A Quadrature Approach for General-Purpose Batch Bayesian Optimization via Probabilistic Lifting

Masaki Adachi MASAKI@ROBOTS.OX.AC.UK
Machine Learning Research Group, University of Oxford, UK

Satoshi Hayakawa HAYAKAWA@MATHS.OX.AC.UK
Mathematical Institute, University of Oxford, UK

Martin Jørgensen MARTIN.JORGENSEN@HELSINKI.FI
Department of Computer Science, University of Helsinki, Finland

Saad Hamid* SAAD.HAMID@AIOILAB-OXFORD.EU
Aioi R&D Lab, Oxford, UK, and Mind Foundry, Oxford, UK

Harald Oberhauser OBERHAUSER@MATHS.OX.AC.UK
Mathematical Institute, University of Oxford, UK

Michael A. Osborne MOSB@ROBOTS.OX.AC.UK
Machine Learning Research Group, University of Oxford, UK

Abstract

Parallelisation in Bayesian optimisation is a common strategy but faces several challenges: the need for flexibility in acquisition functions and kernel choices, flexibility dealing with discrete and continuous variables simultaneously, model misspecification, and lastly fast massive parallelisation. To address these challenges, we introduce a versatile and modular framework for batch Bayesian optimisation via probabilistic lifting with kernel quadrature, called *SOBER*, which we present as a Python library based on GPyTorch/BoTorch. Our framework offers the following unique benefits: (1) Versatility in downstream tasks under a unified approach. (2) A gradient-free sampler, which does not require the gradient of acquisition functions, offering domain-agnostic sampling (e.g., discrete and mixed variables, non-Euclidean space). (3) Flexibility in domain prior distribution. (4) Adaptive batch size (autonomous determination of the optimal batch size). (5) Robustness against a misspecified reproducing kernel Hilbert space. (6) Natural stopping criterion.

Keywords: Batch Bayesian Optimisation, Bayesian Quadrature, Kernel Quadrature

1 Introduction

Bayesian optimisation (BO; Mockus (1975); Garnett (2023)) is a model-based global optimisation strategy for black-box functions. It involves constructing a surrogate model to approximate the function, and subsequently using the model to efficiently select forthcoming query points, thereby offering sample-efficient optimisation. However, with the advancement of the machine learning field, the complexity and variety of practical applications have increased. For example, many hyperparameter optimisation tasks involve mixed variables, diverging from the typical assumption of solely continuous or discrete variables (Ru et al., 2020; Wan et al., 2021; Daulton et al., 2022). Drug discovery (Gómez-Bombarelli et al., 2018; Adachi, 2021), a prominent area for BO and experimental design, necessitates

*. Work done while at Machine Learning Research Group, University of Oxford, UK

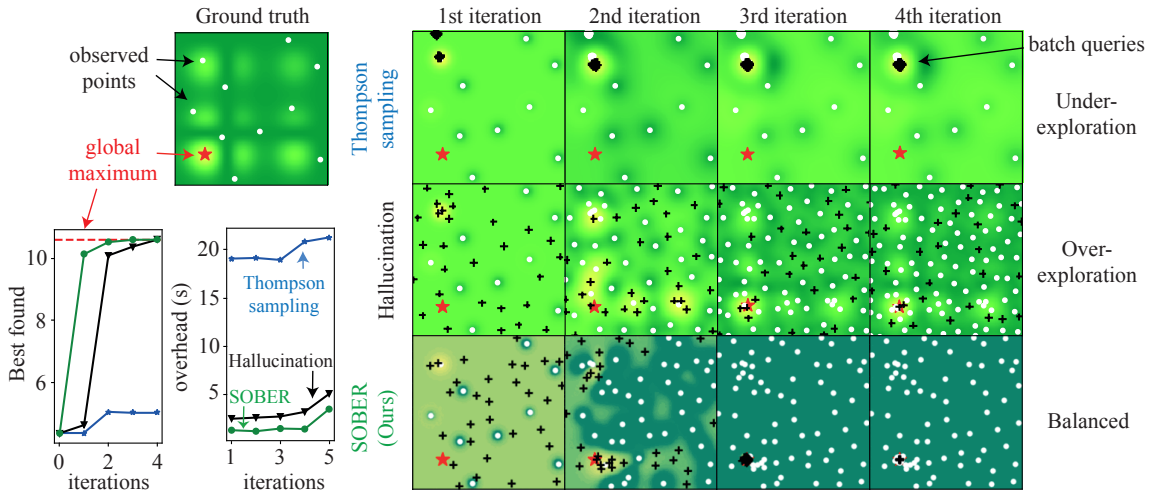


Figure 1: A demonstrating example featuring 2D Branin-Hoo function with nine peaks and the global maximum at the bottom-left corner (red star). Initial 10 i.i.d. samples (white dots) unluckily misidentify the top-left peak as the promising area. Thompson sampling (blue lines) under-explores, erroneously focusing 30 queries (black crosses) near the top-left. Conversely, hallucination (black lines) over-explores, constantly venturing into new regions, yet allocating only a few queries towards the bottom-left area. Our SOBER approach (green lines) starts with wide exploration, then narrows down to the global maximum, demonstrating balanced exploration. The convergence plot illustrates that SOBER outperforms the baselines with the least wall-clock time overhead. The image's colour scheme represents different functions: upper confidence bound for Thompson and hallucination, $\log\pi$ for SOBER.

specialised kernels and non-Euclidean space due to the molecular and graph representations required (Griffiths et al., 2023). Furthermore, real-world tasks often operate under numerous constraints, and such constraint functions can be black-box (also known as unknown constraints, Gelbart et al. (2014)). This complexity has spurred the development of numerous specialised acquisition functions (AFs). Furthermore, these AFs are often incompatible with each other, which is a hindrance for practitioners. Particularly in batch settings, where multiple points are selected simultaneously for parallelising the costly evaluations such as physical experiments, compatibility issues become more evident. The batch setting typically suffers from (1) no compatibility to arbitrary AFs, kernels, or downstream tasks (e.g., constrained optimisation), (2) limited scalability to large batch size, (3) under-/over-explorative samples. Figure 1 demonstrates these issues in popular batch Thompson sampling (TS; Thompson (1933); Kandasamy et al. (2018)) and hallucination (Azimi et al., 2010). Table 1 and §2.2 delineates the details.

In response to these challenges, we introduce *SOBER* (Solving Optimisation as Bayesian Estimation via Recombination), which not only offers more balanced explorative sampling and faster computation times but also exhibits unique advantages: (1) adaptive batch sizes—autonomous determination of the optimal batch size at each iteration, (2) robustness against misspecified GPs—our worst-case error is uniformly bounded in misspecified reproducing Kernel Hilbert Spaces (RKHS), (3) stopping criterion as integral variance, and (4) the domain prior distribution—flexibility to model input domains based on any distri-

Batch methods	task	gradient-free?	adaptive batch?	misspec. RKHS?	stopping criterion?	Any AF?	Any kernel?	const-raint?	large batch?	discrete space?	non-Euclidean?
Batch Thompson sampling (Kandasamy et al., 2018)	BO	✓	✗	✗	✗	✗	✓	✗	✓	✓	✓
DPP-TS (Nava et al., 2022)	BO	✓	✗	✗	✗	✗	✓	✗	✗	✓	✓
MC-SAA (Balandat et al., 2020)	BO	✗	✗	✗	✗	✗	✓	✓	✓	✗	✓
TurBO (Eriksson et al., 2019)	BO	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
SCBO (Eriksson and Poloczek, 2021)	BO	✓	✗	✗	✗	✗	✗	✓	✓	✓	✓
PESC (Hernández-Lobato et al., 2015)	BO	✗	✗	✗	✗	✗	✓	✓	✗	✓	✓
GIBBON (Moss et al., 2021)	BO	✗	✗	✗	✗	✗	✓	✓	✗	✓	✓
B3O (Nguyen et al., 2016)	BO	✗	✓	✗	✗	✓	✓	✗	✗	✗	✗
Hallucination (Azimi et al., 2010)	Any	✗	✗	✗	✗	✓	✓	✓	✗	✓	✓
Local Penalisation (González et al., 2016)	Any	✗	✗	✗	✗	✓	✓	✓	✗	✗	✗
SOBER (Ours)	Any	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparisons between our proposed SOBER with popular batch methods. Task refers to batch BO, BQ, and AL. misspec. RKHS refers to bounded worst-case error against the misspeficied RKHS. Our SOBER is the most versatile algorithm with unique benefits.

bution, unlike typical uniform distribution assumptions. This approach, as illustrated in Figure 1 and summarised in Table 1, positions our algorithm as a highly versatile solution not only for BO, but also for Active Learning (AL; Settles (2009)) and Bayesian Quadrature (BQ; O'Hagan (1991); Hennig et al. (2022)). We firstly introduced the idea of *probabilistic lifting* to batch BO, which enables us to leverage a flexible emphkernel quadrature (KQ) method, thereby offering a versatile and modular approach. As detailed later, specifying the downstream tasks, AFs, and variable types is equivalent to specification in domain distribution. Thus, users can enjoy a plug-and-play parallelisation library for AL, BO, and BQ interchangeably. We have created the open-source library SOBER based on PyTorch (Paszke et al., 2019), GPyTorch (Gardner et al., 2018), and BoTorch (Balandat et al., 2020), providing detailed tutorials with varied use cases.

In summary, we offer:

1. A modular and flexible open-sourced Python library for batch BO, AL, and BQ is ready for `pip install sober-bo` on `https://github.com/ma921/SOBER`, with versatility summarised in Table 1.
2. The unique benefits of adaptive batch sizes, robustness against misspecified RKHS, and domain prior distribution enhances further efficiency and flexibility.
3. An evaluation of the performance of SOBER against baselines in various synthetic and real-world tasks involving large batch sizes, mixed variables, constraints, and non-Euclidean space.

2 Background

In this section, we first introduce the GP, then review the batch BO tasks and related work.

2.1 Gaussian Process

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and $\mathcal{X} \subseteq \mathbb{R}^d$ be the input domain. A GP (Stein, 1999; Rasmussen et al., 2006) is a stochastic process $g : \mathcal{X} \times \Omega \to \mathbb{R}$, whose properties are captured by the mean function $m : \mathcal{X} \to \mathbb{R}$, $m(x) = \mathbb{E}[g(x, \cdot)]$ and covariance function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, $K(x, x') = \mathbb{E}[(g(x, \cdot) - m(x))(g(x', \cdot) - m(x'))]$. The covariance function is symmetric ($K(x, x') = K(x', x), \forall x, x' \in \mathcal{X}$) and positive definite ($\forall t \in \mathbb{N}, \{a_i\}_{i=1}^t \in \mathbb{R}, \{x_i\}_{i=1}^t \subset \mathcal{X}, \sum_{i,j=1}^t a_i a_j K(x_i, x_j) \geq 0$). We refer to any function satisfying the above two properties as a kernel. A GP induces a probability measure over functions, and is capable of conditioning on data in closed form for conjugate likelihood cases. In the regression setting, we further assume the labels $y = f(x)+\epsilon$, where f is the function to estimate, $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is i.i.d. zero-mean Gaussian noise, and σ^2 is the noise variance. Given a labelled data set $\mathcal{D}_t = \{x_i, y_i\}_{i=1}^t := (\mathbf{X}_t, \mathbf{Y}_t)$ and corresponding covariance matrix $\mathbf{K}_{XX} = (K(x_i, x'_j))_{1 \leq i,j \leq t} \in \mathbb{R}^{t \times t}$, the conditioned GP regression model is given by $f \mid D_t \sim \mathcal{GP}(m_t, C_t)$, where

$$m_t(x) = m(x) + K(x, \mathbf{X}_t)(\mathbf{K}_{XX} + \sigma^2 \mathbf{I}_{t\times t})^{-1}(\mathbf{Y}_t - m(\mathbf{X}_t)),$$
$$C_t(x, x') = K(x, x') - K(x, \mathbf{X}_t)(\mathbf{K}_{XX} + \sigma^2 \mathbf{I}_{t\times t})^{-1} K(\mathbf{X}_t, x'),$$

$m_t(\cdot)$ and $C_t(\cdot, \cdot)$ are the mean and covariance functions of the GP posterior predictive distribution conditioned on t-th data set D_t, and $\mathbf{I}_{t\times t}$ is an identity matrix of size t.

2.2 Batch Bayesian Optimisation and Related Work

BO is the task to find the global maximum of a blackbox function f:

$$x^*_{\text{true}} = \operatorname*{argmax}_{x \in \mathcal{X}} f(x),$$

where x^*_{true} represents the global optimum. BO is a model-based optimiser that typically uses a GP as a surrogate model (Osborne et al., 2009). It uses GP predictive uncertainty to solve the blackbox optimisation problem, treating it as active learning to locate the global optimum. BO must balance the trade-off between exploitation (using current knowledge of the optimum from m_t) and exploration (exploring unseen optima considering uncertainty from C_t). Unnecessary exploration can lead to a slower convergence rate for the regret, defined as $r_t := f(x^*_{\text{true}}) - f(x_t)$, where x_t is the t-th query point. The next query point is determined by maximising an acquisition function (AF), with the upper confidence bound (UCB; Srinivas et al. (2010)) being a popular choice: $\alpha_{f_t}(x) := \mu_t(x)+\beta_t^{1/2}\sqrt{C_t(x, x)}$, where β_t represents an optimisation hyperparameter. The rationale behind UCB is the decaying nature of the maximum information gain as more data is acquired (c.f., Cover (1999)). This decay is sublinear for popular kernels (Nemhauser et al., 1978; Krause and Guestrin, 2012), indicating progressively smaller changes for larger values of t, allowing us to demonstrate the *no-regret property*, $\lim_{t\to\infty} \frac{r_t}{t} = 0$. Although there is a vast array of AFs, those with a proven no-regret property are limited to variants of either UCB, expected improvement (EI; Bull (2011); for the noiseless case), or TS, to the best of our knowledge.

However, most AFs are designed for sequential settings, and extending them to a batch setting often results in the loss of the no-regret property. Consequently, batch selection methods are mostly heuristic, yet they are widely accepted due to their practical significance and effectiveness. Batch BO methods can be categorised into the following two:

Optimisation-based approach. A prime example on this approach is the hallucination (Azimi et al., 2010), which extends sequential methods by simulating the sequential process using a random sample from the GP predictive posterior. Despite its simplicity and empirical effectiveness, this method suffers from over-exploration due to mispecified GP models resulting from pseudo-labels (see Figure 1), and scalability issues with large batches. Each sequential query involves AF optimisation, essentially non-convex optimisation reliant on heuristic optimisers (e.g., CMA-ES; Hansen (2016)), thus introducing optimisation errors and overhead for each batch query. An alternative approach, using Monte Carlo (MC)-based AFs (Wilson et al., 2018; Balandat et al., 2020) for efficient parallel computations. However, as the authors noted, popular information-theoretic AFs (Hennig et al., 2015; Hernández-Lobato et al., 2014; Wang and Jegelka, 2017) are not supported. Furthermore, the optimisation-based approach is challenged by a combinatorial explosion in scenarios involving discrete optimisation. As the number of categorical classes grows, the number of potential combinations becomes prohibitively large. Specifically, optimising for large batch sizes requires enumerating all conceivable permutations of *both* batch samples and discrete variables, presenting a significant combinatorial challenge (Moss et al., 2021). While recent work (Daulton et al., 2022) has tackled this, the proof is only applicable to sequential BO.

Thompson sampling-based approach. TS-based approaches can avoid the combinatorial and scalability issues through randomness. The AF of TS is $x_t = \text{argmax}_{x\in\mathcal{X}}\, g(x)$, where $g \sim \mathcal{GP}(m_t, C_t)$ represents a function sample from the GP. This approach can be seen as sampling from the belief about the global optimum locations, $x_t \sim \mathbb{P}(\hat{x}_t^* \mid \mathcal{D}_t)$, where $\hat{x}_t^*$ is the estimated global optimum location. Kandasamy et al. (2018) extended TS to batch BO, which still preserves the no-regret property. In batch BO, the key metric is the Bayesian regret (BR), defined by:

$$\text{BR}(t) := \mathbb{E}_{x_t\in\mathbf{X}_t^n}[f(x_{\text{true}}^*) - f(x_t)]$$

where $\mathbf{X}_t^n$ is the batch TS samples drawn from $\mathbb{P}(\hat{x}_t^* \mid \mathcal{D}_t)$. By using the same rationale of UCB, when the maximum information gain is sublinear in the iteration t, the batch TS enjoys the no-regret properties for BR.

However, it faces limitations: incompatibility with other AFs and under-exploration. While the theory of batch TS depends on a well-specified GP, the common practice of using maximum likelihood estimation (MLE) for kernel hyperparameters does not ensure consistent estimation (Berkenkamp et al., 2019; Ziomek et al., 2024). This misspecification can invalidate no-regret property, leading to aggregated samples (see Figure 1), contrary to theoretical expectations. Although attempts have been made to address these issues (Nava et al., 2022), they often introduce significant overhead due to diversification, such as determinantal point process (DPP; Kathuria et al. (2016)). Exact computation requires costly $\mathcal{O}(|\mathcal{X}| \cdot n^{6.5} + n^{9.5})$, and the best known inexact sampling with Markov chain Monte Carlo (MCMC) still demands $\mathcal{O}(n^5 \log n)$ MCMC steps (Rezaei and Gharan, 2019).

Consequently, a versatile and lightweight batch BO algorithm remains elusive.

3 Connection with Batch Uncertainty Sampling and Kernel Quadrature

In this section, we will demonstrate how KQ can provide a flexible and efficient solution for batch uncertainty sampling. We begin by introducing the concept of *quantisation*, then establish the connection between batch uncertainty sampling and KQ.

Quantisation. Consider π as a probability distribution defined over the domain $\mathcal{X}$. The task of *quantisation* is to find a discrete distribution $\nu := \frac{1}{n}\sum_{i=1}^{n} \delta_{x_i}$, which best approximates π using n representative points x_i. To tackle the quantisation task, one initially identifies an optimality criterion, typically based on a notion of *discrepancy* between π and ν, and then devises an algorithm to approximately minimise this discrepancy.

Kernel Quadrature. KQ is a numerical integration that computes the integral of a function f within an RKHS $\mathcal{H}$ associated with a kernel K. Its goal is to approximate an, otherwise intractable, integral with a weighted sum. A KQ rule, $Q_{\pi,K}(n)$ is defined by weights $\mathbf{w}^n = \{w_i\}_{i=1}^n$ and points $\mathbf{X}^n = \{x_i\}_{i=1}^n$,

$$Q_{\pi,K}(n) := \sum_{i=1}^{n} w_i f(x_i) \approx \int f(x)\mathrm{d}\pi(x). \tag{1}$$

The KQ rule can also be interpreted with a discrete distribution $\pi_{\mathrm{KQ}} := \sum_{i=1}^{n} w_i \delta_{x_i}$, namely, $Q_{\pi,K}(n) = \sum_{i=1}^{n} w_i f(x_i) = \int f(x)\mathrm{d}\pi_{\mathrm{KQ}}(x)$. The *worst-case error*, given π and $\mathcal{H}$, is

$$\mathrm{wce}[Q_{\pi,K}(n)] := \sup_{\|f\|_{\mathcal{H}}\leq 1} \left| Q_{\pi,K}(n) - \int f(x)\mathrm{d}\pi(x) \right|,$$

and KQ aims to approximate $Q_{\pi,K}(n)$ that minimises this worst-case error,

$$\mathbf{X}^n, \mathbf{w}^n \approx \operatorname*{argmin}_{\mathbf{X}^n\subset\mathcal{X}, \mathbf{w}^n\subset\mathbb{R}} \mathrm{wce}\left[Q_{\pi,K}(n)\right]. \tag{2}$$

There are a vast list of KQ algorithms; herding (Chen et al., 2010; Bach et al., 2012), leverage score (Bach, 2017), DPP (Belhadji et al., 2019), continuous volume sampling (Belhadji et al., 2020), kernel thinning (Dwivedi and Mackey, 2021, 2022), to name a few.

Connection to Quantisation. The worst-case error can be considered a *divergence* between π and π_{KQ}. There is a theoretical link between KQ and quantisation, as KQ represents *weighted* quantisation under the maximum mean discrepancy (MMD) metric (Karvonen, 2019; Teymur et al., 2021). MMD is a method for quantifying the divergence between two distributions (Sriperumbudur et al., 2010; Muandet et al., 2017), defined as:

$$\mathrm{MMD}_{\mathcal{H}}(\pi_{\mathrm{KQ}}, \pi) := \left\| \int K(\cdot, x)\mathrm{d}\pi_{\mathrm{KQ}}(x) - \int K(\cdot, x)\mathrm{d}\pi(x) \right\|_{\mathcal{H}},$$

and we can rewrite as (Huszár and Duvenaud, 2012):

$$\mathrm{MMD}^2_{\mathcal{H}}(\pi_{\mathrm{KQ}}, \pi) := \sup_{\|f\|_{\mathcal{H}}=1} \left| \int f(x)\mathrm{d}\pi_{\mathrm{KQ}}(x) - \int f(x)\mathrm{d}\pi(x) \right|^2.$$

This squared formulation equates to the worst-case error: solving for KQ is the same to finding the discrete distribution π_{KQ} that best approximates π in terms of MMD. Note, KQ performs *weighted* quantisation, differing from the previous unweighted quantisation.

