No-Regret Approximate Inference via Bayesian Optimisation

Rafael Oliveira¹,² Lionel Ott³ Fabio Ramos¹,²,⁴

¹ School of Computer Science, the University of Sydney, Australia
² ARC Centre for Data Analytics for Resources and Environments, Australia
³ Autonomous Systems Lab, ETH Zürich, Switzerland
⁴ NVIDIA, Seattle, WA, USA

Abstract

We consider Bayesian inference problems where the likelihood function is either expensive to evaluate or only available via noisy estimates. This setting encompasses application scenarios involving, for example, large datasets or models whose likelihood evaluations require expensive simulations. We formulate this problem within a Bayesian optimisation framework over a space of probability distributions and derive an upper confidence bound (UCB) algorithm to propose non-parametric distribution candidates. The algorithm is designed to minimise regret, which is defined as the Kullback-Leibler divergence with respect to the true posterior in this case. Equipped with a Gaussian process surrogate model, we show that the resulting UCB algorithm achieves asymptotically no regret. The method can be easily implemented as a batch Bayesian optimisation algorithm whose point evaluations are selected via Markov chain Monte Carlo. Experimental results demonstrate the method’s performance on inference problems.

1 INTRODUCTION

Bayesian inference problems have been traditionally solved via exact inference methods such as Markov chain Monte Carlo (MCMC) (Andrieu et al., 2003) when the likelihood function is reasonably cheap to evaluate. In these settings, often with the aid of other information, including gradients (Neal, 2011), MCMC methods explore posterior surfaces by performing typically thousands of evaluations until attaining a suitable sample-based posterior approximation. As a lightweight alternative, variational inference methods (Bishop, 2007) tradeoff approximation accuracy by sample-efficiency with methods which can reduce the number of likelihood evaluations by orders of magnitude (Ranganath et al., 2014). However, most state-of-the-art variational inference algorithms rely on gradient information from the probabilistic model within automatic inference frameworks (Salvatier et al., 2016; Bingham et al., 2019).

In some application scenarios in science and engineering, such as geophysical processes (Wellmann et al., 2010), ecological systems (Beaumont, 2010) and robotics (Ramos et al., 2019), likelihood evaluations are often expensive or intractable due to the use of complex simulations of physical phenomena. In the simulation of high-energy collisions in particle physics, for example, a single event may involve sampling of millions of random variables, making likelihood evaluations completely intractable (Brehmer et al., 2018).

When likelihood functions are not available, one resorts to likelihood-free inference methods, such as approximate Bayesian computation (ABC) (Robert, 2016), conditional density estimation (Papamakarios and Murray, 2016), synthetic likelihoods (Ong et al., 2018), etc. Simulations might still be expensive and limited by resource constraints, imposing challenges to traditional approaches, like ABC. Recent methods address the efficiency problem by sequentially learning approximations to the likelihood (Gutmann and Corander, 2016; Papamakarios et al., 2019), or the posterior directly (Greenberg et al., 2019), from simulation data. In particular, Gutmann and Corander (2016) derive an active learning approach using Bayesian optimisation (BO) (Shahriari et al., 2016) to propose simulation parameters.

In this paper, we address the problem of learning posterior approximations via Bayesian optimisation. We consider settings with black-box likelihood functions which might be noisy and expensive to evaluate. Contrary to the traditional point optimisation for BO, which only seeks extreme values, we address the problem of learning the distribution as a whole via non-parametric, sample-based approximations. For this purpose, we derive an upper confidence bound (UCB) algorithm to sample approximate posteriors learnt by a Gaussian process (GP) (Rasmussen and Williams, 2006) model on the log-likelihood. Compared to...
previous approaches using GP-based methods for approximate inference (Gutmann and Corander, 2016; Acerbi, 2018; Järvenpää et al., 2020), our method proposes a sequence of non-parametric sample-based posterior approximations which allow us to bound the Kullback-Leibler (KL) divergence between the estimate and the true unknown posterior. Theoretical results demonstrate that the algorithm is able to converge with sub-linear regret bounds towards the posterior distribution. We also provide experimental results, assessing the method’s performance in practice on approximate inference problems.

2 RELATED WORK

Bayesian optimisation approaches for black-box approximate inference have recently been proposed in the literature. In the context of ABC, Gutmann and Corander (2016) proposed learning a likelihood function from discrepancy evaluations of simulations using a UCB criterion for simulation parameters selection, with other acquisition rules later developed (Järvenpää et al., 2017).

Adaptive Bayesian quadrature (ABQ) methods (Kanagawa and Hennig, 2019) are also closely related to BO when considering a search space consisting of probability distributions. Acerbi (2018) presented variational Bayesian Monte Carlo (VBMC), an ABQ framework which sequentially learns variational approximations to a posterior distribution on top of a GP surrogate of the log-joint probability density between parameters and observations. More recently, Järvenpää et al. (2020) proposed a GP-based method to sequentially select informative batches of parameters for parallel likelihood evaluations. Their method selects batch elements based on an optimal posterior approximation derived from the current GP log-likelihood estimate. Although both ABQ methods do not explicitly involve an optimisation objective, the intermediate posterior approximations at each iteration can be seen as a sequence of candidate solutions optimising an implicit distributional objective.

Non-BO methods based on sequential approximations to a posterior are also plentiful in the likelihood-free inference literature (Papamakarios and Murray, 2016; Papamakarios et al., 2019; Greenberg et al., 2019; Durkan et al., 2020; Cranmer et al., 2020). These methods rely on neural density, or density-ratio, estimators to learn surrogate posterior densities from simulated data. Although our framework can also be applied to likelihood-free inference, we do not assume direct access to simulated data, but only to evaluations of a synthetic likelihood or discrepancy function.

Regarding theoretical results on posterior approximation errors, Kanagawa and Hennig (2019) analyse the convergence rate of adaptive Bayesian quadrature methods, which applies, for example, to Acerbi (2018, 2020). However, in this work, we are concerned with finite-time upper bounds on the posterior approximation error, defined by the KL divergence. We derive an algorithm which not only comes equipped with such finite-time guarantees, but which is also shown to achieve asymptotic optimality.

3 PROBLEM FORMULATION

Our goal is to estimate a posterior distribution over parameters \( \theta \in \Theta \subset \mathbb{R}^d \) given observed data \( x \in \mathcal{X} \):

\[
p_x(\theta) := p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} ,
\]

where we consider a likelihood \( p(x|\theta) \) only available via noisy and/or expensive evaluations, the evidence \( p(x) \) is assumed intractable, but we have a prior \( p(\theta) \).

We formulate the posterior estimation in Equation 1 as an optimisation problem over the space \( \mathcal{P} \) of probability distributions on \( \Theta \):

\[
q^* \in \operatorname{argmin}_{q \in \mathcal{Q}} \mathbb{D}_{KL}(q||p_x) ,
\]

where \( \mathcal{Q} \subset \mathcal{P} \) is a set of probability distributions over \( \Theta \), and \( \mathbb{D}_{KL}(q||p_x) \geq 0 \) denotes the Kullback-Leibler (KL) divergence between \( q \) and \( p_x \), which is such that \( \mathbb{D}_{KL}(q||p_x) = 0 \iff q = p_x \). The problem above can be shown to be equivalent to:

\[
q^* \in \operatorname{argmax}_{q \in \mathcal{Q}} \mathbb{E}_{q \sim \theta} [\ell(\theta)] - \mathbb{D}_{KL}(q||p) ,
\]

where \( \ell(\theta) := \log p(x|\theta) \) denotes the log-likelihood. The objective function in Equation 3 is known as the evidence lower bound (ELBO), as it lower bounds \( \log p(x) \).

Finding the global optimum in Equation 3 is usually intractable for an arbitrary log-likelihood function \( \ell \) and distributional search space \( \mathcal{Q} \). In addition, we also consider cases where the estimates of \( \ell \) are possibly noisy. Therefore, we make a few regularity assumptions on the log-likelihood function and its estimates.

