
Multilevel Bayesian Quadrature

Kaiyu Li

University College London

Daniel Giles

University College London

Toni Karvonen

University of Helsinki

Serge Guillas

University College London
The Alan Turing Institute

François-Xavier Briol

University College London
The Alan Turing Institute

Abstract

Multilevel Monte Carlo is a key tool for approximating integrals involving expensive scientific models. The idea is to use approximations of the integrand to construct an estimator with improved accuracy over classical Monte Carlo. We propose to further enhance multilevel Monte Carlo through Bayesian surrogate models of the integrand, focusing on Gaussian process models and the associated Bayesian quadrature estimators. We show, using both theory and numerical experiments, that our approach can lead to significant improvements in accuracy when the integrand is expensive and smooth, and when the dimensionality is small or moderate. We conclude the paper with a case study illustrating the potential impact of our method in landslide-generated tsunami modelling, where the cost of each integrand evaluation is typically too large for operational settings.

1 INTRODUCTION

This paper considers the task of approximating an unknown integral, or expectation, when evaluations of the integrand are expensive, either from a computational or financial point of view. This is a common problem in statistics and machine learning, where one commonly needs to marginalise random variables, compute normalisation constants of probability density functions or compute posterior expectations. However the problem is even more pronounced when doing uncertainty quantification for large mathematical models in science and engineering. For example, a scientist might be uncertain about the value of certain model parameters,

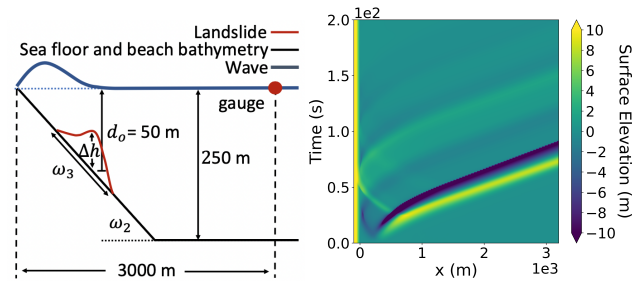


Figure 1: Tsunami Model. *Left*: Sketch of the submerged landslide-generated tsunami. *Right*: Solution of the differential equation through time and space.

and might therefore wish to estimate the expected value of some quantity of interest involving the model with respect to distributions on these parameters.

An example which illustrates this problem (later revisited in Section 6) is the modelling of landslide-generated tsunamis, where the evolution of the wave through space and time is described through a complex system of differential equations (Behrens and Dias, 2015; Reguly et al., 2018; Giles et al., 2020; Marras and Mandli, 2021); see Figure 1 for an illustration. In this context, designers of tsunami resistant buildings, prevention structures or early warning systems might be interested in estimating the total wave energy or momentum flux of the tsunami at a fixed location. These quantities are functions of the solution of the differential equations, but there will usually be some uncertainty associated with certain physical parameters, such as those characterising the slope or size of the landslide. This uncertainty is represented through probability distributions, leading to the need to compute the expected value of the quantities of interest. The main challenge is that in order to obtain high accuracy estimates, it is necessary to use very fine time and space meshes to solve the differential equations, leading to prohibitively large computational costs.

A common approach to the approximation of such integrals is Monte Carlo (MC) methods, which include a wide range

of simulation-based algorithms. Of particular relevance is *multilevel Monte Carlo* (MLMC) (Giles, 2015) and its various extensions (Giles and Waterhouse, 2009; Dick et al., 2016; Kuo et al., 2015, 2017). MLMC is designed for expensive integrands where cheap approximations are available at several levels of accuracy. Such models are called *multifidelity models* (Peherstorfer et al., 2018), and are widely used, including for atmospheric dispersion modelling (Katsiolides et al., 2018), biochemical reaction network modelling (Warne et al., 2019), reliability theory (Aslett et al., 2017), erosion and flood risk modelling (Clare et al., 2022), pricing in finance (Dempster et al., 2018), wind farm modelling (Kirby et al., 2023), the design of advanced aerospace vehicles (Geraci et al., 2017), or tsunami modelling (Sánchez-Linares et al., 2016).

MLMC evaluates the cheap but inaccurate approximation of the integrand a large number of times, and only evaluates the high-accuracy but expensive approximation of the integrand a small number of times. For the tsunami example above, standard MC would use a fine time and space mesh, and evaluate the integrand at fixed high accuracy level. In contrast, MLMC will use several approximations with different meshes (each corresponding to a level), and use fewer evaluations of the expensive levels. For a fixed computational budget, this allows MLMC to obtain much more accurate estimate than standard MC. Beyond the scientific application areas above, this has also led MLMC to be used to enhance computational tools including Markov chain Monte Carlo (Dodwell et al., 2019; Wang and Wang, 2022), particle filters (Gregory and Cotter, 2017), approximate Bayesian computation (Jasra et al., 2019), Bayesian experimental design (Goda et al., 2020) or variational inference (Shi and Cornish, 2021; Fujisawa and Sato, 2021).

Unfortunately, most multilevel methods suffer from the fact that they are simulation-based methods which neglect all known properties of the integrand. This makes the methods widely applicable, but means that their convergence rate will be slow when the integrand satisfies nice regularity conditions. This is clearly sub-optimal when working with expensive models, where the number of evaluations will be limited. In this work, we propose to enhance MLMC through the use of surrogate models which encodes properties of the integrand, such as smoothness, sparsity or even periodicity. We focus in particular on Gaussian processes (GPs), which naturally lead to a class of algorithms that we call *multilevel Bayesian quadrature* (MLBQ).

MLBQ is a Bayesian probabilistic numerical method (Hennig et al., 2015; Cockayne et al., 2019; Wenger et al., 2021; Hennig et al., 2022), and more specifically a specific Bayesian quadrature algorithm (BQ; Diaconis, 1988; O’Hagan, 1991; Rasmussen and Ghahramani, 2003); see Briol et al. (2019) for a recent overview. As we will see in the remainder of the paper, this approach can lead to a posterior distribution on the value of the integral, with (i)

significant improvements in accuracy over existing methods when using the posterior mean as a point estimate, and (ii) the ability to quantify our uncertainty (given limited integrand evaluations) over the value of the integral.

2 BACKGROUND

We now review key components of our approach: MC, multilevel models, MLMC and BQ.

Monte Carlo Methods Let Π be a probability distribution on $\Omega \subseteq \mathbb{R}^d$ ($d \in \mathbb{N}_+$) and let $f: \Omega \rightarrow \mathbb{R}$ be some integrand of interest. We focus on approximating

$$\Pi[f] := \int_{\Omega} f(\omega) \Pi(d\omega)$$

and assume that f is square integrable with respect to Π (i.e. $\Pi[f^2] < \infty$). To tackle this task, we use pointwise evaluations of $f: \{\omega_i, f(\omega_i)\}_{i=1}^n$ for $n \in \mathbb{N}_+$ and $\omega_i \in \Omega$ for $i \in \{1, \dots, n\}$. For example, a MC estimator (Robert et al., 2004; Rubinstein and Kroese, 2016) takes the form

$$\hat{\Pi}_{\text{MC}}[f] := \frac{1}{n} \sum_{i=1}^n f(\omega_i),$$

where $\{\omega_i\}_{i=1}^n \sim \Pi$; that is, $\{\omega_i\}_{i=1}^n$ are independent and identically distributed (IID) realisations from Π . As $n \rightarrow \infty$ and under mild regularity conditions, MC estimators converge to $\Pi[f]$, making these approaches widely applicable. However, their performance when n is finite and relatively small can be quite poor, which is a common issue when f is expensive to evaluate. Alternative equal-weight estimators suffering from similar drawbacks include quasi-Monte Carlo (QMC) or randomised QMC (Owen, 2013), which use $\{\omega_i\}_{i=1}^n$ that form a space-filling design.

Multilevel Monte Carlo For multifidelity models, we can improve on MC through MLMC. Suppose that $f_L = f$, and $f_l: \Omega \rightarrow \mathbb{R}$ for $l \in \{0, \dots, L-1\}$ are approximations of f which increase both in accuracy and cost with the level l . The integral of interest can be expressed through a telescoping sum as

$$\Pi[f] = \Pi[f_L] = \Pi[f_0] + \sum_{l=1}^L \Pi[f_l - f_{l-1}]. \quad (1)$$

Instead of using a single MC estimator for $\Pi[f]$, we can estimate each term in the sum separately. Suppose that $\{\{\omega_{(l,i)}\}_{i=1}^{n_l}\}_{l=0}^L \sim \Pi$, the MLMC estimator is

$$\begin{aligned} \hat{\Pi}_{\text{MLMC}}[f] &:= \hat{\Pi}_{\text{MC}}[f_0] + \sum_{l=1}^L \hat{\Pi}_{\text{MC}}[f_l - f_{l-1}] \\ &= \frac{1}{n_0} \sum_{i=1}^{n_0} f_0(\omega_{(0,i)}) \\ &\quad + \sum_{l=1}^L \frac{1}{n_l} \sum_{i=1}^{n_l} (f_l(\omega_{(l,i)}) - f_{l-1}(\omega_{(l,i)})). \end{aligned}$$

For expensive integrands, there are two main advantages to this approach over MC. Firstly, each integrand (but the first) in the telescoping sum is of the form $f_l - f_{l-1}$, which will have low variance since we expect $f_l \approx f_{l-1}$ and hence

$\mathbb{V}[f_l - f_{l-1}] \approx \mathbb{V}[0] = 0$. As a result, a small n_l is sufficient to estimate such terms accurately through MC. Secondly, we have assumed that the functions are cheaper to evaluate for small l , so some of the initial terms in the sum can be estimated accurately through MC estimation with a large n_l .

These remarks can be made precise by considering the computational cost necessary to obtain a given accuracy ε , or equivalently a given mean-squared error (MSE) ε^2 . For an estimator $\hat{\Pi}[f]$, denote by $\text{Cost}(\hat{\Pi}, \varepsilon)$ this cost and by $\text{MSE}(\hat{\Pi}) := \mathbb{E}[(\hat{\Pi}[f] - \Pi[f])^2] = \mathbb{V}[\hat{\Pi}[f]] + (\mathbb{E}[\hat{\Pi}[f]] - \Pi[f])^2$ the MSE, where \mathbb{E} and \mathbb{V} denote the mean and variance with respect to all random variables in the estimator. For MC, $\mathbb{E}[\hat{\Pi}_{\text{MC}}[f]] = \Pi[f]$ and $\text{MSE}(\hat{\Pi}_{\text{MC}}) = \mathbb{V}[\hat{\Pi}_{\text{MC}}[f]] = n^{-1}\mathbb{V}[f]$. To achieve a MSE of ε^2 , n should be at least $\varepsilon^{-2}\mathbb{V}[f]$. If C is the computational cost per sample, a MSE of ε^2 will lead to $\text{Cost}(\hat{\Pi}_{\text{MC}}, \varepsilon) = \varepsilon^{-2}\mathbb{V}[f]C$.

As we will now see, MLMC can provide significant improvements over MC. Let C_0 denote the cost of f_0 , C_l the cost of $f_l - f_{l-1}$, $V_0 = \mathbb{V}[f_0]$ and $V_l = \mathbb{V}[f_l - f_{l-1}]$. The total cost of MLMC is $\sum_{l=0}^L n_l C_l$. The MSE and cost to achieve a MSE of ε^2 are hence

$$\begin{aligned} \text{MSE}(\hat{\Pi}_{\text{MLMC}}) &= \mathbb{V}[\hat{\Pi}_{\text{MLMC}}[f]] = \sum_{l=0}^L n_l^{-1} V_l, \\ \text{Cost}(\hat{\Pi}_{\text{MLMC}}, \varepsilon) &= \varepsilon^{-2} \left(\sum_{l=0}^L \sqrt{V_l C_l} \right)^2. \end{aligned}$$

To compare this cost with that of MC, we will consider two cases. Firstly, if $V_l C_l$ increases rapidly with levels, we will have $\text{Cost}(\hat{\Pi}_{\text{MLMC}}, \varepsilon) \approx \varepsilon^{-2} V_L C_L$. Secondly, if $V_l C_l$ decreases rapidly with levels, $\text{Cost}(\hat{\Pi}_{\text{MLMC}}, \varepsilon) \approx \varepsilon^{-2} V_0 C_0$. In contrast, for standard MC, assuming the cost of evaluating f_L is similar to that of evaluating $f_L - f_{L-1}$ and the variance of the estimate is $\mathbb{V}[f] = \mathbb{V}[f_L] \approx \mathbb{V}[f_0]$, we have $\text{Cost}(\hat{\Pi}_{\text{MC}}, \varepsilon) \approx \varepsilon^{-2} V_0 C_L$. Since $V_0 > V_L$ and $C_L > C_0$, we will therefore have $\text{Cost}(\hat{\Pi}_{\text{MC}}, \varepsilon) > \text{Cost}(\hat{\Pi}_{\text{MLMC}}, \varepsilon)$ regardless of the behaviour of $V_l C_l$.

This analysis of MLMC can be extended to find the optimal sample sizes per level given a fixed computational cost T (see Appendix B.1 or Giles, 2015, Section 1.3 for a similar analysis with optimal sample sizes for a fixed MSE):

$$\begin{aligned} n^{\text{MLMC}} &= (n_0^{\text{MLMC}}, \dots, n_L^{\text{MLMC}}) \\ &:= \left(D \sqrt{\frac{V_0}{C_0}}, \dots, D \sqrt{\frac{V_L}{C_L}} \right) \end{aligned}$$

where $D = T \left(\sum_{l=0}^L \sqrt{V_l C_l} \right)^{-1}$. Despite the potential advantages of the approach above, there are also limitations which prevent the direct use of n_l^{MLMC} . Firstly, V_l is usually unknown. It could be estimated from data, but unfortunately estimates of V_l may be unreliable if the sample size at level l is small. Secondly, f_L is usually an approximation to f (as opposed to $f = f_L$). Thirdly, as for our tsunami example, the number of levels can often be chosen by the user and it is often unclear how to decide which approximations f_0, \dots, f_L to include.

Bayesian Quadrature Clearly, the MLMC estimator can lead to significant gains, but we note that it focuses solely on sampling from Π and does not utilise properties of f . This is in contrast to BQ, an approach to integration which is based on a GP model of f . GPs are widely used as models for deterministic but computationally expensive functions, especially in computer experiments (Santner et al., 2018; Sacks et al., 1989) and in spatial statistics (Stein, 1999). We will denote a GP by $\mathcal{GP}(m, c)$ to emphasise the mean function $m : \Omega \rightarrow \mathbb{R}$ and the (symmetric and positive semi-definite) covariance function $c : \Omega \times \Omega \rightarrow \mathbb{R}$ (also called kernel), which uniquely identify the model. Given a $\mathcal{GP}(m, c)$ prior on f and some observations $\{\omega_i, f(\omega_i)\}_{i=1}^n$ at pairwise distinct $\{\omega_i\}_{i=1}^n \subset \Omega$ for some $n \in \mathbb{N}_+$, the posterior on f is also a GP with mean and covariance (Williams and Rasmussen, 2006)

$$\begin{aligned} \tilde{m}(\omega) &= m(\omega) + c(\omega, W)c(W, W)^{-1}(f(W) - m(W)), \\ \tilde{c}(\omega, \omega') &= c(\omega, \omega') - c(\omega, W)c(W, W)^{-1}c(W, \omega') \end{aligned}$$

for all $\omega, \omega' \in \Omega$. Here, we use vectorised notation: $W = (\omega_1, \omega_2, \dots, \omega_n)^\top$, $f(W) = (f(\omega_1), f(\omega_2), \dots, f(\omega_n))^\top$, $c(\omega, W) = c(W, \omega)^\top = (c(\omega, \omega_1), \dots, c(\omega, \omega_n))$ and $(c(W, W))_{i,j} = c(\omega_i, \omega_j)$ for all $i, j \in \{1, \dots, n\}$.

Prior knowledge on f , such as smoothness and periodicity, can be incorporated by specifying m and c . For example, the squared exponential covariance function $c_{\text{SE}}(\omega, \omega') = \exp(-\|\omega - \omega'\|_2^2 / \gamma^2)$ with length-scale $\gamma > 0$ implies a prior belief that f has infinitely many derivatives. Alternatively, the Matérn covariance function $c_{\text{Matérn}}(\omega, \omega') = 2^{1-v} \Gamma^{-1}(v) (\sqrt{2v} \|\omega - \omega'\|_2 / \gamma)^v K_v(\sqrt{2v} \|\omega - \omega'\|_2 / \gamma)$ with smoothness $v > 0$ and length-scale $\gamma > 0$, where K_v is a modified Bessel function of the second kind, implies a belief that f is $\lceil v \rceil - 1$ times differentiable.