Connection to Gaussian Process. Assuming a function f_t is modelled by a GP, $f_t \sim \mathcal{GP}(m_t, C_t)$, with noisy observed points, $\mathcal{D}_t := (\boldsymbol{X}_t, \boldsymbol{Y}_t)$. Our objective is to estimate the expectation of the function $\hat{Z} := \int f(x)\mathrm{d}\pi(x)$. This scenario is referred to as Bayesian quadrature (BQ) (O'Hagan, 1991; Hennig et al., 2022), with integral estimates given by:

$$\mathbb{E}_{f_t\sim\mathcal{GP}(m_t,C_t)}[\hat{Z}] = \int m_t(x)\mathrm{d}\pi(x) = \boldsymbol{z}_t^\top(\mathbf{K}_{XX} + \sigma^2\mathbf{I}_{t\times t})^{-1}\mathbf{Y}_t, \tag{3a}$$

$$\mathbb{V}_{f_t\sim\mathcal{GP}(m_t,C_t)}[\hat{Z}] = \int C_t(x,x')\mathrm{d}\pi(x)\mathrm{d}\pi(x') = z_t' - \boldsymbol{z}_t^\top(\mathbf{K}_{XX} + \sigma^2\mathbf{I}_{t\times t})^{-1}\boldsymbol{z}_t, \tag{3b}$$

where $\boldsymbol{z}_t := \int K(x, \mathbf{X}_t)\mathrm{d}\pi(x)$ and $z_t' := \int K(x,x')\mathrm{d}\pi(x)\mathrm{d}\pi(x')$ represent the kernel mean and variance, respectively. To enhance the accuracy of integration, it is desirable to minimise the uncertainty in the integral estimation as expressed in Eq.(3b). Therefore, Eq.(3b) can be regarded as the metric to assess the reduction in integral variance, which has been employed as the AF for BQ (Rasmussen and Ghahramani, 2002; Osborne et al., 2012).

Connecting it All Together. Huszár and Duvenaud (2012) demonstrated that all of the worst-case error, MMD, and the integral variance are *equivalent*. The BQ expectation in Eq.(3a) is a weighted sum; $\boldsymbol{z}^\top\boldsymbol{K}^{-1}\boldsymbol{y}_0 = \sum_{i=1}^n w_{BQ,i}y_i$, where $w_{\mathrm{BQ,\,j}} := \sum_{i=1}^n \boldsymbol{z}_i^\top\boldsymbol{K}^{-1}_{i,j}$ and $\boldsymbol{K}^{-1} := (\mathbf{K}_{XX} + \sigma^2\mathbf{I}_{t\times t})^{-1}$. These weights can be considered as forming a discrete distribution $\pi_{\mathrm{BQ}} := \sum_{i=1}^n w_{\mathrm{BQ},i}\delta_{x_i}$, thereby allowing the integral variance estimation to be expressed as:

$$\mathbb{V}_{f_t\sim\mathcal{GP}(m_t,C_t)}[\hat{Z}] = \mathrm{MMD}^2_{\mathcal{H}}(\pi_{\mathrm{BQ}}, \pi) = \inf_{\mathbf{w}_{\mathrm{BQ}}} \mathrm{wce}[Q_{\pi,C_t}]^2 \tag{4}$$

for a fixed X, where the kernel for MMD and KQ is the predictive covariance $C_t(\cdot,\cdot)$. This choice is due to $C_t(\cdot,\cdot)$ representing the posterior belief about f, which is expected to be more accurate than the prior belief represented by $K(\cdot,\cdot)$.

This demonstrates the close connection between KQ, GP, and quantisation. This equivalence shows that KQ is *domain-aware batch uncertainty sampling*. Solving KQ is minimising the worst-case error, which is equivalent to minimising both MMD and the integral variance. MMD, being the quantisation, ensures that the resulting discrete points are spread over the distribution π_t to approximate, representing domain-aware diversified sampling. The integral variance represents the expected uncertainty of GP, and its minimisation indicates batch uncertainty sampling. At first glance, minimising $\mathbb{V}_{f_t}[\hat{Z}]$ for uncertainty sampling might seem counterintuitive, as sequential uncertainty sampling typically maximises $C_t(x,x)$. However, $\mathbb{V}_{f_t}[\hat{Z}]$ is a scalar value, not a function like $C_t(\cdot,\cdot)$, and computes a summary statistic indicating the quality of the selected nodes' approximation of the integral. Therefore, minimising this metric by selecting batch samples can be understood as batch uncertainty sampling. Importantly, KQ is the approximation of intractable integration, making it applicable to an arbitrary combination of (K, π), unlike BQ[1].

In Summary. A quantisation task can be regarded as a KQ task. The selected batch samples aim to minimise the divergence between the target distribution π and the batch samples' distribution π_{KQ}. By employing the GP predictive covariance $C(\cdot,\cdot)$ as the kernel

1. See Eq.(1). While BQ needs the analytical kernel mean for the right hand side (Eqs.(3a)-(3b)), KQ is approximating it with the weighted sum. Our previous work (Adachi et al., 2022) applied to batch BQ.

for the MMD, KQ transitions into batch exploration of GP uncertainty, concurrently minimising divergence from the target distribution. Consequently, batch construction through KQ offers a means to quantise the target distribution while incorporating uncertainty sampling. The benefits of KQ are:

1. Applicable to any kernel, given that the primary goal of the KQ objective is to approximate the intractable integral of the kernel function.
2. Versatile across any domain, AFs, or constraints, provided the target distribution can be described as a probability measure π.
3. To naturally produce diversified batch samples, and is able to assess its diversity using the widely recognised MMD criterion.

4 Batch Bayesian Optimisation as Quadrature

We now consider the application of KQ to the batch BO task. First, we demonstrate that the probabilistic lifting technique can transform the batch BO task into a KQ problem. Next, we explain how to solve this reinterpreted task using a KQ algorithm. Finally, we customise this general batch BO algorithm for varied cases.

4.1 Probabilistic Lifting

Algorithm 1 SOBER algorithm.

Require: domain prior π_0, initial data set $\mathcal{D}_0 = (\mathbf{X}_0, \mathbf{Y}_0)$, stopping criterion Δ_n

$f_{t-1} \leftarrow$ Initialise-GP($\mathcal{D}_0$)
while $\mathbb{V}_x[\tilde{\pi}] < \Delta_n$ **do**
  $\pi_{t-1}, \alpha_{t-1}, C_{t-1} \leftarrow$ Fit-GP-and-Update-$\pi(f_{t-1})$
  $\mathbf{X}_t^n, \mathbf{w}_t^n, \mathbb{E}_{f_t}[\hat{Z}], \mathbb{V}_x[\tilde{\pi}] \leftarrow \mathrm{KQ}(\pi_{t-1}, \alpha_{t-1}, C_{t-1})$
  $\mathbf{Y}_t^n =$ Parallel-Query($f_{\text{oracle}}(\mathbf{X}_t^n)$)
  Update dataset $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup (\mathbf{X}_t^n, \mathbf{Y}_t^n)$ and model $f_t \leftarrow$ Update-GP($f_{t-1}, \mathcal{D}_t$).
  Proceed next round $t \leftarrow t - 1$.
end while
return global maximum estimate $\hat{y}_t^* = \max[\mathbf{Y}_T]$, evidence estimate $\mathbb{E}_{f_t}[\hat{Z}]$

To recast the batch BO task as a KQ task using probabilistic lifting, consider the dual formulation presented below:

$$x^*_{\text{true}} \in \operatorname*{argmax}_{x \in \mathcal{X}} f(x) \quad \overset{\text{dual}}{\Longleftrightarrow} \quad \delta_{x^*_{\text{true}}} \in \operatorname*{argmax}_{\pi \in \mathbb{P}(\mathcal{X})} \int f(x) \mathrm{d}\pi(x), \tag{5}$$

where δ_x denotes the delta distribution at x, making $\delta_{x^*_{\text{true}}}$ the point mass at the global maximum. Consequently, our goal aligns with the KQ objective in Eq.(1), allowing the application of KQ algorithms to the batch BO task.

How do we interpret this dual formulation? We transform a non-convex optimisation problem, $\max f(x)$, into an infinite-dimensional optimisation over the set of probability measures $\mathbb{P}(\mathcal{X})$. In other words, we do not consider pointwise updates: $\operatorname{plim}_{t\to\infty} x_t = x^*_{\text{true}}$ as in the conventional approach. Instead, we aim at *distributional* updates, $\operatorname{plim}_{t\to\infty} \pi_t = \delta_{x^*_{\text{true}}}$,

i.e., $\operatorname{plim}_{t\to\infty} \mathbb{E}_x[\pi_t] = x^*_{\text{true}}$, $\operatorname{plim}_{t\to\infty} \mathbb{V}_x[\pi_t] = 0$. This yields $\max \int f(x)\mathrm{d}\pi(x)$, making the non-convex objective f linear and convex for π. This distributional formulation is attractive due to parallelisability and convexity, widely used in optimisation, from traditional primal-dual interior-point methods (Vandenberghe and Boyd, 1996; Wright, 1997) to contemporary Bayesian machine learning theories (Rudi et al., 2020; Wild et al., 2023).

Algorithm 1 outlines our algorithm, SOBER: Line 3 updates π_{t-1} based on the GP f_{t-1}. Then, Line 4 employs the KQ algorithm to perform batch uncertainty sampling over π_{t-1} to effectively reduce the uncertainty $C_{t-1}(\cdot,\cdot)$ by selecting batch points as quantisation π_{KQ}, where $\pi_{\text{KQ}} = \sum_{i=1}^n w_{KQ,i}\delta_{x_i}$, with $w_{KQ,i} \in \mathbf{w}_t^n$ and $x_i \in \mathbf{X}_t^n$. The resulting $\mathbf{X}_t^n$ is the n-point batch BO samples, ensuring diversified batch uncertainty sampling over π_{t-1}. As illustrated in Figure 1, π_{t-1} initially spans the domain, with $\mathbf{X}_t^n$ diversified and progressively concentrating towards the global maximum over iterations. The variance $\mathbb{V}_x[\pi_t]$ becomes a natural choice of stopping criterion for a distributional convergence.

The next question is, *what is π, and how do we update it?* Our probabilistic lifting transforms the original non-convex problem into an even more computationally demanding problem. Traditional algorithms often assume f is polynomial, allowing for a further transition to moment space due to closed-form moments of f. However, with our black-box f and the GP surrogate model's lack of closed-form kernel mean and variance for arbitrary π, a possible remedy is to presuppose a functional form for π. We propose two assumptions regarding π.

Thompson Sampling Interpretation (SOBER-TS). The first approach interprets π as a probability distribution over the estimated global maxima $\hat{x}_t^*$, denoted as $\mathbb{P}(\hat{x}_t^*)$, where $\hat{x}_t^*$ represents the current estimation of the global maxima at f_t. The advantage of this perspective is that it aligns with existing theories on batch TS.

We unpack the interpretation of $\pi = \mathbb{P}(\hat{x}_t^*)$ step by step:

1. $\hat{x}_t^* \sim \pi_t(x)$ is TS, namely $\hat{x}_t^* = \operatorname{argmax}_{x\in\mathcal{X}} g_t(x)$ and $g_t \sim \mathcal{GP}(m_t, C_t)$.
2. π_t is updated through conditioning f_t with the new observations.
3. The KQ approach selects batch samples that minimise the expected uncertainty $\mathbb{V}_{f_t}[\hat{Z}]$, allowing us to view $\mathbf{X}_t^n$ as TS samples that most contribute to reducing uncertainty across the distribution $\mathbb{P}(\hat{x}_t^*)$.

How can we interpret applying KQ for batch TS with regard to the domain-aware batch uncertainty sampling? Firstly, the domain-aware means that the resulting KQ samples $\mathbf{X}_t^n$ adhere to the original TS distribution, i.e., $\mathbf{X}_t^n \sim \mathbb{P}(\hat{x}_t^*) = \pi_t(x)$. This method can thus be seen as a variant of batch TS, referenced in studies such as Kandasamy et al. (2018); Hernández-Lobato et al. (2017); Ren and Li (2024); Dai et al. (2023).

Secondly, how does batch uncertainty sampling help the regret converge faster? As noted in §2.2, BR convergence rate depends on the spectral decay of maximum information gain defined as $\mathbb{I}(\mathbf{Y}_t; f) = \mathbb{H}(\mathbf{Y}_t) - \mathbb{H}(\mathbf{Y}_t \mid f)$, quantifying the reduction in uncertainty about f from revealing $\mathbf{Y}_t$. For a GP, $\mathbb{I}(\mathbf{Y}_t; f) = \mathbb{I}(\mathbf{Y}_t; \mathbf{f}_t) = \frac{1}{2}\log|\mathbf{I}_{t\times t} + \sigma^{-2}\mathbf{K}_{XX}|$, where $\mathbf{f}_t := f(\mathbf{X}_t)$. Nemhauser et al. (1978); Krause and Guestrin (2012); Srinivas et al. (2010) have shown that the information gain maximiser can be approximated by an uncertainty sampling with $(1 - 1/e)$ approximation guarantee. Thus, roughly speaking, the maximum information

gain can be approximately seen as the largest predictive uncertainty C_t. Ultimately, a faster spectral decay in the maximum information gain in iteration t leads to faster BR convergence rate. This is the reason why typical BO theoretical paper has the kernel-specific convergence rate as each kernel has different spectral decay. Here, in later §4.2.2 and Theorem 1, we show that KQ batch uncertainty sampling can be understood as selecting the largest possible spectral decay of the given kernel C_t. Thus, roughly speaking, KQ is trying to select the samples with the largest possible information gain in batch n, thereby accelerating the spectral decay in iteration t. Moreover, in later Proposition 2, we show that KQ is robust against model misspecification, thereby avoiding remaining stuck in local minima. Therefore, KQ can give robust, fast spectral decay sampling for batch TS.

In distributional interpretation, KQ selects the points that minimise $\mathbb{V}_f[\hat{Z}]$, which minimises the predictive uncertainty C_t over the current TS distribution π_t. The main source of variance $\mathbb{V}_x[\pi_{t-1}]$ is the predictive uncertainty C_t. Hence, the batch uncertainty sampling with KQ narrows the subsequent TS distribution π_t, steering it closer to x^*_{true}.

However, deriving the BR convergence rate of SOBER-TS is non-trivial because it requires an analysis of double spectral decay (one for batch n in KQ, and one for iteration t in maximum information gain). While the focus here is not on the SOBER-TS algorithm, future work may explore its BR convergence rate. Given the relationship between DPP and KQ (Belhadji et al., 2019; Belhadji, 2021), SOBER-TS is expected to match the convergence rate of DPP-TS (Nava et al., 2022), which has demonstrated a tighter Bayesian cumulative regret bound compared to standard batch TS approaches (Kandasamy et al., 2018).

Likelihood-Free Inference Interpretation (SOBER-LFI). Figure 1 illustrates that sampling directly from the TS distribution tends to remain stuck in local minima, contrary to theoretical expectations (Kandasamy et al., 2018). This discrepancy arises from two primary causes: model misspecification and the non-closed-form nature of the distribution. Model misspecification leads to a mis-estimated distribution of $\hat{x}^*_t$. This causes sampling to be biased toward less promising regions, especially in the initial stages, as seen in Figure 1. This phenomenon is well-documented in the bandit literature (Simchowitz et al., 2021; Kim et al., 2021; Aouali et al., 2023). Although exploratory adjustments through diversified sampling (Nava et al., 2022) can alleviate this issue, they entail prohibitive computational costs. This is attributed to the challenge of sampling from low-probability regions due to the non-closed-form distribution, as random sampling $\hat{x}^*_t = \text{argmax}_{x\in\mathcal{X}}\, g_t(x)$ is governed by its probability $\mathbb{P}(\hat{x}^*_t)$. Drawing samples from low-probability areas requires an exhaustive number of attempts (or luck). A closed-form expression enables more flexible sampling schemes, such as importance sampling.

To devise a more robust and fast sampling algorithm, we now consider a closed-form definition for π. Unlike previous bandit approaches that improve TS algorithms, we explore a non-TS approach. Given the uncertain nature of the global maximiser $\mathbb{P}(\hat{x}^*_t)$, x^*_{true} could be at any location with values potentially exceeding the estimated maximum, denoted as $\hat{y}^*_t := \max f(\mathbf{X}_t)$. With this insight, we can define $\pi_t(x)$ as follows:

$$\pi_t(x) := \mathbb{P}\Big(f_t(x) \geq \hat{y}^*_t \mid \mathcal{D}_t\Big) \propto \Phi\left[\frac{m_t(x) - \hat{y}^*_t}{\sqrt{C_t(x,x)}}\right], \tag{6}$$

where Φ is the cumulative distribution function (CDF) of the standard normal distribution. This formulation aligns with the probability of improvement (PI; Kushner (1964)), another widely-used AF in BO, offering a closed-form (albeit unnormalised) distribution that is easier to sample from than TS.

We interpret, for pedagogical reasons, the sequential update of π as likelihood-free inference (LFI, Hinton (2002); Hyvärinen (2005); Gutmann and Hirayama (2011); Huang et al. (2023))[2]. LFI is particularly valuable when the analytical form of the likelihood is unavailable. In the context of BO, while we have a Gaussian likelihood $\mathbb{P}(y \mid x)$ for *observed* data, the likelihood of the global optimum $\mathbb{P}(x^*_{\text{true}} \mid \hat{y}^*_t, \mathcal{D}_t)$ lacks an analytical form, because the true value of x^*_{true} is unknown. This prevents the evaluation of the distance between a queried point x_t and x^*_{true}. LFI addresses this by replacing the exact likelihood function with a 'synthetic likelihood', which assesses divergence between observed and simulated data using summary statistics. This synthetic likelihood is updated with new observations and converges to the true likelihood as $t \to \infty$. This approach can be understood as a variant of the Bernstein-von Mises theorem (Van der Vaart, 2000); under an infinite data scenario ($t \to \infty$), the Bayesian posterior converges to the MLE. The synthetic likelihood exhibits similar asymptotic behaviour (Pacchiardi et al., 2021). Gutmann and Corander (2016) demonstrated that PI function can be interpreted as a synthetic likelihood, where the CDF serves as summary statistic and $\hat{y}^*_t$ as the optimal threshold for LFI, asymptotically converging to standard Bayesian inference. Wilson (2024) also rediscovered this (c.f., they framed Eq.(6) as ϵ-optima, where $\epsilon = \hat{y}^*_t$). Song et al. (2022) extended on this LFI idea to allow non-GP surrogate models to employ for BO. Furthermore, Wild et al. (2023) showed that the probabilistic lifting formulation could be understood within the framework of Bayesian inference. In this light we view $\pi_t(x)$ as the LFI synthetic likelihood of the global maximum $\mathbb{P}(\hat{x}^*_t \mid \hat{y}^*_t, \mathcal{D}_t)$, wherein $\pi_t(x)$ is updated and converging to $\text{plim}_{t\to\infty} \mathbb{P}(\hat{x}^*_t \mid \hat{y}^*_t, \mathcal{D}_t) \to \mathbb{P}(x^*_{\text{true}}) = \delta_{x^*_{\text{true}}}$ as the iteration t progresses, i.e., $\text{plim}_{t\to\infty} \hat{y}^*_t \to y^*_{\text{true}} = f(x^*_{\text{true}})$. We denote this approach SOBER-LFI.

Figure 2 illustrates SOBER-LFI. The purple curve represents the synthetic likelihood π_t as defined in Eq.(6), with the KQ algorithm selecting 10 batch samples that closely approximate the π_t distribution and significantly contribute to reducing uncertainty. As the process progresses, the KQ-selected samples effectively reduce uncertainty: $\pi_{t=1}$ evolves into a much sharper $\pi_{t=2}$, centering around x^*_{true}, and by $\pi_{t=3}$, it nearly mirrors the delta distribution $\delta_{x^*_{\text{true}}}$. Remarkably, LFI ensures non-zero values in uncertain regions, enabling KQ to sample from these zones with very small probability in the last iteration, adhering to Cromwell's rule[3] (Jackman, 2009). This behaviour is assured by the decaying, yet non-zero, nature of the CDF in Eq.(6), illustrating SOBER-LFI as a fusion of the exploitative PI π_t and the explorative KQ batching algorithm.