Regularity assumptions: We assume \( \ell : \Theta \to \mathbb{R} \) lies in a reproducing kernel Hilbert space (RKHS) (Schölkopf and Smola, 2002). Given a positive-definite kernel \( k : \Theta \times \Theta \to \mathbb{R} \), a RKHS \( \mathcal{H}_k \) is a Hilbert space of functions with inner product \( \langle \cdot, \cdot \rangle_k \) and norm \( \|\cdot\|_k = \sqrt{\langle \cdot, \cdot \rangle_k} \), such that \( f(\theta) = \langle f, k(\cdot, \theta) \rangle_k \) for any \( f \in \mathcal{H}_k \) and any \( \theta \in \Theta \). We assume \( k \) is continuous and bounded with \( k(\theta, \theta) \leq 1, \forall \theta \in \Theta \), and that \( \|\ell\|_k \leq b \), where \( b > 0 \) is known.\(^1\)

Likelihood estimates: We consider a general sequential setting where we may not have direct access to the log-likelihood \( \ell \), but only to noisy estimates \( \hat{\ell} \). In particular, we

\(^1\)These assumptions are met by most of the popular kernels and are common in the online learning literature.
assume that the approximation error:

$$\epsilon_t := \ell(\theta_t) - \ell(\theta_*)$$  \hspace{1cm} (4)

is zero-mean \(\sigma_\epsilon\)-sub-Gaussian for a given \(\sigma_\epsilon > 0\) (Boucheron et al., 2013; Chowdhury and Gopalan, 2017). The sub-Gaussian assumption implies unbiased log-likelihood estimates with an error which is bounded or whose distribution tails decay at least as fast as a Gaussian. Examples include unbiased synthetic log-likelihood models (Ong et al., 2018) and mini-batching estimates when considering models with bounded log-likelihood functions.

**Regret:** Our performance analysis is based on the instant regret of an algorithm choosing a sequence of distributions \(\{q_t\}_{t \geq 1}\). The instant regret at iteration \(t \geq 1\) is defined as:

$$r_t := \mathbb{D}_{\text{KL}}(q_t || p_\mathbf{x}) - \mathbb{D}_{\text{KL}}(q^* || p_\mathbf{x}) .$$  \hspace{1cm} (5)

If our class of distributions \(Q\) is flexible enough, so that \(p_\mathbf{x} \in Q\), then \(\mathbb{D}_{\text{KL}}(q^* || p_\mathbf{x}) = 0\), setting \(q^* = p_\mathbf{x}\), and:

$$r_t = \mathbb{D}_{\text{KL}}(q_t || p_\mathbf{x}) .$$  \hspace{1cm} (6)

An upper bound on the cumulative regret \(R_T := \sum_{t=1}^{T} r_t\) tells us how far the candidates \(q_t\) picked by the algorithm are to the global optimum \(q^*\). In particular, the regret for the best candidate up to iteration \(T\) is bounded by:

$$\min_{1 \leq t \leq T} r_t \leq \frac{1}{T} R_T .$$  \hspace{1cm} (7)

Therefore, if \(R_T\) grows sub-linearly with \(T\), the term above vanishes as \(T \to \infty\) and the choices of the algorithm get arbitrarily close to the optimal \(q^*\) in terms of KL divergence with respect to the posterior \(p_\mathbf{x}\).

**4 BACKGROUND**

This section briefly reviews background on Gaussian processes, Bayesian optimisation via upper confidence bounds, and MCMC algorithms, which we apply for our results.

**4.1 GAUSSIAN PROCESS MODELS**

A Gaussian process with mean function \(m : \Theta \to \mathbb{R}\) and a positive-definite covariance function \(k : \Theta \times \Theta \to \mathbb{R}\) represents a prior over functions \(f : \Theta \to \mathbb{R}\) so that, for any finite collection of query points \(\{\theta_i\}_{i=1}^{N} \subset \Theta\), the vector \(\mathbf{f}_N = [f(\theta_1), \ldots, f(\theta_N)]^T\) is Gaussian, \(\mathbf{f}_N \sim \mathcal{N}(\mathbf{m}_N, \mathbf{K}_N)\), where \(\mathbf{K}_N_{ij} = k(\theta_i, \theta_j)\). Given a set of observations \(\mathcal{D}_N = \{(\theta_i, z_i)\}_{i=1}^{N}\), where \(z_i = f(\theta_i) + \epsilon_i\), and \(\epsilon_i \sim \mathcal{N}(0, \sigma^2)\), the posterior over \(f\) is:

$$f(\theta_*) | \mathcal{D}_N \sim \mathcal{N}(\mu_N(\theta_*), \sigma^2_N(\theta_*)) ,$$  \hspace{1cm} (8)

where:

$$\mu_N(\theta_*) = m(\theta_*) + k_N(\theta_*)^T (\mathbf{K}_N + \eta \mathbf{I})^{-1} (\mathbf{z}_N - \mathbf{m}_N)$$  \hspace{1cm} (9)

$$k_N(\theta, \theta') = k(\theta, \theta') - k_N(\theta)^T (\mathbf{K}_N + \eta \mathbf{I})^{-1} k_N(\theta')$$  \hspace{1cm} (10)

$$\sigma^2_N(\theta_*) = k_N(\theta_*, \theta_*) ,$$  \hspace{1cm} (11)

where \(k_N(\theta_*) := [k(\theta_1, \theta_*), \ldots, k(\theta_N, \theta_*)]^T\).

To model log-likelihoods, we need to consider that in log-scale low-likelihood values tend to \(-\infty\). Following previous works (Acerbi, 2018; Järvenpää et al., 2020), we use a negative quadratic mean function and, as covariance function, the squared-exponential kernel:

$$k(\theta, \theta') := \sigma^2 \exp \left(-\frac{1}{2 \ell^2} \|\theta - \theta'\|^2_2\right) , \ \theta, \theta' \in \Theta$$  \hspace{1cm} (12)

where \(\ell \geq 0\) and \(\sigma_f > 0\) are hyper-parameters (see Rasmussen and Williams, 2006, Ch. 4), and \(\|\cdot\|_2\) denotes the Euclidean 2-norm of a vector. However, our algorithm and results are not restricted to this particular choice of kernel.

**4.2 THE UPPER CONFIDENCE BOUND ALGORITHM**

The upper confidence bound (UCB) algorithm has been a popular method in the multi-armed bandits and Bayesian optimisation literature due to its theoretical properties and simplicity (Abbasi-Yadkori et al., 2010; Srinivas et al., 2010; Shahriari et al., 2016; Chowdhury and Gopalan, 2017). The algorithm selects queries to solve a sequential optimisation problem by maximising an upper confidence bound over the objective function. In the case of Gaussian processes, the algorithm selects queries by:

$$\theta_t \in \arg \max_{\theta \in \Theta} \mu_{t-1}(\theta) + \beta_{t-1}(\theta) \sigma^2_{t-1}(\theta) ,$$  \hspace{1cm} (13)

where \(\beta_t\) is a parameter which can be adjusted based on the information gain of the GP model, and \(\mu_t\) and \(\sigma^2_t\) define the GP posterior mean and variance, respectively, at time \(t \geq 0\). UCB algorithms enjoy bounded regret under reasonable assumptions in a point optimisation setting (Abbasi-Yadkori et al., 2010; Durand et al., 2018). In this paper, however, we consider a distribution optimisation setting.

**4.3 MARKOV CHAIN MONTE CARLO**

MCMC constitutes a broad class of sampling-based Bayesian inference algorithms, most of which enjoy asymptotic convergence guarantees (Andrieu et al., 2003). These methods formulate sampling as a Markov process whose stationary distribution is given by the desired target distribution, i.e. the posterior in Bayesian inference. The main building
5 DISTRIBUTIONAL BAYESIAN OPTIMISATION

We propose a BO algorithm for approximate Bayesian inference problems (Equation 2) with black-box and/or expensive-to-evaluate likelihood functions. The algorithm iterates between estimating posterior approximations and collecting likelihood evaluations, which are then passed on to update an internal GP model. In its general form, the algorithm may operate with any class of variational posteriors \( Q \). We, however, focus on the class of non-parametric sample-based approximations via MCMC, which offer theoretical and practical benefits, detailed in the next sections.

The algorithm is outlined in Algorithm 1. At each iteration, the algorithm selects a variational posterior by maximising a distributional acquisition function consisting of an upper-confidence bound on the ELBO (cf. Equation 3). From the variational posterior, we sample a batch of \( S \) parameters to evaluate the log-likelihood function \( \ell \) via its estimator \( \hat{\ell} \). The data from the evaluations is fed back to update the algorithm’s internal GP model, and the cycle repeats for a given number of \( T \) iterations. In the following, we present the proposed distributional acquisition function and a method to directly sample from its maximiser.

