Bayesian Analysis (2020) **TBA**, Number TBA, pp. 1–32

# Parallel Gaussian Process Surrogate Bayesian Inference with Noisy Likelihood Evaluations

Marko Järvenpää[*], Michael U. Gutmann[†], Aki Vehtari[*] and Pekka Marttinen[*]

**Abstract.** We consider Bayesian inference when only a limited number of noisy log-likelihood evaluations can be obtained. This occurs for example when complex simulator-based statistical models are fitted to data, and synthetic likelihood (SL) method is used to form the noisy log-likelihood estimates using computationally costly forward simulations. We frame the inference task as a sequential Bayesian experimental design problem, where the log-likelihood function is modelled with a hierarchical Gaussian process (GP) surrogate model, which is used to efficiently select additional log-likelihood evaluation locations. Motivated by recent progress in the related problem of batch Bayesian optimisation, we develop various batch-sequential design strategies which allow to run some of the potentially costly simulations in parallel. We analyse the properties of the resulting method theoretically and empirically. Experiments with several toy problems and simulation models suggest that our method is robust, highly parallelisable, and sample-efficient.

**Keywords:** expensive likelihoods, likelihood-free inference, surrogate modelling, Gaussian processes, sequential experiment design, parallel computing.

## 1 Introduction

When the analytic form of the likelihood function of a statistical model is available, standard sampling techniques such as Markov Chain Monte Carlo (MCMC, see e.g. Robert and Casella 2004) can often be used for Bayesian inference. However, many models of interest in several areas of science, for example in computational biology and ecology, have an expensive-to-evaluate or intractable likelihood function which severely complicates inference. When the likelihood is intractable but forward simulation of the model is feasible, simulation-based inference methods (also called likelihood-free inference) such as approximate Bayesian computation (ABC) can be used. Unfortunately, such algorithms typically require a huge number of simulations making inference computationally costly. Examples of models with intractable likelihoods can be found in e.g. Beaumont et al. (2002); Marin et al. (2012); Lintusaari et al. (2017); Marttinen et al. (2015); Järvenpää et al. (2018) and Section 6.2 of this article.

Surrogate models, also called meta-models or emulators, such as Gaussian processes (Rasmussen and Williams, 2006) have been used extensively to calibrate deterministic computer codes, see e.g. Kennedy and O'Hagan (2001). GP surrogates have recently also

[*] Helsinki Institute for Information Technology HIIT, Department of Computer Science, Aalto University, marko.j.jarvenpaa@aalto.fi, aki.vehtari@aalto.fi, pekka.marttinen@aalto.fi

[†] School of Informatics, University of Edinburgh, michael.gutmann@ed.ac.uk

https://doi.org/10.1214/20-BA1200

been used to accelerate Bayesian inference by modelling some part of the inferential process, such as the log-likelihood function. The model allows extracting information from the simulations efficiently, and can be used e.g. to determine where additional simulations are needed. For example, Rasmussen (2003); Kandasamy et al. (2015); Sinsbeck and Nowak (2017); Drovandi et al. (2018); Wang and Li (2018); Acerbi (2018) have developed GP-based techniques to accelerate Bayesian inference when the exact likelihood or the corresponding deterministic model is tractable but expensive. Various GP surrogate techniques have been proposed also for ABC, where one can only draw samples i.e. pseudo-data from a statistical model but not evaluate the likelihood. These include Meeds and Welling (2014); Jabot et al. (2014); Wilkinson (2014); Gutmann and Corander (2016); Järvenpää et al. (2019).

In this paper we focus on GP surrogate modelling of noisy log-likelihood evaluations. Earlier works on emulating the log-likelihood function have mostly assumed exact, i.e., noiseless evaluations or the noise has not been explicitly modelled. We show that noisy evaluations cause extra challenges. Also, although not the focus of this work, we remark that one often has some control over the noise level. While our approach is applicable whenever noisy, expensive log-likelihood evaluations of a statistical model of interest are available, we mainly focus on likelihood-free inference using the synthetic likelihood method (Wood, 2010; Price et al., 2018), where the intractable log-likelihood is approximated using repeated forward simulations at each evaluation location.

Recently, Järvenpää et al. (2019) developed a Bayesian decision theoretic framework for ABC inference and considered sequential strategies (also called active learning) to select the next evaluation location for an expensive simulation model. Here we extend this framework in two ways: 1) we modify it to address the problem of Bayesian inference using noisy log-likelihood evaluations, which is different from ABC, and 2) we develop batch-sequential design strategies to efficiently parallelise the estimation of the surrogate likelihood. In earlier related works the simulation locations have been selected either sequentially (Kandasamy et al., 2015; Sinsbeck and Nowak, 2017; Wang and Li, 2018; Acerbi, 2018; Järvenpää et al., 2019) or using simple heuristics (Wilkinson, 2014; Gutmann and Corander, 2016). Batch strategies are useful when a computing cluster is available and, as we show, can substantially reduce the computation time compared to the corresponding sequential strategies. We also analyse some properties of the proposed methods theoretically, and conduct an extensive empirical comparison.

Our approach is closely related to Bayesian quadrature (BQ), see e.g. O'Hagan (1991); Hennig et al. (2015); Karvonen et al. (2018). In particular, BQ methods have been used by Osborne et al. (2012); Gunter et al. (2014); Chai and Garnett (2019) to compute the marginal likelihood and to quantify the numerical error of this integral probabilistically. In this article we are not interested in this particular integral but in obtaining an accurate estimate of the posterior. Also, we allow noisy log-likelihood evaluations. Another related problem is Bayesian optimisation (BO), see e.g. Brochu et al. (2010); Shahriari et al. (2015). Our objective to parallelise simulations is motivated by recent research on batch Bayesian optimisation (Ginsbourger et al., 2010; Azimi et al., 2010; Snoek et al., 2012; Contal et al., 2013; Desautels et al., 2014; Shah and Ghahramani, 2015; Wu and Frazier, 2016; Gonzalez et al., 2016; Wilson et al., 2018).

However, while BO methods can be used to accelerate likelihood-free Bayesian inference (Gutmann and Corander, 2016), they are not specifically designed for estimating the posterior (see discussion in e.g. Kandasamy et al. 2015; Järvenpää et al. 2019). Similarly to Järvenpää et al. (2019), we explicitly design our algorithms from the first principles of Bayesian decision theory to acknowledge the goal of the analysis, i.e. estimation of the posterior density. Finally, we note that GPs in conjunction with Bayesian experimental designs have also been successful in estimating level and excursion sets of expensive-to-evaluate functions, see e.g. Bect et al. (2012); Chevalier et al. (2014); Lyu et al. (2018).

This paper is organised as follows. Section 2 briefly reviews ABC and the SL. Sections 3 and 4 contain the details of our GP surrogate model and posterior estimation. Batch-sequential design strategies for sample-efficient estimation of the (approximate) posterior distribution are developed in Section 5 while Section 6 contains experiments. Finally, Section 7 contains discussion and concluding remarks. Proofs, implementation details and additional experiments can be found in the supplementary material (Järvenpää et al. 2020).

## 2 ABC and the synthetic likelihood methods

Our goal is to estimate parameters $\boldsymbol{\theta} \in \Theta$ of a simulation model given observed data $\mathbf{x} \in \mathcal{X}$. We assume $\Theta$ is a compact subset of $\mathbb{R}^d$ and that the prior information about feasible values of $\boldsymbol{\theta}$ is coded into a (continuous) prior pdf $\pi(\boldsymbol{\theta})$. For simplicity we consider only continuous parameters but discrete parameters can be handled similarly. If evaluating the likelihood function $\pi(\mathbf{x} \,|\, \boldsymbol{\theta})$ is feasible, the posterior distribution can be computed using Bayes' theorem $\pi(\boldsymbol{\theta} \,|\, \mathbf{x}) \propto \pi(\boldsymbol{\theta})\pi(\mathbf{x} \,|\, \boldsymbol{\theta})$ up to a normalisation constant and hence be used as a target distribution in MCMC. However, when the likelihood is too costly to evaluate or unavailable, standard MCMC algorithms become infeasible.

Even when the likelihood is intractable, simulating "pseudo-data" from the model, i.e., drawing samples $\mathbf{x}_{\boldsymbol{\theta}} \sim \pi(\cdot \,|\, \boldsymbol{\theta})$, is often feasible. In this case, ABC can be used for inference, see e.g. Marin et al. (2012); Turner and Van Zandt (2012); Lintusaari et al. (2017). Standard ABC techniques approximate the posterior as

$$\pi_{\mathrm{ABC}}(\boldsymbol{\theta} \,|\, \mathbf{x}) \propto \pi(\boldsymbol{\theta}) \int_{\mathcal{X}} \mathbb{1}_{\Delta(\mathbf{x},\mathbf{x}_s) \leq \varepsilon} \pi(\mathbf{x}_s \,|\, \boldsymbol{\theta}) \, \mathrm{d}\mathbf{x}_s, \tag{1}$$

where $\mathbb{1}$ denotes the indicator function, $\varepsilon$ is a tolerance parameter and $\Delta : \mathcal{X}^2 \to \mathbb{R}_+$ is the discrepancy function used to compute the similarity between the simulated data $\mathbf{x}_s$ and the observed data $\mathbf{x}$. The discrepancy is typically constructed from low-dimensional summary statistics $S : \mathcal{X} \to \mathbb{R}^p$, so that $\Delta(\mathbf{x}, \mathbf{x}_s) = \Delta'(S(\mathbf{x}), S(\mathbf{x}_s))$, where $\Delta' : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}_+$ is, for example, the weighted Euclidean distance. For each proposed parameter $\boldsymbol{\theta}$, an unbiased ABC posterior estimate can be obtained by replacing the integral in (1) with a Monte Carlo sum using some $N$ simulated pseudo-data sets $\mathbf{x}_{\boldsymbol{\theta}}^{(i)} \sim \pi(\cdot \,|\, \boldsymbol{\theta})$ for $i = 1, \ldots, N$.

An alternative to ABC is the synthetic likelihood method (Wood, 2010; Price et al., 2018). In SL the summary statistics $S(\mathbf{x}_{\boldsymbol{\theta}})$ are assumed to have a Gaussian distribution for each parameter $\boldsymbol{\theta}$, that is

$$\pi(\mathbf{x} \,|\, \boldsymbol{\theta}) \approx \pi(S(\mathbf{x}) \,|\, \boldsymbol{\theta}) \approx \pi_{\mathrm{SL}}(S(\mathbf{x}) \,|\, \boldsymbol{\theta}) \triangleq \mathcal{N}(S(\mathbf{x}) \,|\, \boldsymbol{\mu}_{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}}). \tag{2}$$

The first approximation results from replacing the full data $\mathbf{x}$ with a potentially non-sufficient summary statistics $S(\mathbf{x})$. The second approximation is due to the possible violations of the Gaussianity of $S(\mathbf{x})$. The expectation $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ in (2) are unknown and are estimated for each proposed parameter $\boldsymbol{\theta}$ using maximum likelihood (ML):

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{\theta}} = \frac{1}{N}\sum_{i=1}^{N} S(\mathbf{x}_{\boldsymbol{\theta}}^{(i)}), \quad \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}} = \frac{1}{N-1}\sum_{i=1}^{N}(S(\mathbf{x}_{\boldsymbol{\theta}}^{(i)}) - \hat{\boldsymbol{\mu}}_{\boldsymbol{\theta}})(S(\mathbf{x}_{\boldsymbol{\theta}}^{(i)}) - \hat{\boldsymbol{\mu}}_{\boldsymbol{\theta}})^{\top}, \tag{3}$$

where $\mathbf{x}_{\boldsymbol{\theta}}^{(i)} \sim \pi(\cdot \,|\, \boldsymbol{\theta})$ for $i = 1, \ldots, N$. As investigated by Price et al. (2018), the standard Metropolis algorithm can be combined with SL. The likelihood is then computed using (2) and the ML estimates in (3) or, alternatively, using an unbiased estimate of $\mathcal{N}(S(\mathbf{x}) \,|\, \boldsymbol{\mu}_{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}})$ shown in Section 2.1 of Price et al. (2018) which produces an exact pseudo-marginal MCMC if the Gaussianity assumption holds. See supplementary material C for discussion on the use of different SL estimators.

The advantage of SL over ABC is that specifying suitable ABC tuning parameters such as the tolerance and the discrepancy is avoided. While the Gaussianity of the summary statistics may not hold in practice, Price et al. (2018) have found that SL is often robust to deviations from normality. SL and its extensions (Thomas et al., 2018; An et al., 2019b,a; Frazier et al., 2019) produce pointwise noisy log-likelihood evaluations because in practice the number of repeated simulations $N$ at each point is finite. Using (pseudo-marginal) MCMC or other sampling-based techniques for inference with these noisy targets thus requires a large number of simulations. Assuming noisy log-likelihood evaluations are available, e.g. obtained by using SL, the goal of the following sections is to develop an inference algorithm that can minimise the number of evaluations needed.

## 3 Gaussian process surrogate for the noisy log-likelihood

We denote the log-likelihood or its approximation, such as the log-SL obtained as the logarithm of (2), as $f(\boldsymbol{\theta}) \triangleq \log \pi(\mathbf{x} \,|\, \boldsymbol{\theta})$. We assume that we have access to noisy log-likelihood evaluations at $\boldsymbol{\theta}_i$ denoted by $y_i \in \mathbb{R}$ for building the surrogate model and that the "noise" i.e. the numerical or sampling error in evaluating the log-likelihood is independently Gaussian distributed. Treating the noisy log-likelihood evaluations $y_i$ as "observations", our measurement model is

$$y_i = f(\boldsymbol{\theta}_i) + \sigma_n(\boldsymbol{\theta}_i)\varepsilon_i, \quad \varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1), \tag{4}$$

where $\sigma_n : \Theta \to (0, \infty)$ is a (continuous) function of $\boldsymbol{\theta}$ that determines the standard deviation of the observation noise and is assumed known. To justify our model in (4), we show empirically that log-SL is well approximated by a Gaussian distribution using six benchmark simulation models in the supplementary material D.1.

We place the following hierarchical GP prior for the log-likelihood function $f$:

$$f \,|\, \boldsymbol{\gamma} \sim \mathcal{GP}(m_0(\boldsymbol{\theta}), k(\boldsymbol{\theta}, \boldsymbol{\theta}')), \quad m_0(\boldsymbol{\theta}) = \sum_{i=1}^{q} \gamma_i h_i(\boldsymbol{\theta}), \quad \boldsymbol{\gamma} \sim \mathcal{N}(\mathbf{b}, \mathbf{B}), \tag{5}$$

where $k : \Theta^2 \to \mathbb{R}$ is a covariance function and $h_i : \Theta \to \mathbb{R}$ are fixed basis functions (both assumed continuous). The nuisance parameters $\boldsymbol{\gamma}$ in (5) are marginalised, see e.g. O'Hagan and Kingman (1978); Rasmussen and Williams (2006), to obtain the following equivalent GP prior

$$f \sim \mathcal{GP}(\mathbf{h}(\boldsymbol{\theta})^\top \mathbf{b}, k(\boldsymbol{\theta}, \boldsymbol{\theta}') + \mathbf{h}(\boldsymbol{\theta})^\top \mathbf{B} \mathbf{h}(\boldsymbol{\theta}')), \tag{6}$$

where $\mathbf{h}(\boldsymbol{\theta}) \in \mathbb{R}^q$ is a column vector consisting of the basis functions $h_i$ evaluated at $\boldsymbol{\theta}$. We use basis functions of the form $1, \theta_i, \theta_i^2$. A similar GP prior has been considered in Wilkinson (2014); Gutmann and Corander (2016); Drovandi et al. (2018), however, different from those articles, we take a fully Bayesian approach and marginalise $\boldsymbol{\gamma}$ as in Riihimäki and Vehtari (2014). Since little initial information is typically available on the magnitude and shape of the log-likelihood, we use relatively uninformative hyperpriors so that $\mathbf{b} = \mathbf{0}$ and $B_{ij} = 30^2 \mathbb{1}_{i=j}$. We assume that the log-likelihood function is smooth, and adopt the squared exponential covariance function $k(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sigma_f^2 \exp(-\sum_{i=1}^{d} (\theta_i - \theta_i')^2/(2l_i^2))$ although other choices, such as the Matérn covariance function, are also possible. We denote the $d+1$ covariance function hyperparameters as $\boldsymbol{\phi} = (\sigma_f^2, l_1, \ldots, l_d)$. For now, we assume $\boldsymbol{\phi}$ is known and omit it from our notation for simplicity.

Given observations $D_t = \{(y_i, \boldsymbol{\theta}_i)\}_{i=1}^t$, which we call training data, our knowledge of the log-likelihood function is $f \,|\, D_t \sim \mathcal{GP}(m_t(\boldsymbol{\theta}), c_t(\boldsymbol{\theta}, \boldsymbol{\theta}'))$, where

$$m_t(\boldsymbol{\theta}) \triangleq k_t(\boldsymbol{\theta}) \mathbf{K}_t^{-1} \mathbf{y}_t + \mathbf{R}_t^\top(\boldsymbol{\theta}) \bar{\gamma}_t, \tag{7}$$

$$c_t(\boldsymbol{\theta}, \boldsymbol{\theta}') \triangleq k(\boldsymbol{\theta}, \boldsymbol{\theta}') - k_t(\boldsymbol{\theta}) \mathbf{K}_t^{-1} k_t^\top(\boldsymbol{\theta}') + \mathbf{R}_t^\top(\boldsymbol{\theta}) [\mathbf{B}^{-1} + \mathbf{H}_t \mathbf{K}_t^{-1} \mathbf{H}_t^\top]^{-1} \mathbf{R}_t(\boldsymbol{\theta}'), \tag{8}$$

with $[\mathbf{K}_t]_{ij} \triangleq k(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j) + \mathbb{1}_{i=j} \sigma_n^2(\boldsymbol{\theta}_i)$ for $i, j \in \{1, \ldots, t\}$, $k_t(\boldsymbol{\theta}) \triangleq (k(\boldsymbol{\theta}, \boldsymbol{\theta}_1), \ldots, k(\boldsymbol{\theta}, \boldsymbol{\theta}_t))^\top$,

$$\bar{\gamma}_t \triangleq [\mathbf{B}^{-1} + \mathbf{H}_t \mathbf{K}_t^{-1} \mathbf{H}_t^\top]^{-1} (\mathbf{H}_t \mathbf{K}_t^{-1} \mathbf{y}_t + \mathbf{B}^{-1} \mathbf{b}), \tag{9}$$

and $\mathbf{R}_t(\boldsymbol{\theta}) \triangleq \mathbf{H}(\boldsymbol{\theta}) - \mathbf{H}_t \mathbf{K}_t^{-1} k_t^\top(\boldsymbol{\theta})$. Above $\mathbf{H}_t$ is the $q \times t$ matrix whose columns consist of basis function values evaluated at training points $\boldsymbol{\theta}_{1:t} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_t]$ which is itself a $d \times t$ matrix, and $\mathbf{H}(\boldsymbol{\theta})$ is the corresponding $q \times 1$ vector at test point $\boldsymbol{\theta}$. From now on, we denote the GP variance function as $s_t^2(\boldsymbol{\theta}) \triangleq c_t(\boldsymbol{\theta}, \boldsymbol{\theta})$ and the probability law of $f$ given $D_t$ as $\Pi_{D_t}^f$, that is, $\Pi_{D_t}^f \triangleq \mathcal{GP}(m_t(\boldsymbol{\theta}), c_t(\boldsymbol{\theta}, \boldsymbol{\theta}'))$.

## 4 Estimators of the posterior from the GP surrogate

Using the GP surrogate model for the noisy log-likelihood, we here derive estimators for the posterior which can be e.g. plugged-in to a MCMC algorithm. Resulting sampling algorithms do not require further simulator runs (unlike e.g. SL-MCMC) producing potentially huge computational savings. Figure 1 demonstrates our approach. We want to use our knowledge of the log-likelihood function represented by $\Pi^f_{D_t}$ to determine the optimal point estimate of the probability density function (pdf) of the posterior.[1] The uncertainty of the log-likelihood $f$ can be propagated to the posterior distribution of the simulation model which consequently becomes a random quantity denoted as $\pi_f$:

$$\pi_f(\boldsymbol{\theta}) \triangleq \frac{\pi(\boldsymbol{\theta})\exp(f(\boldsymbol{\theta}))}{\int_\Theta \pi(\boldsymbol{\theta}')\exp(f(\boldsymbol{\theta}'))\,\mathrm{d}\boldsymbol{\theta}'}. \tag{10}$$

The expectation of the posterior pdf $\pi_f$ at each parameter $\boldsymbol{\theta}$ can be formally written as

$$\mathbb{E}_{f\,|\,D_t}(\pi_f(\boldsymbol{\theta})) = \int \frac{\pi(\boldsymbol{\theta})\exp(f(\boldsymbol{\theta}))}{\int_\Theta \pi(\boldsymbol{\theta}')\exp(f(\boldsymbol{\theta}'))\,\mathrm{d}\boldsymbol{\theta}'}\Pi^f_{D_t}(\mathrm{d}f) \tag{11}$$

and the variance can be obtained similarly (assuming these quantities exist). In principle, one could sample posterior pdfs by first drawing $f^{(i)} \sim \Pi^f_{D_t}$ (a continuous function $\Theta \to \mathbb{R}$), then computing $\pi(\boldsymbol{\theta})\exp(f^{(i)}(\boldsymbol{\theta}))$, and finally normalising. However, in practice this would require discretisation of the $\Theta$-space and involves computational challenges. For this reason and similarly to Sinsbeck and Nowak (2017); Järvenpää et al. (2019), we instead take our quantity of interest to be the unnormalised posterior

$$\tilde{\pi}_f(\boldsymbol{\theta}) \triangleq \pi(\boldsymbol{\theta})\exp(f(\boldsymbol{\theta})), \tag{12}$$

which follows log-Gaussian process that allows analytical computations.

Next we derive an optimal estimator for the unnormalised posterior $\tilde{\pi}$ in (12) using Bayesian decision-theory. We proceed here similarly to Sinsbeck and Nowak (2017) and consider the integrated quadratic loss function $l_2(\tilde{\pi}_1, \tilde{\pi}_2) \triangleq \int_\Theta (\tilde{\pi}_1(\boldsymbol{\theta}) - \tilde{\pi}_2(\boldsymbol{\theta}))^2\,\mathrm{d}\boldsymbol{\theta}$ between two (unnormalised) posterior densities $\tilde{\pi}_1$ and $\tilde{\pi}_2$. We assume $\tilde{\pi}_1$ and $\tilde{\pi}_2$ are square-integrable functions in $\Theta$, i.e. $\tilde{\pi}_1, \tilde{\pi}_2 \in L^2(\Theta)$. The optimal Bayes estimator, denoted by $\hat{\tilde{\pi}} \in \mathbb{D}$, is the minimiser of the expected loss, where $\mathbb{D} = L^2(\Theta)$ denotes the set of candidate estimators. In detail,

$$\begin{aligned}\hat{\tilde{\pi}} &= \arg\min_{\tilde{d}\in\mathbb{D}} \mathbb{E}_{f\,|\,D_t} l_2(\tilde{\pi}_f, \tilde{d}) = \arg\min_{\tilde{d}\in\mathbb{D}} \mathbb{E}_{f\,|\,D_t} \int_\Theta (\tilde{\pi}_f(\boldsymbol{\theta}) - \tilde{d}(\boldsymbol{\theta}))^2\,\mathrm{d}\boldsymbol{\theta} \\ &= \arg\min_{\tilde{d}\in\mathbb{D}} \int_\Theta \mathbb{E}_{f\,|\,D_t}(\tilde{\pi}_f(\boldsymbol{\theta}) - \tilde{d}(\boldsymbol{\theta}))^2\,\mathrm{d}\boldsymbol{\theta},\end{aligned} \tag{13}$$

where Tonelli theorem is used to change the order of expectation and integration and where $\tilde{d} \in \mathbb{D} = L^2(\Theta)$ is a candidate estimator of $\tilde{\pi}$. Equation 13 shows that the expected loss is minimised when the integrand on the second row is minimised independently for (almost) each $\boldsymbol{\theta} \in \Theta$. It follows from the basic results of Bayesian decision theory (see e.g. Robert 2007) that the minimum is obtained when $\tilde{d}(\boldsymbol{\theta}) = \mathbb{E}_{f\,|\,D_t}(\tilde{\pi}_f(\boldsymbol{\theta}))$, i.e., the optimal estimator is the posterior expectation. The minimum value of (13), called

[1] While in this article we are mainly concerned with point estimators of the posterior pdf, we can also quantify its (epistemic) uncertainty similarly to *probabilistic numerics* literature (see e.g. Hennig et al. 2015; Cockayne et al. 2019; Briol et al. 2019) as illustrated in Figure 1b. Such uncertainty estimates are also used to intelligently select the next simulation locations in Section 5.

