*International Journal for Uncertainty Quantification*, x(x): 1–18 (2017)

# CONTROL VARIATES WITH A DIMENSION REDUCED-BAYESIAN MONTE CARLO SAMPLER

*Xin Cai,[1] Junda Xiong,[1] Hongqiao Wang,[2,*] & Jinglai Li[3]*

[1]*School of Mathematical Sciences , Shanghai Jiao Tong University, 800 Dongchuan Rd, Shanghai 200240, China*

[2]*School of Mathematics and Statistics, Central South University, Changsha 410083,China*

[3]*School of Mathematics, University of Birmingham, Birmingham B15 2TT, UK*

*Address all correspondence to: Hongqiao Wang, E-mail: Hongqiao.Wang@csu.edu.cn

*Original Manuscript Submitted: mm/dd/yyyy; Final Draft Received: mm/dd/yyyy*

*Evaluating expectations of random functions is an important task in many fields of science and engineering. In practice, such an expectation is often evaluated with the Monte Carlo methods that approximate the sought expectation with sample average. It is well known that the Monte Carlo methods typically suffer from slow convergence, making it especially unfriendly to problems where generating samples requires expensive computer simulations. An alternative to the standard Monte Carlo method is the so-called Bayesian Monte Carlo (BMC) algorithm, which formulates the expectation estimation as a Bayesian inference problem. As has been demonstrated in literature, BMC is more efficient than standard Monte Carlo in a range of problems. However, a major limitation of BMC is that it models the integrand function as a Gaussian process which can not handle problems of high dimension. In this work, we propose a method to address this issue. Specifically, we incorporate the BMC framework with a likelihood-based dimension reduction technique, which allows us to evaluate the expectation of functions with very high dimension. Then based on the dimension-reduced BMC, we propose a control variate scheme for problems where a direct use of dimension-reduced BMC is not sufficiently accurate. We verify the effectiveness of the proposed method with both mathematical and practical examples.*

**KEY WORDS:** *Bayesian Monte Carlo, Control Variate, Dimension Reduction, Likelihood-Acquired Direction, Monte Carlo, Gaussian Process, Numerical Integration*

## 1. INTRODUCTION

Evaluating the expectation of a function with random inputs, mathematically cast as a numerical integration problem, has vast applications in many fields of science and engineering such as data science, signal processing, uncertainty quantification and reliability engineering. For example, the Expectation Maximization (EM) algorithm [1] is often used to determine hyperparamters in Bayesian inference. In each iteration of the EM algorithm, the expectation of the log-likelihood function needs to be evaluated. In general, such an expectation can be evaluated via Monte Carlo (MC) simulations [2,3]. A well-known issue with the MC methods is that they suffer from a very slow rate of convergence, which is reciprocal to the square root of sample size [3]. Variance reduction techniques such as importance sampling and control variate are often used to reduce the estimator variance and improve the accuracy of MC simulation.

Alternatively, another very interesting class of methods is developed based on the idea of evaluating the integral in a Bayesian inference framework [4,5]. The methods, known as Bayesian Monte Carlo (BMC), cast the integrand function as a random function (more precisely a Gaussian Process) and subsequently the expectation becomes stochastic. Then Bayesian inference is conducted to obtain the posterior of the expectation based on the data obtained by querying the integrand function at a number of different locations. In this framework, the posterior distribution, which is Gaussian, can be computed analytically [4,5]; the posterior mean provides an estimate of the sought expectation and

2152–5080/17/$35.00  

the posterior variance quantifies the uncertainty in the estimate. It has been shown in [4,6–10] that BMC can achieve better performance than MC type of methods in a wide range of problems.

A major limitation of BMC is that since the method relies on the Gaussian Process (GP) model, it usually can not handle problems of high dimension [11]. While in many realistic problems, the integrand function often (approximately) possesses a low dimensional structure: namely the function dominantly depends on a low dimensional projection of the high dimensional input [12]. In that case, a very popular practice is to identify the low dimension subspace of the problem and construct surrogate models in this subspace, see [13–18].To this end, the so-called sufficient dimension reduction methods are often used to reduce the dimensionality of the problem. Further details of such methods are provided in Section 3 and we recommend the overview [19] for readers interested in this topic. In this work, by assuming a low dimensional structure of the integrand function, we incorporate BMC with a likelihood-based sufficient dimension reduction technique [20], which enables BMC to deal with high dimensional problems. Specifically, the method consists of two steps, first identifying a low-dimensional subspace, then converting the original integration problem into the subspace and estimating it with BMC. We note here that similar ideas have been used to construct GP models [13–17] for various uncertainty quantification problems, but they are not in the BMC context. A major difference here is that with the GP model used as an emulator, BMC can analytically compute an estimator (as well as its variance) of the sought integral.

Even with the assistance of dimension reduction (DR) techniques, the performance of BMC is limited in two aspects. First, in many practical problems, low dimensional structure only exists approximately, which results in approximation error in BMC. This approximation error can not be reduced by increasing sample size. Second, BMC can not handle arbitrary increase of the size of training dataset, as that will lead to inversion of a very large matrix. That means even for problems where a large amount of data is available, BMC may not be able to fully take advantage of the data. Due to these limitations, BMC estimation is unavoidably subject to approximation error or bias, regardless of the number of data points available. To correct such approximation error, we employ a control variate scheme that can provide an unbiased estimator like standard MC and potentially reduce the estimator variance. The control variate scheme presented here is similar to that used in [21], except that the authors of [21] use polynomial chaos expansion (PCE) as surrogate model. The main advantage of BMC is that the integrals in BMC framework can be computed analytically, while with PCE one has to compute the integrals numerically. Finally, we emphasize that with the control variate scheme we can obtain an unbiased estimator of the integral of interest.

The rest of the paper is organized as follows. In Section 2 we describe the basic setup of the expectation evaluation problem and then briefly introduce the BMC method for such problems. In Section 3, we introduce a likelihood-based dimension reduction technique, then the dimension-reduced BMC algorithm, and the control variate method to improve the performance of DR-BMC. We then test the proposed method with some mathematical functions in Section 4 and apply it to practical examples in Section 5. Finally, in Section 6 we provide some concluding remarks.

## 2. THE BAYESIAN MONTE CARLO METHOD

We consider a generic integration problem. Let $x \in \mathbb{R}^d$ be a random variable whose probability density is $p(x)$, and $f(x)$ be a real-valued function defined on $\mathbb{R}^d$. Suppose that we are interested in the expectation of $f(x)$, expressed as the integral:

$$\bar{f} = \int f(x)p(x)dx. \tag{1}$$

Without loss of generality we can assume that $p(x)$ is a multivariate Gaussian distribution $N(\mu_x, \Sigma_x)$. If $p(x)$ is not Gaussian, we can choose a Gaussian distribution and use it to conduct standard importance sampling to convert the problem into this desired form. That is, we rewrite the integral as

$$\bar{f} = \int \frac{f(x)p(x)}{q(x)} q(x)dx,$$

where $q(x)$ is a Gaussian distribution density and $\frac{f(x)p(x)}{q(x)}$ now becomes the integrand function.

A popular way to evaluate the integral $\bar{f}$ is to use the MC estimator:

$$\hat{f} = \sum_{n=1}^{N} f(x^{(n)}) \tag{2}$$

where $x^{(1)}, ..., x^{(N)}$ are i.i.d. samples drawn from distribution $p(x)$. As is well known, the MC estimator suffers a rather slow convergence rate and often yields highly inaccurate estimate of the integral unless one is able to use an exceedingly large number of samples.

Bayesian MC is an alternative to MC for evaluating the integral. The key idea of BMC is to turn the integral evaluation into a Bayesian inference problem. To do this, first we cast the quantity of interest $\bar{f}$ as a random variable, treating $f(x)$ as a random field. Namely, the function $f(x)$, essentially an unknown function, becomes "stochastic", as its uncertain characterization due to limited knowledge is captured in a probabilistic manner. Note that the quantity of interested $\bar{f}$ is a function of $f(x)$ and it follows directly that $\bar{f}$ is random. In a Bayesian framework, a prior distribution is imposed on $f$, and with observations of $f$ we obtain the posterior of $f$. Then the posterior distribution of $\bar{f}$ is also obtained.