Deriving the BR bound for this LFI framework presents more complexities compared to traditional TS strategies. To our knowledge, Wang et al. (2018) stands alone in offering a simple regret analysis for PI. They predicated on the strong assumption that $y^*_{\text{true}} = f(x^*_{\text{true}})$ is known beforehand, and even if we accept it, the convergence rate does

2. LFI is often called 'indirect inference' (Gourieroux et al., 1993), 'synthetic likelihood' (Wood, 2010; Price et al., 2018), or Approximate Bayesian Computation (ABC, Csilléry et al. (2010); Fujisawa et al. (2021))
3. According to this principle, the prior probability should remain non-zero for all possibilities. Since synthetic likelihood is updated sequentially, π_{t-1} can be considered the prior for π_t.

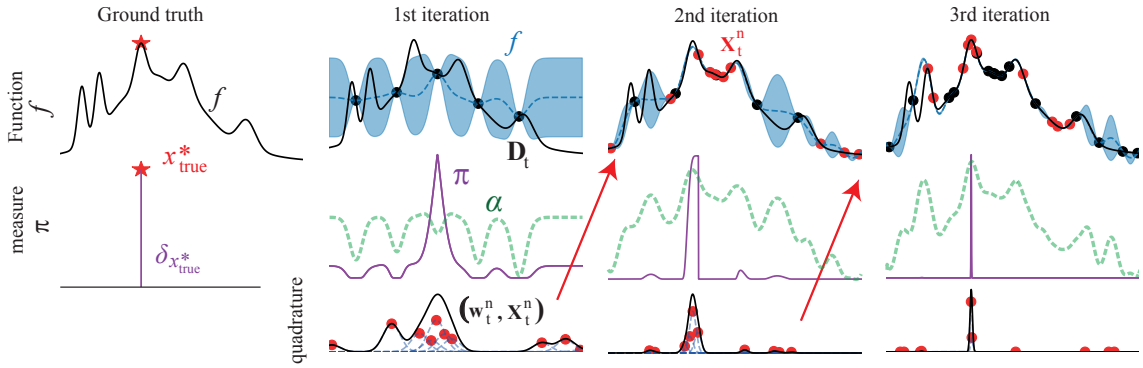


Figure 2: SOBER algorithm. Finding the location of global maximum x^*_{true} is equivalent to finding the delta distribution $\delta_{x^*_{\text{true}}}$. Based on the surrogate f_t, we approximate the probability of global maximum $\mathbb{P}(\hat{x}^*_t)$ as π. We can also set the user-defined acquisition function α_t to adjust batch samples (UCB in this case). KQ algorithm gives a weighted point set $(\mathbf{w}^n_t, \mathbf{X}^n_t)$ that makes a discrete probability measure approximating π (quantisation). Here, we have used a weighted kernel density estimation based on $(\mathbf{w}^n_t, \mathbf{X}^n_t)$ to approximately visualise the quantisation via KQ. Over iterations, π shrinks toward global maximum, which ideally becomes the delta function in a single global maximum case.

not hold a no-regret guarantee (Takeno et al., 2023). Nonetheless, although PI is not theoretically well-motivated, there are numerous successful studies in practice (e.g., Bergstra et al. (2011); Akiba et al. (2019)). Additionally, our focus diverges from mere maximisation, $x_t = \max_{x \in \mathcal{X}} \pi_t(x)$, to probabilistic sampling, $x_t \sim \pi_t(x)$. Hence, our SOBER-LFI aims diverge from those of sequential PI maximisation strategy. Interestingly, recent theoretical studies (Takeno et al., 2023; Ren and Li, 2024) have established a tighter BR bound by integrating TS with PI, surpassing the results of conventional batch TS (Kandasamy et al., 2018). Viewing TS as a randomness generator, merging TS with PI can be conceptually likened to PI supplemented by exploratory adjustments, aligning with our SOBER-LFI's concept. Thus, this emerging theoretical discourse might eventually elucidate the BR bounds for our SOBER-LFI algorithm. Still, despite their theoretical importance, we highlight that their methodologies are a variant of batch TS, thereby exhibiting limitations in misspecification robustness and universality, as depicted in Table 1.

4.2 Recombination: Kernel Quadrature Algorithm

In this section, we reinterpret Eq.(5) as a KQ problem and introduce a KQ algorithm to address it. Although any KQ method listed in §3 may be applied, we opt for the recombination approach (Hayakawa et al., 2022) to afford greater flexibility.

4.2.1 Problem Setting of Kernel Quadrature

For simplicity, we begin by considering the discrete optimisation scenario where $|\mathcal{X}| < \infty$. Suppose we have a kernel $C_t(\cdot, \cdot)$, which represents the GP posterior predictive covariance at the t-th iteration, and a set of N-point samples $\mathbf{X}^N_t \in \mathcal{X}$, alternatively denoted by $x_i \in \mathbf{X}^N_t$, associated with non-negative weights $\mathbf{w}^N_t$, where $\{w_i \in \mathbf{w}^N_t \mid w_i > 0, \sum_{i=1}^N w_i = 1\}$. We

express this configuration as $\pi_t(x) := \sum_{i=1}^N w_i \delta_{x_i}$, treating it as a discrete distribution, or alternatively, as the ordered pair $(\mathbf{w}_t^N, \mathbf{X}_t^N)$. The goal is to identify a weighted subset $\pi_{\text{KQ}}(x) := (\mathbf{w}_t^n, \mathbf{X}_t^n) = \sum_{j=1}^n w_j \delta_{x_j}$, where $n \ll N$, that minimises the MMD between π_t and π_{KQ}, given the initial π_t and the kernel $C_t(\cdot,\cdot)$. The quantised subset π_{KQ}, with $\mathbf{X}_t^n$ being a subset of $\mathbf{X}_t^N$, determines the batch samples for batch BO. In this discrete framework, we are equipped to compute the analytical worst-case error for an arbitrary kernel:

$$\begin{aligned}
\mathbb{E}_{f_{t-1}}[\hat{Z}] &= \int m_{t-1}(x) \mathrm{d}\pi_{t-1}(x) \approx \mathbf{w}_t^{n\top} m_{t-1}(\mathbf{X}_t^n), \\
\mathbb{V}_{f_{t-1}}[\hat{Z}] &= \text{wce}[Q_{\pi_t, C_{t-1}}(n)], \\
&= \mathbf{w}_t^{n\top} C_{t-1}(\mathbf{X}_t^n, \mathbf{X}_t^n) \mathbf{w}_t^n - 2\mathbf{w}_t^{n\top} C_{t-1}(\mathbf{X}_t^n, \mathbf{X}_t^N) \mathbf{w}_t^N + \mathbf{w}_t^{N\top} C_{t-1}(\mathbf{X}_t^N, \mathbf{X}_t^N) \mathbf{w}_t^N.
\end{aligned}$$

Initially, $\pi_{t=0}$ represents the pool of unlabelled inputs, each assigned equal weights. As the iteration t progresses, π_t evolves into a subset of $\pi_{t=0}$, defined as $\mathbf{X}_t^N = \mathbf{X}_{t=0}^N \backslash \mathbf{X}_{t-1}^N$[4]. The corresponding weights, $\mathbf{w}_t^N$, are determined by whether π is interpreted as TS or LFI.

Departing from the settings in previous works (Hayakawa et al., 2022; Adachi et al., 2022), we introduce the following conditions to our framework:

(a) A reward function $\alpha : \mathcal{X} \to \mathbb{R}$ is introduced to add flexibility, integrating additional considerations (e.g., AF). The objective is to maximise the expected reward $\mathbf{w}_t^{n\top} \alpha(\mathbf{X}_t^n)$ while minimising the worst-case error $\text{wce}[Q_{\pi_t, C_{t-1}}(n)]$.

4.2.2 Kernel Quadrature via Nyström Approximation

While the Nyström method (Williams and Seeger, 2000; Drineas and Mahoney, 2005; Kumar et al., 2012) is commonly used for approximating large Gram matrices through low-rank matrices, it also offers a direct approach for approximating the kernel function itself. By selecting a set of M-points $X_t^M = \{x_k\}_{k=1}^M \subset \mathcal{X}$, the Nyström approximation for $C_t(x, y)$ can be described as follows:

$$C_t(x, y) \approx \tilde{C}_t(x, y) := \sum_{j=1}^{n-1} \lambda_j^{-1} \varphi_j(x) \varphi_j(y), \tag{7}$$

where $\varphi_j(\cdot) := u_j^\top C_t(X_{\text{nys}}, \cdot)$, for $j = 1, \ldots, n-1$, are termed *test functions* and are derived from the broader M-dimensional space $\text{span}\{C_t(x_k, \cdot)\}_{k=1}^M$. To facilitate Eq.(7), the optimal rank-s approximation of the Gram matrix $C_t(\mathbf{X}_t^M, \mathbf{X}_t^M) = U \Lambda U^\top$ is determined via eigen-decomposition. Here, $U = [u_1, \ldots, u_M] \in \mathbb{R}^{M \times M}$ represents a real orthogonal matrix, and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_M)$ consists of eigenvalues $\lambda_1 \geq \ldots \geq \lambda_M \geq 0$. Eq.(7) holds if $\lambda_s > 0$.

We can leverage these test functions for the integration estimator $\hat{Z} = \int f(x) \mathrm{d}\pi_t(x)$. With a spectral decay in eigenvalues, the Nyström method efficiently approximates the original kernel function using a limited set of test functions. Defining $\boldsymbol{\varphi} = \{\varphi_1, \ldots, \varphi_{n-1}\}^\top$ as the vector of test functions spanning $\mathcal{H}_{\tilde{C}_t}$—the RKHS linked with the approximated kernel $\tilde{C}_t$—we assume knowledge of expectations, $\int \boldsymbol{\varphi}(x) \mathrm{d}\pi_t(x) = \mathbf{w}_t^{N\top} \boldsymbol{\varphi}(\mathbf{X}_t^N)$, is accessible.

4. For brevity, we use $\mathbf{X}_t^N$, despite $\mathbf{X}_t^{N-tn}$ being more precise as $|\mathbf{X}_t^N| = N - nt$. The same for $\mathbf{w}_t^N$.

This knowledge facilitates the construction of a convex quadrature:

$$\sum_{j=1}^{n-1} w_j \varphi_i(x_j) = \int \boldsymbol{\varphi}(x) \mathrm{d}\pi_t(x) \approx \int f(x) \mathrm{d}\pi_t(x). \tag{8}$$

Consequently, we can approximate the integral using $n-1$ test functions, interpreting Eq.(8) as $n-1$ *equality constraints* that both w_j and x_j must satisfy.

This method's advantage lies in its use of spectral decay information from the Gram matrix, promoting faster convergence. If the target function f is smooth and exhibits fast spectral decay, then a small array of test functions can accurately represent the function, enhancing the efficiency of batch BO.

4.2.3 Linear Programming Formulation

To solve the KQ task in Eq.(2), we introduce a linear programming (LP) problem. This problem is designed to simultaneously maximise the reward and minimise the worst-case error, modifying the approach from (Adachi et al., 2022):

$$\begin{aligned} \max_{\mathbf{w} \subset \mathbb{R}_{\geq 0}} \quad & \mathbf{w}^\top \alpha_t(\mathbf{X}_t^N), \\ \text{subject to} \quad & (\mathbf{w} - \mathbf{w}_t^N)^\top \varphi_j(\mathbf{X}_t^N) = \mathbf{0}, \\ & \mathbf{w}^\top \mathbf{1} = 1, \quad \mathbf{w} \geq \mathbf{0}, \quad |\mathbf{w}|_0 = n, \forall j : 1 \leq j \leq n-1, \end{aligned} \tag{9}$$

where (λ_j, φ_j) are derived from the Nyström approximation (recall §4.2.2), $\mathbf{1} = [1, \ldots, 1]^N$ signifies an all-ones vector, similarly for $\mathbf{0}$, $\mathbb{R}_{\geq 0}$ represents the set of non-negative real numbers, and $|\cdot|_0$ indicates the count of non-zero entries.

The intuition of this formulation is as follows:

(1) The solutions are defined by sparse weights $\mathbf{w}$, where each non-zero weight corresponds to batch selection. The associated samples $\mathbf{X}_t^N$ define the batch samples $\mathbf{X}_t^n \subset \mathbf{X}_t^N$. We denote the non-zero weights and their respective samples as the solution $\pi_{\mathrm{KQ}} = (\mathbf{w}_t^n, \mathbf{X}_t^n)$, with the batch size being $|\mathbf{X}_t^n| = |\mathbf{w}|_0 = n$. Thus, this LP problem aims to subsample batch samples π_{KQ} from the given discrete distribution $(\mathbf{w}_t^N, \mathbf{X}_t^N) \sim \pi_t$, effectively performing quantisation.

(2) The goal is to maximise the expected reward α_t. For example, if UCB is chosen as α_t, it steers the batch samples towards the highest expected reward.

(3) The first set of LP constraints[a] are equality constraints employing test functions from Eq.(8). These $n-1$ equality constraints are stringent, significantly restricting the flexibility typically afforded by LP problems. Within this constrained space, the algorithm seeks the largest possible expected reward. When $\alpha(x) = 0$, the problem reverts to the standard KQ task, with the algorithm generating candidate sets $(\mathbf{w}_t^n, \mathbf{X}_t^n)$ that fulfill these constraints.

(4) The other constraints ensure that the number of non-zero elements in $\mathbf{w}$ matches the requested batch size n, maintaining convex and positive weights.

a. This is the worst-case error. $\left|\mathbf{w}^\top \boldsymbol{\varphi}(\mathbf{X}_t^N) - {\mathbf{w}_t^N}^\top \boldsymbol{\varphi}(\mathbf{X}_t^N)\right| \approx \left|\int f(x) \mathrm{d}\pi_{\mathrm{KQ}}(x) - \int f(x) \mathrm{d}\pi_t(x)\right|$.

As such, solving the LP problem as defined in Eq.(9) equates to addressing the KQ task using the Nyström approximation. This equivalence provides an efficient solution framework for the batch BO algorithm.

4.2.4 Constrained Optimisation Formulation

We now show that a minor adjustment to SOBER can solve batch BO under unknown constraints. Consider global optimisation subject to unknown constraints, the problem setting proposed by Gelbart et al. (2014):

$$\begin{aligned} x^*_{\text{true}} &= \underset{x\in\mathcal{X}}{\operatorname{argmax}}\, f(x), \\ \text{subject to } & g_l(x) \geq 0, \forall l \in [L], \end{aligned} \tag{10}$$

where f and each g_l are unknown black-box functions[5]. Notably, this approach *permits constraint violations*, in contrast to settings that provide a known set of feasible solutions upfront to permit constraint breaches to be avoided completely (Sui et al., 2015), an assumption invalidated by the black-box nature of our problem. In many practical scenarios, access to such feasible solutions is not available, nor is the feasibility of the problem itself. Additionally, stringent safety-critical constraints could trap these algorithms at local maxima. Nevertheless, our goal is to minimise the total violation incurred throughout the optimisation journey.

We model these black-box constraints using GPs, similarly to how we model the objective function. To maintain focus and brevity, we defer the comprehensive discussion on constraints modeling via GPs to Gelbart et al. (2014); Adachi et al. (2024a), concentrating instead on addressing Eqs.(10) with the given probabilistic models. Consequently, we assess the feasibility of constraints through a probabilistic lens, represented as q_l, rather than through a deterministic but unknown constraint function g_l.

Integrating the following conditions (b)(c)(d) with those in §4.2.1, we refine our tasks:

(b) A tolerance for quadrature precision ϵ_{LP} is given.

(c) The specified batch size n serves as an upper limit, with the actual batch size adaptively modified to achieve the desired precision within ϵ_{LP}.

(d) After selecting the batch querying points $(\mathbf{w}_t^n, \mathbf{X}_t^n)$, each point x within $\mathbf{X}_t^n$ is subject to the probabilistic constraint $q_l(x)$ (and violated w.p., $1 - q_l(x)$). The functions $q_l : \mathcal{X} \to [0, 1]$ are modelled using GPs. Upon querying the true constraints $g_l(\mathbf{X}_t^n)$, we isolate only those points that meet the constraints, along with their corresponding weights, denoted as $\tilde{\pi}_{\text{KQ}} = (\tilde{\mathbf{w}}_t^n, \tilde{\mathbf{X}}_t^n)$. With $\tilde{\mathbf{X}}_t^n \subseteq \mathbf{X}_t^n$, this subset is used for both quadrature and batch BO[a].

a. $\tilde{\mathbf{X}}_t^n = \mathbf{Z}^\top \mathbf{X}_t^n$, where $\mathbf{Z}$ is a vector of Bernoulli random variables with probabilities $q_l(\mathbf{X}_t^n)$

5. Eqs.(10) are only for the inequality constraints yet they can represent the equality constraint using two inequality constraints by bounding the upper and lower limit to be the same threshold.

These conditions lead to a reformulated LP problem:

$$\begin{aligned} \max_{\mathbf{w}\subset\mathbb{R}_{\geq 0}} \quad & \mathbf{w}^\top \left[\alpha_t(\mathbf{X}_t^N) \odot \tilde{q}_t(\mathbf{X}_t^N)\right], \\ \text{subject to} \quad & \left|(\mathbf{w}-\mathbf{w}_t^N)^\top \varphi_j(\mathbf{X}_t^N)\right| \leq \epsilon_{\text{LP}} \sqrt{\lambda_j/(n-2)} \quad (1 \leq {}^\forall j \leq n-2), \\ & (\mathbf{w}-\mathbf{w}_t^N)^\top \tilde{q}_t(\mathbf{X}_t^N) \geq 0, \quad \mathbf{w}^\top \mathbf{1} = 1, \quad \mathbf{w} \geq \mathbf{0}, \quad |\mathbf{w}|_0 \leq n, \end{aligned}$$

where $\epsilon_{\text{LP}} \geq 0$ acts as a *tolerance* level, representing the quadrature precision requirement—lower values indicate higher accuracy. The Hadamard product is denoted by $\odot$, and $\tilde{q}_t(\mathbf{X}_t^N) = \bigodot_{l=1}^{L} q_l(\mathbf{X}_t^N)$ signifies the joint probability of feasibility across all constraints q_l at iteration t, with $\bigodot$ indicating multiple Hadamard products. In scenarios with a single constraint ($L = 1$), the joint feasibility mirrors the individual feasibility, making $\tilde{q}_t = q_1$.

The rationale behind these adjustments includes:

(1) The objective is to maximise the product of the reward α_t and the joint feasibility $\tilde{q}_t$, guiding batch samples towards maximising expected 'safe' reward.

(2) We relaxed equality constraints to inequality ones to accommodate the ϵ_{LP} tolerance. The tolerance parameter ϵ_{LP} controls the trade-off between quadrature precision and the expansion of the solution space to find a larger objective.

(3) Applicable for $n \geq 3$, this new LP formulation introduces an additional constraint, altering the Nyström approximation constraint count in Eq.(8). It requires that the expected joint feasibility of the solution π_{KQ} be equal to or greater than that of the initial candidate set π_t.

Consequently, the solution to this LP problem yields batch samples that not only comply with convex quadrature rules within the given tolerance but also maximise the expected safe reward. The balance between quadrature precision and reward optimisation is tunable via the single parameter ϵ_{LP}, enabling the solution of constrained batch BO tasks within this framework. This approach parallels our prior work (Adachi et al., 2024a) that focuses on the adaptive strategy.

4.2.5 Adaptive Batch Sizes

Our focus now shifts to the aspect of adaptive batch sizes within our methodology. As illustrated in Table 1, traditional algorithms maintain a constant batch size throughout experiments. This fixed strategy can be inefficient due to the dynamic balance between cost and speed—larger batches are more costly, smaller batches lead to slower wall-clock run-times—and the trade-off may change over the run (larger batches are often preferable earlier). To navigate this balance, we introduce an novel approach that adaptively adjusts batch sizes.

