarrow \mathbf{Y}_t \longleftrightarrow y_t. \tag{10}$$

[2]Depending on its argument, $f(\mathbf{x}_t)$ represents a scalar value, or a vector, $f(\mathbf{X}_t)$.

[3]Note that here we don't have a Gaussian assumption (for f or noise), and therefore we cannot use the closed form for Gaussian distribution entropy)

Based on the data-processing inequality (Cover & Thomas, 2012) we have

$$I(\mathbf{X}_t; y_t) \leq I_G = I(f(\mathbf{X}_t), \mathbf{Y}_t). \tag{11}$$

By expanding the mutual information via chain rule we have:

$$I(\mathbf{X}_t; y_t) = I(\mathbf{x}_t; y_t) + I(\mathbf{x}_{t-1}; y_t \mid \mathbf{x}_t) \tag{12}$$

$$+ I(\mathbf{x}_{t-2}; y_t \mid \mathbf{x}_t, \mathbf{x}_{t-1}) \tag{13}$$

$$+ \cdots + I(\mathbf{x}_1; y_t \mid \mathbf{x_t}, \ldots \mathbf{x}_2), \tag{14}$$

except for the first term, all the conditional terms are zero due to Markov properties (Cover & Thomas, 2012). On the other hand, by using the variational representation in (4):

$$I_\theta(\mathbf{x}_t, y_t) = \sup_{\theta\in\Theta} \mathbb{E}_{P_{\mathbf{x}_t,y_t}}[D_\theta(\mathbf{x}_t, y_t)] + \log(\mathbb{E}_{P_{\mathbf{x}_t}\times P_{y_t}}[e^{D_\theta(\mathbf{x}_t,y_t)}]). \tag{15}$$

By noting that $I_\theta(\mathbf{x}_t, y_t) \leq I(\mathbf{x}_t, y_t)$, and combining (11) and (15) we have the following bound on the exploration term, or IG bound in (3):

$$\mathbb{E}_{P_{\mathbf{x}_t,y_t}}[D_\theta(\mathbf{x}_t, y_t)] + \log(\mathbb{E}_{P_{\mathbf{x}_t}\times P_{y_t}}[e^{D_\theta(\mathbf{x}_t,y_t)}]) \leq I_G. \tag{16}$$

We choose y_t, the current observation, rather than the full history $\mathbf{Y}_t$: by the Markov chain in (10), y_t captures the *marginal* information gained at step t given the already-observed history, avoiding double-counting of information already incorporated in previous iterations. This choice allows the exploration signal to remain focused on the incremental uncertainty reduction provided by the newest query, rather than considering redundant historical observations which would also increase the complexity of evaluating the mutual information bound.

4.2. A Data-Driven Exploitation term

To take into account the exploitation capability of our algorithm, we also estimate exploitation term using a data-driven approach. The goal is use these results in our acquisition function (loss of a neural network) to choose the best actions to approximate and maximize f. In other word, our loss function should simultaneously consider the balance of exploration and exploitation in searching the input space. In the previous subsection, we have seen how the information gain quantity can be leveraged to consider the exploration of the input space. To account for the exploitation capability of our algorithm, we need to consider finding the maximum of f by utilizing our current knowledge. A straightforward approach to find the maximum is to consider the expectation of the function f conditioned on previous points (posterior of f), e.g:

$$\mathbb{E}[f \mid y_{t-1}, .., y_1, \mathbf{x}_{t-1}, .., \mathbf{x}_1]. \tag{17}$$

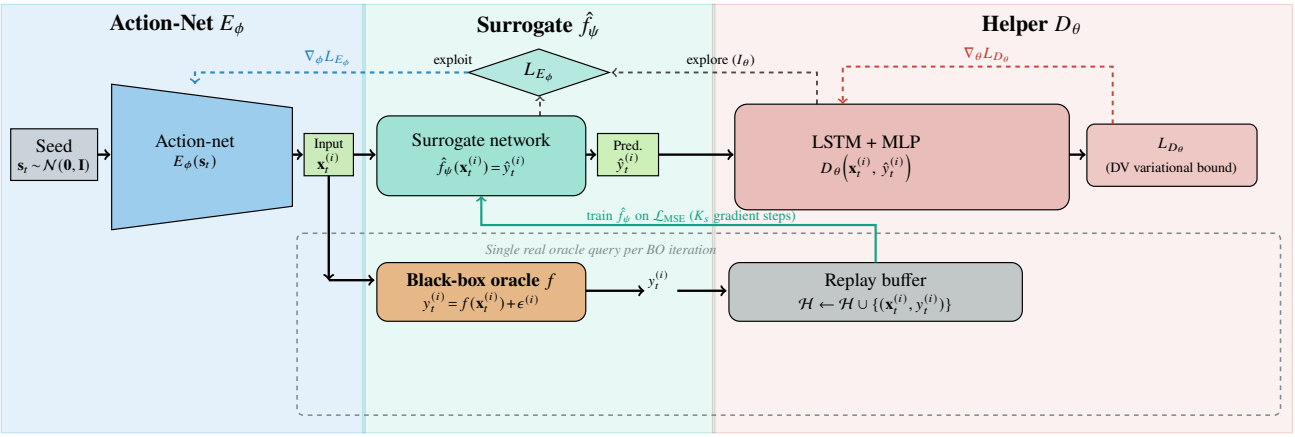


Figure 1. VBO-MI framework. An *action-net* E_ϕ maps random seeds $\mathbf{s}_t$ to candidate inputs $\mathbf{x}_t^{(i)}$. A *surrogate* $\hat{f}_\psi$ predicts rewards $\hat{y}_t^{(i)}$, enabling oracle-free updates of the helper D_θ (K_a steps) and action-net (K_b steps) via the acquisition loss L_{E_ϕ}, which balances exploitation (surrogate mean) and exploration (variational MI estimate I_θ). One real oracle query per iteration yields $y_t^{(i)}$, which replenishes the replay buffer $\mathcal{H}$ used to retrain $\hat{f}_\psi$ via MSE (K_s steps). Dashed arrows indicate gradient flows.

Furthermore, we note that by assuming that f and noise are independent, and noise has a zero mean, we have:

$$\mathbb{E}[f \mid y_{t-1}, .., y_1, \mathbf{x}_{t-1}, .., \mathbf{x}_1] =$$
$$\mathbb{E}[f + \epsilon_t \mid y_{t-1}, .., y_1, \mathbf{x}_{t-1}, .., \mathbf{x}_1] = \quad (18)$$
$$\mathbb{E}[y_t \mid y_{t-1}, .., y_1, \mathbf{x}_{t-1}, .., \mathbf{x}_1]. \quad (19)$$

In practice, computing $\mathbb{E}[y_t \mid y_{t-1}, \ldots, y_1, \mathbf{x}_{t-1}, \ldots, \mathbf{x}_1]$ analytically requires a parametric surrogate such as a Gaussian Process, which reintroduces the assumptions we seek to avoid. Instead, we approximate this conditional expectation directly from data using a neural network surrogate $\hat{f}_\psi : \mathcal{D} \to \mathbb{R}$, trained by minimizing the mean squared error on all past observations in replay buffer $\mathcal{H} = \{(\mathbf{x}_{t'}^{(i)}, y_{t'}^{(i)})\}$, $t' \leq t$:

$$\mathcal{L}_{\text{MSE}}(\psi) = \frac{1}{B} \sum_{i=1}^{B} \big(\hat{f}_\psi(\mathbf{x}^{(i)}) - y^{(i)}\big)^2, \quad (20)$$

$$\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{B} \overset{\text{i.i.d.}}{\sim} \text{Uniform}(\mathcal{H}). \quad (21)$$

The trained surrogate $\hat{f}_\psi(\mathbf{x})$ then serves as a point estimate of $\mathbb{E}[y_t \mid \mathcal{H}]$, which we call it $\hat{y}_t$ at any proposed action $\mathbf{x}$, and its gradient $\nabla_{\mathbf{x}} \hat{f}_\psi$ provides a differentiable exploitation signal for updating E_ϕ without any parametric assumption on f and without querying the black-box function.