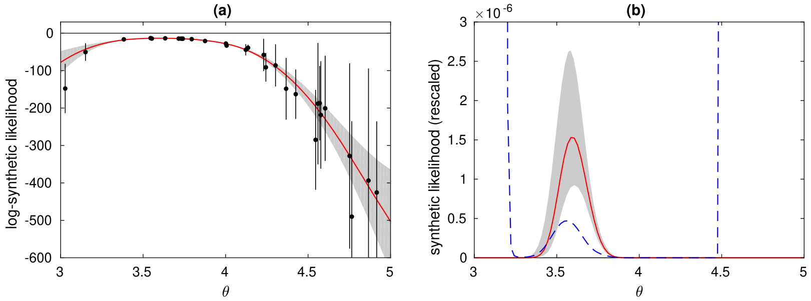

Figure 1: (a) GP surrogate model for the log-SL of the Ricker model of Section 6.2 when only the first parameter, $\theta = \log(r)$, is varied. The black dots show the noisy log-SL evaluations and the black lines their approximate 95% confidence intervals, the grey area the 95% credible interval, and the red line the GP mean function. (b) Uncertainty of the SL. The grey area shows the 95% credible interval of the SL and the red line is the median estimate, obtained from (15). The dashed blue line shows the standard deviation of SL computed as the square root of (14).

Bayes risk, is the integrated variance $\int_\Theta \mathbb{V}_{f\,|\,D_t}(\tilde{\pi}_f(\boldsymbol{\theta}))\,\mathrm{d}\boldsymbol{\theta}$. The posterior expectation and variance can be computed from the log-Normal distribution as

$$\mathbb{E}_{f|D_t}(\tilde{\pi}_f(\boldsymbol{\theta}))=\pi(\boldsymbol{\theta})e^{m_t(\boldsymbol{\theta})+\frac{1}{2}s_t^2(\boldsymbol{\theta})}, \quad \mathbb{V}_{f|D_t}(\tilde{\pi}_f(\boldsymbol{\theta}))=\pi^2(\boldsymbol{\theta})e^{2m_t(\boldsymbol{\theta})+s_t^2(\boldsymbol{\theta})}\left(e^{s_t^2(\boldsymbol{\theta})}-1\right). \quad (14)$$

If we instead use $L^1$ loss $l_1(\tilde{\pi}_1,\tilde{\pi}_2) \triangleq \int_\Theta |\tilde{\pi}_1(\boldsymbol{\theta}) - \tilde{\pi}_2(\boldsymbol{\theta})|\,\mathrm{d}\boldsymbol{\theta}$ where $\tilde{\pi}_1, \tilde{\pi}_1 \in L^1(\Theta)$, we can similarly show that the optimal point estimator is the marginal median. The median and the $\alpha$-quantile $q^\alpha$ with $\alpha \in (0,1)$ can be computed as

$$\mathrm{med}_{f\,|\,D_t}(\tilde{\pi}_f(\boldsymbol{\theta})) = \pi(\boldsymbol{\theta})e^{m_t(\boldsymbol{\theta})}, \quad q^\alpha_{f\,|\,D_t}(\tilde{\pi}_f(\boldsymbol{\theta})) = \pi(\boldsymbol{\theta})e^{m_t(\boldsymbol{\theta})+\Phi^{-1}(\alpha)s_t(\boldsymbol{\theta})}, \quad (15)$$

where $\Phi^{-1}$ is the quantile function of the standard normal distribution. As we show in the supplementary material A, the Bayes risk corresponding to the $L^1$ loss is

$$\min_{\tilde{d}\in\mathbb{D}} \mathbb{E}_{f\,|\,D_t} l_1(\tilde{\pi}_f, \tilde{d}) = \int_\Theta \pi(\boldsymbol{\theta}) \exp(m_t(\boldsymbol{\theta}) + s_t^2(\boldsymbol{\theta})/2)(2\Phi(s_t(\boldsymbol{\theta})) - 1)\,\mathrm{d}\boldsymbol{\theta}. \quad (16)$$

When MCMC is used with the point estimator of the unnormalised posterior in either (14) or (15), we are in fact targeting the following mean and median based estimators of the (normalised) posterior

$$\pi_t^{\mathrm{mean}}(\boldsymbol{\theta}) \triangleq \frac{\pi(\boldsymbol{\theta})e^{m_t(\boldsymbol{\theta})+\frac{1}{2}s_t^2(\boldsymbol{\theta})}}{\int_\Theta \pi(\boldsymbol{\theta}')e^{m_t(\boldsymbol{\theta}')+\frac{1}{2}s_t^2(\boldsymbol{\theta}')}\,\mathrm{d}\boldsymbol{\theta}'}, \quad \pi_t^{\mathrm{med}}(\boldsymbol{\theta}) \triangleq \frac{\pi(\boldsymbol{\theta})e^{m_t(\boldsymbol{\theta})}}{\int_\Theta \pi(\boldsymbol{\theta}')e^{m_t(\boldsymbol{\theta}')}\,\mathrm{d}\boldsymbol{\theta}'}. \quad (17)$$

These are obtained by simply normalising the Bayes optimal estimators of the unnormalised posterior (and, as a consequence, a guarantee of optimality for normalised posterior does not follow). Similar estimators were also considered by Stuart and Teckentrup (2018). Both are clearly valid density functions, and tractable, unlike (11). The latter, i.e., the marginal median based estimate, is equal to (10) if we replace the unknown log-likelihood function $f(\boldsymbol{\theta})$ with a GP mean function $m_t(\boldsymbol{\theta})$ and neglect GP uncertainty. On the other hand, the former, i.e., the marginal mean estimate, takes into account the GP uncertainty through the variance function $s_t^2(\boldsymbol{\theta})$. These two point estimates become the same if the GP variance is negligible.

# 5 Parallel designs of simulations

In the previous section we showed how to quantify the uncertainty of the GP surrogate-based unnormalised posterior density and we derived computable and (in a certain sense) optimal point estimates of it. Next we develop Bayesian experimental design strategies to select further locations to evaluate the log-likelihood, so that the uncertainty in the unnormalised posterior decreases as fast as possible. We focus on batch strategies and denote the batch size as $b \in \{1, 2, \ldots\}$. Before moving on, we introduce some terminology. The next batch of $b$ evaluation locations is obtained as the solution to an optimisation problem. We call the objective function of this optimisation problem a *design criterion* and the resulting batch of evaluation locations as *design points* or just *design*. The complete procedure of selecting the design points is called a *batch-sequential* (or, when $b = 1$, just *sequential*) *strategy*.[2]

In this paper we focus on synchronous parallelisation where a batch of $b$ design points is constructed at each iteration and the corresponding $b$ simulations are simultaneously submitted to the workers. However, the "greedy" design strategies developed in Sections 5.3 and 5.5 can also be used for asynchronous parallelisation, where a new location is immediately chosen and submitted for processing, whenever any of the running simulations completes, instead of waiting all the other $b - 1$ simulations to finish.

## 5.1 Analytical expressions for the design criteria

We first derive some general results needed for efficient evaluation of the design criteria. These can be useful also for developing batch designs for other related GP-based problems such as BQ and BO. Given $D_t = \{(y_i, \boldsymbol{\theta}_i)\}_{i=1}^t$ it is useful to know how additional $b$ candidate evaluations at points $\boldsymbol{\theta}^* = [\boldsymbol{\theta}_1^*, \ldots, \boldsymbol{\theta}_b^*]$ would affect our knowledge about the log-likelihood $f$ and the unnormalised posterior $\tilde{\pi}$. The following Lemma is central to our analysis. It shows how the GP mean and variance functions are affected by supplementing the training data $D_t$ with a new batch of evaluations $D^* = \{(y_i^*, \boldsymbol{\theta}_i^*)\}_{i=1}^b$ when the unknown $\mathbf{y}^*$ is assumed to be distributed according to the posterior predictive distribution of the GP given $D_t$. The Lemma is a generalisation of a similar result by Järvenpää et al. (2019); Lyu et al. (2018).

**Lemma 5.1.** *Consider the mean and variance functions of the GP model in Section 3 for a fixed $\boldsymbol{\theta}$, given the training data $D_t \cup D^*$ and when treated as functions*

[2] The design criterion is often called an acquisition function and the resulting strategy sometimes an acquisition rule in the BO literature.

*of* $\mathbf{y}^*$*. Assume* $\mathbf{y}^*$ *follows the posterior predictive distribution, that is* $\mathbf{y}^* \,|\, \boldsymbol{\theta}^*, D_t \sim \mathcal{N}(m_t(\boldsymbol{\theta}^*), c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}^*) + \mathrm{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*), \ldots, \sigma_n^2(\boldsymbol{\theta}_b^*)))$*. Then,*

$$m_{t+b}(\boldsymbol{\theta}; \boldsymbol{\theta}^*) \,|\, \boldsymbol{\theta}^*, D_t \sim \mathcal{N}(m_t(\boldsymbol{\theta}), \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)), \tag{18}$$

$$s_{t+b}^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) \,|\, \boldsymbol{\theta}^*, D_t \sim \delta(s_t^2(\boldsymbol{\theta}) - \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) - s_{t+b}^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)), \tag{19}$$

*where* $\delta(\cdot)$ *is the Dirac measure and*

$$\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) = c_t(\boldsymbol{\theta}, \boldsymbol{\theta}^*)[c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}^*) + \mathrm{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*), \ldots, \sigma_n^2(\boldsymbol{\theta}_b^*))]^{-1} c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}). \tag{20}$$

In the Lemma, $m_{t+b}(\boldsymbol{\theta}; \boldsymbol{\theta}^*)$ is the GP mean function at iteration $t+b$ whose dependence on $\boldsymbol{\theta}^*$ is shown explicitly. Importantly, the above Lemma shows how the GP variance decreases from $s_t^2(\boldsymbol{\theta})$ to $s_{t+b}^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)$ when the extra $b$ evaluations at $\boldsymbol{\theta}^*$ are included, and the reduction $\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)$ is deterministic. We see, for example, that if $c_t(\boldsymbol{\theta}_i^*, \boldsymbol{\theta}_j^*) = 0$ for all $i, j = 1, \ldots, b, i \neq j$ which might hold approximately, e.g., if the evaluation points $\boldsymbol{\theta}_i^*$ are located far from each other, then

$$\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) = \sum_{i=1}^{b} \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_i^*) = \sum_{i=1}^{b} \frac{c_t^2(\boldsymbol{\theta}, \boldsymbol{\theta}_i^*)}{s_t^2(\boldsymbol{\theta}_i^*) + \sigma_n^2(\boldsymbol{\theta}_i^*)}. \tag{21}$$

This shows that the reduction of GP variance at $\boldsymbol{\theta}$, $\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)$, factorises over the new evaluation points $\boldsymbol{\theta}_i^*$ in $\boldsymbol{\theta}^*$. Intuitively, if the test point $\boldsymbol{\theta}$ is strongly correlated with some evaluation point $\boldsymbol{\theta}_i^*$, including the evaluation at $\boldsymbol{\theta}_i^*$ will result in a large reduction of variance at the test point. Furthermore, the larger the noise variance $\sigma_n^2(\boldsymbol{\theta}_i^*)$ at the evaluation point $\boldsymbol{\theta}_i^*$ is, the less the GP variance will decrease.

It clearly holds that $0 \leq \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) \leq s_t^2(\boldsymbol{\theta})$. Items (i–ii) of the following Lemma summarise some additional properties of the variance reduction function in (20) and (iii–iv) show two further useful identities needed later. Item (i) shows the (rather obvious) result that the order of evaluation points in $\boldsymbol{\theta}^*$ does not change $\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)$ and in the following we often identify the $d \times b$ matrix $\boldsymbol{\theta}^*$ with a multiset whose elements are the columns of $\boldsymbol{\theta}^*$ although this leads to some abuse of notation.

**Lemma 5.2.** *Let* $\boldsymbol{\theta}^* \in \mathbb{R}^{d \times b}$ *and let* $\boldsymbol{\theta} \in \mathbb{R}^d$ *be any test point. The function* $\boldsymbol{\theta}^* \mapsto \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)$ *in* (20) *for any fixed* $\boldsymbol{\theta}$ *has the following properties.*

*(i) The function value is invariant to the permutation of the evaluation locations i.e. the columns of* $\boldsymbol{\theta}^*$*.*

*(ii) Let* $\boldsymbol{\theta}_A^* \subseteq \boldsymbol{\theta}_B^*$*. Then* $\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_A^*) \leq \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_B^*)$*, i.e., including new evaluations never increases variance.*

*(iii) Let* $\boldsymbol{\theta}^* = [\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*]$ *so that* $b = 2$ *and denote the predictive variance at* $\boldsymbol{\theta}_j^*$ *by* $\bar{s}_t^2(\boldsymbol{\theta}_j^*) \triangleq s_t^2(\boldsymbol{\theta}_j^*) + \sigma_n^2(\boldsymbol{\theta}_j^*)$ *for* $j \in \{1, 2\}$*. Then*

$$\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) = \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_1^*) + \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_2^*) + r_t(\boldsymbol{\theta}; \boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*), \tag{22}$$

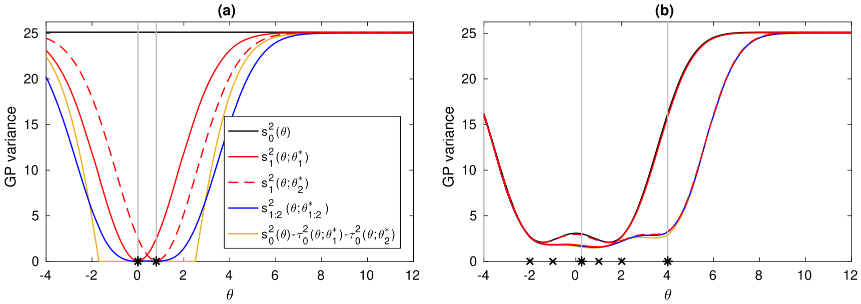

Figure 2: The effect of two new evaluations (black stars) on the GP variance. Black line is the original variance, red lines (solid and dashed) show variance if only one of the evaluations is included. Blue line shows the variance after both evaluations, and yellow the variance if the interaction between the locations is neglected. (a) Noiseless observations at evaluation locations close to each other are obtained. (b) similar to (a), but showing four earlier evaluations (at black crosses) from which noisy observations were available, such that the GP variance is not exactly zero at these locations.

$$
\begin{aligned}
r_t(\boldsymbol{\theta};\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*) \triangleq\ & \frac{c_t^2(\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta},\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*) + c_t^2(\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta},\boldsymbol{\theta}_2^*)\bar{s}_t^2(\boldsymbol{\theta}_1^*)}{\bar{s}_t^4(\boldsymbol{\theta}_1^*)\bar{s}_t^4(\boldsymbol{\theta}_2^*) - \bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*)} \\
& - 2\frac{c_t(\boldsymbol{\theta},\boldsymbol{\theta}_1^*)c_t(\boldsymbol{\theta},\boldsymbol{\theta}_2^*)c_t(\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*)}{\bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*) - c_t^2(\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*)}.
\end{aligned}
\tag{23}
$$

*(iv) Let* $\boldsymbol{\theta}^* = [\boldsymbol{\theta}_A^*, \boldsymbol{\theta}_b^*]$ *where* $\boldsymbol{\theta}_A^* \in \mathbb{R}^{d\times(b-1)}$ *and* $\boldsymbol{\theta}_b^* \in \mathbb{R}^d$*, and denote* $\bar{\mathbf{S}}_A = c_t(\boldsymbol{\theta}_A^*) + \mathrm{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*),\ldots,\sigma_n^2(\boldsymbol{\theta}_{b-1}^*))$*. Then*

$$
\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*) = \tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}_A^*) + \frac{(c_t(\boldsymbol{\theta},\boldsymbol{\theta}_b^*) - c_t(\boldsymbol{\theta},\boldsymbol{\theta}_A^*)\bar{\mathbf{S}}_A^{-1}c_t(\boldsymbol{\theta}_A^*,\boldsymbol{\theta}_b^*))^2}{s_t^2(\boldsymbol{\theta}_b^*) + \sigma_n^2(\boldsymbol{\theta}_b^*) - c_t(\boldsymbol{\theta}_b^*,\boldsymbol{\theta}_A^*)\bar{\mathbf{S}}_A^{-1}c_t(\boldsymbol{\theta}_A^*,\boldsymbol{\theta}_b^*)}. \tag{24}
$$

Figure 2 demonstrates how two new evaluations at $\theta_1^*$ and $\theta_2^*$ reduce the GP variance in two different one-dimensional examples. Specifically, Figure 2a illustrates the fact of Lemma 5.2 (iii) that the interaction between the evaluation points, represented by the term $r_t(\theta;\theta_1^*,\theta_2^*)$, affects the reduction of the GP variance and its effect can be either positive or negative (the variance after two new evaluations, blue line, is either below or above the yellow line, which represents the reduced variance if the interaction is neglected). In Figure 2b the new evaluation locations are far apart and the factorisation of the variance reduction in (21) holds approximately.

## 5.2 Batch-sequential designs

Given $D_t = \{(y_i,\boldsymbol{\theta}_i)\}_{i=1}^t$, our goal is to select the next batch of $b$ evaluations $\boldsymbol{\theta}^*$ in an optimal fashion. We take a Bayesian decision theoretic approach, where $\boldsymbol{\theta}^*$ is selected to minimise the expected (or median) loss, where the loss measures uncertainty remaining in the unnormalised posterior $\tilde{\pi}^f$ when the hypothetical observations $\mathbf{y}^*$ at locations $\boldsymbol{\theta}^*$ are taken into account. In the following we develop two such techniques based on two different measures of uncertainty: variance and interquartile range (IQR). Design strategies which acknowledge the impact of the next batch, but neglect the whole remaining computational budget, are often called "myopic". It is possible to formulate a non-myopic design as a dynamic programming problem, but this is computationally demanding, see e.g. Bect et al. (2012); González et al. (2016). Consequently, we focus on myopic designs which already produce highly sample-efficient and practical algorithms.

### Expected integrated variance (EIV)

As our first measure of the uncertainty of the unnormalised posterior $\tilde{\pi}^f$ for the selection of the next batch design $\boldsymbol{\theta}^*$, we select the Bayes risk under the $L^2$ loss. In this case, the Bayes risk is the integrated variance function

$$\mathcal{L}^{\mathrm{v}}(\Pi^f_{D_t}) \triangleq \int_{\Theta} \mathbb{V}_{f\,|\,D_t}(\tilde{\pi}_f(\boldsymbol{\theta}))\,\mathrm{d}\boldsymbol{\theta} = \int_{\Theta} \pi^2(\boldsymbol{\theta}) e^{2m_t(\boldsymbol{\theta})+s_t^2(\boldsymbol{\theta})}\left(e^{s_t^2(\boldsymbol{\theta})}-1\right)\mathrm{d}\boldsymbol{\theta}, \tag{25}$$

whose integrand was obtained from (14). This is similar to Sinsbeck and Nowak (2017); Järvenpää et al. (2019) who, however, considered other GP surrogate models and only sequential designs. We compute the expectation over the hypothetical noisy log-likelihoods $\mathbf{y}^*$ for any candidate design $\boldsymbol{\theta}^*$, leading to the expected integrated variance design criterion (EIV). The resulting optimal strategy is a special case of stepwise uncertainty reduction technique, see e.g. Bect et al. (2012). This criterion is evaluated efficiently without numerical simulations from the GP model, using the following result.

**Proposition 5.3.** *With the assumptions of Lemma 5.1, the expected integrated variance design criterion $L_t^{\mathrm{v}}$ at any candidate design $\boldsymbol{\theta}^* \in \Theta^b$ is*

$$L_t^{\mathrm{v}}(\boldsymbol{\theta}^*) \triangleq \mathbb{E}_{\mathbf{y}^*|\boldsymbol{\theta}^*,D_t}\mathcal{L}^{\mathrm{v}}(\Pi^f_{D_t\cup D^*}) = \int_{\Theta} \pi^2(\boldsymbol{\theta}) e^{2m_t(\boldsymbol{\theta})+s_t^2(\boldsymbol{\theta})}\left(e^{s_t^2(\boldsymbol{\theta})} - e^{\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)}\right)\mathrm{d}\boldsymbol{\theta}. \tag{26}$$

### Integrated median interquartile range (IMIQR)

A sequential design strategy based on EIV worked well in the ABC scenario of Järvenpää et al. (2019), who modelled the discrepancy in (1) with a GP. In this article we instead model the log-likelihood with a GP as illustrated in Figure 1 and the goal is to minimise the uncertainty of the unnormalised posterior $\tilde{\pi}^f$, which has a log-Normal distribution for a fixed $\boldsymbol{\theta}$. However, the expectation and variance can be suboptimal estimates of the central tendency and uncertainty of a heavy-tailed distribution such as log-Normal. For example, Figure 1b shows that the standard deviation (dashed blue line) grows very rapidly at the boundaries although at the same time the credible interval clearly indicates that the probability of the log-likelihood, and consequently the likelihood, of having a non-negligible value there is vanishingly small. The mean is similarly affected in a non-intuitive way by the heavy tails: It is fairly easy to see that

$$\mathbb{P}[\exp(f(\boldsymbol{\theta})) \geq \mathbb{E}(\exp(f(\boldsymbol{\theta})))] = \mathbb{P}(f(\boldsymbol{\theta}) \geq m_t(\boldsymbol{\theta}) + s_t^2(\boldsymbol{\theta})/2) = \Phi(-s_t(\boldsymbol{\theta})/2). \tag{27}$$

This means that with a sufficiently large variance of the log-likelihood $s_t^2(\boldsymbol{\theta})$, the probability that the likelihood $\exp(f(\boldsymbol{\theta}))$ is greater than its own mean becomes negligible.

The above analysis suggests (and empirical results in Section 6 further confirm) that mean-based point estimates and variance-based design strategies, such as the EIV and those proposed by Gunter et al. (2014); Kandasamy et al. (2015); Sinsbeck and Nowak (2017); Järvenpää et al. (2019); Acerbi (2018), are unsuitable when log-likelihood evaluations are noisy. A reasonable alternative for the $L^2$-loss used to derive the EIV is to measure the uncertainty in the posterior using the $L^1$-loss, which is less affected by extreme values. As shown in Section 3, the $L^1$-loss leads to the marginal median estimate for the posterior, $\pi_t^{\mathrm{med}}$. While the $L^1$-loss produces a robust median estimator that we adopt, (16) shows that the Bayes risk with $L^1$ loss scales as $\exp(s_t^2(\boldsymbol{\theta})/2)$ since $\Phi(s_t(\boldsymbol{\theta})) \approx 1$ for large $s_t(\boldsymbol{\theta})$, such that also this measure for overall uncertainty of $\tilde{\pi}^f$ is affected by the heavy tails of the log-Normal distribution.