The concept is straightforward: we set a fixed tolerance for quadrature precision, ϵ_{LP}, rather than fixing the batch size. This strategy allows batch sizes to automatically adjust to meet predetermined quadrature precision goals, similar to how standard optimisers cease operations based on convergence threshold. Our KQ strategy eliminates the need for exhaustive batch size trials across all possibilities. Furthermore, we extend this approach to

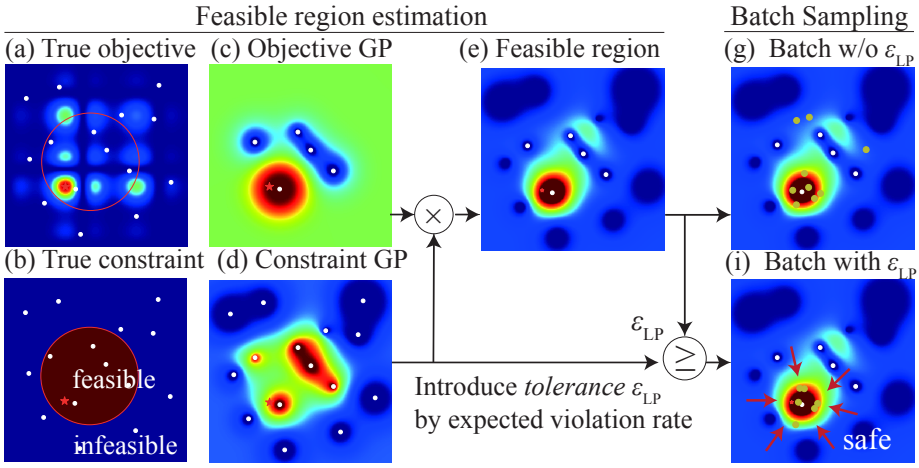


Figure 3: Constrained batch Bayesian optimisation. As the increased violation risk ϵ_{vio} propagates to the tolerance ϵ_{LP}, reward maximisation is subsequently prioritised over quadrature, resulting in safe batch samples.

constrained optimisation scenarios, treating constraint violations as decreases in precision requirements to subsequently adapt batch compositions.

No Constraints. The number of non-zero elements, $|\boldsymbol{w}|_0$, adjusts according to ϵ_{LP}. The intuition of the batch size adaptivity is explicated as:

1. Demanding higher precision decreases the quadrature error tolerance, necessitating a larger sample set for more accurate integration.
2. Conversely, lower precision demands require fewer $|\boldsymbol{w}|_0$ to achieve the desired accuracy.

Elaborating further, the batch size is tied to slack variables in LP solvers. An increase in ϵ_{LP} leads to the deactivation of some inequality constraints, as discussed by Dantzig (2002). The active constraints determine the batch size, often resulting in sparse weights where $|\boldsymbol{w}|_0 < n$. Large fixed batch sizes become inefficient when fewer samples can fulfill the precision criteria. Thus, without resorting to brute-force searches across all potential batch sizes, we can identify the adaptive batch size $|\boldsymbol{w}|_0$.

ϵ_{LP} serves as a control lever for *all* components: batch size, quadrature accuracy, and reward maximisation. Interestingly, its behaviour is not a monotonic decrease in its magnitude. As ϵ_{LP} approaches infinity, the batch size converges to 1, mirroring the sequential BO scenario. An increase in ϵ_{LP} reduces the batch size, as observed in §5.5, embodying a heuristic for adaptive batch sizes. While it ensures meeting a pre-defined worst-case error threshold, it does not promise the optimal outcome based on other established metrics like mutual information. However, as Leskovec et al. (2007) notes, when greedily maximising mutual information under the weighted candidates and a budget constraint (limitation in the number of the total queries T), the approximation factor can be arbitrarily bad. Hence, even popular information-theoretic strategies also cannot achieve a solution within $1 - 1/e$ of the optimal in our problem setting (Li et al., 2022).

Under Constraints. Adaptively adjusting batch size in the presence of probabilistic constraints q_l is examined further. Given the uncertainty in accurately predicting the true constraint g_l, the candidate solution $\mathbf{X}_t^N$ carries a violation risk. The *expected* violation

rate, $\epsilon_{\text{vio}} := 1 - {\mathbf{w}_t^N}^\top \tilde{q}_t(\mathbf{X}_t^N)$, estimates the ratio of non-compliance. Infeasible points are excluded from the quadrature nodes for calculation, diminishing quadrature accuracy. ϵ_{vio}, hence, represents an uncontrollable risk. High-risk scenarios demand cautious exploration to conserve valuable queries, suggesting smaller batches and selections where $\mathbf{X}_t^N$ is likelier to meet the true constraint g_l. Low risk tolerates a more optimistic exploration approach.

We propose an *adaptive* exploration strategy in response to varying risk levels, simply by setting $\epsilon_{\text{LP}} = \epsilon_{\text{vio}}$. This method permits automatic adjustment to exploration safety levels. With a high ϵ_{vio} indicating greater risk, a higher ϵ_{LP} leads to looser quadrature precision, smaller batch sizes, and a solution set more likely to comply with constraints[6]. Thus, a higher ϵ_{LP} ensures safer batch sampling. Conversely, a lower risk level, indicated by a reduced ϵ_{vio}, allows for setting a smaller ϵ_{LP}, facilitating larger batches and more exploratory solutions. Figure 3 showcases this adaptive mechanism: elevated risk ϵ_{vio} influences ϵ_{LP}, leading to safer batch selections. This adaptive strategy effectively balances computational uncertainty and real-world risk, providing a flexible and automated means to navigate between ensuring safety and fostering exploration.

4.3 Theoretical Bounds on Worst-case Error

We now address the theoretical bounds on the worst-case errors in LP formulations, both with and without constraints (referenced in §4.2.3 and §4.2.4).

4.3.1 Kernel quadrature without constraints

In the simplest scenario, we assess the worst-case error bounds within the context outlined in §4.2.3. Here, rather than focusing on an *exact* quadrature, we consider an *approximate* quadrature using a MC estimate with a significantly large number of samples, denoted as the *empirical measure*, $\tilde{\pi}_t := (\mathbf{w}_t^N, \mathbf{X}_t^N) \sim \pi_t$, where ${\mathbf{w}_t^N}^\top \mathbf{1} = 1$ and $\mathbf{w}_t^N \geq 0$. The empirical measure represents a practical approximation of the true measure π_t, which could be a discrete distribution with an innumerable number of candidates $|\mathcal{X}|$, or a continuous distribution. This approach provides a versatile KQ method applicable across various samplable distributions, where the worst-case error is primarily influenced by the Nyström approximation error on the kernel and the distribution approximation error on the empirical measure. Studies such as those by Drineas and Mahoney (2005); Kumar et al. (2012); Hayakawa et al. (2023c) have thoroughly explored error bounds for this approximation:

Theorem 1. *If an n-point convex quadrature $Q_{\pi_t, C_{t-1}}(n)$ satisfies $\pi_{KQ}(\varphi_j) = \tilde{\pi}_t(\varphi_j)$[7] for $1 \leq j \leq n-1$ and $\pi_{KQ}\left(\sqrt{C_{t-1} - \tilde{C}_{t-1}}\right) \leq \tilde{\pi}_t\left(\sqrt{C_{t-1} - \tilde{C}_{t-1}}\right)$, then we have:*

$$\begin{aligned}\operatorname{wce}[Q_{\pi_t, C_{t-1}}(n)] &\leq \operatorname{MMD}_{\mathcal{H}}(\pi_{KQ}, \tilde{\pi}_t) + \operatorname{MMD}_{\mathcal{H}}(\tilde{\pi}_t, \pi_t), \\ &\leq 2 \underbrace{\tilde{\pi}_t\left(\sqrt{C_{t-1} - \tilde{C}_{t-1}}\right)}_{\textit{Nyström approximation}} + \underbrace{\operatorname{MMD}_{\mathcal{H}}(\tilde{\pi}_t, \pi_t)}_{\textit{empirical measure}},\end{aligned}$$

6. Lower $\epsilon_{\text{LP}} \to$ looser LP inequality constraints $\to$ expanding solution space $\to$ larger LP objective $\to$ larger expected reward $\to$ larger feasibility $\to$ safer batch sampling, and vice versa.
7. For brevity, we denote $\pi(f) := \int f(x) \mathrm{d}\pi(x)$.

which is taken from Hayakawa et al. (2023c, Eq. (13)) Notably, if the finite number of candidates $|\mathcal{X}| = N$ is manageable, then $\tilde{\pi}_t = \pi_t$ and $\mathrm{MMD}_{\mathcal{H}}(\tilde{\pi}_t, \pi_t) = 0$, leaving the Nyström approximation error as the sole determinant of the error bound.

Crucially, this outcome is *independent of dimensionality* but hinges on the kernel's spectral decay. Expanding the Nyström samples M diminishes the first term, while increasing candidate numbers N lessens the second term as it enhances $\tilde{\pi}_t$'s approximation of π_t. This indicates that enlarging N and M as permissible by time constraints can tighten the error bounds. However, the spectral decay's impact on maximum information gain in sequential BO is constrained by dimensionality, implying that the overall efficiency of batch-sequential algorithms is similarly affected by high-dimensional spaces, much like other BO methods.

Why Empirical Measure? The necessity for an empirical measure might seem superfluous when the exact integral $\mathbb{E}_{f_{t-1}}[\hat{Z}]$ is available for specific kernels, rendering the second term of the error negligible. There are several justifications for employing an empirical measure, and we now outline them.

The efficacy and rationale of using an empirical measure includes:

(1) **Generality:** An empirical measure can be formulated for any combination of (π, K), extending beyond the scope of traditional BQ methods (Briol et al., 2019). This is particularly relevant as our π_t undergoes sequential updates, potentially lacking a parametric form over x (e.g., in TS scenarios).

(2) **Hypercontractivity:** Insights from the study of random convex hulls and hypercontractivity (Hayakawa et al., 2023b,a) suggest that the requisite number of N might be substantially lower than the actual search space, lending empirical support to the practicality of employing an empirical measure[a].

(3) **Sequential π update:** With each iteration, π narrows towards the global maximum, effectively reducing the number of viable candidates N over time (as demonstrated in Figures 1 and 2, where batch samples in later iterations aggregated around similar locations).

(4) **Normalisation:** The discrete nature of candidates simplifies normalisation, especially since our LFI π_t in Eq.(6) is inherently unnormalised.

a. Still, this is in a slightly different setting and has not been fully understood yet.

4.3.2 Kernel quadrature with constraints

Now consider the setting of constrained optimisation. Here, we introduce a tolerance for quadrature precision, necessitating an adjustment to the theoretical bound as follows:

Proposition 1. *Under the setting in the §4.2.4, let $\mathbf{w}_*$ be the optimal solution of the LP, and let $\mathbf{X}_t^n$ be the subset of $\mathbf{X}_t^N$, corresponding to the non-zero entries of $\mathbf{w}_*$ (denoted by $\mathbf{w}_t^n$). Suppose that $\tilde{\mathbf{X}}_t^n$ is given by a random subset of $\mathbf{X}_t^n$, where each point x satisfies the constraints with probability $\tilde{q}_t(x)$, and let $\tilde{\mathbf{w}}_t^n$ be the corresponding weights. Then, we have*

$$\mathbb{E}[\tilde{\mathbf{w}}_t^{n\top} \alpha_t(\tilde{\mathbf{X}}_t^n)] \geq \mathbf{w}_t^{n\top} \big[\alpha_t(\mathbf{X}_t^n) \odot \tilde{q}_t(\mathbf{X}_t^n)\big], \tag{11}$$

and, for any function f_{t-1} in the RKHS with kernel C_{t-1},

$$\mathbb{E}\left[\left|\tilde{\mathbf{w}}_t^{n\top} f_{t-1}(\tilde{\mathbf{X}}_t^n) - \mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n)\right|\right] \leq (\epsilon_{\text{vio}} K_{\max} + 2\epsilon_{\text{nys}} + \epsilon_{\text{LP}})\|f_{t-1}\|, \tag{12}$$

where $\|f_{t-1}\|$ is the RKHS norm of f_{t-1}, $K_{\max} := \max_{x \in \boldsymbol{X}_t^N} \sqrt{C_{t-1}(x,x)}$, and $\epsilon_{\text{vio}} := 1 - \mathbf{w}_t^{N\top} \tilde{q}_t(\mathbf{X}_t^N)$ is the expected violation rate with respect to the empirical measure given by $(\mathbf{w}_t^N, \mathbf{X}_t^N)$, and $\epsilon_{nys} := \max_{x \in \boldsymbol{X}_t^N} |\tilde{C}_{t-1}(x,x) - C_{t-1}(x,x)|^{1/2}$.

The proof is given in Appendix A. This proposition elucidates that a quantitative approximation of the two tasks highlighted in §4.2.4 is achievable concurrently. It guarantees that, at minimum, the expected reward of the original batch is matched while ensuring the resulting measure $\tilde{\pi}_{\text{KQ}}$ (potentially non-probabilistic) conforms to the functions within the RKHS, all within a predefined error margin. This approach offers a quantitative framework for navigating the dual challenges of reward maximisation and constraint satisfaction in constrained optimisation scenarios.

4.3.3 Robustness against misspecified RKHS

Finally, we address the robustness of our approach to misspecified RKHS. In BO, a common source of misspecification arises from inaccurate estimation in the hyperparameters of GPs. While the BO community has developed robust strategies (Berkenkamp et al., 2019; Bogunovic and Krause, 2021; Ziomek et al., 2024), these are predominantly adaptations of the UCB and do not universally apply across all AFs. Conversely, the KQ community has thoroughly explored misspecification, offering robust estimations of worst-case errors for a broad range of conditions (Kanagawa et al., 2016; Oates et al., 2017; Karvonen et al., 2018; Kanagawa et al., 2020). Notably, the our KQ method also guarantees robustness against misspecification (Appendix B.4 in Hayakawa et al. (2022), using $|\pi_{\text{KQ}}|_{\text{TV}} = |\tilde{\pi}_t|_{\text{TV}} = 1$):

Proposition 2. *Under the setting in the §4.2.3, let $\mathcal{H}_{K_{mis}}$ be the misspecified RKHS and $\tilde{f} \in \mathcal{H}_{K_{mis}}$ be a function in the misspecified RKHS, and π_{KQ} be a quadrature rule applied to a function $f \notin \mathcal{H}_{K_{mis}}$, leading only to the following bound via triangle equality and standard integral estimates:*

$$\begin{aligned}
&\left| \int f(x) d\pi_{KQ}(x) - \int f(x) d\tilde{\pi}_t(x) \right| \\
&\leq (|\pi_{KQ}|_{TV} + |\tilde{\pi}_t|_{TV}) \sup_{x \in \mathcal{X}} |f(x) - \tilde{f}(x)| + \|\tilde{f}\|_{\mathcal{H}_{K_{mis}}} \text{wce}[Q_{\tilde{\pi}_t, K_{mis}}(n)] \\
&= 2 \sup_{x \in \mathcal{X}} |f(x) - \tilde{f}(x)| + \|\tilde{f}\|_{\mathcal{H}_{K_{mis}}} \text{wce}[Q_{\tilde{\pi}_t, K_{mis}}(n)].
\end{aligned}$$

The first inequality in Proposition 2 highlights the advantage of employing convex weights within the KQ rule. Non-convex weights can inflate the total variation $|\pi_{\text{KQ}}|_{\text{TV}}$, potentially resulting in significant integration errors. Unlike traditional BQ, which adopts negative weights and thus suffer from misspecification challenges (Huszár and Duvenaud, 2012), the use of convex weights as delineated here mitigates such risks at least within uniform bounds, underscoring the robustness of our KQ approach against RKHS misspecification.

4.4 In Practice. How to Solve LP and Apply SOBER

4.4.1 How to solve LP problem

We introduce the following two algorithms to solve the above LP problems. We detail two algorithms for addressing LP problems: the *recombination* algorithm for unconstrained settings (§4.2.3) and the *LP solver* for constrained scenarios (§4.2.4).

Recombination. The recombination algorithm (Litterer and Lyons, 2012; Tchernychova, 2015; Cosentino et al., 2020; Hayakawa et al., 2022) offers an efficient algorithm to solve scenarios that meet the following conditions: (i) absence of constraints, (ii) exact solution requirements ($\epsilon_{\text{LP}} = 0$), and (iii) a fixed batch size ($|\mathbf{X}_t^n| = n$). Recombination is an efficient solver for special linear programming task, differing from a general solver like the simplex method. Recombination leverages Carathéodory's theorem for fast computation through mere matrix operations, with further details available in Tchernychova (2015). Its computational complexity, $\mathcal{O}(C_\varphi N + n^3 \log(N/n))$, where C_φ represents the cost of evaluating $(\varphi_j)_{j=1}^{n-1}$ at any given point, is the most efficient for the stated conditions. Given that typical batch BO settings align with these prerequisites, recombination is the recommended primary solver.

General LP solver. For broader applications, including those with constraints, a general LP solver using the simplex method becomes relevant. In unconstrained scenarios with adaptive batch sizes, setting ϵ_{LP} to a minimal value like 10^{-8} is recommended to minimise numerical errors linked to floating-point precision limits. In the presence of constraints, ϵ_{LP} auto-adjusts based on the estimated risk level, $\epsilon_{\text{LP}} = \epsilon_{\text{vio}}$, eliminating the need for manual tolerance settings. Additionally, the incorporation of randomised singular value decomposition (SVD; Halko et al. (2011)) for Nyström approximation enhances computational speed, with practical performance surpassing the theoretical time complexity of $\mathcal{O}(NM + M^2 \log n + Mn^2 \log(N/n))$ as noted in Hayakawa et al. (2022). This approach ensures that LP problems, whether constrained or not, can be solved with efficiency and precision, making it an essential component of the SOBER algorithm's practical application.

4.4.2 How to sample from π_t

Algorithm 2 Sequential importance resampling.

Require: domain prior π_0, target distribution π_t

1: Initial sampling $\tilde{\mathbf{X}}_t^N \sim \pi_0$:
2: Compute normalised importance weights $\tilde{\mathbf{w}}_t^N = \text{Normalise}(\pi_t(\tilde{\mathbf{X}}_t^N)/\pi_0(\tilde{\mathbf{X}}_t^N))$
3: Maximum likelihood Estimate $\tilde{\pi}_0 \leftarrow \text{MLE}(\pi_0, \tilde{\mathbf{X}}_t^N, \tilde{\mathbf{w}}_t^N)$
4: Refined resampling $\mathbf{X}_t^N \sim \tilde{\pi}_0$:
5: Compute normalised importance weights $\mathbf{w}_t^N = \text{Normalise}(\pi_t(\mathbf{X}_t^N)/\tilde{\pi}_0(\mathbf{X}_t^N))$
6: **return** empirical measure $\tilde{\pi}_t = (\mathbf{w}_t^N, \mathbf{X}_t^N)$

As elucidated in §4.3.1, we use the empirical measure $\tilde{\pi}_t(x)$ to approximate π_t. Essentially, constructing empirical measure $\tilde{\pi}_t(x) = (\mathbf{w}_t^N, \mathbf{X}_t^N) \sim \pi_t(x)$ is sampling from π_t. For directly samplable distributions $\pi_t(x)$, we generate i.i.d. samples from $\pi_t(x)$ and assign

$\mathbf{w}_t^N$ as a discrete uniform distribution $\{w_i \in \mathbf{w}_t^N \mid w_i = 1/N, \forall i \in [N]\}$. In cases where direct sampling is not feasible, the classic *sequential importance resampling (SIR)* method (Kitagawa, 1993) becomes invaluable, as detailed in subsequent sections and Algorithm 2. Our Python library simplifies this process, allowing users to easily select the domain corresponding to their optimisation objectives without diving into the technicalities.

Thompson Sampling. The TS variant of SOBER, primarily for baseline comparison, incurs a time-intensive sampling process, a challenge shared by other TS-based algorithms (Nava et al., 2022). Constructing the empirical measure $\tilde{\pi}_t$ with TS is straightforward: draw N exhaustive TS samples and set $\mathbf{w}_t^N$ as a discrete uniform distribution. Although recent advancements in fast posterior sampling methods (Wilson et al., 2020; Lin et al., 2023) slightly mitigate the exhaustive nature of this sampling process, the SOBER-LFI procedures are markedly simpler and more efficient.

Discrete Domain. Generally, a discrete domain implies a discrete distribution with equal weights, $\nu = 1/N \sum_{i=1}^{N} \delta_{x_i}$. If all candidates $\mathcal{X}$ can be enumerated (e.g., in drug discovery, where all viable candidates are known), using the parametric distribution $\mathbb{P}(\mathcal{X}) = \mathcal{U}(\mathcal{X})$, where $\mathcal{U}$ represents the discrete uniform distribution, maintains generality without loss of precision. Herein, the distribution class generating $x \sim \mathbb{P}(\mathcal{X})$ is termed *the domain prior* $\pi_0(x) = \mathbb{P}(\mathcal{X})$. In combinatorial optimisation of binary variables, for instance, the domain prior is essentially a Bernoulli distribution with equal weights. For discrete variables with categories (e.g., choices among $\{0, 1, 2, 3, 4\}$), this equates to a categorical distribution with equal weights, enabling the use of Bernoulli, categorical, or general discrete distributions as the domain prior π_0 through SIR to construct $\tilde{\pi}_t$.