4.3. Variational Formulation of BO

Our first observation is that the GP-UCB policy (3) implicitly depends on the MI between f and the observations $\mathbf{y}_t$ conditioned on selected actions. Based on the previous sections proposed exploration and exploitation terms, we formulate the BO problem as:

$$\phi^* = \arg\max_{\phi} \min_{\theta} \; g(\phi, \theta), \qquad \mathbf{x}_t = E_{\phi^*}(\mathbf{s}_t), \quad (22)$$

where E_ϕ (action-net) and D_θ (helper) are two neural networks trained in an alternating actor-critic fashion. In a high-level view the action-net search in the input space to find optimal inputs and helper network computes mutual information and contributes to computation of our acquisition function (g) to guide action-net. To the best of our knowledge, this is the first fully gradient-based BO framework based on variational formulations without GP or specific variational family assumptions.

4.4. Phase I: Surrogate Training and Helper Update

At each BO iteration t, the algorithm maintains a replay buffer $\mathcal{H} = \{(\mathbf{x}_{t'}^{(i)}, y_{t'}^{(i)})\}$ of all past observations. A lightweight surrogate $\hat{f}_\psi : \mathcal{D} \to \mathbb{R}$ approximates the conditional expectation $\mathbb{E}[y_t \mid \mathcal{H}]$ by minimizing:

$$\mathcal{L}_{\text{MSE}}(\psi) = \frac{1}{B} \sum_{i=1}^{B} \big(\hat{f}_\psi(\mathbf{x}^{(i)}) - y^{(i)}\big)^2, \quad (23)$$

$$\{(\mathbf{x}^{(i)}, y^{(i)})\} \overset{\text{i.i.d.}}{\sim} \text{Uniform}(\mathcal{H}). \quad (24)$$

Because $\hat{f}_\psi$ is trained exclusively on observed input–output pairs and its gradient flows through the surrogate network (never through f), the black-box condition is never violated. Surrogate predictions $\hat{y}_t^{(i)} = \hat{f}_\psi(\mathbf{x}_t^{(i)})$ replace real observations in the helper's loss, allowing D_θ to update without additional queries to f:

$$L_{D_\theta}(\mathbf{x}_t, \hat{y}_t) = -\mathbb{E}_{P_{\mathbf{x}_t, \hat{y}_t}}[D_\theta(\mathbf{x}_t, \hat{y}_t)] + \log \mathbb{E}_{P_{\mathbf{x}_t} \times P_{\hat{y}_t}}[e^{D_\theta(\mathbf{x}_t, \hat{y}_t)}], \quad (25)$$

with ϕ and ψ frozen. This phase runs for K_a gradient steps.

Remark 4.1. Because $\hat{f}_\psi$ is trained by minimizing the MSE loss over the full replay buffer $\mathcal{H}$, it summarizes all past

observations into a single differentiable model. This allows the helper loss (25) to operate on only the current-batch pair $(\mathbf{x}_t^{(i)}, \hat{y}_t^{(i)})$ rather than variable-length history sequences, removing the need for the helper to process growing observation histories directly and keeping the per-iteration computation cost constant in t.

4.5. Phase II: Action-Net Update

With the helper frozen, the action-net E_ϕ is trained to jointly maximise exploitation of the surrogate's predicted reward and exploration measured by the MI estimate from Phase I. The combined loss (to be minimized) is:

$$L_{E_\phi}(\mathbf{x}_t, \hat{y}_t) = -g(\phi, \theta) = -\hat{y}_t - \sqrt{\beta}\,\mathbb{E}_{P_{\mathbf{x}_t, \hat{y}_t}}[D_\theta(\mathbf{x}_t, \hat{y}_t)] + \sqrt{\beta}\log \mathbb{E}_{P_{\mathbf{x}_t} \times P_{\hat{y}_t}}[e^{D_\theta(\mathbf{x}_t, \hat{y}_t)}], \tag{26}$$

where β controls exploration–exploitation. Minimizing L_{E_ϕ} is equivalent to maximising $\hat{f}_\psi(\mathbf{x}_t) + \sqrt{\beta}\, I_\theta(\mathbf{x}_t, \hat{y}_t)$. Implemented as sample averages:

$$L_{E_\phi} = -\frac{1}{B}\sum_{i=1}^{B} \hat{y}_t^{(i)} + \frac{1}{B}\sum_{i=1}^{B} D_\theta(\mathbf{x}_t^{(i)}, \hat{y}_t^{(i)}) - \log \frac{1}{B^2}\sum_{i,j} e^{D_\theta(\mathbf{x}_t^{(i)}, \hat{y}_t^{(j)})}, \tag{27}$$

where $\hat{y}_t^{(i)} = \hat{f}_\psi(\mathbf{x}_t^{(i)})$. This runs for K_b steps with D_θ and $\hat{f}_\psi$ frozen.

Remark 4.2. VBO-MI uses gradients to update ϕ, θ, and ψ via the variational and MSE losses respectively. These gradients flow entirely through the neural architectures; f remains a strict black box with no gradients required or used.

In summary, our framework has three neural network components: E_ϕ searches the input space and selects $\mathbf{x}_t$; D_θ estimates information gain (exploration); $\hat{f}_\psi$ approximates the conditional expectation of f (exploitation) (see Fig.1). The algorithm is summarized in Algorithm 1.

4.6. Consistency Theorem

Theorem 4.3 (Consistency of VBO-MI)**.** *Let $f : \mathcal{D} \to \mathbb{R}$ be continuous, and let $\widehat{g}(\phi, \theta)$ be the sample-average approximation of $g(\phi, \theta)$ as defined in* (22) *and* (27)*. Assume: (i) finite DV expectations for all $\theta \in \Theta$; (ii) E_ϕ, D_θ, $\hat{f}_\psi$ belong to universal MLP and RNN classes; (iii) the replay buffer $\mathcal{H}$ is a ergodic sequence and $\hat{f}_\psi$ is trained by minimizing sample-average MSE over $\mathcal{H}$; (iv) Φ, Θ, Ψ are compact. Then: (a) $\widehat{g} \to g$ uniformly $\mathbb{P}$-a.s. as $B \to \infty$; (b) $\sup_{\mathbf{x}} |\hat{f}_{\psi^*}(\mathbf{x}) - f(\mathbf{x})| \to 0$ $\mathbb{P}$-a.s. as $|\mathcal{H}| \to \infty$; (c) ϕ^* converges $\mathbb{P}$-a.s. to the optimal action-network parameters and $E_{\phi^*}(\mathbf{s})$ achieves the global maximizer of f over $\mathcal{D}$.*