5.1 KULLBACK-LEIBLER UPPER CONFIDENCE BOUND

As the log-likelihood \( \ell(\theta) \) is the only unknown term in Equation 3, a GP model on \( \ell : \Theta \to \mathbb{R} \) allows us to derive an upper confidence bound on the ELBO. From this observation, we define the following acquisition function:

\[
h(q|D_t) := \mathbb{E}_{\theta \sim q}[u_t(\theta)] - \text{KL}(q||p), \quad t \geq 0,
\]

where \( u_t(\theta) := \mu_t(\theta) + \beta_t \sigma_t(\theta) \) corresponds to a pointwise UCB (Srinivas et al., 2010; Durand et al., 2018), and we derive settings for \( \beta_t > 0 \) in Section 6. With this function, we select candidate distributions as:

\[
q_t \in \text{argmax}_{q \in Q} h(q|D_{t-1}), \quad t \geq 1.
\]

Algorithm 1: KL-UCB

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( D_0 := \emptyset )</td>
</tr>
<tr>
<td>2</td>
<td>for ( t \in {1, \ldots, T} ) do</td>
</tr>
<tr>
<td>3</td>
<td>( q_t \leftarrow \text{argmax}_{q \in Q} h(q</td>
</tr>
<tr>
<td>4</td>
<td>Sample ( {\theta_{t,i}}_{i=1}^S \sim q_t )</td>
</tr>
<tr>
<td>5</td>
<td>Evaluate ( z_{t,i} = \hat{\ell}(\theta_{t,i}), \quad i \in {1, \ldots, S} )</td>
</tr>
<tr>
<td>6</td>
<td>Update GP with ( D_t := D_{t-1} \cup {\theta_{t,i}, z_{t,i}}_{i=1}^S )</td>
</tr>
</tbody>
</table>

Comparing \( h \) in Equation 15 with the main ELBO formulation (Equation 3), we notice that Equation 15 defines an ELBO with respect to a surrogate posterior obtained by taking the pointwise UCB \( u_t \) as a log-likelihood function:

\[
\hat{\rho}_{t-1}(\theta) := \frac{p(\theta) \exp u_{t-1}(\theta)}{\int_{\Theta} p(\theta) \exp u_{t-1}(\theta) \, d\theta}, \quad t \geq 1.
\]

The equation above defines a valid probability density function as long as \( u_t \) is bounded, which is usually the case for practical GP models. We formalise this result in Section 6 (Lemma 2). Due to its connection with the KL divergence, we refer to the acquisition function in Equation 15 as Kullback-Leibler upper confidence bound (KL-UCB).

5.2 OPTIMISATION OVER PROBABILITY DISTRIBUTIONS VIA MCMC

We now address the choice of the class \( Q \) of variational posteriors and how to sample from the optimal candidates \( q_t \in Q \) at each BO iteration (cf. Equation 16). As discussed in the previous section, KL-UCB as defined in Equation 15 corresponds to an ELBO with respect to \( \hat{\rho}_t \) in Equation 17, while also being an upper confidence bound on the main ELBO in Equation 3. Therefore, for a large enough class of distributions \( Q \), the maximiser \( q_t \) of Equation 16 is \( \hat{\rho}_{t-1} \) itself, and we may sample directly from it, bypassing the optimisation step in Equation 16.

To sample from the candidates \( q_t := \hat{\rho}_{t-1} \) at each BO iteration, we need to consider that the denominator in Equation 17 involves a possibly intractable integral. We address this issue by applying MCMC to sample, which only requires the unnormalised density \( p(\theta) \exp u_{t-1}(\theta) \).

Due to its asymptotic convergence guarantees (Andrieu et al., 2003), running MCMC with \( \hat{\rho}_{t-1} \) as our target distribution gives us samples which converge in distribution to \( \hat{\rho}_{t-1} \), the optimiser of Equation 16. We may see MCMC in this case as a form of optimisation over a space of non-parametric probability distributions. Further discussions on
the topic of MCMC as optimisation are available in the literature of gradient-based methods (Wibisono, 2018).

Note that we do not need to draw an infinite amount of samples from the MCMC chain. We only need to run it so that a batch of $S < \infty$ samples coming out of the chain approximately follow $\hat{p}_t$, i.e. after a long enough burn-in period, or by initialising the chain at a high-probability region, such as a mode of $\hat{p}_t$ (van Ravenzwaaij et al., 2018).

In our experiments, we use the EMCEE sampler (Foreman-Mackey et al., 2013) which can deal with the multi-modality of the UCB surface at reasonably fast mixing times.

6 THEORETICAL RESULTS

In this section, we present theoretical results on the regret of Algorithm 1 with respect to the objective in Equation 2. UCB’s theoretical guarantees are based on uniform concentration bounds. In particular, we use the following result for point-wise approximations.

**Theorem 1** (Durand et al. (2018)). Let $k(\theta, \theta') \leq 1$ for all $\theta, \theta' \in \Theta$, $\|\ell\|_k \leq b$ and $\epsilon_t$ be conditionally $\sigma_t$-sub-Gaussian. Then, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, uniformly over all $t \geq 0$ and $\theta \in \Theta$,

$$|\ell(\theta) - \mu_t(\theta)| \leq \beta_t(\delta) \sigma_t(\theta),$$

where $\beta_t(\delta) = b + \sigma_t \sqrt{2\eta_t \log((1 + \eta_t^{-1}K_t)^{1/2}/\delta)}$.

Notice that the confidence bound in Theorem 1 is simultaneous over all $\Theta$ and is currently the tightest known bound in the UCB literature. In our case, we extend the result above to a concentration bound over the space of probability distributions $\mathcal{P}$ with support on $\Theta$.

**Lemma 1.** Under our regularity assumptions (cf. Section 3), with probability at least $1 - \delta$, $\delta \in (0, 1)$, we have that the following holds for all $t \geq 0$:

$$|E_{\theta \sim q}[\ell(\theta)] - \mu_t(q)| \leq \beta_t(\delta) E_{\theta \sim q}[\sigma_t(\theta)], \quad \forall q \in \mathcal{P},$$

where $\beta_t(\delta) := b + \sigma_t \sqrt{2\eta_t \log((1 + \eta_t^{-1}K_t)^{1/2}/\delta)}$, $K_D := [k(\theta, \theta')]_{\theta, \theta' \in D}$ represents the GP covariance matrix for a set of observations $D$, and $|A|$ denotes the determinant of a square matrix $A$.

**Proof sketch.** This result follows by an application of Jensen’s inequality to the confidence bound in Durand et al. (2018, Theorem 1) for GP-UCB.

The result in Lemma 1 allows us to place a high-probability upper bound on the instant regret of Algorithm 1 when using the acquisition function in Equation 15 and the UCB parameter set to $\beta_t := \beta_t(\delta)$. In addition to the regularity assumptions in Section 3, we need to place conditions on the sampling process $\{\theta_t\}_{t=1}^S$ for the bounds to hold. We defer full proofs to the supplementary material.

The main result depends on the information gain of the GP, which is bounded by:

$$\gamma_N := \sup_{\Theta_N \subset \Theta: |\Theta_N| = N} I(\mathbf{z}_N, \mathbf{g}_N|\Theta_N),$$

where $I(\mathbf{z}_N, \mathbf{g}_N|\Theta_N)$ denotes the mutual information between $\mathbf{z}_N = \mathbf{g}_N + \epsilon'_N$ and $\mathbf{g}_N \sim N(0, \mathbf{K}_N)$, with $[\mathbf{K}_N]_{ij} = k(\mathbf{\theta}_i, \mathbf{\theta}_j), \mathbf{\theta}_i, \mathbf{\theta}_j \in \Theta_N$ and $\epsilon'_N \sim N(0, \eta I)$. For a compact parameter space $\Theta$, or equivalently when dealing with a compactly supported prior $p(\theta)$, the supremum in Equation 19 corresponds to a maximum, which is referred to as the maximum information gain. For popular choices of kernels, information gain bounds are available in the literature, such as the squared exponential kernel, which is $O((\log T)^{d+1})$ (Srinivas et al., 2010). We can now provide the following bound on the cumulative regret of Algorithm 1.

**Theorem 2.** Under the same assumptions in Lemma 1, running Algorithm 1 with $h(q|D_{t-1})$ given by Equation 15 and $\beta_t := \beta_t(\delta)$, we obtain a bound on the instant regret:

$$r_t \leq 2\beta_{t-1}(\delta) E_{\eta_t}[\sigma_{t-1}], \quad t \geq 1,$$

and on the cumulative regret at $T \geq 1$:

$$R_T \leq 4\beta_T(\delta) \sqrt{(T + 2)\gamma_T} \in O(\sqrt{T}(b\sqrt{T} + \sqrt{T\gamma_{ST}}))$$

which hold with probability at least $1 - \delta$.

**Proof sketch.** The first result follows by applying the upper confidence bound on $\ell$ to the regret defined in Equation 5 for an optimal choice according to Equation 16. The second result follows a similar derivation to previous bounds in the literature (cf. Chowdhury and Gopalan, 2017, Theorem 2).