BQ (Diaconis, 1988; O'Hagan, 1991; Rasmussen and Ghahramani, 2003; Briol et al., 2019) is an estimator for $\Pi[f]$ motivated through Bayesian inference. The idea is to specify a prior on f , obtain the posterior on f given evaluations of f , then consider the implied (pushforward) posterior on $\Pi[f]$. The most common approach uses a $\mathcal{GP}(m, c)$ prior; in that case, the posterior on $\Pi[f]$ is Gaussian with mean and variance

$$\begin{aligned} \mathbb{E}_{\text{BQ}}[\Pi[f]] &= \hat{\Pi}_{\text{BQ}}[f] = \Pi[\tilde{m}] \\ &= \Pi[m] + \Pi[c(\cdot, W)]c(W, W)^{-1}(f(W) - m(W)), \\ \mathbb{V}_{\text{BQ}}[\Pi[f]] &= \Pi[\Pi[\tilde{c}]] \\ &= \Pi[\Pi[c]] - \Pi[c(\cdot, W)]c(W, W)^{-1}\Pi[c(W, \cdot)], \end{aligned}$$

where $\Pi[c(\cdot, W)] = (\Pi[c(\cdot, \omega_1)], \dots, \Pi[c(\cdot, \omega_n)])^\top$ and we use the convention that for a function with two inputs, $\Pi[\Pi[\cdot]]$ always denotes integration once with respect to each input. In contrast with MC methods which rely on central limit theorems, $\mathbb{V}_{\text{BQ}}[\Pi[f]]$ can quantify our uncertainty about $\Pi[f]$ for finite (and possibly small) n .

A particular advantage of the formulae above is that they are defined for arbitrary $\{\omega_i\}_{i=1}^n$. A number of point sets have been studied including IID (Rasmussen and Ghahramani, 2003), QMC (Briol et al., 2019; Jagadeeswaran and Hickernell, 2019), realisations from determinantal point processes (Belhadji et al., 2019), point sets with symmetry properties (Karvonen and Särkkä, 2018; Karvonen et al., 2019) and adaptive designs (Osborne et al., 2012; Gunter et al., 2014; Briol et al., 2015). For specific point sets and GP priors, $\hat{\Pi}_{\text{BQ}}[f]$ also coincides with classical quadrature rules (Diaconis, 1988; Karvonen and Särkkä, 2017).

The two main disadvantages of BQ are that: (i) as per GPs, the computational cost is $\mathcal{O}(n^3)$, due to the need to invert $n \times n$ matrices, and (ii) $\Pi[c(\cdot, \omega)]$ for $\omega \in \Omega$ and $\Pi[\Pi[c]]$ are only tractable for some pairs of distributions and covariance functions (see Table 1 in Briol et al., 2019). On the other hand, BQ also has much faster convergence rates than classical Monte Carlo methods when d is small or moderate (Briol et al., 2019; Kanagawa et al., 2020; Wynne et al., 2021). For this reason, BQ has mostly been applied to problems where n is constrained to be small (for example when the integrand is expensive) and the integration measure is relatively simple. This includes problems in global illumination in computer graphics (Brouillat et al., 2009), cardiac modelling (Oates et al., 2017), engineering control (Paul et al., 2018), econometrics (Oettershagen, 2017), risk (Cadini and Gioletta, 2016), likelihood free inference (Bharti et al., 2023) and in variational inference (Acerbi, 2018).

3 METHODOLOGY

Although MLMC is particularly well-suited to integrals involving multifidelity models, it usually disregards any prior information on the integrand. We now remedy this issue by designing a novel estimator which combines the advantages of BQ and MLMC. Our proposed algorithm is relatively straightforward: it uses the telescopic sum in Equation (1) and approximates each of the terms through BQ rather than MC. Here and throughout the remainder of the paper, we use the convention that $f_{-1} \equiv 0$ to simplify all expressions. Suppose we have access to the evaluations $\{\{f_l(\omega_{(l,i)}) - f_{l-1}(\omega_{(l,i)})\}_{i=0}^{n_l}\}_{l=0}^L$ of the approximate integrands on Ω . We will specify a sequence of priors such that $\mathcal{GP}(m_l, c_l)$ is a prior on the increment $f_l - f_{l-1}$, and we will take these increments to be independent a-priori.

Proposition 1. *Given the priors and datasets described above, the posterior on f is a Gaussian process and the posterior on $\Pi[f]$ is a univariate Gaussian with mean*

$$\begin{aligned} \mathbb{E}_{\text{MLBQ}}[\Pi[f]] &:= \sum_{l=0}^L \hat{\Pi}_{\text{BQ}}[f_l - f_{l-1}] \\ &= \sum_{l=0}^L \left(\Pi[m_l] + \Pi[c_l(\cdot, W_l)]c_l(W_l, W_l)^{-1} \right. \\ &\quad \left. \times (f_l(W_l) - f_{l-1}(W_l) - m_l(W_l)) \right) \end{aligned}$$

and variance

$$\begin{aligned} \mathbb{V}_{\text{MLBQ}}[\Pi[f]] &:= \sum_{l=0}^L \mathbb{V}_{\text{BQ}}[\Pi[f_l - f_{l-1}]] \\ &= \sum_{l=0}^L \left(\Pi[\Pi[c_l]] - \Pi[c_l(\cdot, W_l)]c_l(W_l, W_l)^{-1}\Pi[c_l(W_l, \cdot)] \right) \end{aligned}$$

where $W_l = (\omega_{(l,1)}, \dots, \omega_{(l,n_l)})^\top$ for $l \in \{0, \dots, L\}$.

The proof is given in Appendix B.2. Once again, a point estimator can be obtained through the posterior mean $\hat{\Pi}_{\text{MLBQ}}[f] := \mathbb{E}_{\text{MLBQ}}[\Pi[f]]$ and we will call this the *multilevel Bayesian quadrature* (MLBQ) estimator. Although MLBQ requires only a straightforward modification of the MLMC algorithm, we will see in the remainder of the paper that it will allow us to take advantage of the properties of both MLMC and BQ.

A simple illustration example comparing BQ and MLBQ ($L = 2$) with the same evaluation constraint is shown in Figure 2. We used the approximations from the Poisson equation experiment in Section 6 and Appendix C.1. As we observed, the GP for MLBQ fits f_2 better than the GP for BQ. The MLBQ estimator has smaller error and smaller variance than the BQ estimator. The cost for implementing MLBQ is $\mathcal{O}(\sum_{l=0}^L n_l^3)$, which is larger than the $\mathcal{O}(\sum_{l=0}^L n_l)$ of MLMC. However, for most multifidelity models, we expect these costs to be dwarfed by the cost of function evaluations which is $\sum_{l=0}^L n_l C_l$. Additionally, we will see in the next Section that MLBQ can have a much faster convergence rate than MLMC. Due to the independence assumption, we can estimate the GP hyperparameters separately for each level; see Section 5. If the assumption is violated, we could be under- or over-estimating our uncertainty. It is possible to do away with this assumption by modelling levels jointly as demonstrated in Appendix B.3 or following the work on multi-output BQ of Xi et al. (2018), but this would prohibitively increase the cost to $\mathcal{O}((\sum_{l=0}^L n_l)^3)$.

4 THEORY

We now prove an upper bound on the error of MLBQ and derive the optimal number of samples per level.

Let $L^2(\Omega)$ denote the space of square-integrable functions on $\Omega \subseteq \mathbb{R}^d$ with respect to the Lebesgue measure. The Sobolev space $W_2^\alpha(\Omega)$ of integer order $\alpha \geq 0$ consists of functions $f \in L^2(\Omega)$ for which $\|f\|_\alpha := (\sum_{\beta \in \mathbb{N}^d: |\beta| \leq \alpha} \|D^\beta f\|_{L^2(\Omega)}^2)^{1/2} < \infty$, where $|\beta| = \sum_{i=1}^d \beta_i$ and $D^\beta f$ is the weak derivative (Adams and Fournier, 2003, p. 22) of order β . For non-integer $\alpha \geq 0$, the Sobolev norm can be defined via Fourier transforms and the two definitions coincide, up to a constant, for integer α if Ω is sufficiently regular (Wynne et al., 2021, Section 2.2). The space $W_2^\alpha(\Omega)$ is a Hilbert space.

By the Moore–Aronszajn Theorem (Berlinet and Thomas-Agnan, 2004, Theorem 3 in Chapter 1), every positive semi-definite covariance function $c: \Omega \times \Omega \rightarrow \mathbb{R}$ induces a unique

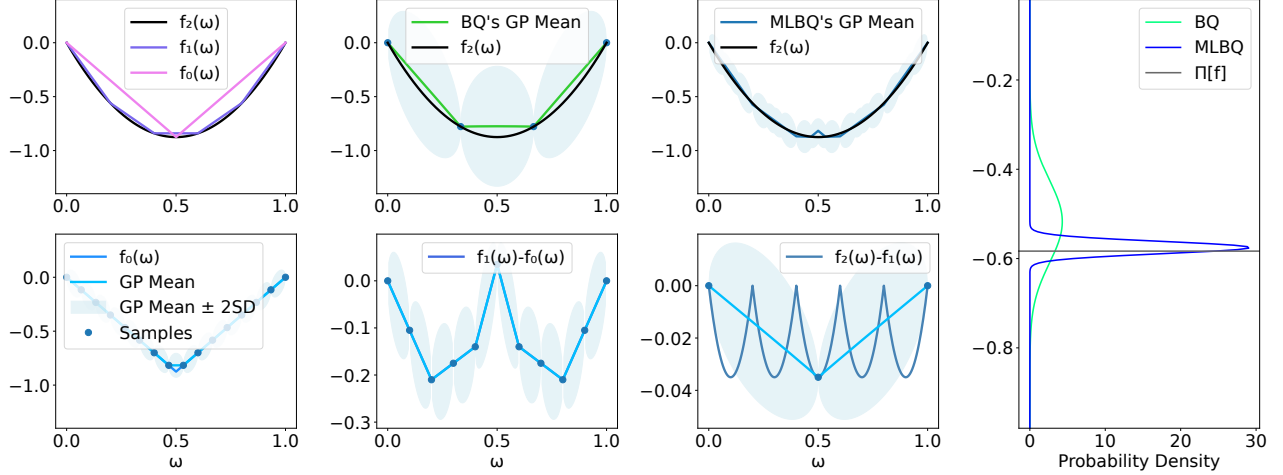


Figure 2: Illustration Example: *Upper Left*: the approximations to f , GP for BQ, GP for MLBQ. *Bottom Left*: GP for level 0, 1, 2 of MLBQ. *Right*: BQ and MLBQ estimators.

reproducing kernel Hilbert space (RKHS) $\mathcal{H}(c)$ consisting of functions $f: \Omega \rightarrow \mathbb{R}$ and equipped with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}(c)}$ and norm $\|\cdot\|_{\mathcal{H}(c)}$. The RKHS satisfies: (1) $c(\cdot, \omega) \in \mathcal{H}(c)$ for every $\omega \in \Omega$, and (2) the reproducing property that $f(\omega) = \langle f, c(\cdot, \omega) \rangle_{\mathcal{H}(c)}$ for every $f \in \mathcal{H}(c)$ and $\omega \in \Omega$.

The following assumptions are used in our results:

- A1. The domain is of the form $\Omega = \Omega_1 \times \dots \times \Omega_d$ for each Ω_i a non-empty interval.
- A2. The distribution Π has a bounded density function π ; i.e. $\|\pi\|_{L^\infty(\Omega)} := \sup_{\omega \in \Omega} \pi(\omega) < \infty$.
- A3. For each $l \in \{0, \dots, L\}$, the RKHS $\mathcal{H}_l := \mathcal{H}(c_l)$ is norm-equivalent to $W_2^{\alpha_l}(\Omega)$ for $\alpha_l > d/2$. Two Hilbert spaces H_1 and H_2 are norm-equivalent if and only if they are equal as sets and there are constants $b_1, b_2 > 0$ such that $b_1 \|f\|_{H_1} \leq \|f\|_{H_2} \leq b_2 \|f\|_{H_1}$ for all $f \in H_1 = H_2$.
- A4. There are $\beta_0, \dots, \beta_L > d/2$ such that $f_0 \in W_2^{\beta_0}(\Omega)$ and $f_l, f_{l-1} \in W_2^{\beta_l}(\Omega)$ for every $l \in \{1, \dots, L\}$.
- A5. For each $l \in \{0, \dots, L\}$, the fill-distance $h_{W_l, \Omega} = h_{l, \Omega} := \sup_{\omega \in \Omega} \min_{i=1, \dots, n_l} \|\omega - \omega_{(l,i)}\|_2$ satisfies $h_{l, \Omega} \leq h_{\text{qu}} n_l^{-1/d}$ for a constant $h_{\text{qu}} > 0$.
- A6. The prior means are $m_l \equiv 0$ for all $l \in \{0, \dots, L\}$.

The purpose of Assumption A1 is to ensure that the domain is sufficiently regular for the use of Sobolev extension and embedding theorems. This assumption could be generalised to allow more complex domains without affecting the convergence rate (Wynne et al., 2021, Section 3.1). Assumption A3 and its relatives are standard in the error analysis

of GP and BQ methods (e.g., Karvonen et al., 2020; Teckentrup, 2020; Wynne et al., 2021) and are important for deriving our theoretical results. The RKHS of a Matérn kernel $c_{\text{Matérn}}$ with smoothness ν and any length-scale is norm-equivalent to $W_2^\alpha(\Omega)$ for $\alpha = \nu + d/2$ whenever Ω satisfies Assumption A1. Assumption A5, known as the quasi-uniformity assumption (Wendland, 2004, Section 14.1), ensures that each of the sets W_l covers Ω in a sufficiently uniform manner, because the fill-distance of a set W equals the radius of the largest ball in Ω which contains no point from W . Regular grids are examples of sets that satisfy Assumption A5. Assumption A6 is made out of convenience and could be replaced with the assumption that $m_l \in W_2^{\beta_l}(\Omega)$ for each l .

Theorem 2. *Suppose that assumptions A1–A6 hold and define $\tau_l := \min\{\alpha_l, \beta_l\}$. Then*

$$\begin{aligned} \text{Err}(\hat{\Pi}_{\text{MLBQ}}) &= |\Pi[f_L] - \hat{\Pi}_{\text{MLBQ}}[f_L]| \\ &\leq \|\pi\|_{L^\infty(\Omega)} \sum_{l=0}^L a_l \|f_l - f_{l-1}\|_{\mathcal{H}_l} n_l^{-\tau_l/d} \end{aligned}$$

whenever each n_l is sufficiently large. Each constant $a_l > 0$ depends on $\alpha_l, \beta_l, c_l, h_{\text{qu}}, d$, and Ω , but not on f_l or the data points.

Theorem 2 is proved in Appendix B.4. The proof is similar to the convergence proofs in Kanagawa et al. (2020); Karvonen et al. (2020); Teckentrup (2020); Wynne et al. (2021). The Sobolev norm in the bound may be replaced with the RKHS norm $\|f_l - f_{l-1}\|_{\mathcal{H}_l}$ if $\beta_l \geq \alpha_l$ due to assumption A3.

If it is assumed that $\beta_l \geq \alpha_l$ for each l , one may use Theorem 1 and Corollary 2 in Krieg and Sonnleitner (2022) to prove a variant of Theorem 2 in which the points are independent samples from a uniform distribution on Ω and the upper bound is for the expected error of the MLBQ.

Various other generalisations of Theorem 2 are possible but are not included here so as to simplify the presentation of our assumptions. These include non-zero prior means, varying kernel parameters (Teckentrup, 2020), misspecified likelihoods (Wynne et al., 2021), and improved rates when each f_l has, essentially, twice the smoothness of c_l (Tuo et al., 2020; Karvonen et al., 2020, Sections 3.4 and 4.5) or when both f_l and c_l are infinitely differentiable (Karvonen, 2019, Theorem 2.20).

At each level l , the convergence rate of $\mathcal{O}(n_l^{-\tau_l/d})$ is faster than the rate for MC estimators of $\mathcal{O}(n_l^{-1/2})$ because $\tau_l/d = \min\{\alpha_l, \beta_l\}/d > 1/2$. Since f_0, \dots, f_L approximate the same function, the kernels c_l and the smoothnesses α_l and β_l do not typically change with l , which means that the constants a_l do not change. If, additionally, $\|f_l - f_{l-1}\|_\tau$ tends to zero as l increases, which is usually the case because approximation quality should increase with the level, we see that fewer evaluations are needed at higher levels. However, if τ_l differ significantly, more evaluations than expected may be needed at higher levels.