Proof. See Appendix A. □

Algorithm 1 VBO-MI

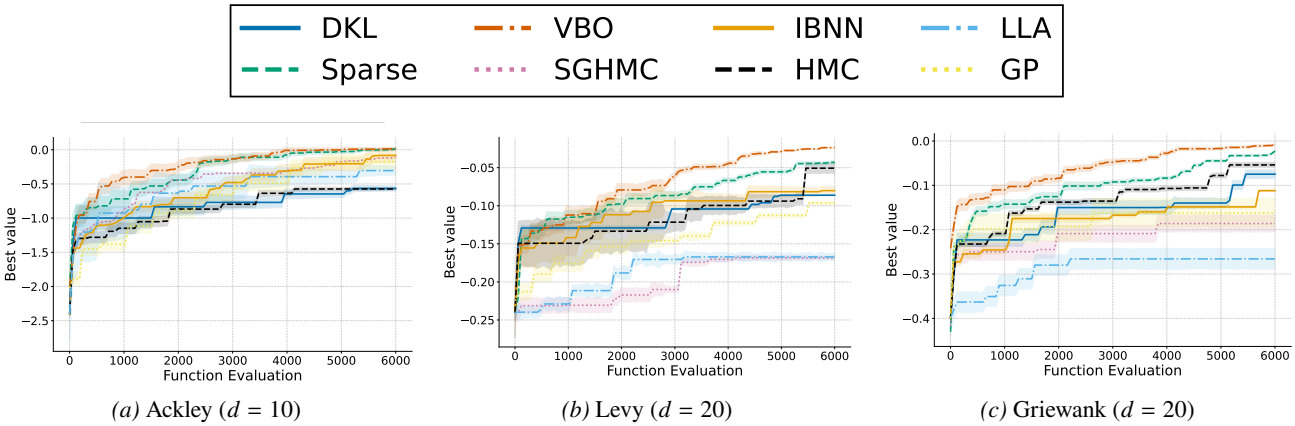
Input: Weights of E_ϕ, D_θ, $\hat{f}_\psi$; $W, T, K_a, K_b, K_s, B, \beta$.
Output: $(\phi^*, \theta^*, \psi^*)$; estimated optimum $\mathbf{x}^*$.
Initialise replay buffer $\mathcal{H} \leftarrow \emptyset$.
for $k = 1, \ldots, W$ **do** ▷ *Warm-up phase*
  Sample $\mathbf{x}^{(i)} = E_\phi(\mathbf{s}_k)$, evaluate $y^{(i)} = f(\mathbf{x}^{(i)}) + \epsilon^{(i)}$, store $(\mathbf{x}^{(i)}, y^{(i)})$ in $\mathcal{H}$.
  Update D_θ on L_{D_θ} with E_ϕ fixed.
end for
for $k = 1, \ldots, 200$ **do** ▷ *Pre-train surrogate $\hat{f}_\psi$*
  Sample mini-batch from $\mathcal{H}$; update $\hat{f}_\psi$ on $\mathcal{L}_{\text{MSE}}$.
end for
for $t = 1, \ldots, T$ **do** ▷ *Main VBO-MI phase*
  Draw $\mathbf{s}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$; sample $\mathbf{x}_t^{(i)} = E_\phi(\mathbf{s}_t)$, $i = 1, \ldots, B$.
  for $k = 1, \ldots, K_a$ **do** ▷ *Update helper D_θ*
    $\hat{y}_t^{(i)} = \hat{f}_\psi(\mathbf{x}_t^{(i)})$; update D_θ on $L_{D_\theta}(\mathbf{x}_t^{(i)}, \hat{y}_t^{(i)})$.
  end for
  for $k = 1, \ldots, K_s$ **do**
    Sample mini-batch $\sim \mathcal{H}$; update $\hat{f}_\psi$ on $\mathcal{L}_{\text{MSE}}$.
  end for
  for $k = 1, \ldots, K_b$ **do** ▷ *Update action-net E_ϕ*
    Compute I_θ via frozen D_θ; update E_ϕ on L_{E_ϕ} (27).
  end for
  Evaluate $y_t^{(i)} = f(\mathbf{x}_t^{(i)}) + \epsilon^{(i)}$; update $\mathcal{H}$. ▷ *single real query per iteration*
  $\mathbf{x}_t^* \leftarrow \arg\max_{(\mathbf{x}, y) \in \mathcal{H}} y$.
end for
Return $(\phi^*, \theta^*, \psi^*)$ and $\mathbf{x}^* = \mathbf{x}_T^*$.

5. Computational Complexity

A key bottleneck in BO is the overhead from surrogate updates, inner-loop acquisition optimization, and real function evaluations. VBO-MI matches the standard BO oracle budget of $O(B)$ queries per iteration: the surrogate $\hat{f}_\psi$, helper D_θ, and action-net E_ϕ are all updated using surrogate predictions or replay-buffer mini-batches, so no oracle calls occur inside the K_a, K_s, or K_b loops. The total per-iteration FLOP cost is $O(K_s B W_s + K_a B H + K_b B W_s)$, which is *constant in T* — qualitatively different from all GP- and BNN-based baselines. By eliminating the inner-loop acquisition optimizer via the amortised action-net, acquisition FLOPs reduce from $O(N_{\text{starts}} \cdot N_{\text{steps}} \cdot S \cdot W)$ to $O(K_b \cdot W_e)$, yielding a near $10^2\times$ reduction. Table 1 summarises the comparison; the complexity figure is in Appendix C.

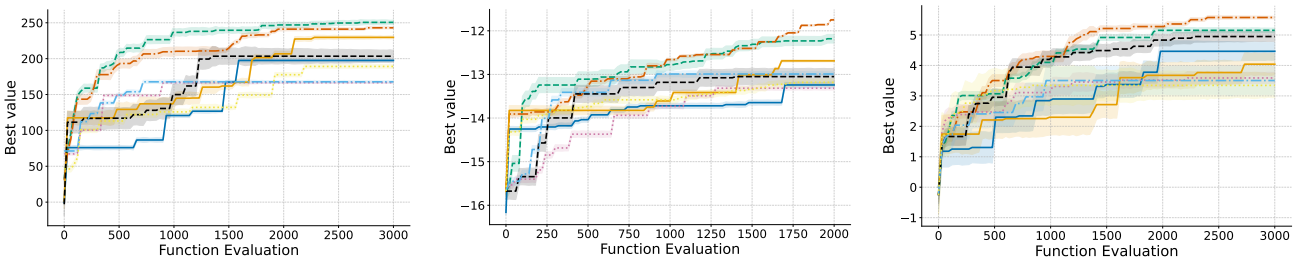
6. Numerical Evaluation

We present numerical evaluations of VBO-MI and compare against: LLA (Immer et al., 2021; Li et al., 2024), HMC (Iz-



(a) Ackley ($d = 10$) *(b)* Levy ($d = 20$) *(c)* Griewank ($d = 20$)

Figure 2. Synthetic benchmarks. Global optimum is 0 (dashed line). Mean ± 1 std, 5 trials.



(a) Lunar Lander ($d = 12$) *(b)* Pest Control ($d = 25$) *(c)* Rover Trajectory ($d = 60$)

Figure 3. Real-world benchmarks. Mean ± 1 std, 5 trials.

Table 1. FLOP and oracle complexity per BO iteration. T: observations; W: BNN weights; W_s/W_e: surrogate/action-net weights; H: LSTM dim; S: HMC samples; L: leapfrog steps.

Method	Surrogate Upd.	Acq. Opt.	f/iter	Scaling
GP	$O(T^3)$	$NO(T^2)$	B	Cubic
HMC-BNN	$O(SLTW)$	$NO(SW)$	B	Hi. Lin.
DKL	$O(TW+T^3)$	$NO(T^2)$	B	Cubic
LLA	$O(TW+W^3)$	$NO(W^2)$	B	Linear
VBO-MI	$O(K_sBW_s+K_aBH)$	$O(K_bW_e)$	**B**	**Const.**

mailov et al., 2021; Li et al., 2024), SGHMC (Chen et al., 2014; Li et al., 2024), DKL (Wilson et al., 2016; Li et al., 2024), IBNN (Lee et al., 2018; Li et al., 2024), GP (Snoek et al., 2012; Li et al., 2024), and Focal Sparse GP (Wei et al., 2024).

Implementation. VBO-MI uses batch size $B = 30/60$ for real-world/synthetic tasks, learning rate 2×10^{-3} (Adam) for all networks, $K_a = 5$, $K_b = 10$, $K_s = 10$, and $W = 30$ warm-up iterations. Moreover, we set $\beta = 10$ in Ackley, and lunar lander, and $\beta = 20$ in other tasks. Baselines follow (Li et al., 2024). All results average over 5 seeds; shaded regions show ±1 std. Architecture details are in Appendix B.