For common choices of kernel functions, such as the squared exponential and the Matérn kernels, the maximum information gain has sub-linear growth (Srinivas et al., 2010). Consequently, Theorem 2 shows that Algorithm 1 achieves vanishing regret under these settings, which allows the algorithm to find arbitrarily close approximations to $q^*$ within a finite amount of time. In particular, let us consider the setting where we run exact inference on Equation 16.

**Lemma 2.** For any bounded kernel $k$ and bounded GP mean function $m$, maximising $h$ in Equation 15 at any $t \geq 1$ is equivalent to:

$$\arg\max_{q \in \mathcal{Q}} h(q|D_{t-1}) = \arg\min_{q \in \mathcal{Q}} D_{KL}(q||\tilde{p}_{t-1}).$$

Lemma 2 tells us that the maximiser of $h(q|D_{t-1})$ is unique and given by $\tilde{p}_{t-1}$ when $\tilde{p}_{t-1} \in \mathcal{Q}$. In fact, we can sample from $\tilde{p}_{t-1}$ via MCMC, which leads us to the final result.
Corollary 1. Under the regularity assumptions in Section 3, with probability greater than $1 - \delta$, $\delta \in (0, 1)$, we have:

$$\forall t \geq 0, \quad \mathbb{D}_{KL}(\hat{p}_t || p_x) \leq 2\beta_1(\delta)E_{\hat{p}_t}[\sigma_1].$$  

In particular, for $\gamma_T \in \mathcal{O}(T^\alpha)$ with $\alpha < 1/2$, we have:

$$\lim_{T \to \infty} \min_{t \leq T} \mathbb{D}_{KL}(\hat{p}_t || p_x) = 0.$$  

This result provides a finite-sample bound on the KL divergence of posterior approximations (see Appendix C for kernel-specific bounds) and confirms that the GP-based posterior approximation asymptotically approaches the true posterior $p_x$ for the case of popular kernels, such as the squared exponential and the Matérn family, which lead to sub-linear regret bounds (Srinivas et al., 2010). The result also tells us that a sequence of MCMC samples over a UCB-based likelihood leads us to optimal posterior approximations and a no-regret approximate inference algorithm.

6.1 CONNECTION WITH OTHER APPROACHES

Contrast with point-based approach: In contrast with applying UCB to learn a log-likelihood model via a point-maximisation problem (Gutmann and Corander, 2016). Theorem 2 establishes a regret bound on distribution choices. The distributional regret bound is $O(\sqrt{T\gamma_T} + \sqrt{T\gamma_{ST}})$, essentially smaller than the regret bound of a GP-UCB algorithm given the same number of likelihood evaluations $N = ST$, which is $O(\sqrt{ST\gamma_{ST}} + \gamma_{ST})$ (Chowdhury and Gopalan, 2017) by a factor dependent on the sample batch size $S$, since $\gamma_{ST} \geq \gamma_T$ by monotonicity. In addition, samples from $q_t$ consider the likelihood surface as a whole via the expectation within the ELBO, while a pointwise UCB algorithm would be ultimately concerned with finding the parameter of maximum likelihood.

Connection with VBMC: VBMC (Acerbi, 2018) selects variational posteriors by maximising a lower confidence bound on the ELBO based on GP estimates. For a GP model on the log-likelihood², VBMC optimises:

$$h_{\text{VBMC}}(q|D_t) := E_{\hat{q}}[\mu_t] - \mathbb{D}_{KL}(\hat{q} || p) - \beta\sigma_1(q),$$  

where $\beta \geq 0$ is fixed (e.g. $\beta = 3$), and $\sigma^2_{t-1}(q) := \int_\Theta \int_\Theta k_{t-1}(\theta, \theta')q(\theta)q(\theta')d\theta d\theta'$ corresponds to the variance of the expected log-likelihood. From the variational posterior $q^{\text{VBMC}}_t$ that maximises $h_{\text{VBMC}}(q|D_{t-1})$, VBMC selects a sequence of points for evaluation:

$$\theta^{\text{VBMC}}_{t+1} \in \arg\max_{\theta \in \Theta} \sigma^2_{t+1}(\theta)q^{\text{VBMC}}_t(\theta) \exp \mu_{t+1}(\theta),$$  

where each point is chosen after updating the GP with the evaluation at the previous point for a total of $S_{\text{VBMC}}$ points (set to 5 by default). VBMC uses a class of non-parametric variational posteriors consisting of mixtures of Gaussians with an adaptible number of components and performs a warm-up run to optimise GP hyper-parameters.

Equation 25 corresponds to a lower confidence bound on the ELBO (Acerbi, 2018) in contrast to KL-UCB (cf. Equation 15), which sets an upper confidence bound. Therefore, if the lower confidence bound in VBMC is replaced by KL-UCB, the algorithm should have a similar distributional regret bound to the one in Theorem 2, since the stochastic process producing the batch does not violate the assumptions in Durand et al. (2018) for Theorem 1 (see also the assumptions in Chowdhury and Gopalan (2017)).

6.2 LIMITATIONS

Our distributional Bayesian optimisation algorithm KL-UCB comes with strong theoretical guarantees allowing one not only to learn approximations of a posterior, but also to quantify the error in the approximation. However, the algorithm also comes with its own limitations. Although not explicit in the regret bounds, the dimensionality of the parameter space $\Theta$ affects the bounds via the dependence of the maximum information gain $\gamma_T$ on it. For instance, the squared-exponential kernel (Equation 12) yields $\gamma_T \in \mathcal{O}(\log^{3/2} T)$ (Srinivas et al., 2010), which is exponential in the dimension of the parameter space. In addition, the inversion of the kernel matrix required for GP predictions scales as $O(N^3)$ in time complexity. Therefore, our method is more suitable for applications with low-dimensional parameter spaces and where we are limited to a small number of likelihood evaluations. Sparse approximations to GPs (Günnerts and Metta, 2013; Hensman et al., 2013) and advances in BO methods for high-dimensional search spaces (Wang et al., 2018; Mutný and Krause, 2018) may alleviate these issues, but adapting these frameworks to this problem setup is objective of future research.

7 EXPERIMENTS

In this section, we present empirical assessments of the proposed KL-UCB algorithm to complement the theoretical results in the previous section. We start with experiments assessing the theoretical regret bounds in situations where the modelling assumptions hold. We then follow with experiments comparing KL-UCB against VBMC (Acerbi, 2018), as an adaptive Bayesian quadrature baseline, in two test cases where the log-likelihood function is not in the RKHS defined by the GP kernel. Lastly, we present an experiment with a likelihood-free inference problem using a single-point UCB strategy as a baseline (Gutmann and Corander, 2016).
For all experiments, KL-UCB was configured with a standard normal prior and a squared exponential kernel. To perform inference over the UCB-based posterior approximation \( \hat{p}_{t-1} \), we run the ensemble-based MCMC sampler EMCEE (Foreman-Mackey et al., 2013) set with 25 walkers to draw each iteration’s batch of \( S := 5 \) after a burn-in period. We set \( \delta := 0.2 \) and adjust the kernel hyper-parameters accordingly. Appendix E provides additional experiment details.

Results are reported in terms of the number of evaluations of the log-likelihood. As performance metrics, we use both the KL divergence and the Gaussian symmetrised KL divergence (gsKL) used by Acerbi (2018). The latter is defined as the symmetric KL divergence between the Gaussian distributions formed by the first two moments of the approximate and the true posterior distribution. In cases where the posterior distribution is not available in closed form, we also run MCMC and kernel density estimation to recover a reference posterior density.

Ablation experiments in the supplementary material analyse the effect of dimensionality in the current framework (see Appendix D). Code and further ablation experiments are also to be made available online.\(^3\)

### 7.1 THEORY ASSESSMENT

We first assess how the theoretical results reflect in terms of practical performance in settings where the regularity assumptions are known to hold. For this experiment, we generate log-likelihood functions in the RKHS \( \mathcal{H}_k \):

\[
\ell := \sum_{i=1}^{M} w_i k(\cdot, \theta_i) \in \mathcal{H}_k ,
\]

where we sample \( w_i \sim N(0, 1) \), \( \theta_i \sim p(\theta) \), for \( i \in \{1, \ldots, M\} \), with a fixed \( M := 40 \).

Figure 1 presents the posterior approximation results for a 1-D RKHS log-likelihood. As seen in Figure 1a, the final posterior approximation, which uses the GP posterior mode as a log-likelihood, mostly matches the true posterior. The exception is a slight under-estimation of the second, smaller mode. This, however, is a well-known effect of using the reverse KL, which induces a mode-seeking behaviour (Wang and Titterington, 2005; Bishop, 2007).

In Figure 2b, the regret plot shows that the bound in Theorem 2 is large enough to accommodate for the measured KL divergence between the proposed distributions and the true posterior. The regret bound is calculated based on the formulation of \( \beta_2 \) in Theorem 1 using the norm of \( \ell \) in \( \mathcal{H}_k \). Details about the KL divergence estimation are available in the supplementary material (see Appendix E).