Continuous Domain. The continuous domain presents more complex challenges for SOBER-LFI due to the absence of a defined parametric form for the domain prior. For computational simplicity and efficiency, we opt for a Gaussian Mixture Model (GMM) (Xuan et al., 2001) as the samplable parametric model for the domain prior π_0. Given the universal approximation capabilities of GMM (Stergiopoulos, 2017), it serves as a suitable proposal distribution for the SIR procedure. Note that our end goal is importance sampling $\tilde{\pi}_t$, negating the need for an exact match of GMM proposal distribution to π_t.

Mixed Domain. In scenarios with a mixture of continuous and discrete variables, assuming independent distributions for each segment is practical. For example, if the initial three dimensions are continuous and the subsequent two are binary, the domain prior could be modelled as a product of continuous and discrete distributions $\pi_0 \propto \mathrm{GMM}(\mathbf{X}_{:3}) \cdot \mathrm{B}(\mathbf{X}_{4:5})$, where GMM and B represent the GMM and Bernoulli distribution, respectively, and $\mathbf{X}_{k:k'}$ specifies the input dimensions from k to k'.

Expert Knowledge. Engagement with human experts can yield beliefs about the global maximum's location $\mathbb{P}(\hat{x}_t^*)$, derived from experience, expertise, or prior knowledge (e.g., Hvarfner et al. (2022); Adachi et al. (2024b)). In such instances, this expert model is directly employed as the domain prior π_0, facilitating a tailored approach to leveraging domain-specific insights in optimisation tasks.

Batch Active Learning and Bayesian Quadrature Tasks. In the context of pool-based batch active learning (AL), the primary goal is the information gain of the model

experiments	task	syn. /real	objective	dimensions d cont.	dimensions d disc.	space $\mathcal{X}$	batch n	const-raint L	kernel K	domain prior π_0 cont.	domain prior π_0 disc.
Branin-Hoo	BO	syn.	$\max(f)$	2	-	cont.	$= 30$	-	RBF	$\mathcal{U}(-3, 2)$	-
Ackley	BO	syn.	$\log_{10} \min(f)$	3	20	mixed	$= 200$	-	RBF	$\mathcal{U}(-1, 1)$	Bernoul.
Rosenbrock	BO	syn.	$\log_{10} \min(f)$	1	6	mixed	$= 100$	-	RBF	$\mathcal{U}(-4, 11)$	Categor.
Hartmann	BO	syn.	$\log_{10} \max(f)$	6	-	cont.	$= 100$	-	RBF	$\mathcal{U}(0, 1)$	-
Snekel	BO	syn.	$\log_{10} \max(f)$	4	-	cont.	$= 100$	-	RBF	$\mathcal{U}(0, 10)$	-
Pest	BO	real.	$\min(f)$	-	15	disc.	$= 200$	-	RBF	-	Categor.
MaxSat	BO	real.	$\min(f)$	-	28	disc.	$= 200$	-	RBF	-	Bernoul.
Ising	BO	real.	$\log_{10} \min(f)$	-	24	disc.	$= 100$	-	RBF	-	Bernoul.
SVM	BO	real.	$\min(f)$	3	20	mixed	$= 200$	-	RBF	$\mathcal{U}(0, 1)$	Bernoul.
Malaria	BO	real.	$\log_{10} \min(f)$	molecule		disc.	$= 100$	-	Tanimoto	-	Categor.
Solvent	BO	real.	$-\max \log_{10} f$	molecule		disc.	$= 200$	-	Tanimoto	-	Categor.
Branin-Hoo	cBO	syn.	$\max(f)$	2	-	cont.	≤ 20	2	RBF	$\mathcal{U}(-3, 2)$	-
Ackley	cBO	syn.	$\log_{10} \min(f)$	3	20	mixed	≤ 200	2	RBF	$\mathcal{U}(-1, 1)$	Bernoul.
Hartmann	cBO	syn.	$\log_{10}$ regret	6	-	cont.	≤ 5	2	RBF	$\mathcal{U}(0, 1)$	-
Pest	cBO	real.	$\min(f)$	-	15	disc.	≤ 200	2	RBF	-	Categor.
Malaria	cBO	real.	$\log_{10} \min(f)$	molecule		disc.	≤ 100	4	Tanimoto	-	Categor.
FindFixer	cBO	real.	$\max(f)$	node		graph	≤ 100	3	graph	-	Categor.
TeamOpt	cBO	real.	$\log_{10}$ regret	subgraph		graph	≤ 100	3	graph	-	Categor.
2 RC	BQ	real.	$\int f(x) \mathrm{d}\pi_0(x)$	6	-	cont.	$= 100$	-	RBF	Gaussian	-
5 RC	BQ	real.	$\int f(x) \mathrm{d}\pi_0(x)$	12	-	cont.	$= 100$	-	RBF	Gaussian	-

Table 2: Experimental Setup. Task: either BO, constrained BO (cBO), or BQ. Syn./real: synthetic or real-world. Dimensions: the number of dimensions over input space categorised into continuous (cont.), discrete (disc.). Batch: the fixed batch size $= n$ or upper bound of adaptive batch size $\leq n$. Constraint: the number of constraints L. Prior: the domain prior π_0, Bernoul. and Categor.: Bernoulli and categorical distributions with equal weights. Special kernels are used: Tanimoto kernel (Ralaivola et al., 2005) for molecules and the diffusion graph kernel (Zhi et al., 2023) for graphs.

parameters for more accurate prediction. Specifically, within a GP model, this often translates to batch uncertainty sampling. We usually operate under the assumption that a set of unlabelled candidates, $\mathcal{X}$, is provided and can be fully enumerated. Consequently, in the batch AL scenario, our target distribution, π_t, is defined as a uniform distribution over the available candidates, $\pi_t := \mathcal{U}(\mathcal{X})$. Conversely, batch BQ presupposes a prior distribution, π_0, that remains constant throughout the process. Therefore, in the batch BQ framework, our distribution π_t aligns with this predefined prior, $\pi_t = \pi_0$. In both tasks, the distribution π_t is stationary. As such, the main distinction between batch AL, BQ, and BO lies in mere definition of π_t, and our SOBER framework is applicable for these tasks.

5 Experiments

We now move into the evaluation of our algorithm, SOBER, through both synthetic and real-world examples. Initially, we empirically analyse our proposed methodologies, focusing on measure convergence, robustness against misspecified RKHS, scalability, hyperparameter sensitivity, and adaptability of batch sizes. Subsequently, we juxtapose SOBER's performance in terms of regret convergence with that of popular baselines.

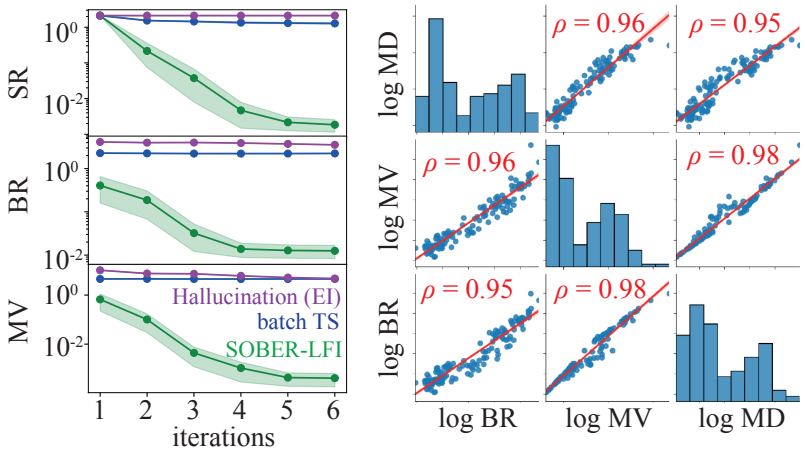


Figure 4: Correlations between Bayesian regret (BR) and measure optimisation. (Left) the convergence of simple regret (SR), BR, and mean variance (MV) for three batching methods. (Right) the linear correlations between mean distance (MD), MV, and BR.

Our assessment spans *20 experiments, benchmarked against 17 baselines*; 14 baselines for BO; random, batch TS (Kandasamy et al., 2015), decoupled TS (Wilson et al., 2020), DPP-TS (Nava et al., 2022), TurBO (Eriksson et al., 2019), GIBBON (Moss et al., 2021), hallucination (Azimi et al., 2010), local penalisation (LP[8]; González et al. (2016)), B3O (Nguyen et al., 2016), cEI (Letham et al., 2019), PESC (Hernández-Lobato et al., 2016), SCBO (Eriksson and Poloczek, 2021), cTS (Eriksson and Poloczek, 2021), and PropertyDAG (Park et al., 2022), and 3 baselines for BQ; batchWSABI (Wagstaff et al., 2018), BASQ (Adachi et al., 2022), and logBASQ (Adachi et al., 2023b).

The 20 experimental setups are detailed in Table 2, comprising 11 BO experiments (5 synthetic and 6 real-world data sets), 7 constrained BO experiments (3 synthetic and 4 real-world data sets), and 2 real-world experiments tailored to BQ. Comprehensive experimental methodologies are provided in Appendix B. Our implementations leverage PyTorch-based libraries (Paszke et al., 2019; Gardner et al., 2018; Balandat et al., 2020; Griffiths et al., 2023), with all tests averaged over 10 iterations and executed in parallel on multicore CPUs for fair comparison. We note that GPU could further enhance SOBER's performance. Experimental outcomes are presented as the mean $\pm$ standard error of the mean, adhering to default SOBER hyperparameters $N = 20,000, M = 500$, unless otherwise specified.

To facilitate comparisons in discrete or mixed domains where certain algorithms (e.g., TurBO, GIBBON, Hallucination, and LP) encounter challenges due to combinatorial complexities, we employ a thresholding approach, optimising discrete variables as continuous ones and then categorising solutions through nearest neighbours. For the special yet popular tasks, such as drug discovery and graph tasks, non-Euclidean spaces or specialised kernels preclude the application of most algorithms.

5.1 Measure Convergence Analysis

We empirically investigated the relationship between regret and π_t convergence. Recall that the empirical measure $\tilde{\pi}_t = (\mathbf{w}_t^N, \mathbf{X}_t^N)$ is sampled from π_t, and the KQ rule $\pi_{\text{KQ}} = (\mathbf{w}_t^n, \mathbf{X}_t^n)$ is the subset further extracted from $\tilde{\pi}_t$. As such, all measures approximate the same distribution, $\pi_t \sim \tilde{\pi}_t \sim \pi_{\text{KQ}}$, with only the level of discretisation differing. We consider the following two metrics for π_t convergence: mean Euclidean distance (MD) $|x^*_{\text{true}} - \mathbb{E}_x[\pi_t(x)]|$ and mean variance (MV) $\mathbb{V}_x[\pi_t(x)]$, which can be approximated using $\tilde{\pi}_t(x)$:

$$\mathbb{V}_x[\pi_t(x)] \approx \mathbf{w}_t^{N\top} \text{diag}\Big[(\mathbf{X}_t^N - \mathbb{E}_x[\tilde{\pi}_t(x)])^\top (\mathbf{X}_t^N - \mathbb{E}[\tilde{\pi}_t(x)])\Big],$$

where $\mathbb{E}_x[\pi_t(x)] \approx \mathbb{E}_x[\tilde{\pi}_t(x)] = \mathbf{w}_t^{N\top} \mathbf{X}_t^N$ is the barycenter of π_t, MV and MD correspond to the convergence of π_t, with a smaller value indicating convergence to the global maximum.

We compared these two metrics against BR and simple regret, $f(x^*_{\text{true}}) - \max_{x \in \mathbf{X}_t} f(x)$. Experiments were conducted using the Ackley (see Table 2) over six iterations with 20 repeats (120 data points). Firstly, the left side in Figure 4 shows a similar convergence trend for SR, BR, and MV, particularly noting that SOBER-LFI converges surprisingly quickly. The linear correlation matrix on the right implies that both MD and MV are highly correlated with BR, clearly explaining that π_t convergence in Eq.(5) is a good proxy for BR. π_t (the MC estimate of $\hat{x}_t^*$) shrinks toward the true global maximum, x^*_{true}, with smaller variance (more confidence), and both are linearly correlated with BR minimisation.

One potential explanation for the significant performance improvement of SOBER-LFI is the synergy between the explorative KQ approach and the exploitative LFI synthetic likelihood. As illustrated in Figure 2, the LFI exhibits greater peakedness around the current maximum, $\hat{y}_t^*$, compared to the TS distribution. Such a distribution is likely to result in smaller MV. Our KQ method is capable of robustly selecting small areas of uncertainty, even with such a peaked distribution (refer back to Figure 2). In essence, the exploitative nature of LFI contributes to the reduction of MV, and consequently, to a decrease in BR, whereas the KQ facilitates robust exploration under peaked distribution.

5.2 Robustness Analysis

Misspecified Domain Prior. We evaluated the robustness of our approach to a misspecified domain prior, π_0, by introducing noise to the hyperparameters of π_0 as depicted in Figure 5(i). Note that, although noise is added to the π_0 hyperparameters, they are quickly updated via SIR, suggesting that the system should be resilient to initial misspecifications. Hence, this experiment primarily assesses robustness against skewed initial sample configurations, considering we generate the initial 100 samples by drawing from π_0. For continuous optimisation, we employed a uniform distribution as the non-informative prior (reference) and a truncated Gaussian distribution as the misspecified (biased) prior. We maintained a fixed covariance matrix, $\mathbf{I}_{d\times d}$, but introduced noise to the mean vector, $\mu_\pi = \sigma\epsilon$, where $\epsilon \sim \mathcal{U}(0,1)$ represents uniform noise and σ denotes the noise scale. In the case of binary optimisation, we used a Bernoulli distribution, with its probability vector treated as the stochastic variable $p = \sigma\epsilon$ ($p = 0.5$ denotes uniform). In both scenarios, as σ approaches

8. Only within the exerimental section, LP refers to local penalisation, and we use LP for linear programming for other sections.

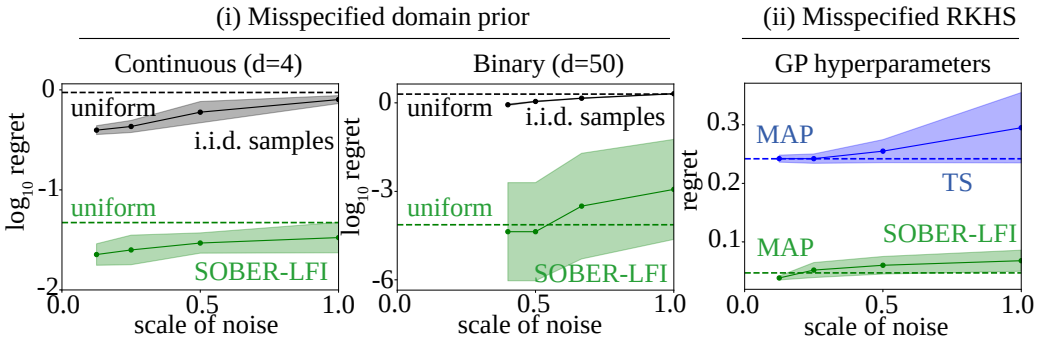


Figure 5: Robustness analysis on Ackley ($n = 200$, $\log_{10}$ regret at 10th iteration) (i) misspecified domain prior: The left and middle experiments are examined misspecified domain prior for continuous and binary optimisation. (ii) Misspecified RKHS: We added noise to the GP hyperparameters that were tuned by MLE. In all misspecification cases, SOBER showed great resilience against misspecification noise.

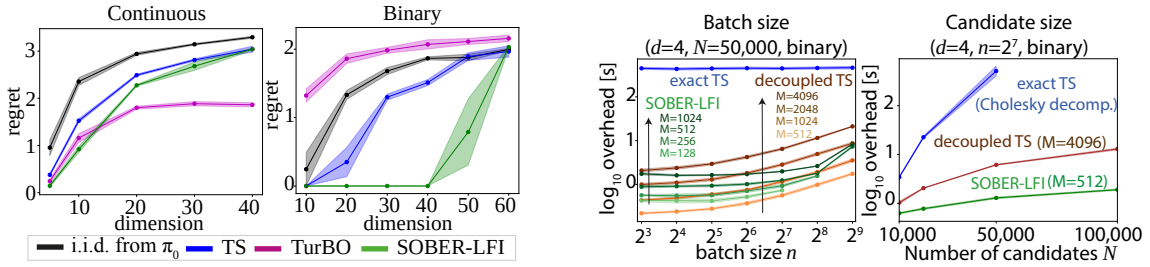
zero, the system converges to the global maximum, $x^*_{\text{true}} = [0]^d$, indicating that a smaller σ favours the identification of the global maximum. The observed simple regrets remain nearly constant across different noise scales and significantly outperform i.i.d. batch samples drawn from the domain prior π_0.

Misspecified RKHS. We investigated the robustness against a misspecified RKHS, specifically misfit GP hyperparameters, as shown in Figure 5(ii). Referring to §4.3.3, the worst-case error estimate in Proposition 2 is guaranteed to be uniformly bounded. Noise was introduced to the GP hyperparameters, which were initially optimised using type-II MLE with the BoTorch optimiser (Balandat et al., 2020). The hyperparameters were adjusted as $\theta := \theta_{\text{MLE}}(1 + \sigma_\theta \epsilon)$, where $\epsilon \sim \mathcal{U}(-0.5, 0.5)$. The dashed lines represent the scenarios without noise (MAP cases). While the regret associated with batch TS (Kandasamy et al., 2015) worsened and exhibited greater variance with increasing noise scale, SOBER-LFI achieved a plateau, indicating uniform robustness against the worst-case error. This demonstrates the susceptibility of TS to model misspecification, as exemplified in Figure 1.

5.3 Scalability Analysis

Dimensional Scalability. We assessed dimensional scalability by comparing our method against TurBO (Eriksson et al., 2019), a widely recognised method for high-dimensional BO. Since TurBO is designed solely for continuous domains, we adapted its algorithm by thresholding (recall §5). In the continuous domain, as shown in Figure 6(a), while TurBO exhibits superior performance in dimensions exceeding 15, SOBER-LFI is more effective in lower dimensions. In binary optimisation, SOBER-LFI surpasses all baselines, even in 60 dimensions. This is because the binary space has fewer potential candidates, 2^d, than the continuous space, allowing the hypercontractivity of the random convex hull to ensure that the empirical measure, $\tilde{\pi}_t$, adequately spans the entire domain.

Computational Complexity. We evaluated the computational overhead for batch queries. As detailed in §4.4.1, the complexity is $\mathcal{O}(C_\varphi N + n^3 \log(N/n))$. At first glance, the cubic term related to batch size, n, appears unscalable; however, it is actually competitive, com-



(a) dimensional scalability (b) Computational complexity

Figure 6: Scalability Analysis: (a) The regret over a varied number of dimensions of noisy Ackley function on both continuous and binary domains (batch size $n = 200$, and regrets are at 10th iterations. (b). The overhead time over varied batch size n and the number of candidates N. The number of function samples M approximates original kernel, which corresponds to number of samples for Nyström approximation in SOBER, and the ones for random Fourier features (RFF) approximation in decoupled TS.

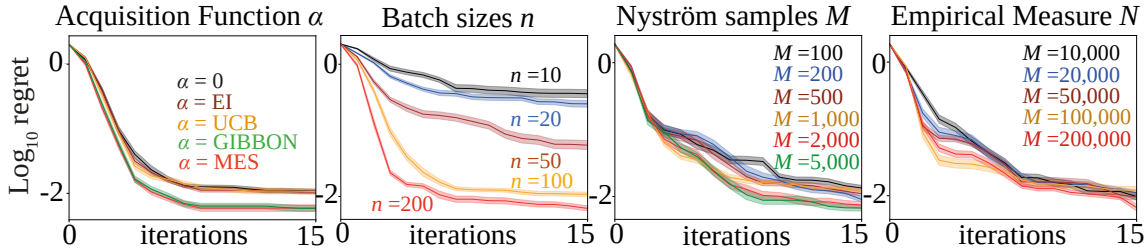
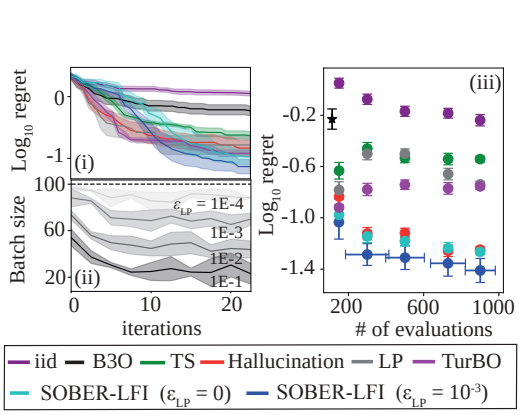


Figure 7: Hyperparameter sensitivity analysis using the Ackley function.