We propose a new, robust design criterion for selecting the next design. In place of the variance in EIV, we use a robust measure of uncertainty, the interquartile range $\mathrm{IQR}(\boldsymbol{\theta}) = q^{3/4}(\boldsymbol{\theta}) - q^{1/4}(\boldsymbol{\theta})$. The integrated IQR loss measuring the uncertainty of the posterior pdf is defined as

$$\mathcal{L}^{\mathrm{IQR}}(\Pi_{D_t}^f) \triangleq \int_\Theta \mathrm{IQR}_{f \,|\, D_t}(\tilde{\pi}_f(\boldsymbol{\theta}))\,\mathrm{d}\boldsymbol{\theta} = 2\int_\Theta \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} \sinh(u s_t(\boldsymbol{\theta}))\,\mathrm{d}\boldsymbol{\theta}, \tag{28}$$

where $u \triangleq \Phi^{-1}(p_u)$ and $\sinh(z) = (\exp(z) - \exp(-z))/2$ for $z \in \mathbb{R}$ is the hyperbolic sine, which emerges after using (15). While we use $p_u = 0.75$, other quantiles $p_u \in (0.5, 1)$ are also possible. A theoretical downside of the IQR loss is that it does not formally coincide with the Bayes risk for the $L^1$ or $L^2$ loss, which correspond to the optimal point estimators of the unnormalised posterior (see Section 4).

We also use the median in place of the mean to measure the effect of the next design $\boldsymbol{\theta}^*$ to the loss function. That is, we use median loss decision theory (see Yu and Clarke 2011), and define the median integrated IQR loss function as

$$L_t^{\mathrm{IQR}}(\boldsymbol{\theta}^*) \triangleq \mathrm{med}_{\mathbf{y}^* \,|\, \boldsymbol{\theta}^*, D_t} \mathcal{L}^{\mathrm{IQR}}(\Pi_{D_t \cup D^*}^f). \tag{29}$$

The median integrated IQR loss in (29) is intractable but it can be approximated by the integrated median IQR loss (IMIQR). This approximation[3] follows by replacing the predictive distribution of $\mathbf{y}^*$ with a point mass, i.e., $\pi(\mathbf{y}^* \,|\, \boldsymbol{\theta}^*, D_t) \approx \delta(m_t(\boldsymbol{\theta}^*) - \mathbf{y}^*)$. This approximation resembles the so-called kriging believer heuristic in Ginsbourger et al. (2010). The next result gives a useful formula to calculate IMIQR.

[3] Instead of approximation, which may be inaccurate when the GP variance function is large, the integrated median criterion in (30) can be seen as an alternative decision-theoretic formulation with infinitely many dependent variables of interest (one for each $\boldsymbol{\theta} \in \Theta$) and where the median outcomes after considering the effect of the design $\boldsymbol{\theta}^*$ are all computed separately for each $\boldsymbol{\theta} \in \Theta$, and the corresponding losses are combined through averaging. The median integrated loss in (29) instead has a single combined loss function.

**Proposition 5.4.** *With the assumptions of Lemma 5.1, the integrated median* IQR *loss, denoted as* $\tilde{L}_t^{\mathrm{IQR}}$*, at any candidate design* $\boldsymbol{\theta}^* \in \Theta^b$ *is*

$$\tilde{L}_t^{\mathrm{IQR}}(\boldsymbol{\theta}^*) \triangleq \int_{\Theta} \mathrm{med}_{\mathbf{y}^*|\boldsymbol{\theta}^*,D_t} \mathrm{IQR}_{f|D_t \cup D^*}(\tilde{\pi}_f(\boldsymbol{\theta}))\mathrm{d}\boldsymbol{\theta} = 2\int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} \sinh(u s_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*))\mathrm{d}\boldsymbol{\theta}. \tag{30}$$

The integrand of (30) is recognised as a product of the marginal median estimate of the posterior in (15) and the function $\sinh(us_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*))$. Hence, to minimise IMIQR, the simulation locations $\boldsymbol{\theta}^*$ need to be chosen as a compromise between regions where the current posterior estimate is non-negligible and where the GP variance $s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)$ decreases efficiently when the simulations are run at $\boldsymbol{\theta}^*$. Similar interpretation holds also for the EIV function in (26). However, EIV assigns significantly more weight to areas with high GP variance than IMIQR.

## 5.3 Joint and greedy optimisation for batch-sequential designs

We can now evaluate EIV and IMIQR design criteria for any candidate design $\boldsymbol{\theta}^*$ and choose $\boldsymbol{\theta}^*$ as the minimiser, i.e.,

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\Theta^b} L_t(\boldsymbol{\theta}), \tag{31}$$

where $L_t$ is either the EIV in (26) or IMIQR in (30). The objective function is typically smooth but multimodal so global optimisation is needed. We call (31) as "joint" optimisation which does not scale to high dimensional parameter spaces or to large batch sizes. Even if computing the design criterion is cheap as compared to the run times of typical simulation models, solving the $db$-dimensional global optimisation problem is often impractical as discussed in Wilson et al. (2018). Hence, we consider greedy optimisation as also used in batch BO (Ginsbourger et al., 2010; Snoek et al., 2012; Wilson et al., 2018). The greedy optimisation procedure for both EIV and IMIQR works as follows: the first point $\boldsymbol{\theta}_1^*$ is chosen as in the sequential case i.e. by solving (31) with $b = 1$. The rest of the points $\boldsymbol{\theta}_{2:b}^*$ are obtained by iteratively solving

$$\boldsymbol{\theta}_r^* = \arg\min_{\boldsymbol{\theta}\in\Theta} L_t([\boldsymbol{\theta}_{1:r-1}^*, \boldsymbol{\theta}]), \quad r = 2, 3, \ldots, b. \tag{32}$$

This greedy approach simplifies the difficult $db$-dimensional optimisation into $b$ separate $d$-dimensional problems, and makes it scalable as a function of $b$.

In general, the design found by the greedy optimisation does not equal the minimiser of the joint criterion. It follows from Lemma 5.2 (i) that both EIV and IMIQR are invariant to the order of evaluation locations in $\boldsymbol{\theta}^*$ but this does not hold for the greedy procedure. Bounds for the performance of greedy maximisation of a set function have been studied in literature, see e.g. Nemhauser et al. (1978); Krause et al. (2008); Bach (2013). For example, if the design criterion (when defined equivalently using a utility so that (32) becomes a maximisation problem) is submodular and non-decreasing in batch size $b$, then the worst-case outcome of greedy optimisation is at least $1 - 1/e \approx 0.63$

of the corresponding optimal joint value. A utility function corresponding to IMIQR defined below is not submodular but an approximation of it is weakly submodular (see e.g. Krause et al. 2008; Krause and Cevher 2010). We use this fact to derive a weaker but still useful bound.

We here consider an approximation $\tilde{L}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}^*)$ of $\tilde{L}_t^{\mathrm{IQR}}(\boldsymbol{\theta}^*)$ so that

$$\tilde{L}_t^{\mathrm{IQR}}(\boldsymbol{\theta}^*) \approx \tilde{L}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}^*) \triangleq 2u^2 \int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} s_{t+b}^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*)\,\mathrm{d}\boldsymbol{\theta}, \tag{33}$$

which follows from the observation that $\sinh(us_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*)) \approx u^2 s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)$ and where we had $u = \Phi^{-1}(p_u)$. The approximation in (33) is reasonable when $s_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*) \in [0, 3/u]$ in the region where $\pi(\boldsymbol{\theta})e^{m_t(\boldsymbol{\theta})}$ is non-negligible. For simplicity, we consider a discretised setting where the optimisation is done over a finite set $\tilde{\Theta} \subset \Theta$ and define (approximate) IMIQR utility function as

$$\tilde{U}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}^*) \triangleq \tilde{L}_t^{\mathrm{IQR},a}(\emptyset) - \tilde{L}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}^*) \tag{34}$$

for $\boldsymbol{\theta}^* \in 2^{\tilde{\Theta}}$. Clearly, maximising $\tilde{U}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}^*)$ is equivalent to minimising $\tilde{L}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}^*)$. The following theorem gives a bound for the greedy optimisation of the (approximate) IMIQR utility function. The proof can be found in supplementary material B.2.

**Theorem 5.5.** *Consider the set function $\tilde{U}_t^{\mathrm{IQR},a} : 2^{\tilde{\Theta}} \to \mathbb{R}_+$ in (34). Let $\boldsymbol{\theta}_O$ be a (joint) optimal solution for maximising $\tilde{U}_t^{\mathrm{IQR},a}(\boldsymbol{\theta})$ over $\boldsymbol{\theta} \subset \tilde{\Theta}, |\boldsymbol{\theta}| \leq b$. The greedy algorithm for this maximisation problem outputs a set $\boldsymbol{\theta}_G \subset \tilde{\Theta}$ satisfying*

$$\tilde{U}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}_G) \geq (1 - 1/e)\tilde{U}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}_O) - b^2\varepsilon_t, \quad \textit{where} \tag{35}$$

$$\varepsilon_t \triangleq \max\{0, 2u^2\varepsilon_t'\}, \quad \varepsilon_t' \triangleq \max_{\substack{\boldsymbol{\theta}_A \subset \tilde{\Theta}, |\boldsymbol{\theta}_A| = i \leq 2b \\ \boldsymbol{\theta}_j, \boldsymbol{\theta}_k \in \tilde{\Theta}}} \int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} r_{1:t+i}(\boldsymbol{\theta}; \boldsymbol{\theta}_j, \boldsymbol{\theta}_k)\,\mathrm{d}\boldsymbol{\theta}. \tag{36}$$

Computing $\varepsilon_t$ explicitly is difficult but we expect that often $\varepsilon_t \ll \tilde{U}_t^{\mathrm{IQR},a}(\boldsymbol{\theta}_O)$ since $r_{1:t+i}(\boldsymbol{\theta};\boldsymbol{\theta}_j,\boldsymbol{\theta}_k)$ given by (23) tends to be small. However, $\varepsilon_t$ may not always be small, the term $b^2\varepsilon_t$ scales quadratically for batch size $b$, and the bound holds only approximately for IMIQR. This bound still suggests that, at least in some iterations of the algorithm, greedy IMIQR produces near-optimal batch locations. On the other hand, an approximation similar to (33) for EIV would be reasonable in a very limited number of situations, and experiments in supplementary material D.3 suggest that greedy EIV scales worse as a function of $b$ compared to the corresponding greedy IMIQR strategy. Finally, we note that even when the bound is weak, new design points cannot increase the value of EIV or IMIQR loss function as shown in supplementary material B.1. Hence, the batch strategies cannot be worse than the corresponding sequential designs and, in practice, they are highly useful as is seen empirically in Section 6.

## 5.4 Implementation details

Using the GP surrogate model and the analysis from the previous sections, we show the resulting inference method as Algorithm 1. Some key implementation details are discussed below and further details (e.g. on handling GP hyperparameters, MCMC methods used, and optimisation of the design criteria) are given in supplementary material C. The algorithm is shown for the SL case using the IMIQR strategy, but it works similarly for EIV, heuristic designs developed in the next section and other log-likelihood estimators besides SL. The potentially expensive simulations on the lines 2–5 and 18–21 can be done in parallel. In the SL case, the simulations can be parallelised in terms of both the number of repeated simulations $N$ and batch size $b$.

**Algorithm 1** GP-based SL inference using IMIQR with synchronous batch design.

```
Input: Prior density $\pi(\boldsymbol{\theta})$, simulation model $\pi(\cdot \,|\, \boldsymbol{\theta})$, GP prior $\Pi^f$, number of repeated
    samples $N$, summary function $S$, batch size $b$, initial batch size $b_0$, max. iterations
    $i_{\max}$, number of IS samples $s_{\text{IS}}$, number of MCMC samples $s_{\text{MC}}$
 1: Sample $\boldsymbol{\theta}_{1:b_0} \overset{\text{i.i.d.}}{\sim} \pi(\cdot)$ (or use some other space-filling initial design)
 2: for $r = 1 : b_0$ do                                                                      } in parallel
 3:     Simulate $\mathbf{x}_r^{(1:N)} \overset{\text{i.i.d.}}{\sim} \pi(\cdot \,|\, \boldsymbol{\theta}_r)$ and compute $S_r^{(1:N)} = S(\mathbf{x}_r^{(1:N)})$
 4:     Compute $y_r$ from $\{S_r^{(j)}\}_{j=1}^N$
 5: end for
 6: Set initial training data $D_{b_0} \leftarrow \{(y_r, \boldsymbol{\theta}_r)\}_{r=1}^{b_0}$
 7: for $i = 1 : i_{\max}$ do
 8:     Use MAP estimation to obtain GP hyperparameters $\boldsymbol{\phi}$ using $D_{b_0+(i-1)b}$
 9:     Sample $\boldsymbol{\theta}^{(j)} \sim \pi_q$ using MCMC and compute $\omega^{(j)}$ in Eq. 37 for $j = 1, \ldots, s_{\text{IS}}$
10:     if joint_optim then
11:         Obtain $\boldsymbol{\theta}_{1:b}^{(i)*}$ by solving Eq. 31 using Eq. 30 and 37
12:     else if greedy_optim then
13:         Obtain $\boldsymbol{\theta}_1^{(i)*}$ by minimising Eq. 31 using Eq. 30 and 37
14:         for $r = 2 : b$ do
15:             Obtain $\boldsymbol{\theta}_r^{(i)*}$ by solving Eq. 32 using Eq. 30 and 37
16:         end for
17:     end if
18:     for $r = 1 : b$ do                                                                     } in parallel
19:         Simulate $\mathbf{x}_r^{(i,1:N)} \overset{\text{i.i.d.}}{\sim} \pi(\cdot \,|\, \boldsymbol{\theta}_r^{(i)*})$, compute $S_r^{(i,1:N)} = S(\mathbf{x}_r^{(i,1:N)})$
20:         Compute $y_r^{(i)*}$ using $\{S_r^{(i,j)}\}_{j=1}^N$
21:     end for
22:     Update training data $D_{b_0+ib} \leftarrow D_{b_0+(i-1)b} \cup \{(y_r^{(i)*}, \boldsymbol{\theta}_r^{(i)*})\}_{r=1}^b$
23: end for
24: Use MAP estimation to obtain GP hyperparameters $\boldsymbol{\phi}$ using $D_{b_0+i_{\max}b}$
25: Sample $\boldsymbol{\vartheta}^{(1:s_{\text{MC}})}$ from the marginal median estimate in Eq. 17 using MCMC
26: return Samples $\boldsymbol{\vartheta}^{(1:s_{\text{MC}})}$ from the approximate SL posterior
```

Evaluation of EIV and IMIQR requires numerical integration over $\Theta \subset \mathbb{R}^d$. Similar computational challenges emerge also in the state-of-the-art BO methods such as Hennig and Schuler (2012); Hernández-Lobato et al. (2014); Wu and Frazier (2016) and in Chevalier et al. (2014). If $d \leq 2$ we discretise the parameter space $\Theta$ and approximate the integral in the resulting grid. In higher dimensions, we use self-normalised importance sampling (IS) as in Chevalier et al. (2014); Järvenpää et al. (2019). Specifically, we draw samples from the importance distribution $\boldsymbol{\theta}^{(j)} \sim \pi_q(\boldsymbol{\theta})$ and use these as integration points to approximate

$$\int_{\Theta} I_t(\boldsymbol{\theta};\boldsymbol{\theta}^*)\,\mathrm{d}\boldsymbol{\theta} \approx \sum_{j=1}^{s_{\mathrm{IS}}} \omega^{(j)} I_t(\boldsymbol{\theta}^{(j)};\boldsymbol{\theta}^*), \quad \omega^{(j)} = \frac{1/\pi_q(\boldsymbol{\theta}^{(j)})}{\sum_{k=1}^{s} 1/\pi_q(\boldsymbol{\theta}^{(k)})}, \tag{37}$$

where the integrand of either (26) or 30 is denoted by $I_t(\boldsymbol{\theta};\boldsymbol{\theta}^*)$. As the proposal $\pi_q$ we use the current loss (see (14) and the integrand of (16)), which is a function $\Theta \to \mathbb{R}_+$, and can be interpreted as an unnormalised pdf. This is a natural choice because the current loss has a similar shape as the expected/median loss as a function of $\boldsymbol{\theta}$. We use the same proposal in the greedy optimisation in (32) although it would be also possible to adapt the proposal $\pi_q$ according to the pending points $\boldsymbol{\theta}^*_{1:r-1}$ when optimising with respect to the $r$th point $\boldsymbol{\theta}^*_r$.

We have assumed that the noise function $\sigma_n^2$ in (4) is known. In practice, this is a valid assumption only in the noiseless case where $\sigma_n^2(\boldsymbol{\theta}) = 0$. As our focus is on the noisy setting, we need to estimate $\sigma_n^2$. Sometimes $\sigma_n^2$ can be assumed to be an unknown constant to be determined together with the GP hyperparameters $\boldsymbol{\phi}$ using MAP estimation. However, we observed that $\sigma_n^2$ often depends on the magnitude of the log-likelihood (see Figure 1), making the assumption of homoscedastic noise questionable. Similarly to Wilkinson (2014), we estimate $\sigma_n^2$ using the bootstrap. Specifically, with each new training data point $\boldsymbol{\theta}_i$, we resample with replacement $N$ summary vectors from the original population $\{S_i^{(j)}\}_{j=1}^N$ for 2000 times. We then compute the empirical variance of the resulting log-SL values and use it as a plug-in estimator for $\sigma_n^2(\boldsymbol{\theta}_i)$.

For EIV, IMIQR and greedy batch versions of MAXV and MAXIQR (see Section 5.5), $\sigma_n^2$ needs to be also known at candidate design points $\boldsymbol{\theta}^*$. Bootstrap cannot be used because the simulated summaries are only available for training data. We take a pragmatic approach and set $\sigma_n = 10^{-2}$ at the candidate design points as if the future evaluations were almost exact. This simplification effectively reduces the occurrence of (potentially redundant) simulations at nearby points to encourage exploration. Alternatively, one could use another GP to model the bootstrapped variances or their logarithms and use the GP mean function as a point estimate for the function $\sigma_n^2$ as in Ankenman et al. (2010).

## 5.5 Alternative heuristic designs strategies

Here we present some heuristic alternative design strategies. These are empirically compared to the more principled EIV and IMIQR strategies in Section 6. We first focus on sequential designs where $b = 1$.

**MAXIQR**: A natural and simple approach is to evaluate where the current variance, IQR or some other suitable (local) measure of uncertainty is maximised. Such strategies are in some contexts called “uncertainty sampling”. The advantage over EIV and IMIQR is cheaper computation because the effect of the candidate design point to the whole posterior is not acknowledged. Using IQR produces the design strategy

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} \sinh(u s_t(\boldsymbol{\theta})), \tag{38}$$

which we abbreviate as MAXIQR because it evaluates at the maximiser of IQR. Taking the logarithm of (38), the MAXIQR strategy can be equivalently written as

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Theta} \left(\log \pi(\boldsymbol{\theta}) + m_t(\boldsymbol{\theta}) + u s_t(\boldsymbol{\theta}) + \log(1 - e^{-2u s_t(\boldsymbol{\theta})})\right), \tag{39}$$

which shows a tradeoff between evaluating where the log-posterior is presumed to be large (the first two terms in (39)) and unexplored regions where the GP variance is large (the last two terms). This formula also shows an interesting connection to the upper confidence bound (UCB) criterion commonly used in BO, see e.g. Srinivas et al. (2010); Shahriari et al. (2015). The UCB acquisition function is $\text{UCB}(\boldsymbol{\theta}) = m_t(\boldsymbol{\theta}) + \beta_t s_t(\boldsymbol{\theta})$, where $\beta_t$ is a tradeoff parameter, here automatically chosen to be $\beta_t = \Phi^{-1}(p_u)$. Compared to the standard UCB, there is, however, an extra term in (39) which further penalises regions having small variance $s_t^2$. If the variance $s_t^2$ is large everywhere and/or if $p_u$ is an extreme quantile, then the last term in (39) is approximately zero and the MAXIQR design criterion approximately equals UCB.

**MAXV**: When the variance is used instead of IQR, we obtain a strategy

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Theta} \pi^2(\boldsymbol{\theta}) e^{2m_t(\boldsymbol{\theta}) + s_t^2(\boldsymbol{\theta})} \left(e^{s_t^2(\boldsymbol{\theta})} - 1\right). \tag{40}$$

This strategy is abbreviated as MAXV which, in fact, is used by Gunter et al. (2014); Kandasamy et al. (2015) in the noiseless case, and it is called "exponentiated variance" by Kandasamy et al. (2015). Taking logarithm of (40) shows that this design also features a tradeoff between large posterior and large variance, similarly to MAXIQR.

Since these two strategies are not derived from Bayesian decision theory, it is not immediately clear how one should parallelise these inherently sequential strategies. However, it seems reasonable to use the fact the $s_t^2(\boldsymbol{\theta})$ is always reduced near the pending evaluation locations. Motivated by this and related BO techniques in Ginsbourger et al. (2010); Snoek et al. (2012); Desautels et al. (2014), we compute the median value of the design criterion with respect to the posterior predictive distribution of the pending simulations. The next locations are chosen iteratively such that, for MAXIQR, the first point $\boldsymbol{\theta}_1^*$ in the batch is chosen using (38) and the rest $\boldsymbol{\theta}_{2:b}$ by iteratively solving

$$\boldsymbol{\theta}_r^* = \arg\max_{\boldsymbol{\theta}\in\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} \sinh(u s_{t+r-1}(\boldsymbol{\theta}; \boldsymbol{\theta}_{1:r-1}^*)), \quad r = 2, 3, \ldots, b. \tag{41}$$

MAXV is parallelised similarly but using the expected value instead of the median.

Finally, we provide some intuition to (41) and show a connection to the local penalisation method used to parallelise sequential BO designs by Gonzalez et al. (2016). Suppose we are selecting the $r$th point of a batch where $2 \leq r \leq b$. Comparison of (38) and (41) shows that (41) equals the original design criterion in (38) multiplied by a weight function $\omega(\boldsymbol{\theta};\boldsymbol{\theta}^*_{1:r-1}) \triangleq \sinh(us_{t+r-1}(\boldsymbol{\theta};\boldsymbol{\theta}^*_{1:r-1}))/\sinh(us_t(\boldsymbol{\theta}))$. It is easy to see that $\omega(\boldsymbol{\theta};\boldsymbol{\theta}^*_{1:r-1}) \in [0,1]$. This shows that when we take the median over the log-likelihood evaluation at the pending points $\boldsymbol{\theta}^*_{1:r-1}$, we are implicitly making the original acquisition function smaller around the pending points and, consequently, penalising additional evaluations there. This resembles the heuristic method by Gonzalez et al. (2016), who proposed to multiply the non-negative acquisition function, such as the objective of (38), with $\prod_{1\leq j<r}\varphi(\boldsymbol{\theta};\boldsymbol{\theta}^*_j)$, where $\varphi(\boldsymbol{\theta};\boldsymbol{\theta}^*_j)$ are local penalising functions around the pending evaluation locations $\boldsymbol{\theta}^*_j$, when selecting the $r$th point $\boldsymbol{\theta}^*_r$ of the current batch. However, one difference between these approaches is that our weight function $\omega$ takes the interactions between the pending points into account and it cannot be factorised as $\omega(\boldsymbol{\theta},\boldsymbol{\theta}^*_{1:r-1}) = \prod_{1\leq j<r}\varphi(\boldsymbol{\theta};\boldsymbol{\theta}^*_j)$. Also, our weight function is not a tuning parameter but follows automatically from our analysis.

## 6 Experiments

We empirically investigate the performance of the proposed algorithm with different design strategies developed in Section 5. We compare the sequential, batch, and greedy batch strategies based on EIV and IMIQR to sequential and greedy versions of MAXV (which is essentially the same as the BAPE method by Kandasamy et al. 2015) and MAXIQR. As a simple baseline we also sample design points from the prior (always uniform) and this method is abbreviated as RAND.