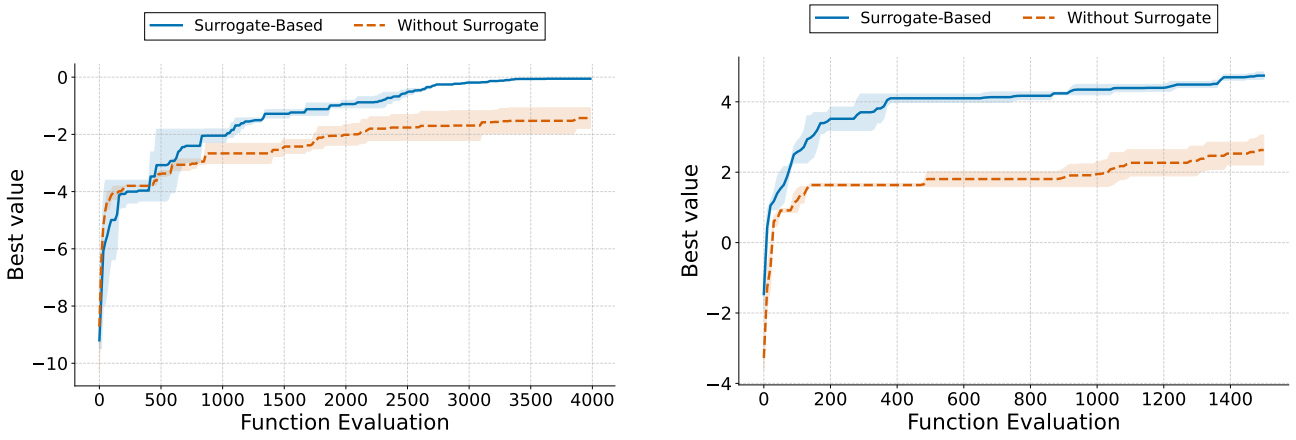
6.1. Synthetic Functions

We evaluate on three benchmark functions: Ackley, Levy, and Griewank (Li et al., 2024; Eriksson et al., 2019) — all highly non-convex with many local optima and global minimum at zero. We optimize $-f$ so the global maximum is 0.

Ackley ($d = 10$, Fig. 2a). Ackley is deceptive due to its near-flat outer region and sharp global basin, making exploration critical. VBO-MI achieves competitive performance, reaching higher final average rewards than BNN baselines, and surpasses Focal Sparse GP as iterations progress.

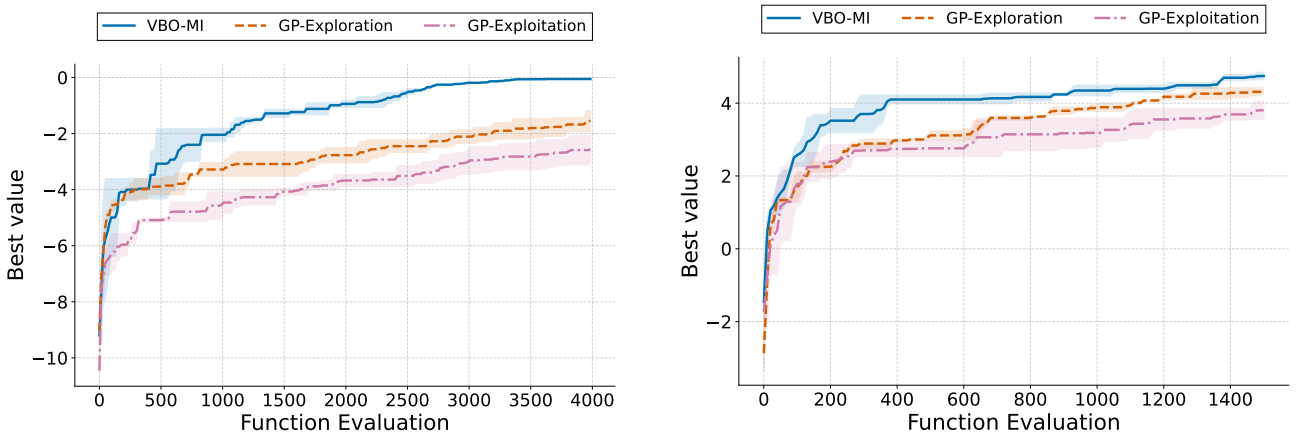
Levy ($d = 20$, Fig. 2b). Levy's difficulty scales with dimension. VBO-MI demonstrates faster initial convergence and higher final reward compared to all baselines, including GP and IBNN.

Griewank ($d = 20$, Fig. 2c). Unlike Ackley and Levy, Griewank is *non-separable* due to its product term, making stationary kernels less effective. VBO-MI achieves the highest final reward, which we attribute to its non-parametric variational exploration signal.



(a) Ackley ($d = 10$) *(b)* Rover ($d = 60$)

Figure 4. Ablation I: surrogate vs. direct queries. Mean ± 1 std, 3 trials.



(a) Ackley ($d = 10$) *(b)* Rover ($d = 60$)

Figure 5. Ablation II: acquisition term contribution. Full VBO-MI vs. GP-exploration and GP-exploitation hybrids. Mean ± 1 std, 3 trials.

6.2. Real-World Tasks

We evaluate on three tasks of increasing dimensionality (Li et al., 2024; Deng et al., 2022).

Lunar Lander ($d = 12$, Fig. 3a). The goal is to optimise a controller for OpenAI Gym `LunarLander-v2`, maximising average return over 50 random environments. VBO-MI achieves the highest final reward with a clear margin over BNN baselines.

Pest Control ($d = 25$, categorical, Fig. 3b). 25 categorical variables, 5 values each (Eriksson et al., 2019). We apply one-hot encoding (Garrido-Merchán & Hernández-Lobato, 2020). VBO-MI achieves superior performance, with the advantage growing with more iterations.

Rover Trajectory ($d = 60$, Fig. 3c). 30 waypoints in a 2D environment (60 total dimensions) (Eriksson et al., 2019). VBO-MI achieves comparable or superior final reward while requiring significantly fewer oracle evaluations per iteration.

We isolate the contribution of (I) the surrogate network $\hat{f}_\psi$ and (II) each term of the acquisition function L_{E_ϕ}, evaluated on Ackley ($d = 10$) and Rover ($d = 60$) with all other hyperparameters fixed.

Ablation I: Surrogate network (Fig. 4). We compare surrogate-assisted VBO-MI against a variant that queries f directly inside the K_a and K_b loops. On Ackley the surrogate already provides a higher final reward in early iterations. On Rover the advantage is substantially more pronounced: the no-surrogate variant plateaus at a noticeably lower reward. With default hyperparameters ($K_a = 5$, $K_b = 10$), the surrogate reduces oracle queries from $(K_a + K_b + 1) \times B$

to B per iteration — a 16× reduction — at no performance cost.

Ablation II: Acquisition terms (Fig. 5). We replace the exploration term $\sqrt{\beta}\, I_\theta$ with GP posterior variance $\sqrt{\beta}\, \sigma^2_{\text{GP}}$ (first hybrid), or replace the exploitation term $\hat{f}_\psi$ with GP posterior mean μ_{GP} (second hybrid). Full VBO-MI outperforms both hybrids on both tasks. Replacing the exploitation term produces a larger degradation, particularly on Rover, confirming that $\hat{f}_\psi$ plays a more critical role than the exploration signal alone. Both terms are necessary; neither can be substituted without measurable cost. We discuss other ablation studies in the appendix on the $beta$ and batch size parameters in the appendix C.3.

7. Conclusion

We presented VBO-MI, a fully gradient-based Bayesian optimization framework that eliminates GP priors and specific variational family assumptions. By combining a variational MI estimator for exploration with a data-driven surrogate for exploitation, all three networks in VBO-MI are trained via backpropagation while treating f as a strict black box. The surrogate reduces oracle queries to one batch per iteration (a 16× reduction), and the amortised action-net replaces the inner-loop acquisition optimizer (a $10^2\times$ FLOP reduction). We established consistency guarantees and demonstrated competitive or superior performance on diverse benchmarks spanning $d = 10$ to $d = 60$. Future work includes extending VBO-MI to multi-objective settings and investigating tighter MI estimators for sparse reward landscapes.

Impact Statement

This work proposes a general-purpose framework for Bayesian optimization based on variational mutual information estimation and a neural actor-critic structure. As a methodological contribution, it is intended to improve optimization performance across a wide range of tasks without targeting any specific application domain. Bayesian optimization techniques are used in engineering design, scientific experimentation, and machine learning model tuning, with potential societal consequences that we do not feel must be specifically highlighted here.