Using Theorem 2 and assuming we use the same prior at each level, we can also derive the optimal number of samples for MLBQ under a limited computational budget. To do so, we assume that the cost of fitting GPs at each level is dwarfed by the cost of function evaluations. This is reasonable because function evaluation costs tend to be relatively large for applications where MLMC is commonly used. For example, for differential equation models the cost is usually driven by the cost of the solvers such as finite difference, finite element or finite volume methods, and this can be large for fine meshes. For example, for the tsunami example in Section 6, fitting all the GPs takes less than 25 seconds whereas a single evaluation of $f_L - f_{L-1}$ takes 150 seconds. For this reason, we therefore assume that the total cost of running MLBQ and functions evaluations is given by $\gamma \sum_{l=0}^L C_l n_l$ for some $\gamma \geq 1$ but close to 1.

Theorem 3. *Suppose that assumptions A1–A6 hold and c_l and $\tau := \tau_l = \min\{\alpha_l, \beta_l\} = \alpha_l$ do not depend on l . Then*

$$\begin{aligned} n^{\text{MLBQ}} &= \left(n_0^{\text{MLBQ}}, \dots, n_L^{\text{MLBQ}} \right) \\ &:= \underset{\substack{n_0, n_1, \dots, n_L \\ \text{s.t. } \gamma \sum_{l=0}^L C_l n_l = T}}{\operatorname{argmin}} \sum_{l=0}^L a_l \|f_l - f_{l-1}\|_\tau n_l^{-\tau/d} \end{aligned}$$

for $\gamma \geq 1$ and $T > 0$ is solved by

$$n_l^{\text{MLBQ}} = D \left(\frac{\|f_l - f_{l-1}\|_\tau}{C_l} \right)^{\frac{d}{\tau+d}} \quad \forall l \in \{0, \dots, L\},$$

where $D = T \left(\gamma \sum_{l=0}^L C_l^{\frac{\tau}{\tau+d}} (\|f_l - f_{l-1}\|_\tau)^{\frac{d}{\tau+d}} \right)^{-1}$.

The proof is given in Appendix B.5. The additional assumptions were introduced to simplify the result by ensuring that a_l does not depend on l . If the function evaluation costs do not dominate or if τ_l differ, one can still calculate the

optimal sample sizes by solving the optimisation problem in Theorem 3 numerically. The theorem provides a solution for the relaxed optimisation problem where n_1, \dots, n_L are real numbers. In practice, it will be needed to use natural numbers, and this is possible by taking the floor or ceiling of each n_1, \dots, n_L .

The optimal sample sizes for MLMC and MLBQ are similar; here, $\|f_l - f_{l-1}\|_\tau$ is analogous to V_l in that it measures the size of each element in the telescoping sum. We expect $\|f_l - f_{l-1}\|_\tau$ to be a decreasing function of l which converges to zero. If the convergence is slow, the sample size for large l has to be relatively large, whereas it can be relatively small otherwise. Additionally, a large cost C_l also leads to relatively smaller sample sizes. For MLMC, the optimal sample size at level l is proportional to $C_l^{-1/2}$ whereas for MLBQ it is proportional to $C_l^{-d/(\tau+d)}$. Therefore, when $\tau > d$, the penalisation for large C_l is smaller for MLBQ than MLMC, and vice-versa. This is intuitive because when τ is large, the integrands are smoother and we expect BQ to be able to approximate them fast in the number of samples.

Plugging in the optimal samples sizes of Theorem 3 to the bound in Theorem 2, we obtain that

$$\operatorname{Err}(\hat{\Pi}_{\text{MLBQ}}) \leq AT^{-\frac{\tau}{d}} \left(\sum_{l=0}^L C_l^{\frac{\tau}{\tau+d}} \|f_l - f_{l-1}\|_\tau^{\frac{d}{\tau+d}} \right)^{\frac{\tau+d}{d}},$$

where $A = \|\pi\|_{L^\infty(\Omega)} a \gamma^{\tau/d}$. For BQ based on evaluations of f_L and utilising the same computational budget we obtain

$$\operatorname{Err}(\hat{\Pi}_{\text{BQ}}) \leq AT^{-\frac{\tau}{d}} C_L^{\frac{\tau}{d}} \|f_L\|_\tau$$

from Theorem 2 by setting $f_l \equiv 0$ and $C_l = 0$ for every $l \in \{0, \dots, L-1\}$. Let us denote the two upper bounds above by B_{MLBQ} and B_{BQ} . To compare these bounds, we consider two cases. Firstly, if the term $b_l := C_l^{\tau/(\tau+d)} \|f_l - f_{l-1}\|_\tau^{d/(\tau+d)}$ grows rapidly with l , then B_{MLBQ} is dominated by the highest level L , so that $B_{\text{MLBQ}} \approx AT^{-\tau/d} C_L^{\tau/d} \|f_L - f_{L-1}\|_\tau$. Secondly, if b_l decreases rapidly with l , then $B_{\text{MLBQ}} \approx AT^{-\tau/d} C_0^{\tau/d} \|f_0\|_\tau$. In either case, the bound on $\operatorname{Err}(\hat{\Pi}_{\text{MLBQ}})$ is smaller than that on $\operatorname{Err}(\hat{\Pi}_{\text{BQ}})$ under natural assumptions. In the first case

$$B_{\text{BQ}} \approx \frac{\|f_L\|_\tau}{\|f_L - f_{L-1}\|_\tau} B_{\text{MLBQ}} \geq B_{\text{MLBQ}}$$

if $\|f_L\|_\tau \geq \|f_L - f_{L-1}\|_\tau$, whilst in the second case

$$B_{\text{BQ}} \approx \left(\frac{C_L}{C_0} \right)^{\tau/d} \frac{\|f_L\|_\tau}{\|f_0\|_\tau} B_{\text{MLBQ}} \geq B_{\text{MLBQ}}$$

if $C_L \geq C_0$ and $\|f_L\|_\tau \geq \|f_0\|_\tau$.

5 PRACTICAL CONSIDERATIONS

Before moving on to experimental results, we briefly discuss practical considerations for the implementation of MLBQ.

Firstly, for each level l , we will usually include at least one amplitude σ_l parameter (so that the covariance function takes the form $\tilde{c}_l(\omega, \omega') = \sigma_l^2 c_l(\omega, \omega')$ for some covariance function c_l) and a lengthscale γ_l (or a lengthscale per dimension of the data). We suggest to select these by maximising the marginal log-likelihood separately for each level:

$$L(\gamma_l, \sigma_l) = -\frac{1}{2}\sigma_l^{-2} (f_l(W_l) - f_{l-1}(W_l))^\top c_l(W_l, W_l)^{-1} \times (f_l(W_l) - f_{l-1}(W_l)) - n_l \log \sigma_l^2 + \frac{1}{2} \log |c_l(W_l, W_l)| - \frac{n_l}{2} \log 2\pi,$$

For a given σ_l , this can be done in closed-form as follows:

$$\sigma_l^* = \sqrt{\frac{(f_l(W_l) - f_{l-1}(W_l))^\top c_l(W_l, W_l)^{-1} (f_l(W_l) - f_{l-1}(W_l))}{n_l}}.$$

For the lengthscale, the maximum of $L(\gamma_l, \sigma_l)$ as a function of γ_l needs to be obtained numerically. When n_l is large, we can use mini-batches with stochastic optimization.

Note that it is essential to select the hyperparameters for each level independently. To illustrate this, consider each level having prior $\mathcal{GP}(0, \sigma^2 c_l)$. All other parameters besides the amplitude are fixed, and maximum likelihood is used to estimate the amplitude. The resulting maximum marginal likelihood estimate (MLE) is

$$\sigma_{\text{all}} = \sqrt{\frac{y^\top c(W, W)^{-1} y}{\sum_{l=0}^L n_l}},$$

where the vectors W and y are formed by stacking all W_l and $f_l(W_l) - f_{l-1}(W_l)$, respectively and the matrix $c(W, W)$ is formed with diagonal blocks all $c_l(W_l, W_l)$ and off-diagonal components 0s. Inserting this MLE in the equation for the MLBQ variance yields

$$\mathbb{V}_{\text{MLBQ}}[\Pi[f]] = \frac{y^\top c(W, W)^{-1} y}{\sum_{l=0}^L n_l} \sum_{l=0}^L (\Pi[\Pi[c_l(\cdot, \cdot)]] - \Pi[c_l(\cdot, W_l)] c_l(W_l, W_l)^{-1} \Pi[c_l(W_l, \cdot)]). \quad (2)$$

Because each term in the sum above depends only on W_l , the knowledge that $f_l - f_{l-1}$ tends to zero as l increases is not exploited. The essential property of a multilevel method that less data is needed on higher levels is not reflected in the MLBQ variance. This defect is eliminated if each level has an independently estimated amplitude parameter. In this case the variance becomes

$$\mathbb{V}_{\text{MLBQ}}[\Pi[f]] = \sum_{l=0}^L \sigma_l^{*2} (\Pi[\Pi[c_l(\cdot, \cdot)]] - \Pi[c_l(\cdot, W_l)] c_l(W_l, W_l)^{-1} \Pi[c_l(W_l, \cdot)]).$$

Now the magnitude of $f_l - f_{l-1}$ directly affects the l th term: if $f_l - f_{l-1}$ is small, the contribution of the l th term to $\mathbb{V}_{\text{MLBQ}}[\Pi[f]]$ is small even if W_l contains only few points, unlike in (2).

6 EXPERIMENTS

We now evaluate MLBQ for synthetic differential equation models and landslide-generated tsunami modelling. The

code to reproduce our experiments is available at <https://github.com/CeciliaKaiyu/MLBQ>. The MLBQ method is also implemented in the ProbNum open-source Python package (Wenger et al., 2021).

Poisson Equation The Poisson equation is a canonical partial differential equation which arises in physics (e.g., Mathews and Walker, 1970, Chapter 8). We consider a synthetic model where for $f: (0, 1) \rightarrow \mathbb{R}$,

$$f''(\omega) = z(\omega) \text{ for } \omega \in (0, 1) \quad \& \quad f(0) = f(1) = 0$$

where $z(\omega) = 1$. Here, $\Pi[f] = \int_0^1 f(\omega) d\omega$ so that Π is a $\text{Unif}(0, 1)$. To obtain f_0, \dots, f_L , we use piecewise linear finite element approximations as described in Appendix C.1. We use $L = 2$ and have $C = (C_0, C_1, C_2) = (3.6, 8.5, 42.4)$ (all measured in 10^{-3} seconds). This problem is relatively simple and could be brute-forced with MC, but has the advantage that we can compute the optimal sample sizes for MLBQ and MLMC since A1–A6 are all satisfied when using a uniform grid of points and $\|f_l - f_{l-1}\|_{\pi}$ can be computed in closed form for all l . It therefore makes for a good test-bed for our method.

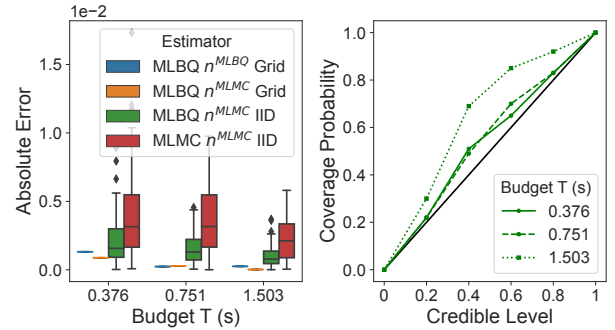


Figure 3: Poisson Equation. *Left*: Absolute integration error. *Right*: Calibration of MLBQ, IID points.

We compare four different settings: MLBQ using n^{MLBQ} and uniform grid points, MLBQ using n^{MLMC} and uniform grid points or IID points, and MLMC using n^{MLMC} and IID points. To implement n^{MLMC} , we brute-forced the computation of V_0, \dots, V_L through an MC approximation. All MLBQ algorithms use a mean-zero GP with Matérn 0.5 kernel, and all sample sizes are given in Appendix C.1.

Figure 3 visualizes the result of 100 repetitions of the experiment, where for each repetition, we evaluated f_0, \dots, f_L at new point sets, and used the same dataset for MLBQ and MLMC to estimate $\Pi[f]$. When using uniform grids, there is no randomness and the experiment is therefore done only once. The left-hand side plot shows that $\hat{\Pi}_{\text{MLBQ}}[f]$ significantly outperforms $\hat{\Pi}_{\text{MLMC}}[f]$ across a range of budgets T . For MLBQ, we also see that the impact of the sample size per level is not as significant as that of type of points

used, with the uniform grid outperforming IID points. This is promising since the optimal sample sizes will be difficult to obtain in general due to the need to access V_l or $\|f_l - f_{l-1}\|_{\tau_l}$ for each level l (in the cases of n^{MLMC} and n^{MLBQ} respectively). The right-hand side plot shows coverage frequencies for various credible level. Most of the results lie closely to the identity line, indicating that MLBQ has good frequentist coverage. The only exception is for larger budget T , in which case MLBQ is under-confident in the sense that the posterior variance is too large relative to frequentist coverage probabilities. This is generally preferable to being over-confident.

ODE with Random Coefficient and Forcing We now consider a popular test-bed for MLMC as first studied in Section 7.1 of Giles (2015):

$$\frac{d}{dx} (c(x) \frac{du}{dx}) = -50^2 \omega_2^2 \text{ for } x \in (0, 1)$$

with $u(0) = u(1) = 0$, $c(x) = 1 + \omega_1 x$, $\omega_1 \sim \text{Unif}(0, 1)$ and $\omega_2 \sim \mathcal{N}(0, 1)$. The integral is

$$\Pi[f] = \int_{\Omega} f(\omega) \Pi(d\omega) = \int_{\Omega} (\int_0^1 u(x, \omega) dx) \Pi(d\omega)$$

where $\omega = (\omega_1, \omega_2)$ and Π is a product of the marginal distributions for ω_1 and ω_2 , and $\Omega = [0, 1] \times (-\infty, \infty)$. We take $L = 2$ and each level is obtained through a finite difference approximation of f with grid size h_l (see Appendix C.2). We have $C = (C_0, C_1, C_2) = (1.0, 2.6, 21.8)$ (in 10^{-3} seconds).

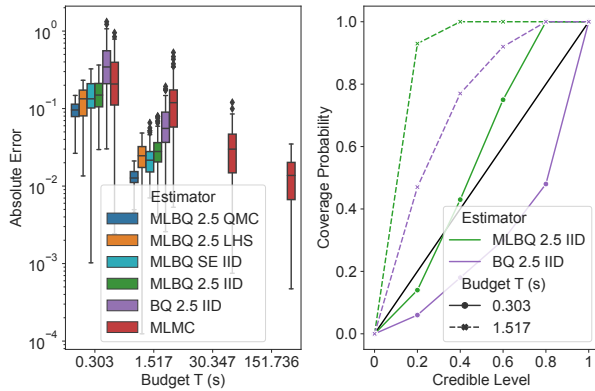


Figure 4: ODE with Random Coefficient and Forcing. *Left*: Absolute integration error. *Right*: Calibration of MLBQ and BQ with IID points.

The assumptions from Section 4 do not hold here since Ω_2 is unbounded (which breaks A1), but we still use this example to study our method beyond the setting of our theoretical results. We compare MLBQ with different point sets, MLMC and BQ with IID samples. For all multilevel methods, we select the sample size according to n^{MLMC} (see Appendix C.2). In this example, we cannot use n^{MLBQ}

since $\|f_l - f_{l-1}\|_{\tau_l}$ is not available in closed form. All methods using a GP with covariance taken to be a product of univariate Matérn kernels per dimension with $\nu = 2.5$, or a squared exponential kernel (“SE”).

There are three interesting observations in the left-hand side plot in Figure 4. Firstly, MLBQ with a Halton sequence (“QMC”) or a Latin hypercube design (“LHS”) performs better than with IID sampling, once again reflecting the importance of the choice of point set. Secondly, the choice of kernel also has some impact, with the MLBQ estimator with squared exponential kernel outperforming the corresponding estimator with Matérn kernel. Thirdly, MLBQ significantly outperforms BQ and MLMC, even though a sub-optimal sample size per level was used here. More precisely, MLBQ (with any point set) at $T = 1.517$ s is able to outperform MLMC with a budget 20 times larger ($T = 30.347$ s) and is comparable to MLMC with a budget 100 times larger ($T = 151.736$ s). A similar conclusion holds when comparing MLBQ with BQ.