We report the results as figures whose y-axis shows the accuracy between the estimated and the ground truth posterior using total variation distance (TV). TV between pdfs $\pi_1$ and $\pi_2$ is defined as $\text{TV}(\pi_1,\pi_2) = 1/2\int_\Theta |\pi_1(\boldsymbol{\theta}) - \pi_2(\boldsymbol{\theta})|\,\mathrm{d}\boldsymbol{\theta}$ and is computed using numerical integration in 2D. In higher dimensional cases we compute the average TV between the marginal posterior densities using MCMC samples. The marginal median estimator in (17) is used to obtain the point estimate for the posterior pdf. The x-axis represents the iteration $i$ of the Algorithm 1 (unless explicitly stated otherwise) which serves as a proxy to the total wall-time when the noisy likelihood evaluations are assumed to dominate the total computational cost. We use a fixed simulation budget so that the batch-sequential methods terminate earlier than the sequential ones because they spend the evaluation budget $b$ times faster due to the parallel computation.

We consider two sets of experiments: toy models where noisy log-likelihood evaluations are directly evaluated (Section 6.1) and real-world simulator-based statistical models where SL is used to obtain noisy log-likelihood evaluations using $N$ repeated simulations at each proposed parameter (Section 6.2). Although in the SL case it is often possible to adjust $N$ adaptively, we use $N = 100$ (unless explicitly stated otherwise) for simplicity. In the supplementary material D.4 we show that our batch methods are beneficial for SL even though in principle it would be possible to directly parallelise the $N$ simulations themselves. For example, when $1,000$ computer cores are available for the simulations (e.g. in a high performance computing cluster), it is beneficial to use the batch strategies (e.g. $N = 100$ and $b = 10$) instead parallelising the evaluations at a single location ($N = 1000$ and $b = 1$). More elaborate analysis of resource allocation is left for future work. A MATLAB implementation of our algorithms is available at `https://github.com/mjarvenpaa/parallel-GP-SL`.

## 6.1 Noisy toy model likelihoods

We first define three 2D densities with different characteristics: a simple Gaussian density called 'Simple', a banana-shaped density 'Banana' and a bimodal density 'Bimodal'. We then construct three 6D densities so that their 2D blocks are independent and have the corresponding 2D densities as their 2D marginals. Detailed specification and illustrations can be found in supplementary material D.2. We use the same names for the 6D densities as for the corresponding 2D ones (except that 'Bimodal' is called 'Multimodal' in 6D because it has $2^3 = 8$ modes). The independence assumption is not taken into account in the GP model to make the inference problem more challenging. For simplicity, $\sigma_n(\boldsymbol{\theta})$ is assumed constant i.e. it does not depend on the magnitude of the log-likelihood and its value is obtained using MAP estimation together with other GP hyperparameters $\boldsymbol{\phi}$ at each iteration $i$ in Algorithm 1. As an initial design in 6D we generate $b_0 = 20$ parameters ('Simple') or $b_0 = 50$ ('Banana' and 'Multimodal') from uniform priors. In the 2D case we always use $b_0 = 10$. We use a fixed total budget of $t = 620$ noisy log-likelihood evaluations ('Simple') or $t = 650$ ('Banana' and 'Multimodal') for both sequential and batch methods in 6D.

The results with different sequential and greedy batch-sequential strategies in 6D case with batch size $b = 5$ are shown in Figure 3. Good posterior approximations for the Simple example are obtained earlier than for the two other models. This is a consequence of the quadratic terms in the GP prior mean function and the exact Gaussian shape of the posterior. However, more complicated posteriors are also estimated accurately although more iterations are needed to obtain reasonable approximations. The IMIQR method works clearly the best outperforming EIV and the heuristic MAXV and MAXIQR methods which either need more iterations to obtain good approximation or fail completely to reach good results. The uniform design RAND works adequately in the Simple and Banana models but often produces poor estimates for the Multimodal case. Unsurprisingly, its performance is also poor in the real-world scenarios in Section 6.2.

The batch-sequential strategies improve the convergence speed as compared to the corresponding sequential strategies in all cases of Figure 3. In particular, the greedy batch versions of MAXV and MAXIQR even outperform the corresponding sequential methods. The greedy batch strategy in these cases encourages exploration as compared to the corresponding sequential strategy and this effect counterbalances the exploitative nature of MAXV and MAXIQR. In supplementary material D.3 we compare the joint and greedy batch strategies in 2D case where joint maximisation is still feasible. Their difference is found small for IMIQR and small or moderate for EIV suggesting that the greedy strategies are in practice nearly optimal.

Figure 4 and further examples in the supplementary material D.3 illustrate the design points and estimated posteriors for various design strategies in 2D case. An important observation is that MAXV and IMIQR are exploitative i.e. they produce

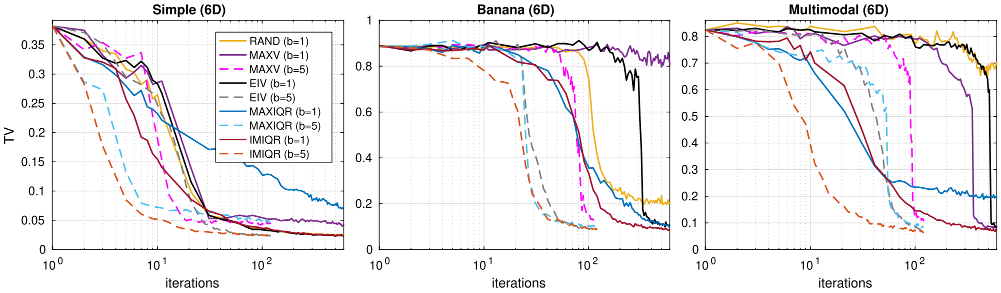

Figure 3: Results for the 6D toy densities. The lines show the median TV over 50 repeated simulations. Note that x-axis is on log-scale and the maximum number of iterations for the sequential methods is $i = 600$ and for batch methods $i = 120$.

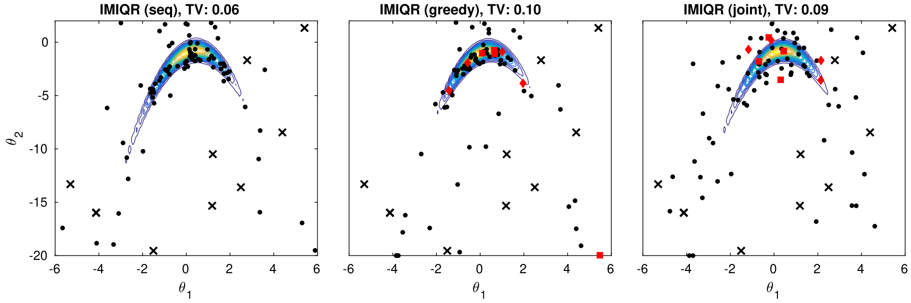

Figure 4: The design locations for IMIQR are shown after 90 noisy log-likelihood evaluations of the 2D Banana example with noise level $\sigma_n = 1$. The black crosses show the $b_0 = 10$ initial evaluations, black dots show obtained design points except for the last two batches, and the red squares and diamonds show the last two batches.

points near the mode of the posterior where the local measure of uncertainty they use tend to be highest. Also, in general, the sequential and batch methods produce similar designs. However, greedy MAXIQR generates more points on the boundary than the corresponding sequential strategy and the joint IMIQR produces slightly more diverse design points as the sequential and greedy batch IMIQR. In all cases, IMIQR avoids redundant evaluations on the boundaries.

We investigate the effect of batch size $b$ in the greedy batch MAXIQR and MAXIQR algorithms in Figure 5. In general, the convergence speed of both methods scales well as a function of $b$ and $b = 10$ already yields useful improvements. However, increasing $b$ over 40 would improve the results only slightly. Greedy batch IMIQR works overall better than batch MAXIQR. The variability in the posterior approximations produced by IMIQR is small in all cases unlike for MAXIQR which occasionally produced poor approximations (not shown for clarity). In the supplementary material D.3 we compare EIV and IMIQR in 2D. These results show that the greedy IMIQR batch-sequential

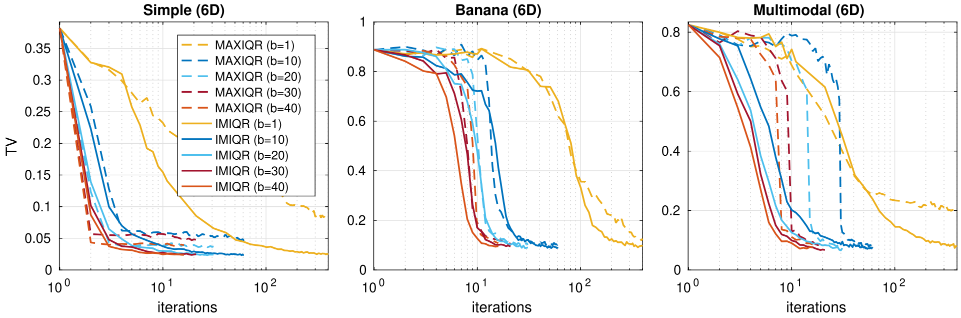

Figure 5: Results with greedy batch strategies and varying batch size $b$ for the 6D toy models. For each method, the median TV computed over 50 repeated simulations is shown. The x-axis is truncated after $i = 400$ iterations to ease visualisation.

strategy outperforms the corresponding EIV strategy although their difference is small in the corresponding sequential cases. This suggests that, even when $\sigma_n$ is small so that the variance in EIV serves as a reasonable measure of uncertainty and the sequential EIV works similarly to IMIQR, the greedy batch median-based IMIQR design strategy better mimics the sequential decisions than EIV.

## 6.2 Simulation models

We perform experiments with three benchmark problems used previously in the ABC literature. Two of these are shown here and the third one in the supplementary material D.5. While the proposed methodology is particularly useful for expensive simulation models, we however consider only relatively cheap models as this allows to repeat the computations many times with different realisations of randomness to assess the variability and robustness, and to conduct accurate comparisons to reasonable ground truth posteriors. Nevertheless, these experiments serve as examples of challenging real-world inference scenarios where the GP and SL modelling assumptions do not hold exactly. In each problem, we set the unknown parameter of the simulation model to a value used previously in the literature and generated one data set from the simulation model using this “true” parameter. The posterior used as the ground truth was computed using SL-MCMC. Multiple chains each with length $10^6$ were used to ensure that the variability due to Monte Carlo error was small.

### Ricker model

We first consider the Ricker model presented in Wood (2010). In this model $N_t$ denotes the number of individuals in a population at time $t$ which evolves according to the discrete time stochastic process $N_{t+1} = rN_t \exp(-N_t + \varepsilon_t)$, for $t = 1, \ldots, T$, where $\varepsilon_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_\varepsilon^2)$. The initial population size is $N_0 = 1$. It is assumed that only a noisy

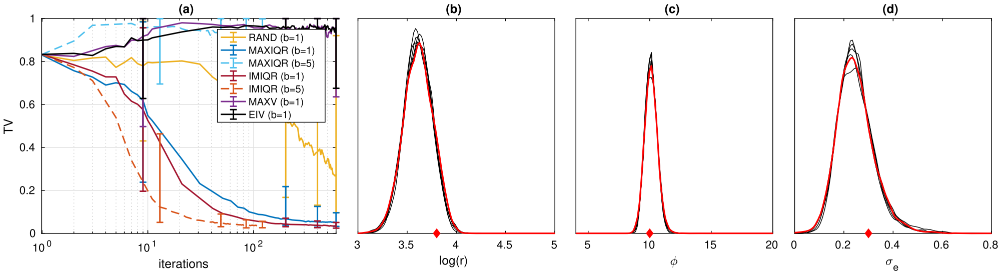

Figure 6: Results for the Ricker model. (a) The median TV and 90% variability interval over 100 repeated runs of the algorithms. (b-d) Estimated posterior marginal densities (black) shown for five typical runs of the algorithm with the greedy batch IMIQR strategy. The ground truth computed with SL-MCMC (red) and the true parameter value (red diamond) are also shown for comparison.

measurement $x_t$ of the population size $N_t$ at each time point is available with the Poisson observation model $x_t \,|\, N_t, \phi \sim \text{Poi}(\phi N_t)$. Given data $\mathbf{x} = (x_t)_{t=1}^T$, the goal is to infer the three parameters $\boldsymbol{\theta} = (\log(r), \phi, \sigma_\varepsilon)$. We use the uniform prior $(\log(r), \phi, \sigma_\varepsilon) \sim \mathcal{U}([3,5] \times [4,20] \times [0,0.8])$. The same 13 summary statistics as in Wood (2010); Gutmann and Corander (2016); Price et al. (2018) are used to compute log-SL evaluations. The number of repeated simulations is fixed to $N = 100$. The "true" parameter to be estimated is $\boldsymbol{\theta}_{\text{true}} = (3.8, 10, 0.3)$ and it is used to generate the observed data with length $T = 50$. The initial training data size is $b_0 = 30$ and the additional budget of simulations is 600 so that the total budget is 630 SL evaluations corresponding 63000 simulations. The integrals of EIV and IMIQR are approximated using IS and $\sigma_n^2$ is estimated using bootstrap as described in Section 5.4.

Figure 6 shows the results. We see that the EIV and MAXV strategies perform poorly. These strategies tend to evaluate where the variance of the posterior is high, although as discussed in Section 5, these do not necessarily correspond to the regions with non-negligible likelihood. In fact, the magnitude of the log-likelihood and its noise variance $\sigma_n^2$ grow fast near the boundaries of the parameter space where the chaotic nature of the model also makes the log-likelihood surface irregular which further causes difficulties with GP modelling. The IMIQR method again produces the best posterior approximations which are comparable to the true SL posterior. Some examples are shown in Figure 6b-d. Also, MAXIQR method works well on average but it produces less coherent results than IMIQR which is likely the result of its exploitative nature. In addition, unexpectedly, the greedy MAXIQR method performs poorly. The probable reason is that the batch evaluations become too diverse having many evaluations in the boundary which leads to poor GP fitting and subsequently poor future designs. However, the robust batch-sequential IMIQR method with $b = 5$ works as expected producing useful improvement to the convergence speed as compared to sequential IMIQR.

### g-and-k model

We consider the g-and-k distribution as in Price et al. (2018). The g-and-k model is a flexible probability distribution defined via its quantile function

$$Q(\Phi^{-1}(p); \boldsymbol{\theta}) = a + b\left(1 + c\frac{1-\exp(-g\Phi^{-1}(p))}{1+\exp(-g\Phi^{-1}(p))}\right)(1 + (\Phi^{-1}(p))^2)^k\Phi^{-1}(p), \tag{42}$$

where $a, b, c, g$ and $k$ are parameters and $p \in [0, 1]$ is a quantile. We fix $c = 0.8$ and estimate the parameters $\boldsymbol{\theta} = (a, b, g, k)$ using a uniform prior $\pi(\boldsymbol{\theta}) = \mathcal{U}([2.5, 3.5] \times [0.5, 1.5] \times [1.5, 2.5] \times [0.3, 0.7])$. We use the same four summary statistics as Price et al. (2018) who fitted an auxiliary model, skew $t$-distribution, to the set of samples generated from (42) using maximum likelihood, and took the resulting skew $t$ score vector at the ML estimate as the summary statistic. Although there are only 4 summary statistics, we again use $N = 100$. We use the same settings as for the Ricker model except that the initial design is increased to $b_0 = 40$ so that the total budget is 640 SL evaluations. The true value of the parameter is chosen to be $\boldsymbol{\theta}_{\text{true}} = (3, 1, 2, 0.5)$.

Overall, the results in Figure 7 are similar to those of the Ricker model. However, the larger parameter space slows down the convergence speeds initially as expected, as compared to the Ricker model. Low dimension of the summary statistic and the moderately large value $N = 100$ cause the log-likelihood evaluations to be quite accurate near the modal area of the likelihood ($\sigma_n(\boldsymbol{\theta}_{\text{true}}) \approx 0.15$) and we expect that smaller $N$ might be already enough. However, using $N = 100$ ensures accurate variance estimates using

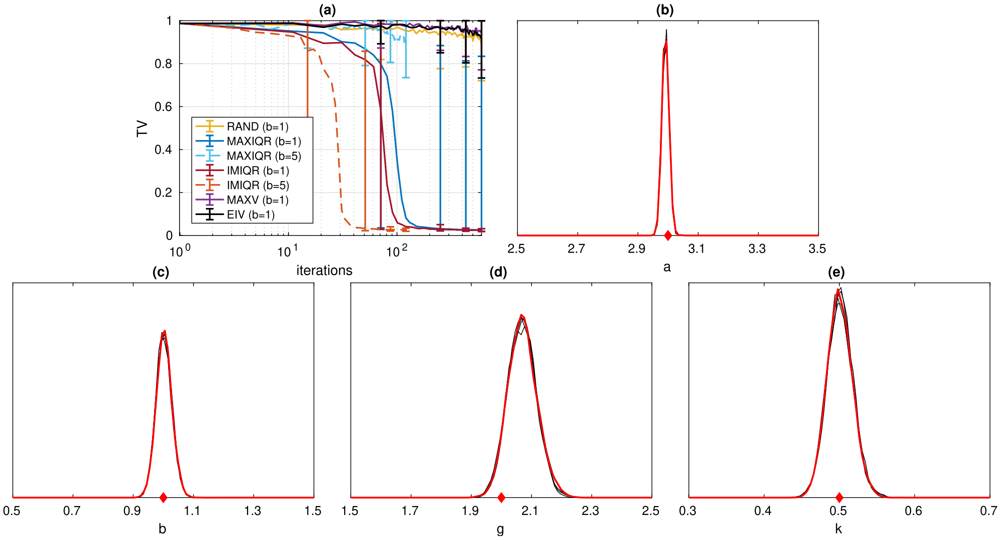

Figure 7: (a) Results for the g-and-k model. (a) Median TV and 90% variability interval over 100 repeated runs. (b-e) Some estimated posterior marginal densities illustrated as in Figure 6.

the bootstrap. While MAXIQR strategy works almost as well as IMIQR on average, it completely fails in some individual repeated experiments producing long variability intervals in Figure 7 leaving IMIQR as the only successful method.

### Effect of batch size

As the last experiment, we investigate the improvements brought by the batch-sequential IMIQR strategy in the case of real-world simulation models. We use the Ricker and g-and-k models from the previous subsections. The experiment details are the same except that we consider only IMIQR strategy with several batch sizes $b \in \{2, 5, 10, 20, 30\}$. The results in Figure 8 show that, on average, the greedy batch-sequential IMIQR with batch sizes up to 30 produces as good approximations as the corresponding sequential strategy. The convergence speed is also improved almost linearly.

However, the variability in the quality of the estimated posteriors increases with larger batch sizes when the total budget of simulations is kept fixed. While most of the repeated runs of the algorithm have converged to excellent approximations in all cases as seen in Figure 8, there were some individual runs where the algorithm did not yet converge when the budget was used. The posterior estimate at the final iteration is often quite poor in these cases. Most of these happen with Ricker model when $b \geq 20$ and with g-and-k model when $b = 30$. However, this behaviour is not surprising: When $b$ is large, the complete batch is constructed using the same limited information which

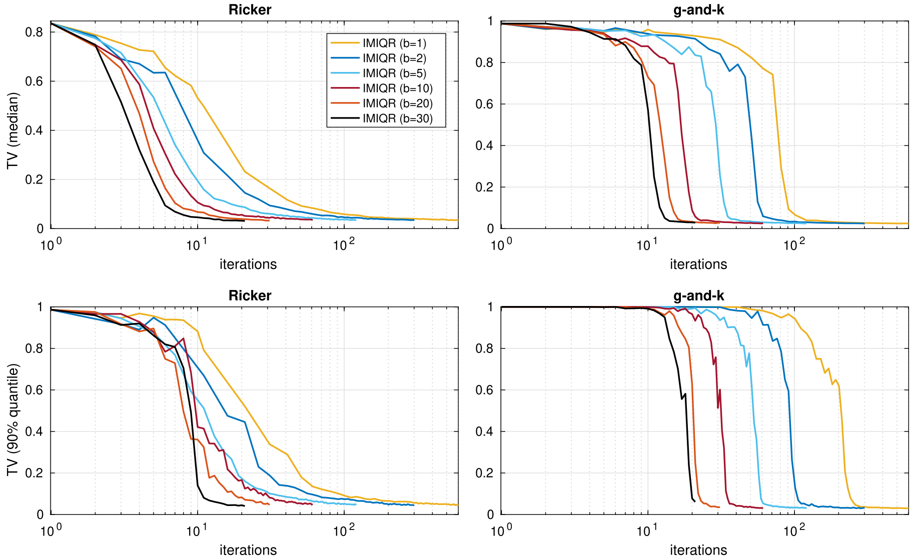

Figure 8: Results of the greedy batch strategy IMIQR with various batch sizes $b$ for Ricker and g-and-k models. Top row shows the median TV over 100 repeated experiments and the bottom row shows the corresponding 90% quantile.

necessarily produces occasional poor batches providing little information. Furthermore, the importance density in (37) is likely to get worse when the batch size is increased and cause the last points in the batch to be less useful. It is thus inevitable that the batch size should not be chosen too large. Nevertheless, it is seen that batch size $b = 10$ already produces substantial gains (especially considering that further parallelisation is often possible with respect to $N$) and produces consistently accurate posterior approximations.

## 7 Discussion and conclusions

If only a limited number of noisy log-likelihood evaluations can be computed, standard techniques such as MCMC become difficult to use for Bayesian inference. To tackle the problem, we constructed a hierarchical GP surrogate model for the noisy log-likelihood and discussed properties of the resulting estimators of the (unnormalised) posterior. We developed two batch-sequential strategies (EIV and IMIQR) based on Bayesian decision theory, to (semi-)optimally select the next evaluation locations and to parallelise the costly simulations. We also considered heuristic design strategies (MAXV and MAXIQR). We provided some theoretical analysis: We derived an approximate bound for the greedy optimisation of the batch IMIQR method using the concept of weak submodularity, showed a connection between the UCB (a common BO method) and the MAXIQR strategy, and between batch MAXIQR and the local penalisation method by Gonzalez et al. (2016). The proposed methods were investigated experimentally.

The IMIQR strategy was found to be robust both to violations of the GP surrogate model assumptions and to the heavy-tails of the resulting distributions. Unlike the other design strategies, it consistently produced posterior approximations comparable to the ground truth. Greedy batch-sequential IMIQR strategy was found to be highly useful to parallelise the potentially expensive simulations. In our experiments it produced substantial, sometimes even linear, speed improvements for batch sizes $b \lesssim 20$. We thus recommend the IMIQR strategy. In general we were able to obtain useful posterior approximations with $10,000$ to $20,000$ simulations that can be easily parallelised. This is considerably less than e.g. using (pseudo-marginal) MCMC requiring typically at least tens of thousands of iterations corresponding to millions of simulations, careful convergence assessment and tuning of the proposal density. Another important observation was that the heuristic strategies that evaluate where the current uncertainty is highest, despite their small computational cost and good performance in earlier studies with deterministic evaluations, worked poorly with noisy log-likelihood evaluations.

Similarly to other GP surrogate techniques such as BO, fitting the GP and finding the next evaluation locations by optimising the design criterion is however not free. Our unoptimised MATLAB implementations of MAXV and MAXIQR are fast, but optimisation of the EIV and IMIQR design criteria takes a couple of seconds in 2D and around 20 to 80 seconds per parameter in 3D and 4D. This means that the proposed algorithm is useful when the simulation time is several seconds or more, which is however true with many real-world simulation models. Furthermore, the quality of the posterior approximation also depends on the choice of the surrogate model. We used the same GP model in all of our experiments with no problem-specific tuning, which already produced good results. However, some problems would certainly benefit from further adjustment and incorporation of domain knowledge. For example, if the likelihood is expected to be flat, a GP prior with a constant mean function might be appropriate.