\(^3\)Code repository: https://github.com/rafaol/no-regret-approximate-inference-via-bo

### 7.2 COMPARISONS

In this section, we test KL-UCB on settings where the main assumption \( \ell \in \mathcal{H}_k \) does not hold. As a related approximate inference baseline, we use VBMC (Acerbi, 2018).

**Lumpy likelihood:** This likelihood function is formulated as a mixture of equally weighted Gaussians:

\[
p(x|\theta) = \exp(\ell(\theta)) = \frac{1}{M} \sum_{i=1}^{M} N(\theta; \theta_i, \Sigma_i) ,
\]

where we sample \( \theta_i \sim U[0, 1]^d \) and \( \Sigma_i = \text{diag}(\sigma_{i,1}^2, \ldots, \sigma_{i,d}^2) \) with \( \sigma_{i,j} \sim U[0.2, 0.6] \), \( j \in \{1, \ldots, d\} \) for a fixed number of components \( M := 12 \), following the setup in Acerbi (2018). Placing a Gaussian prior \( p(\theta) := N(\theta; 0, \Sigma_p) \) with \( \sigma_p := 0.5 \) on \( \theta \), the resulting posterior is available in closed-form as also a mixture of Gaussians with properly adjusted means and covariances.

Figure 2 presents the results for the experiment on the lumpy likelihood function. As we can see, KL-UCB’s performance
is surpassed by that of VBMC. As this problem setting involves a mixture of Gaussians as a likelihood, VBMC has a natural advantage, since this algorithm uses variational posteriors based on mixtures of Gaussians as well.

**Circular likelihood:** We define a circular 2-dimensional log-likelihood function by:

\[
\ell(\theta) := -\left(\|\theta - \theta_c\|_2 - \rho_c\right)^2 / s_c,
\]  

(29)

where \(\theta_c := 0\) denotes the circle centre, \(\rho_c := 1.5\) is the circle radius, and \(s_c := 0.25\) is a length-scale.

In Figure 3, we see the performance results for the circular likelihood. This time KL-UCB noticeably outperforms VBMC. One of the reasons for this outcome is that to be able to cover the narrow circle defined by the likelihood function, VBMC needs a large number of mixture components (reaching 50 in our tests). KL-UCB instead samples on top of the non-parametric surface defined by the GP directly.

### 7.3 LIKELIHOOD-FREE INFEERENCE

We applied KL-UCB to a likelihood-free inference problem. In this experiment, we infer the parameters of a classic control environment in OpenAI Gym: the cart-pole. We fix a given setting for its physics parameters \(\theta_{\text{real}}\) and generate a dataset of 10 trajectories by executing randomly sampled actions. Summary statistics \(\xi\) were the same as Ramos et al. (2019), which consist of state-action inner products and the first two moments of the trajectory state-differences distribution. The discrepancy was set to \(\Delta_\theta := \|\xi_\theta - \xi_{\text{real}}\|_2^2 / \sigma^2\), and we evaluate \(\ell(\theta) := -\Delta_\theta\). We also placed a Gaussian prior \(p(\theta) := N(\theta; 0, I)\) on the parameters.

Our GP model was configured with the Matérn kernel set with smoothness parameter \(\nu := 3/2\) (Rasmussen and Williams, 2006). Other hyper-parameters include the kernel length-scales, signal variance and noise variance, and were adjusted based on maximum a posteriori estimates from previous runs. The GP mean function was set as the log-prior probability density \(m(\theta) := \log p(\theta)\).

We compared KL-UCB against GP-UCB in this experiment. Both methods were configured with a fixed \(\beta_t := 3\), \(\forall t \geq 0\). The main difference between the methods is that GP-UCB sequentially selects single points for evaluations, which maximise the pointwise UCB, while KL-UCB selects an i.i.d. batch of points according to the UCB-based posterior approximation. The GP-UCB approach in this case corresponds to the approach by Gutmann and Corander (2016), providing a baseline as a point-based likelihood-learning method.

Figure 4 presents the final posterior approximations obtained by KL-UCB and the KL-divergence-based regret for each method. As a reference for the "true" posterior, we ran rejection ABC with an adaptable threshold parameter to obtain 1000 samples, and then fit a kernel density estimator to measure the KL divergences. As the plots show, KL-UCB was able to obtain approximations much closer to the true posterior when compared to GP-UCB. It is worth noting that the true posterior’s mass is highly concentrated on a narrow region around the true parameters, making the inference problem relatively hard for a point-by-point method.

---

\(^4\)OpenAI Gym: [https://gym.openai.com](https://gym.openai.com)

\(^5\)ABC code: [https://github.com/elfi-dev/elfi]
In terms of regret, we measured the KL divergence between the corresponding UCB-based posterior approximation and the true posterior for both methods. These results are also in Figure 4. Even though GP-UCB and KL-UCB had the same settings for their GP models, the results show that, within the given budget of 200 evaluations of the discrepancy function, GP-UCB was not able to obtain reasonable posterior approximations. Its performance is also more unstable.

8 CONCLUSION

This paper presented an approach for approximate Bayesian inference via Bayesian optimisation in settings where we need to evaluate black-box likelihood functions. The algorithm is composed of an acquisition function formulated as an upper confidence bound over the ELBO of a variational objective. In contrast to traditional variational inference approaches, the algorithm samples the non-parametric posterior approximation defined by a GP model over the log-likelihood function to propose evaluation points at each iteration. The use of a flexible non-parametric class of distributions allows us to propose a sequence of optimal distributional candidates, which theoretical results have shown to lead to a vanishing KL divergence. A set of experiments also demonstrated the method in practice. As future work, the method offers a few avenues for improvement, such as possible batch design approaches (Järvenpää et al., 2020) on top of the MCMC samples, scalability to high-dimensional spaces and applications to general optimisation problems.

Bibliography


David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In Kamalika Chaudhuri and Ruslan


APPENDIX

This appendix complements the main paper with proofs, additional results and experiment details. In Appendix A, we start with additional theoretical background. Appendix B presents full proofs for the main theoretical results in the paper. Appendix C provides kernel-specific theoretical upper bounds for the KL divergence of posterior candidates obtained by KL-UCB. In Appendix D, we present a sensitivity analysis on dimensionality effects on KL-UCB. Finally, in Appendix E, we conclude with further details on the experiments setup.

A FURTHER BACKGROUND

Definition 1. Let $Q$ and $P$ be two probability measures such that $Q$ is absolutely continuous with respect to $P$. The Kullback-Leibler (KL) divergence between $Q$ and $P$ is defined as:

$$D_{KL}(Q||P) := \int \log \left( \frac{dQ}{dP} \right) dQ. \quad (30)$$

Case $Q$ and $P$ are both defined on a Euclidean space $\mathbb{R}^d$ and are absolutely continuous with respect to the Lebesgue measure on this space, the equation above can be rewritten as:

$$D_{KL}(Q||P) = D_{KL}(q||p) := \int_{\mathbb{R}^d} q(\theta) \log \left( \frac{q(\theta)}{p(\theta)} \right) d\theta, \quad (31)$$

where $q(\theta) := \frac{dQ}{d\theta}$ and $p(\theta) := \frac{dP}{d\theta}$ are the probability density functions of $Q$ and $P$, respectively.

GP predictive equations: The theoretical results are given in terms of a Gaussian process model $G\mathcal{P}(m, k)$ for the log-likelihood function $\ell : \Theta \to \mathbb{R}$ which is learnt with batches of $S$ i.i.d. samples from a probability distribution $q_t$ at each iteration $t \in \{1, \ldots, T\}$. Therefore, at each iteration $t \geq 1$, the GP predictive mean and variance are given by:

$$\mu_t(\theta) := m(\theta) + k_{N_t}(\theta) (K_{N_t} + \eta I)^{-1} (z_{N_t} - m_{N_t}) \quad (32)$$

$$k_t(\theta, \theta') := k(\theta, \theta') - k_{N_t}(\theta) (K_{N_t} + \eta I)^{-1} k_{N_t}(\theta') \quad (33)$$

$$\sigma^2_t(\theta) := k_t(\theta, \theta), \quad (34)$$

where $k_{N_t}(\theta) := \begin{bmatrix} k(\theta, \theta_1), \ldots, k(\theta, \theta_{N_t}) \end{bmatrix}^T$, $[K_{N_t}]_{i,j} := k(\theta_i, \theta_j)$, $i,j \in \{1, \ldots, N_t\}$, and $N_t := tS$.