Finally, the right-hand side plot shows that the calibration performances of MLBQ and BQ are very similar. The methods tend to be over-confident when T is very small, and become under-confident when T is larger.

Landslide-Generated Tsunami We now consider a variation of the submerged landslide-generated tsunami of Lynett and Liu (2005). The movement of the landslide mass on the beach slope results in the generation of tsunami waves (see Figure 1, left), and we consider the temporal evolution of this wave. We use a tsunami simulator called Volna-OP2 (Reguly et al., 2018; Giles et al., 2020), which is a differential equation solver capable of simulating the complete life-cycle of a tsunami: generation, propagation and inundation. Volna-OP2 is an advanced tsunami simulation tool using unstructured meshes accelerated on GPUs that has been utilised widely by geoscientists, e.g. for real-time tsunami warning systems (Giles et al., 2021) or hazard assessments (Gopinathan et al., 2021; Salmanidou et al., 2021). Volna-OP2 numerically solves the nonlinear shallow water equations (see Appendix C.3) with a finite volume method. The simulations with Volna-OP2 are run on a single NVIDIA P100 graphical processing unit (the Wilkes2 machine in Cambridge’s CSD3).

We use a bathymetry $h(x, t)$ with $x \in [-100, 3100]$ (in meters) and $t \in [0, 300]$ (in seconds). The parameters of interest are: ω_1 , defined to be the ratio of the maximum vertical thickness of the slide (Δh) to the initial vertical distance from the center point of the slide to the surface ($d_o = 50$ m); ω_2 , the slope angle; and ω_3 , the length of the slide. All of these parameters lead to nonlinear effects which can greatly influence the amplification of tsunami waves. The value of these parameters tends to be unknown a-priori and we take Π to consist of marginal distributions representing our uncertainty, given by $\text{Unif}(0.125, 0.5)$,

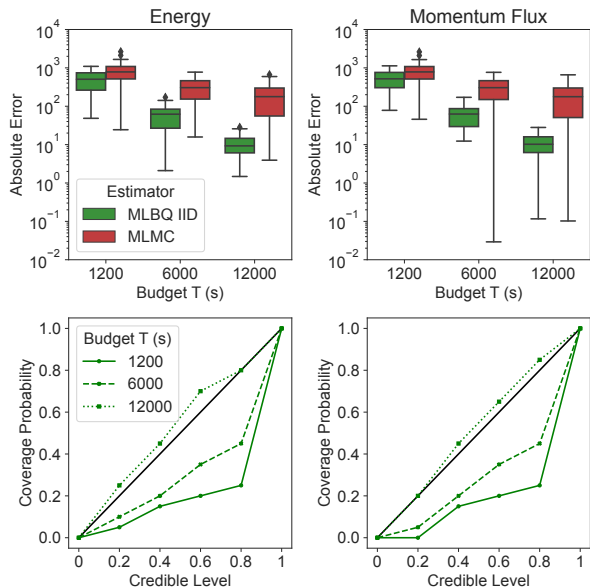


Figure 5: Landslide-Generated Tsunami. *Top*: Absolute integration error. *Bottom*: Calibration of MLBQ. The left-hand side plots correspond to $\Pi[f^e]$ and the right-hand side plots to $\Pi[f^m]$.

$\text{Unif}(5^\circ, 15^\circ)$ and $\text{Unif}(100m, 200m)$ respectively. A representative example of the solution provided by Volna-OP2 for $\omega = (0.375, 10^\circ, 150m)$ is on the right-hand side in Figure 1. In tsunami modelling, two functionals of the solution of the model which are often of interest are the total energy flux (Degueudre et al., 2016), denoted $f^e: \Omega \rightarrow \mathbb{R}$, and the momentum flux (Park et al., 2017), denoted $f^m: \Omega \rightarrow \mathbb{R}$, and we therefore want to compute $\Pi[f^e]$ and $\Pi[f^m]$.

In the experiments, we estimate these quantities at a gauge at $x = 3000$ with MLBQ and MLMC using the same IID point sets and repeat the experiment 20 times. We take $L = 4$ and each level corresponds to a different spatial and temporal resolution used in the solver. The number of evaluations per level are listed in Appendix C.3. We have $C = (C_0, C_1, C_2, C_3, C_4) = (5, 15, 30, 65, 150)$ (measured in seconds). These costs are significantly larger than the cost of fitting all GPs, which is carried out on a laptop and ranges from 1 second to 25 seconds depending on sample sizes per level. We use a tensor product Matérn kernel with smoothness $\nu = 2.5$ for MLBQ. The related analytical formulas are provided in Appendix C.6.

The upper box plots of Figure 5 show the absolute error of our estimates. As we observed, MLBQ always significantly outperforms MLMC. More precisely, given a fixed computational budget, MLMC tends to have an error between 10 and 100 times smaller than MLBQ. We did not compare to BQ here because f_L is too computationally expensive to obtain a reliable estimate. The calibration plots show that

MLBQ tends to be overconfident when the budget is small ($T = 1200$ or $T = 6000$) but becomes under-confident when budget is larger ($T = 12000$).

Overall, although the computational complexity setup studied in this paper could be considered a ‘toy model’ for the tsunami warning community, any method which showcases such a drastic reduction in computing time could have a significant impact on tsunami warning centres given their tight budget constraints.

7 CONCLUSION

We introduced MLBQ, a method for computing integrals involving multifidelity models. MLBQ enhances MLMC by bringing to it the advantages of Bayesian methods, namely: (1) the ability to make use of prior information about the integrand, which leads to faster convergence rates, and (2) the ability to provide Bayesian quantification of uncertainty over the value of the integral of interest. From the point of view of Bayesian probabilistic numerics, this algorithm is also a step forward towards making the field reach applications where it can be most impactful, including specifically when models are computationally expensive and it is therefore desirable to make use of as much prior knowledge as possible to improve estimates.

There are a large number of possible extensions and we therefore only mention some of the most promising. Firstly, one could consider extending MLBQ to multi-index Monte Carlo (Haji-Ali et al., 2016), which can be useful for models where levels can have multiple indices. For example, in partial differential equation models, one index could be discretisation through time and the other through space, and using this structure could bring further gains. Secondly, one could consider improving scalability through hybrid strategies where BQ is used on the more expensive levels and alternatives, such as MC or scalable BQ methods (e.g. Karvonen and Särkkä, 2018; Jagadeeswaran and Hickernell, 2019), are used on the cheaper levels. Finally, since we observed that the choice of point set had a large impact on performance, one could consider designing novel acquisition functions for adaptive experimental design (e.g. following the work of Ehara and Guillas, 2021).

Acknowledgements

The authors would like to thank Dimitra Salmadinou for support in accessing tsunami simulations, and Zhuo Sun for helpful discussions. KL and SG acknowledge funding from the Lloyd’s Tercentenary Research Foundation, the Lighthill Risk Network and the Lloyd’s Register Foundation-Data Centric Engineering Programme of the Alan Turing Institute for the project “Future Indonesian Tsunamis: Towards End-to-end Risk quantification (FITTER)”. SG also acknowledges support from The Alan Turing Institute project

“Uncertainty Quantification of multi-scale and multiphysics computer models: applications to hazard and climate models” under the EPSRC grant EP/N510129/1. DG and SG were supported by the EPSRC project EP/W007711/1 “Software Environment for Actionable & VVUQ-evaluated Exascale Application” (SEAVEA). TK was supported by the Academy of Finland postdoctoral researcher grant #338567 “Scalable, adaptive and reliable probabilistic integration”. FXB was supported by the Lloyd’s Register Foundation Programme on Data-Centric Engineering and The Alan Turing Institute under the EPSRC grant [EP/N510129/1], and through an Amazon Research Award on “Transfer Learning for Numerical Integration in Expensive Machine Learning Systems”.

References

- Acerbi, L. (2018). Variational Bayesian Monte Carlo. In *Neural Information Processing Systems*, pages 8223–8233.
- Adams, R. A. and Fournier, J. J. (2003). *Sobolev Spaces*. Elsevier.
- Arcangéli, R., de Silanes, M. C. L., and Torrens, J. J. (2007). An extension of a bound for functions in Sobolev spaces, with applications to (m, s) -spline interpolation and smoothing. *Numerische Mathematik*, 107(2):181–211.
- Aslett, L. J., Nagapetyan, T., and Vollmer, S. J. (2017). Multilevel Monte Carlo for reliability theory. *Reliability Engineering and System Safety*, 165:188–196.
- Behrens, D. and Dias, F. (2015). New computational methods in tsunami science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 373(20140382).
- Belhadji, A., Bardenet, R., and Chainais, P. (2019). Kernel quadrature with DPPs. In *Neural Information Processing Systems*, pages 12927–12937.
- Berlinet, A. and Thomas-Agnan, C. (2004). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Springer.
- Bharti, A., Naslidnyk, M., Key, O., Kaski, S., and Briol, F.-X. (2023). Optimally-weighted estimators of the maximum mean discrepancy for likelihood-free inference. *arXiv:2301.11674*.
- Briol, F.-X., Oates, C. J., Girolami, M., and Osborne, M. A. (2015). Frank-Wolfe Bayesian quadrature: Probabilistic integration with theoretical guarantees. In *Neural Information Processing Systems*, pages 1162–1170.
- 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.
- Brouillat, J., Bouville, C., Loos, B., Hansen, C., and Bouatouch, K. (2009). A Bayesian Monte Carlo approach to global illumination. *Computer Graphics Forum*, 28(8):2315–2329.
- Cadini, F. and Gioletta, A. (2016). A Bayesian Monte Carlo-based algorithm for the estimation of small failure probabilities of systems affected by uncertainties. *Reliability Engineering and System Safety*, 153:15–27.
- Clare, M. C. A., Piggott, M. D., and Cotter, C. J. (2022). Assessing erosion and flood risk in the coastal zone through the application of multilevel Monte Carlo methods. *Coastal Engineering*, 174(104118).
- Cockayne, J., Oates, C. J., Sullivan, T. J., and Girolami, M. (2019). Bayesian probabilistic numerical methods. *SIAM review*, 61(4):756–789.
- Degueudre, H., Metzger, J. J., Geisel, T., and Fleischmann, R. (2016). Random focusing of tsunami waves. *Nature Physics*, 12(3):259–262.
- Dempster, M. A. H., Kannianen, J., Keane, J., and Vynckier, E. (2018). *High-Performance Computing in Finance: Problems, Methods, and Solutions*. CRC Press.
- Diaconis, P. (1988). Bayesian numerical analysis. *Statistical Decision Theory and Related Topics IV*, pages 163–175.
- Dick, J., Kuo, F. Y., Le Gia, Q. T., and Schwab, C. (2016). Multilevel higher order QMC Petrov–Galerkin discretization for affine parametric operator equations. *SIAM Journal on Numerical Analysis*, 54(4):2541–2568.
- Dodwell, T. J., Ketelsen, C., Scheichl, R., and Teckentrup, A. L. (2019). Multilevel Markov chain Monte Carlo. *SIAM Review*, 61(3):509–545.
- Ehara, A. and Guillas, S. (2021). An adaptive strategy for sequential designs of multilevel computer experiments. *arXiv:2104.02037*.
- Fujisawa, M. and Sato, I. (2021). Multilevel Monte Carlo variational inference. *Journal of Machine Learning Research*, 22(278):1–44.
- Geraci, G., Eldred, M. S., and Iaccarino, G. (2017). A multifidelity multilevel Monte Carlo method for uncertainty propagation in aerospace applications. In *19th AIAA non-deterministic approaches conference*, page 1951.
- Giles, D., Gopinathan, D., Guillas, S., and Dias, F. (2021). Faster than real time tsunami warning with associated hazard uncertainties. *Frontiers in Earth Science*, 8:597865.
- Giles, D., Kashdan, E., Salmanidou, D. M., Guillas, S., and Dias, F. (2020). Performance analysis of Volna-OP2–massively parallel code for tsunami modelling. *Computers & Fluids*, 209:104649.
- Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, 24:259.
- Giles, M. B. and Waterhouse, B. J. (2009). Multilevel quasi-Monte Carlo path simulation. *Advanced Financial Modelling, Radon Series on Computational and Applied Mathematics*, 8:165–181.

- Goda, T., Hironaka, T., and Iwamoto, T. (2020). Multilevel Monte Carlo estimation of expected information gains. *Stochastic Analysis and Applications*, 38(4):581–600.
- Gopinathan, D., Heidarzadeh, M., and Guillas, S. (2021). Probabilistic quantification of tsunami current hazard using statistical emulation. *Proceedings of the Royal Society A*, 477(2250):20210180.
- Gregory, A. and Cotter, C. J. (2017). A seamless multi-level ensemble transform particle filter. *SIAM Journal on Scientific Computing*, 39(6):A2684–A2701.
- Gunter, T., Garnett, R., Osborne, M., Hennig, P., and Roberts, S. (2014). Sampling for inference in probabilistic models with fast Bayesian quadrature. In *Advances in Neural Information Processing Systems*, pages 2789–2797.
- Haji-Ali, A. L., Nobile, F., and Tempone, R. (2016). Multi-index Monte Carlo: when sparsity meets sampling. *Numerische Mathematik*, 132(4):767–806.
- Hennig, P., Osborne, M. A., and Girolami, M. (2015). Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142.
- Hennig, P., Osborne, M. A., and Kersting, H. (2022). *Probabilistic Numerics: Computation as Machine Learning*. Cambridge University Press.
- Jagadeeswaran, R. and Hickernell, F. J. (2019). Fast automatic Bayesian cubature using lattice sampling. *Statistics and Computing*, 29(6):1215–1229.
- Jasra, A., Jo, S., Nott, D., Shoemaker, C., and Tempone, R. (2019). Multilevel Monte Carlo in approximate Bayesian computation. *Stochastic Analysis and Applications*, 37(3):346–360.
- Kanagawa, M., Hennig, P., Sejdinovic, D., and Sriperumbudur, B. K. (2018). Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv:1807.02582*.
- Kanagawa, M., Sriperumbudur, B. K., and Fukumizu, K. (2020). Convergence analysis of deterministic kernel-based quadrature rules in misspecified settings. *Foundations of Computational Mathematics*, 20(1):155–194.
- Karvonen, T. (2019). *Kernel-based and Bayesian methods for numerical integration*. PhD thesis, Aalto University.
- Karvonen, T. and Särkkä, S. (2017). Classical quadrature rules via Gaussian processes. *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*.
- Karvonen, T. and Särkkä, S. (2018). Fully symmetric kernel quadrature. *SIAM Journal on Scientific Computing*, 40(2):697–720.
- Karvonen, T., Särkkä, S., and Oates, C. J. (2019). Symmetry exploits for Bayesian cubature methods. *Statistics and Computing*, 29:1231–1248.
- Karvonen, T., Wynne, G., Tronarp, F., Oates, C. J., and Särkkä, S. (2020). Maximum likelihood estimation and uncertainty quantification for gaussian process approximation of deterministic functions. *SIAM/ASA Journal on Uncertainty Quantification*, 8(3):926–958.
- Katsiolides, G., Müller, E. H., Scheichl, R., Shardlow, T., Giles, M. B., and Thomson, D. J. (2018). Multilevel Monte Carlo and improved timestepping methods in atmospheric dispersion modelling. *Journal of Computational Physics*, 354:320–343.
- Kirby, A., Briol, F.-X., Dunstan, T. D., and Nishino, T. (2023). Data-driven modelling of turbine wake interactions and flow resistance in large wind farms. *arXiv:2301.01699*.
- Krieg, D. and Sonnleitner, M. (2022). Random points are optimal for the approximation of Sobolev functions. *arXiv:arXiv:2009.11275v2*.
- Kuo, F., Scheichl, R., Schwab, C., Sloan, I., and Ullmann, E. (2017). Multilevel quasi-Monte Carlo methods for log-normal diffusion problems. *Mathematics of Computation*, 86(308):2827–2860.
- Kuo, F. Y., Schwab, C., and Sloan, I. H. (2015). Multi-level quasi-Monte Carlo finite element methods for a class of elliptic PDEs with random coefficients. *Foundations of Computational Mathematics*, 15(2):411–449.
- Lynett, P. and Liu, P. L.-F. (2005). A numerical study of the run-up generated by three-dimensional landslides. *Journal of Geophysical Research: Oceans*, 110(C3).
- Marras, S. and Mandli, K. T. (2021). Modeling and simulation of tsunami impact: A short review of recent advances and future challenges. *Geosciences*, 11(5).
- Mathews, J. and Walker, R. L. (1970). *Mathematical Methods of Physics*. W.A. Benjamin, 2nd edition.
- Narcowich, F. J., Ward, J. D., and Wendland, H. (2006). Sobolev error estimates and a Bernstein inequality for scattered data interpolation via radial basis functions. *Constructive Approximation*, 24(2):175–186.
- Oates, C. J., Niederer, S., Lee, A., Briol, F.-X., and Girolami, M. (2017). Probabilistic models for integration error in the assessment of functional cardiac models. In *Neural Information Processing Systems*, pages 110–118.
- Oetershagen, J. (2017). *Construction of Optimal Cubature Algorithms with Applications to Econometrics and Uncertainty Quantification*. PhD thesis, Rheinischen Friedrich-Wilhelms-Universität Bonn.
- O’Hagan, A. (1991). Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260.
- Osborne, M. A., Duvenaud, D., Garnett, R., Rasmussen, C. E., Roberts, S., and Ghahramani, Z. (2012). Active learning of model evidence using Bayesian quadrature. In *Advances In Neural Information Processing Systems*, pages 46–54.