In this work, similarly to Rasmussen (2003); Wilkinson (2014); Kandasamy et al. (2015); Gutmann and Corander (2016); Drovandi et al. (2018), we built our surrogate GP model for the log-likelihood. An alternative way would be to model the summary statistics. Meeds and Welling (2014) used such an approach but they assumed that the summary statistics are independent. However, modelling the scalar-valued log-likelihood is simpler and our approach also applies as such to non-ABC scenarios with exact (but potentially expensive) log-likelihood evaluations as in Osborne et al. (2012); Kandasamy et al. (2015); Wang and Li (2018); Acerbi (2018). Gutmann and Corander (2016); Järvenpää et al. (2018, 2019) modelled the discrepancy between simulated and observed data with a GP and obtained reasonable posterior approximations with only a few hundred model simulations. Here we need $N$ repeated simulations just to compute the log-likelihood for a single parameter value, which can be seen as the price of not having to specify an explicit discrepancy measure and the ABC tolerance.

We see several avenues for future research. The consistency and convergence rates of our algorithms could be investigated theoretically. Some work towards that direction has been done by Bect et al. (2019); Stuart and Teckentrup (2018). Adaptive control of the number of repeated simulations $N$ could likely be used to further reduce the number of simulations required, possibly as in Picheny et al. (2013). Some simulation models may behave unexpectedly near the boundaries of the parameter space violating GP model assumptions as we saw with the Ricker model in Section 6. Similarly, situations where the prior is significantly more diffuse than the posterior may be unsuitable for our approach that relies on a global GP surrogate. Consequently, it would be useful to learn adaptively not only where to evaluate next but also which parameter regions to rule out completely. This could be done as in Wilkinson (2014) or possibly by adapting ideas from the constrained BO literature (Gardner et al., 2014; Sui et al., 2015).

## Supplementary Material

Supplementary material of "Parallel Gaussian process surrogate Bayesian inference with noisy likelihood evaluations"
(DOI: 10.1214/20-BA1200SUPP; .pdf). The supplementary material contains proofs, implementation details and additional experimental results.

## References

Acerbi, L. (2018). "Variational Bayesian Monte Carlo." In *Advances in Neural Information Processing Systems 31*, 8223–8233. 2, 12, 26

An, Z., Nott, D. J., and Drovandi, C. (2019a). "Robust Bayesian synthetic likelihood via a semi-parametric approach." *Statistics and Computing*. MR4065218. doi: https://doi.org/10.1007/s11222-019-09904-x. 4

An, Z., South, L. F., Nott, D. J., and Drovandi, C. C. (2019b). "Accelerating Bayesian Synthetic Likelihood with the Graphical Lasso." *Journal of Computational and Graphical Statistics*, 28(2): 471–475. MR3974895. doi: https://doi.org/10.1080/10618600.2018.1537928. 4

Ankenman, B., Nelson, B. L., and Staum, J. (2010). "Stochastic Kriging for Simulation Metamodeling." *Operations Research*, 58(2): 371–382. MR2674803. doi: https://doi.org/10.1287/opre.1090.0754. 16

Azimi, J., Alan, F., and Fern, X. Z. (2010). "Batch Bayesian Optimization via Simulation Matching." In *Advances in Neural Information Processing Systems 23*, 109–117. 2

Bach, F. (2013). *Learning with Submodular Functions: A Convex Optimization Perspective*. Hanover, MA, USA: Now Publishers Inc. 13

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). "Approximate Bayesian computation in population genetics." *Genetics*, 162(4): 2025–2035. 1

Bect, J., Bachoc, F., and Ginsbourger, D. (2019). "A supermartingale approach to Gaussian process based sequential design of experiments." *Bernoulli*, 25(4A): 2883–2919. MR4003568. doi: https://doi.org/10.3150/18-BEJ1074. 26

Bect, J., Ginsbourger, D., Li, L., Picheny, V., and Vazquez, E. (2012). "Sequential design of computer experiments for the estimation of a probability of failure." *Statistics and Computing*, 22(3): 773–793. MR2909621. doi: https://doi.org/10.1007/s11222-011-9241-4. 3, 11

Briol, F.-X., Oates, C. J., Girolami, M., Osborne, M. A., and Sejdinovic, D. (2019). "Probabilistic Integration: A Role in Statistical Computation?" *Statistical Science*, 34(1): 1–22. MR3938958. doi: https://doi.org/10.1214/18-STS660. 6

Brochu, E., Cora, V. M., and de Freitas, N. (2010). "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning." Available at: https://arxiv.org/abs/1012.2599. 2

Chai, H. R. and Garnett, R. (2019). "Improving Quadrature for Constrained Integrands." In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2751–2759. 2

Chevalier, C., Bect, J., Ginsbourger, D., Vazquez, E., Picheny, V., and Richet, Y. (2014). "Fast Parallel Kriging-Based Stepwise Uncertainty Reduction With Application to the Identification of an Excursion Set." *Technometrics*, 56(4): 455–465. MR3290615. doi: https://doi.org/10.1080/00401706.2013.860918. 3, 15, 16

Cockayne, J., Oates, C., Sullivan, T., and Girolami, M. (2019). "Bayesian Probabilistic Numerical Methods." *SIAM Review*, 61(4): 756–789. MR4027836. doi: https://doi.org/10.1137/17M1139357. 6

Contal, E., Buffoni, D., Robicquet, A., and Vayatis, N. (2013). "Parallel Gaussian process optimization with upper confidence bound and pure exploration." In *Lecture Notes in Computer Science*. 2

Desautels, T., Krause, A., and Burdick, J. W. (2014). "Parallelizing Exploration-Exploitation Tradeoffs in Gaussian Process Bandit Optimization." *Journal of Machine Learning Research*, 15: 4053–4103. MR3317214. 2, 17

Drovandi, C. C., Moores, M. T., and Boys, R. J. (2018). "Accelerating pseudo-marginal MCMC using Gaussian processes." *Computational Statistics & Data Analysis*, 118: 1–17. MR3715260. doi: `https://doi.org/10.1016/j.csda.2017.09.002`. 2, 5, 26

Frazier, D. T., Nott, D. J., Drovandi, C., and Kohn, R. (2019). "Bayesian inference using synthetic likelihood: asymptotics and adjustments." Available at: `https://arxiv.org/abs/1902.04827`. 4

Gardner, J., Kusner, M., Zhixiang, Weinberger, K., and Cunningham, J. (2014). "Bayesian Optimization with Inequality Constraints." In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, 937–945. 26

Ginsbourger, D., Le Riche, R., and Carraro, L. (2010). *Kriging Is Well-Suited to Parallelize Optimization*, 131–162. Berlin, Heidelberg: Springer Berlin Heidelberg. 2, 12, 13, 17

Gonzalez, J., Dai, Z., Lawrence, N. D., and Hennig, P. (2016). "Batch Bayesian Optimization via Local Penalization." In *International Conference on Artificial Intelligence and Statistics*, 1, 648–657. 2, 17, 18, 25

González, J., Osborne, M., and Lawrence, N. D. (2016). "GLASSES: Relieving The Myopia of Bayesian Optimisation." In *Proceedings of the Nineteenth International Workshop on Artificial Intelligence and Statistics*. 11

Gunter, T., Osborne, M. A., Garnett, R., Hennig, P., and Roberts, S. J. (2014). "Sampling for Inference in Probabilistic Models with Fast Bayesian Quadrature." In *Advances in Neural Information Processing Systems 27*, 2789–2797. 2, 12, 17

Gutmann, M. U. and Corander, J. (2016). "Bayesian optimization for likelihood-free inference of simulator-based statistical models." *Journal of Machine Learning Research*, 17(125): 1–47. MR3555016. 2, 3, 5, 22, 26

Hennig, P., Osborne, M. A., and Girolami, M. (2015). "Probabilistic numerics and uncertainty in computations." *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 471(2179): 20150142. MR3378744. doi: `https://doi.org/10.1098/rspa.2015.0142`. 2, 6

Hennig, P. and Schuler, C. J. (2012). "Entropy Search for Information-Efficient Global Optimization." *Journal of Machine Learning Research*, 13(1999): 1809–1837. MR2956343. 15

Hernández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. (2014). "Predictive Entropy Search for Efficient Global Optimization of Black-box Functions." *Advances in Neural Information Processing Systems 28*, 1–9. 15

Jabot, F., Lagarrigues, G., Courbaud, B., and Dumoulin, N. (2014). "A comparison of emulation methods for Approximate Bayesian Computation." Available at: http://arxiv.org/abs/1412.7560. 2

Järvenpää, M., Gutmann, M. U., Pleska, A., Vehtari, A., and Marttinen, P. (2019). "Efficient Acquisition Rules for Model-Based Approximate Bayesian Computation." *Bayesian Analysis*, 14(2): 595–622. MR3934099. doi: https://doi.org/10.1214/18-BA1121. 2, 3, 6, 8, 11, 12, 16, 26

Järvenpää, M., Gutmann, M. U., Vehtari, A., and Marttinen, P. (2018). "Gaussian process modelling in approximate Bayesian computation to estimate horizontal gene transfer in bacteria." *The Annals of Applied Statistics*, 12(4): 2228–2251. MR3875699. doi: https://doi.org/10.1214/18-AOAS1150. 1, 26

Järvenpää, M., Gutmann, M. U., Vehtari, A., and Marttinen, P. (2020). "Parallel Gaussian Process Surrogate Bayesian Inference with Noisy Likelihood Evaluations – Supplementary Material." *Bayesian Analysis*. doi: https://doi.org/10.1214/20-BA1200SUPP. 3

Kandasamy, K., Schneider, J., and Póczos, B. (2015). "Bayesian active learning for posterior estimation." In *International Joint Conference on Artificial Intelligence*, 3605–3611. 2, 3, 12, 17, 18, 26

Karvonen, T., Oates, C. J., and Särkkä, S. (2018). "A Bayes-Sard Cubature Method." In *Advances in Neural Information Processing Systems 31*, 5886–5897. MR4026668. doi: https://doi.org/10.1007/s11222-019-09896-8. 2

Kennedy, M. C. and O'Hagan, A. (2001). "Bayesian calibration of computer models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3): 425–464. MR1858398. doi: https://doi.org/10.1111/1467-9868.00294. 1

Krause, A. and Cevher, V. (2010). "Submodular Dictionary Selection for Sparse Representation." In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 567–574. 14

Krause, A., Singh, A., and Guestrin, C. (2008). "Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies." *Journal of Machine Learning Research*, 9: 235–284. 13, 14

Lintusaari, J., Gutmann, M. U., Dutta, R., Kaski, S., and Corander, J. (2017). "Fundamentals and Recent Developments in Approximate Bayesian Computation." *Systematic biology*, 66(1): e66–e82. 1, 3

Lyu, X., Binois, M., and Ludkovski, M. (2018). "Evaluating Gaussian Process Metamodels and Sequential Designs for Noisy Level Set Estimation." Available at: http://arxiv.org/abs/1807.06712. 3, 8

Marin, J. M., Pudlo, P., Robert, C. P., and Ryder, R. J. (2012). "Approximate Bayesian computational methods." *Statistics and Computing*, 22(6): 1167–1180. MR2992292. doi: https://doi.org/10.1007/s11222-011-9288-2. 1, 3

Marttinen, P., Gutmann, M. U., Croucher, N. J., Hanage, W. P., and Corander, J.

(2015). “Recombination produces coherent bacterial species clusters in both core and accessory genomes.” *Microbial Genomics*, 1(5). 1

Meeds, E. and Welling, M. (2014). “GPS-ABC: Gaussian Process Surrogate Approximate Bayesian Computation.” In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*. 2, 26

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). “An analysis of approximations for maximizing submodular set functions—I.” *Mathematical Programming*, 14(1): 265–294. MR0503866. doi: `https://doi.org/10.1007/BF01588971`. 13

O’Hagan, A. (1991). “Bayes-Hermite quadrature.” *Journal of Statistical Planning and Inference*. MR1144171. doi: `https://doi.org/10.1016/0378-3758(91)90002-V`. 2

O’Hagan, A. and Kingman, J. F. C. (1978). “Curve Fitting and Optimal Design for Prediction.” *Journal of the Royal Statistical Society. Series B (Methodological)*, 40(1): 1–42. MR0512140. 5

Osborne, M. A., Duvenaud, D., Garnett, R., Rasmussen, C. E., Roberts, S. J., and Ghahramani, Z. (2012). “Active Learning of Model Evidence Using Bayesian Quadrature.” *Advances in Neural Information Processing Systems 26*, 1–9. 2, 26

Picheny, V., Ginsbourger, D., Richet, Y., and Caplin, G. (2013). “Quantile-Based Optimization of Noisy Computer Experiments With Tunable Precision.” *Technometrics*, 55(1): 2–13. MR3038476. doi: `https://doi.org/10.1080/00401706.2012.707580`. 26

Price, L. F., Drovandi, C. C., Lee, A., and Nott, D. J. (2018). “Bayesian Synthetic Likelihood.” *Journal of Computational and Graphical Statistics*, 27(1): 1–11. MR3788296. doi: `https://doi.org/10.1080/10618600.2017.1302882`. 2, 3, 4, 22, 23

Rasmussen, C. E. (2003). “Gaussian Processes to Speed up Hybrid Monte Carlo for Expensive Bayesian Integrals.” *Bayesian Statistics 7*, 651–659. MR2003529. 2, 26

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press. MR2514435. 1, 5

Riihimäki, J. and Vehtari, A. (2014). “Laplace Approximation for Logistic Gaussian Process Density Estimation and Regression.” *Bayesian Analysis*, 9(2): 425–448. MR3217002. doi: `https://doi.org/10.1214/14-BA872`. 5

Robert, C. P. (2007). *The Bayesian Choice*. New York: Springer, second edition. MR2723361. 6

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. New York: Springer, second edition. MR2080278. doi: `https://doi.org/10.1007/978-1-4757-4145-2`. 1

Shah, A. and Ghahramani, Z. (2015). “Parallel Predictive Entropy Search for Batch Global Optimization of Expensive Objective Functions.” In *Advances in Neural Information Processing Systems 28*, 12. 2

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and de Freitas, N. (2015). "Taking the human out of the loop: A review of Bayesian optimization." *Proceedings of the IEEE*, 104(1). 2, 17

Sinsbeck, M. and Nowak, W. (2017). "Sequential Design of Computer Experiments for the Solution of Bayesian Inverse Problems." *SIAM/ASA Journal on Uncertainty Quantification*, 5(1): 640–664. MR3679325. doi: https://doi.org/10.1137/15M1047659. 2, 6, 11, 12

Snoek, J., Larochelle, H., and Adams, R. P. (2012). "Practical Bayesian optimization of machine learning algorithms." In *Advances in Neural Information Processing Systems 25*, 1–9. 2, 13, 17

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. (2010). "Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design." In *Proceedings of the 27th International Conference on Machine Learning*, 1015–1022. MR2952544. doi: https://doi.org/10.1109/TIT.2011.2182033. 17

Stuart, A. M. and Teckentrup, A. L. (2018). "Posterior consistency for Gaussian process approximations of Bayesian posterior distributions." *Mathematics for Computing*, 87: 721–753. MR3739215. doi: https://doi.org/10.1090/mcom/3244. 8, 26

Sui, Y., Gotovos, A., Burdick, J., and Krause, A. (2015). "Safe Exploration for Optimization with Gaussian Processes." In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, 997–1005. 26

Thomas, O., Dutta, R., Corander, J., Kaski, S., and Gutmann, M. U. (2018). "Likelihood-free inference by ratio estimation." Available at: https://arxiv.org/abs/1611.10242. 4

Turner, B. M. and Van Zandt, T. (2012). "A tutorial on approximate Bayesian computation." *Journal of Mathematical Psychology*, 56(2): 69–85. MR2909506. doi: https://doi.org/10.1016/j.jmp.2012.02.005. 3

Wang, H. and Li, J. (2018). "Adaptive Gaussian Process Approximation for Bayesian Inference with Expensive Likelihood Functions." *Neural Computation*, 30(11): 3072–3094. MR3873817. doi: https://doi.org/10.1162/neco_a_01127. 2, 26

Wilkinson, R. D. (2014). "Accelerating ABC methods using Gaussian processes." In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*. 2, 5, 16, 26

Wilson, J., Hutter, F., and Deisenroth, M. (2018). "Maximizing acquisition functions for Bayesian optimization." In *Advances in Neural Information Processing Systems 31*, 9906–9917. 2, 13

Wood, S. N. (2010). "Statistical inference for noisy nonlinear ecological dynamic systems." *Nature*, 466: 1102–1104. 2, 3, 21, 22

Wu, J. and Frazier, P. (2016). "The Parallel Knowledge Gradient Method for Batch Bayesian Optimization." In *Advances in Neural Information Processing Systems 29*, 3126–3134. MR3755126. 2, 15

Yu, C. W. and Clarke, B. (2011). "Median loss decision theory." *Journal of Statistical Planning and Inference*, 141(2): 611–623. MR2732931. doi: https://doi.org/10.1016/j.jspi.2010.08.013. 12

**Acknowledgments**

The authors are grateful to an associate editor and two anonymous reviewers for their constructive feedback that helped to significantly improve this article. This work was funded by the Academy of Finland (grants no. 286607 and 294015 to PM). We acknowledge the computational resources provided by Aalto Science-IT project.

# Supplementary material

Marko Järvenpää[1], Michael U. Gutmann[2], Aki Vehtari[1], and Pekka Marttinen[1]

[1]Helsinki Institute for Information Technology HIIT, Department of Computer Science, Aalto University
[2]School of Informatics, University of Edinburgh

March 6, 2020

## Contents

**A Proofs** **1**

**B Analysis of the greedy optimisation of design criteria** **4**
- B.1 On the monotonicity of design criteria . . . . 4
- B.2 Proof of the greedy optimisation bound . . . . 5

**C Additional implementation details** **6**

**D Additional experiments and illustrations** **7**
- D.1 Normality of noisy log-SL evaluations . . . . 7
- D.2 Details of the toy densities . . . . 7
- D.3 Additional experiments and illustration with 2D toy densities . . . . 8
- D.4 Parallellisation of the synthetic likelihood method with IMIQR . . . . 10
- D.5 Experiments with Lorenz model . . . . 13

## A Proofs

We first show that the (marginal) median minimises the expected $L^1$ loss defined in Section 4 and then derive the corresponding Bayes risk in (16). The expected $L^1$ loss can be written similarly to (13) but with the absolute value in place of the quadratic term. It then again follows from the basic results of Bayesian decision theory that the integrand, and thus also the original formula, is minimised when $\tilde{d}(\boldsymbol{\theta}) = \mathrm{med}_{f \mid D_t}(\tilde{\pi}_f(\boldsymbol{\theta})) = \pi(\boldsymbol{\theta}) \exp(m_t(\boldsymbol{\theta}))$ for (almost all) $\boldsymbol{\theta} \in \Theta$.

To derive the formula for the Bayes risk, we fix $\boldsymbol{\theta}$ and shorten the notation so that $f_{\boldsymbol{\theta}} = f(\boldsymbol{\theta}), m_{\boldsymbol{\theta}} = m_t(\boldsymbol{\theta})$ and $s_{\boldsymbol{\theta}} = s_t(\boldsymbol{\theta})$. Then we obtain

$$\mathbb{E}_{f \mid D_t}(|e^{f_{\boldsymbol{\theta}}} - e^{m_{\boldsymbol{\theta}}}|) = \int_{-\infty}^{\infty} |e^{f_{\boldsymbol{\theta}}} - e^{m_{\boldsymbol{\theta}}}| \frac{1}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}} - m_{\boldsymbol{\theta}})^2} \, \mathrm{d}f_{\boldsymbol{\theta}} \tag{A.1}$$

$$\begin{aligned} &= \int_{-\infty}^{m_{\boldsymbol{\theta}}} (e^{m_{\boldsymbol{\theta}}} - e^{f_{\boldsymbol{\theta}}}) \frac{1}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}} - m_{\boldsymbol{\theta}})^2} \, \mathrm{d}f_{\boldsymbol{\theta}} \\ &\quad + \int_{m_{\boldsymbol{\theta}}}^{\infty} (e^{f_{\boldsymbol{\theta}}} - e^{m_{\boldsymbol{\theta}}}) \frac{1}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}} - m_{\boldsymbol{\theta}})^2} \, \mathrm{d}f_{\boldsymbol{\theta}} \end{aligned} \tag{A.2}$$

$$= \frac{e^{m_{\boldsymbol{\theta}}}}{2} - \int_{-\infty}^{m_{\boldsymbol{\theta}}} \frac{e^{f_{\boldsymbol{\theta}}}}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}}-m_{\boldsymbol{\theta}})^2} \,\mathrm{d}f_{\boldsymbol{\theta}} + \int_{m_{\boldsymbol{\theta}}}^{\infty} \frac{e^{f_{\boldsymbol{\theta}}}}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}}-m_{\boldsymbol{\theta}})^2} \,\mathrm{d}f_{\boldsymbol{\theta}} - \frac{e^{m_{\boldsymbol{\theta}}}}{2} \tag{A.3}$$

$$= 2\int_{m_{\boldsymbol{\theta}}}^{\infty} \frac{e^{f_{\boldsymbol{\theta}}}}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}}-m_{\boldsymbol{\theta}})^2} \,\mathrm{d}f_{\boldsymbol{\theta}} - \mathbb{E}_{f\,|\,D_t}(e^{f_{\boldsymbol{\theta}}}) \tag{A.4}$$

$$= 2e^{m_{\boldsymbol{\theta}}+s_{\boldsymbol{\theta}}^2/2} \underbrace{\int_{m_{\boldsymbol{\theta}}}^{\infty} \frac{1}{\sqrt{2\pi s_{\boldsymbol{\theta}}^2}} e^{\frac{1}{2s_{\boldsymbol{\theta}}^2}(f_{\boldsymbol{\theta}}-(m_{\boldsymbol{\theta}}+s_{\boldsymbol{\theta}}^2))^2} \,\mathrm{d}f_{\boldsymbol{\theta}}}_{=1-\Phi\left(\frac{m_{\boldsymbol{\theta}}-(m_{\boldsymbol{\theta}}+s_{\boldsymbol{\theta}}^2)}{s_{\boldsymbol{\theta}}}\right)=\Phi(s_{\boldsymbol{\theta}})} - \underbrace{\mathbb{E}_{f\,|\,D_t}(e^{f_{\boldsymbol{\theta}}})}_{=e^{m_{\boldsymbol{\theta}}+s_{\boldsymbol{\theta}}^2/2}} \tag{A.5}$$

$$= e^{m_{\boldsymbol{\theta}}+s_{\boldsymbol{\theta}}^2/2}(2\Phi(s_{\boldsymbol{\theta}})-1), \tag{A.6}$$

where on the fifth line we have completed the square and used the moment-generating function $M_z(t) \triangleq \mathbb{E}(e^{tz}) = \exp(t\mu + \sigma^2 t^2/2)$ of the Gaussian distribution $z \sim \mathcal{N}(\mu, \sigma^2)$. The desired result follows by multiplying the above with prior density $\pi(\boldsymbol{\theta})$ and integrating the resulting formula over $\Theta$.

Next we show a result from matrix algebra that we need in the following several times.