References

Aguiar, M., Das, A., and Johansson, K. H. Universal approximation of flows of control systems by recurrent neural networks. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pp. 2320–2327, 2023. doi: 10.1109/CDC49753.2023.10383457.

Algoet, P. H. and Cover, T. M. A Sandwich Proof of the Shannon-McMillan-Breiman Theorem. *The Annals of Probability*, 16(2):899 – 909, 1988. doi: 10.1214/aop/1176991794.

Belghazi, M. I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, D. Mutual information neural estimation. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 531–540. PMLR, 10–15 Jul 2018.

Breiman, L. The Individual Ergodic Theorem of Information Theory. *The Annals of Mathematical Statistics*, 28(3):809 – 811, 1957. doi: 10.1214/aoms/1177706899.

Chen, T., Fox, E., and Guestrin, C. Stochastic gradient hamiltonian monte carlo. In Xing, E. P. and Jebara, T. (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1683–1691, Bejing, China, 22–24 Jun 2014. PMLR.

Chen, X. and White, H. Improved rates and asymptotic normality for nonparametric neural network estimators. *IEEE Transactions on Information Theory*, 45(2):682–691, 1999. doi: 10.1109/18.749011.

Cheng, N., Papenmeier, L., Becker, S., and Nardi, L. A unified framework for entropy search and expected improvement in bayesian optimization. In *Forty-second International Conference on Machine Learning*, 2025.

Cover, T. and Thomas, J. *Elements of Information Theory*. Wiley, 2012. ISBN 9781118585771.

Cox, D. and John, S. A statistical method for global optimization. In *[Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1241–1246 vol.2, 1992. doi: 10.1109/ICSMC.1992.271617.

Deng, Z., Zhou, F., and Zhu, J. Accelerated linearized laplace approximation for bayesian deep learning. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 2695–2708. Curran Associates, Inc., 2022.

Donsker, M. D. and Varadhan, S. R. S. Asymptotic evaluation of certain markov process expectations for large time, i. *Communications on Pure and Applied Mathematics*, 28(1):1–47, 1975. doi: https://doi.org/10.1002/cpa.3160280102.

Eriksson, D., Pearce, M., Gardner, J., Turner, R. D., and Poloczek, M. Scalable global optimization via local bayesian optimization. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Frazier, P. I. A Tutorial on Bayesian Optimization. *arXiv e-prints*, art. arXiv:1807.02811, July 2018. doi: 10.48550/arXiv.1807.02811.

Garnett, R. *Bayesian Optimization*. Cambridge University Press, 2023.

Garrido-Merchán, E. C. and Hernández-Lobato, D. Dealing with categorical and integer-valued variables in bayesian optimization with gaussian processes. *Neurocomputing*, 380:20–35, 2020. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2019.11.004.

Harrison, J. et al. Variational last layer Bayesian neural networks. In *International Conference on Learning Representations (ICLR)*, 2024.

Hennig, P. and Schuler, C. J. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(57):1809–1837, 2012.

Hennig, P., Osborne, M. A., and Kersting, H. P. *Probabilistic Numerics: Computation as Machine Learning*. Cambridge University Press, 2022.

Hernandez-Lobato, J. M., Gelbart, M., Hoffman, M., Adams, R., and Ghahramani, Z. Predictive entropy search for bayesian optimization with unknown constraints. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1699–1707, Lille, France, 07–09 Jul 2015. PMLR.

Hornik, K., Stinchcombe, M. B., and White, H. L. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

Hvarfner, C., Hutter, F., and Nardi, L. Joint entropy search for maximally-informed bayesian optimization. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 11494–11506. Curran Associates, Inc., 2022.

Immer, A., Korzepa, M., and Bauer, M. Improving predictions of bayesian neural nets via local linearization. In Banerjee, A. and Fukumizu, K. (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 703–711. PMLR, 13–15 Apr 2021.

Ishikura, M. and Karasuyama, M. Pareto-frontier entropy search with variational lower bound maximization. In Singh, A., Fazel, M., Hsu, D., Lacoste-Julien, S., Berkenkamp, F., Maharaj, T., Wagstaff, K., and Zhu, J. (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 26490–26522. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/ishikura25a.html.

Izmailov, P., Vikram, S., Hoffman, M. D., and Wilson, A. G. G. What are bayesian neural network posteriors really like? In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4629–4640. PMLR, 18–24 Jul 2021.

Jimenez, F. and Katzfuss, M. Scalable bayesian optimization using vecchia approximations of gaussian processes. In Ruiz, F., Dy, J., and van de Meent, J.-W. (eds.), *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 1492–1512. PMLR, 25–27 Apr 2023. URL https://proceedings.mlr.press/v206/jimenez23a.html.

Lee, J., Sohl-dickstein, J., Pennington, J., Novak, R., Schoenholz, S., and Bahri, Y. Deep neural networks as gaussian processes. In *International Conference on Learning Representations*, 2018.

Li, Y. L., Rudner, T. G. J., and Wilson, A. G. A study of bayesian neural network surrogates for bayesian optimization. In *The Twelfth International Conference on Learning Representations*, 2024.

McIntire, M., Ratner, D., and Ermon, S. Sparse gaussian processes for bayesian optimization. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'16, pp. 517–526, Arlington, Virginia, USA, 2016. AUAI Press. ISBN 9780996643115.

Müller, S., Hollmann, N., Arango, S. P., Grabocka, J., and Hutter, F. PFNs: Prior-fitted networks for efficient Bayesian optimization. In *International Conference on Learning Representations (ICLR)*, 2023.

Rudner, T. G. J., Bickford Smith, F., Feng, Q., Teh, Y. W., and Gal, Y. Continual learning via sequential function-space variational inference. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18871–18887. PMLR, 17–23 Jul 2022.

Snoek, J., Larochelle, H., and Adams, R. P. Practical bayesian optimization of machine learning algorithms. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'12, pp. 2951–2959, Red Hook, NY, USA, 2012. Curran Associates Inc.

Song, J. and Ermon, S. Understanding the limitations of variational mutual information estimators. In *International Conference on Learning Representations*, 2020.

Springenberg, J. T., Klein, A., Falkner, S., and Hutter, F. Bayesian optimization with robust bayesian neural networks. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*, pp. 1015–1022. Omnipress, 2010.

Tsur, D., Aharoni, Z., Goldfeld, Z., and Permuter, H. H. Neural estimation and optimization of directed information over continuous spaces. *IEEE Transactions on Information Theory*, 69:4777–4798, 2022.

Wang, Z. and Jegelka, S. Max-value entropy search for efficient Bayesian optimization. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3627–3635. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/wang17e.html.

Wei, Y., Zhuang, V., Soedarmadji, S., and Sui, Y. Scalable bayesian optimization via focalized sparse gaussian processes. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 120443–120467. Curran Associates, Inc., 2024.

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. Deep kernel learning. In Gretton, A. and Robert, C. C. (eds.), *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 370–378, Cadiz, Spain, 09–11 May 2016. PMLR.

A. Consistency Analysis of Losses

First, to demonstrate that the expected values can be replaced with averages using tools from the generalized Birkhoff Theorem (Breiman, 1957) and generalized AEP theorem (Algoet & Cover, 1988), this is not trivial since the selected actions are not independent; we cannot directly use the law of large numbers in this case (LLN). In our proofs, we follow a similar approach to other neural information estimation methods specifically, (Belghazi et al., 2018), and (Tsur et al., 2022). Second, to show that NNs can effectively approximate the underlying functions to compute these estimations using the universal approximation theorem (Hornik et al., 1989). This allows us to establish the consistency of our results. We focus on proving consistency for L_{E_ϕ}, noting that the same approach can be extended to L_{D_θ}. We do not go through all the details of proof for L_{D_θ}.