The key of the BMC method lies in the probabilistic model imposed on the unknown function $f(x)$. A popular formulation is to assume that $f$ is a Gaussian Process. Under the GP assumption, the joint distribution of any (finite) number of function values (indexed by inputs, $x$) is Gaussian. In particular, we can assume that the prior mean is $\mu(x)$ and the covariance is specified by a kernel function $k(x, x')$:

$$\mathrm{Cov}[f(x), f(x')] = k(x, x'). \tag{3}$$

We note here that in general a variety of kernels can be used, but for the sake of convenience (which will be explained later), we choose the popular squared exponential kernel in the following form [11]:

$$k(x, x') = w_0 \exp(-\frac{1}{2}\sum_{i=1}^{d}(x_i - x_i')^2/w_d^2), \tag{4}$$

where $w_0, ..., w_d$ are hyperparameters. Given the dataset $\mathcal{D} = \{(x^{(n)}, y^{(n)} = f(x^{(n)})\}_{n=1}^{N}$, the GP model allows one to estimate the unknown function $f$ in a Bayesian framework [11].

In particular, one can derive directly the posterior distribution* $\pi(f|\mathcal{D})$, also being Gaussian, whose mean is

$$\tilde{f}_{\mathcal{D}}(x) = \mu(x) + k(x, X)k(X, X)^{-1}(Y - \mu(X)), \tag{5}$$

and the covariance is

$$\mathrm{Cov}[f(x)|\mathcal{D}] = k_{\mathcal{D}}(x, x) = k(x, x) - k(x, X)k(X, X)^{-1}k(X, x). \tag{6}$$

Here $Y = \left[y^{(1)}, \ldots, y^{(N)}\right]$, $X = \left[x^{(1)}, \ldots, x^{(N)}\right]$, and the notation $k(A, B)$ denotes the matrix of covariance evaluated at all pairs of points in set $A$ and in set $B$ using the kernel function $k(\cdot, \cdot)$. Several technical issues of GP construction such as how to determine the hyperparameters are left out of this paper, and for more such details of this method we refer readers to [11,22].

Now we return to the integration problem: given a set of samples $\mathcal{D} = \{(x^{(n)}, y^{(n)} = f(x^{(n)})\}_{n=1}^{N}$, we want to estimate the value of $\bar{f}$. In a Bayesian setting, that is to compute the posterior $\pi(f|\mathcal{D})$, and under GP model the posterior $p(\bar{f}|\mathcal{D})$ is also Gaussian, which is fully characterized by its mean and variance [11]. Moreover, through some elementary calculus, one can derive the posterior mean of $\bar{f}$,

$$\begin{aligned}
\bar{f}_{\mathcal{D}} = E_{f|\mathcal{D}}[\bar{f}] &= \int\int f(x)p(x)\mathrm{d}x\mu(df|\mathcal{D}) \\
&= \int \left[\int f(x)p(f|\mathcal{D})\mathrm{d}f\right] p(x)\mathrm{d}x \\
&= \int \tilde{f}_{\mathcal{D}}(x)p(x)\mathrm{d}x,
\end{aligned} \tag{7}$$

*We note that in what follows for conciseness we abuse the notation to treat $f$ as a finite dimensional random variable.

where $\tilde{f}_{\mathcal{D}}$ is the posterior mean function. Similarly, the posterior variance of $\bar{f}$ is

$$\begin{aligned} V_{f|\mathcal{D}}[\bar{f}] &= \int \left[ \int f(x)p(x)\mathrm{d}x - \int \bar{f}(x')p(x')\mathrm{d}x' \right]^2 p(f|\mathcal{D})\mathrm{d}f \\ &= \int\int k_{\mathcal{D}}(x,x')p(x)p(x')\mathrm{d}x\mathrm{d}x', \end{aligned} \tag{8}$$

where $k_{\mathcal{D}}$ is the posterior covariance of $f$. Now recall that we have assumed that the density $p(x)$ is Gaussian, and the prior covariance is the kernel given in Eq. (4). In detail, if $p(x) = \mathcal{N}(\mu_x, \Sigma_x)$ and the Gaussian kernels on the data points are $\mathcal{N}(a_n = x^{(n)}, W = diag(w_1^2, \ldots, w_d^2))$, $a = [a_1, \ldots, a_N]^{\mathrm{T}}$, then the expectation evaluates to

$$E_{f|\mathcal{D}}[\bar{f}] = E[\mu(\mathrm{x})] + z^T k(X,X)^{-1}(Y - \mu(X)), \tag{9a}$$

$$z = w_0 |W^{-1}\Sigma_x + I|^{-1/2} \exp[-\frac{1}{2}(a-\mu_x)^T (W+\Sigma_x)^{-1}(a-\mu_x)], \tag{9b}$$

a result derived under the name of Bayes-Hermite Quadrature[4]. For the variance, we get

$$V_{f|\mathcal{D}}[\bar{f}] = w_0 |2W^{-1}\Sigma_x + I|^{-1/2} - z^T K^{-1} z, \tag{10}$$

with $z$ defined in Eq. (9b).

## 3. THE DIMENSION-REDUCED BMC METHOD

### 3.1 Conducting BMC in a low dimensional subspace

A key assumption of dimension-reduced BMC is that there is a low dimensional subspace of $\mathbb{R}^d$ such that $f(x)$ only depends on the projection of $x$ in this low dimensional subspace. If we can identify this low dimensional subspace, we can conduct BMC to compute the integration only in the subspace. Mathematically, we assume that there exists a $r \times d$ matrix $A$ with $r \ll d$ such that

$$f(x) = f_r(\tilde{x}) \quad \text{with} \quad \tilde{x} = Ax. \tag{11}$$

Under this assumption, it is easy to derive that

$$\bar{f} = \int f(x)p(x)dx = \int f_r(\tilde{x})p_A(\tilde{x})d\tilde{x}, \tag{12}$$

where $p_A(\cdot) = \mathcal{N}(A\mu_x, A\Sigma_x A^T)$. Then one can conduct BMC with respect to $\tilde{x}$ in the $\mathbb{R}^r$ space, using Eq. (12). It should be noted here that usually to conduct either BMC or MC, we should be able to evaluate $f_r(\tilde{x})$ for any given $\tilde{x}$, which is not possible in our setting. However, we can generate a set of samples $x^{(1)}, ...., x^{(N)}$ from distribution $p(x)$ and construct the dataset

$$\mathcal{D}_A = \{(\tilde{x}^{(n)} = Ax^{(n)}, y^{(n)} = f_r(\tilde{x}^{(n)}) = f(x^{(n)})\}_{n=1}^N.$$

It is easy to verify that $\tilde{x}^{(1)}, ...., \tilde{x}^{(N)}$ follows the distribution $p_A(\cdot)$. For notational convenience, we use the corresponding notation $\tilde{\mathrm{x}} = A\mathrm{x}$, $\tilde{X} = AX$, $\tilde{a} = Aa$ and $\tilde{W} = \mathrm{diag}(\tilde{w}_1^2, \ldots, \tilde{w}_r^2)$. Now applying Eq. (9a), we obtain

$$E_{f|\mathcal{D}}[\bar{f}] = E[\mu(\tilde{\mathrm{x}})] + \tilde{z}^T k(\tilde{X},\tilde{X})^{-1}(Y - \mu(\tilde{X})), \tag{13a}$$

$$\tilde{z} = \tilde{w}_0 |\tilde{W}^{-1}A\Sigma_x A^T + I|^{-1/2} \exp[-\frac{1}{2}(\tilde{a}-A\mu_x)^T (\tilde{W}+A\Sigma_x A^T)^{-1}(\tilde{a}-A\mu_x)], \tag{13b}$$

which is the posterior mean of the integral of interest in the dimension-reduced situation. We note that the hyperparameters $w_0, \ldots, w_r$ of kernel function $k(\cdot,\cdot)$ and the mean function $\mu(\cdot)$ are trained on the dataset $\mathcal{D}_A$. Thus we have a dimension-reduced BMC (DR-BMC) method, which proceeds as Algorithm1.

Now the remaining question is how to identify the low dimensional subspace or equivalently the projection matrix $A$, which is discussed in the next section.

**Algorithm 1:** The dimension-reduced Bayesian Monte Carlo method

1. draw samples $x^{(1)}, ..., x^{(N)}$ from distribution $p(\cdot) = \mathcal{N}(\mu_x, \Sigma_x)$;
2. evaluate $y^{(n)} = f(x^{(n)})$ for $n = 1...N$;
3. perform a dimension reduction with dataset $\{(x^{(n)}, y^{(n)})\}_{n=1}^N$ to acquire projection matrix $A$
4. let $\tilde{x}^{(n)} = Ax^{(n)}$ and $\mathcal{D}_A = \{(\tilde{x}^{(n)}, y^{(n)})\}_{n=1}^N$;
5. let $p_A(\cdot) = \mathcal{N}(A\mu_x, A\Sigma_x A^T)$;
6. apply BMC on the transformed dataset $\mathcal{D}_A$ and prior distribution $p_A(\cdot)$ with the formulations (13).