B PROOFS

We here present full proofs for the theoretical results in the paper. We start with a few auxiliary results and then follow with the proofs for the main results.

B.1 AUXILIARY RESULTS

Lemma 3. In our settings, the GP posterior variance is always bounded, i.e.: 

$$\forall t \geq 0, \|\sigma_t\|_\infty \leq 1. \quad (35)$$

Proof. As the kernel $k$ is bounded, with $k(\theta, \theta) \leq 1$, $\forall \theta \in \Theta$, the definition in Equation 34 leads to:

$$\forall t \geq 1, \sigma^2_t(\theta) \leq k(\theta, \theta) \leq 1, \forall \theta \in \Theta. \quad (36)$$

Lemma 4. Assuming a bounded mean function $m : \Theta \to \mathbb{R}$, the GP posterior mean is bounded, i.e.: 

$$\forall t \geq 0, \|\mu_t\|_\infty < \infty. \quad (37)$$

Proof. As defined in Equation 32, for $t \geq 1$, we have:

$$\mu_t(\theta) := m(\theta) + k_{N_t}(\theta) (K_{N_t} + \eta I)^{-1} (z_{N_t} - m_{N_t}) \leq m(\theta) + \eta^{-1} k_{N_t}(\theta) (z_{N_t} - m_{N_t}) \leq \|m\|_\infty + \eta^{-1} \|k_{N_t}(\theta)\|_1 \|z_{N_t} - m_{N_t}\|_\infty \leq \|m\|_\infty + \eta^{-1} f(\|z_{N_t}\|_\infty + \|m\|_\infty) < \infty, \forall \theta \in \Theta, \quad (38)$$

where we first used the fact that $(K_{N_t} + \eta I)^{-1} = \eta^{-1} (\eta^{-1} K_{N_t} + I)^{-1} \leq \eta^{-1} I$ due to positive-definiteness, then applied Hölder’s inequality, and finally the observation that $\|k_{N_t}(\theta)\|_1 \leq t$, due to the kernel bound.

Lemma 5 (Srinivas et al. (2010, Lemma 5.3)). The information gain for a sequence of evaluations $\{\theta_i, z_i\}_{i=1}^N$, where $z_i = f(\theta_i) + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, \eta)$, can be expressed in terms of the predictive variances. Namely, if $f \sim G\mathcal{P}(m, k)$, then:

$$I(z_N, f_N|\Theta_N) = \frac{1}{2} \sum_{i=1}^N \log(1 + \eta^{-1} \sigma^2_{t-1}(\theta_i)). \quad (39)$$

Lemma 6 (Chowdhury and Gopalan (2017, Lemma 4)). Following the setting of Lemma 5, the sum of predictive standard deviations at a sequence of evaluation points is bounded in terms of the maximum information gain:

$$\sum_{i=1}^N \sigma_{t-1}(\theta_i) \leq \sqrt{4(N+2)\gamma_N}. \quad (40)$$

Lemma 7. Let $A \subset \Theta$ be a set of points where a function $f \sim G\mathcal{P}(m, k)$ was evaluated, so that the GP posterior covariance function and the corresponding variance are given by:

$$k_A(\theta, \theta') := k(\theta, \theta') - k(\theta, A)^T (K(A) + \eta I)^{-1} k(A, \theta'), \quad (41)$$

$$\sigma^2_A(\theta) := k_A(\theta, \theta), \quad \theta, \theta' \in \Theta, \quad (42)$$


\[ k(\theta, A) := \langle k(\theta, a) \rangle_{a \in A} \quad \text{and} \quad K(A) := \langle k(a, a') \rangle_{a, a' \in A}. \]

Then, for any given set \( B \supset A \) of evaluations of \( f \), we have:

\[
\sigma_A^2(\theta) \leq \sigma_B^2(\theta), \quad \forall \theta \in \Theta.
\]

**Proof.** The result follows by observing that the GP posterior given observations at \( A \) is a prior for the GP with the new observations at the complement \( C := B \setminus A \). Then we obtain, for all \( \theta \in \Theta \):

\[
\begin{align*}
\sigma_B^2(\theta) & := k(\theta, \theta) - k(\theta, B)(K(B) + \eta I)^{-1}k(B, \theta) \\
& = \sigma_A^2(\theta) - k_A(\theta, C)(K_A(C) + \eta I)^{-1}k_A(C, \theta) \\
& \leq \sigma_A^2(\theta),
\end{align*}
\]

since \( k_A(\theta, C)(K_A(C) + \eta I)^{-1}k_A(C, \theta) \) is non-negative. \( \square \)

**B.2 PROOF OF LEMMA 1**

We prove Lemma 1 by applying Jensen’s inequality to Theorem 1. Specifically, consider that the following holds with probability greater than \( 1 - \delta \) simultaneously over all \( \Theta \):

\[
\forall t \geq 0, \quad |\ell(\theta) - \mu_t(\theta)| \leq \beta_t(\delta)\sigma_t(\theta). \quad (45)
\]

With the same probability, we then have that:

\[
\begin{align*}
|E_{\theta \sim q_t}[\ell(\theta) - \mu_t(\theta)]| & \leq E_{\theta \sim q_t}[|\ell(\theta) - \mu_t(\theta)|] \\
& \leq \beta_t(\delta)E_{\theta \sim q_t}[\sigma_t(\theta)],
\end{align*}
\]

since the absolute value \(|\cdot| \) is convex. Lastly, we note that the GP in Algorithm 1 is taking batches of samples per iteration. Therefore, we have to replace the original \( K_t \), which holds \( t \) observations, by \( K_\beta \), which holds all the \( t \) observations collected up to iteration \( t \).\(^6\) This concludes the proof.

**B.3 PROOF OF THEOREM 2**

We prove Theorem 2 in two parts, one for each component of the result. In the first part we establish a bound on the instant regret. We then propagate this bound to the cumulative regret. For these derivations, we will make use of Lemma 1 and some of the auxiliary results in Section B.1.

\[ r_t \leq 2\beta_{t-1}(\delta)E_{q_t}[\sigma_{t-1}]. \quad (47) \]

**Proof of Theorem 2.** We start by proving that, with \( h(q|D_{t-1}) \) given by KL-UCB and \( \beta_t := \beta_t(\delta) \) according to Theorem 1, we obtain a bound on the instant regret:

\[ r_t \leq 2\beta_{t-1}(\delta)E_{q_t}[\sigma_{t-1}]. \quad (47) \]

By Lemma 1, uniformly over all \( t \geq 1 \) with probability at least \( 1 - \delta \), we have:

\[
\begin{align*}
\max_{q \in Q} E_q[\ell] - D_{KL}(q||p) & \leq \max_{q \in Q} E_q[u_t-1] - D_{KL}(q||p) \\
& = E_{q_t}[u_t-1] - D_{KL}(q_t||p) .
\end{align*}
\]

Applying this bound to the definition of instant regret yields the first part of Theorem 2:

\[
\begin{align*}
r_t := D_{KL}(q_t||p_\star) & - D_{KL}(q^*||p_\star) \\
& = \max_{q \in Q} E_q[\ell] - D_{KL}(q||p) - E_q[\ell] + D_{KL}(q||p) \\
& \leq E_{q_t}[u_{t-1}] - E_q[\ell] \\
& = E_{\theta \sim q_t}[\mu_t(\theta) + \beta_{t-1}\sigma_{t-1}(\theta) - \ell(\theta)] \\
& \leq 2\beta_{t-1}E_{\theta \sim q_t}[\sigma_{t-1}(\theta)],
\end{align*}
\]

which holds with probability at least \( 1 - \delta \).

For the second part, we apply the bound to the cumulative regret and initially obtain:

\[
R_T := \sum_{t=1}^T r_t \leq 2 \sum_{t=1}^T \beta_{t-1}E_{\theta \sim q_t}[\sigma_{t-1}(\bar{\theta}_t)].
\]

since \( \beta_t \geq \beta_{t-1}, \forall t \geq 1 \), and expectations are linear operations. Considering the predictive variances above, recall that, at each iteration \( t \geq 1 \), the algorithm selects a batch of i.i.d. points \( B_t := \{ \theta_t, i \}_{i=1}^S \), sampled from \( q_t \), where to evaluate the log-likelihood function \( \ell \). The predictive variance \( \sigma_t^2 \) is conditioned on all previous observations, which are grouped by batches. We can then decompose, for any \( t \geq 1 \):

\[
\sigma_t^2(\theta) = \sigma_{t-1}^2(\theta) - k_{t-1}(\theta, B_t)(K_{t-1}(B_t) + \eta I)^{-1}k_{t-1}(B_t, \theta),
\]

where we use the notation introduced in Lemma 7, and:

\[
\begin{align*}
k_t(\theta, \theta') & := k_t(\theta, \theta') \\
k_{0}(\theta, \theta') & := k(\theta, \theta').
\end{align*}
\]

\[ k_0(\theta, \theta') := k(\theta, \theta'). \quad (53) \]

\[ k_0(\theta, \theta') := k(\theta, \theta'). \quad (53) \]
Therefore, the predictive variance of the batched algorithm is not the same as the predictive variance of a sequential algorithm, and we cannot directly apply Lemma 6 to bound the last term in Equation 50.