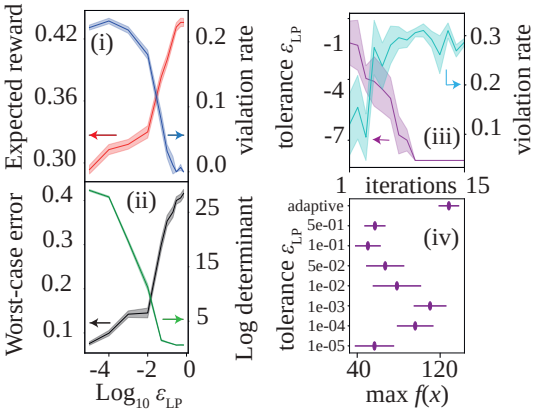
pared to TS and its variants. This is attributed to our candidate size, $N \gg n$, which makes the linearity in N more impactful than the cubic term in n^3 for practical applications. Figure 6(b) illustrates that SOBER-LFI significantly outpaces exact TS, which relies on Cholesky decomposition with a complexity of $\mathcal{O}(N^3)$. The fast variant using random Fourier features (decoupled TS, Wilson et al. (2020)) achieves linearity with N, similar to our approach. However, it incurs a larger approximation error than the Nyström approximation (Yang et al., 2012), necessitating more function samples, M, than Nyström. As Figure 6(b) shows, within the practical range of parameter sets ($N = 20,000$, $M = 512$, $n \leq 2^9$), our SOBER-LFI achieves fast computation. Yet, the cubic term, n^3, escalates quickly for $n > 1,000$, limiting scalability to such batch sizes. Nevertheless, considering the maximum batch sizes used in high-throughput drug discovery are typically 384 compounds (Carpentier et al., 2016), we argue that SOBER-LFI remains sufficiently scalable for practical applications.

5.4 Hyperparameter Sensitivity

The hyperparameter sensitivity of SOBER-LFI was examined using the Ackley, focusing on the effects of AFs (α), batch size (n), the number of Nyström samples (M), and empirical measure sizes (N). The baseline conditions were set to $n = 100$, $\alpha = 0$ (no acquisition functions as reward), $M = 500$, and $N = 20,000$. For AFs, the information-theoretic AFs



(a) Adaptive batch sizes on Hartmann ($d = 6$)



(b) Adaptive safe exploration on Branin ($d = 2$)

Figure 8: Adaptability Analysis: (a)(i) convergence plot with ($n \leq 5$). (ii) batch size variability ($n \leq 100$). The tolerance is set ($\epsilon_{\text{LP}} = 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}$). (iii) Total queries vs. simple regret at the last iteration results of (i)(ii). For fixed batch size baselines, the mean batch size of SOBER-LFI with $\epsilon_{\text{LP}} = 10^{-2}$ is used ($n = 5, 30, 50, 73, 90$). (b) Tolerance effect on constrained batch BO: the balance between (i) violation rate and expected reward, and (ii) worst-case error and log determinant. (iii) Tolerance adaptively controls violation rate, and (iv) outperforms the fixed cases. (i)(ii)(iii) are the two Y-axis plots where the colour and arrow indicate which Y axis to see.

significantly enhances the convergence rate, whereas UCB/EI show no substantial change. This can be because MES function gives more dissimilar reward function shape to π (= PI) than EI/UCB. Notably, our SOBER library integrates seamlessly with the popular GPyTorch/BoTorch library, allowing users to directly use AF and kernel instances defined by these libraries. More details can be found in our GitHub tutorials.

Regarding batch size (n), we observed an improvement in the convergence rate proportional to the batch size. Although this increase seems intuitive, it is not consistently observed in existing baselines (e.g., see Figure 6 in GIBBON (Moss et al., 2021)). For the Nyström samples (M) and empirical measure sizes (N), larger sample numbers correspond to faster convergence, reflecting our bound in Theorem 1. Specifically, a larger M reduces the Nyström approximation error (the first term), and a larger N decreases the empirical measure approximation error (the second term). This straightforward relationship between MC estimate error and sample size is not always evident in existing baselines (e.g., max-value entropy search, MES; Wang and Jegelka (2017), see §2.3 in Takeno et al. (2023)). However, an increase in sample numbers also results in higher computational overhead, as discussed in §5.3. Our default settings are competitive for real-world experiments discussed later, though they can be adjusted to balance the cost of queries (Adachi et al., 2022)[9].

5.5 Adaptability Analysis

Within this section, we have explored the $\epsilon_{\text{LP}} > 0$ setting described in §4.2.4.

9. See guidelines in Appendix 2.2.2 in Adachi et al. (2022)

Adaptive Batch Sizes. Our initial investigation focused on the effectiveness of adaptive batch sizes. To facilitate a comparison with B3O (Nguyen et al., 2016), the only other baseline method employing adaptive batch sizes, we examined batch BO *without* unknown constraints, i.e., $\tilde{q}(x) = 1$, albeit with $\epsilon_{\text{LP}} > 0$. As shown in Figure 8(a), SOBER-LFI, employing adaptive batch sizes, consistently outperformed the baseline methods across all experiments. An increase in ϵ_{LP} led to a reduction in batch size, which consistently decreased over iterations for all values of ϵ_{LP}. This pattern indicates that SOBER-LFI initially requires a larger number of exploratory samples before narrowing its search space for exploitation. When compared to methods using fixed batch sizes within the same total cost framework, SOBER-LFI achieved lower regret, surpassing even the original SOBER with fixed batch sizes. Unlike B3O, which tends towards small batch sizes (around 4), the batch size of SOBER-LFI is tunable based on n and ϵ_{LP}.

Adaptive Safe Exploration. We then examined the influence of the expected violation rate, ϵ_{vio}, in constrained BO, treating it as a time-varying tolerance, $\epsilon_{\text{vio}} = \epsilon_{\text{LP}}$ with $\epsilon_{\text{LP}} > 0$. The main findings are depicted in Figure 8(b).

Four key metrics:

(1) *The expected reward* (LP objective): A proxy for the safety level of our exploration strategies.

(2) *The violation rate* $1 - |\tilde{\mathbf{X}}_{\text{batch}}|/|\mathbf{X}_{\text{batch}}|$: A proxy for the actual safety level achieved during exploration.

(3) *The worst-case error* $\text{wce}[Q_{\pi_t, C_{t-1}}(|\tilde{\mathbf{X}}_{\text{batch}}|)]$: the precision of quadrature.

(4) *log determinant* $\log|K(\tilde{\mathbf{X}}_{\text{batch}}, \tilde{\mathbf{X}}_{\text{batch}})|$: a proxy for the batch sample diversity.

We evaluated the impact of ϵ_{LP} on these metrics, aligning ϵ_{LP} with ϵ_{vio} to enable adaptive exploration in line with specified risk levels, ϵ_{vio}, illustrated in Figures 8(b)(i)(ii). As risk levels increase, ensuring safety becomes a priority, leading to an uptick in the expected reward and a corresponding reduction in the violation rate, signifying safer exploration practices. Numerically, a higher risk level correlates with an increased worst-case error, indicating a relaxation in precision requirements, and a reduced log determinant, suggesting a decrease in the diversity of batch samples due to the proximity of selected points ($\mathbf{X}_t^N$) to each other. Conversely, lower risk levels favour a more optimistic and exploratory approach. Our results affirm that setting ϵ_{LP} equal to ϵ_{vio} allows our batch exploration strategy to adeptly adjust to varying risk levels.

Additionally, we observed the evolution of the expected violation rate, ϵ_{vio}, throughout the optimisation process. As depicted in Figure 8(b)(iii), ϵ_{vio} starts high and gradually decreases, highlighting an initial focus on safe data collection before shifting towards broader exploration. This strategy is reminiscent of 'safe' BO approaches like those proposed by (Sui et al., 2015), which have shown strong empirical performance and theoretical support (e.g., Figure 4 in Xu et al. (2023)). The inherent adaptability of our method to adjust batch sizes and tolerance levels showcases its efficiency, particularly as demonstrated by the more fast convergence compared to fixed tolerance approaches in Figure 8(b)(iv). Interestingly, the most effective fixed tolerance was $\epsilon_{\text{LP}} = 10^{-3}$, indicating that SOBER-LFI surpasses the performance of the exact case ($\epsilon_{\text{LP}} = 0$) under constraints, even with a fixed tolerance.

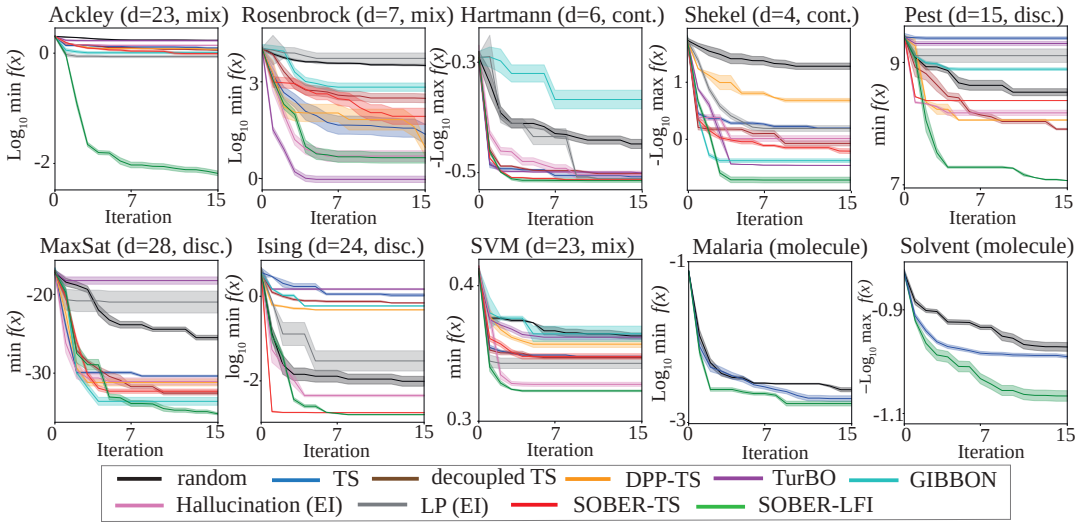


Figure 9: Tolerance effect on constrained batch BO on Branin ($d = 2$): the balance between (a) violation rate and expected reward, and (b) worst-case error and log determinant. (c) Tolerance adaptively controls violation rate, and (d) outperforms the fixed cases. (a)(b)(c) are the two Y-axis plots where the colour and arrow indicate which Y axis to see.

5.6 Discrete and Mixed Variable Experiments

baselines	Ackley	Rosenbrock	Hartmann	Shekel	Pest	MaxSat	Ising	SVM	Malaria	Solvent	Mean rank
Random	-1.92	-1.96	-1.26	-1.17	-1.92	-1.89	-1.64	0.82	1.40	1.49	-
batch TS	2.71	3.10	2.79	2.86	3.00	3.70	3.22	3.36	2.71	2.85	3.1
decoupled TS	**2.20**	**2.04**	**2.01**	**2.04**	3.17	3.22	3.65	3.90	-	-	2.6
DPP-TS	4.85	4.56	4.35	4.62	5.67	4.49	4.73	4.73	-	-	7.4
TurBO	3.42	3.06	2.12	3.07	**2.91**	2.97	3.45	3.58	-	-	3.3
GIBBON	4.92	4.18	3.71	3.52	3.72	4.71	4.25	4.41	-	-	6.8
Hallucination	4.52	4.09	4.42	3.68	4.68	4.75	4.14	5.05	-	-	7.4
LP	5.50	5.48	5.23	4.78	3.84	5.48	5.10	4.53	-	-	8.5
SOBER-TS	3.10	3.43	3.16	3.17	3.30	4.01	3.20	3.21	-	-	4.1
SOBER-LFI	2.58	2.19	2.08	2.65	2.99	**2.96**	**2.28**	**2.31**	**2.43**	**2.35**	**1.5**

Table 3: Average cumulative wall-clock time for 15 iterations (log10 second).

Figure 9 and Table 3 showcase the convergence performance and the wall-clock time for sampling overhead at the 15th iteration, respectively. SOBER-LFI surpasses nine baselines in nine out of ten experiments, demonstrating its versatility and effectiveness across a wide range of multimodal and noisy functions in continuous, discrete, and mixed spaces. Although SOBER-LFI did not achieve the top performance on the unimodal Rosenbrock function—which tends to favour more exploitative algorithms like TurBO—it secured a strong second place. This performance underscores the efficiency of SOBER-LFI's strategy in dynamically narrowing the sampling region around the global maximum. In the realm of drug discovery, SOBER-LFI distinguished itself by showing fast convergence, areas where most algorithms falter due to specific kernel and space requirements. The solvent data set, in particular, highlights scenarios where batch TS quickly converges in early stages but fails

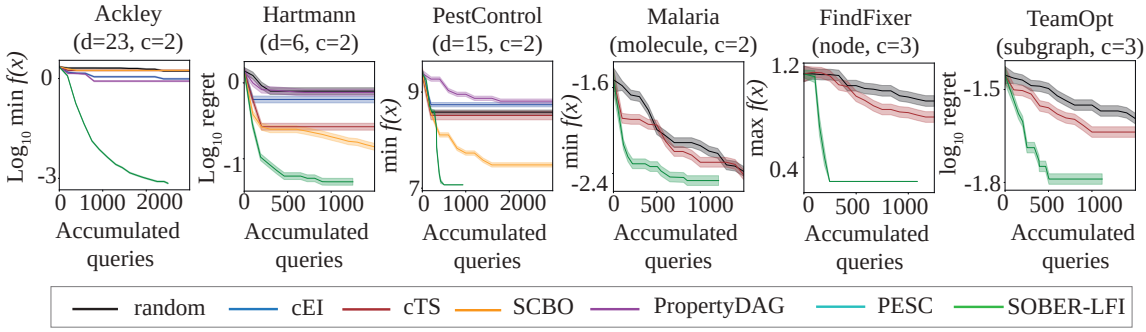


Figure 10: Convergence plot of constrained batch Bayesian optimisation results. d is the dimension, c is the number of unknown constraints.

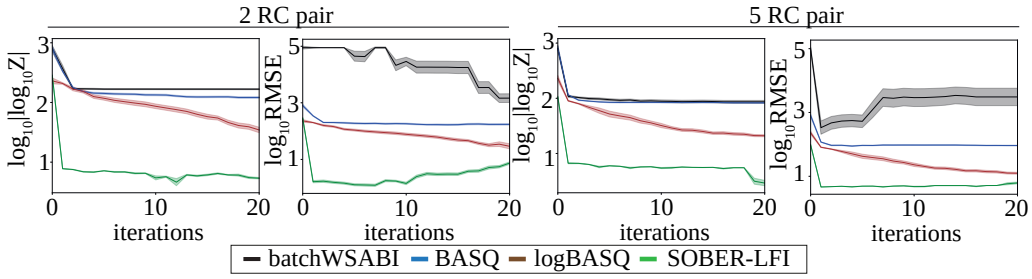


Figure 11: Batch Bayesian Quadrature baseline comparisons across two real-world simulation-based inference tasks. These tasks evaluate both approximation error of evidence (Z) and posterior approximation as root-mean-squared-error (RMSE) against true posterior distribution via exhaustive MCMC sampling and its kernel density estimation. Lower is better for both metrics.

to escape local maxima, eventually equating its final regret with that of random search. Conversely, SOBER-LFI avoids such pitfalls via exploratory KQ sampling.

5.7 Constrained Optimisation Experiments

In the domain of constrained BO tasks with adaptive batch sizes, SOBER-LFI stands out as the sole method offering adaptive batching under constraints. We defined the upper limit of batch sizes for comparison (as detailed in Table 2). While baseline methods maintain fixed upper bound batch sizes across iterations, SOBER-LFI adeptly adjusts its batch sizes, leading to a more efficient use of queries. Consequently, SOBER-LFI typically requires fewer queries to reach the same iteration T. Figure 10 highlights SOBER-LFI's robust empirical performance across various tasks.

5.8 Simulation-based Inference Experiments

SOBER-LFI also demonstrates superior performance against batch BQ baselines in simulation-based inference tasks, as documented by Adachi et al. (2023b). In these tasks, the prior variance significantly exceeds that of the likelihood, resulting in a sharply peaked posterior distribution. This condition renders exploration of the prior's tail as excessive. While the

original BASQ method tends toward over-exploration of the prior distribution, leading to performance plateaus, the logBASQ variant mitigates this issue through log-warp modeling. Nevertheless, SOBER-LFI outpaces all baselines by a significant margin in both posterior and evidence inference across all tasks. It achieves this by effectively concentrating π towards the posterior mode, thereby circumventing unnecessary over-exploration.

6 Discussion

We introduced *SOBER*, a novel quadrature approach to batch BO through probabilistic lifting, showcasing its versatility and theoretical robustness. Initially, we elucidated the theoretical underpinnings of SOBER as a KQ method, emphasizing that its worst-case error bound predominantly stems from the Nyström approximation and empirical measure approximation errors. Crucially, SOBER maintains bounded errors even in scenarios where the RKHS is misspecified, thereby ensuring robustness against GP misspecification. The method's incorporation of closed-form LFI synthetic likelihood, SIR, and recombination techniques facilitates versatile, fast, and diverse sampling strategies. Empirical evidence further verifies that the diminishing variance of π_t correlates strongly with BR, suggesting that the synergistic effect of the exploitative LFI π_t and the explorative KQ sampling constitutes an effective heuristic for addressing the probabilistically lifted dual objective presented in Eq.(5). Through extensive empirical analysis, we demonstrated SOBER-LFI's robustness, scalability, insensitivity to hyperparameters, and adaptivity, further bolstered by an extensive comparative study across a broad spectrum of real-world applications.

Despite its promising empirical performance, there is a pressing need for deeper theoretical analysis to enhance our understanding and identify avenues for improvement. The practical superiority of subsample-based KQ, relative to theoretical predictions, remains a puzzle. Theories on random convex hulls and hypercontractivity provide promising insights, although these explanations originate from slightly different contexts. Additionally, the convergence rate of the KQ method employed is limited to the non-adaptive case (single batch selection). This limitation arises because KQ is based on assumptions of a more general probability measure, such as a non-compact domain. However, Kanagawa and Hennig (2019) established the convergence rate for sequential adaptive BQ within a compact domain. In BO tasks, the assumption of a compact domain is essential (without it, convergence cannot be guaranteed), indicating that this area of research could potentially lead to establishing a full convergence rate. Additionally, exploring the implications of SOBER-LFI in the context of Bayesian cumulative regret convergence rates is pivotal for unraveling the mechanisms behind its empirical success. However, we wish to highlight that our addressed issues—misspecified RKHS for general AFs, and adaptive batch sizes for a limited budget—are not yet accommodated by regret convergence analysis in the existing theoretical BO literature (refer to §4.2.5 and §4.3.3). Our extensive empirical study, along with the theoretical bounds we present for worst-case errors, could lay the groundwork for enhanced theoretical insights. It is also worth noting that SOBER currently does not accommodate asynchronous batch settings (Kandasamy et al., 2018), a limitation identified in previous works (Adachi et al., 2022, 2024a). However, given its generality and flexibility, integrating SOBER with other advanced methods represents a fertile direction for future research, promising valuable contributions to the field for both practitioners and theoreticians.

This paper is the journal extension of the non-archival ICML workshop paper (Adachi et al., 2023a) and AISTATS paper (Adachi et al., 2024a). Although the ICML workshop paper (Adachi et al., 2023a) has not undergone rigorous peer review, it presents content similar to our current work but includes only a limited number of experiments and no theories. The current paper builds upon this ICML workshop paper, providing an in-depth discussion on how to connect batch Bayesian optimization and Bayesian/kernel quadrature through a probabilistic lifting technique. In AISTATS paper (Adachi et al., 2024a), we focused on adaptivity, we posited that the link between quadrature and optimization was established in an ICML workshop paper (Adachi et al., 2023a).

Acknowledgments and Disclosure of Funding

We thank Leo Klarner for the insightful discussion of Bayesian optimistaion for drug discovery, Samuel Daulton, Binxin Ru, and Xingchen Wan for the insightful discussion of Bayesian optimisation for graph and mixed space, Yannick Kuhn for preparing PyPI. Masaki Adachi was supported by the Clarendon Fund, the Oxford Kobe Scholarship, the Watanabe Foundation, and Toyota Motor Corporation. Satoshi Hayakawa was supported by the Clarendon Fund, the Oxford Kobe Scholarship, and the Toyota Riken Overseas Scholarship. Harald Oberhauser was supported by the DataSig Program [EP/S026347/1], the Hong Kong Innovation and Technology Commission (InnoHK Project CIMDA), and the Oxford-Man Institute. Martin Jørgensen was partly supported by the Research Council of Finland (grant 356498) and the Carlsberg foundation. Saad Hamid is grateful for funding from the Engineering and Physical Sciences Research Council of the UK.