### 3.2 Likelihood-based dimension reduction

In this section, we shall discuss how to reduce the dimensionality of the integrand function $f(x)$. Recall that we have assumed that the integrand function $y = f(x)$ can be written as

$$y = f(x) = f_r(Ax, \epsilon), \tag{14}$$

where $A$ is a $r \times d$ matrix, and $\epsilon$ is a small noise independent of $x$ (in our problem setup there is usually no observation noise, i.e. $\epsilon = 0$). It should be clear that matrix $A$ projects the $d$-dimensional random variable $x$ into a $r$-dimensional subspace of $\mathbb{R}^d$, and such a subspace is called a dimension reduction (DR) subspace. Under mild conditions, the intersection of all DR subspace is still a DR subspace. The intersection is called the central subspace and denoted by $S_{Y|X}$. Since $y$ is conditionally independent of $x$ given the projection of $x$ onto $S_{Y|X}$, the corresponding linear combination of $x$ contains all the regression information on the response.

Noting that there is a wide range of methods that can be used to identify the central subspace [19,23], here we choose likelihood-based dimension reduction methods. Such methods use model based inverse regression to identify the central subspace. More precisely, it is assumed that the conditional distribution $p(x|y)$ is Gaussian:

$$x|y \sim N(\mu_y, \Delta_y), \tag{15}$$

where $\mu_y$ and $\Delta_y$ are the mean and the variance of the Gaussian distribution depending on $y$. The likelihood-based methods impose different structures (or models) on $\mu_y$ and $\Delta_y$, and then identify the projection matrix $A$ via a maximum likelihood estimation (MLE). The actual likelihood function depends on the model adopted and in the present work we use the so-called likelihood-acquired direction (LAD) [20]. In our setup, the response variable $y$ is continuous, thus it is typical to partition its range into $H$ slices, $I_1, \ldots, I_H$, following the practice of SIR and SAVE [24]. We assume that the data consists of $n_h$ independent observations in interval $I_h, h = 1, \ldots, H$. Next, let $\tilde{\Delta}_h$ denote the sample covariance matrix of data $x_i^{(n)}$ whose corresponding $y^{(n)}$ falls in the interval $I_h$. Let $\tilde{\Delta} = \sum_{h=1}^H f_h \tilde{\Delta}_h$, where $f_h$ is the fraction of cases observed in $I_h$. Without going through the derivation, we proceed directly to the log-likelihood function in the LAD model, which reads [20]

$$l(A) = \frac{n}{2} \log |A\Sigma_x A^T|_0 - \frac{n}{2} \log |\Sigma_x| - \frac{1}{2} \sum_{h=1}^H n_h \log |A\tilde{\Delta}_h A^T|_0, \tag{16}$$

where $|A|_0$ indicates the product of the nonzero eigenvalues of a positive semidefinite symmetric matrix $A$. Then the projection matrix $A$ can be obtained by maximizing Eq. (16). As it is usually impossible to find a closed-form solution to $\max l(A)$, we use numerical optimisation technique for the problem. When the likelihood is accurate, LAD inherits properties from general likelihood theory [20]. In this work, we employ the software package LDR provided in [23] to conduct LAD-based dimension reduction.

Finally, an important question is how to determine the dimensionality $r$. As is suggested in [20], the value of $r$ can be estimated using likelihood ratio testing or an information criterion like AIC or BIC, and conditional independence hypotheses involving the predictors can be tested in a straightforward manner. We leave out the details here and encourage readers to consult [20,25]. In the numerical experiments in this paper, the dimensionality $r$ is automatically determined by the package [23].

### 3.3 Control variate

It should be clear that in most practical problems low dimension structure only exits approximately, posing a main limitation to the accuracy of DR-BMC. To further improve the accuracy of DR-BMC, we introduce a control variate scheme which is similar to [26] where the surrogate models are built with polynomial chaos expansion. We start with a basic introduction to the control variate (CV) method. Note that for any $f^*(x)$ and $\bar{f}^*$ satisfying

$$\bar{f}^* = \int f^*(x)p(x)dx,$$

we can re-write Eq. (1) as

$$\bar{f} = \int f(x)p(x)dx = \int \left(f(x) - c(f^*(x) - \bar{f}^*)\right)p(x)dx = \int \left(f(x) - cf^*(x)\right)p(x)dx + c\bar{f}^*, \quad (17)$$

where $c$ is an arbitrary constant. Now suppose that $\bar{f}^*$ is known in advance, and we define a new function

$$f_{\mathrm{CV}} = (f(x) - cf^*(x)) + c\bar{f}^*. \quad (18)$$

Thus we can estimate $\bar{f}$ as

$$\hat{f}_{\mathrm{CV}} = \frac{1}{M}\sum_{m=1}^{M} f_{\mathrm{CV}}(x^{(m)}), \quad (19)$$

where $x^{(1)}, ..., x^{(M)}$ are drawn from $p(x)$. It is easy to derive that the optimal value of $c$ that minimizes the variance of $\hat{f}_{\mathrm{CV}}$ is

$$c^* = \frac{\mathrm{Cov}(f, f^*)}{\mathrm{Var}(f)} = \rho(f, f^*). \quad (20)$$

As we see, $\rho(f, f^*)$ is the correlation coefficient between $f$ and $f^*$, and the resulting minimum variance is

$$\mathrm{Var}(\hat{f}_{\mathrm{CV}}) = (1 - \rho^2(f, f^*))\mathrm{Var}(f).$$

It is easy to see here that the larger $|\rho(f, f^*)|$, the smaller the estimator variance is. Now the key issue is to find a good choice of $f^*(x)$, and that is where BMC enters the method. To be specific, we shall assume that the GP model constructed in the previous sections is not exactly equal to $f(x)$, but is nonetheless a rather good approximation that is highly correlated to $f(x)$. Under this assumption we can derive $c^* \approx 1$ and the variance of $\hat{f}_{CV}$ is much smaller than that of MC estimator. Next we shall discuss how to derive CV estimator with BMC. We now assume that we have $N + M$ samples, and first all the samples are used to compute the low dimensional subspace. We then split the whole dataset into two: set $\mathcal{D}$ with $N$ samples that are used for conducting BMC, then set $\mathcal{D}'$ with $M$ samples used for CV. Let $\tilde{f}_{\mathcal{D}}$ be the mean function of the GP model constructed from dataset $\mathcal{D}$ and $\bar{f}_{\mathcal{D}}$ be the DR-BMC mean value of the integral $\bar{f}$, then the BMC-CV estimator is

$$\hat{f}_{CV} = \frac{1}{M}\sum_{x^{(m)} \in \mathcal{D}'} (f(x^{(m)}) - \tilde{f}_{\mathcal{D}}(x^{(m)})) + \bar{f}_{\mathcal{D}}.$$

We reinstate that a major advantage of the BMC-CV method is that $\hat{f}_{CV}$ is an unbiased estimator of $\bar{f}$, which means the mean square error decays to zero as the number of samples increases. Finally, for computational efficiency, it is often important to decide in advance whether the CV correction should be used. One possible strategy is to use the variance of the estimator provided by DR-BMC as an indicator,i.e. if the variance exceeds a prescribed threshold value, then the CV correction is implemented.

## 4. MATHEMATICAL EXAMPLES

We apply the proposed method to several numerical examples in this section. In each example we compare the performance of the proposed DR-BMC method with MC. We emphasize that the dimensionality of the examples considered here are typically beyond the capacity of standard BMC.

### 4.1 Tests for DR-BMC

Specifically, we consider several mathematical functions with explicit expressions so that we can validate the integration results with exact solutions. In this example, we compute the integral

$$E[f] = \int f(x)p(x)\mathrm{d}x,$$

where $p(x)$ is a standard Gaussian distribution density function. We choose the following four different integrand functions $f(x)$ to test our method:

$$f_1(x) = 2\sin(1.4x_1) + (x_2 + 10)^2, \tag{21a}$$

$$f_2(x) = \frac{10x_1}{1 + \exp(\frac{x_2+2x_3+2x_4}{3})}, \tag{21b}$$

$$f_3(x) = \frac{10x_1}{0.5 + (0.1x_2 + 1.5)^2}, \tag{21c}$$

$$f_4(x) = x_1 + \exp(\frac{x_2}{2}), \tag{21d}$$

where $x = (x_1, ..., x_d)$ with $x_i$ denoting the $i$-th dimension of the $d$-dimensional vector $x$. These functions are often used benchmark functions for dimension reduction methods with easy-to-identify low dimension structures. For all these functions, we consider two situations: the total dimensionality of $x$ is 50 or 100, i.e. $d = 50$ or $d = 100$, and we shall reinstate that both cases are beyond the capacity of standard BMC. Moreover, it should be clear that the intrinsic dimensionality of all these four functions is 2. Taking function $f_2$ as an example, one can see from its expression that the function value only depends on the two directions $(\beta_1 x, \beta_2 x)$, where $\beta_1 = (1, 0, \ldots, 0)$ and $\beta_2 = (0, 1, 2, 2, 0, \ldots, 0)$. For comparison purpose, we estimate the integration with three different methods: MC, DR-BMC and BMC performed over the exact low dimensional subspace (referred to as BMC-LD). It should be clear that BMC-LD should perform better than DR-BMC since the BMC is conducted over the exact low dimensional subspace, eliminating the error induced by imperfect dimension reduction.