- Owen, A. B. (2013). Monte Carlo theory, methods and examples.
- Park, H., Cox, D. T., Alam, M. S., and Barbosa, A. R. (2017). Probabilistic seismic and tsunami hazard analysis conditioned on a megathrust rupture of the Cascadia subduction zone. *Frontiers in built environment*, 3:32.
- Paul, S., Chatzilygeroudis, K., Ciosek, K., Mouret, J.-B., Osborne, M. A., and Whiteson, S. (2018). Alternating optimisation and quadrature for robust control. *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*.
- Peherstorfer, B., Willcox, K., and Gunzburger, M. (2018). Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *Siam Review*, 60(3):550–591.
- Rasmussen, C. E. and Ghahramani, Z. (2003). Bayesian Monte Carlo. *Advances in neural information processing systems*, pages 505–512.
- Reguly, I. Z., Giles, D., Gopinathan, D., Quivy, L., Beck, J. H., Giles, M. B., Guillas, S., and Dias, F. (2018). The VOLNA-OP2 tsunami code (version 1.5). *Geoscientific Model Development*, 11(11):4621–4635.
- Robert, C. P., Casella, G., and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer, 2nd edition.
- Rubinstein, R. Y. and Kroese, D. P. (2016). *Simulation and the Monte Carlo method*. John Wiley & Sons.
- Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical science*, 4(4):409–423.
- Salmanidou, D. M., Beck, J., Pazak, P., and Guillas, S. (2021). Probabilistic, high-resolution tsunami predictions in northern Cascadia by exploiting sequential design for efficient emulation. *Natural Hazards and Earth System Sciences*, 21(12):3789–3807.
- Sánchez-Linares, C., de la Asunción, M., Castro, M. J., González-Vida, J. M., Macías, J., and Mishra, S. (2016). Uncertainty quantification in tsunami modeling using multi-level Monte Carlo finite volume method. *Journal of Mathematics in Industry*, 6(1):1–26.
- Santner, T. J., Williams, B. J., and Notz, W. I. (2018). *The Design and Analysis of Computer Experiments*. Springer, 2nd edition.
- Shi, Y. and Cornish, R. (2021). On multilevel Monte Carlo unbiased gradient estimation for deep latent variable models. In *Artificial Intelligence and Statistics*, volume 130, pages 3925–3933.
- Stein, M. L. (1999). *Interpolation of Spatial Data: Some Theory for Kriging*. Springer Science & Business Media.
- Teckentrup, A. L. (2020). Convergence of Gaussian process regression with estimated hyper-parameters and applications in Bayesian inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, 8(4):1310–1337.
- Tuo, R., Wang, Y., and Jeff Wu, C. (2020). On the improved rates of convergence for Matérn-type kernel ridge regression with application to calibration of computer models. *SIAM/ASA Journal on Uncertainty Quantification*, 8(4):1522–1547.
- Wang, G. and Wang, T. (2022). Unbiased multilevel Monte Carlo methods for intractable distributions: MLMC meets MCMC. *arXiv:2204.04808*.
- Warne, D. J., Baker, R. E., and Simpson, M. J. (2019). Simulation and inference algorithms for stochastic biochemical reaction networks: from basic concepts to state-of-the-art. *Journal of the Royal Society Interface*, 16(151).
- Wendland, H. (2004). *Scattered Data Approximation*, volume 17. Cambridge University Press.
- Wenger, J., Krämer, N., Pförtner, M., Schmidt, J., Bosch, N., Effenberger, N., Zenn, J., Gessner, A., Karvonen, T., Briol, F.-X., Mahsereci, M., and Hennig, P. (2021). ProbNum: Probabilistic numerics in Python. *arXiv:2112.02100*.
- Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian Processes for Machine Learning*. MIT Press Cambridge, MA.
- Wynne, G., Briol, F.-X., and Girolami, M. (2021). Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. *Journal of Machine Learning Research*, 22.
- Xi, X., Briol, F.-X., and Girolami, M. (2018). Bayesian quadrature for multiple related integrals. In *35th International Conference on Machine Learning*, volume 12, pages 8533–8564.

Supplementary Material

In Appendix A, we discuss the broader impact of our work. In Appendix B, we provide the proofs of all theoretical results in the main text. In Appendix C, we provide details on the experiments introduced in the main text.

A BROADER IMPACT

Our paper focuses on numerical integration, a common computational problem in statistics and machine learning. Our proposed approach improves the accuracy of approximations and provides probabilistic uncertainty quantification for the value of the integral given limited function evaluations. Our experiments show that applying our approach has the potential to reduce financial and time costs in applications in environmental science and engineering, where high-performance computing clusters are widely used.

However, it is the specific application that is the determining factor in the broader impact. It depends on whether the user uses the approach for socially beneficial research to have a positive impact. For example, in the landslide tsunami example studied in the experiments section, we showed that the computational requirements can be reduced by using our approach to get useful approximations for tsunami researchers, e.g. designers of tsunami resistant buildings or prevention structures.

B PROOFS

In this section, we provide the proofs of all theoretical results in the main text. This includes the proof of Proposition 1 in Appendix B.2, the extension in Appendix B.3, the proof of Theorem 2 in Appendix B.4 and the proof of Theorem 3 in Appendix B.5. Additionally, for completeness we recall a well-known derivation of the optimal sample size for MLMC in Appendix B.1.

B.1 Optimal Sample Size for MLMC Given Cost Constraint

The optimal sample size $n_0^{\text{MLMC}}, \dots, n_L^{\text{MLMC}}$ that minimize the MSE of MLMC estimates with an overall cost constraint T is the solution to the problem

$$n_0^{\text{MLMC}}, \dots, n_L^{\text{MLMC}} := \underset{n_0, n_1, \dots, n_L}{\operatorname{argmin}} \sum_{l=0}^L V_l n_l^{-1} \quad \text{s.t.} \quad \sum_{l=0}^L C_l n_l = T.$$

In this section, we show how the equation above can be solved by using Lagrange multipliers. For some $\lambda > 0$, define

$$F_{\text{MLMC}}(n_0, \dots, n_L, \lambda) = \sum_{l=0}^L V_l n_l^{-1} - \lambda(T - \sum_{l=0}^L C_l n_l).$$

By taking the derivative of $F_{\text{MLMC}}(n_0, \dots, n_L, \lambda)$ with respect to n_0, \dots, n_L, λ and setting these equal to 0, we have

$$-V_l n_l^{-2} + \lambda C_l = 0 \quad \Leftrightarrow \quad n_l = \left(\frac{\lambda C_l}{V_l} \right)^{-\frac{1}{2}} \quad \text{for } l \in \{0, \dots, L\} \quad \text{and} \quad \sum_{l=0}^L C_l n_l = T.$$

By plugging the first equation into the second, we have

$$\sum_{l=0}^L C_l \left(\frac{\lambda C_l}{V_l} \right)^{-\frac{1}{2}} = T \quad \Leftrightarrow \quad \lambda^{-\frac{1}{2}} = \frac{T}{\sum_{l=0}^L C_l \left(\frac{C_l}{V_l} \right)^{-\frac{1}{2}}} \quad \Leftrightarrow \quad \lambda = \left(\frac{1}{T} \sum_{l=0}^L C_l \left(\frac{C_l}{V_l} \right)^{-\frac{1}{2}} \right)^2.$$

Finally, plugging this expression for λ into our expression for n_l , we get

$$n_l^{\text{MLMC}} = T \sqrt{\frac{V_l}{C_l}} \left(\sum_{l'=0}^L \sqrt{V_{l'} C_{l'}} \right)^{-1} \quad \text{for } l \in \{0, \dots, L\}.$$

B.2 Proof of Proposition 1

Proof of Proposition 1. If $f_l - f_{l-1}$ for $l \in \{0, \dots, L\}$ are a-priori independent, and $f_l - f_{l-1} \sim \mathcal{GP}(m_l, c_l)$, then

$$\begin{bmatrix} f_0(W_0) \\ f_1(W_1) - f_0(W_1) \\ \vdots \\ f_L(W_L) - f_{L-1}(W_L) \\ \sum_{l=0}^L (f_l(W_*) - f_{l-1}(W_*)) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m_0(W_0) \\ m_1(W_1) \\ \vdots \\ m_L(W_L) \\ \sum_{l=0}^L m_l(W_*) \end{bmatrix}, \begin{bmatrix} c_0(W_0, W_0) & 0 & \cdots & 0 & c_0(W_0, W_*) \\ 0 & c_1(W_1, W_1) & \cdots & 0 & c_1(W_1, W_*) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & c_L(W_L, W_L) & c_L(W_L, W_*) \\ c_0(W_*, W_0) & c_1(W_*, W_1) & \cdots & c_L(W_*, W_L) & \sum_{l=0}^L c_l(W_*, W_*) \end{bmatrix} \right),$$

where $W_* = (\omega_{(*,1)}, \dots, \omega_{(*,n_*)})^\top$ are query locations. Applying the formula for the conditional distribution from the multivariate Gaussian distribution, given the vector values $(f_0(W_0), f_1(W_1) - f_0(W_1), \dots, f_L(W_L) - f_{L-1}(W_L))^\top$, the conditional distribution of $\sum_{l=0}^L (f_l(W_*) - f_{l-1}(W_*))$ is Gaussian, with mean

$$\begin{aligned} & \sum_{l=0}^L m_l(W_*) + [c_0(W_*, W_0) \quad c_1(W_*, W_1) \quad \cdots \quad c_L(W_*, W_L)] \\ & \quad \times \begin{bmatrix} c_0(W_0, W_0) & 0 & \cdots & 0 \\ 0 & c_1(W_1, W_1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_L(W_L, W_L) \end{bmatrix}^{-1} \begin{bmatrix} f_0(W_0) - m_0(W_0) \\ f_1(W_1) - f_0(W_1) - m_1(W_1) \\ \vdots \\ f_L(W_L) - f_{L-1}(W_L) - m_L(W_L) \end{bmatrix} \\ & = \sum_{l=0}^L (m_l(W_*) + c_l(W_*, W_l) c_l(W_l, W_l)^{-1} (f_l(W_l) - f_{l-1}(W_l) - m_l(W_l))) \\ & = \sum_{l=0}^L \tilde{m}_l(W_*), \end{aligned}$$

and variance

$$\begin{aligned} & \sum_{l=0}^L c_l(W_*, W_*) + [c_0(W_*, W_0) \quad c_1(W_*, W_1) \quad \cdots \quad c_L(W_*, W_L)] \\ & \quad \times \begin{bmatrix} c_0(W_0, W_0) & 0 & \cdots & 0 \\ 0 & c_1(W_1, W_1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_L(W_L, W_L) \end{bmatrix}^{-1} \begin{bmatrix} c_0(W_0, W_*) \\ c_1(W_1, W_*) \\ \vdots \\ c_L(W_L, W_*) \end{bmatrix} \\ & = \sum_{l=0}^L (c_l(W_*, W_*) + c_l(W_*, W_l) c_l(W_l, W_l)^{-1} c_l(W_l, W_*)) \\ & = \sum_{l=0}^L \tilde{c}_l(W_*, W_*). \end{aligned}$$

As a result, the posterior on $\sum_{l=0}^L (f_l - f_{l-1})$ is $\mathcal{GP}(\sum_{l=0}^L \tilde{m}_l, \sum_{l=0}^L \tilde{c}_l)$. The posterior on $\sum_{l=0}^L \Pi[f_l - f_{l-1}]$ can be obtained (following the usual derivation for the BQ distribution on integrals) integrating the posterior mean and covariance functions and takes the form of a univariate Gaussian with mean

$$\begin{aligned} \mathbb{E}_{\text{MLBQ}}[\Pi[f]] & = \sum_{l=0}^L (\Pi[m_l] + \Pi[c_l(\cdot, W_l)] c_l(W_l, W_l)^{-1} (f_l(W_l) - f_{l-1}(W_l) - m_l(W_l))) \\ & = \sum_{l=0}^L \hat{\Pi}_{\text{BQ}}[f_l - f_{l-1}] \end{aligned}$$

and variance

$$\begin{aligned} \mathbb{V}_{\text{MLBQ}}[\Pi[f]] & = \sum_{l=0}^L (\Pi[\Pi[c_l]] - \Pi[c_l(\cdot, W_l)] c_l(W_l, W_l)^{-1} \Pi[c_l(W_l, \cdot)]) \\ & = \sum_{l=0}^L \mathbb{V}_{\text{BQ}}[\Pi[f_l - f_{l-1}]]. \end{aligned}$$

□

B.3 Extension of Proposition 1

For the vector-valued function $(f_0, f_1 - f_0, \dots, f_L - f_{L-1})^\top$, suppose we specify a separable kernel $C(\omega, \omega') = Bc(\omega, \omega')$, where $B \in \mathbb{R}^{L \times L}$ is symmetric and positive definite with $B_{i,j}$ denoting the (i, j) -entry of B , then

$$\begin{bmatrix} f_0(W_0) \\ f_1(W_1) - f_0(W_1) \\ \vdots \\ f_L(W_L) - f_{L-1}(W_L) \\ \sum_{l=0}^L (f_l(W_*) - f_{l-1}(W_*)) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m_0(W_0) \\ m_1(W_1) \\ \vdots \\ m_L(W_L) \\ \sum_{l=0}^L m_l(W_*) \end{bmatrix}, \begin{bmatrix} B_{0,0}c(W_0, W_0) & B_{0,1}c(W_0, W_1) & \cdots & B_{0,L}c(W_0, W_L) & \sum_{l=0}^L B_{0,l}c(W_0, W_*) \\ B_{1,0}c(W_1, W_0) & B_{1,1}c(W_1, W_1) & \cdots & B_{1,L}c(W_1, W_L) & \sum_{l=0}^L B_{1,l}c(W_1, W_*) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ B_{L,0}c(W_L, W_0) & B_{L,1}c(W_L, W_1) & \cdots & B_{L,L}c(W_L, W_L) & \sum_{l=0}^L B_{L,l}c(W_L, W_*) \\ \sum_{l=0}^L B_{l,0}c(W_*, W_0) & \sum_{l=0}^L B_{l,1}c(W_*, W_1) & \cdots & \sum_{l=0}^L B_{l,L}c(W_*, W_L) & \sum_{l=0}^L \sum_{l'=0}^L B_{l,l'}c(W_*, W_*) \end{bmatrix} \right).$$