Appendix A. Proof of Proposition 1

Proof of Proposition 1. Note that the constraint $|\mathbf{w}|_0 \leq n$ is automatically satisfied when we use the simplex method or its variant. Without this constraint, we have a trivial feasible solution $\mathbf{w} = \mathbf{w}_t^N$, so, for the optimal solution $\mathbf{w}_*$, we have $\mathbf{w}_*^\top \left[\alpha_t(\mathbf{X}_t^n) \odot \tilde{q}_t(\mathbf{X}_t^n)\right] \geq \mathbf{w}_t^{n\top} \left[\alpha_t(\mathbf{X}_t^N) \odot \tilde{q}_t(\mathbf{X}_t^N)\right]$. Since $\mathbb{E}[\tilde{\mathbf{w}}_t^{n\top} \alpha_t(\tilde{\mathbf{X}}_t^n)] = \mathbf{w}_t^{n\top} \left[\alpha_t(\mathbf{X}_t^n) \odot \tilde{q}_t(\mathbf{X}_t^n)\right] = \mathbf{w}_*^\top \left[\alpha_t(\mathbf{X}_t^N) \odot \tilde{q}_t(\mathbf{X}_t^N)\right]$, we obtain the first estimate Eq. (11).

For the latter estimate, we first decompose the error into two parts:

$$\begin{aligned}
&\mathbb{E}\left[\left|\tilde{\mathbf{w}}_t^{n\top} f_{t-1}(\tilde{\mathbf{X}}_t^n) - \mathbf{w}_t^{N\top} f_{t-1}(\mathbf{X}_t^N)\right|\right] \\
&\leq \mathbb{E}\left[\left|\tilde{\mathbf{w}}_t^{n\top} f_{t-1}(\tilde{\mathbf{X}}_t^n) - \mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n)\right|\right] + \left|\mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n) - \mathbf{w}_t^{N\top} f_{t-1}(\mathbf{X}_t^N)\right|. \qquad (13)
\end{aligned}$$

For the first term, considering each $x \in \mathbf{X}_t^n$ on whether or not it gets included in $\tilde{\mathbf{X}}_t^n$, we have

$$\begin{aligned}
&\mathbb{E}\left[\left|\tilde{\mathbf{w}}_t^{n\top} f_{t-1}(\tilde{\mathbf{X}}_t^n) - \mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n)\right|\right] \\
&\leq \mathbf{w}_t^{n\top} \left[|f_{t-1}|(\mathbf{X}_t^n) \odot (1 - \tilde{q}_t(\mathbf{X}_t^n))\right] \leq \mathbf{w}_t^{n\top} (1 - \tilde{q}_t(\mathbf{X}_t^n)) \max_{x \in \mathbf{X}_t^n} |f_{t-1}(x)| \\
&= \left[1 - \mathbf{w}_t^{n\top} \tilde{q}_t(\mathbf{X}_t^n)\right] \max_{x \in \mathbf{X}_t^n} |f_{t-1}(x)| \leq \left[1 - \mathbf{w}_t^{N\top} \tilde{q}_t(\mathbf{X}_t^N)\right] \max_{x \in \mathbf{X}_t^n} |f_{t-1}(x)|,
\end{aligned}$$

where the last inequality follows from the inequality constraint $(\mathbf{w} - \mathbf{w}_t^N)^\top \tilde{q}_t(\mathbf{X}_t^N) \geq 0$ in the LP. Since $|f_{t-1}(x)| = |\langle f_{t-1}, C_{t-1}(\cdot, x)\rangle| \leq \|f_{t-1}\| C_{t-1}(x,x)^{1/2}$ from the reproducing property of RKHS, we obtain

$$\mathbb{E}\left[\left|\tilde{\mathbf{w}}_t^{n\top} f_{t-1}(\tilde{\mathbf{X}}_t^n) - \mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n)\right|\right] \leq \epsilon_{\mathrm{vio}} K_{\max} \|f_{t-1}\|. \tag{14}$$

Let us then bound the second term of the RHS of Eq. (13). Note that, from the formula of worst-case error of kernel quadrature (see, e.g., (Hayakawa et al., 2022, Eq. (14))), we can bound

$$\left|\mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n) - \mathbf{w}_t^{N\top} f_{t-1}(\mathbf{X}_t^N)\right|^2 \leq \|f_{t-1}\|^2 (\mathbf{w}_* - \mathbf{w}_t^N)^\top C_{t-1}(\mathbf{X}_t^N, \mathbf{X}_t^N)(\mathbf{w}_* - \mathbf{w}_t^N) \tag{15}$$

(recall $\mathbf{w}_*$ has the same dimension as $\mathbf{w}_t^N$). We now want to estimate

$$(\mathbf{w}_* - \mathbf{w}_t^N)^\top C_{t-1}(\mathbf{X}_t^N, \mathbf{X}_t^N)(\mathbf{w}_* - \mathbf{w}_t^N).$$

Consider approximating C_{t-1} by $\tilde{C}_{t-1}$. Since $C_{t-1} - \tilde{C}_{t-1}$ is positive semi-definite from the property of Nyström approximation (see, e.g., the proof of (Hayakawa et al., 2022, Corollary 4)), for any $x, y \in \mathbf{X}_t^N$, we have

$$|(C_{t-1} - \tilde{C}_{t-1})(x,y)| \leq |(C_{t-1} - \tilde{C}_{t-1})(x,x)|^{1/2} |(C_{t-1} - \tilde{C}_{t-1})(y,y)|^{1/2} \leq \epsilon_{\mathrm{nys}}^2.$$

Thus, we have

$$\begin{aligned}
&(\mathbf{w}_* - \mathbf{w}_t^N)^\top \left[(C_{t-1} - \tilde{C}_{t-1})(\mathbf{X}_t^N, \mathbf{X}_t^N)\right](\mathbf{w}_* - \mathbf{w}_t^N) \\
&\leq (\mathbf{w}_* + \mathbf{w}_t^N)^\top (\epsilon_{\mathrm{nys}}^2 \mathbf{1}\mathbf{1}^\top)(\mathbf{w}_* + \mathbf{w}_t^N) = 4\epsilon_{\mathrm{nys}}^2.
\end{aligned} \tag{16}$$

Finally, we estimate

$$\begin{aligned}
&(\mathbf{w}_* - \mathbf{w}_t^N)^\top \tilde{C}_{t-1}(\mathbf{X}_t^N, \mathbf{X}_t^N)(\mathbf{w}_* - \mathbf{w}_t^N) \\
&= (\mathbf{w}_* - \mathbf{w}_t^N)^\top \sum_{j=1}^{n-2} \mathbf{1}_{\{\lambda_j > 0\}} \lambda_j^{-1} \varphi_j(\mathbf{X}_t^N) \varphi_j(\mathbf{X}_t^N)^\top (\mathbf{w}_* - \mathbf{w}_t^N) \\
&= \sum_{j=1}^{n-2} \mathbf{1}_{\{\lambda_j > 0\}} \lambda_j^{-1} \left[(\mathbf{w}_* - \mathbf{w}_t^N)^\top \varphi_j(\mathbf{X}_t^N)\right]^2.
\end{aligned} \tag{17}$$

From the inequality constraint in the LP, we have $|(\mathbf{w}_* - \mathbf{w}_t^N)^\top \varphi_j(\mathbf{X}_t^N)| \leq \epsilon_{\mathrm{LP}} \sqrt{\lambda_j/(n-2)}$, so that Eq. (17) is further bounded as

$$(\mathbf{w}_* - \mathbf{w}_t^N)^\top \tilde{C}_{t-1}(\mathbf{X}_t^N, \mathbf{X}_t^N)(\mathbf{w}_* - \mathbf{w}_t^N) \leq \sum_{j=1}^{n-2} \mathbf{1}_{\{\lambda_j > 0\}} \lambda_j^{-1} \epsilon_{\mathrm{LP}}^2 \frac{\lambda_j}{n-2} \leq \epsilon_{\mathrm{LP}}^2. \tag{18}$$

By adding the both sides of Eqs. (16) and (18), we obtain

$$(\mathbf{w}_* - \mathbf{w}_t^N)^\top C_{t-1}(\mathbf{X}_t^N, \mathbf{X}_t^N)(\mathbf{w}_* - \mathbf{w}_t^N) \leq 4\epsilon_{\mathrm{nys}}^2 + \epsilon_{\mathrm{LP}}^2 \leq (2\epsilon_{\mathrm{nys}} + \epsilon_{\mathrm{LP}})^2.$$

By applying this to Eq. (15), we have $\left|\mathbf{w}_t^{n\top} f_{t-1}(\mathbf{X}_t^n) - \mathbf{w}_t^{N\top} f_{t-1}(\mathbf{X}_t^N)\right| \leq \|f\|(2\epsilon_{\mathrm{nys}} + \epsilon_{\mathrm{LP}})$. Combining this with Eqs. (13) and (14) yields the desired inequality Eq. (12). □

This proof is mirrored from our previous work (Adachi et al., 2024a).

Appendix B. Experimental details

Due to the page restriction to be within 35 pages on submission, we defer the details to our prior papers. While our previous papers and GitHub code work as the explanation to reproduce our results, we will update the experimental details to here in appendix together to be self-contained. The experimental details of batch constrained BO is delineated in our previous work (Adachi et al., 2024a). The batch BQ experiments are detailed in our previous work (Adachi et al., 2023b). For batch unconstrained BO, we used the constrained BO tasks without constraints. We explain the details of the rest; Rosenbrock, Shekel, MaxSat, Ising, SVM, and Solvent:

Synthetic: Rosenbrock function We modified the original Rosenbrock function (Surjanovic and Bingham, 2024) into a 7-dimensional function with the mixed spaces of 1 continuous and 6 discrete variables, following Daulton et al. (2022). The first 1 dimension is continuous with bounds $[-4, 11]^1$. The other 6 dimensions are discretised to be categorical variables, with 4 possible values $x_1 \in \{-4, 1, 6, 11\}$.

Synthetic: Shekel function We use Shekel function without any modification from (Surjanovic and Bingham, 2024), 4 dimensional continuous variables bounded $[0, 10]^4$.

Real-world: Maximum Satisifiability Maximum satisfiability (MaxSat in the main) is proposed in Oh et al. (2019), which is 28 dimensional binary optimisation problem. The objective is to find boolean values that maximise the combined weighted satisfied clauses for the data set provided by Maximum Satisfiability competition 2018. Both code and data set are used in `https://github.com/xingchenwan/Casmopolitan` (Wan et al., 2021).

Real-world: Ising Model Sparsification Ising Model Sparsification (Ising in the main) is proposed in Oh et al. (2019), which is 24 dimensional binary optimisation problem. The objective is to sparsify an Ising model using the regularised Kullback-Leibler divergence between a zero-field Ising model and the partition function, considering 4×4 grid of spins with regularisation coefficient $\lambda = 10^{-4}$. Code is in `https://github.com/QUVA-Lab/COMBO`.

Real-world: Support Vector Machine Feature Selection Support vector machine feature selection (SVM in the main) is proposed in Daulton et al. (2022), which is 23 dimensional mixed-type input optimisation problem (20 dimensional binary and 3 dimensional continuous variables). The objective is jointly performing feature selection (20 features) and hyperparameter optimisation (3 hyperparameters) for a support vector machine trained in the CTSlice UCI data set (Graf et al., 2011; Dua and Graff, 2017). Code is used in `https://github.com/facebookresearch/bo_pr`.

Real-world: Polar solvent for batteries The data set with 133,055 small molecules represented as 2048-dimensional binary features were optimised and predicted by the quantum-chemical computations, known as QM9 data set (Ramakrishnan et al., 2014). The target variable is the dipole moment, which is basically correlated with the solvation capability in electrolytes in lithium-ion batteries, increasing the ratio of electro-mobile lithium-ions. The higher the dipole moment becomes, the larger (better) the ionic conductivity does. The data set is downloaded from `http://quantum-machine.org/datasets/`. The coding was done with Gauche (Griffiths et al., 2023). Due to the low expressive capability of

2048-dimensional binary features, we removed the duplicated candidates that show identical binary features from the QM9 data set, then applied the batch BO experiments.

References

Masaki Adachi. High-dimensional discrete Bayesian optimization with self-supervised representation learning for data-efficient materials exploration. In *NeurIPS 2021 AI for Science Workshop*, 2021. URL https://openreview.net/forum?id=xJhjehqjQeB.

Masaki Adachi, Satoshi Hayakawa, Martin Jørgensen, Harald Oberhauser, and Michael A Osborne. Fast Bayesian inference with batch Bayesian quadrature via kernel recombination. *Advances in Neural Information Processing Systems (NeurIPS)*, 35, 2022. URL https://doi.org/10.48550/arXiv.2206.04734.

Masaki Adachi, Satoshi Hayakawa, Saad Hamid, Martin Jørgensen, Harald Oberhauser, and Michael A Osborne. SOBER: Highly parallel Bayesian optimization and Bayesian quadrature over discrete and mixed spaces. In *ICML 2023 Workshop: Sampling and Optimization in Discrete Space*, 2023a. doi: https://doi.org/10.48550/arXiv.2301.11832.

Masaki Adachi, Yannick Kuhn, Birger Horstmann, Arnulf Latz, Michael A Osborne, and David A Howey. Bayesian model selection of lithium-ion battery models via Bayesian quadrature. *IFAC-PapersOnLine*, 56(2):10521–10526, 2023b. URL https://doi.org/10.1016/j.ifacol.2023.10.1073.

Masaki Adachi, Satoshi Hayakawa, Martin Jørgensen, Xingchen Wan, Vu Nguyen, Harald Oberhauser, and Michael A. Osborne. Adaptive batch sizes for active learning a probabilistic numerics approach. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024a. URL https://doi.org/10.48550/arXiv.2306.05843.

Masaki Adachi, Brady Planden, David A Howey, Michael A Osborne, Sebastian Orbell, Natalia Ares, Krikamol Muandet, and Siu Lun Chau. Looping in the human: Collaborative and explainable Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024b. URL https://doi.org/10.48550/arXiv.2310.17273.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *International Conference on Knowledge Discovery & Data Mining (KDD)*, page 2623–2631, 2019. URL https://doi.org/10.1145/3292500.3330701.

Imad Aouali, Branislav Kveton, and Sumeet Katariya. Mixed-effect Thompson sampling. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2087–2115. PMLR, 2023. URL https://proceedings.mlr.press/v206/aouali23a/aouali23a.pdf.

Javad Azimi, Alan Fern, and Xiaoli Fern. Batch Bayesian optimization via simulation matching. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 23, 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/e702e51da2c0f5be4dd354bb3e295d37-Paper.pdf.

Francis Bach. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research*, 18:714, 2017. URL https://jmlr.org/papers/volume18/15-178/15-178.pdf.

Francis Bach, Simon Lacoste-Julien, and Guillaume Obozinski. On the equivalence between herding and conditional gradient algorithms. In *International Conference on Machine Learning (ICML)*, page 1355–1362, 2012. URL https://icml.cc/2012/papers/683.pdf.

Maximilian Balandat, Brian Karrer, Daniel Jiang, Samuel Daulton, Ben Letham, Andrew G Wilson, and Eytan Bakshy. BoTorch: a framework for efficient Monte-Carlo Bayesian optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 21524–21538, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f5b1b89d98b7286673128a5fb112cb9a-Paper.pdf.

Ayoub Belhadji. An analysis of Ermakov-Zolotukhin quadrature using kernels. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 27278–27289, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/e531e258fe3098c3bdd707c30a687d73-Paper.pdf.

Ayoub Belhadji, Rémi Bardenet, and Pierre Chainais. Kernel quadrature with DPPs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/7012ef0335aa2adbab58bd6d0702ba41-Paper.pdf.

Ayoub Belhadji, Rémi Bardenet, and Pierre Chainais. Kernel interpolation with continuous volume sampling. In Hal Daumé III and Aarti Singh, editors, *International Conference on Machine Learning (ICML)*, volume 119, pages 725–735, 2020. URL http://proceedings.mlr.press/v119/belhadji20a/belhadji20a.pdf.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 24, 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf.

Felix Berkenkamp, Angela P Schoellig, and Andreas Krause. No-regret Bayesian optimization with unknown hyperparameters. *Journal of Machine Learning Research*, 20(50): 1–24, 2019. URL http://jmlr.org/papers/v20/18-213.html.

Ilija Bogunovic and Andreas Krause. Misspecified Gaussian process bandit optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 3004–3015, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/177db6acfe388526a4c7bff88e1feb15-Paper.pdf.

François-Xavier Briol, Chris J Oates, Mark Girolami, Michael A Osborne, and Dino Sejdinovic. Probabilistic integration: A role in statistical computation? *Statistical Science*, 34(1):1–22, 2019. URL https://www.jstor.org/stable/26771026.

Adam D Bull. Convergence rates of efficient global optimization algorithms. *Journal of Machine Learning Research*, 12(88):2879–2904, 2011. URL http://jmlr.org/papers/v12/bull11a.html.

Arnaud Carpentier, Ila Nimgaonkar, Virginia Chu, Yuchen Xia, Zongyi Hu, and T Jake Liang. Hepatic differentiation of human pluripotent stem cells in miniaturized format suitable for high-throughput screen. *Stem Cell Research*, 16(3):640–650, 2016. URL https://doi.org/10.1016/j.scr.2016.03.009.

Yutian Chen, Max Welling, and Alex Smola. Super-samples from kernel herding. In *International Conference on Uncertainty in Artificial Intelligence (UAI)*, page 109–116, 2010. URL https://doi.org/10.48550/arXiv.1203.3472.

Francesco Cosentino, Harald Oberhauser, and Alessandro Abate. A randomized algorithm to reduce the support of discrete measures. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 15100–15110, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/ac4395adcb3da3b2af3d3972d7a10221-Paper.pdf.

Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

Katalin Csilléry, Michael GB Blum, Oscar E Gaggiotti, and Olivier François. Approximate Bayesian computation (ABC) in practice. *Trends in Ecology & Evolution*, 25(7):410–418, 2010. URL https://doi.org/10.1016/j.tree.2010.04.001.

Zhongxiang Dai, Quoc Phong Nguyen, Sebastian Tay, Daisuke Urano, Richalynn Leong, Bryan Kian Hsiang Low, and Patrick Jaillet. Batch Bayesian optimization for replicable experimental design. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 36476–36506, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/727a5a5c77be15d053b47b7c391800c2-Paper-Conference.pdf.

George B Dantzig. Linear programming. *Operations Research*, 50(1):42–47, 2002. URL https://doi.org/10.1287/opre.50.1.42.17798.

Samuel Daulton, Xingchen Wan, David Eriksson, Maximilian Balandat, Michael A Osborne, and Eytan Bakshy. Bayesian optimization over discrete and mixed spaces via probabilistic reparameterization. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:12760–12774, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/531230cfac80c65017ad0f85d3031edc-Paper-Conference.pdf.

Petros Drineas and Michael W. Mahoney. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*, 6(72):2153–2175, 2005. URL http://jmlr.org/papers/v6/drineas05a.html.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Raaz Dwivedi and Lester Mackey. Kernel thinning. In *Annual Conference on Learning Theory (COLT)*, volume 134, pages 1753–1753, 2021. URL http://proceedings.mlr.press/v134/dwivedi21a/dwivedi21a.pdf.

Raaz Dwivedi and Lester Mackey. Generalized kernel thinning. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=IfNu7Dr-3fQ.

David Eriksson and Matthias Poloczek. Scalable constrained Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 130, pages 730–738. PMLR, 2021. URL http://proceedings.mlr.press/v130/eriksson21a/eriksson21a.pdf.

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 (NeurIPS)*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/6c990b7aca7bc7058f5e98ea909e924b-Paper.pdf.

Masahiro Fujisawa, Takeshi Teshima, Issei Sato, and Masashi Sugiyama. γ-ABC: Outlier-robust approximate Bayesian computation based on a robust divergence estimator. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 130, pages 1783–1791. PMLR, 2021. URL http://proceedings.mlr.press/v130/fujisawa21a/fujisawa21a.pdf.

Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 7576–7586, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/27e8e17134dd7083b050476733207ea1-Paper.pdf.

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

Michael A Gelbart, Jasper Snoek, and Ryan P Adams. Bayesian optimization with unknown constraints. In *International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 250–259, 2014. doi: https://doi.org/10.48550/arXiv.1403.5607.

Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018. URL https://doi.org/10.1021/acscentsci.7b00572.

Javier González, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch Bayesian optimization via local penalization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 51, pages 648–657. PMLR, 2016.