In all the numerical tests here, we use four different sample sizes: $N = 300$, 500, 1000, 2000 for $d = 50$ and three sample sizes $N = 600$, 1000, 2000 for $d = 100$. All these evaluations are repeated 100 times to obtain the median and the interquartile range(IQR) between the 25% and the 75% percentiles of the absolute error across test runs. The performance of three methods are compared in Fig. 1, where median and IQR of the absolute error for the four test functions are plotted against sample size. We note here that the IQR in BMC-LD is so narrow that it is nearly invisible in the plots.

First, we note that in all the test cases both BMC-LD and DR-BMC yield better estimation results than standard MC, suggesting that the BMC method can significantly improve the estimation accuracy for functions with low dimensional structure. On the other hand, the results of BMC-LD yields lower estimation error than the results of DR-BMC, which is not surprising because BMC-LD utilizes the exact low dimension structure directly while DR-BMC suffers from dimension reduction error. Nevertheless, in all but one cases ($f_4$ with $d = 100$), as the sample size increases to 2000, the absolute error results of DR-BMC decrease to approximately the same level as those of BMC-LD. The IQR results show that the DR-BMC estimator has a narrower IQR interval than MC, implying more stable estimation. We have also tested the cases with standard BMC on original space to estimate integrals and all produced completely erroneous results, so we omit those results.

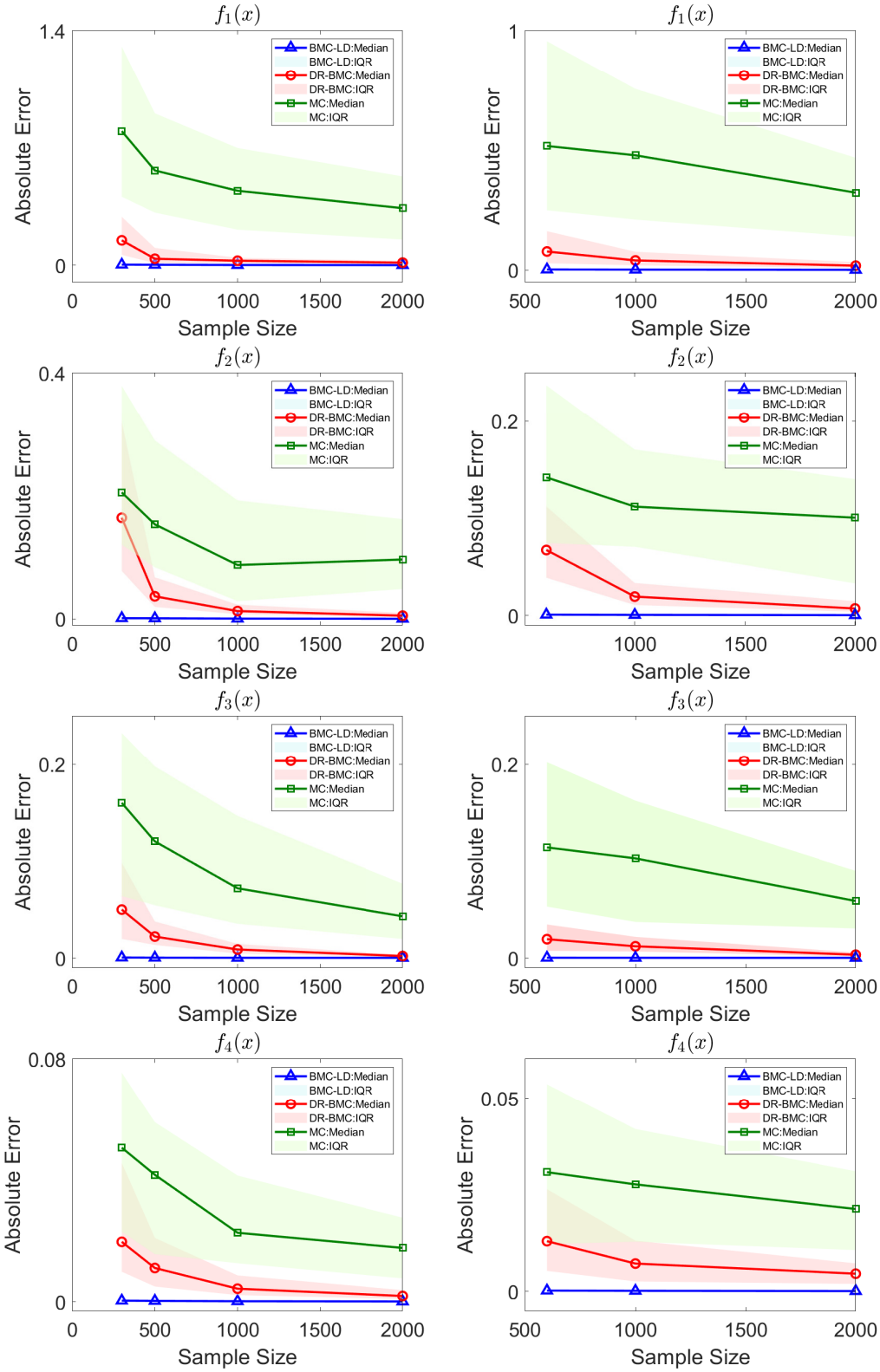


**FIG. 1:** Median and IQR of the absolute error in different methods plotted against sample size for functions in Example 4.1. The figures on the left are the results of 50 dimensions and those on the right are the results of 100 dimensions.

### 4.2 Tests for BMC-CV

As is discussed earlier, most practical problems do not have an exact low dimensional structure, which implies that the estimates of DR-BMC will be surely subject to approximation error. In this section, we shall consider such cases and show how the BMC-CV scheme can be applied to such problems. To mimic such behavior (i.e. the low-dimensional structure is only approximate), we add a high dimensional perturbation to the four test functions in (21) which reads

$$g(x) = \kappa \sum_{i=1}^{d} x_i^2, \tag{22}$$

where $d = 50$ and $\kappa = 0.1$ in all of our numerical tests in this section. Clearly, the function $g(x)$ does not have a linear low-dimensional structure and $E[g(x)] = 5$. As in Section 4.1, we want to evaluate $E[f_i(x) + g(x)]$ for $i = 1, \ldots, 4$.

As mentioned earlier, we shall apply BMC-CV scheme to this problem and compare its results with those of standard MC. In the numerical experiments, we first use 1000 samples to construct a GP model, and then substitute it into the CV estimator (19) to evaluate the expectation of interest. In the CV estimator (19), several different sample sizes are used, ranging from 500 to 2000. As a comparison, we also conduct standard MC with the same samples used in CV estimator and construction of GP. We then repeat the simulations 50 times for each method and calculate the median and IQR of the absolute error as before. We plot the results of the CV estimator as a function of CV sample size in Figs. 2 and the results of MC with the same samples are also presented. One can see from the plots that the CV estimator yields substantially lower estimation error than the standard MC estimator. This example therefore demonstrates that the BMC-CV scheme can be a useful variance reduction technique for problems with no exact low-dimensional structure.

## 5. APPLICATIONS TO PRACTICAL PROBLEMS

In this section, we consider several examples where the underlying models are computationally intensive.

### 5.1 A random ODE example

The first practical example we consider is a simple random ODE problem in [27] with Gaussian random input

$$\frac{df}{dt} = -k(t)f(t) + g(t), f(1) = 1, t \in [1, 5], \tag{23}$$

where $g(t) = 3\sin(t) - \frac{1}{4}$ is a deterministic function and $k(t)$ is a Gaussian random function

$$k(t) \sim \mathcal{GP}(\mu(t), \Sigma(t, t)). \tag{24}$$

The mean function and covariance kernel are

$$\mu(t) = t^2 - t - 3, \quad C(t_1, t_2) = \frac{1}{4}\exp(-(t_1 - t_2)^2). \tag{25}$$

Next, we represent $k(t)$ by a truncated Karhunen-Loève (KL) expansion

$$k(t) \approx k^d(t, \boldsymbol{x}) = \bar{k}(t) + \sum_{i=1}^{d} \sqrt{\lambda_i} x_i a_i(t), \tag{26}$$

where $\bar{k}(t) = \mu(t)$, the coefficients $x_i$'s follow the standard Gaussian distribution, and $\{\lambda_i, e_i\}$ are the eigenpairs of the kernel function. In the numerical tests, we take $d = 50$, thus this is a 50 dimensional problem.