**Lemma A.1.** *Suppose $\mathbf{X}_1, \mathbf{X}_2, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{Y}_1$ and $\mathbf{Y}_2$ are such matrices that the equation below is well-defined, that is, the sizes of the matrices are correct and all the required inverses exist. Then*

$$\begin{bmatrix}\mathbf{X}_1 & \mathbf{X}_2\end{bmatrix}\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{Y}_1\\ \mathbf{Y}_2\end{bmatrix} = \mathbf{X}_1\mathbf{A}^{-1}\mathbf{Y}_1 - (\mathbf{X}_2 - \mathbf{X}_1\mathbf{A}^{-1}\mathbf{B})[\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}]^{-1}(\mathbf{Y}_2 - \mathbf{C}\mathbf{A}^{-1}\mathbf{Y}_1). \tag{A.7}$$

*Proof of Lemma 5.1.* The proof is rather straightforward but laborious. To shorten notation, we rename the training data $\boldsymbol{\theta}_{1:t}$ as $\boldsymbol{\theta}_0$ and the test point $\boldsymbol{\theta}$ as $\boldsymbol{\theta}_\bullet$ and we change the subscripts of various matrices appearing in the GP formulas similarly. With this compact notation, the GP formulas in (7) and (8) become

$$m_0(\boldsymbol{\theta}_\bullet) = \underbrace{\mathbf{k}_{\bullet 0}\mathbf{K}_0^{-1}\mathbf{y}_0}_{\triangleq \tilde{m}_0(\boldsymbol{\theta}_\bullet)} + \mathbf{R}_0^\top(\boldsymbol{\theta}_\bullet)\bar{\gamma}_0, \tag{A.8}$$

$$c_0(\boldsymbol{\theta}_\bullet, \boldsymbol{\theta}_\star) = \underbrace{\mathbf{k}_{\bullet\star} - \mathbf{k}_{\bullet 0}\mathbf{K}_0^{-1}\mathbf{k}_{0\star}}_{\triangleq \tilde{c}_0(\boldsymbol{\theta}_\bullet, \boldsymbol{\theta}_\star)} + \mathbf{R}_0^\top(\boldsymbol{\theta}_\bullet)\underbrace{[\mathbf{B}^{-1} + \mathbf{H}_0\mathbf{K}_0^{-1}\mathbf{H}_0^\top]^{-1}}_{\triangleq \mathbf{W}_0}\mathbf{R}_0(\boldsymbol{\theta}_\star). \tag{A.9}$$

We also define $\boldsymbol{\Lambda}^* = \operatorname{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*), \ldots, \sigma_n^2(\boldsymbol{\theta}_b^*))$.

Given the full training data $D_{0*} = D_0 \cup D^* = \{(y_{0i}, \boldsymbol{\theta}_{0i})\}_{i=1}^t \cup \{(y_i^*, \boldsymbol{\theta}_i^*)\}_{i=1}^b$, using Lemma A.1 analogously as in the proof of Proposition 3.2 in Järvenpää et al. [2019] but acknowledging that $\boldsymbol{\theta}^*$ contains $b$ points and $c_0(\boldsymbol{\theta}^*)$ is thus a $b \times b$ matrix, one obtains the following GP formulas when the GP prior mean function is zero

$$\tilde{m}_{0*}(\boldsymbol{\theta}_\bullet) = \tilde{m}_0(\boldsymbol{\theta}_\bullet) + \tilde{c}_0(\boldsymbol{\theta}_\bullet, \boldsymbol{\theta}^*)[\tilde{c}_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}(\mathbf{y}^* - \tilde{m}_0(\boldsymbol{\theta}^*)), \tag{A.10}$$

$$\tilde{s}_{0*}^2(\boldsymbol{\theta}_\bullet) = \tilde{s}_0^2(\boldsymbol{\theta}_\bullet) - \tilde{c}_0(\boldsymbol{\theta}_\bullet, \boldsymbol{\theta}^*)[\tilde{c}_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}\tilde{c}_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}_\bullet), \tag{A.11}$$

where we have denoted $\tilde{s}_{0*}^2(\boldsymbol{\theta}_\bullet) \triangleq \tilde{c}_{0*}(\boldsymbol{\theta}_\bullet, \boldsymbol{\theta}_\bullet)$ similarly as before.

It remains to handle the extra terms in (A.8) and (A.9) which are due to the non-zero GP prior mean function assumption. We first compute using Lemma A.1 that

$$\begin{aligned}\mathbf{R}_{0*}(\boldsymbol{\theta}_\bullet) &= \mathbf{H}_\bullet - \begin{bmatrix}\mathbf{H}_0 & \mathbf{H}_*\end{bmatrix}\begin{bmatrix}\mathbf{K}_0 & \mathbf{k}_{0*}\\ \mathbf{k}_{*0} & \mathbf{K}_*\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{k}_{0\bullet}\\ \mathbf{k}_{*\bullet}\end{bmatrix}\\ &= \mathbf{H}_\bullet - \mathbf{H}_0\mathbf{K}_0^{-1}\mathbf{k}_{0\bullet} - (\mathbf{H}_* - \mathbf{H}_0\mathbf{K}_0^{-1}\mathbf{k}_{0*})[\mathbf{K}_* - \mathbf{k}_{*0}\mathbf{K}_0^{-1}\mathbf{k}_{0*}]^{-1}(\mathbf{k}_{*\bullet} - \mathbf{k}_{*0}\mathbf{K}_0^{-1}\mathbf{k}_{0\bullet})\\ &= \mathbf{R}_0(\boldsymbol{\theta}_\bullet) - \mathbf{R}_0(\boldsymbol{\theta}^*)[\tilde{c}_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}\tilde{c}_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}_\bullet).\end{aligned} \tag{A.12}$$

A similar computation shows that

$$\mathbf{H}_{0*}\mathbf{K}_{0*}\mathbf{y}_{0*} + \mathbf{B}^{-1}\mathbf{b} = \mathbf{H}_0\mathbf{K}_0^{-1}\mathbf{y}_0 + \mathbf{B}^{-1}\mathbf{b} + \mathbf{R}_0(\boldsymbol{\theta}^*)[\tilde{c}_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}(\mathbf{y}^* - \tilde{m}_0(\boldsymbol{\theta}^*)). \tag{A.13}$$

Similarly we obtain also the formula

$$\mathbf{W}_{0*} = \mathbf{B}^{-1} + \mathbf{H}_{0*}\mathbf{K}_{0*}^{-1}\mathbf{H}_{0*}^{\top} = \mathbf{W}_0 + \mathbf{R}_0(\boldsymbol{\theta}^*)[\tilde{c}(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}\mathbf{R}_0^{\top}(\boldsymbol{\theta}^*) \tag{A.14}$$

from which we further obtain by using the matrix inversion lemma that

$$\mathbf{W}_{0*}^{-1} = \mathbf{W}_0^{-1} - \mathbf{W}_0^{-1}\mathbf{R}_0(\boldsymbol{\theta}^*)[c_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}\mathbf{R}_0^{\top}(\boldsymbol{\theta}^*)\mathbf{W}_0^{-1}. \tag{A.15}$$

Using (A.12), (A.15) and (A.9), as well as some straightforward manipulations, we obtain the formulas

$$\mathbf{R}_{0*}^{\top}(\boldsymbol{\theta}_{\bullet})\mathbf{W}_{0*}^{-1} = \mathbf{R}_0^{\top}(\boldsymbol{\theta}_{\bullet})\mathbf{W}_0^{-1} - c_0(\boldsymbol{\theta}_{\bullet}, \boldsymbol{\theta}^*)[c_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}\mathbf{R}_0^{\top}(\boldsymbol{\theta}^*)\mathbf{W}_0^{-1}, \tag{A.16}$$

$$\begin{aligned}\mathbf{R}_{0*}^{\top}(\boldsymbol{\theta}_{\bullet})\mathbf{W}_{0*}^{-1}\mathbf{R}_{0*}(\boldsymbol{\theta}_{\bullet}) &= \mathbf{R}_0^{\top}(\boldsymbol{\theta}_{\bullet})\mathbf{W}_0^{-1}\mathbf{R}_0(\boldsymbol{\theta}_{\bullet}) + \tilde{c}_0(\boldsymbol{\theta}_{\bullet}, \boldsymbol{\theta}^*)[\tilde{c}_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}\tilde{c}_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{\bullet}) \\ &\quad - c_0(\boldsymbol{\theta}_{\bullet}, \boldsymbol{\theta}^*)[c_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}c_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{\bullet}).\end{aligned} \tag{A.17}$$

Putting the results in (A.10), (A.16) and (A.13) together and after some additional straightforward simplifications, we see that

$$\begin{aligned}m_{0*}(\boldsymbol{\theta}_{\bullet}) &= \tilde{m}_{0*}(\boldsymbol{\theta}_{\bullet}) + \mathbf{R}_{0*}^{\top}(\boldsymbol{\theta}_{\bullet})\bar{\gamma}_{0*} \\ &= m_0(\boldsymbol{\theta}_{\bullet}) + c_0(\boldsymbol{\theta}_{\bullet}, \boldsymbol{\theta}^*)[c_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}(\mathbf{y}^* - m_0(\boldsymbol{\theta}^*)).\end{aligned} \tag{A.18}$$

By the assumption $\mathbf{y}^* \,|\, \boldsymbol{\theta}^*, D_0 \sim \mathcal{N}(m_0(\boldsymbol{\theta}^*), c_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*)$ and using (A.18) we see that $m_{0*}(\boldsymbol{\theta}_{\bullet}) \,|\, \boldsymbol{\theta}^*, D_0 \sim \mathcal{N}(m_0(\boldsymbol{\theta}_{\bullet}), c_0(\boldsymbol{\theta}_{\bullet}, \boldsymbol{\theta}^*)[c_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}c_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{\bullet}))$. Thus (18) holds.

The variance formula now follows similarly. Using (A.11) and (A.17) we obtain

$$\begin{aligned}s_{0*}^2(\boldsymbol{\theta}_{\bullet}) &= \tilde{s}_{0*}^2(\boldsymbol{\theta}_{\bullet}) + \mathbf{R}_{0*}^{\top}(\boldsymbol{\theta}_{\bullet})\mathbf{W}_{0*}^{-1}\mathbf{R}_{0*}(\boldsymbol{\theta}_{\bullet}) \\ &= s_0^2(\boldsymbol{\theta}_{\bullet}) - c_0(\boldsymbol{\theta}_{\bullet}, \boldsymbol{\theta}^*)[c_0(\boldsymbol{\theta}^*) + \boldsymbol{\Lambda}^*]^{-1}c_0(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{\bullet}),\end{aligned} \tag{A.19}$$

from which the claim follows. □

*Proof of Lemma 5.2.* (i) If $\mathbf{P}$ is a permutation matrix that changes the order of the columns of $\boldsymbol{\theta}^*$, then it is easy to see that

$$\begin{aligned}\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*\mathbf{P}) &= c_t(\boldsymbol{\theta}, \boldsymbol{\theta}^*)\mathbf{P}^{\top}[\mathbf{P}c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}^*)\mathbf{P}^{\top} + \mathbf{P}\operatorname{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*), \ldots, \sigma_n^2(\boldsymbol{\theta}_b^*))\mathbf{P}^{\top}]^{-1}\mathbf{P}c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}) && \text{(A.20)} \\ &= c_t(\boldsymbol{\theta}, \boldsymbol{\theta}^*)\mathbf{P}^{\top}\mathbf{P}[c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}^*) + \operatorname{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*), \ldots, \sigma_n^2(\boldsymbol{\theta}_b^*))]^{-1}\mathbf{P}^{\top}\mathbf{P}c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}) && \text{(A.21)} \\ &= c_t(\boldsymbol{\theta}, \boldsymbol{\theta}^*)[c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}^*) + \operatorname{diag}(\sigma_n^2(\boldsymbol{\theta}_1^*), \ldots, \sigma_n^2(\boldsymbol{\theta}_b^*))]^{-1}c_t(\boldsymbol{\theta}^*, \boldsymbol{\theta}) && \text{(A.22)} \\ &= \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*). && \text{(A.23)}\end{aligned}$$

(ii) This claim follows from (iv) since the rightmost term of (24) is clearly non-negative.
(iii) Straightforward computations show that

$$\begin{aligned}\tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}^*) &= \begin{bmatrix} c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}) \\ c_t(\boldsymbol{\theta}_2^*, \boldsymbol{\theta}) \end{bmatrix}^{\top} \begin{bmatrix} c_t(\boldsymbol{\theta}_1^*) + \sigma_n^2(\boldsymbol{\theta}_1^*) & c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*) \\ c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*) & c_t(\boldsymbol{\theta}_2^*) + \sigma_n^2(\boldsymbol{\theta}_2^*) \end{bmatrix}^{-1} \begin{bmatrix} c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}) \\ c_t(\boldsymbol{\theta}_2^*, \boldsymbol{\theta}) \end{bmatrix} && \text{(A.24)} \\ &= \frac{1}{\bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*) - c_t^2(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)} \begin{bmatrix} c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}) \\ c_t(\boldsymbol{\theta}_2^*, \boldsymbol{\theta}) \end{bmatrix}^{\top} \begin{bmatrix} \bar{s}_t^2(\boldsymbol{\theta}_2^*) & -c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*) \\ -c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*) & \bar{s}_t^2(\boldsymbol{\theta}_1^*) \end{bmatrix} \begin{bmatrix} c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}) \\ c_t(\boldsymbol{\theta}_2^*, \boldsymbol{\theta}) \end{bmatrix} && \text{(A.25)} \\ &= \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_1^*) + \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_2^*) - \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_1^*) - \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_2^*) \\ &\qquad + \frac{c_t^2(\boldsymbol{\theta}, \boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*) + c_t^2(\boldsymbol{\theta}, \boldsymbol{\theta}_2^*)\bar{s}_t^2(\boldsymbol{\theta}_1^*) - 2c_t(\boldsymbol{\theta}, \boldsymbol{\theta}_1^*)c_t(\boldsymbol{\theta}, \boldsymbol{\theta}_2^*)c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)}{\bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*) - c_t^2(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)}, && \text{(A.26)} \\ &= \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_1^*) + \tau_t^2(\boldsymbol{\theta}; \boldsymbol{\theta}_2^*) + \frac{c_t^2(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta}, \boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*) + c_t^2(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta}, \boldsymbol{\theta}_2^*)\bar{s}_t^2(\boldsymbol{\theta}_1^*)}{\bar{s}_t^4(\boldsymbol{\theta}_1^*)\bar{s}_t^4(\boldsymbol{\theta}_2^*) - \bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)} \\ &\qquad - 2\frac{c_t(\boldsymbol{\theta}, \boldsymbol{\theta}_1^*)c_t(\boldsymbol{\theta}, \boldsymbol{\theta}_2^*)c_t(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)\bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*)}{\bar{s}_t^4(\boldsymbol{\theta}_1^*)\bar{s}_t^4(\boldsymbol{\theta}_2^*) - \bar{s}_t^2(\boldsymbol{\theta}_1^*)\bar{s}_t^2(\boldsymbol{\theta}_2^*)c_t^2(\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*)} && \text{(A.27)}\end{aligned}$$

$$= \tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}_1^*) + \tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}_2^*) + r_t(\boldsymbol{\theta};\boldsymbol{\theta}_1^*,\boldsymbol{\theta}_2^*). \tag{A.28}$$

(iv) The result follows immediately by applying Lemma A.1 to an equation corresponding to (A.24) in the proof of (iii) but where one has the matrix $\bar{\mathbf{S}}_A$ in place of the scalar $c_t(\boldsymbol{\theta}_1^*) + \sigma_n^2(\boldsymbol{\theta}_1^*)$. □

*Proof of Proposition 5.3.* We compute

$$\begin{aligned}
L_t^{\mathrm{v}}(\boldsymbol{\theta}^*) &= \mathbb{E}_{\mathbf{y}^*\,|\,\boldsymbol{\theta}^*,D_t} \int_\Theta \pi^2(\boldsymbol{\theta}) e^{2m_{t+b}(\boldsymbol{\theta};\mathbf{y}^*,\boldsymbol{\theta}^*)+s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} \left(e^{s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} - 1\right) \mathrm{d}\boldsymbol{\theta} && \text{(A.29)}\\
&= \int_\Theta \pi^2(\boldsymbol{\theta}) \mathbb{E}_{\mathbf{y}^*\,|\,\boldsymbol{\theta}^*,D_t} \left(e^{2m_{t+b}(\boldsymbol{\theta};\mathbf{y}^*,\boldsymbol{\theta}^*)+s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} \left(e^{s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} - 1\right)\right) \mathrm{d}\boldsymbol{\theta} && \text{(A.30)}\\
&= \int_\Theta \pi^2(\boldsymbol{\theta}) \underbrace{\mathbb{E}_{m_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*)\,|\,\boldsymbol{\theta}^*,D_t} \left(e^{2m_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*)}\right)}_{=e^{2m_t(\boldsymbol{\theta})+2\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)}} e^{s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} \left(e^{s_{t+b}^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} - 1\right) \mathrm{d}\boldsymbol{\theta} && \text{(A.31)}\\
&= \int_\Theta \pi^2(\boldsymbol{\theta}) e^{2m_t(\boldsymbol{\theta})+s_t^2(\boldsymbol{\theta})} \left(e^{s_t^2(\boldsymbol{\theta})} - e^{\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)}\right) \mathrm{d}\boldsymbol{\theta}, && \text{(A.32)}
\end{aligned}$$

where on the second line we have used Tonelli theorem to change the order of expectation and integration, on the third line we have used Lemma 5.1, and the expectation on the third line is computed using the moment-generating function of the Gaussian distribution. □

*Proof of Proposition 5.4.* Using Lemma 5.1, we see that the pointwise median in the integrand of (30) is computed as $\mathrm{med}_{\mathbf{y}^*\,|\,\boldsymbol{\theta}^*,D_t}\left(e^{m_{t+b}(\boldsymbol{\theta};\mathbf{y}^*,\boldsymbol{\theta}^*)}\right) = \mathrm{med}_{m_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*)\,|\,\boldsymbol{\theta}^*,D_t}\left(e^{m_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*)}\right) = e^{m_t(\boldsymbol{\theta})}$ from which the final result follows. □

# B Analysis of the greedy optimisation of design criteria

## B.1 On the monotonicity of design criteria

We show that EIV and IMIQR design criteria are non-increasing functions of the batch size $b$. We also discuss why this does not generally hold for expected integrated IQR (abbreviated EIIQR) which further justifies our choice of IMIQR over EIIQR.

Suppose that $\boldsymbol{\theta}_A^* \subseteq \boldsymbol{\theta}_B^*$ where we allow $\boldsymbol{\theta}_A^*$ to be an empty multiset so that $\tau_t^2(\boldsymbol{\theta};\emptyset) = 0$. Now using Lemma 5.2, we see that EIV is non-increasing because

$$L_t^{\mathrm{v}}(\boldsymbol{\theta}_A^*) - L_t^{\mathrm{v}}(\boldsymbol{\theta}_B^*) = \int_\Theta \pi^2(\boldsymbol{\theta}) e^{2m_t(\boldsymbol{\theta})+s_t^2(\boldsymbol{\theta})} \underbrace{\left[e^{\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}_B^*)} - e^{\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}_A^*)}\right]}_{\geq 0} \mathrm{d}\boldsymbol{\theta} \geq 0. \tag{B.1}$$

Similarly, using the fact that $z \mapsto \sinh(z)$ is an increasing function and recalling that $u = \Phi^{-1}(p_u) > 0$, we see that IMIQR is non-increasing:

$$\tilde{L}_t^{\mathrm{IQR}}(\boldsymbol{\theta}_A^*) - \tilde{L}_t^{\mathrm{IQR}}(\boldsymbol{\theta}_B^*) = 2\int_\Theta \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} \underbrace{[\sinh(u s_{t+b_A}(\boldsymbol{\theta};\boldsymbol{\theta}_A^*)) - \sinh(u s_{t+b_B}(\boldsymbol{\theta};\boldsymbol{\theta}_B^*))]}_{\geq 0} \mathrm{d}\boldsymbol{\theta} \geq 0. \tag{B.2}$$

We next analyse the EIIQR strategy. The design criterion for EIIQR, denoted $L_t^{\mathrm{IQR,e}}$, is given by

$$L_t^{\mathrm{IQR,e}}(\boldsymbol{\theta}^*) = 2\int_\Theta \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})+\frac{1}{2}\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)} \sinh(u s_{t+b}(\boldsymbol{\theta};\boldsymbol{\theta}^*))\, \mathrm{d}\boldsymbol{\theta}. \tag{B.3}$$

The proof of this fact is analogous to that of the Proposition 5.3 and details are thus omitted. As compared to IMIQR, the extra term $\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*)/2$ appears to the integrand so that $\tilde{L}_t^{\mathrm{IQR}}(\boldsymbol{\theta}^*) \leq L_t^{\mathrm{IQR,e}}(\boldsymbol{\theta}^*)$ holds for all $\boldsymbol{\theta}^*$.

We now briefly analyse EIIQR which is not non-increasing for $b$ in general. Presenting an explicit counterexample is not straightforward so we only heuristically justify why the integrand of (B.4) can be negative. It is enough to consider the special case where $\boldsymbol{\theta}_A^* = \emptyset$. We obtain

$$L_t^{\mathrm{IQR,e}}(\boldsymbol{\theta}_A^*) - L_t^{\mathrm{IQR,e}}(\boldsymbol{\theta}_B^*) = 2\int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} \underbrace{\Big(\sinh(us_t(\boldsymbol{\theta})) - e^{\frac{1}{2}\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}_B^*)} \sinh(us_{t+b_B}(\boldsymbol{\theta};\boldsymbol{\theta}_B^*))\Big)}_{\triangleq \omega(\boldsymbol{\theta};\boldsymbol{\theta}_B^*)} \,\mathrm{d}\boldsymbol{\theta}. \tag{B.4}$$

Suppose $\sigma_n^2(\boldsymbol{\theta}) > 0$ for all $\boldsymbol{\theta} \in \Theta$ and consider $\boldsymbol{\theta} \in \Theta$ so that $s_t(\boldsymbol{\theta}) > 0$ and $\pi(\boldsymbol{\theta}) > 0$. Then one can take $\boldsymbol{\theta}^* \in \Theta$ so that there exists $c = c(\boldsymbol{\theta}, \boldsymbol{\theta}^*) \in (0,1)$ and $\tau_t^2(\boldsymbol{\theta};\boldsymbol{\theta}^*) = cs_t^2(\boldsymbol{\theta})$. We can now write

$$\omega(\boldsymbol{\theta};\boldsymbol{\theta}^*) = \sinh(us_t(\boldsymbol{\theta})) - e^{\frac{c}{2}s_t^2(\boldsymbol{\theta})} \sinh(us_t(\boldsymbol{\theta})\sqrt{1-c}). \tag{B.5}$$

If $c$ can be kept fixed but $s_t(\boldsymbol{\theta}) \to \infty$, then we can see that $\omega(\boldsymbol{\theta};\boldsymbol{\theta}^*) \to -\infty$. That is, if $s_t(\boldsymbol{\theta})$ is chosen large enough, then $\omega(\boldsymbol{\theta};\boldsymbol{\theta}^*) < 0$. It can be further reasoned by continuity that $\omega(\boldsymbol{\theta};\boldsymbol{\theta}^*) < 0$ holds in set of nonzero measure around $\boldsymbol{\theta}$. Simulations suggest that there indeed exists practical scenarios where some choices of $\boldsymbol{\theta}^*$ make the expected loss to increase, that is, (B.4) is negative. In these cases EIIQR algorithm can get stuck to "safe" regions of the parameter space because evaluations elsewhere would increase the expected loss. This behaviour produces poor posterior estimates in practice and implies possibly non-convergence of the inference algorithm.