1. Estimation from samples: In the first step, we aim to demonstrate the consistency of estimating the actual loss. Specifically, we aim to show our loss:

$$\mathbb{E}[y_t \mid y_{t-1}, .., y_1, \mathbf{x}_{t-1}, .., \mathbf{x}_1] + \tag{28}$$

$$\sqrt{\beta}\big[- \mathbb{E}_{P_{\mathbf{x}_t, \mathbf{Y}_t}}[D_{\theta_2}(\mathbf{x}_t, y_t)] + \log(\mathbb{E}_{P_{\mathbf{x}_t} \times P_{y_t}}[e^{D_{\theta_2}(\mathbf{x}_t, y_t)}])\big], \tag{29}$$

can be estimated using samples:

$$L_{E_\phi} = \frac{1}{B}\sum_{i=1}^{B} y_t^{(i)} - \frac{1}{B}\sum_{i=1}^{B} D_\theta(\mathbf{x}^{(i)}, y^{(i)}) \tag{30}$$

$$+ \log \frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B} e^{D_\theta(\mathbf{x}^{(i)}, y^{(j)})}, \tag{31}$$

where $y_t^{(i)} = f(\mathbf{x}_t^{(i)}) + \epsilon_t^{(i)}$[4]

In our analysis, we separately consider each term in the above formulation. To establish consistency. To that end, we rely on the following two key theorems to prove our results:

Theorem A.1. *Generalized AEP(Generalized Asymptotic Equipartition Property for Markov Approximation). (Algoet & Cover, 1988) Let $\mathcal{H}$ be a standard Borel space, and consider the infinite product space $\mathcal{H}_0^\infty = \prod_{t=0}^\infty \mathcal{H}$, equipped with its canonical Borel σ-algebra $(\Omega, \mathcal{F})$. Let $\mathbb{P}$ represent the probability law of a stationary, ergodic stochastic process $\{X_t\}$ defined over $(\Omega, \mathcal{F})$, and let $\mathbb{M}$ denote a finite-order Markov measure that has a stationary transition mechanism. Suppose that for each finite n, the n-dimensional marginal of $\mathbb{P}$ is absolutely continuous with respect to that of $\mathbb{M}$, with associated joint density $p(x_0, \ldots, x_{n-1})$.*

Then, the generalized asymptotic equipartition property asserts that:

$$\frac{1}{n}\log p(X_0, \ldots, X_{n-1}) \xrightarrow{n\to\infty} \lim_{k\to\infty} \mathbb{E}\left[\log p(X_k \mid X_{k-1}, ., X_0)\right]. \tag{32}$$

Generalized AEP is particularly useful when we do not have the i.i.d. assumption, and when we are dealing with a stationary ergodic process. By applying the generalized

[4] $s_t^{(i)}$ refers to element i of random seed vector at time step t in Alg.1

AEP for the first two terms in L_{E_ϕ}, we conclude:

$$\mathbb{E}[y_t \mid y_{t-1}, .., y_1, \mathbf{x}_{t-1}, .., \mathbf{x}_1] \to \frac{1}{B}\sum_{i=1}^{B} \hat{y}_t^{(i)}, \tag{33}$$

$$\mathbb{E}_{P_{\mathbf{x}_t, y_t}}[D_\theta(\mathbf{x}_t, y_t)] \to \frac{1}{B}\sum_{i=1}^{B} D_\theta(\mathbf{x}^{(i)}, y^{(i)}). \tag{34}$$

Where the first term follows from (33) follows from consistency result for MSE estimation of neural networks (Chen & White, 1999), and second term directly from Generalized AEP. For sample estimation of the other term in L_{E_ϕ}, we need another theorem, as noted in other works in the literature (such as (Tsur et al., 2022)). As we are dealing with a function of expectation, we invoke the Generalized Birkhoff Theorem (Breiman, 1957) to estimate

$$\log(\mathbb{E}_{P_{\mathbf{x}_t} \times P_{y_t}}[e^{D_\theta(\mathbf{x}_t, y_t)}]). \tag{35}$$

Theorem A.2. ***Birkhoff's Ergodic Theorem. (Breiman, 1957)***

Let $(X, \mathcal{F}, \mu)$ be a probability space, and suppose $T : X \to X$ is a measure-preserving transformation, meaning that for all $A \in \mathcal{F}$,

$$\mu(T^{-1}(A)) = \mu(A).$$

Assume $f : X \to \mathbb{R}$ is integrable with respect to μ (i.e., $f \in L^1(\mu)$).

Then the following statements hold:

1. ***Almost Sure Convergence:*** *For μ-almost every $\mathbf{x} \in X$, the time average of f along the orbit of $\mathbf{x}$ under T converges:*

$$\lim_{n\to\infty} \frac{1}{n}\sum_{k=0}^{n-1} f(T^k(\mathbf{x})) = f^*(\mathbf{x}),$$

where f^* is a T-invariant function, meaning $f^*(T(\mathbf{x})) = f^*(\mathbf{x})$.

2. ***In the Ergodic Case:*** *If T is ergodic (i.e., the only T-invariant sets have measure 0 or 1), then $f^*(\mathbf{x})$ is constant μ-almost everywhere and equal to the integral of f:*

$$f^*(\mathbf{x}) = \int f \, d\mu \quad \text{for } \mu\text{-almost every } \mathbf{x}.$$

By the assumption of Ergodicity of input data sequences, the T transform in this case translates into the condition $\mathbb{P}(A) = \mathbb{P}(T^{-1}(A))$ for any $A \in \mathcal{F}$, where in this case we use a time-shift. By applying this to the first term, we conclude:

$$\log(\mathbb{E}_{P_{\mathbf{x}_t} \times P_{y_t}}[e^{D_\theta(\mathbf{x}_t, y_t)}]) \overset{n\to\infty}{\underset{\mathbb{P}-as}{\to}} \log \frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B} e^{D_\theta(\mathbf{x}^{(i)}, y^{(j)})}. \tag{36}$$

By putting together all terms, we conclude:

$$\frac{1}{B}\sum_{i=1}^{B} \hat{y}_t^{(i)} + \frac{1}{B}\sum_{i=1}^{B} D_\theta(\mathbf{x}^{(i)}, y^{(i)}) \tag{37}$$

$$- \log \frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B} e^{D_\theta(\mathbf{x}^{(i)}, y^{(j)})} \overset{n\to\infty}{\to} \mathbb{E}[y_t \mid y_{t-1}, ., y_1, \mathbf{x}_{t-1}, ., \mathbf{x}_1]$$

$$+ \mathbb{E}_{P_{\mathbf{x},\mathbf{Y}}}[D_\theta(\mathbf{x}, y)] - \log \mathbb{E}_{P_{\mathbf{x}\times y}}[e^{D_\theta(\mathbf{x}, y)}]. \tag{38}$$

This completes the estimation step from samples.

2. Approximation Step: Up to now, we have provided the proof for the consistent estimation of our loss. In this stage, we aim to show the possibility to approximate functional space with the space of Feed-forward network (FNN), $E_\phi(\mathbf{s}) = \mathbf{x}_t, f(\mathbf{x}_t)$, and RNN networks, D_θ. we apply universal approximation theorem for FNN [Theorem.2.2 (Hornik et al., 1989)], and RNNs (Aguiar et al., 2023).

Our proof in this section is a generalization of (Belghazi et al., 2018; Tsur et al., 2022), as our loss has an additional term, and both feed-forward and RNN neural networks.