Now suppose that we are interested in the following expectations at $t = 1.5$ and $t = 3$:

$$E[f] = \int f(t, x)p(x)\mathrm{d}x, \tag{27}$$

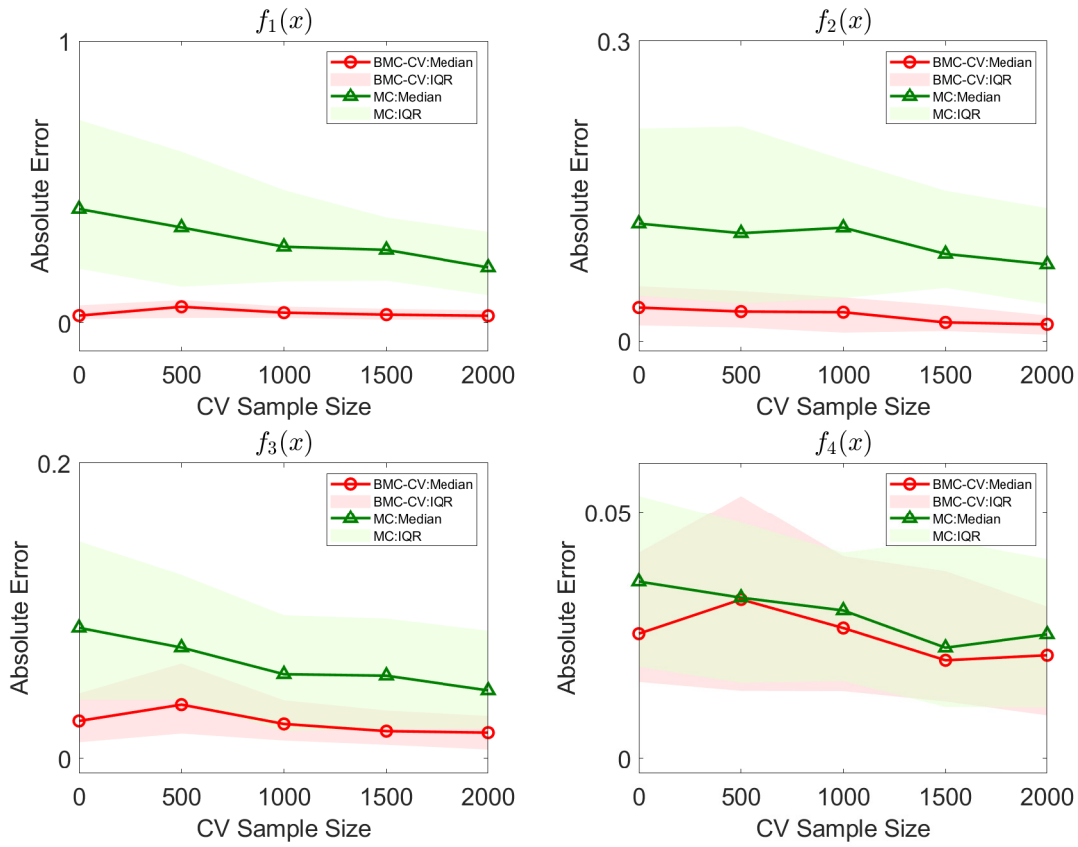


**FIG. 2:** Median and IQR of the absolute error in BMC-CV and MC plotted against sample size for functions in Example 4.2.

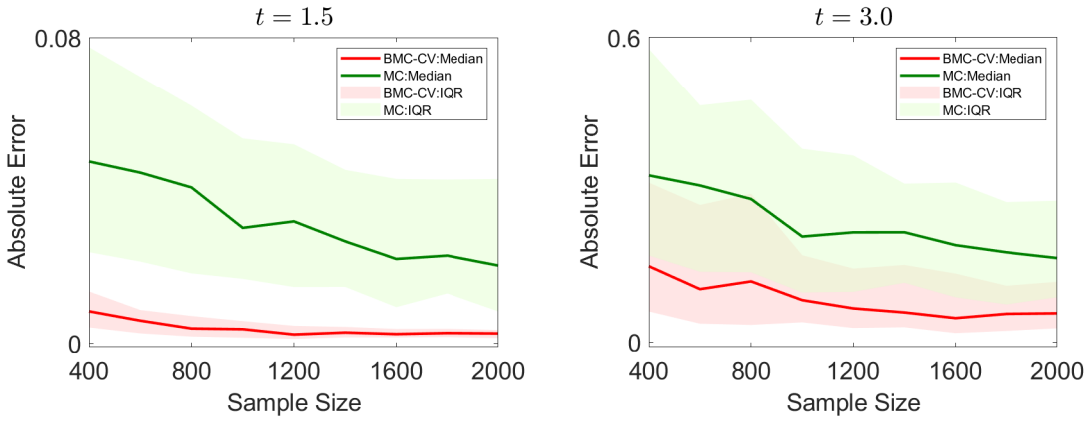


**FIG. 3:** Median and IQR of the absolute error in BMC-CV and MC in Example 5.1.

where $p(x)$ is the density function of $\mathcal{N}(0, I_{50})$. First, we use $2\times10^5$ crude MC samples to generate benchmark values, which are 6.4471 and 11.0794 respectively. Then we compare the performance of BMC-CV and MC as in Example 4.2, where the total sample sizes are 400, 600, ..., 2000. In the BMC-CV tests, we use the whole dataset to perform dimension reduction, then the first 200 samples to compute BMC estimator, and the rest to compute CV estimator. We repeat the simulations 100 times, and for each method we calculate the median and IQR of the absolute error across the test runs. From Figs. 3 one can see that the performance largely agree with the mathematical examples, where the CV estimator yields substantially lower estimation error than the MC estimator, indicating that the proposed method performs well in this example.

### 5.2 An elliptic PDE example

Our second practical example is the following elliptic partial differential equation studied in [12]:

$$-\nabla_s \cdot (a(s)\nabla_s u) = 1,\ s \in [0,1]^2. \tag{28}$$

We set homogeneous Dirichlet boundary conditions on the left, top, and bottom of the spatial domain and denote these boundaries by $\Gamma_1$. The right side of the spatial domain denoted as $\Gamma_2$ has a homogeneous Neumann boundary condition. That is,

$$\begin{aligned} u(s) &= 0,\ s \in \Gamma_1, \\ \nabla u(s) \cdot \mathbf{n} &= 0,\ s \in \Gamma_2, \end{aligned} \tag{29}$$

where $\mathbf{n}$ is the unit normal vector to the boundary. In this problem, we assume that the coefficient $a = a(s,x)$ of the differential operator is a log-Gaussian random field with zero-mean and covariance kernel

$$C(s,s') = \exp(-\|s-s'\|_1/\sigma), \tag{30}$$

where $\sigma = 1$. Next, we represent $a(\mathbf{s})$ by a truncated KL expansion:

$$\log(a(s)) = \sum_{i=1}^{d} x_i \gamma_i \phi_i(s), \tag{31}$$

where the $x_i$'s follow the standard Gaussian distribution, and $\{\phi_i, \gamma_i^2\}$ are the eigenpairs of the kernel (30). In the numerical tests, we take $d = 100$. Our function of interest is a linear functional of the solution [12]

$$f(x) = \int_{\Gamma_2} u(s,x)/|\Gamma_2| ds, \tag{32}$$

and we want to compute the expectation $E[f(x)]$.

In the numerical experiments, we employ BMC-CV and MC to estimate $E[f(x)]$ with different sample sizes: 1000, 1500, 2000, 2500 and 3000. We use $5 \times 10^5$ crude MC samples to generate a benchmark value of the integral, which is $4.94 \times 10^{-4}$. In BMC-CV, the first 1000 samples are used to construct GP model (for example, if the total sample size is 1500, 1000 samples are used to construct GP model and remaining 500 samples are used in CV estimation). Each estimation is repeated 30 times, and since both CV and MC estimators are unbiased, we evaluate their performance by examining the estimator variances or equivalently their standard deviations across the runs. In Fig. 4 (left), we plot the standard deviations of both estimators against sample size. From the figure, we can see that the BMC-CV method yields much lower STD than MC for all sample sizes, indicating that the CV approach can significantly reduce estimator variance. The effectiveness of the BMC-CV method can be further demonstrated by visualizing the estimate results. In Fig. 4 (right), we compare the histograms of the estimate results of two methods. One can see that the results of MC are subject to much larger variation/uncertainty than those of BMC-CV, suggesting that BMC-CV is an effective variance reduction technique for this example.