Lemma 7 tells us that the predictive variance given a set of observations is less than the predictive variance given a subset of observations. Selecting only the first point from within each batch and applying Lemma 7, we get, for \( t \geq 1 \):

\[
\sigma_t^2(\theta) \leq s_t^2(\theta) := k(\theta, \theta) - k(\theta, \Theta_t)(K(\Theta_t) + \eta I)^{-1}k(\Theta_t, \theta),
\]

where \( \Theta_t := \{\theta_{t,1}\}_{i=1}^t \), with \( \theta_{t,1} \in B_i, i \in \{1, \ldots, t\} \).

Note that the right-hand side of the equation above is simply the non-batched GP predictive variance. Furthermore, sample points within a batch are i.i.d., so that \( \theta_{t,1} \sim q_t \) and \( \tilde{\theta}_t \sim q_t \) are identically distributed. We can now apply Lemma 6, yielding:

\[
\mathbb{E}_{\tilde{t}_1 \sim q_t, \ldots, \tilde{t}_T \sim q_T} \left[ \sum_{t=1}^T \sigma_{t-1}(\tilde{\theta}_t) \right]
\leq \mathbb{E}_{\tilde{t}_1 \sim q_t, \ldots, \tilde{t}_T \sim q_T} \left[ \sum_{t=1}^T s_{t-1}(\tilde{\theta}_t) \right]
\leq 2\sqrt{(T + 2)\gamma_T}.
\]

Combining this result with Equation 50, we obtain:

\[
R_T \leq 4\beta_T \sqrt{(T + 2)\gamma_T} \in \mathcal{O}(\beta_T \sqrt{T\gamma_T}).
\]

Lastly, from the definition of \( \beta_T(\delta) \), we have:

\[
\beta_T(\delta) := b + \sigma e \sqrt{\frac{2\eta^{-1}\log((1 + \eta^{-1})\gamma_T^{1/2})}{\delta}},
\]

where:

\[
\log((1 + \eta^{-1})\gamma_T^{1/2}) = I(z_{NT}, g_{NT}) \leq \gamma_{NT} = \gamma_{ST},
\]

for \( g \sim \mathcal{GP}(m, k) \). Therefore, the KL-UCB cumulative regret is such that:

\[
R_T \in \mathcal{O}(\sqrt{T(b\sqrt{\gamma_T} + \sqrt{\gamma_T\gamma_{ST}})}),
\]

which concludes the proof.

\[ \square \]

### B.4 PROOF OF LEMMA 2

Lemma 2 states that, for a bounded kernel \( k \), maximising \( h \) at any \( t \geq 1 \) is equivalent to:

\[
\arg\max_{q \in \mathcal{Q}} E_q[h_{D_{t-1}}] = \arg\min_{q \in \mathcal{Q}} D_{KL}(q || \tilde{p}_{t-1}),
\]

where \( h \) is defined as:

\[
h(q|D_{t-1}) := E_{\theta \sim q}[u_{t-1}(\theta)] - D_{KL}(q || p),
\]

with \( u_t(\theta) := \mu_t(\theta) + \beta_t\sigma_t(\theta), 0 \leq \beta_t < \infty, \) and \( \mu_t \) and \( \sigma_t^2 \) define GP posterior mean and variance at iteration \( t \). The proof follows by the same argument which turns the general KL divergence minimisation problem into an ELBO maximisation. The only part to verify is whether \( \tilde{p}_{t-1} \) defines a valid probability density function.

**Proof of Lemma 2.** In general, let \( f : \Theta \rightarrow \mathbb{R} \) be a bounded function on \( \Theta \subset \mathbb{R}^d \), i.e. \( \|f\|_{\infty} < \infty \), and \( p \) a probability density function on \( \Theta \). Then we have that:

\[
\mathbb{E}_p[f] := \int_{\Theta} f(\theta)p(\theta) \, d\theta \leq \|f\|_{\infty} \int_{\Theta} p(\theta) \, d\theta = \|f\|_{\infty}.
\]

(62)

Combining Lemma 3 and Lemma 4, we have that \( u_t \) is always bounded, since:

\[
\forall t \geq 1, \quad u_t(\theta) := \mu_t(\theta) + \beta_t\sigma_t(\theta) \leq \|\mu_t\|_{\infty} + \beta_t\|\sigma_t\|_{\infty} < \infty, \quad \forall \theta \in \Theta.
\]

(63)

Then the normalisation constant \( \zeta_t \) for \( \hat{p}_t(\theta) = \frac{1}{\zeta_t}p(\theta) \exp u_t(\theta) \) is bounded, for:

\[
\forall t \geq 0, \quad \zeta_t := \mathbb{E}_p[\exp u_t] = \int_{\Theta} p(\theta) \exp u_t(\theta) \, d\theta \leq \exp\|u_t\|_{\infty} < \infty.
\]

(64)

Now, for the lemma’s main result, we obtain:

\[
\arg\max_{q \in \mathcal{Q}} \mathbb{E}_q[u_{t-1}] - D_{KL}(q || p)
\]

\[
= \arg\max_{q \in \mathcal{Q}} \mathbb{E}_q[u_{t-1} + \log p - \log q]
\]

\[
= \arg\min_{q \in \mathcal{Q}} \mathbb{E}_p[\exp u_{t-1}] - \mathbb{E}_q[u_{t-1} + \log p - \log q]
\]

\[
= \arg\min_{q \in \mathcal{Q}} \mathbb{E}_p[\exp u_{t-1}] \int_{\Theta} q(\theta) \log \left( \frac{q(\theta)\mathbb{E}_p[\exp u_{t-1}]}{p(\theta)\exp u_{t-1}} \right) \, d\theta
\]

\[
= \arg\min_{q \in \mathcal{Q}} D_{KL}(q || \hat{p}_{t-1}),
\]

(65)

which concludes the proof.

\[ \square \]

### B.5 PROOF OF COROLLARY 1

The corollary is simply a restatement of Theorem 2 in terms of \( q_t = \hat{p}_{t-1} \). In this case, the KL divergence with respect to the optimal solution is \( D_{KL}(q^\ast || p_x) = 0 \), since we are considering a class of non-parametric distributions which can recover arbitrary distributions via MCMC. By Theorem 2, we then know that:

\[
\min_{t \leq T} r_t \leq \frac{1}{T} R_T \in \mathcal{O}(\gamma_{ST}T^{-1/2}).
\]

(66)
If \( \gamma_{ST} \in \mathcal{O}(T^{\alpha}) \), for some \( \alpha < 1/2 \), so that \( \alpha - 1/2 < 1 \), then:

\[
\frac{1}{T} R_T \in \mathcal{O}(T^{\alpha-1/2}) \implies \lim_{T \to \infty} \min_{t \leq T} r_t \leq \lim_{T \to \infty} \frac{1}{T} R_T = 0 .
\]

Replacing \( r_t = \mathbb{D}_{KL}(\hat{p}_t \| p_\infty) \) above concludes the proof.