First, consider the first network, E_ϕ. The action-net E_ϕ does not approximate f directly — since f is a strict black box, it is never accessible pointwise to E_ϕ. Instead, E_ϕ approximates the *optimal policy map* $\pi^* : \mathcal{S} \to \mathcal{D}$, defined as

$$\pi^*(\mathbf{s}) = \arg\max_{\mathbf{x}\in\mathcal{D}} \; g_\theta(\mathbf{x}), \tag{39}$$

where g_θ is the acquisition objective with the helper network parameters θ fixed, and $\mathcal{S}$ is the seed space from which $\mathbf{s}_t$ is drawn. Since $\mathcal{D}$ is compact and f is continuous by assumption, g_θ is continuous and π^* is well-defined. As $\pi^* : \mathcal{S} \to \mathcal{D}$ is a continuous map between compact subsets of Euclidean space, the Universal Approximation Theorem (Hornik et al., 1989) guarantees that for any $\epsilon \geq 0$, there exists a feedforward network $E_{\hat{\phi}}$ with sufficient capacity such that:

$$\sup_{\mathbf{s}\in\mathcal{S}} \|E_{\hat{\phi}}(\mathbf{s}) - \pi^*(\mathbf{s})\| \leq \epsilon. \tag{40}$$

Furthermore, since f is uniformly continuous on the compact domain $\mathcal{D}$, there exists a modulus of continuity L_f such that:

$$\left|f(E_{\hat{\phi}}(\mathbf{s})) - f(\pi^*(\mathbf{s}))\right| \leq L_f \cdot \epsilon. \tag{41}$$

Since $\pi^*(\mathbf{s})$ achieves $\max_{\mathbf{x}\in\mathcal{D}} f(\mathbf{x})$ by definition, (41) establishes that the action-net output approaches the global maximum of f as $\epsilon \to 0$, completing the approximation argument for E_ϕ.

Now our main focus is on proving that it is possible for our "helper" network, which consists of RNNs, to approximate the functional space to find information gain between selected actions and observations. Specifically, we want to

show there is an RNN with parameters θ such that:

$$| I(\mathbf{x};\mathbf{y}) - I_\Theta(\mathbf{x},\mathbf{y}) | \leq \eta, \tag{42}$$

where $I_\Theta(\mathbf{x},\mathbf{y}) = \max_{\theta\in\Theta}(\mathbb{E}_{P_{x,y}}[T_\theta(\mathbf{x},\mathbf{y})] - \log(\mathbb{E}_{P_{\mathbf{x}\times\mathbf{y}}}[e^{T_\theta(\mathbf{x},y)}]))$.

Based on DV representation (Donsker & Varadhan, 1975)

Let: $T^*(\mathbf{x},\mathbf{y}) = \log\frac{p(\mathbf{x},\mathbf{y})}{p(\mathbf{x})p(\mathbf{y})}$, where be the discriminator function for estimating information gain. Based on (Aguiar et al., 2023) there exist a RNN, for any $\epsilon \geq 0$ such that:

$$\sup_{\{\mathbf{x}\},\{y\}} | T^*(\mathbf{x},\mathbf{y}) - \hat{T}(\mathbf{x},\mathbf{y}) | \leq \epsilon. \tag{43}$$

Since (43) implies $|T^*(\mathbf{x},\mathbf{y}) - \hat{T}(\mathbf{x},\mathbf{Y})| \leq \epsilon \quad \forall(\mathbf{X},\mathbf{y})$, By applying expectation with respect to $p(\mathbf{x},\mathbf{y})$ we conclude:

$$\left|\mathbb{E}_{p(\mathbf{x},\mathbf{y})}[T^*(\mathbf{x},\mathbf{y}) - \hat{T}(\mathbf{x},\mathbf{y})]\right| \leq \epsilon. \tag{44}$$

Thus, the first term DV representation is satisfied.

Now we turn to the second term in DV representation:

[5]. Now define:

$$A = \mathbb{E}_{p(\mathbf{x})p(\mathbf{y})}[e^{T^*(\mathbf{x},\mathbf{y})}], \quad B = \mathbb{E}_{p(\mathbf{x})p(\mathbf{y})}[e^{\hat{T}(\mathbf{x},\mathbf{y})}]. \tag{45}$$

Now it is easy to see that by using the triangular inequality, and Taylor expansion, we can find the following lower bound

$$|\log A - \log B| \leq \frac{|A-B|}{\min(A,B)}. \tag{46}$$

Now to find the bound on $|\log A - \log B|$ we bound each term, $A - B$, and $\min(A,B)$:

If the T^* is bounded by M, e.g, $T^* \leq M$ we have:

$$|T^*(\mathbf{x},\mathbf{y}) - \hat{T}(\mathbf{x},\mathbf{y})| \leq \epsilon \implies \left|e^{T^*(\mathbf{x},\mathbf{y})} - e^{\hat{T}(\mathbf{x},\mathbf{y})}\right| \leq e^M \cdot \epsilon. \tag{47}$$

Now we turn to bound $\min(A-B)$.

If we assume that T^* is bounded from below, e.g, $T^* \geq L$, then we conclude:

$$A \geq e^L, \quad B \geq C.e^L, \tag{48}$$

where $C = e^{-|\epsilon|}$. So, as exp is monotonic, we have: $\min(A,B) \geq C.e^L$.

Now, substitute the bounds for $|A-B|$ and $\min(A,B)$ in (46) to conclude:

$$|\log A - \log B| \leq \frac{e^M \cdot \epsilon}{e^{L-\epsilon}} = e^{M-L+\epsilon} \cdot \epsilon. \tag{49}$$

[5]We apply the universal approximation in the last step as a tool to prove the accurate estimation of $\mathbf{x}_t$, as an output of E_ϕ, and as result $y_t = f(\mathbf{x}_t) + \epsilon_t$

Since e^{M-L} is bounded, as $\epsilon \to 0$:

$$|\log A - \log B| \leq O(\epsilon) \to 0. \tag{50}$$

By combining (44), and (50) we conclude (42). Now by combining (40), and (42) we conclude that estimation and approximation of L_{E_ϕ} is consistent. As L_{D_θ} has similar terms to L_{E_ϕ}, the proof is similar and we omit the details here.

B. Experimental Setups

B.1. VBO-MI Implementation Details

Network architectures. The *action-net* E_ϕ: three fully connected layers, width 128, `ReLU`, `Tanh` output. The *helper* D_θ: pairwise LSTM-MLP critic — two single-layer LSTMs (hidden size 64) whose outputs are concatenated and fed into a three-layer MLP with `ReLU`. The *surrogate* $\hat{f}_\psi$: two-layer MLP, `ReLU`.

Hyperparameters. Adam optimizer. Learning rate: 2×10^{-3} for E_ϕ, D_θ; 1×10^{-3} for $\hat{f}_\psi$. Batch size $B = 30$. Warm-up $W = 30$ iterations; 200-step surrogate pre-training. $K_a = 5$, $K_s = 10$, $K_b = 10$.

BNN baselines. Three-layer MLP, `ReLU`, width 128, following (Li et al., 2024). we apply a grid search over method-specific hyperparameters to find the best performance; Moreover, we use a expected improvement (EI) acquisition function in all methods as reported by(Li et al., 2024).

B.2. Synthetic Benchmarks

Ackley ($d = 10$, domain $[-32.768, 32.768]^{10}$): $a = 20$, $b = 0.2$, $c = 2\pi$; global minimum $f(\mathbf{0}) = 0$.

Levy ($d = 20$, domain $[-10, 10]^{20}$): sinusoidal multimodal function; global minimum $f(\mathbf{1}) = 0$.

Griewank ($d = 20$, domain $[-600, 600]^{20}$): $f(\mathbf{x}) = \sum_i x_i^2/4000 - \prod_i \cos(x_i/\sqrt{i}) + 1$; global minimum $f(\mathbf{0}) = 0$.

B.3. Real-World Benchmarks

Lunar Lander ($d = 12$): `LunarLander-v2` controller optimisation; reward averaged over 50 random environments.

Pest Control ($d = 25$, categorical): 25 categorical variables, 5 values each (Eriksson et al., 2019); one-hot encoding (Garrido-Merchán & Hernández-Lobato, 2020).

Rover Trajectory ($d = 60$): 30 waypoints in a 2D environment with obstacles (Eriksson et al., 2019).

Complexity parameters.

Table 2. Parameters used for FLOP analysis (Fig. 6, and Sec. 5).