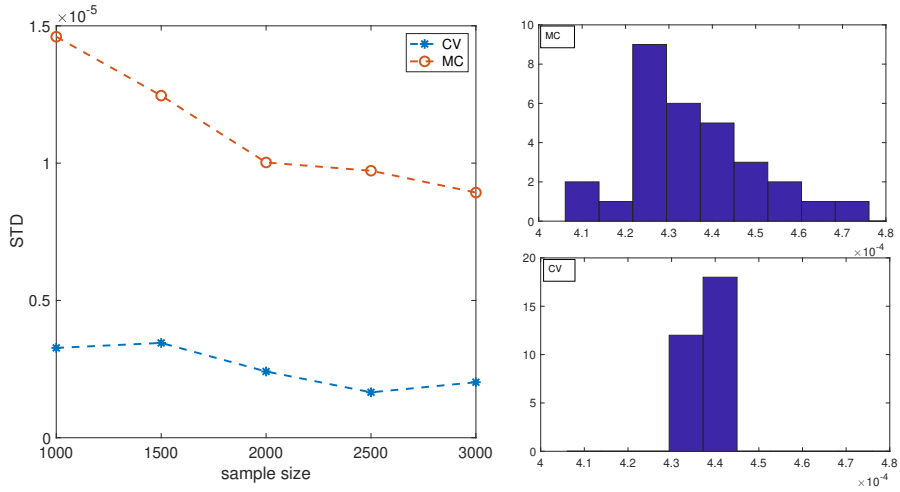


**FIG. 4:** The error comparison of Example 5.2. Left: the standard deviation of the absolute error in BMC-CV and MC plotted against sample size. Right: the histograms of the estimate results for sample size 1500; the top figure is the plot for MC and the bottom is for CV.

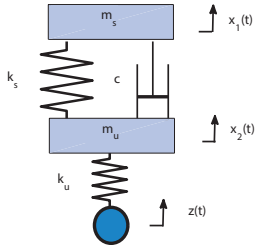


**FIG. 5:** The quarter car model

**TABLE 1:** Example 5.3: the parameter values of the quarter car model.

| $m_s$ | $m_u$ | $k_s$ | $k_u$ | c | σ | $l_c$ |
|---|---|---|---|---|---|---|
| 20 | 40 | 400 | 2000 | 600 | 5 | 1 |

### 5.3 The quarter car model

In this section, we consider the quarter car for vehicle suspension system [28]. The schematic illustrating of the model is shown in Fig 5, where the sprung mass $m_s$ and the unsprung mass $m_u$ are connected by a nonlinear spring and a linear damper. The stiffness of the nonlinear spring is $k_s$ and the damping coefficient of the linear damper is $c$. The displacement of the wheel $z(t)$ represents the interaction of the quarter car system with the terrain. Mathematically, the model is described by a two-degree-of-freedom ODE system [28]:

$$\begin{aligned} m_s\frac{d^2x_1}{dt^2} &= -k_s(x_1-x_2)^3 - c(\frac{dx_1}{dt}-\frac{dx_2}{dt}),\\ m_u\frac{d^2x_2}{dt^2} &= k_s(x_1-x_2)^3 + c(\frac{dx_1}{dt}-\frac{dx_2}{dt}) + k_u(z(t)-x_2), \end{aligned} \tag{33}$$

where $x_1$ and $x_2$ are the displacements of the sprung and the unsprung masses respectively. In our example, we assume that uncertainty in the system arises from the random road profile. The wheel displacement $z(t)$ is modeled as a Gaussian random field, with mean $\mu(t) = 10t\sin(\pi/6)$ and covariance kernel $C(t_1,t_2) = \sigma^2\exp(-(t_1-t_2)^2/l_c)$. As in Example 5.1, We represent $z(t)$ with KL expansion (26), thus the random parameter is of 50 dimensions. Other model parameters are all taken to be fixed and the values are shown in Table 1. The initial conditions of Eqs. (33) are

$$x_1(0) = \frac{dx_1}{dt}(0) = 0, \quad x_2(0) = \frac{dx_2}{dt}(0) = 0.$$

Eqs. (33) are numerically solved with the classical Runge-Kutta method. In numerical simulations, the quantity of interest is the expectation of the difference between displacements of the sprung and the unsprung springs at $t = 0.2$ and $t = 0.4$ :

$$I = \int (x_1(t,w) - x_2(t,w))p(w)dw, \tag{34}$$

where $p(w)$ is the density function of $\mathcal{N}(0, I_{50})$.

As is in example 5.1, here we use $2 \times 10^5$ crude MC samples to calculate benchmark values of the integrals, which are $-0.0789$ and $-0.2257$ respectively. We compare the median and IQR of the absolute error across 100 runs in the same way as example 5.1 and show the results in Figs.6. We can see from the figure that for both methods the absolute error clearly decreases as the sample size grows, while BMC-CV yields much lower estimation error than MC for all sample sizes. In addition, BMC-CV estimation also yields much narrower IQR than MC for all sample sizes, indicating that BMC-CV is more statistically stable than the MC method.

### 5.4 An empirical Bayesian inference example

In this section, we discuss a potential application of the proposed method to Bayesian inverse problems. We start with a brief introduction to Bayesian inverse problem. In such problems, our goal is to estimate certain parameters of interest from indirect, incomplete and noisy data. More specifically, let the parameter of interest be $x$ and the observed data be $w$, then one can estimate $x$ from $w$ using the Bayes' formula [8,29]:

$$p(x|w) = \frac{p(w|x)p(x)}{p(w)}, \tag{35}$$

where $p(x)$ is prior distribution, $p(w|x)$ is likelihood and $p(x|w)$ is posterior distribution. In reality, the prior and/or the likelihood may depend on some hyperparameters that are not known in advance. In this case, the hyperparameters

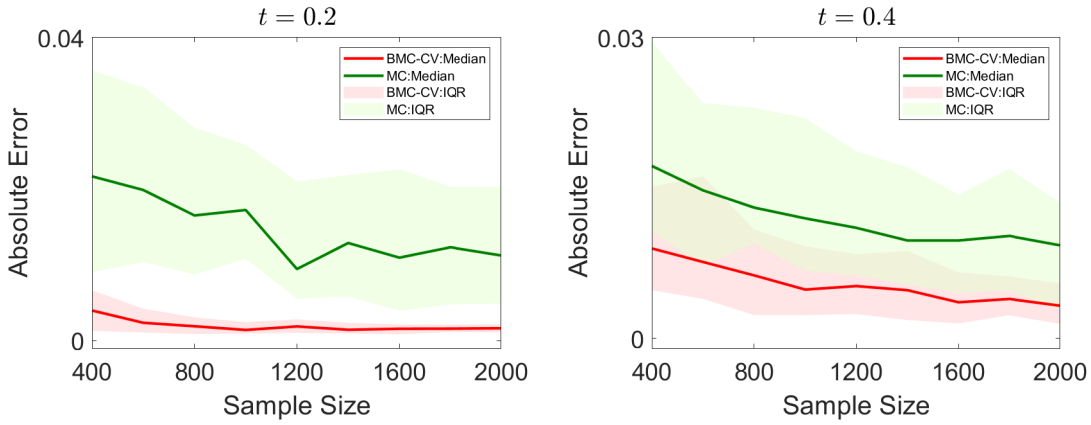


**FIG. 6:** Median and IQR of the absolute error in BMC-CV and MC in Example 5.3.

denoted as $\alpha$ are determined by maximizing the marginal likelihood function

$$\max_{\alpha} p(w|\alpha) = \int p(w|x,\alpha)p(x|\alpha)dx, \tag{36}$$

and such a method is usually referred to as empirical Bayes approach. As is mentioned earlier, the EM algorithm is often used to solve the optimization problem (36). Simply put, the EM algorithm iteratively updates the hyperparamter $\alpha$ as following:

$$\alpha_k = \arg\min_{\alpha} l(\alpha|\alpha_{k-1}), \tag{37a}$$

$$l(\alpha|\alpha_{k-1}) = -\int \log p(w|x,\alpha)p(x|w,\alpha_{k-1})dx. \tag{37b}$$

As we can see from Eq. (36), a critical step in the EM algorithm is to evaluate the integral in Eq. (37b), which in many real-world applications can be of very high dimension.