**Alternative proof.** Another way of proving the same bound on the KL divergence is the following. The KL divergence from \( \hat{p}_t \) to \( p_\infty \) is bounded via Lemma 1. Namely, with probability greater than \( 1 - \delta \), we have that:

\[
\mathbb{D}_{KL}(\hat{p}_t \| p_\infty) = \mathbb{E}_{\theta \sim \hat{p}_t} [ \log \hat{p}_t(\theta) - \log p_\infty(\theta) ] \\
= \mathbb{E}_{\theta \sim \hat{p}_t} [ \log p(\theta) - \mu_t(\theta) + \beta_t \sigma_t(\theta) - \log p_\infty(\theta) ] \\
= \mathbb{E}_{\theta \sim \hat{p}_t} [ \mu_t(\theta) + \beta_t \sigma_t(\theta) - \ell(\theta) ] + \mathbb{E}_{\theta \sim \hat{p}_t} [ \log p(x) - \log \zeta_t ] \\
\leq 2\beta_t \mathbb{E}_{\theta \sim \hat{p}_t} [ \sigma_t(\theta) ] + \log p(x) - \log \zeta_t .
\]

For the second part of the last term in the right-hand side above, we also have that:

\[
p(x) = \int_\Theta p(\theta) \exp \ell(\theta) \, d\theta \leq \int_\Theta p(\theta) \exp u_t(\theta) \, d\theta = \zeta_t ,
\]

which holds with the same probability as Equation 68. Therefore, we conclude the proof with:

\[
\log p(x) - \log \zeta_t \leq 0 \\
\implies \mathbb{D}_{KL}(\hat{p}_t \| p_\infty) \leq 2\beta_t \mathbb{E}_{\theta \sim \hat{p}_t} [ \sigma_t(\theta) ] .
\]

\[\square\]

### C KL DIVERGENCE BOUNDS FOR KL-UCB WITH SPECIFIC KERNELS

Corollary 1 connects the regret bound in Theorem 2 with the KL divergence of posterior approximations by KL-UCB when sampling directly from the posterior surrogate induced by UCB. In this case, we can bound the KL divergence of the posterior approximations by KL-UCB with respect to the posterior as:

\[
\min_{t \leq T} \mathbb{D}_{KL}(\hat{p}_t \| p_\infty) \leq \frac{R_T}{T}, \quad T \geq 1 .
\]

According to Theorem 2, we have \( R_T \in \mathcal{O}(\gamma_{ST} \sqrt{T}) \). Therefore, to bound the KL divergence, we need kernel-specific upper bounds for the maximum information gain \( \gamma_T \) (Srinivas et al., 2010; Vakili et al., 2021). In particular, we consider the case of two popular stationary kernel classes. The first one is the squared-exponential kernel, used in our experiments. As previously mentioned, this kernel yields a bound \( \gamma_T \in \mathcal{O}\left(\log^{d+1}(T)\right) \) (Srinivas et al., 2010).

The second type of kernel is the Matérn class with smoothness parameter \( \nu > 1/2 \):

\[
k(\theta, \theta') := \frac{1}{\Gamma(\nu) 2^{\nu-1}} \left( \frac{\sqrt{2 \nu \rho_\theta \rho_{\theta'}}}{l} \right) \nu B_\nu \left( \frac{\sqrt{2 \nu \rho_\theta \rho_{\theta'}}}{l} \right),
\]

where \( \rho_\theta \rho_{\theta'} := \| \theta - \theta' \|_2, \ l > 0 \) is a length-scale parameter controlling the smoothness of the functions in the RKHS, \( \Gamma \) is the gamma function, and \( B_\nu \) is the modified Bessel function of the second kind, for \( \theta, \theta' \in \Theta \). This kernel leads to a maximum information gain \( \gamma_T \in \mathcal{O}\left(T^{\nu/\nu-1} \log^{2d+2}(T)\right) \) according to recent results (Vakili et al., 2021).

Table 1 presents upper bounds for the KL divergence of posterior approximations by KL-UCB. As the table shows, both kernels lead to an asymptotically vanishing KL divergence in the approximations with respect to the true posterior in general. An exception to asymptotic convergence, however, is that convergence does not necessarily hold for \( d \geq 2\nu \) in the case of the Matérn kernel, noticing the exponent in its rate. For instance, a Matérn kernel with \( \nu = 3/2 \) would not guarantee convergence in a problem whose parameter space has dimension \( d \geq 3 \). The guarantees for the squared-exponential kernel do not suffer from this drawback, though the rates are possibly worse due to the exponential dependence on \( d \) via the logarithmic term.

In terms of approximation bounds with respect to the number of likelihood evaluations \( N = ST \), we have a KL divergence bound of \( \mathcal{O}\left(N^{-1/2}\right) \) and \( \mathcal{O}\left(N^{\nu/\nu-1} (\frac{2}{\nu})\right) \) for the squared-exponential and the Matérn kernels, respectively. Here the \( \mathcal{O} \)-notation suppresses logarithmic factors. Compared to the exponential convergence rates in Kanagawa and Hennig (2019), notice that their results are for a noise-free setting, while we consider settings with (sub-Gaussian) noise. In the noise-free setting, one is usually able to obtain tighter concentration bounds for the GP approximation (see de Freitas et al., 2012).

### D DIMENSIONALITY EFFECT

In this section, we present a short analysis on the effect of dimensionality on the regret of the KL-UCB algorithm.

---

**Table 1: KL divergence bounds for KL-UCB**

<table>
<thead>
<tr>
<th>Kernel class</th>
<th>( \min_{t \leq T} \mathbb{D}_{KL}(\hat{p}<em>t | p</em>\infty) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Squared exponential</td>
<td>( \mathcal{O}\left(T^{-1/2} \log^{d+1}(ST)\right) )</td>
</tr>
<tr>
<td>Matérn ( \nu &gt; 1/2 )</td>
<td>( \mathcal{O}\left(T^{\nu/\nu-1} (\frac{2}{\nu}) \log^{2d+2}(ST)\right) )</td>
</tr>
</tbody>
</table>
Figure 5 presents the KL-UCB regret for the problem in Section 7.1 when we increase the dimensionality of the parameter space $\Theta = \mathbb{R}^d$. As the plot shows, the regret has an exponential dependence on the dimensionality of the parameter space. Therefore, the practitioner might need to run the method for longer to obtain reasonable posterior approximations or apply dimensionality reduction methods.

Figure 5: Dimensionality effect on the mean regret of KL-UCB on the RKHS log-likelihood problem after 20 iterations as a function of the dimensionality of the parameter space. The results of 5 independent runs were combined to produce the box plot.

E DETAILS OF THE EXPERIMENTAL SETUP

In this section, we present further details on the experimental setup for the empirical results in the paper. In particular, we describe the settings for KL-UCB. For VBMC, we used an implementation provided by its author.\(^7\)

KL-UCB setup: For the GP model in KL-UCB, we used GPyTorch (Gardner et al., 2018) with adaptations to perform fast rank-1 Cholesky updates on the GP covariance matrix (see Rasmussen and Williams, 2006, Algorithm 2.1). Given the theoretical nature of this work, we did not perform online hyper-parameters learning, contrary to what is usual in other GP-based approximate inference methods (Gutmann and Corander, 2016; Acerbi, 2018). The kernel lengthscale was set as 0.5, a value which provided fitting GP estimates for the generated problem scenarios. The GP was configured as zero mean $m := 0$ for the RKHS-based problem in Section 7.1, while we used the log-prior probability as the mean function $m := \log p$ for the problems in Section 7.2. The latter allows the GP to provide low likelihood estimates for parameters of low prior probability, avoiding excessive exploration of the parameter space. In terms of noise settings, observations in the RKHS problem (Section 7.1) were added with Gaussian noise $\sigma := 10^{-6}$, since this algorithm was not originally designed to handle noise (see Acerbi, 2020, for a recent noise-adapted version). The GP noise parameter was correspondingly set as $\eta := 10^{-2}$ for the RKHS problem, and $\eta := 10^{-4}$ for the noise-free problems. Having a small, non-zero $\eta$ avoids numerical issues with matrix inversions. Lastly, for the setting of $\beta$, the RKHS norm for the log-likelihood function in Section 7.1 is available in closed form, but for the non-RKHS functions we set $b := 3$ which yields the common 3 standard deviations UCB parameter (also in VBMC (Acerbi, 2018)) in a noiseless setting.

MCMC setup: We configured the EMCEE sampler (Foreman-Mackey et al., 2013) with 25 walkers and a burn-in of 400 samples. We selected the $S := 5$ evaluation points in each BO iteration’s batch out of 500 samples drawn by EMCEE. Sub-sampling from a larger batch of samples reduces the correlation between samples from the chain.

Estimation of KL divergence: To verify theoretical bounds, we measured the KL divergence between KL-UCB’s posterior approximations and the true posterior distribution, which is unknown to KL-UCB. As the posterior distribution approximations from KL-UCB are sampled-based MCMC estimates, we had to estimate the KL divergence based on samples. We also took into account that MCMC samples are usually correlated. To decorrelate the samples, we built a kernel density estimator out of the MCMC samples using a Gaussian kernel with the rule by Scott (1992) for kernel bandwidth selection\(^8\) and then sampled from this continuous density. For a large enough number of MCMC samples (2000 for the likelihood-free inference problem, 10000 for the circular likelihood, and 1000 for the other problems), the i.i.d. samples from the density estimator approximately follow the stationary distribution of the MCMC chain, i.e. its target posterior. We then applied a k-nearest-neighbours method (Szabó, 2014)\(^9\) to obtain the KL divergence estimates.

---

\(^7\)VBMC experiments code: https://github.com/lacerbi/infbench

\(^8\)Implementation available with the KDEpy package: https://github.com/tommyod/KDEpy

\(^9\)We used the Python version of the ITE toolbox: https://bitbucket.org/szzoli/ite-in-python/