Christian Gourieroux, Alain Monfort, and Eric Renault. Indirect inference. *Journal of Applied Econometrics*, 8(S1):S85–S118, 1993. URL https://doi.org/10.1002/jae.3950080507.

Franz Graf, Hans-Peter Kriegel, Matthias Schubert, Sebastian Pölsterl, and Alexander Cavallaro. 2D image registration in CT images using radial image descriptors. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 607–614. Springer, 2011. URL https://doi.org/10.1007/978-3-642-23629-7_74.

Ryan-Rhys Griffiths, Leo Klarner, Henry Moss, Aditya Ravuri, Sang Truong, Yuanqi Du, Samuel Stanton, Gary Tom, Bojana Rankovic, Arian Jamasb, et al. GAUCHE: A library for Gaussian processes in chemistry. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 76923–76946, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/f2b1b2e974fa5ea622dd87f22815f423-Paper-Conference.pdf.

Michael U. Gutmann and Jukka Corander. Bayesian optimization for likelihood-free inference of simulator-based statistical models. *Journal of Machine Learning Research*, 17 (125):1–47, 2016. URL http://jmlr.org/papers/v17/15-017.html.

Michael U Gutmann and Jun-ichiro Hirayama. Bregman divergence as general framework to estimate unnormalized statistical models. In *International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 283–290, 2011. URL https://doi.org/10.48550/arXiv.1202.3727.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011. URL https://doi.org/10.1137/090771806.

Nikolaus Hansen. The CMA evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016. URL https://doi.org/10.48550/arXiv.1604.00772.

Satoshi Hayakawa, Harald Oberhauser, and Terry Lyons. Positively weighted kernel quadrature via subsampling. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 6886–6900, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/2dae7d1ccf1edf76f8ce7c282bdf4730-Paper-Conference.pdf.

Satoshi Hayakawa, Terry Lyons, and Harald Oberhauser. Estimating the probability that a given vector is in the convex hull of a random sample. *Probability Theory and Related Fields*, 185:705–746, 2023a. URL https://doi.org/10.1007/s00440-022-01186-1.

Satoshi Hayakawa, Harald Oberhauser, and Terry Lyons. Hypercontractivity meets random convex hulls: analysis of randomized multivariate cubatures. *Proceedings of the Royal Society A*, 479(2273):20220725, 2023b. doi: 10.1098/rspa.2022.0725. URL https://royalsocietypublishing.org/doi/abs/10.1098/rspa.2022.0725.

Satoshi Hayakawa, Harald Oberhauser, and Terry Lyons. Sampling-based Nyström approximation and kernel quadrature. In *International Conference on Machine Learning (ICML)*, volume 202, pages 12678–12699, 2023c. URL https://proceedings.mlr.press/v202/hayakawa23a/hayakawa23a.pdf.

Philipp Hennig, Michael A Osborne, and Mark Girolami. Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142, 2015. URL https://doi.org/10.1098/rspa.2015.0142.

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

José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 27, 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/069d3bb002acd8d7dd095917f9efe4cb-Paper.pdf.

José Miguel Hernández-Lobato, Michael Gelbart, Matthew Hoffman, Ryan Adams, and Zoubin Ghahramani. Predictive entropy search for Bayesian optimization with unknown constraints. In *International Conference on Machine Learning (ICML)*, volume 37, pages 1699–1707. PMLR, 2015. URL http://proceedings.mlr.press/v37/hernandez-lobatob15.pdf.

José Miguel Hernández-Lobato, Michael A. Gelbart, Ryan P. Adams, Matthew W. Hoffman, and Zoubin Ghahramani. A general framework for constrained Bayesian optimization using information-based search. *Journal of Machine Learning Research*, 17(1):5549–5601, 2016. URL http://jmlr.org/papers/v17/15-616.html.

José Miguel Hernández-Lobato, James Requeima, Edward O Pyzer-Knapp, and Alán Aspuru-Guzik. Parallel and distributed Thompson sampling for large-scale accelerated exploration of chemical space. In *International Conference on Machine Learning (ICML)*, pages 1470–1479. PMLR, 2017. URL http://proceedings.mlr.press/v70/hernandez-lobato17a/hernandez-lobato17a.pdf.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. URL https://doi.org/10.1162/089976602760128018.

Daolang Huang, Ayush Bharti, Amauri Souza, Luigi Acerbi, and Samuel Kaski. Learning robust statistics for simulation-based inference under model misspecification. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 7289–7310, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/16c5b4102a6b6eb061e502ce6736ad8a-Paper-Conference.pdf.

Ferenc Huszár and David Duvenaud. Optimally-weighted herding is Bayesian quadrature. In *International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 377—-386, 2012. URL https://doi.org/10.48550/arXiv.1204.1664.

Carl Hvarfner, Danny Stoll, Artur Souza, Marius Lindauer, Frank Hutter, and Luigi Nardi. πBO: Augmenting acquisition functions with user beliefs for Bayesian optimization. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://doi.org/10.48550/arXiv.2204.11051.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL http://jmlr.org/papers/v6/hyvarinen05a.html.

Simon Jackman. *Bayesian analysis for the social sciences*. John Wiley & Sons, 2009.

Motonobu Kanagawa and Philipp Hennig. Convergence guarantees for adaptive Bayesian quadrature methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/165a59f7cf3b5c4396ba65953d679f17-Paper.pdf.

Motonobu Kanagawa, Bharath K Sriperumbudur, and Kenji Fukumizu. Convergence guarantees for kernel-based quadrature rules in misspecified settings. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/81c650caac28cdefce4de5ddc18befa0-Paper.pdf.

Motonobu Kanagawa, Bharath K Sriperumbudur, and Kenji Fukumizu. Convergence analysis of deterministic kernel-based quadrature rules in misspecified settings. *Foundations of Computational Mathematics*, 20:155–194, 2020. URL https://doi.org/10.1007/s10208-018-09407-7.

Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. High dimensional Bayesian optimisation and bandits via additive models. In *International Conference on Machine Learning (ICML)*, pages 295–304. PMLR, 2015. URL http://proceedings.mlr.press/v37/kandasamy15.pdf.

Kirthevasan Kandasamy, Akshay Krishnamurthy, Jeff Schneider, and Barnabás Póczos. Parallelised Bayesian optimisation via Thompson sampling. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 84, pages 133–142. PMLR, 2018. URL http://proceedings.mlr.press/v84/kandasamy18a/kandasamy18a.pdf.

Toni Karvonen. *Kernel-based and Bayesian Methods for Numerical Integration*. PhD thesis, Aalto University, 2019.

Toni Karvonen, Chris J Oates, and Simo Sarkka. A Bayes-Sard cubature method. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/6775a0635c302542da2c32aa19d86be0-Paper.pdf.

Tarun Kathuria, Amit Deshpande, and Pushmeet Kohli. Batched Gaussian process bandit optimization via determinantal point processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/a1d7311f2a312426d710e1c617fcbc8c-Paper.pdf.

Wonyoung Kim, Gi-Soo Kim, and Myunghee Cho Paik. Doubly robust Thompson sampling with linear payoffs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 15830–15840, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/84d5711e9bf5547001b765878e7b0157-Paper.pdf.

Genshiro Kitagawa. A Monte Carlo filtering and smoothing method for non-Gaussian nonlinear state space models. In *Proceedings of the 2nd US-Japan Joint Seminar on Statistical Time Series Analysis*, volume 110, 1993. URL https://www.ism.ac.jp/~kitagawa/1993_US-Japan.pdf.

Andreas Krause and Carlos E Guestrin. Near-optimal nonmyopic value of information in graphical models. In *International Conference on Uncertainty in Artificial Intelligence (AISTATS)*, pages 324–331, 2012. URL https://doi.org/10.48550/arXiv.1207.1394.

Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. Sampling methods for the Nyström method. *Journal of Machine Learning Research*, 13(34):981–1006, 2012. URL http://jmlr.org/papers/v13/kumar12a.html.

Harold J Kushner. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering*, 86:97 – 106, 1964. URL https://doi.org/10.1115/1.3653121.

Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. Cost-effective outbreak detection in networks. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 420–429, 2007. URL https://doi.org/10.1145/1281192.1281239.

Benjamin Letham, Brian Karrer, Guilherme Ottoni, and Eytan Bakshy. Constrained Bayesian optimization with noisy experiments. *Bayesian Analysis*, 14(2):495 – 519, 2019. URL https://doi.org/10.1214/18-BA1110.

Shibo Li, Jeff M Phillips, Xin Yu, Robert Kirby, and Shandian Zhe. Batch multi-fidelity active learning with budget constraints. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 995–1007, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/06ea400b9b7cfce6428ec27a371632eb-Paper-Conference.pdf.

Jihao Andreas Lin, Javier Antorán, Shreyas Padhy, David Janz, José Miguel Hernández-Lobato, and Alexander Terenin. Sampling from Gaussian process posteriors using stochastic gradient descent. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 36886–36912, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7482e8ce4139df1a2d8195a0746fa713-Paper-Conference.pdf.

Christian Litterer and Terry Lyons. High order recombination and an application to cubature on Wiener space. *The Annals of Applied Probability*, 22(4):1301 – 1327, 2012. URL https://doi.org/10.1214/11-AAP786.

Jonas Mockus. On the Bayes methods for seeking the extremal point. *IFAC Proceedings Volumes*, 8(1, Part 1):428–431, 1975. doi: https://doi.org/10.1016/S1474-6670(17)67769-3.

Henry B. Moss, David S. Leslie, Javier Gonzalez, and Paul Rayson. GIBBON: General-purpose information-based Bayesian optimisation. *Journal of Machine Learning Research*, 22(235):1–49, 2021. URL http://jmlr.org/papers/v22/21-0120.html.

Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Bernhard Schölkopf, et al. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141, 2017.

Elvis Nava, Mojmir Mutny, and Andreas Krause. Diversified sampling for batched Bayesian optimization with determinantal point processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 151, pages 7031–7054. PMLR, 2022. URL https://proceedings.mlr.press/v151/nava22a/nava22a.pdf.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14: 265–294, 1978. URL https://doi.org/10.1007/BF01588971.

Vu Nguyen, Santu Rana, Sunil K Gupta, Cheng Li, and Svetha Venkatesh. Budgeted batch Bayesian optimization. In *International Conference on Data Mining (ICDM)*, pages 1107–1112. IEEE, 2016. URL https://doi.org/10.1109/ICDM.2016.0144.

Chris J Oates, Mark Girolami, and Nicolas Chopin. Control functionals for Monte Carlo integration. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 79 (3):695–718, 2017. URL https://doi.org/10.1111/rssb.12185.

Changyong Oh, Jakub Tomczak, Efstratios Gavves, and Max Welling. Combinatorial Bayesian optimization using the graph Cartesian product. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/2cb6b10338a7fc4117a80da24b582060-Paper.pdf.

Anthony O'Hagan. Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991. URL https://doi.org/10.1016/0378-3758(91)90002-V.

Michael Osborne, Roman Garnett, Zoubin Ghahramani, David K Duvenaud, Stephen J Roberts, and Carl Rasmussen. Active learning of model evidence using Bayesian quadrature. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 25, 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/6364d3f0f495b6ab9dcf8d3b5c6e0b01-Paper.pdf.

Michael A Osborne, Roman Garnett, and Stephen J Roberts. Gaussian processes for global optimization. In *International Conference on Learning and Intelligent Optimization (LION3)*, 2009. URL https://ora.ox.ac.uk/objects/uuid:7d2b38d0-43be-4bb4-852c-50001a28ead9/files/sq237ht37z.

Lorenzo Pacchiardi, Sherman Khoo, and Ritabrata Dutta. Generalized Bayesian likelihood-free inference using scoring rules estimators. *arXiv preprint arXiv:2104.03889*, 2021. URL https://doi.org/10.48550/arXiv.2104.03889.

Ji Won Park, Samuel Stanton, Saeed Saremi, Andrew Watkins, Henri Dwyer, Vladimir Gligorijevic, Richard Bonneau, Stephen Ra, and Kyunghyun Cho. PropertyDAG: Multi-objective Bayesian optimization of partially ordered, mixed-variable properties for biological sequence design. In *NeurIPS 2022 AI for Science: Progress and Promises*, 2022. URL https://doi.org/10.48550/arXiv.2210.04096.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, and Luca Antiga. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Leah F Price, Christopher C Drovandi, Anthony Lee, and David J Nott. Bayesian synthetic likelihood. *Journal of Computational and Graphical Statistics*, 27(1):1–11, 2018. URL https://doi.org/10.1080/10618600.2017.1302882.

Liva Ralaivola, Sanjay J Swamidass, Hiroto Saigo, and Pierre Baldi. Graph kernels for chemical informatics. *Neural networks*, 18(8):1093–1110, 2005. URL https://doi.org/10.1016/j.neunet.2005.07.009.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1): 1–7, 2014. URL https://doi.org/10.1038/sdata.2014.22.

Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 15, pages 505–512, 2002. URL https://proceedings.neurips.cc/paper_files/paper/2002/file/24917db15c4e37e421866448c9ab23d8-Paper.pdf.

Carl Edward Rasmussen, Christopher KI Williams, et al. *Gaussian processes for machine learning*, volume 1. Springer, 2006.

Zhaolin Ren and Na Li. Minimizing the Thompson sampling regret-to-sigma ratio (TS-RSR): a provably efficient algorithm for batch Bayesian optimization. *arXiv preprint arXiv:2403.04764*, 2024.

Alireza Rezaei and Shayan Oveis Gharan. A polynomial time MCMC method for sampling from continuous determinantal point processes. In *International Conference on Machine Learning (ICML)*, volume 97, pages 5438–5447, 2019. URL http://proceedings.mlr.press/v97/rezaei19a/rezaei19a.pdf.

Binxin Ru, Ahsan Alvi, Vu Nguyen, Michael A Osborne, and Stephen Roberts. Bayesian optimisation over multiple continuous and categorical inputs. In *International Conference on Machine Learning (ICML)*, volume 119, pages 8276–8285. PMLR, 2020. URL http://proceedings.mlr.press/v119/ru20a/ru20a.pdf.

Alessandro Rudi, Ulysse Marteau-Ferey, and Francis Bach. Finding global minima via kernel approximations. *arXiv preprint arXiv:2012.11978*, 2020. URL https://doi.org/10.48550/arXiv.2012.11978.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. URL https://minds.wisconsin.edu/handle/1793/60660.

Max Simchowitz, Christopher Tosh, Akshay Krishnamurthy, Daniel J Hsu, Thodoris Lykouris, Miro Dudik, and Robert E Schapire. Bayesian decision-making under misspecified priors with applications to meta-learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 26382–26394, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/ddcbe25988981920c872c1787382f04d-Paper.pdf.

Jiaming Song, Lantao Yu, Willie Neiswanger, and Stefano Ermon. A general recipe for likelihood-free Bayesian optimization. In *International Conference on Machine Learning (ICML)*, volume 162, pages 20384–20404. PMLR, 2022. URL https://proceedings.mlr.press/v162/song22b/song22b.pdf.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *International Conference on Machine Learning (ICML)*, pages 1015–1022, 2010. URL https://icml.cc/Conferences/2010/papers/422.pdf.

Bharath K Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert RG Lanckriet. Hilbert space embeddings and metrics on probability measures. *Journal of Machine Learning Research*, 11:1517–1561, 2010. URL https://www.jmlr.org/papers/volume11/sriperumbudur10a/sriperumbudur10a.pdf.

Michael L Stein. *Interpolation of spatial data*. Springer Science & Business Media, 1999.

Stergios Stergiopoulos. *Advanced signal processing handbook: theory and implementation for radar, sonar, and medical imaging real time systems*. CRC press, 2017.

Yanan Sui, Alkis Gotovos, Joel Burdick, and Andreas Krause. Safe exploration for optimization with Gaussian processes. In *International Conference on Machine Learning (ICML)*, volume 37, pages 997–1005. PMLR, 2015. URL http://proceedings.mlr.press/v37/sui15.pdf.

Sonja Surjanovic and Derek Bingham. Virtual library of simulation experiments: Test functions and datasets. Retrieved March 29, 2024, from http://www.sfu.ca/~ssurjano, 2024.

Shion Takeno, Yu Inatsu, Masayuki Karasuyama, and Ichiro Takeuchi. Posterior sampling-based Bayesian optimization with tighter Bayesian regret bounds. *arXiv preprint arXiv:2311.03760*, 2023. URL https://doi.org/10.48550/arXiv.2311.03760.

Maria Tchernychova. *Carathéodory cubature measures*. PhD thesis, University of Oxford, 2015.

Onur Teymur, Jackson Gorham, Marina Riabiz, and Chris Oates. Optimal quantisation of probability measures using maximum mean discrepancy. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 130, pages 1027–1035. PMLR, 2021. URL http://proceedings.mlr.press/v130/teymur21a/teymur21a.pdf.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933. URL https://doi.org/10.1093/biomet/25.3-4.285.

Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. *SIAM review*, 38(1): 49–95, 1996. URL https://doi.org/10.1137/1038003.

Ed Wagstaff, Saad Hamid, and Michael Osborne. Batch selection for parallelisation of Bayesian quadrature. *arXiv preprint arXiv:1812.01553*, 2018. URL https://doi.org/10.48550/arXiv.1812.01553.

Xingchen Wan, Vu Nguyen, Huong Ha, Binxin Ru, Cong Lu, and Michael A. Osborne. Think global and act local: Bayesian optimisation over high-dimensional categorical and mixed search spaces. In *International Conference on Machine Learning (ICML)*, volume 139, pages 10663–10674, 2021. URL http://proceedings.mlr.press/v139/wan21b/wan21b.pdf.

Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient Bayesian optimization. In *International Conference on Machine Learning (ICML)*, volume 70, pages 3627–3635. PMLR, 2017. URL http://proceedings.mlr.press/v70/wang17e/wang17e.pdf.

Zi Wang, Beomjoon Kim, and Leslie P Kaelbling. Regret bounds for meta Bayesian optimization with an unknown Gaussian process prior. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/41f860e3b7f548abc1f8b812059137bf-Paper.pdf.

Veit David Wild, Sahra Ghalebikesabi, Dino Sejdinovic, and Jeremias Knoblauch. A rigorous link between deep ensembles and (variational) Bayesian methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 39782–39811, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7d25b1db211d99d5750ec45d65fd6e4e-Paper-Conference.pdf.

Christopher Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 13, 2000. URL https://proceedings.neurips.cc/paper/2000/file/19de10adbaa1b2ee13f77f679fa1483a-Paper.pdf.

James Wilson. Stopping bayesian optimization with probabilistic regret bounds. *arXiv preprint arXiv:2402.16811*, 2024.

James Wilson, Frank Hutter, and Marc Deisenroth. Maximizing acquisition functions for bayesian optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/498f2c21688f6451d9f5fd09d53edda7-Paper.pdf.

James Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Deisenroth. Efficiently sampling functions from Gaussian process posteriors. In *International Conference on Machine Learning (ICML)*, volume 119, pages 10292–10302, 2020. URL http://proceedings.mlr.press/v119/wilson20a/wilson20a.pdf.

Simon N Wood. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466(7310):1102–1104, 2010. URL https://doi.org/10.1038/nature09319.

Stephen J Wright. *Primal-dual interior-point methods*. SIAM, 1997.

Wenjie Xu, Yuning Jiang, Bratislav Svetozarevic, and Colin Jones. Constrained efficient global optimization of expensive black-box functions. In *International Conference on Machine Learning (ICML)*, volume 202, pages 38485–38498. PMLR, 2023. URL https://proceedings.mlr.press/v202/xu23h/xu23h.pdf.

Guorong Xuan, Wei Zhang, and Peiqi Chai. EM algorithms of Gaussian mixture model and hidden Markov model. In *International Conference on Image Processing*, volume 1, pages 145–148. IEEE, 2001. URL https://doi.org/10.1109/ICIP.2001.958974.

Tianbao Yang, Yu-Feng Li, Mehrdad Mahdavi, Rong Jin, and Zhi-Hua Zhou. Nyström method vs random Fourier features: A theoretical and empirical comparison. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 25, pages 476–484, 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/621bf66ddb7c962aa0d22ac97d69b793-Paper.pdf.

Yin-Cong Zhi, Yin Cheng Ng, and Xiaowen Dong. Gaussian processes on graphs via spectral kernel learning. *IEEE Transactions on Signal and Information Processing over Networks*, 2023. URL https://doi.org/10.1109/TSIPN.2023.3265160.

Juliusz Ziomek, Masaki Adachi, and Michael A Osborne. Beyond lengthscales: No-regret bayesian optimisation with unknown hyperparameters of any type. *arXiv preprint arXiv:2402.01632*, 2024. URL https://doi.org/10.48550/arXiv.2402.01632.