The specific example that we consider here is a nonlinear inverse heat conduction problem studied in [30]. The model in this example is a one-dimensional heat conduction equation

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left[c(u)\frac{\partial u}{\partial x}\right], \tag{38}$$

where $x$ and $t$ are spatial and temporal variable, $u(x,t)$ is temperature, and $c(u)$ is temperature dependent thermal conductivity, all in dimensionless units. The equation is subject to initial condition $u(x,0) = u_o(x)$ and a Neumann boundary condition:

$$\frac{\partial}{\partial x}u(0,t) = q(t), \tag{39a}$$

$$\frac{\partial}{\partial x}u(L,t) = 0, \tag{39b}$$

where $L$ is the length of the medium. Namely, one end ($x = L$) is insulated and the other ($x = 0$) has heat flux. Suppose that we place a temperature sensor at location $x = x_s$, and our goal is to infer the heat flux $q(t)$ for $t \in [0, T]$ from the temperature history measured by a sensor in the time interval. The schematic of this problem is shown in Fig. 7.

In the simulation, we let $L = 1$, $T = 1$, and the initial condition $u_o(x) = 0$. We parametrize the flux function with a Fourier series:

$$q(t) = a_o + \sum_{j=1}^{N_f}(a_j \cos(2j\pi t/T) + b_j \sin(2j\pi t/T)), \qquad 0 \le t \le T, \tag{40}$$

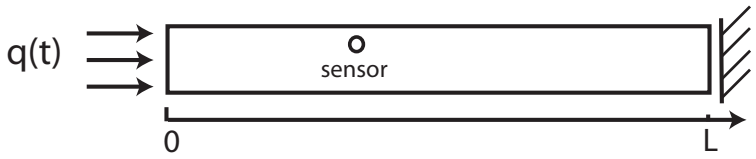


**FIG. 7:** Schematic diagram of the one dimensional heat conduction problem.

where $a_j$ and $b_j$ are the coefficients of the cosine and the sine components respectively, and $N_f$ is the total number of Fourier modes. We fix the number of Fourier modes $N_f = 20$, then the total dimensionality of unknown parameter is 41. The prior distribution of these coefficients are assumed to be the standard Gaussian $\mathcal{N}(0, 1)$. The sensor is placed at $x_s = 0.4$, and the temperature is measured at $t_s = 0.2,\ 0.4,\ 0.6,\ 0.8$. The error in each measurement is assumed to be an independent zero-mean Gaussian random variable with standard deviation σ. The "true" flux is

$$q_{\rm true}(t) = 0.5 + 0.5 \sum_{j=1}^{N_f} (\cos(2j\pi t) + \sin(2j\pi t)) \,, \tag{41}$$

i.e. $a_o = 0$, and $a_j = b_j = 0.5$ for $j = 1, ..., N_f$. In our numerical test, we use synthetic data generated with noise standard deviation $\sigma_{\rm true} = 0.1$. We assume that the noise variance is not known in advance and needs to be determined according to the data. As has been discussed above, we want to evaluate the integral $l(\sigma|\sigma_0)$ defined in Eq. (37b), where $\sigma_0 = 1$. When evaluating $l(\sigma|\sigma_0)$, we use the Markov Chain Monte Carlo method to draw samples from the conditional distribution $p(x|w, \sigma_0)$, and we choose one sample in every 20 iterations to reduce the correlation between samples.

We apply BMC-CV and MC to compute integral $l(\sigma|\sigma_0)$ with different values of σ. For both methods, we test two sample sizes: 3000 and 10000. In BMC-CV 1000 samples are used to construct GP model, and 2000 or 9000 samples are used in the CV estimation. Each estimation is repeated 20 times and the mean and the STD of the estimation results are calculated. In Figs. 8 we plot the integral $l(\sigma|\sigma_0)$ as a function of σ. In particular, the top two figures are the results for the simple size 3000 and the bottom ones for the sample size 10000. The figures on the left are computed with MC and those on the right are computed with BMC-CV. In each plot, the solid line plots the mean of the estimates and the shaded region represents the standard deviation across the 20 runs. First, as one can see, the mean of the two methods are largely close to each other, which is not surprising as both MC and CV are unbiased. While for both sample sizes, the CV method yields significantly smaller standard deviations than MC, suggesting that BMC-CV can effectively reduce estimation variance and improve the performance of EM algorithm for determining hyperparameters.

Finally, to further compare the two methods, we conduct full EM iteration with both methods. In particular, the maximization is conducted with the Nelder-Mead method, while the expectation calculation is done by using standard MC or the proposed BMC-CV method. For both methods, EM iteration is terminated after 100 steps. The experiment is repeated 100 times and the results are recorded. On the other hand, we conduct EM using standard MC with $5 \times 10^5$ samples in each Expectation step, and obtain an estimate $\sigma_{\rm opt} = 0.14$ as the true optimal value. We compare the results of both MC and BMC-CV against the true optimal value. In particular, we look at two situations: if an EM result $\hat{\sigma}$ satisfies $|\hat{\sigma} - \sigma_{\rm opt}|/\sigma_{\rm opt} \leq 0.3$, we regard this EM trial as a very successful one, and if an EM result $\hat{\sigma}$ satisfies $|\hat{\sigma} - \sigma_{\rm opt}|/\sigma_{\rm opt} \geq 1$, we regard it as a failed trial. Note here that there are also cases that are neither very successful or failed (i.e. if $0.3 < |\hat{\sigma} - \sigma_{\rm opt}| < 1$). The successful rate is an often used measure for benchmarking optimization algorithms [31]. In Figs. 9 (left) we plot the number of the very successful trials of the following cases: MC with 3000 samples, BMC-CV with 3000 samples, MC with 10000 samples, and BMC-CV with 10000 samples, while Figs. 9 (right) shows the number of the failed trials with the same set of samples. As one can see from the figures, the BMC-CV methods provides evidently more very successful trials and less failed trials than MC for both sample sizes, suggesting that the use of BMC-CV can considerably improve the performance of EM compared to standard MC integration.

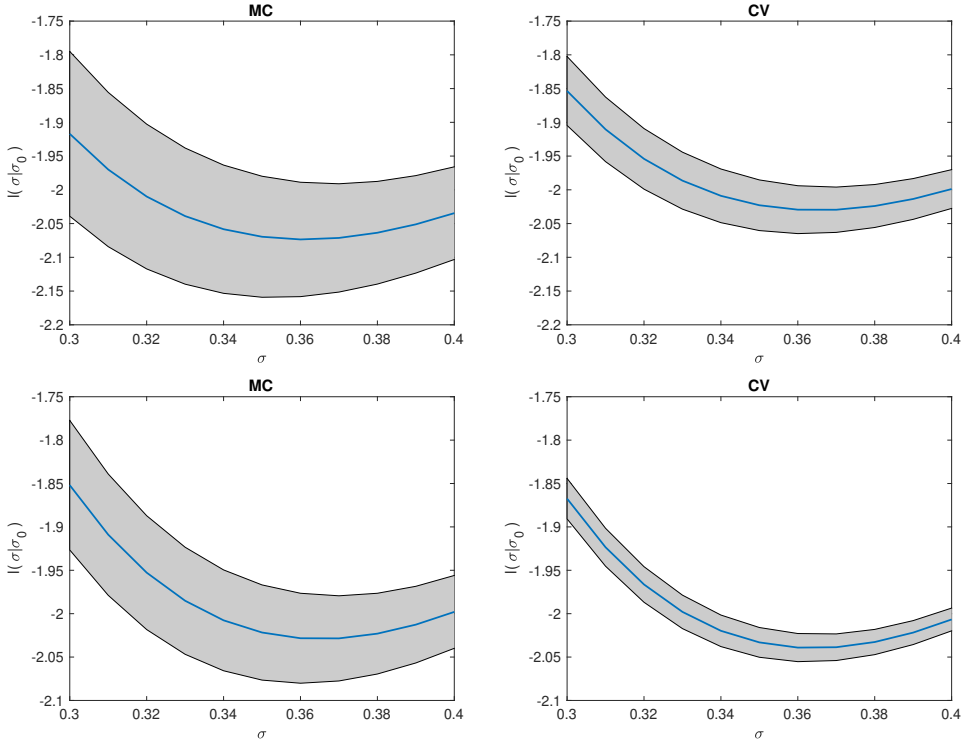


**FIG. 8:** The function $l(\sigma|\sigma_0)$ plotted against σ. The solid lines are the mean of the estimates and the shaded region shows the standard deviation of the estimation across test runs . The left and right rows are computed with MC and CV respectively; the figures on the top are the results with sample size 3000 and those at the bottom are with sample size 10000.