## B.2 Proof of the greedy optimisation bound

*Proof of Theorem 5.5.* The main idea is to first show that $\tilde{U}_t^{\mathrm{IQR},a}$ in (33) is a weakly submodular set function (see e.g. Krause et al. [2008], Krause and Cevher [2010] for definition) and then derive the bound using similar reasoning as in Nemhauser et al. [1978] and the observation that (weak) submodularity in their proof is required only for sets with size up to $2b$ instead for all sets. In the following we drop abbreviation "IQR, $a$" from $\tilde{U}_t^{\mathrm{IQR},a}$. Let $\boldsymbol{\theta}_A \subset \tilde{\Theta}, i \triangleq |\boldsymbol{\theta}_A| \leq 2b$ and $\boldsymbol{\theta}_j, \boldsymbol{\theta}_k \in \tilde{\Theta} \setminus \boldsymbol{\theta}_A$. We identify singletons with the corresponding element, that is, we write e.g. $\boldsymbol{\theta}_j$ for $\{\boldsymbol{\theta}_j\}$. Then

$$\begin{aligned}
&\tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_A) - \tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_j \cup \boldsymbol{\theta}_k) + \tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_j) \\
&= 2u^2 \int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} [s_{t+i}^2(\boldsymbol{\theta};\boldsymbol{\theta}_A) + s_{t+i+2}^2(\boldsymbol{\theta};\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_j \cup \boldsymbol{\theta}_k) \\
&\qquad - s_{t+i+1}^2(\boldsymbol{\theta};\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_j) - s_{t+i+1}^2(\boldsymbol{\theta};\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_k)] \,\mathrm{d}\boldsymbol{\theta} &\text{(B.6)}\\
&= 2u^2 \int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} [\tau_{t+i}^2(\boldsymbol{\theta};\boldsymbol{\theta}_j) + \tau_{t+i}^2(\boldsymbol{\theta};\boldsymbol{\theta}_k) - \tau_{t+i}^2(\boldsymbol{\theta};\boldsymbol{\theta}_j \cup \boldsymbol{\theta}_k)] \,\mathrm{d}\boldsymbol{\theta} &\text{(B.7)}\\
&= -2u^2 \int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} r_{t+i}(\boldsymbol{\theta};\boldsymbol{\theta}_j, \boldsymbol{\theta}_k) \,\mathrm{d}\boldsymbol{\theta} &\text{(B.8)}\\
&\geq -2u^2 \max_{\substack{\boldsymbol{\theta}_A \subset \tilde{\Theta}, |\boldsymbol{\theta}_A| = i \leq 2b \\ \boldsymbol{\theta}_j, \boldsymbol{\theta}_k \in \tilde{\Theta}}} \int_{\Theta} \pi(\boldsymbol{\theta}) e^{m_t(\boldsymbol{\theta})} r_{t+i}(\boldsymbol{\theta};\boldsymbol{\theta}_j, \boldsymbol{\theta}_k) \,\mathrm{d}\boldsymbol{\theta} &\text{(B.9)}\\
&\geq -\max\{0, 2u^2 \varepsilon_t'\} &\text{(B.10)}\\
&= -\varepsilon_t. &\text{(B.11)}
\end{aligned}$$

We have thus shown that

$$\tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_A) \geq \tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_j \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_j) - \varepsilon_t. \tag{B.12}$$

Now assume $|\boldsymbol{\theta}_A| \leq b$ and consider set $\boldsymbol{\theta}_B = \boldsymbol{\theta}_A \cup \{\boldsymbol{\theta}_1^B, \ldots, \boldsymbol{\theta}_b^B\} \subset \tilde{\Theta}$. Assume also $\boldsymbol{\theta}_k \notin \boldsymbol{\theta}_B$. If we replace $\boldsymbol{\theta}_A$ with $\boldsymbol{\theta}_A \cup \{\boldsymbol{\theta}_1^B, \ldots, \boldsymbol{\theta}_{m-1}^B\}$ and set $\boldsymbol{\theta}_j = \boldsymbol{\theta}_m^B$ in the above formula, we obtain

$$\begin{aligned}
&\tilde{U}_t(\boldsymbol{\theta}_A \cup \{\boldsymbol{\theta}_1^B, \ldots, \boldsymbol{\theta}_{m-1}^B\} \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_A \cup \{\boldsymbol{\theta}_1^B, \ldots, \boldsymbol{\theta}_{m-1}^B\}) \\
&\qquad \geq \tilde{U}_t(\boldsymbol{\theta}_A \cup \{\boldsymbol{\theta}_1^B, \ldots, \boldsymbol{\theta}_m^B\} \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_A \cup \{\boldsymbol{\theta}_1^B, \ldots, \boldsymbol{\theta}_m^B\}) - \varepsilon_t.
\end{aligned} \tag{B.13}$$

If we sum up all the above inequalities for $m = 1, \dots, b$, we obtain

$$\tilde{U}_t(\boldsymbol{\theta}_A \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_A) \geq \tilde{U}_t(\boldsymbol{\theta}_B \cup \boldsymbol{\theta}_k) - \tilde{U}_t(\boldsymbol{\theta}_B) - b\varepsilon_t. \tag{B.14}$$

Next we proceed similarly as the proof of Proposition 11.1 in Bach [2013] but use our weak submodularity condition in (B.14). $\tilde{U}_t$ is clearly bounded and non-decreasing and $\tilde{U}_t(\emptyset) = 0$. Let $\boldsymbol{\theta}_j^G, j = 1, \dots, b$ be the $j$th element selected during the greedy algorithm, $\boldsymbol{\theta}_{1:j}^G \triangleq \{\boldsymbol{\theta}_1^G, \dots, \boldsymbol{\theta}_j^G\}$ and $\rho_j \triangleq \tilde{U}_t(\boldsymbol{\theta}_{1:j}^G) - \tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G)$. For a given $j$, denote $\boldsymbol{\theta}_O \backslash \boldsymbol{\theta}_{1:j}^G = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_m\}$ so that $m \leq b$. Then

$$\begin{aligned}
&\tilde{U}_t(\boldsymbol{\theta}_O) \\
&\leq \tilde{U}_t(\boldsymbol{\theta}_O \cup \boldsymbol{\theta}_{1:j-1}^G) && \text{(B.15)} \\
&= \tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G) + \sum_{i=1}^{m} [\tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G \cup \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_i\}) - \tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G \cup \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_{i-1}\})] && \text{(B.16)} \\
&\leq \tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G) + \sum_{i=1}^{m} [\tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G \cup \boldsymbol{\theta}_i) - \tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G) + b\varepsilon_t] && \text{(B.17)} \\
&\leq \tilde{U}_t(\boldsymbol{\theta}_{1:j-1}^G) + m\rho_j + mb\varepsilon_t && \text{(B.18)} \\
&\leq \sum_{i=1}^{j-1} \rho_i + b\rho_j + b^2\varepsilon_t. && \text{(B.19)}
\end{aligned}$$

If we multiply both sides of the inequality $\sum_{i=1}^{j-1} \rho_i + b\rho_j \geq \tilde{U}_t(\boldsymbol{\theta}_O) - b^2\varepsilon_t$ by $(1 - 1/b)^{b-j}$ and add the inequalities up for $j = 1, \dots, b$, we obtain[1]

$$\sum_{i=1}^{b} (1 - 1/b)^{b-i} \left( \sum_{j=1}^{i-1} \rho_j + b\rho_i \right) \geq (\tilde{U}_t(\boldsymbol{\theta}_O) - b^2\varepsilon_t) \sum_{i=0}^{b-1} (1 - 1/b)^i. \tag{B.20}$$

After some simplifications, we see that this inequality is equivalent with

$$\tilde{U}_t(\boldsymbol{\theta}_G) = \sum_{i=1}^{b} \rho_i \geq (1 - (1 - 1/b)^b)(\tilde{U}_t(\boldsymbol{\theta}_O) - b^2\varepsilon_t), \tag{B.21}$$

from which the claim follows. □

# C Additional implementation details

In this section we briefly present additional implementation details of the algorithm in Section 5.4. We start by pointing out that the initial design locations are drawn from the prior $\pi(\boldsymbol{\theta})$ but other techniques such as random or quasi-Monte Carlo designs over $\Theta$ are also possible.

Different estimators of log-SL can be used in our algorithm. In fact, the logarithm of (2) with plug-in ML estimators in (3) produces a biased estimator of the logarithm of the Normal pdf so it might be reasonable to use an unbiased estimator instead. Such an estimator exists and has been used in Ong et al. [2018]. However, we nevertheless used the logarithm of (2) with plug-in ML estimates because the resulting estimator was found slightly more robust than the unbiased estimator in Ong et al. [2018] and because both estimators produced similar results in practice. While a systematic comparison of different log-SL estimators was not done, we expect the bias to be small with moderate $N$. Also, in practice, the Gaussianity assumption

[1] This last part of our proof is analogous to that of theorem 7 in Krause et al. [2008] and we have corrected the mistake of having $(1 - 1/k)^{k-1}$ where it should read as $(1 - 1/k)^{k-i}$ (in their notation where $k$ corresponds our batch size $b$ and $i$ corresponds our index $j$).

usually holds only approximately. As mentioned in the main text, there also exists an unbiased SL estimator. However, since we are modelling the logarithm of SL, using an unbiased estimator of SL is not advantageous.

To handle the unknown GP hyperparameters $\boldsymbol{\phi}$, a plug-in approach is often used where $\boldsymbol{\phi}$ is substituted with ML or MAP estimate, see e.g. Rasmussen and Williams [2006]. For fully Bayesian approach, one can use MCMC methods but this is expensive. Uncertainty in $\boldsymbol{\phi}$ could be acknowledged also when computing the design criterion as discussed e.g. in the Section 3.5 of Järvenpää et al. [2019]. However, we used the plug-in approach with MAP estimate in our experiments and we re-estimated $\boldsymbol{\phi}$ after each iteration as shown on line 7 of Algorithm 1 using the `gp_optim` function of GPstuff 4.7 [Vanhatalo et al., 2013].

A relatively tuning-free adaptive MCMC method by Haario et al. [2006] is used for sampling from $\pi_q$ (and from the posterior estimate on line 25 of Algorithm 1). However, because the IS proposal can be multimodal, the sampler may get stuck to a local mode. To alleviate this, we use multiple chains and initialise the sampler at the point with the highest current loss over the training points. This way, even if sampling over the full range of $\Theta$ is not perfect, the loss measures uncertainty in regions where it is relatively large and, consequently, reasonable designs are obtained. Furthermore, we found that computing this integral very accurately in not necessary for obtaining good designs and ultimately converging to a good posterior approximation.

Several methods for the global optimisation of the design criterion have been used in literature: random search, multistart gradient-based methods, evolution strategies (such as CMA-ES) and partitioning based algorithm DIRECT. Here the optimisation is carried out by first using random search to roughly locate potential optima and then improving the best 10 points found this way by initialising gradient-based algorithm (MATLAB function `fmincon`) at these points. Finally, the best point evaluated is reported as the solution. While systematic comparison between optimisers was not done, we observed that this approach produced satisfactory results with reasonable computation time.

We also precompute many quantities in the GP and design criterion formulas to speed-up the optimisation and sampling steps. For example, the Cholesky factor of the full data covariance matrix $\mathbf{K}_t$ in (7) and (8) is precomputed and used for prediction at new locations $\boldsymbol{\theta}$. For EIV or IMIQR, all the quantities depending only on the integration points $\boldsymbol{\theta}^{(j)}$ are precomputed so that only those terms that depend on candidate design points $\boldsymbol{\theta}^*$ need to be re-evaluated during the optimisation. Furthermore, in the greedy optimisation, the Lemma 5.2 (iv) is used to avoid always inverting the whole covariance matrix in (20) when only the last column $\boldsymbol{\theta}^*_r$ of the matrix $\boldsymbol{\theta}^*_{1:r}$ is varied. When computing the design criteria, we use logarithms and the logsumexp trick to avoid numerical under- and overflows which otherwise occur when exponentiating the GP mean and variance function values with high magnitude.

While we consider a fixed budget of simulations $b_0 + i_{\max} b$, the algorithm can be prematurely terminated when some suitable stopping criterion is met. For example, if the SL posterior estimate or the value of the design criterion has changed little during a fixed number of the most recent iterations, one could terminate the algorithm. Such stopping criteria have been used by Acerbi [2018], Wang and Li [2018].

# D Additional experiments and illustrations

## D.1 Normality of noisy log-SL evaluations

Figure D.1 shows the sampling distribution of the log-SL for six benchmark models.

## D.2 Details of the toy densities

The 2D toy densities are shown in Figure D.2. Their log-densities, denoted here as $f_{2\text{D}}$, are defined explicitly so that the log-density for the 'Simple' model is obtained as $f_{2\text{D}}(\boldsymbol{\theta}) = -\boldsymbol{\theta}^\top \mathbf{S}_\rho^{-1}\boldsymbol{\theta}/2$ where $\rho = 0.25$, for the 'Banana' model as $f_{2\text{D}}(\boldsymbol{\theta}) = -[\theta_1, \theta_2 + \theta_1^2 + 1]\mathbf{S}_\rho^{-1}[\theta_1, \theta_2 + \theta_1^2 + 1]^\top/2$ where $\rho = 0.9$ and, finally, for the 'Bimodal' model as $f_{2\text{D}}(\boldsymbol{\theta}) = -[\theta_1, \theta_2^2 - 2]\mathbf{S}_\rho^{-1}[\theta_1, \theta_2^2 - 2]^\top/2$ where $\rho = 0.5$. In all of these cases we have $(S_\rho)_{11} = (S_\rho)_{22} = 1$ and $(S_\rho)_{12} = (S_\rho)_{21} = \rho$. The prior densities for 'Simple', 'Banana' and 'Bimodal'

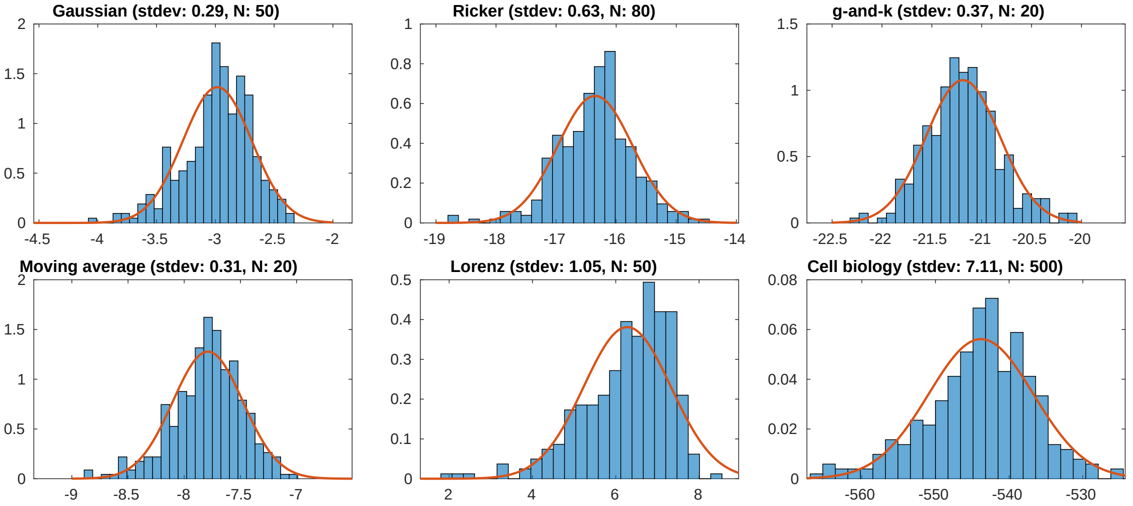

Figure D.1: Sampling distribution of log-SL for six benchmark models evaluated at their "true" parameter values. The densities are approximately Gaussian, although the distribution corresponding to the Lorenz model shows some evidence of skewness. The details of the Ricker, g-and-k and Lorenz model are given in Sections 6.2 and D.5, "Gaussian" is a simple 2D problem where the expectation is estimated, "Moving average" is the benchmark model used in Marin et al. [2012], and the cell biology model is as in Price et al. [2018].

test cases are chosen as $\pi_{\mathrm{2D}}(\boldsymbol{\theta}) = \mathcal{U}([-16,16]^2)$, $\pi_{\mathrm{2D}}(\boldsymbol{\theta}) = \mathcal{U}([-6,6] \times [-20,2])$ and $\pi_{\mathrm{2D}}(\boldsymbol{\theta}) = \mathcal{U}([-6,6]^2)$, respectively.

The 6D log-densities, denoted here as $f_{\mathrm{6D}}$, are then constructed from the 2D log-densities so that $f_{\mathrm{6D}}(\boldsymbol{\theta}) = f_{\mathrm{2D}}(\boldsymbol{\theta}_{1:2}) + f_{\mathrm{2D}}(\boldsymbol{\theta}_{3:4}) + f_{\mathrm{2D}}(\boldsymbol{\theta}_{5:6})$. The corresponding priors are chosen as $\pi_{\mathrm{6D}}(\boldsymbol{\theta}) = \pi_{\mathrm{2D}}(\boldsymbol{\theta}_{1:2})\pi_{\mathrm{2D}}(\boldsymbol{\theta}_{3:4})\pi_{\mathrm{2D}}(\boldsymbol{\theta}_{5:6})$. That is, the priors for the 6D 'Simple', 'Banana' and 'Multimodal' densities are $\mathcal{U}([-16,16]^6)$, $\mathcal{U}(\bigtimes_{i=1}^{3}([-6,6] \times [-20,2]))$ and $\mathcal{U}([-6,6]^6)$, respectively.

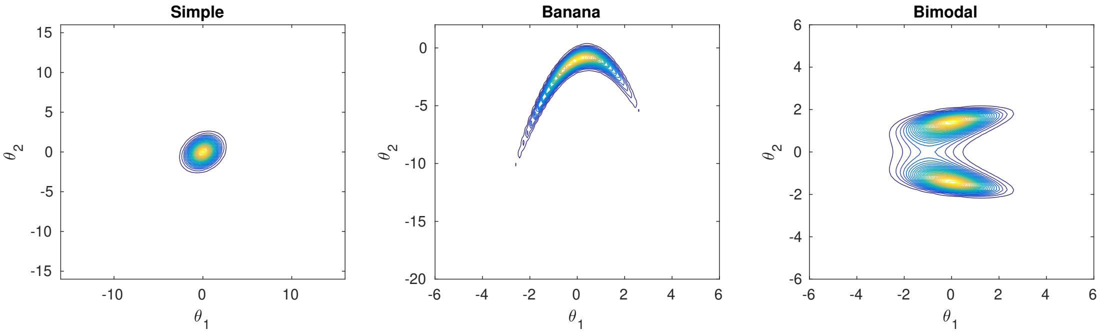

Figure D.2: Illustration of the 2D toy densities which coincide with the 2D marginals of $\boldsymbol{\theta}_{1:2}, \boldsymbol{\theta}_{3:4}$ and $\boldsymbol{\theta}_{5:6}$ of the corresponding 6D toy densities.

## D.3 Additional experiments and illustration with 2D toy densities

We first analyse how the noise level of the log-likelihood evaluations affects the estimation accuracy. We use our three 2D toy densities and corrupt their exact log-likelihoods with additive zero mean i.i.d. Gaussian noise with variance $\sigma_n^2 \in \{1^2, 2^2, 5^2\}$ to obtain "noisy evaluations". We use initial designs with $b_0 = 10$

points. The integrals in EIV and IMIQR are here numerically computed using a $50 \times 50$ grid. We use batch size $b = 4$ for all batch methods and we compare both the joint and greedy batch methods. The results are shown in Figure D.3. We see that the principled EIV and IMIQR methods have fairly similar overall performance and they clearly outperform the heuristic MAXV and MAXIQR methods. Unsurprisingly, the noisy evaluations require more iterations.

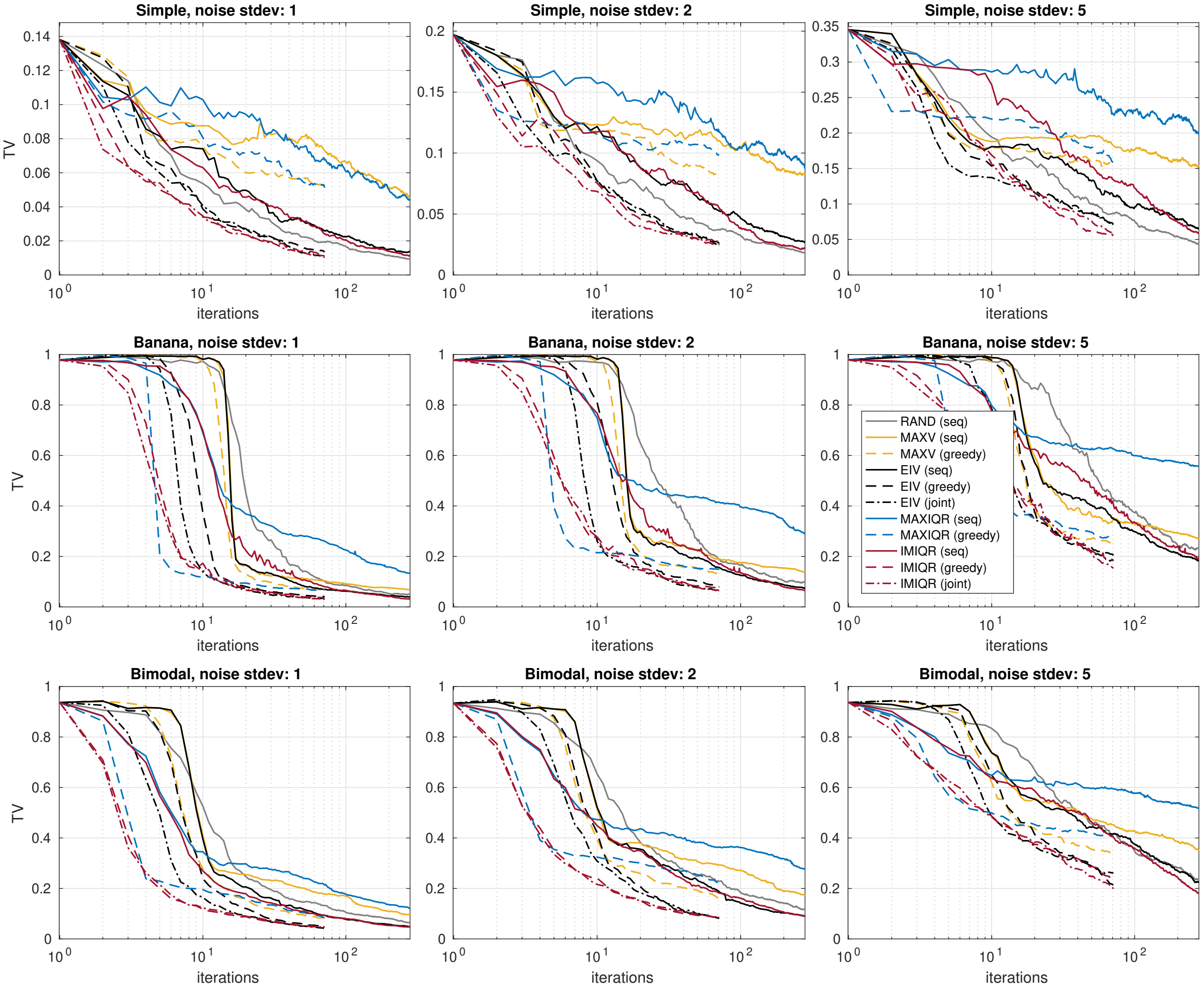

Figure D.3: Results for the 2D toy densities with various noise levels. The rows correspond to the test cases and the columns different noise levels of the log-likelihood evaluations. The lines show the average TV over 50 repeated simulations. Note that x-axis is on log-scale and the maximum number of evaluations is here $t = 290$. The batch size of all batch methods is $b = 4$.

Figure D.4 shows the design locations of the 2D Banana example for MAXV, EIV and MAXIQR. Figures D.5 and D.6 show the design locations for the 2D Bimodal example. The Bimodal example shows similar general observations as the Banana example. Figure D.5 shows that all IMIQR methods produce similar designs.