Param	Value	Description
W	10,000	BNN parameter count
W_s	≈ 500	Surrogate MLP params
W_e	≈ 400	Action-net params
H	64	Helper LSTM hidden dim
S	50	HMC posterior samples
L	20	Leapfrog steps
B	256	Batch size
K_a, K_b, K_s	5, 10, 10	Update steps
N_{starts}	10	Acq. restarts
N_{steps}	50	Iters per restart

C. Additional Results

C.1. Complexity Figure

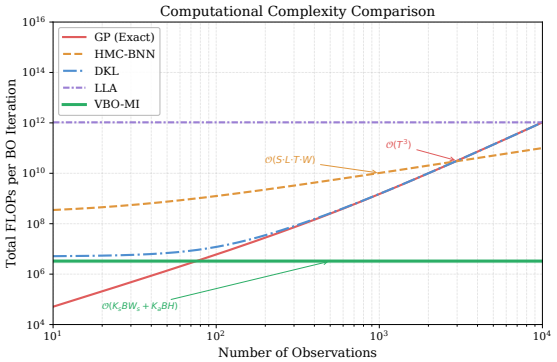


Figure 6. Theoretical FLOP complexity per BO iteration.

C.2. Acquisition Term Effect on Other Benchmarks

Fig. 7 shows the effect of replacing the exploration term $\sqrt{\beta}\, I_\theta$ with the GP posterior variance, and the exploitation term $\hat{f}_\psi$ with the GP posterior mean, evaluated on four additional real-world tasks used in earlier versions of the benchmark suite.

C.3. Batch Size and Beta Sensitivity

Figs. 8 and 9 report the sensitivity of VBO-MI to the batch size B and the exploration parameter β, evaluated on PDE optimization and Lunar Lander. VBO-MI shows consistent performance across a range of values, confirming robustness to these hyperparameters.

C.4. Final Reward and Variance Tables

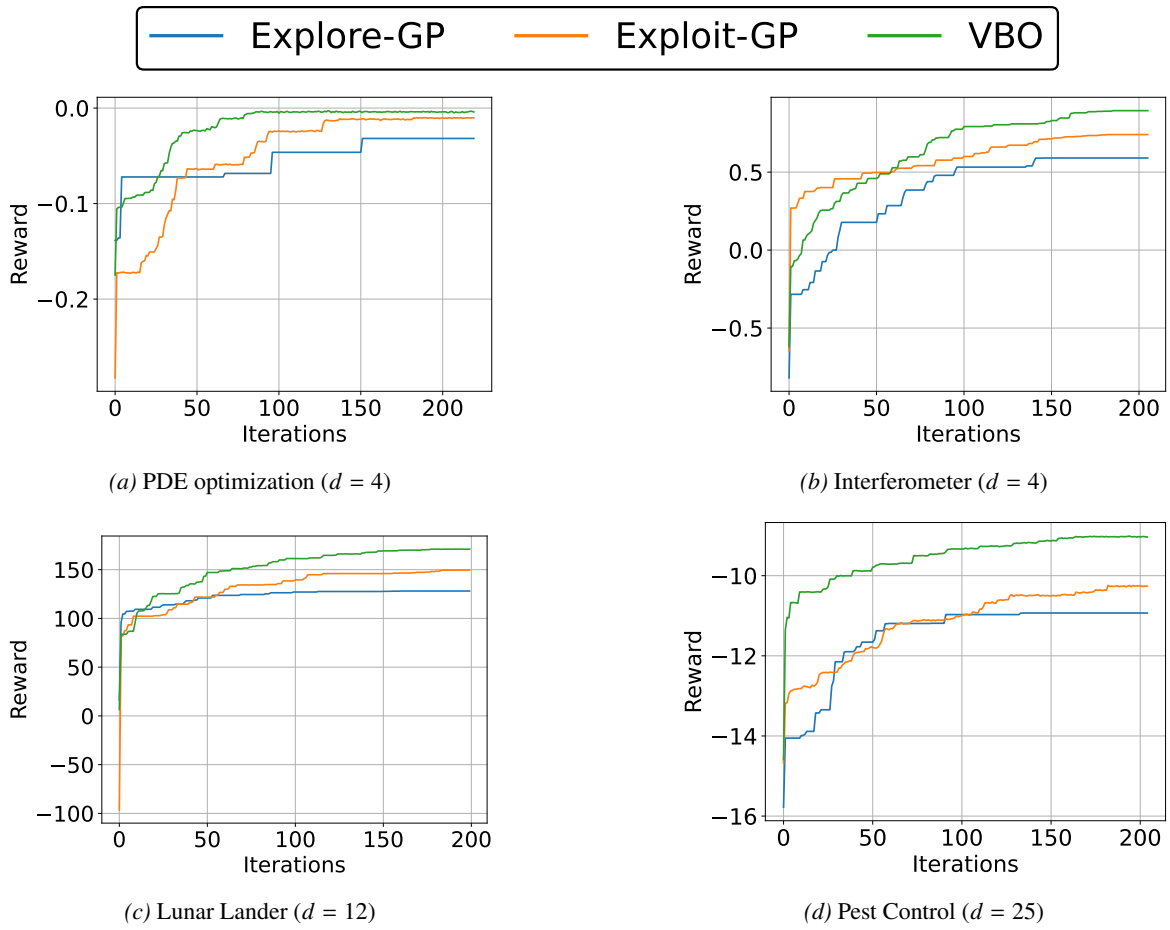
Tables 3–4 report the average reward and variance over the last 20 iterations for each method on four tasks, including the two acquisition-term ablation variants.

Table 3. PDE — Average over last 20 iterations.

Method	**Reward**	**Var.** ($\times 10^{-5}$)
VBO-MI w/ GP expl.	−0.0293	0.010
VBO-MI w/ GP expt.	−0.011	0.003
DKL	−0.0289	0.001
Sparse	−0.0175	0.032
HMC	−0.0139	0.001
GP	−0.0349	0.030
VBO-MI	**−0.0071**	0.007
IBNN	−0.018	0.001
LLA	−0.031	0.001
SGHMC	−0.051	0.001

Table 4. Interferometer — Average over last 20 iterations.

Method	**Reward**	**Var.** ($\times 10^{-1}$)
VBO-MI w/ GP expl.	0.379	0.073
VBO-MI w/ GP expt.	0.586	0.023
DKL	0.619	0.001
Sparse	0.730	0.004
HMC	0.668	0.000
GP	0.567	0.003
VBO-MI	**0.890**	0.001
IBNN	0.710	0.000
LLA	0.664	0.002
SGHMC	0.590	0.000



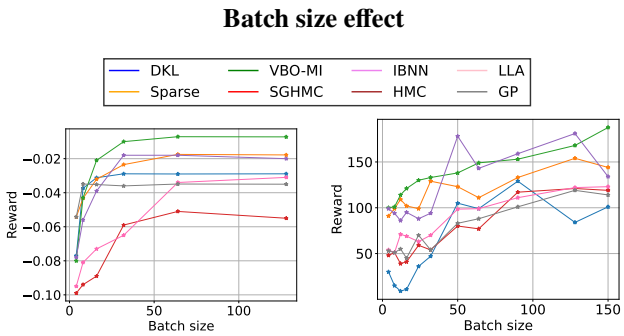
(a) PDE optimization ($d = 4$)

(b) Interferometer ($d = 4$)

(c) Lunar Lander ($d = 12$)

(d) Pest Control ($d = 25$)

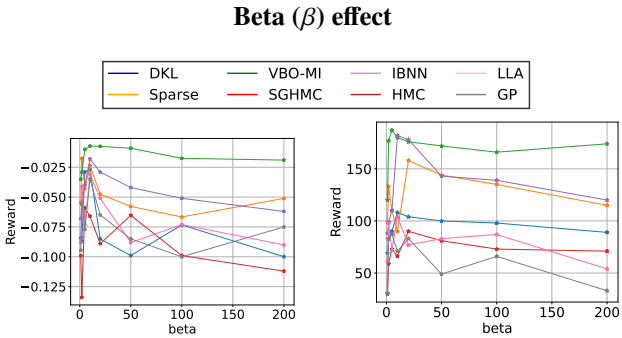
Figure 7. Impact of replacing exploration and exploitation terms in Eq. (26) with GP posterior components (additional real-world tasks).



(a) PDE

(b) Lunar Lander

Figure 8. VBO-MI vs. baselines at different batch sizes.



(a) PDE

(b) Lunar Lander

Figure 9. VBO-MI vs. baselines at different β values.