Similarly, applying the formula for Gaussian conditionals, given the vector values $(f_0(W_0), f_1(W_1) - f_0(W_1), \dots, f_L(W_L) - f_{L-1}(W_L))^\top$, the conditional distribution of $\sum_{l=0}^L (f_l(W_*) - f_{l-1}(W_*))$ is Gaussian, with mean

$$\begin{aligned} \tilde{m}^B(W_*) &= \sum_{l=0}^L m_l(W_*) + \left[\sum_{l=0}^L B_{l,0}c(W_*, W_0) \quad \sum_{l=0}^L B_{l,1}c(W_*, W_1) \quad \cdots \quad \sum_{l=0}^L B_{l,L}c(W_*, W_L) \right] \\ &\quad \times \begin{bmatrix} B_{0,0}c(W_0, W_0) & B_{0,1}c(W_0, W_1) & \cdots & B_{0,L}c(W_0, W_L) \\ B_{1,0}c(W_1, W_0) & B_{1,1}c(W_1, W_1) & \cdots & B_{1,L}c(W_1, W_L) \\ \vdots & \vdots & \ddots & \vdots \\ B_{L,0}c(W_L, W_0) & B_{L,1}c(W_L, W_1) & \cdots & B_{L,L}c(W_L, W_L) \end{bmatrix}^{-1} \begin{bmatrix} f_0(W_0) - m_0(W_0) \\ f_1(W_1) - f_0(W_1) - m_1(W_1) \\ \vdots \\ f_L(W_L) - f_{L-1}(W_L) - m_L(W_L) \end{bmatrix} \end{aligned}$$

and variance

$$\begin{aligned} \tilde{c}^B(W_*, W_*) &= \sum_{l=0}^L \sum_{l'=0}^L B_{l,l'}c(W_*, W_*) \\ &\quad + \left[\sum_{l=0}^L B_{l,0}c(W_*, W_0) \quad \sum_{l=0}^L B_{l,1}c(W_*, W_1) \quad \cdots \quad \sum_{l=0}^L B_{l,L}c(W_*, W_L) \right] \\ &\quad \times \begin{bmatrix} B_{0,0}c(W_0, W_0) & B_{0,1}c(W_0, W_1) & \cdots & B_{0,L}c(W_0, W_L) \\ B_{1,0}c(W_1, W_0) & B_{1,1}c(W_1, W_1) & \cdots & B_{1,L}c(W_1, W_L) \\ \vdots & \vdots & \ddots & \vdots \\ B_{L,0}c(W_L, W_0) & B_{L,1}c(W_L, W_1) & \cdots & B_{L,L}c(W_L, W_L) \end{bmatrix}^{-1} \begin{bmatrix} \sum_{l=0}^L B_{0,l}c(W_0, W_*) \\ \sum_{l=0}^L B_{1,l}c(W_1, W_*) \\ \vdots \\ \sum_{l=0}^L B_{L,l}c(W_L, W_*) \end{bmatrix}. \end{aligned}$$

As a result, the posterior on $\sum_{l=0}^L (f_l - f_{l-1})$ is $\mathcal{GP}(\tilde{m}^B, \tilde{c}^B)$. Similarly, the posterior on $\sum_{l=0}^L \Pi[f_l - f_{l-1}]$ can be obtained (following the usual derivation for the BQ distribution on integrals) integrating the posterior mean and covariance functions and takes the form of a univariate Gaussian with mean

$$\begin{aligned} \sum_{l=0}^L \Pi[m_l] &+ \left[\sum_{l=0}^L B_{l,0}\Pi[c(\cdot, W_0)] \quad \sum_{l=0}^L B_{l,1}\Pi[c(\cdot, W_1)] \quad \cdots \quad \sum_{l=0}^L B_{l,L}\Pi[c(\cdot, W_L)] \right] \\ &\quad \times \begin{bmatrix} B_{0,0}c(W_0, W_0) & B_{0,1}c(W_0, W_1) & \cdots & B_{0,L}c(W_0, W_L) \\ B_{1,0}c(W_1, W_0) & B_{1,1}c(W_1, W_1) & \cdots & B_{1,L}c(W_1, W_L) \\ \vdots & \vdots & \ddots & \vdots \\ B_{L,0}c(W_L, W_0) & B_{L,1}c(W_L, W_1) & \cdots & B_{L,L}c(W_L, W_L) \end{bmatrix}^{-1} \begin{bmatrix} f_0(W_0) - m_0(W_0) \\ f_1(W_1) - f_0(W_1) - m_1(W_1) \\ \vdots \\ f_L(W_L) - f_{L-1}(W_L) - m_L(W_L) \end{bmatrix} \end{aligned}$$

and variance

$$\begin{aligned} \sum_{l=0}^L \sum_{l'=0}^L B_{l,l'}\Pi[\Pi[c]] &+ \left[\sum_{l=0}^L B_{l,0}\Pi[c(\cdot, W_0)] \quad \sum_{l=0}^L B_{l,1}\Pi[c(\cdot, W_1)] \quad \cdots \quad \sum_{l=0}^L B_{l,L}\Pi[c(\cdot, W_L)] \right] \\ &\quad \times \begin{bmatrix} B_{0,0}c(W_0, W_0) & B_{0,1}c(W_0, W_1) & \cdots & B_{0,L}c(W_0, W_L) \\ B_{1,0}c(W_1, W_0) & B_{1,1}c(W_1, W_1) & \cdots & B_{1,L}c(W_1, W_L) \\ \vdots & \vdots & \ddots & \vdots \\ B_{L,0}c(W_L, W_0) & B_{L,1}c(W_L, W_1) & \cdots & B_{L,L}c(W_L, W_L) \end{bmatrix}^{-1} \begin{bmatrix} \sum_{l=0}^L B_{0,l}\Pi[c(W_0, \cdot)] \\ \sum_{l=0}^L B_{1,l}\Pi[c(W_1, \cdot)] \\ \vdots \\ \sum_{l=0}^L B_{L,l}\Pi[c(W_L, \cdot)] \end{bmatrix}. \end{aligned}$$

An example is provided in Appendix C.1.

B.4 Proof of Theorem 2

Proof of Theorem 2. Suppose that c is a covariance function such that $\mathcal{H}(c)$ is norm-equivalent to $W_2^\alpha(\Omega)$ and $f \in W_2^\beta(\Omega)$ for $\alpha \geq \beta > d/2$. Since the density π of Π is bounded by Assumption A2, we have

$$|\Pi[f] - \hat{\Pi}_{\text{BQ}}[f]| = \left| \int_{\Omega} f(\omega)\pi(\omega)d\omega - \int_{\Omega} \tilde{m}(\omega)\pi(\omega)d\omega \right| \leq \|\pi\|_{L^\infty(\Omega)} \int_{\Omega} |f(\omega) - \tilde{m}(\omega)|d\omega,$$

where \tilde{m} is the GP posterior mean given observations of f at n points $W = (\omega_1, \dots, \omega_n)^\top$ and with $m \equiv 0$. Because $c(\cdot, \omega) \in \mathcal{H}(c) = W_2^\alpha(\Omega) \subset W_2^\beta(\Omega)$, the posterior mean is an element of $W_2^\beta(\Omega)$. Assumption A1 ensures that Ω satisfies the assumptions of Theorem 4.1 in Arcangéli et al. (2007). Using this theorem with $p = 2$, $q = 1$, $l = 0$, $r = \beta$, and $n = d$ gives

$$\int_{\Omega} |f(\omega) - \tilde{m}(\omega)|d\omega \leq \tilde{a}h_{W,\Omega}^\beta \|f - \tilde{m}\|_\beta$$

whenever $h_{W,\Omega}$ is sufficiently small. The positive constant \tilde{a} depends only on α , β , d , and Ω . Theorem 4.2 in Narcowich et al. (2006) with $\mu = \beta$ and $\tau = \alpha$ and the well known identification of the GP posterior mean with the minimum-norm kernel interpolant (e.g., Kanagawa et al., 2018, Section 3) yield

$$\|f - \tilde{m}\|_\beta \leq a' \rho_{W,\Omega}^{\alpha-\beta} \|f\|_\beta,$$

where $\rho_{W,\Omega} = h_{W,\Omega}/q_X$ for $q_X = \frac{1}{2} \min_{i \neq j} \|\omega_i - \omega_j\|$ is known as the mesh ratio and a' is a positive constant that does not depend on f . Assume that the points W are quasi-uniform with the constant $h_{\text{qu}} > 0$, which is to say that $h_{W,\Omega} \leq h_{\text{qu}} n^{-1/d}$. This implies that $\rho_{W,\Omega} \leq \tilde{h}_{\text{qu}}$ for a constant \tilde{h}_{qu} that depends on h_{qu} , d , and Ω . Combining all the bounds above yields

$$|\Pi[f] - \hat{\Pi}_{\text{BQ}}[f]| \leq \|\pi\|_{L^\infty(\Omega)} \tilde{a} h_{W,\Omega}^\beta \|f - \tilde{m}\|_\beta \leq \|\pi\|_{L^\infty(\Omega)} \tilde{a} h_{\text{qu}}^\beta n^{-\beta/d} a' \tilde{h}_{\text{qu}}^{\alpha-\beta} \|f\|_\beta =: \|\pi\|_{L^\infty(\Omega)} a \|f\|_\beta n^{-\beta/d} \quad (3)$$

for a positive constant a that depends only on α , β , c , h_{qu} , d , and Ω . If $f \in W_2^\beta(\Omega)$ for $\beta > \alpha > d/2$, we use the fact that in this case $f \in W_2^\alpha(\Omega)$ and obtain (3) with $\beta = \alpha$. Let $\tau = \min\{\alpha, \beta\}$. We can then write

$$|\Pi[f] - \hat{\Pi}_{\text{BQ}}[f]| \leq \|\pi\|_{L^\infty(\Omega)} a \|f\|_\tau n^{-\tau/d} \quad (4)$$

for any reals α and β that exceed $d/2$.

We then apply (4) to each term in MLBQ error under Assumptions A1–A6 (recall the convention $f_{-1} \equiv 0$):

$$\begin{aligned} \text{Err}(\hat{\Pi}_{\text{MLBQ}}) &= |\Pi[f_L] - \hat{\Pi}_{\text{MLBQ}}[f_L]| = \left| \Pi[f_0] + \sum_{l=1}^L \Pi[f_l - f_{l-1}] - \hat{\Pi}_{\text{BQ}}[f_0] - \sum_{l=1}^L \hat{\Pi}_{\text{BQ}}[f_l - f_{l-1}] \right| \\ &\leq \sum_{l=0}^L |\Pi[f_l - f_{l-1}] - \hat{\Pi}_{\text{BQ}}[f_l - f_{l-1}]| \\ &\leq \|\pi\|_{L^\infty(\Omega)} \sum_{l=0}^L a_l \|f_l - f_{l-1}\|_\tau n_l^{-\tau/d}, \end{aligned}$$

where Assumption A4 ensures that $f_l - f_{l-1} \in W_2^{\beta_l}(\Omega) \subset W_2^{\tau_l}(\Omega)$. \square

B.5 Proof of Theorem 3

Proof of Theorem 3. The sample sizes $n_0^{\text{MLBQ}}, \dots, n_L^{\text{MLBQ}}$ that minimize the upper bound of the absolute error of MLBQ in Theorem 2 with the overall cost constraint T are

$$n_0^{\text{MLBQ}}, \dots, n_L^{\text{MLBQ}} := \underset{n_0, n_1, \dots, n_L}{\text{argmin}} \sum_{l=0}^L A_l n_l^{-\frac{\tau}{d}} \quad \text{s.t.} \quad \gamma \sum_{l'=0}^L C_{l'} n_{l'} = T,$$

where $A_l = a \|f_l - f_{l-1}\|_\tau$. We note that the $\|\pi\|_{L^\infty(\Omega)}$ term can be ignored since it does not depend on the sample sizes. Similarly to the derivation for MLMC in Appendix B.1, the optimisation problem above can be solved by using Lagrange multipliers. For some $\lambda > 0$, we define

$$F_{\text{MLBQ}}(n_0, \dots, n_L, \lambda) = \sum_{l=0}^L A_l n_l^{-\frac{\tau}{d}} - \lambda (T - \gamma \sum_{l'=0}^L C_{l'} n_{l'}). \quad (5)$$

Differentiating $F_{\text{MLBQ}}(n_0, \dots, n_L, \lambda)$ with respect to n_0, \dots, n_L, λ and setting the equations equal to 0 gives

$$-\frac{\tau}{d} A_l n_l^{-\frac{\tau}{d}-1} + \lambda \gamma C_l = 0 \quad \Leftrightarrow \quad n_l = \left(\frac{d\lambda \gamma C_l}{\tau A_l} \right)^{-\frac{d}{\tau+d}} \quad \text{for } l \in \{0, \dots, L\} \quad \text{and} \quad \gamma \sum_{l'=0}^L C_{l'} n_{l'} = T.$$

By plugging the first equation into the second, we get

$$\sum_{l'=0}^L \gamma C_{l'} \left(\frac{d\lambda \gamma C_{l'}}{\tau A_{l'}} \right)^{-\frac{d}{\tau+d}} = T \quad \Leftrightarrow \quad \lambda = T^{-\frac{\tau+d}{d}} \left(\sum_{l'=0}^L \gamma C_{l'} \left(\frac{d\gamma C_{l'}}{\tau A_{l'}} \right)^{-\frac{d}{\tau+d}} \right)^{\frac{\tau+d}{d}}.$$

Plugging this last expression for λ into our expression for n_l , we get

$$\begin{aligned} n_l^{\text{MLBQ}} &= \frac{T}{\gamma} \left(\frac{C_l}{A_l} \right)^{-\frac{d}{\tau+d}} \left(\sum_{l'=0}^L C_{l'} \left(\frac{C_{l'}}{A_{l'}} \right)^{-\frac{d}{\tau+d}} \right)^{-1} \\ &= \frac{T}{\gamma} \left(\frac{\|f_l - f_{l-1}\|_{\tau}}{C_l} \right)^{\frac{d}{\tau+d}} \left(\sum_{l'=0}^L C_{l'}^{\frac{\tau}{\tau+d}} (\|f_{l'} - f_{l'-1}\|_{\tau})^{\frac{d}{\tau+d}} \right)^{-1} \quad \text{for } l \in \{0, \dots, L\}. \end{aligned}$$

□

C ADDITIONAL EXPERIMENTS

In this section, we provide details of the experimental setup and additional experiments. This includes details of the three experiments in main text in Appendix C.1, Appendix C.2 and Appendix C.3, additional experiments in Appendix C.1, Appendix C.4 and Appendix C.5 and the analytical formulae for the kernel mean and initial error in Appendix C.6.

C.1 Experiment 1: Poisson Equation

Construction of the Levels Given the specific example of Poisson equation under consideration, we are able to obtain a closed form solution to the PDE: $f(\omega) = \frac{1}{2}\omega(\omega - 1)$. We construct a piecewise linear finite element approximation f_l of the solution f on level l as follows. Let $p_l \in \mathbb{N}$ and $0 < \omega_{l,1} < \dots < \omega_{l,p_l} < 1$. Define the piecewise linear finite element basis functions as

$$v_{l,i}(\omega) = \begin{cases} \frac{\omega - \omega_{l,i-1}}{\omega_{l,i} - \omega_{l,i-1}} & \text{if } \omega \in [\omega_{l,i-1}, \omega_{l,i}], \\ \frac{\omega_{l,i+1} - \omega}{\omega_{l,i+1} - \omega_{l,i}} & \text{if } \omega \in [\omega_{l,i}, \omega_{l,i+1}], \\ 0 & \text{otherwise.} \end{cases}$$