We also investigated how the accuracy of the greedy batch EIV and IMIQR methods scales as a function of the batch size using the three 2D toy densities. Figure D.7 shows these results. Figure D.8 further shows the same experimental results as Figure D.7 but plotted as a function of the total evaluations. That is, the results obtained when the batch-sequential strategies are used as if they were sequential strategies so that

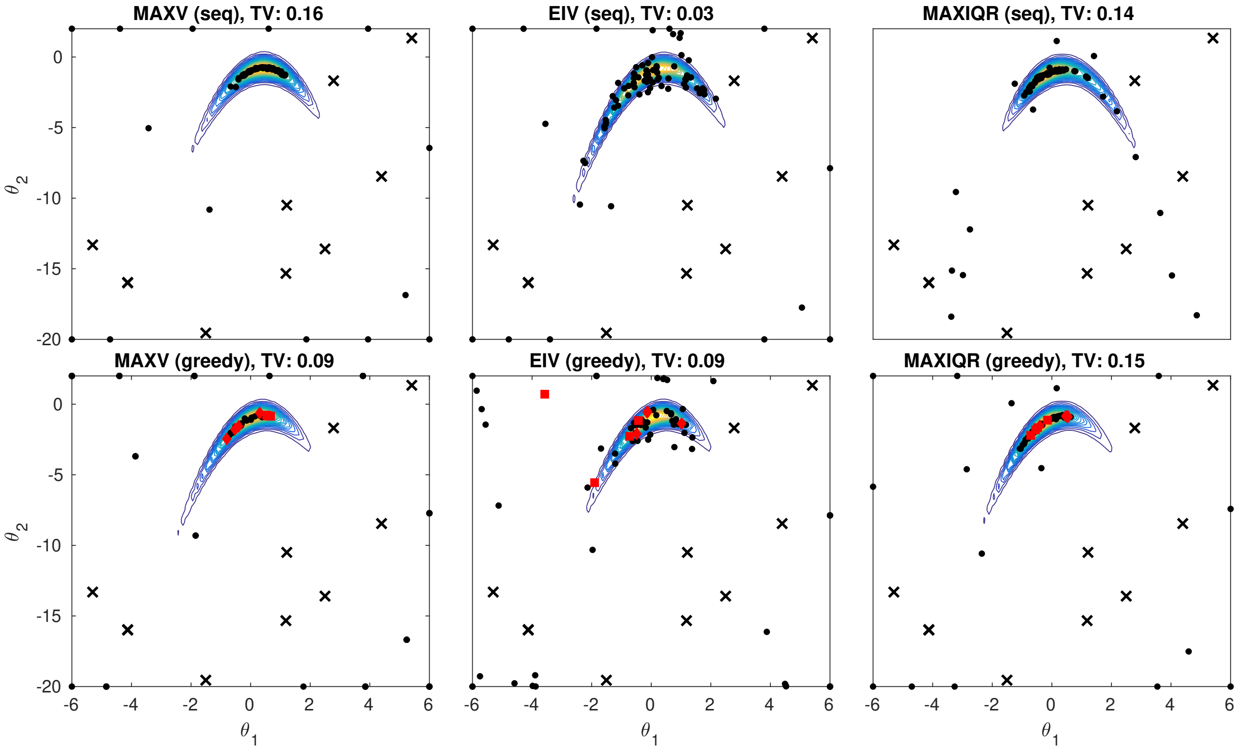

Figure D.4: Illustration of the designs of MAXV (left column), EIV (center), and MAXIQR (right). The results are shown after 90 noisy log-likelihood evaluations of the 2D Banana example with noise level $\sigma_n = 1$. The top row shows designs obtained with a sequential strategy ($b = 1$), and the bottom row from a greedy batch-sequential strategy ($b = 4$). The black crosses show the initial evaluations, black dots show obtained design points except for the last two batches, and the red squares and diamonds show the last two batches.

the computations are done in a sequential manner. This demonstrates the penalty of not being able to use the unknown outputs of the pending simulations when they would in fact be available. The main observation is that the IMIQR greedy batch strategy produces batches that better mimic the sequential decisions than the corresponding EIV batch strategy which shows as a faster convergence speed of IMIQR.

## D.4 Parallellisation of the synthetic likelihood method with IMIQR

We demonstrate that it is useful to parallellise our algorithm in the SL case with respect to $b$. This justifies our experiments in Section 6.2. We use the same set-up for Ricker and g-and-k models as in the main text and the set-up for Lorenz model is as in Section D.5. We consider a scenario where 1000 computer cores are available and we compare three different combinations of $N$ and $b$ that all use 1000 simulations per batch. In addition, we consider a baseline where $N = 100$ and $b = 1$. We use IMIQR design criterion.

The results in Figure D.9 show that $b = 10$ gives the fastest convergence speed. This is not surprising since, intuitively, obtaining 10 noisy log-likelihood evaluations gives more information on the shape of the posterior than a single, although less noisy, evaluation. Also, computing log-likelihood values accurately near the boundaries of the parameter space wastes computational resources because a noisy evaluation is often enough to effectively rule out such regions. A surprising aspect is that $N = 100$ produces faster convergence than the corresponding runs with $N = 1000$. The likely reason for this seemingly counterintuitive behaviour is that GP modelling becomes easier with the noisier evaluations which helps to faster locate the modal region. When $N = 1000$, the IMIQR produced slightly more evaluations near the boundary areas as compared to $N = 100$ leading to slower initial convergence speed. On the other hand, especially in the Lorenz case, the

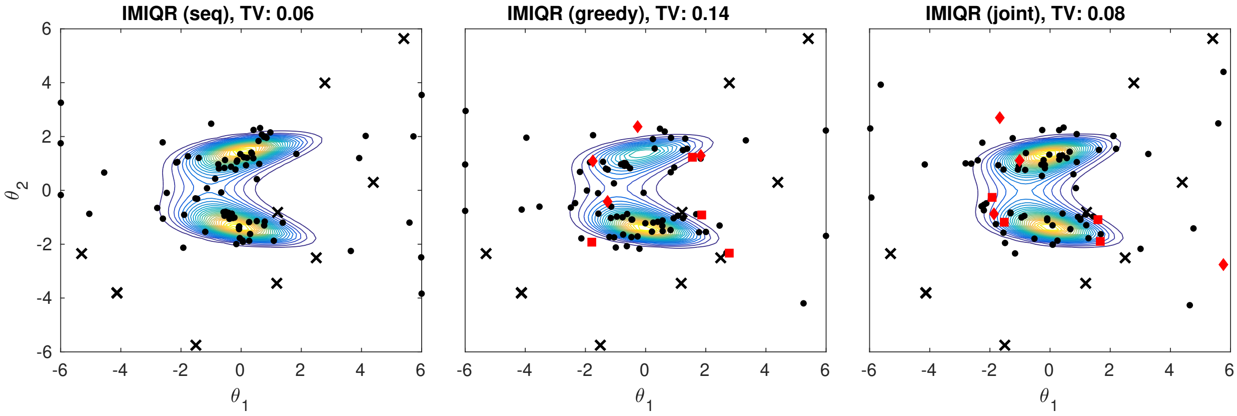

Figure D.5: Illustration of the design locations for the Bimodal example with IMIQR methods. See caption of Figure D.4 for details.

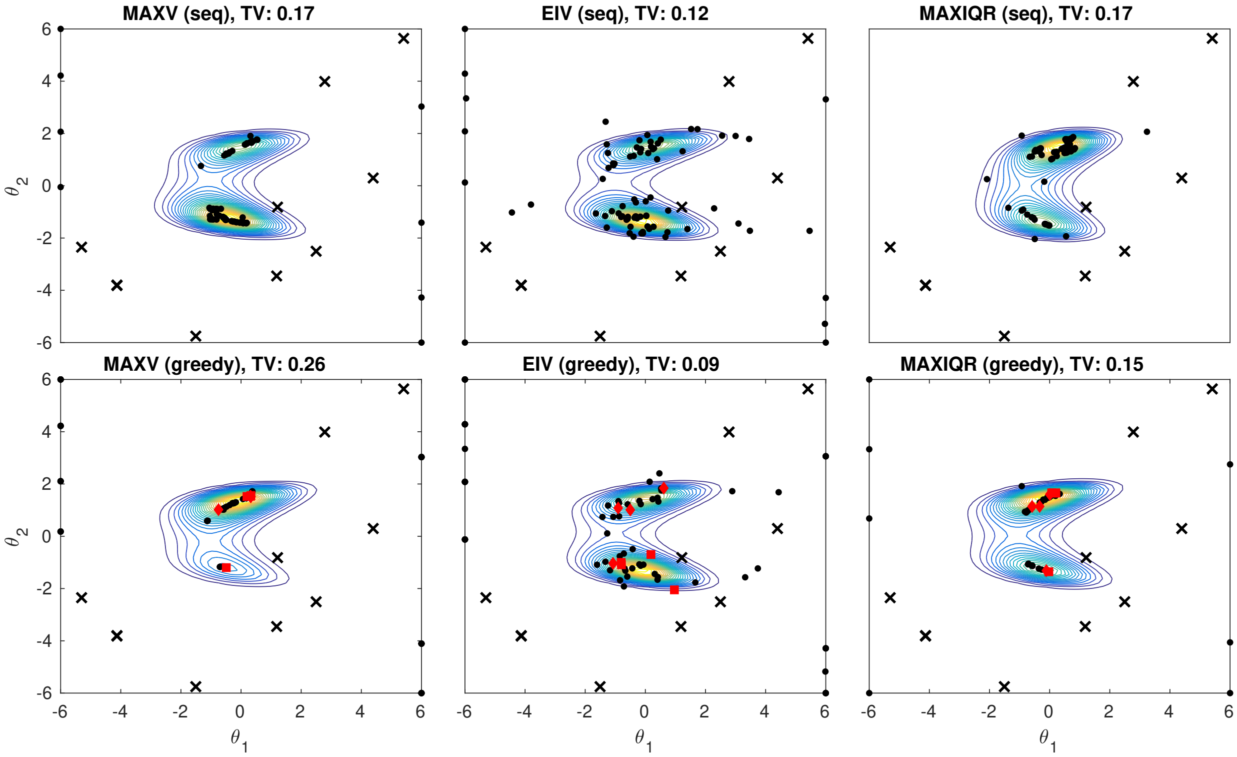

Figure D.6: Illustration of the design locations for the Bimodal example with MAXV, EIV and MAXIQR methods. See caption of Figure D.4 for details.

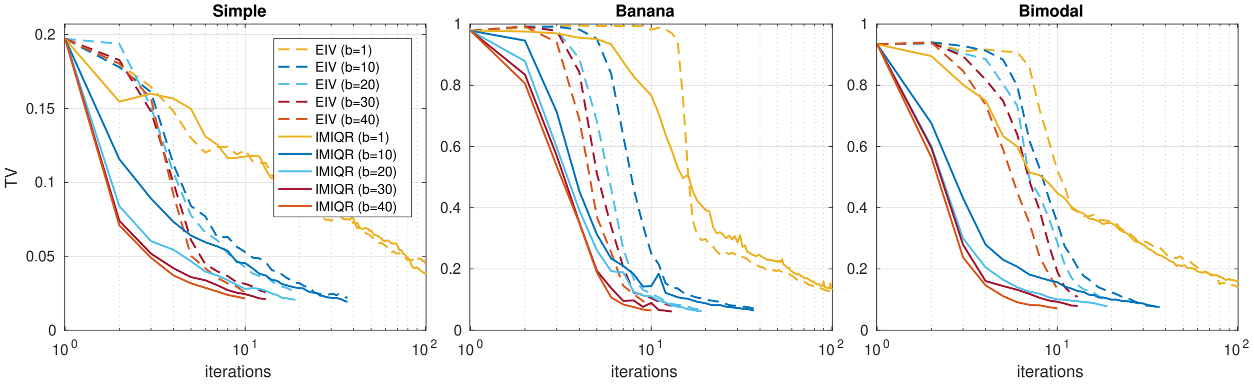

Figure D.7: Results with the greedy batch strategies with varying batch sizes $b$ for the 2D toy examples.

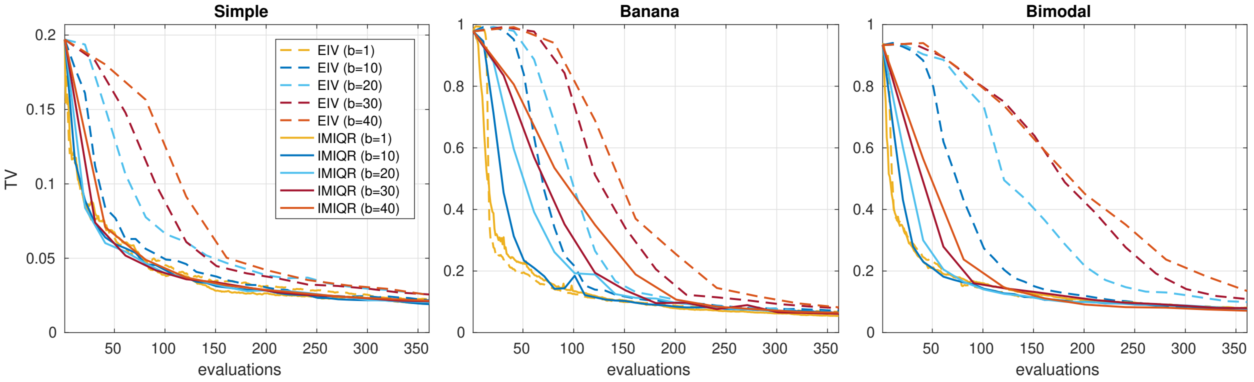

Figure D.8: Results with the greedy batch strategies with varying batch sizes $b$ for the 2D toy examples. The x-axis shows the total evaluations of the log-likelihood (and not the iterations in log-scale as in Figure D.7).

more accurate log-likelihood computations can be useful at the later iterations when the modal area of the posterior have been roughly located.

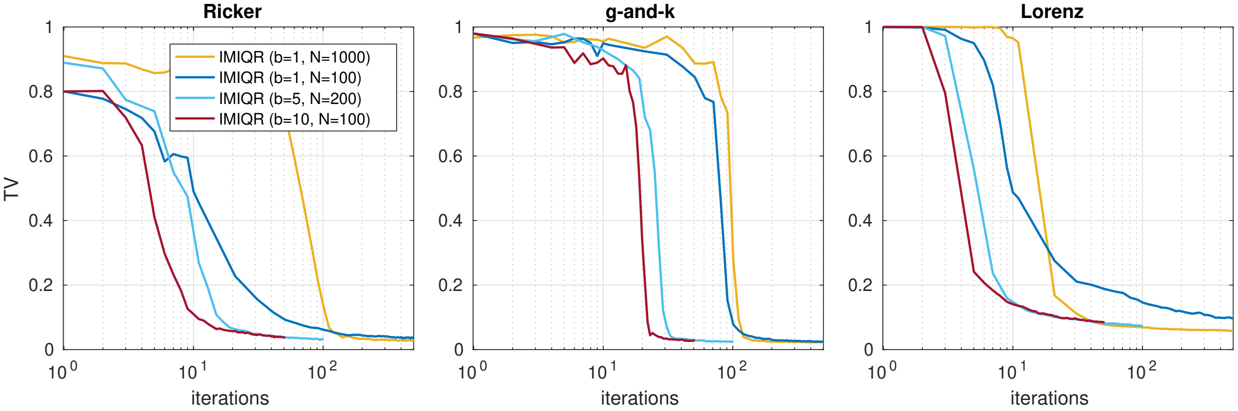

Figure D.9: The effect of the number of repeated simulations $N$ and the batch size $b$ on the convergence speed in the SL case. The details of the Lorenz model can be found in Section D.5.

## D.5 Experiments with Lorenz model

As an additional real-world test case, we consider a modified version of the well-known Lorenz weather model. We briefly describe this model below, for more details see Thomas et al. [2018] and the references therein. In the model it is assumed that weather stations measure a high-dimensional time-series of slow weather variables $x_k^{(t)}$, $k = 1, \ldots, 40$, whose dynamics is described by a coupled stochastic differential equation (SDE)

$$\frac{\mathrm{d}x_k^{(t)}}{\mathrm{d}t} = -x_{k-1}^{(t)}(x_{k-2}^{(t)} - x_{k+1}^{(t)}) - x_k^{(t)} + 10 - g(x_k^{(t)}, \boldsymbol{\theta}) + \eta_k^{(t)}, \tag{D.1}$$

for $k = 1, \ldots, 40$ and where $g(x_k^{(t)}, \boldsymbol{\theta}) = \theta_1 + \theta_2 x_k^{(t)}$ and where $x_k^{(t)}$ are cyclic so that $x_0^{(t)} = x_{40}^{(t)}$ and $x_{-1}^{(t)} = x_{39}^{(t)}$. The initial states of the weather variables $x_k^{(0)}$, $k = 1, \ldots, 40$ are assumed known and the time interval $t \in [0, 4]$ considered here corresponds to 20 days. The function $g$ in (D.1) models the net effect of the fast weather variables on the observable slow weather variables $x_k^{(t)}$, $k = 1, \ldots, 40$ and $\eta_k^{(t)}$ is a stochastic forcing term describing the uncertainty due to the forcing of the fast weather variables. As in Thomas et al. [2018], the time interval is discretised to 160 equidistant intervals producing time step $\Delta t = 0.025$ and the SDEs are then solved using 4th order Runge-Kutta method. In this discretised setting, the forcing term is assumed to follow the first-order autoregressive process

$$\eta_k^{(t+\Delta t)} = \phi \eta_k^{(t)} + \sqrt{1 - \phi^2} \varepsilon^{(t)} \tag{D.2}$$

for $k = 1, \ldots, 40$ and $t = 0, \Delta t, 2\Delta t, \ldots, 160\Delta t$, and where $\varepsilon^{(t)}$ are i.i.d. standard Gaussian, $\eta_k^{(0)} = \sqrt{1 - \phi^2} \varepsilon^{(0)}$ and $\phi = 0.4$.

We need to compute the posterior distribution of the parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$ given the slow weather variables $x_k^{(t)}$, $k = 1, \ldots, 40$ measured over 20 days. We use $\boldsymbol{\theta} \sim \mathcal{U}([0, 5] \times [0, 0.5])$ which is wider than in Thomas et al. [2018], to make the inference task more challenging, and the true parameter to generate the observed data is $\boldsymbol{\theta}_{\text{true}} = (2.0, 0.1)$. We use the six summary statistics suggested by Hakkarainen et al. [2012] and used by Thomas et al. [2018] to be in line with previous work although it was recently shown by Dinev and Gutmann [2018] that learning them from data can improve the estimation accuracy. We use $b_0 = 10$ and an additional budget of 400 SL evaluations with $N = 100$.

The results in Figure D.10 again show that the best approximations are obtained with IMIQR and its greedy batch variant with $b = 5$, which converges five times faster than the sequential IMIQR. This time MAXIQR, despite its exploitative behaviour, and its batch version with $b = 5$, both work reasonably well although they produce slightly worse and more variable posterior approximations than IMIQR. EIV and MAXV perform again very poorly because they mostly evaluate near the boundaries where the noise variance of log-likelihood is large although these evaluations are uninformative for estimating the likelihood in its modal area.

Overall, the TV values with Lorenz model are slighty worse than in the corresponding synthetic 2D examples although we use $N = 100$ so that $\sigma_n \lesssim 1$ in the modal area of the posterior. The probable reason is that the Gaussian assumption does not hold here as well as in other cases as suggested by Figure D.1. The third panel of Figure D.9 suggest that larger $N$ would lead to better approximation, likely because large $N$ makes the density of log-SL more peaked and also more Gaussian. Nevertheless, we conclude that the approximations obtained by IMIQR are reasonable and the convergence speed is fast since only a few hundred SL evaluations are needed.

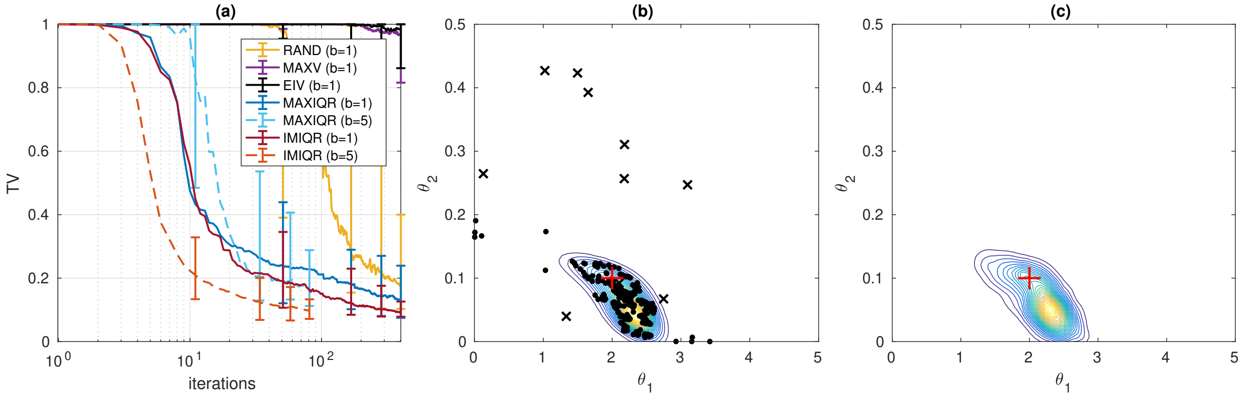

Figure D.10: Results for the Lorenz model. (a) Median TV and 90% variability over 50 repeated runs of the algorithms. (b) A typical example of the estimated posterior density obtained using the greedy batch IMIQR strategy. The black crosses show the initial design and the dots the design points. (c) Exact SL posterior computed using SL-MCMC for comparison. The red plus sign shows the true parameter.

## References

L. Acerbi. Variational Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems 31*, pages 8223–8233. 2018. 7

F. Bach. *Learning with Submodular Functions: A Convex Optimization Perspective*. Now Publishers Inc., Hanover, MA, USA, 2013. 6

T. Dinev and M. U. Gutmann. Dynamic Likelihood-free Inference via Ratio Estimation (DIRE). Available at `https://arxiv.org/pdf/1810.09899.pdf`, 2018. 13

H. Haario, M. Laine, A. Mira, and E. Saksman. Dram: Efficient adaptive mcmc. *Statistics and Computing*, 16(4):339–354, 2006. 7

J. Hakkarainen, A. Ilin, A. Solonen, M. Laine, H. Haario, J. Tamminen, E. Oja, and H. Järvinen. On closure parameter estimation in chaotic systems. *Nonlinear Processes in Geophysis*, 19:127–143, 2012. 13

M. Järvenpää, M. U. Gutmann, A. Pleska, A. Vehtari, and P. Marttinen. Efficient acquisition rules for model-based approximate bayesian computation. *Bayesian Analysis*, 14(2):595–622, 06 2019. 2, 7

A. Krause and V. Cevher. Submodular dictionary selection for sparse representation. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 567–574, 2010. 5

A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, 2008. 5, 6

J. M. Marin, P. Pudlo, C. P. Robert, and R. J. Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012. 8

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294, 1978. 5

V. M. H. Ong, D. J. Nott, M. N. Tran, S. A. Sisson, and C. C. Drovandi. Variational Bayes with synthetic likelihood. *Statistics and Computing*, 2018. 6

L. F. Price, C. C. Drovandi, A. Lee, and D. J. Nott. Bayesian synthetic likelihood. *Journal of Computational and Graphical Statistics*, 27(1):1–11, 2018. 8

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006. 7

O. Thomas, R. Dutta, J. Corander, S. Kaski, and M. U. Gutmann. Likelihood-free inference by ratio estimation. Available at `https://arxiv.org/abs/1611.10242`, 2018. 13

J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari. GPstuff: Bayesian modeling with Gaussian processes. *Journal of Machine Learning Research*, 14:1175–1179, 2013. 7

H. Wang and J. Li. Adaptive Gaussian process approximation for Bayesian inference with expensive likelihood functions. *Neural Computation*, 30(11):3072–3094, 2018. 7