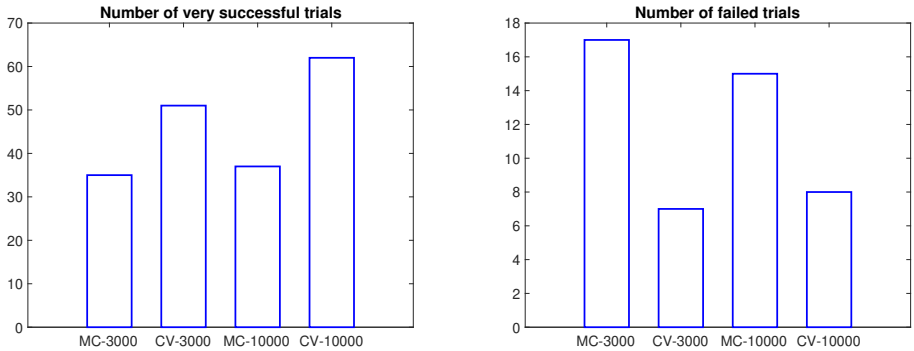


**FIG. 9:** The performance of EM algorithm with the BMC-CV and MC methods across 100 runs.

## 6. CONCLUSION

In summary, in this work we have proposed a method for conducting numerical integration. In particular, we adopt Bayesian Monte Carlo framework that treats the integrand function as a random field. To address the dimensionality issue, we propose to first use dimension reduction techniques (in particular the LAD algorithm) to identify the low dimensional subspace and then implement Bayesian Monte Carlo in the reduced subspace. As a result, the Dimension-Reduced BMC can deal with numerical integration problems of high dimension. Finally, we incorporate DR-BMC model in a CV framework to resolve the issue of approximation errors in both dimension reduction and GP model, yielding an unbiased estimator. To verify the effectiveness of the proposed method, we apply it to several mathematical functions and practical problems. We expect that the method can be useful in a class of problems that are similar to the examples tested in this work.

We also note that some further improvements and extensions of the proposed method are possible. For example, in many practical problems, the gradients of integrand functions are often available, and in this case, it is possible to use gradient information to improve the accuracy of dimension reduction as well as Bayesian Monte Carlo estimation. Moreover, another interesting feature of the Bayesian Monte Carlo method is that it can also estimate the variance of the integral of interest, which can be highly useful if one is interested in quantifying the uncertainty of the estimate. To this end, it is an interesting problem to understand how much uncertainty is introduced by the dimension reduction procedure to the results. We plan to investigate these issues in future studies.

## REFERENCES


1. Dempster, A.P., Laird, N.M., and Rubin, D.B., Maximum Likelihood from Incomplete Data via the EM Algorithm, Journal of the Royal Statistical Society: Series B (Methodological), 39(1):1–38, 1977.
2. Levine, R.A. and Casella, G., Implementations of the Monte Carlo EM Algorithm, Journal of Computational and Graphical Statistics, 10(3):422–439, 2001.
3. Robert, C. and Casella, G., Monte Carlo Statistical Methods, Springer, NY, 2010.
4. O'Hagan, A., Bayes-Hermite Quadrature, Journal of Statistical Planning and Inference, 29(3):245–260, 11 1991.
5. Rasmussen, C. and Ghahramani, Z., Bayesian Monte Carlo, In 16th Annual Conference on Neural Information Processing Systems (NIPS 2002), pp. 489–496. MIT Press, 2003.
6. Osborne, M., Duvenaud, D., Garnett, R., Rasmussen, C., and Ghahramani, Z., Active Learning of Model Evidence Using Bayesian Quadrature, In Advances in Neural Information Processing Systems, Vol. 25, pp. 46–54. Curran Associates, Inc., 2012.
7. Gunter, T., Osborne, M., Garnett, R., Hennig, P., and Roberts, S., Sampling for Inference in Probabilistic Models with Fast Bayesian Quadrature, In Advances in Neural Information Processing Systems, Vol. 4, pp. 2789–2797, 11 2014.
8. Acerbi, L., Variational Bayesian Monte Carlo, In Advances in Neural Information Processing Systems, Vol. 31, pp. 8223–8233, 2018.
9. Law, A.M., Kelton, W.D., and Kelton, W.D., Simulation Modeling and Analysis, Vol. 3, McGraw-Hill, NY, 2000.
10. Briol, F.X., Oates, C.J., Girolami, M., and Osborne, M.A., Frank-Wolfe Bayesian Quadrature: Probabilistic Integration with Theoretical Guarantees, In Advances in Neural Information Processing Systems, Vol. 28, pp. 1162–1170, 2015.
11. Williams, C.K. and Rasmussen, C.E., Gaussian Processes for Machine Learning, MIT Press, MA, 2006.
12. Constantine, P.G., Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies, SIAM, PA, 2015.
13. Djolonga, J., Krause, A., and Cevher, V., High-Dimensional Gaussian Process Bandits, In Advances in Neural Information Processing Systems, pp. 1025–1033, 2013.
14. Liu, X. and Guillas, S., Dimension Reduction for Gaussian Process Emulation: An Application to the Influence of Bathymetry on Tsunami Heights, SIAM/ASA Journal on Uncertainty Quantification, 5(1):787–812, 2017.
15. Xing, W., Shah, A.A., and Nair, P.B., Reduced Dimensional Gaussian Process Emulators of Parametrized Partial Differential Equations Based on Isomap, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 471(2174):20140697–20140697, 2015.

16. Tripathy, R., Bilionis, I., and Gonzalez, M., Gaussian Processes with Built-in Dimensionality Reduction: Applications to High-Dimensional Uncertainty Propagation, Journal of Computational Physics, 321:191–223, 02 2016.
17. Xiong, J., Cai, X., and Li, J., Clustered Active-Subspace Based Local Gaussian Process Emulator for High-Dimensional and Complex Computer Models, arXiv preprint arXiv:2101.00057, 2020.
18. Yang, X., Li, W., and Tartakovsky, A., Sliced-Inverse-Regression-Aided Rotated Compressive Sensing Method for Uncertainty Quantification, SIAM/ASA Journal on Uncertainty Quantification, 6(4):1532–1554, 2018.
19. Li, B., Sufficient Dimension Reduction: Methods and Applications with R, CRC Press, Boca Raton, 2018.
20. Cook, R.D. and Forzani, L., Likelihood-based Sufficient Dimension Reduction, Journal of the American Statistical Association, 104(485):197–208, 2009.
21. Li, W., Lin, G., and Li, B., Inverse Regression-Based Uncertainty Quantification Algorithms for High-dimensional Models: Theory and Practice, Journal of Computational Physics, 321:259–278, 05 2016.
22. Williams, C.K. and Rasmussen, C.E. Gaussian Processes for Regression. In Advances in Neural Information Processing Systems, pp. 514–520. MIT, MA, 1996.
23. Cook, R.D., Forzani, L.M., and Tomassi, D.R., Ldr: A Package for Likelihood-Based Sufficient Dimension Reduction, Journal of Statistical Software, 39(3):1–20, 2011.
24. Li, K.C., Sliced Inverse Regression for Dimension Reduction, Journal of the American Statistical Association, 86(414):316–327, 1991.
25. Korner Nievergelt, F., Roth, T., von Felten, S., Guélat, J., Almasi, B., and Korner Nievergelt, P. Model Selection and Multimodel Inference. In Bayesian Data Analysis in Ecology Using Linear Models with R, BUGS, and STAN, pp. 175–196. Academic Press, Boston, 2015.
26. Ma, X. and Zabaras, N., An Adaptive High-ddimensional Stochastic Model Representation Technique for the Solution of Stochastic Partial Differential Equations, Journal of Computational Physics, 229(10):3884–3915, 2010.
27. Xiu, D. and Karniadakis, G.E., The Wiener-Askey Polynomial Chaos for Stochastic Differential Equations., SIAM Journal on Scientific Computing, 29(2):619–644, 2002.
28. Wong, J.Y., Theory of Ground Vehicles, John Wiley & Sons, NY, 2008.
29. Congdon, P., Bayesian Model Choice Based on Monte Carlo Estimates of Posterior Model Probabilities, Computational Statistics & Data Analysis, 50(2):346–357, 2006.
30. Li, J. and Marzouk, Y.M., Adaptive Construction of Surrogates for the Bayesian Solution of Inverse Problems, SIAM Journal on Scientific Computing, 36(3):A1163–A1186, 2014.
31. Beiranvand, V., Hare, W., and Lucet, Y., Best Practices for Comparing Optimization Algorithms, Optimization and Engineering, 18(4):815–848, 2017.