The i th basis function is supported on $[\omega_{l,i-1}, \omega_{l,i+1}]$. Here we use the conventions $\omega_{l,0} = 0$ and $\omega_{l,p_l+1} = 1$. The finite element approximation f_l to f is given by $f_l(\omega) = \sum_{i=1}^{p_l} a_{l,i} v_{l,i}(\omega)$, where the coefficient vector $a_l = (a_{l,1}, \dots, a_{l,p_l})^\top \in \mathbb{R}^{p_l}$ is solved from the linear system $-L_l a_l = g_l$, where $L_l \in \mathbb{R}^{p_l \times p_l}$ is the tridiagonal stiffness matrix with

$$(L_l)_{i,i} = \int_0^1 v'_{l,i}(\omega)^2 d\omega = \frac{1}{\omega_{l,i} - \omega_{l,i-1}} + \frac{1}{\omega_{l,i+1} - \omega_{l,i}}, \quad (L_l)_{i,i-1} = (L_l)_{i-1,i} = \int_0^1 v'_{l,i}(\omega) v'_{l,i-1}(\omega) d\omega = -\frac{1}{\omega_{l,i} - \omega_{l,i-1}},$$

and the vector $g_l \in \mathbb{R}^{p_l}$ has elements $(g_l)_i = \int_0^1 f(\omega) v_{l,i}(\omega) d\omega$. Consider now the Brownian motion kernel $c_l(\omega, \omega') = \sigma_l^2 c_{\text{BM}}(\omega, \omega') = \sigma_l^2 \min\{\omega, \omega'\}$, for a positive amplitude parameter σ_l , where the RKHS of the Brownian motion kernel $c_l(\omega, \omega')$ on $\Omega = [0, 1]$ consists of functions $g_l(0) = 0$ and $f_l \in W_2^1([0, 1])$ (Karvonen et al., 2020). It is straightforward to verify that a piecewise linear finite element basis function can be written in terms of the Brownian motion kernel translates:

$$v_{l,i}(\omega) = -\frac{1}{\omega_{l,i} - \omega_{l,i-1}} c_{\text{BM}}(\omega, \omega_{l,i-1}) + \left(\frac{1}{\omega_{l,i} - \omega_{l,i-1}} + \frac{1}{\omega_{l,i+1} - \omega_{l,i}} \right) c_{\text{BM}}(\omega, \omega_{l,i}) - \frac{1}{\omega_{l,i+1} - \omega_{l,i}} c_{\text{BM}}(\omega, \omega_{l,i+1})$$

and the full finite element approximation is

$$\begin{aligned} f_l(\omega) &= \sum_{i=1}^{p_l} a_{l,i} v_{l,i}(\omega) \\ &= \sum_{i=1}^{p_l} a_{l,i} \left[-\frac{1}{\omega_{l,i} - \omega_{l,i-1}} c_{\text{BM}}(\omega, \omega_{l,i-1}) + \left(\frac{1}{\omega_{l,i} - \omega_{l,i-1}} + \frac{1}{\omega_{l,i+1} - \omega_{l,i}} \right) c_{\text{BM}}(\omega, \omega_{l,i}) - \frac{1}{\omega_{l,i+1} - \omega_{l,i}} c_{\text{BM}}(\omega, \omega_{l,i+1}) \right] \\ &= \left(\frac{a_{l,1}}{\omega_{l,1}} + \frac{a_{l,1} - a_{l,2}}{\omega_{l,2} - \omega_{l,1}} \right) c_{\text{BM}}(\omega, \omega_{l,1}) + \left(\frac{a_{l,p_l}}{1 - \omega_{l,p_l}} + \frac{a_{l,p_l} - a_{l,p_l-1}}{\omega_{l,p_l} - \omega_{l,p_l-1}} \right) c_{\text{BM}}(\omega, \omega_{l,p_l}) - \frac{a_{l,p_l}}{1 - \omega_{l,p_l}} c_{\text{BM}}(\omega, 1) \\ &\quad + \sum_{i=2}^{p_l-1} \left(\frac{a_{l,i} - a_{l,i-1}}{\omega_{l,i} - \omega_{l,i-1}} + \frac{a_{l,i} - a_{l,i+1}}{\omega_{l,i+1} - \omega_{l,i}} \right) c_{\text{BM}}(\omega, \omega_{l,i}), \end{aligned}$$

where we have used the fact that $c_{\text{BM}}(\omega, \omega_0) = c_{\text{BM}}(\omega, 0) = 0$ for all $\omega \geq 0$ and $\omega_{l,p_l+1} = 1$. For simplicity, suppose that the points are equispaced on $[0, 1]$ so that $\Delta_l = \omega_{l,1} = \omega_{l,i} - \omega_{l,i-1} = 1 - \omega_{l,p_l}$ for every $i = 1, \dots, p_l$. Then the finite element approximation simplifies to

$$\begin{aligned} f_l(\omega) &= \frac{2a_{l,1} - a_{l,2}}{\Delta_l} c_{\text{BM}}(\omega, \omega_{l,1}) + \frac{2a_{l,p_l} - a_{l,p_l-1}}{\Delta_l} c_{\text{BM}}(\omega, \omega_{l,p_l}) - \frac{a_{l,p_l}}{\Delta_l} c_{\text{BM}}(\omega, 1) \\ &\quad + \frac{1}{\Delta_l} \sum_{i=2}^{p_l-1} (2a_{l,i} - a_{l,i-1} - a_{l,i+1}) c_{\text{BM}}(\omega, \omega_{l,i}) \\ &= \frac{1}{\Delta_l} [-a_{l,p_l} c_{\text{BM}}(\omega, 1) + \sum_{i=1}^{p_l} (2a_{l,i} - a_{l,i-1} - a_{l,i+1}) c_{\text{BM}}(\omega, \omega_{l,i})] \\ &= \frac{1}{\Delta_l \sigma_l^2} [-a_{l,p_l} c_l(\omega, 1) + \sum_{i=1}^{p_l} (2a_{l,i} - a_{l,i-1} - a_{l,i+1}) c_l(\omega, \omega_{l,i})] \end{aligned}$$

where we use the convention $a_0 = a_{l,p_l+1} = 0$. Denote $b_{l,i} = 2a_{l,i} - a_{l,i-1} - a_{l,i+1}$. Using the above expression for f_l as a sum of kernel translates and the general formula $\|\sum_{i=1}^n \alpha_i c(\cdot, \omega_i)\|_{\mathcal{H}}^2 = \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j c(\omega_i, \omega_j)$ we are able to compute the squared RKHS norm:

$$\begin{aligned} \|f_l\|_{\mathcal{H}_l}^2 &= \frac{1}{\Delta_l^2} \left[a_{l,p_l}^2 c_{\text{BM}}(1, 1) - 2a_{l,p_l} \sum_{i=1}^{p_l} b_{l,i} c_{\text{BM}}(1, \omega_{l,i}) + \sum_{i=1}^{p_l} \sum_{j=1}^{p_l} b_{l,i} b_{l,j} c_{\text{BM}}(\omega_{l,i}, \omega_{l,j}) \right] \\ &= \frac{1}{\Delta_l^2} \left[a_{l,p_l}^2 - 2a_{l,p_l} \sum_{i=1}^{p_l} b_{l,i} \omega_{l,i} + \sum_{i=1}^{p_l} \sum_{j=1}^{p_l} b_{l,i} b_{l,j} c_{\text{BM}}(\omega_{l,i}, \omega_{l,j}) \right]. \end{aligned}$$

We can compute the norm $\|f_l - f_{l-1}\|_{\mathcal{H}_l}^2$ in a similar way.

$$\begin{aligned} \|f_l - f_{l-1}\|_{\mathcal{H}_l}^2 &= \|f_l\|_{\mathcal{H}_l}^2 + \|f_{l-1}\|_{\mathcal{H}_l}^2 + \frac{2}{\Delta_l \Delta_{l-1}} \left[-a_{l,p_l} a_{l-1,p_{l-1}} + a_{l,p_l} \sum_{i=1}^{p_{l-1}} b_{l-1,i} c_{\text{BM}}(1, \omega_{l-1,i}) \right. \\ &\quad \left. + a_{l-1,p_{l-1}} \sum_{i=1}^{p_l} b_{l,i} c_{\text{BM}}(1, \omega_{l,i}) - \sum_{i=1}^{p_{l-1}} \sum_{j=1}^{p_l} b_{l-1,i} b_{l,j} c_{\text{BM}}(\omega_{l-1,i}, \omega_{l,j}) \right]. \end{aligned}$$

We used a fixed grid to pick the quadrature point. This will mean that A5 is satisfied. A1–A4 will be trivially satisfied, and A4 can be checked according to the derivation above. Since we used Matérn 0.5 kernel, the smoothness is the same as the Brownian motion kernel, we discard the influence of unknown constants and calculate the optimal sample size for MLBQ.

Experimental Settings We used a Matérn kernel with smoothness $\nu = 0.5$ and all the formulae for the kernel mean and initial error are provided in Appendix C.6. The RKHS norms are given by $(\|f_0\|_{\tau}, \|f_1 - f_0\|_{\tau}, \|f_2 - f_1\|_{\tau}) = (62.5, 22.5, 3.125) \times 10^{-3}$, and the variance by $(V_0, V_1, V_2) = (1.305, 0.088, 0.002) \times 10^{-3}$. The number of evaluations at each level for different budget constraints are shown in Table 1. These levels correspond to a very coarse finite element mesh, a moderately fine finite element mesh and a fine finite element mesh, respectively. The lengthscales were optimised using L-BFGS. For the illustration example shown in Figure 2, we used the same approximation for f_0 and f_1 as in the Poisson equation example. However, we used a more accurate approximation of f_2 ($C_2 = 168 \times 10^{-3}$ seconds) and all approximations were multiplied by 7 to make the difference between different levels more significant. For the illustration example, the number of evaluations at level 0, 1 and 2 was 16, 11 and 3 respectively. For BQ, we used 4 evaluations of f_2 . We still used a fixed grid to pick the quadrature point in the illustration example.

Table 1: Number of evaluations at level l given budget constraint T .

	T	$l = 0$	$l = 1$	$l = 2$
n_l^{MLBQ}	0.376s	38	15	3
	0.751s	77	30	5
	1.503s	153	60	10
n_l^{MLMC}	0.376s	67	11	1
	0.751s	133	23	2
	1.503s	266	46	3

Additional Results The number of evaluations at each level for different budget constraints are visualized in the left-hand side plot of Figure 6. The right-hand plot shows the empirical mean and 95% confidence interval of the absolute errors obtained with MLBQ and MLMC with IID points when we keep or remove the third level $\Pi[f_2 - f_1]$. Benefiting from the fast convergence rate of BQ, the improvement of adding a few points in high-fidelity level (level 2) is significant for MLBQ but not for MLMC, especially when the budget constraint T is small.

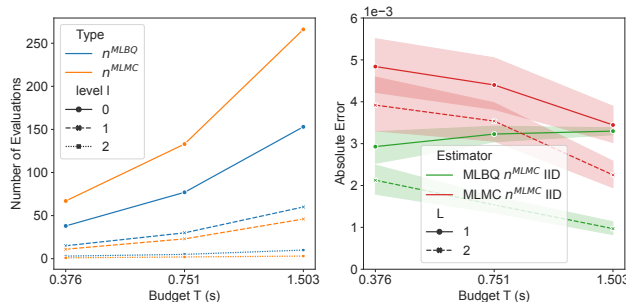


Figure 6: Poisson Equation. *Left*: n^{MLMC} and n^{MLBQ} given budget constraint T . *Right*: Absolute integration error with different L .

Comparison to Multilevel Bayesian Quadrature with Separable Kernel We compare MLMC, MLBQ with a-priori independent $f_l - f_{l-1}$ (MLBQ), and MLBQ with separable kernels (SK-MLBQ) (as in Appendix B.3). We compare three different separable kernels, B_1, B_2, B_3 , where $(B_1)_{i,i} = (B_2)_{i,i} = (B_3)_{i,i} = 1$, and $(B_1)_{i,j} = 0.01$, $(B_2)_{i,j} = 0.05$ and $(B_3)_{i,j} = 0.1$, for $i \neq j, i, j \in \{0, \dots, L\}$. The computational cost of using SK-MLBQ depends on the budget constraint and the number of samples at each level. When $T = 1.503s$, the computational cost of using SK-MLBQ with n^{MLMC} is 0.374s, which is around 1.4 times that of using MLBQ with n^{MLMC} (0.268s). The ratio will increase if we employ a larger budget constraint and use n^{MLBQ} . Figure 7 visualizes the result of 100 repetitions of the experiment, where for each repetition, we evaluated f_0, \dots, f_L at new point sets, and used the same dataset for MLBQ, MLMC and SK-MLBQ to estimate $\Pi[f]$.

The figure reveals that when a low cross-level correlation is established by using separable kernel B_1 , the performance of SK-MLBQ improves slightly on that of MLBQ. However, as the specified cross-level correlation increases, the performance of SK-MLBQ deteriorates and MLBQ outperforms SK-MLBQ when using B_2 and B_3 . Overall, this experiment shows that SK-MLBQ raises computational costs and does not ensure a significant reduction in error.

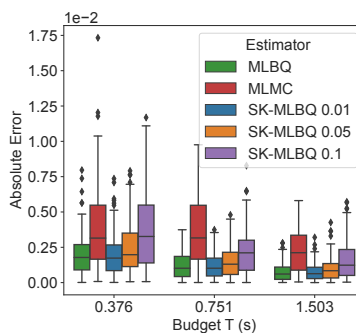


Figure 7: Poisson Equation. Absolute integration error.

C.2 Experiment 2: ODE with Random Coefficient and Forcing

Construction of the Levels We now provide details on the solver (finite difference approximation). We first expand the equation:

$$\frac{d}{dx} \left(c(x) \frac{du}{dx} \right) = -50^2 \omega_2^2 \Leftrightarrow \omega_1 \frac{du}{dx} + (1 + \omega_1 x) \frac{d^2 u}{dx^2} = 50 \omega_2^2$$

for $x \in (0, 1)$. Let $u(x_i) = u(ih)$ for $i \in \{1, \dots, (1-h)/h\}$ with $u(0) = u(1) = 0$, we will approximate the left-hand side of the equation above using a finite difference approximation with spacing $h > 0$:

$$\begin{aligned} \omega_1 \frac{u(x_i) - u(x_{i-h})}{h} + (1 + \omega_1 x_i) \frac{u(x_i+h) - 2u(x_i) + u(x_{i-h}))}{h^2} &= 50\omega_2^2 \\ \omega_1 \frac{u(x_i) - u(x_{i-1}))}{h} + (1 + \omega_1 ih) \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{h^2} &= 50\omega_2^2 \\ \omega_1 \frac{i u((i+1)h) - (2i-1)u(ih) + (i-1)u((i-1)h)}{h} + \frac{u((i+1)h) - 2u(ih) + u((i-1)h)}{h^2} &= 50\omega_2^2 \end{aligned}$$

Then, bringing the random coefficient and the random forcing into consideration, the approximation at level l is

$$f_l(\omega) = \sum_{i=1}^{1/h_l-1} u(ih_l, \omega),$$

where $u_l = (u(h_l, \omega), u(2h_l, \omega), \dots, u(1-h_l, \omega))^\top$ can be solved from the linear system $(\omega_1 Q_l/h_l + L_l/h_l^2)u_l = 50\omega_2^2 \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^{(1-h_l)/h_l}$ is a vector of ones, $Q_l \in \mathbb{R}^{(1-h_l)/h_l \times (1-h_l)/h_l}$ is a tridiagonal stiffness matrix with

$$(Q_l)_{i,i} = -2i + 1, \quad (Q_l)_{i,i-1} = (Q_l)_{i-1,i} = i - 1,$$

and $L_l \in \mathbb{R}^{(1-h_l)/h_l \times (1-h_l)/h_l}$ is a tridiagonal stiffness matrix with

$$(L_l)_{i,i} = -2, \quad (L_l)_{i,i-1} = (L_l)_{i-1,i} = 1.$$

Experimental Setting Table 2 lists the number of evaluations at each level for multilevel estimators and BQ under different budget constraints. Three levels correspond to a very coarse ODE solver, a moderately fine ODE solver and a fine ODE solver, respectively. In this experiment, we used tensor product Matérn kernel with smoothness $\nu = 2.5$ and squared exponential kernel. The related analytical formulae are provided in Appendix C.6, and the Adam optimiser was used to select lengthscales. It is worth mentioning that the closed form of the initial error of the Matérn kernel with respect to Gaussian distributed random variables doesn't exist. Since the closed form equation of the kernel mean in this case is known, we can estimate the initial error very precisely and efficiently with MC estimator by using a large number of IID samples from the Gaussian distribution.

Table 2: Number of evaluations for multilevel estimators and BQ given budget constraint T .

T	$l = 0$	$l = 1$	$l = 2$	BQ
0.303s	166	27	3	15
1.517s	830	135	15	75
30.347s	16579	2701	308	/
151.736s	82984	13505	1538	/

Additional Results Figure 8 shows the empirical mean and 95% confidence interval of the absolute errors obtained with MLBQ and MLMC with IID points when we keep or remove the third level $\Pi[f_2 - f_1]$. Benefiting from the fast convergence rate of BQ, the improvement of adding a few points in high-fidelity level (level 2) is significant in small budget cases (budget $T = 0.303$ s) for MLBQ but not for MLMC.

C.3 Experiment 3: Landslide-Generated Tsunami

Construction of the Levels To model the tsunami wave, Volna-OP2 (Giles et al., 2020) numerically solves the nonlinear shallow water equations:

$$\frac{\partial H}{\partial t} + \nabla \cdot (H\vec{v}) = 0, \quad \frac{\partial H\vec{v}}{\partial t} + \nabla \cdot (H\vec{v} \otimes \vec{v} + \frac{g}{2} H^2 I_2) = gH\nabla h,$$

where $\vec{v} = (u, v)$ is the depth averaged horizontal velocities, g is the acceleration due to the gravity, h is the underwater topography (bathymetry), η is the wave height measured from the sea level at rest (free surface elevation) and therefore $H = h + \eta$ is the total water depth. I_2 denotes a 2×2 identity matrix. The exact form of the time dependent bathymetry of the landslide case was introduced in Lynett and Liu (2005) but is reproduced here. The bathymetry of the sloping beach is prescribed by $h(x, t) = h_S(x, t) - x \tan(\omega_2)$, where h_S is the profile of the sliding mass

$$h_S(x, t) = \omega_1 d_o \frac{\left[1 + \tanh\left(\frac{2(x-x_l(t))}{\omega_3}\right)\right] \left[1 + \tanh\left(\frac{2(x-x_r(t))}{\omega_3}\right)\right]}{\left[1 + \tanh(\cos(\omega_2))\right] \left[1 - \tanh(-\cos(\omega_2))\right]}.$$

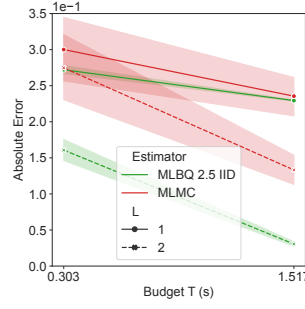


Figure 8: ODE with Random Coefficient and Forcing: Absolute integration error with different L .

The right and left boundaries of the slide are called $x_r(t)$ and $x_l(t)$ respectively and given by:

$$x_r(t) = x_c(t) + \frac{\omega_3}{2} \cos(\omega_2), \quad x_l(t) = x_c(t) - \frac{\omega_3}{2} \cos(\omega_2),$$

where x_c is the horizontal location of the center point of the slide. By assuming the initial depth of the center point of the slide to be 50m, x_c is given by: $x_c(t) = 50/\tan(\omega_2) + u_s t$, where u_s is the time-dependent velocity of the submerged landslide and can be calculated by

$$u_s = u_t \tan\left(\frac{t}{t_0}\right), \quad u_t = \sqrt{g\omega_3 \frac{\pi}{2} \sin(\omega_2)}, \quad t_0 = u_t \frac{2}{g \sin(\omega_2)}.$$

Volna-OP2 uses a finite volume method with two dimensional meshes and thus the set up (Figure 1, left) is extended in the y direction, which is perpendicular to the page and results in translational symmetry along this axis. The domain of $y \in [-\Delta x, \Delta x]$, where Δx is the spatial resolution. A representative example of the solution as given by Volna-OP2 for $\vec{\omega} = (0.375, 10^\circ, 150 \text{ m})$ is presented in Figure 9. The sub figures showcase transects of the bathymetry and surface elevation at various time points.

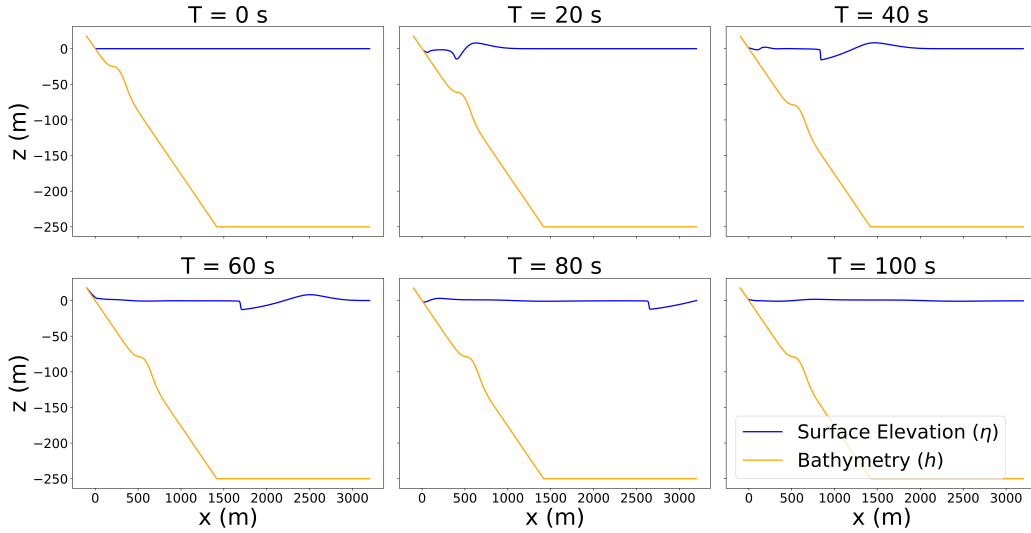


Figure 9: Snapshots of the bathymetry and the solution of the PDE, surface evaluation $\eta(x, t, \omega)$ at various timestamps as given by Volna-OP2 when $\vec{\omega} = (0.375, 10^\circ, 150 \text{ m})$.

Experimental Design The number of evaluations at each level for different budget constraint is listed in Table 3. The lengthscales were optimised using the Adam optimiser.

C.4 Experiment 4: Linear Function

In this section, we explore the impact of breaking our theoretical assumptions A1-A6 through a linear function example. Assumption A1, A2, A6 could be generalised or replaced without affecting the convergence rate e.g. assuming bounded

Table 3: Number of evaluations at level l given budget constraint T .

T	$l = 0$	$l = 1$	$l = 2$	$l = 3$	$l = 4$
1200s	32	16	8	4	2
6000s	160	80	40	20	10
12000s	320	160	80	40	20

domain with Lipschitz boundary, and satisfying an interior cone condition (Wynne et al., 2021) and assuming non-zero prior means (Teckentrup, 2020). Assumption A3-A5 are more crucial for obtaining the desired convergence rate. Assumption A3 is satisfied by using Matérn kernels. Assumption A4 depends on the smoothness of f_l . If f_l is infinitely differentiable, a convergence rate of MLBQ with squared exponential kernel can be derived by generalising Theorem 2.20 of Karvonen (2019). Assumption A5 ensures that we use a dataset that covers the domain well, which is important to the convergence rate of MLBQ.

The integrand we consider is the following linear function,

$$f(\omega) = \omega \text{ for } \omega \in [0, 10].$$

The integral of interest is $\Pi[f] = 0.1 \int_0^{10} f(\omega) d\omega$, so Π is a $\text{Unif}(0, 10)$. We use step functions f_0, f_1 and f_2 as our approximations to f . Let $p_l \in \mathbb{N}$ and $0 = \omega_{l,1} < \omega_{l,2} < \dots < \omega_{l,p_l} = 10$, then for $i = 2, \dots, p_l$

$$f_l(\omega) = \begin{cases} \frac{\omega_{l,i-1} + \omega_{l,i}}{2} & \text{if } \omega \in [\omega_{l,i-1}, \omega_{l,i}), \\ \frac{\omega_{l,p_l-1} + 10}{2} & \text{if } \omega = 10. \end{cases}$$

We compare different settings to assess the impact of breaking our assumptions. Assumption A1, A2, A4 and A6 are upheld for all settings in this example. We compare two different kernels, the Matérn 0.5 kernel and the squared exponential kernel, which violates Assumption A3. Assumption A5 is violated by utilizing IID points and a bad experimental design, in which 90% of the points are sampled from $\text{Unif}(0, 5)$, and the remaining 10% are sampled from $\text{Unif}(5, 10)$. We use n^{MLMC} for all settings in this example. Sample sizes are given in Table 4.

 Table 4: Number of evaluations at level l given budget constraint T .

T	$l = 0$	$l = 1$	$l = 2$
0.002s	37	8	2
0.004s	74	15	4

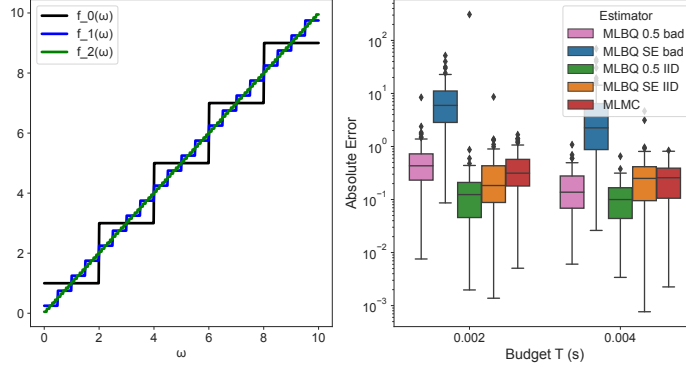
Figure 10 visualizes the approximations f_0, f_1 and f_2 and the results of 100 repetitions of the experiment. The right-hand side plot shows that MLBQ with Matérn 0.5 kernels and IID points significantly outperforms the others. When the budget is small, MLBQ with squared exponential kernels and IID points exhibits a slight advantage over MLMC. However, when the budget is increased, their performance becomes close. When the budget is small, MLMC demonstrate a slight superiority over MLBQ with Matérn 0.5 kernels and the bad experimental design, but when the budget is larger, MLBQ with Matérn 0.5 kernels and the bad experimental design performs slightly better than MLMC. MLMC always outperforms MLBQ with squared exponential kernels and the bad experimental design. This example highlights the substantial impact that the choice of kernels and experimental design have on the performance of MLBQ.

C.5 Experiment 5: Model Evidence

A reviewer recommended approximating model evidence integrals based on likelihoods using datasets of different sizes. The reviewer mentioned that multilevel methods could help here, but unfortunately our initial attempts at using MLBQ in this context did not provide convincing results. This problem might therefore warrant some more efforts in future work.

C.6 Analytical Formulae for the Kernel Means and Initial Errors

As mentioned in the main text, BQ algorithms usually require integrals of the kernel in closed-form. In this section, we provide all of the closed-form formulae used in the paper.


 Figure 10: Linear Function. *Left*: The approximations to f . *Right*: Absolute integration error.

Matérn Covariance Function with Smoothness $v = 1/2$ When Π is a uniform distribution on some interval $[a, b]$ and the covariance function is Matérn covariance function with smoothness $v = 0.5$ and length-scale γ , the kernel mean can be computed analytically as

$$\Pi[c_{1/2}(\cdot, \omega)] = \int_a^b \exp\left(-\frac{|\omega - \omega'|}{\gamma}\right) (b - a)^{-1} d\omega' = (b - a)^{-1} \left(2\gamma - \exp\left(\frac{a - \omega}{\gamma}\right)\gamma - \exp\left(\frac{-b + \omega}{\gamma}\right)\gamma\right),$$

and the initial error can be computed analytically as

$$\Pi\Pi[c_{1/2}(\cdot, \cdot)] = \int_a^b \int_a^b \exp\left(-\frac{|\omega - \omega'|}{\gamma}\right) (b - a)^{-1} d\omega' (b - a)^{-1} d\omega = 2\gamma \left(b - a - \gamma + \exp\left(\frac{a - b}{\gamma}\right)\gamma\right) (b - a)^{-2}.$$

Matérn Covariance Function with Smoothness $v = 5/2$ When Π is a standard Gaussian distribution and the covariance function is Matérn covariance function with smoothness $v = 2.5$ and length-scale γ , the kernel mean can be computed analytically as

$$\begin{aligned} \Pi[c_{5/2}(\cdot, \omega)] &= \int_{-\infty}^{+\infty} \left(1 + \frac{\sqrt{5}\|\omega - \omega'\|_2}{\gamma} + \frac{5\|\omega - \omega'\|_2^2}{3\gamma^2}\right) \exp\left(-\frac{\sqrt{5}\|\omega - \omega'\|_2}{\gamma}\right) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\omega'^2\right) d\omega' \\ &= \exp\left(-\frac{\omega^2}{2}\right) \left(4\sqrt{5}\gamma(-5 + 3\gamma^2) + \sqrt{2\pi} \left(\exp\left(\frac{(\sqrt{5} - \gamma\omega)^2}{2\gamma^2}\right)(25 + 3\gamma^4 - 10\sqrt{5}\gamma\omega\right.\right. \\ &\quad \left.\left.+ 3\sqrt{5}\gamma^3\omega + 5\gamma^2(-2 + \omega^2))\operatorname{erfc}\left(\frac{\frac{\sqrt{5} - \omega}{\gamma}}{\sqrt{2}}\right) + \right.\right. \\ &\quad \left.\left.\exp\left(\frac{(\sqrt{5} + \gamma\omega)^2}{2\gamma^2}\right)(25 + 3\gamma^4 + 10\sqrt{5}\gamma\omega - 3\sqrt{5}\gamma^3\omega + 5\gamma^2(-2 + \omega^2))\operatorname{erfc}\left(\frac{\frac{\sqrt{5} + \omega}{\gamma}}{\sqrt{2}}\right)\right)\right) / (6\gamma^4\sqrt{2\pi}) \end{aligned}$$

but the initial error cannot be computed analytically.

When Π is a uniform distribution on some interval $[a, b]$ and the covariance function is Matérn covariance function with smoothness $v = 2.5$ and length-scale γ , the kernel mean can be computed analytically as

$$\begin{aligned} \Pi[c_{5/2}(\cdot, \omega)] &= \int_a^b \left(1 + \frac{\sqrt{5}\|\omega - \omega'\|_2}{\gamma} + \frac{5\|\omega - \omega'\|_2^2}{3\gamma^2}\right) \exp\left(-\frac{\sqrt{5}\|\omega - \omega'\|_2}{\gamma}\right) (b - a)^{-1} d\omega' \\ &= (b - a)^{-1} \left(16\sqrt{5}\gamma - \exp\left(\frac{\sqrt{5}(a - \omega)}{\gamma}\right) \left(\frac{\sqrt{5}(8\gamma^2 + 5(a - \omega)^2)}{\gamma} + 25(\omega - a)\right)\right. \\ &\quad \left.+ \exp\left(\frac{\sqrt{5}(-b + \omega)}{\gamma}\right) \left(-\frac{\sqrt{5}(8\gamma^2 + 5(b - \omega)^2)}{\gamma} + 25(-b + \omega)\right)\right) / 15. \end{aligned}$$

and the initial error can be computed analytically as

$$\begin{aligned} \Pi\Pi[c_{5/2}(\cdot, \cdot)] &= \int_a^b \int_a^b \left(1 + \frac{\sqrt{5}\|\omega - \omega'\|_2}{\gamma} + \frac{5\|\omega - \omega'\|_2^2}{3\gamma^2}\right) \exp\left(-\frac{\sqrt{5}\|\omega - \omega'\|_2}{\gamma}\right) (b - a)^{-1} d\omega' (b - a)^{-1} d\omega \\ &= 2(b - a)^{-2} \left(8\sqrt{5}(b - a)\gamma - 15\gamma^2 + \exp\left(-\frac{\sqrt{5}(b - a)}{\gamma}\right) (5(b - a)^2 + 7\sqrt{5}(b - a)\gamma + 15\gamma^2)\right) / 15. \end{aligned}$$

Squared Exponential Covariance Function When Π is a uniform distribution on some interval $[a, b]$ and the covariance function is squared exponential with length-scale γ , the kernel mean can be computed analytically as

$$\begin{aligned}\Pi[c_{\text{SE}}(\cdot, \omega)] &= \int_a^b \exp\left(-\frac{(\omega-\omega')^2}{\gamma^2}\right) (b-a)^{-1} d\omega' \\ &= (b-a)^{-1} \sqrt{\pi} \gamma (\text{erf}(\frac{\omega-a}{\gamma}) + \text{erf}(\frac{b-\omega}{\gamma}))/2,\end{aligned}$$

and the initial error can be computed analytically as

$$\begin{aligned}\text{III}[c_{\text{SE}}(\cdot, \cdot)] &= \int_a^b \int_a^b \exp\left(-\frac{(\omega-\omega')^2}{\gamma^2}\right) (b-a)^{-1} d\omega' (b-a)^{-1} d\omega \\ &= \gamma \left(\left(-1 + \exp\left(-\frac{(a-b)^2}{\gamma^2}\right)\right) \gamma + (a-b) \sqrt{\pi} \text{erf}\left(\frac{a-b}{\gamma}\right) \right) (b-a)^{-2}.\end{aligned}$$

When Π is a standard Gaussian distribution and the covariance function is squared exponential covariance function with length-scale γ , the kernel mean can be computed analytically as

$$\begin{aligned}\Pi[c_{\text{SE}}(\cdot, \omega)] &= \int_{-\infty}^{+\infty} \exp\left(-\frac{(\omega-\omega')^2}{\gamma^2}\right) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\omega'^2\right) d\omega' \\ &= \gamma \exp\left(-\frac{\omega^2}{\gamma^2+2}\right) (\gamma^2+2)^{-1/2},\end{aligned}$$

and the initial error can be computed analytically as

$$\begin{aligned}\text{III}[c_{\text{SE}}(\cdot, \cdot)] &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \exp\left(-\frac{(\omega-\omega')^2}{\gamma^2}\right) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\omega'^2\right) d\omega' \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\omega^2\right) d\omega \\ &= \gamma(\gamma^2+4)^{-1/2}.\end{aligned}$$